UNIVERSIDADE DE LISBOA
FACULDADE DE MEDICINA

Towards accreditation in metagenomics for clinical microbiology

Catarina Inês Marques de Sousa Mendes

Orientador: Doutor João André Nogueira Custódio Carriço
Co-orientador: Professor Doutor Mário Nuno Ramos de Almeida Ramirez

Tese especialmente elaborada para obtenção do grau de Doutor em Ciências e Tecnologias da Saúde, especialidade em Biologia Computacional

2023

UNIVERSIDADE DE LISBOA
FACULDADE DE MEDICINA

Towards accreditation in metagenomics for clinical microbiology

Catarina Inês Marques de Sousa Mendes

Orientador: Doutor João André Nogueira Custódio Carriço
Co-orientador: Professor Doutor Mário Nuno Ramos de Almeida Ramirez

Tese especialmente elaborada para obtenção do grau de Doutor em Ciências e Tecnologias da Saúde, especialidade em Biologia Computacional

Júri:
Presidente: Doutora Helena Maria Ramos Marques Coelho Cortez Pinto, Professor Catedrático e Presidente do Conselho Científico da Faculdade de Medicina da Universidade de Lisboa

Vogais:
- PhD Andrew John Page, Professor da *School of Computing Sciences, University of East Anglia (United Kingdom)*;
- Doutora Cátia Raquel Jesus Vaz, Professora Adjunta do Instituto Superior de Engenharia de Lisboa;
- Doutor Vitor Manuel Monteiro Borges, Investigador Auxiliar do Instituto Nacional de Saúde Doutor Ricardo Jorge;
- Doutor Francisco José Moreira Couto, Professor Associado com Agregação da Faculdade de Ciências da Universidade de Lisboa;
- Doutor João André Nogueira Custódio Carriço, Investigador Convidado do Instituto de Medicina Molecular - João Lobo Antunes, unidade de investigação associada à Faculdade de Medicina da Universidade de Lisboa (*Orientador*);
- Doutor Nuno Luís Barbosa Morais, Professor Associado Convidado da Faculdade de Medicina da Universidade de Lisboa.

Fundação para a Ciência e Tecnologia (SFRH/BD/129483/2017 e
COVID/BD/152583/2022)
2023

A impressão desta tese foi aprovada pelo Conselho Científico da Faculdade de Medicina de Lisboa em reunião de 20 de Dezembro de 2022.

"The greatest adventure is what lies ahead.
Today and tomorrow are yet to be said.
The chances, the changes are all yours to make.
The mould of your life is in your hands to break."

-J. R. R. Tolkien, The Hobbit

Acknowledgements

There are several key features necessary for the successful completion of a doctoral dissertation work: perseverance, critical spirit, passion, and a healthy dose of stubbornness. But none of these matter if a support network is not available. Throughout the five years that this project has dominated my life, both in and outside of the lab, several key-people have been crucial so that I could be here today, writing the acknowledgement section of my life's work at the end of a very long road.

To my supervisors, Dr. João Carriço and Prof. Dr. Mário Ramirez, my deepest thanks for being not only a continuous source of inspiration, but of unconditional support. At the very beginning of this journey, when this was not more than just an idea, you've embraced me wanting to "change paths" and switch from comparative genomics to the wonderful and exciting world of metagenomics. Some of the initial excitement has faded, as it became more evident that this was a challenging road, but with a healthy dose of scepticism you've helped me keep my feet on the ground while my head stayed in the clouds, forever passionate about embracing new challenges and taking on new adventures. In particular to Mário, who started out as my mentor and had to step up to the challenge of supervising me, I give my deepest thanks. Without your support I would not be the scientist I am today, and I do hope one day to make you proud. I will forever carry with me the example of what is to lead a team with grace and patience. I would also like to thank Prof. Dr. John Rossen, who albeit briefly, was part of my original advisory team and thanks to whom I was able to give my first steps in the world of metagenomics.

To the MRamirez lab colleagues, both current and past, you all have marked me and my journey in some way. For all the moments shared, all the laughter and the many shoulders to cry on, I thank you. To Miguel and Catarina, you guys have been my rock. To all the "binfies" who crossed my path, Bruno, Mickael, Diogo, Tiago, and my dear "pitufos" Pedro and Rafael, I've learned so much with each and every one of you. It has been a pleasure and a privilege. Last but not least, to Joana, who showed me the value of compassion, of dedication and of true friendship.

To the friends I made in the Netherlands, Casper, Lieke, Boas, and whom I had the pleasure of sharing a fantastic and dynamic workspace with, Erley, Nilay, Peppe, Leo and Natacha, my biggest thanks. You made my adventure of living abroad memorable.

To all the people in Lisbon Data Science Academy, you've showed me that a project moved solely by the passion of teaching, with no monetary interest, can exist and be successful. You've thought me more than data science, but how to be a leader, a manager, and a friend. A particular thanks goes to my fellow women in arms, Minh and Carol, who drive this project to be better, do better, for our students. To all the students who I've had the pleasure of sharing a bit of knowledge with, thank you for being a source of inspiration and motivation.

To my friends who have become more than family, Nuno, Alicia, Sofia, Neto, Margarida and Clara, thank you for being there in my lowest and highest moments. I'm blessed to have you in my life. To you, Rúben, who grew from a little pebble in my shoe to a huge rock for me to lie against, thank you.

To my mum and dad, thank you for your unconditional love, for being my biggest fans, my most loyal allies. To my sister, who never fails to push me to be better, you're my best friend in the whole world. To Miguel e Inês, you bring so much joy to my life. Being your aunt is not only a pleasure, it's an honour to be able to watch you grow. Even though work has made me miss many birthdays over the years... To my brother in law Nuno, thank you for showing me what perseverance means. To my grandma, who is more than a mother to me, and my grandpa, who might not be here anymore but will live forever in my heart, thank you for showing me what love means, now and forever. To you, my dear grandpa, I dedicate my life's work, to honour you and your memory every day of my life. O primeiro bebe-se inteiro.

Summary

By analysing metagenomic data, the present thesis aims to identify pathogens with clinical relevance, analyse their virulence, and predict their antimicrobial susceptibility. Metagenomic methods must be evaluated in a clinical setting as an alternative to current golden standards. Given the dependence of these methodologies on bioinformatics post-processing of the raw data obtained, the significant applications and pitfalls of metagenomics are yet to be identified.

Microbial pathogens are responsible for more than 400 million years of life lost annually across the globe, a higher burden than either cancer or cardiovascular disease. In addition to the emergence of virulent pathogens, the rise of antimicrobial resistance poses a major threat to human health worldwide. Clinical microbiology is a discipline focused on rapidly characterising pathogen samples to direct the management of individual infected patients (diagnostic microbiology) and monitor the epidemiology of the infectious disease (public health microbiology), including the detection of outbreaks and infection prevention. For the purpose of this dissertation work, we will focus on bacterial and viral infections.

Since the publication of the first complete microbial genome, a quarter of a century ago genomics has transformed the field of microbiology, and in particular its clinical application. The development and commercialisation of high-throughput, massively parallel sequencing has democratised sequencing by offering individual laboratories, either in research or in health, access to the technology. Three main revolutions can be considered in genomic sequencing: the first generation, also known as automated Sanger sequencing, the second generation, also known as next-generation sequencing, and the third generation of sequencing, also known as long-read sequencing or single-molecule sequencing. Whole genome sequencing has been used in the routine laboratory workflow for typing pathogens by a method having the highest possible discriminatory power in comparison with the gold-standard molecular methods, such as polymerase chain reaction. Most notably, whole genome sequencing has become a common tool in infection surveillance and prevention, allowing the identification and tracking of pathogens, establishing transmission routes and assisting with outbreak control. Despite this, the implementation of whole genome sequencing in routine diagnostics requires several adaptations in the laboratory workflow, from the ‘wet’ laboratory part (extraction, library preparation, sequencing), to the ‘dry’ bioinformatics part where genomic data is analysed and its results interpreted by specialised personnel.

It is true that DNA sequencing methods are increasingly being adopted in clinical microbiology, but it generally requires a priori knowledge of what a clinical sample will contain. One of the possibilities for overcoming this limitation is through the implementation of metagenomics, delivering culture-independent approaches to microbial ecology, surveillance and diagnosis. While most molecular assays target only a limited number of pathogens, metagenomic approaches characterise all DNA or RNA present in a sample, enabling analysis of the entire microbiome as well as the human host genome or transcriptome in patient samples. Whether or not it can entirely replace routine microbiology depends on several conditions and future developments, both technological and computational. The bioinformatics analysis required, due to the amount of data produced by genomic sequencing technologies, represents one of the major hurdles for the applicability of this methodology in diagnosis and surveillance.

In this work, shotgun metagenomics was successfully applied to nine body fluid samples and one tissue sample from patients at the University Medical Center Groningen with varying degrees of complexity. Shotgun metagenomics was compared to standard culture-based microbiological methods in order to evaluate and compare the accuracy and reliability of the bioinformatics analyses. Furthermore, this methodology was applied to eight concentrated water samples also collected from the University Medical Center Groningen. In one of the samples, the novel detection of an *mcr-5* gene, named *mcr-5.4*, is reported.

With the lessons learned from processing both the clinical and environmental samples collected, we have developed DEN-IM, a one-stop, user-friendly, containerised and reproducible workflow for the analysis of Dengue virus short-read sequencing data from both amplicon and shotgun metagenomics approaches. Genomic sequencing is the most informative approach to monitor viral dissemination and genetic diversity by providing, in a single methodological step, identification and characterization of the whole viral genome at the nucleotide level. DEN-IM was designed to perform a comprehensive analysis in order to generate either assemblies or consensus of full DENV coding sequences and to identify their serotype and genotype. DEN-IM can play a role in the identification of co-infection cases whose prevalence is increasingly detected in highly endemic areas.

The de novo assembly of raw sequence data is key in metagenomic analysis, yielding longer sequences that offer contextual information and afford a more complete picture of the microbial community. The assembly process is the bedrock and may constitute a major bottleneck in obtaining trustworthy, reproducible results. Hence, LMAS, an automated workflow, was developed as a flexible platform to allow users to evaluate traditional and metagenomic dedicated prokaryotic de novo assembly software performance given known standard communities. Similarly to DEN-IM, LMAS implementation ensures the transparency and reproducibility of the results obtained, presented in an interactive HTML report where global and reference-specific performance metrics can be explored.

Despite the relative standardisation of the process to acquire genomic data, either through whole genome sequencing or metagenomics, there's a myriad of different tools available to perform in silico antimicrobial resistance detection, with some using their own reference database and each one generating a unique, non-standardised report of the genes or variants that can possibly confer resistance in a given sample. This is a significant barrier to the comparison of results and the modularity of tools within bioinformatic workflows. Given this predicament, a standardised output specification for the reporting of genes or variants potentially conferring antimicrobial resistance is presented, packaged into hAMRonization, a command-line utility that is able to aggregate results from a wide variety of AMR detection tools, both species-agnostic and species-specific, providing a unified report tabular form, JSON or through an interactive HTML file.

Despite the richness of genomic information available, the same is not observed for the contextual information that accompanies it. Following the same approach developed for antimicrobial resistance detection, a standardised output specification was conceived, this time applied to SARS-CoV-2 contextual data based on harmonisable, publicly available community standards. This is implemented through a collection template, as well as a variety of protocols and tools to support both the harmonisation and submission of sequence data and contextual information to public biorepositories.

Throughout this work, it became obvious that computational algorithms have become an essential component of microbiome research, with great efforts by the scientific community to raise standards on the development and distribution of code. Despite these efforts, sustainability and reproducibility are major issues since continued validation through software testing is still not a widely adopted practice. In an effort to uphold good software engineering practices, we report seven recommendations that help researchers implement software testing in microbial bioinformatics. We propose collaborative software testing as an opportunity to continuously engage software users, developers, and students to unify scientific work across domains. As automated software testing remains underused in scientific software, our set of recommendations not only ensures appropriate effort can be invested into producing high-quality and robust software but also increases engagement in its sustainability.

The impact and applicability of shotgun metagenomic in clinical microbiology, including both diagnosis and surveillance and infection prevention, has been assessed, with the unique challenges of both highlighted. A strong focus on the standardisation and reproducibility of the results obtained, with the employment of new technologies to do so is of the uttermost necessity for the shotgun metagenomic data analysis solutions. Transparency, scalability, and ease of installation are keystones, regardless of the tools chosen. The solutions adopted throughout this work, allied to clear and easy-to-follow documentation, aim to lower the barrier of entry when performing detailed analyses that are complex and computationally expensive. But most importantly, the production of intuitive, responsive and easy-to-follow reports, allows the summary of key results, as well as the detailed exploration of the resulting data, by stakeholders, be it bioinformatic personnel or experts in the given area of expertise. Such reports represent the single most important contribution to lowering the barrier between

those who produce the data and those with the capacity to make informed decisions based on that data.

Keywords: Clinical Microbiology, Genome Sequencing, Metagenomics, Reproducibility, Interoperability

Resumo

A microbiologia é a ciência que estuda os microrganismos, um vasto e diverso grupo de organismos microscópicos que podem ser encontrados como células undividualizadas ou em agrupamentos celulares. Dentro dos microrganismos estão também incluídos os vírus que, apesar de terem uma estrutura acelular e sendo sempre encontrados associados a outros seres vivos, são capazes de realizar os seus próprios processos vitais de crescimento. Todas as células são compostas por quatro componentes químicos (biomoléculas): proteínas, ácidos nucleicos, lípidos e polissacáridos. A célula corresponde à unidade fundamental do ser vivo, sendo os microrganismos representados por uma grande diversidade de organismos com diferentes características celulares. Devido a esta diversidade, a microbiologia encontra-se dividida em diferentes áreas: a virologia, a bacteriologia, a micologia e a parasitologia. Para efeitos desta dissertação apenas nos iremos focar em duas áreas: a virologia, ciência que se dedica ao estudo dos vírus e das viroses, e a bacteriologia, ciência que investiga o que é relativo às bactérias, nome geral dado aos microrganismos unicelulares de formas variadas, sem membrana nuclear.

No ser humano, o maior número de bactérias está na microbiota intestinal, seguido pela pele. A grande maioria das bactérias são inofensivas devido aos efeitos protectores do sistema imunitário, e muitas são benéficas, particularmente na microbiota intestinal. No entanto, várias espécies têm potencial patogénico e causam doenças. Os microbianos patogénicos são atualmente responsáveis por mais de 400 milhões de anos de vida perdidos anualmente em todo o mundo, uma carga maior do que o cancro ou as doenças cardiovasculares. Além do aparecimento de novos agentes patogénicos ou de uma maior virulência dos agentes conhecidos, o aumento da resistência aos fármacos antimicrobianos representa uma das maiores ameaças à saúde humana. A microbiologia clínica é então uma disciplina focada na caracterização rápida de amostras para a identificação de agentes causadores de doença (diagnóstico) e monitorizar a epidemiologia das doenças infeciosas (saúde pública), incluindo a deteção de surtos e prevenção das infeções.

Em microbiologia clínica, tanto na vertente de diagnóstico como na vertente de saúde pública, a identificação rápida de um possível agente de infeção é um passo essencial. A identificação é tipicamente efetuada pela caracterização bioquímica e fenotípica dos microrganismos, como por exemplo pelo crescimento em meios de cultura adequados seguido da análise de múltiplas características fisiológicas, metabólicas e bioquímicas. Com o advento

das técnicas de biologia molecular e imuno-serologia, a microbiologia sofreu uma revolução metodológica, permitindo a identificação rápida e específica das espécies microbianas. Contudo, é necessário o conhecimento *a priori* do possível causador de doença, muitas vezes por cultura. A metagenómica, definida como a sequenciação e análise genómica de uma comunidade de microrganismos, de um determinado ambiente, por técnicas independentes de cultura, apresenta-se como uma alternativa válida à deteção de possíveis agentes patogénicos num único passo metodológico. A bioinformática, definida como a área computacional da microbiologia molecular, é composta por tarefas especializadas onde análises de dados são utilizadas para processamento e análise *in silico* de informação sobre biomoléculas, em oposição aos métodos *in vivo* (num organismo vivo) ou *in vitro* (num ambiente artificial) tradicionalmente utilizados. É composta pelo desenvolvimento de *software* ou de *workflows*, onde software é encadeado para o processamento automático de informação, e pela interpretação dos resultados obtidos.

O trabalho aqui apresentado tem como objetivo avaliar a implementação de métodos de metagenómica em microbiologia clínica, como alternativa aos métodos padrão ainda hoje implementados. Tem como especial foco a análise bioinformática deste tipo de dados que, dado ao seu volume e características, requer um processamento dedicado e delicado, particularmente se implementado em diagnóstico.

Os ácidos nucleicos são compostos por dois grupos de moléculas: o ácido desoxirribonucleico (ADN) e o ácido ribonucleico (ARN). O ADN possui uma estrutura de dupla hélice que é composta por quatro bases que emparelham sempre com o mesmo par. O nucleótido Adenina (A) emparelha com o nucleótido Timina (T) e o nucleótido Citosina (C) com o nucleótido Guanina (G). Estes pares são a base da molécula de ADN e a da replicação, da divisão celular, bem como dos métodos de sequenciação. O ácido ribonucleico (ARN) é também formado por uma cadeia de nucleótidos, mas diferentemente do ADN, sendo monocatenários, embora possam dobrar-se sobre si mesmos. Têm uma composição semelhante, com a diferença em que o nucleótido T é substituído pelo nucleótido Uracilo (U). O dogma principal da biologia dita que a informação genética é passada do ADN para o ARN através de um processo de transcrição, sendo o último expresso em proteínas que acatam uma certa função através da sua tradução.

A sequenciação é um processo que determina a ordem de nucleótidos numa molécula específica de ADN de qualquer organismo. Esta informação é muito útil na investigação e na prática clínica, dado que permite compreender qual o tipo de informação genética que as moléculas carregam. As moléculas de ARN também são possíveis de serem sequenciadas após a sua conversão por transcrição revertida em ADN. O conhecimento das sequências de ADN e ARN nos últimos anos tem-se tornado indispensável. Atualmente, existem já bem estabelecidos três tipos de sequenciação de ADN, a de baixo rendimento ou 1ª geração, a de alto rendimento ou 2ª geração e a sequenciação de cadeias longas ou 3ª geração.

A sequenciação total do genoma é, atualmente, parte da rotina laboratorial aquando da tipagem de microrganismos. Dado o seu elevado poder discriminatório, tornou-se numa ferramenta importante em saúde publica. Permite a identificação de agentes patogénicos, identificação e estudo de vias de transmissão informando as medidas de controlo de surtos. Contudo, a implementação da sequenciação total do genoma em diagnóstico exige várias adaptações no fluxo de trabalho, a nível laboratorial; desde o trabalho de bancada (extração, preparação da biblioteca, sequenciamento), até à analise bioinformática, onde os dados de genómica são analisados e os seus resultados interpretados por pessoal especializado.

Os métodos de sequenciação de ADN estão a ser cada vez mais adotados em microbiologia clínica, contudo, tal como os métodos moleculares, requerem um conhecimento *a priori* de quais os microrganismos que a amostra de um doente poderá conter. Uma das possibilidades para ultrapassar esta limitação é através da implementação da metagenómica. Enquanto a maioria dos ensaios moleculares visa apenas um número limitado de agentes patogénicos, a abordagem metagenómica caracteriza todo o ADN ou ARN presente numa amostra, permitindo identificar e analisar todos os microrganismos presentes. Se poderá ou não substituir totalmente as metodologias atualmente implementadas na rotina de microbiologia clínica, depende de várias condições e desenvolvimentos futuros, tanto tecnológicos como computacionais. A análise bioinformática, exigida devido à quantidade de dados produzidos pelas tecnologias genómicas de sequenciação, representa uma das principais limitações para a aplicabilidade desta metodologia.

Neste trabalho, a metagenómica foi aplicada com sucesso a nove amostras de diferentes doentes do University Medical Center Groningen, com diferentes graus de contaminação por ADN humano, e comparada com métodos microbiológicos baseados em cultura, de modo a avaliar a precisão e fiabilidade das análises bioinformáticas efetuadas. Além disso, esta metodologia foi aplicada a oito amostras de água também recolhidas no University Medical Center Groningen. Numa das amostras, é relatada a deteção de uma nova variante de um gene a resistência a antimicrobianos *MCR-5*, denominado de *MCR-5.4*.

Com as lições aprendidas com o processamento das amostras clínicas e ambientais recolhidas, desenvolvemos DEN-IM, uma *workflow* de fácil utilização e reproduzível, para a análise de dados de sequenciação de 2ª geração do vírus da Dengue (DENV), tanto a partir de abordagens de amplificação de fragmentos do genoma total, como de metagenómica. A sequenciação genómica é a abordagem mais informativa para monitorizar a disseminação viral e a diversidade genética, proporcionando, num único passo metodológico, a identificação e caracterização de todo o genoma viral. O DEN-IM foi concebido para realizar uma análise abrangente, com o objetivo de gerar sequências completas de DENV e identificar o seu serótipo e genótipo. O DEN-IM pode também desempenhar um papel na identificação de casos de coinfecção, por duas ou mais estirpes de DENV, cuja prevalência é cada vez mais elevada em áreas endémicas.

A metagenómica pode oferecer uma deteção microbiana abrangente e caracterização de amostras clínicas complexas. A montagem de dados brutos de sequenciação em sequências mais longas, que oferecem informação contextual e uma imagem mais completa da comunidade microbiana em questão, é um passo fundamental na análise de dados de metagenómica. Contudo, este processo de montagem é frequentemente um passo limitante na obtenção de resultados confiáveis e reprodutíveis. Assim, foi desenvolvido o LMAS, uma *workflow* automatizada que permite aos utilizadores avaliar o desempenho de software de montagem de genomas, tanto tradicionais como específicos para dados de metagenómica, dada uma comunidade de composição conhecida. À semelhança do DEN-IM, a implementação do LMAS garante a transparência e reprodutibilidade dos resultados obtidos, apresentados num relatório interativo HTML, onde podem ser exploradas métricas de desempenho globais e específicas da referência utilizada.

Apesar da relativa normalização do processo para adquirir dados genómicos, seja através de toda a sequenciação do genoma ou da metagenómica, há uma miríade de diferentes ferramentas disponíveis para executar na deteção *in silico* de resistência a fármacos antimicrobianos. Alguns usam a sua própria base de dados com sequências de referência e cada uma gera um relatório único e não padronizado dos genes ou variantes que podem possivelmente conferir resistência numa determinada amostra. Esta é uma enorme barreira para a comparação dos resultados obtidos. Dada esta situação, é apresentada uma especificação para a detecção de genes ou variantes que conferem resistência, disponibilizada no software hAMRonization, capaz de agregar resultados de uma grande variedade de ferramentas de deteção de resistência a antimicrobianos, tanto agnósticas como específicas de uma determinada espécie, fornecendo um relatório unificado, em forma tabular, JSON ou através de um ficheiro HTML interativo.

Apesar da riqueza da informação genómica disponível, o mesmo não se observa para a informação contextual que a acompanha. Seguindo a mesma abordagem desenvolvida para a deteção de resistência a antimicrobianos, foi concebida uma especificação, desta vez aplicada aos dados contextuais do coronavírus do síndrome respiratório agudo grave 2 (SARS-CoV-2), baseados em normas desenvolvidas pela comunidade e publicamente disponíveis. Esta especificação está implementada e disponível através de um modelo de recolha de dados, bem como numa variedade de protocolos e ferramentas para apoiar a harmonização e submissão de dados de sequenciação e informação contextual para bio-repositórios públicos.

Ao longo deste trabalho, tornou-se óbvio que os algoritmos computacionais são um componente essencial da investigação tanto do microbioma como da etiologia das mais diversas infecções. Têm havido grandes esforços por parte da comunidade científica para se elevar os padrões de desenvolvimento e distribuição de *software* para este fim. Apesar destes esforços, a sustentabilidade e a reprodutibilidade continuam a ser questões importantes, uma vez que a continuação da validação através de testes de software ainda não é uma prática amplamente adotada. Num esforço para manter boas práticas de engenharia de *software*, reportamos sete recomendações que ajudam investigadores a implementar testes de *software* em bioinformática. Propomos a utilização destes testes como uma oportunidade para envolver quem desenvolve *software*, assim como quem o utiliza no quotidiano, unificando o trabalho cien-

tífico em todos os domínios. Como os testes de *software* automatizados permanecem em subuso em *software* científico, o nosso conjunto de recomendações não só garante que o esforço adequado pode ser investido na produção de *software* de alta qualidade e robusto, mas também aumenta o envolvimento da comunidade na sua sustentabilidade.

O impacto e aplicabilidade da metagenómica na microbiologia clínica, incluindo o diagnóstico, a vigilância e prevenção de infeções, foi avaliado, com os desafios únicos de cada um destacados. Uma forte aposta na normalização e reprodutibilidade dos resultados obtidos, com o uso de novas tecnologias é absolutamente necessário para aplicação sustentável de soluções de análise de dados, bem como para a aplicabilidade da metagenómica em microbiologia clínica. A transparência, escalabilidade e facilidade de instalação são características essenciais no *software* desenvolvido e utilizado neste tipo de análise, independentemente das ferramentas escolhidas. As soluções adotadas ao longo deste trabalho, aliadas a uma documentação clara e fácil de seguir, visam reduzir a barreira de entrada ao realizar análises detalhadas que sejam complexas e computacionalmente caras na sua natureza. Adicionalmente, a produção de relatórios intuitivos, interactivos e fáceis de seguir, permite o resumo dos resultados-chave, bem como a sua exploração detalhada pelas partes interessadas, seja pessoal bioinformático ou peritos na área de especialização. Isto representa o contributo mais importante para a redução da barreira entre quem produz os dados e quem tem a capacidade de tomar decisões informadas com base nestes.

Com o trabalho aqui apresentado, foram identificadas características chave necessárias em *workflows* e *software* bioinformático que, em ultima instância, facilitam a implementação da metagenómica na rotina laboratorial tanto no diagnóstico como em saúde pública.

Keywords: Microbiologia Clínica, Sequênciação genómica, Metagenómica, Reproducibilidade, Interoperabilidade

Thesis Outline

The work described in the present thesis intends to evaluate the use of bioinformatics methods for the analysis of metagenomic data to allow the rapid identification, virulence analysis and antimicrobial susceptibility prediction of pathogens with clinical relevance. Ultimately, the applicability of metagenomic methods is to be evaluated in a clinical setting as an alternative to current gold standards. Given the dependence of these methodologies on bioinformatics post-processing of the raw data obtained, the major applications and pitfalls of metagenomics are yet to be identified.

The thesis organisation reflects the six publications included, with each of the results chapters containing mostly the content of the publication indicated at the start of that chapter. Given this option, in addition to an introductory preface giving the context of the work performed, each results chapter has its own introduction, materials and methods, discussion, and conclusion. Overall, the thesis comprises 10 chapters, organised as follows: The first, second to last and last chapters include the general introduction, general discussion and main conclusions of the work developed in this thesis, respectively; while the body of the thesis work, corresponding to chapters 2 to 8, is organised into four parts aggregating chapters under a broader topic, numbered from I to IV.

In **Chapter 1** the issues addressed throughout the thesis are put into context, highlighting the current impact of genomics in clinical microbiology, both as a diagnostic or a surveillance tool. The entire process in clinical microbiology for bacterial and viral infections is showcased through its different approaches over time: classical biochemical and molecular methods, whole-genome sequencing, and sequencing through metagenomics, both metataxonomics and shotgun, with a focus on the computational requirements necessary. This chapter elaborates on the evolution of whole-genome sequencing to metagenomic approaches, introducing the possibility of identifying and characterising a potential pathogen without the need for a priori knowledge of the causative agent of disease. The importance of the bioinformatics analysis of this data was underlined, showcasing its complexity and major pitfalls, such as reproducibility and transparency of the analysis methods.

Part I of the results, named "*Applying metagenomics in the clinical context*", includes **Chapters 2, 3 and 4**, where metagenomics is applied to both clinical and environmental samples. In the two first chapters, the performance of metagenomics to identify pathogens and antimicrobial resistance genes was evaluated. Importantly it was recognised that there was a lack of standardisation in the bioinformatic metagenomics analysis, so a fully reproducible workflow was developed for the streamlined analysis of human samples for genomic characterization in the context of Dengue infections, as a model for other applications.

Chapter 2 consists of the application of the shotgun metagenomics approach to nine body fluid samples and one tissue sample from patients at the University Medical Center Groningen (UMCG) as to compare against current golden standards practises in the diagnosis of disease. In this study, the accuracy and reliability of the bioinformatics analyses were evaluated and compared against the results obtained from traditional culture methods. Our aim was to evaluate the applicability of shotgun metagenomics in a routine diagnostic setting, and not only in cases where traditional methods fail to provide an answer. Most pathogens identified by culture were also identified by metagenomics. Substantial differences were noted between the taxonomic classification tools, highlighting the potential and limitations of shotgun metagenomics as a diagnostic tool. The fact that, when applying shotgun metagenomics to diagnostics, the results are highly dependent on the tools, and especially the database that was chosen for the analysis greatly impacts its applicability in a clinical setting. This chapter is included in the following publication: *N. Couto, L. Schuele, E.C. Raangs, M. P. Machado, C. I. Mendes, T. F. Jesus, M. Chlebowicz, S. Rosema, M. Ramirez, J. A. Carriço, I. B. Autenrieth, A. W. Friedrich, S. Peter and J. W. Rossen. Critical steps in clinical shotgun metagenomics for the concomitant detection and typing of microbial pathogens. Sci Rep 8, 13767 (2018). DOI:* `https://doi.org/10.1038/s41598-018-31873-w`

Chapter 3 describes the application of both second and third-generation sequencing technologies, also known as next-generation and long-read sequencing, to eight tap-water samples collected at the University Medical Center Groningen. Our aim was to evaluate the applicability of shotgun metagenomics, but this time in a surveillance setting. Building on the findings from Chapter 2, a hybrid assembly approach was used to increase resolution power. In this sample a new variant of a colistin resistance (mcr) determinant was detected, named *mcr-5.4*, and through hybrid assembly leveraging both short and long-read sequences, its context was determined, albeit with questionable success. This chapter is included in the following publication: *G. Fleres, N. Couto, L. Schuele, M. A. Chlebowicz, C. I. Mendes, L. W. M. van der Sluis, J. W. A. Rossen, A. W Friedrich, S. García-Cobos, Detection of a novel mcr-5.4 gene variant in hospital tap water by shotgun metagenomic sequencing, Journal of Antimicrobial Chemotherapy, Volume 74, Issue 12, December 2019, Pages 3626–3628. DOI:* `https://doi.org/10.1093/jac/dkz363`*.*

With the lessons learnt in Chapters 2 and 3, we developed in **Chapter 4** DEN-IM, a one-stop, user-friendly, containerised and reproducible workflow for the analysis of Dengue virus short-read sequencing data from both amplicon and shotgun metagenomics approaches. This takes into particular consideration the dependency on software and database versions used in the metagenomic bioinformatics downstream analysis in the results obtained. Dengue virus represents a public health threat and economic burden in affected countries, with the risk of exposure, increasing, not only driven by travel to endemic regions but also due to the broader dissemination of the mosquito vector, making the burden of dengue very significant. This makes it a particularly relevant target organism for the development of a straightforward workflow for both the identification and characterization of the virus. DEN-IM was designed to perform a comprehensive analysis in order to generate either de novo assemblies or consensus of full viral coding sequences and to identify their serotype and genotype, including the identification of co-infection cases whose prevalence is increasingly found in highly endemic areas. It was developed in Nextflow, a simple and scalable workflow management system. All tools and dependencies are provided in Docker containerised images. All these steps ensure reproducibility and transparency of the bioinformatic process. This chapter is included in the following publication: *C. I. Mendes*[*]*, E. Lizarazo*[*]*, M. P. Machado, D. N. Silva, A. Tami, M. Ramirez, N. Couto, J. W. A. Rossen, J. A. Carriço, DEN-IM: dengue virus genotyping from amplicon and shotgun metagenomic sequencing. Microbial Genomics, Volume 6, Issue 3, March 2020. DOI:* `https://doi.org/10.1099/mgen.0.000328`.

Part II includes only **Chapter 5**, and is named "*The impact of de novo assemblers in metagenomics*". A key process in metagenomic data analysis is the de novo assembly of raw sequence data since it allows recovering contigs representing the replicons present in the sample, be it genomes, plasmids, or bacteriophages, from a pool of mixed raw reads. **Chapter 5** employs the same core principles as in Chapter 4, describing a one-stop, user-friendly, containerised, and reproducible workflow, named LMAS, to assess the performance of de novo assembly algorithms for the assembly of second-generation metagenomic sequencing data. The LMAS workflow, which allows users to evaluate performance given a known standard community was implemented in Nextflow, ensuring the transparency and reproducibility of the results obtained. Similarly to Chapter 4, the use of Docker containers provides additional flexibility. The results are presented in an interactive HTML report where global and reference-specific performance metrics can be explored. Currently, 12 de novo assemblers are implemented in LMAS, with the possibility of expansion as novel algorithms are developed. This chapter is included in the following publication: *C. I. Mendes, P. Vila-Cerqueira, Y. Motro, J. Moran-Gilad, J. A. Carriço, M. Ramirez. LMAS: Evaluating metagenomic short de novo assembly methods through defined communities. GigaScience, Volume 12, 2023, giac122,* `https://doi.org/10.1093/gigascience/giac122`.

Not only is the analysis of the data critical, but also the post-processing and its contextualisation with other relevant available metadata, to make the most of genomics in the context of clinical microbiology, infection control and public health. **Part III**, covering **Chapters 6 and 7**, is focused on the "*Challenges of data availability in metagenomics and beyond*". Despite the advantages of reproducible, containerised workflow, Chapters 4 and 5 still do not guarantee the interoperability of results obtained from various sources. Chapter 5 highlighted the impact that the tool choice can have on downstream results when working with metagenomic data, therefore, and due to the lack of standardisation, it is pivotal that re-

---

[*]These authors contributed equally to this work.

sults from various tools can be compared for their applicability in the clinic. With a focus on antimicrobial resistance, **Chapter 6** presents a standardised output specification for the bioinformatic detection of antimicrobial resistance directly from genomes or metagenomes. This addresses the problem of combining the outputs of disparate antimicrobial resistance gene detection tools into a single unified format, implemented into a python package and command-line utility hAMRonization. As the detection of antimicrobial resistance directly from genomic or metagenomic data has become a standard procedure in public health, with hAMRonization allowing for the comparison of results within bioinformatics workflows, as these tools, although implementing similar principles, differ in supported inputs, search algorithms, parameterisation, and underlying reference databases.

Chapter 7 presents a direct application of a standardised specification, such as the one presented in Chapter 6. For this purpose, a SARS-CoV-2 contextual data specification package based on harmonisable, publicly available community standards was developed and implemented through a collection template, as well as a variety of protocols and tools to support both the harmonisation and submission of sequence data and contextual information to public biorepositories. In addition to the reproducibility and interoperability of data and software, transparency is also a keystone in the use of bioinformatics methods for the analysis of metagenomic data. This chapter is included in the following publication: *E. J. Griffiths, R. E. Timme, C. I. Mendes, A. J. Page, N. Alikhan, D. Fornika, F. Maguire, J. Campos, D. Park, I. B. Olawoye, P. E. Oluniyi, D. Anderson, A. Christoffels, A. G. da Silva, R. Cameron, D. Dooley, L. S. Katz, A. Black, I. Karsch-Mizrachi, T. Barrett, A. Johnston, T. R. Connor, S. M. Nicholls, A. A. Witney, G. H. Tyson, S. H. Tausch, A. R. Raphenya, B. Alcock, D. M. Aanensen, E. Hodcroft, W. W. L. Hsiao, A. T. R. Vasconcelos, D. R. MacCannell on behalf of the Public Health Alliance for Genomic Epidemiology (PHA4GE) consortium, Future-proofing and maximizing the utility of metadata: The PHA4GE SARS-CoV-2 contextual data specification package. GigaScience, Volume 11, 2022, giac003. DOI:* `https://doi.org/10.1093/gigascience/giac003`.

Part IV includes only **Chapter 8**, and focuses on the use of "*Crowdsourcing to improve software robustness in metagenomics and beyond*". **Chapter 8** showcases an effort to raise standards on the development and distribution of code for bioinformatic analysis. For this, seven recommendations are presented that help researchers implement software testing in microbial bioinformatics. We propose collaborative software testing as an opportunity to continuously engage software users, developers, and students to unify scientific work across domains. As automated software testing remains underused in scientific software, our set of recommendations not only ensures that appropriate effort can be invested in producing high-quality and robust software but also increases engagement in its sustainability. This chapter is included in the following publication: *B. C. L. van der Putten*[*], *C. I. Mendes*[*], *B. M. Talbot, J. de Korne-Elenbaas, R. Mamede, P. Vila-Cerqueira, L. P. Coelho, C. A. Gulvik, L. S. Katz, The Asm Ngs Hackathon Participants, Software testing in microbial bioinformatics:*

---

[*] These authors contributed equally to this work.

a call to action. Microbial Genomics, Volume 8, Issue 3. DOI: `https://doi.org/10.1099/mgen.0.000790`*.*

Chapter 9 corresponds to the general discussion. This chapter provides a summary of the main results obtained in this thesis and its integrated discussion. It is divided into two main sections: the current limitations to the application of metagenomics in clinical microbiology; and the better standards required for metagenomics to become a standard microbiological method, with a clearly defined role in both diagnosis and surveillance. For the first, three major limitations were identified, starting with the limitations inherent to the sequencing technology itself, followed by the unbiased nature of metagenomics, very sensitive to host and/or environmental contamination, and ending in the limitation of the bioinformatic analysis itself, where no standard procedure is *de facto* accepted. Several steps are required to improve the standards in metagenomics before its routine application. The need for proper benchmarks, with the use of well-characterised communities, is paramount for protocol validation. Likewise, the adoption of reproducible and auditable workflows, relying upon well-established software is just as important as wet-lab procedures, with just as much influence on the validity of the results obtained. The use of intuitive and responsive reports will allow clinical and research personnel without the technical know-how to infer knowledge from the complex analysis required for the application of metagenomics. The application of data standards, with controlled vocabulary, will also contribute to crossing the data-to-informative-report bridge. Finally, the community needs to be engaged in adopting these practices, with crowdsourcing being a viable option for the dissemination of better standards worldwide.

Chapter 10 contains the main conclusions derived from this work and also perspectives for future work.

Abbreviation

ACEGID African Center of Excellence for Genomics of Infectious Diseases

AMR Antimicrobial Resistance

ASM NGS American Society for Microbiology Next Generation Sequencing

ATCC American Type Culture Collection

bp basepairs

CAMI Critical Assessment of Metagenome Interpretation

CanCOGeN Canadian COVID Genomics Network

CDS Coding Sequence

cgMLST core-genome Multilocus Sequence Typing

cg/wg MLST core-genome/whole genome Multilocus Sequence Typing

CI Continuous Integration

COG-UK COVID-19 Genomics UK Consortium

COVID-19 Coronavirus disease of 2019

CSIS Code Safety Inspection Service

dBg De Bruijn graphs

ddNTP dideoxynucleotide triphosphate

DENV Dengue virus

DNA Deoxyribonucleic acid

dNTP deoxynucleotide triphosphate

EBI European Bioinformatics Institute

EFO Experimental Phenotype Ontology

EMBL-EBI	European Molecular Biology Laboratory’s European Bioinformatics Institute
ENA	European Nucleotide Archive
FAIR	Findable Accessible Interoperable Reusable
FS	Filtered Set
GAZ	Gazetteer Ontology
GB	Gigabytes
GBD	Global Burden of Disease
GenEpiO	Genomic Epidemiology Ontology
GISAID	Global Initiative on Sharing All Influenza Data
GPP	Global Priority Pathogens
HP	Human Phenotype Ontology
HPC	high-performance computing
HTS	high-throughput sequencing
INSDC	International Nucleotide Sequence Database Collaboration
INSACOG	Indian SARS-CoV-2 Genomic Consortia
JSON	JavaScript Object Notation
LIMS	Laboratory Information Management System
LFIA	Lateral Flow Immunoassays
LSA	Longest single alignment
MAG	Metagenomic Assembled Genome
MERS	Middle East Respiratory Syndrome
MIGS	Minimum Information about a Genomic Sequence
MIxS	Minimum Information about any Sequence
MLST	Multilocus Sequence Typing
MP	Mammalian Phenotype Ontology
NCBI	National Center for Biotechnology Information
NCBITaxon	NCBI Taxonomy Ontology

NCIT	National Cancer Institute Thesaurus
NCR	non-coding region
NIST	National Institute of Standards and Technology
OBI	Ontology for Biological Investigations
OBO Foundry	Open Biological and Biomedical Ontology Foundry
OLC	Overlap-Layout Consensus
ONT	Oxford Nanopore Technologies
OTUs	Operational Taxonomic Units
PacBio	Pacific Biosciences
PCR	Polymerase Chain Reaction
PFGE	Pulse Field Gel Electrophoresis
PHA4GE	Public Health Alliance for Genomic Epidemiology
PHSS	Public Health Surveillance Systems
Pls	Phred-like score
PMC	PubMed Central®
QC	quality control
qPCR	Real-Time Quantitative PCR
RNA	Ribonucleic acid
rRNA	ribossomal RNA
RT-PCR	Reverse Transcription Polymerase Chain Reaction
SANBI	South African National Bioinformatics Institute
SARS-CoV-2	Acute Respiratory Syndrome Coronavirus 2
SMg	Shotgun Metagenomics
SMRT	Single Molecule Real-Time Sequencing
SNP	Single Nucleotide Polymorphism
SOP	standard operating procedure

SPHERES	SARS-CoV-2 Sequencing for Public Health Emergency Response, Epidemiology and Surveillance
SRA	Sequence Read Archive
STEC	Shiga toxin-producing *Escherichia coli*
UBERON	Uber-Anatomy Ontology
URL	Uniform Resource Locator
UO	Unit Ontology
VCS	Version control systems
WGS	Whole Genome Sequencing
WHO	World Health Organization

Table of Contents

List of Tables

List of Figures

Chapter 1

General Introduction

1.1 The global impact of microbial pathogens

The Global Burden of Disease (GBD) 2019 study reported that microbial pathogens are responsible for more than 400 million years of life lost annually across the globe, a higher burden than either cancer or cardiovascular disease [1]. In particular, lower respiratory infections, diarrhoeal diseases, HIV/AIDS and tuberculosis were amongst the five leading causes of global total years of life lost. More recently, the COVID-19 pandemic, declared as such by the World Health Organization (WHO) on 11 March 2020 after the emergence and global spread of the Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2), has caused more than 5 million deaths worldwide [2], making it one of the deadliest pandemics in history. Coronavirus has been responsible for three of the 18 major pandemics recorded throughout modern history [3], all occurring after the year 2000. *Yersinia pestis*, responsible for three plague pandemics, *Vibrio cholerae*, with seven cholera pandemics, and Influenza A virus, the causal agent of five flu pandemics, are responsible for the remaining, Influenza being the only other pathogen with a pandemic registered after the year 2000. Recent decades have also witnessed the emergence of additional virulent pathogens, including the Ebola virus, West Nile virus, Dengue virus and Zika virus, particularly in lower-income countries.

In addition to the emergence of virulent pathogens, the rise of Antimicrobial Resistance (AMR) poses a major threat to human health around the world. Besides tuberculosis, the global priority due to being the most common and lethal airborne AMR disease worldwide today, responsible for 250,000 deaths each year, it includes 12 groups of pathogens in three priority categories.

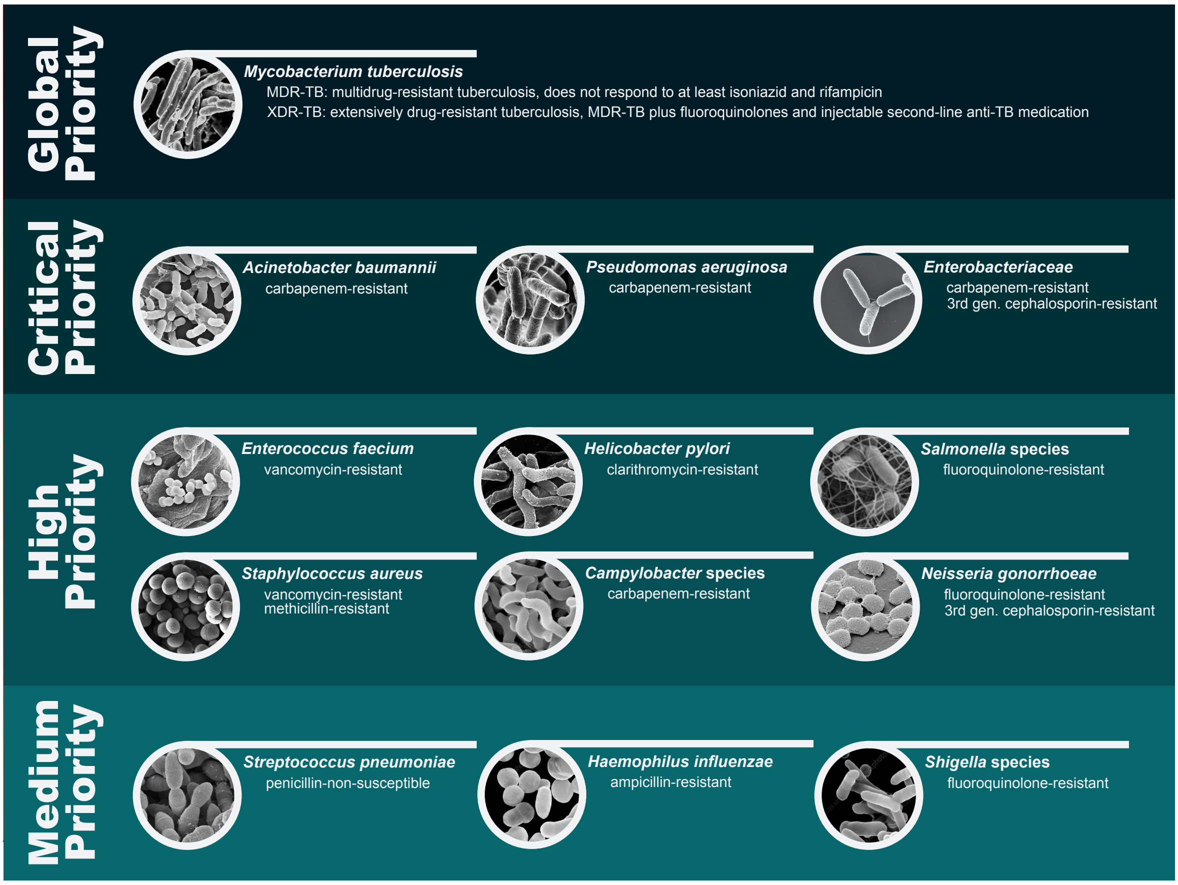


Figure 1.1: **World Health Organisation Global Priority Pathogens list.** This catalogue includes, besides *Mycobacterium tuberculosis* considered the number one global priority, a list of twelve microorganisms grouped under three priority tiers according to their antimicrobial resistance: critical (*Acinetobacter baumannii*, *Pseudomonas aeruginosa* and *Enterobacteriaceae*), high (*Enterococcus faecium*, *Helicobacter pylori*, *Salmonella* species, *Staphylococcus aureus*, *Campylobacter* species and *Neisseria gonorrhoeae*), and medium (*Streptococcus pneumoniae*, *Haemophilus influenzae* and *Shigella* species). The major objective was to encourage the prioritisation of funding and incentives, align research and development priorities of public health relevance, and garner global coordination in the fight against antimicrobial-resistant bacteria. Adapted from [4].

Clinical microbiology is a discipline focused on rapidly characterising pathogen samples to direct the management of individual infected patients (diagnostic microbiology) and monitor the epidemiology of infectious disease (public health microbiology), including the detection of outbreaks and infection prevention. According to the WHO Global Health Spending Report from 2000 to 2019, of the 51 countries that reported health spending by disease and condition, an average of 37% of health spending went to infectious and parasitic diseases, corresponding to the largest share of health spending [5]. About 21% of total health spending went to three major infectious diseases - HIV / AIDS (9%), tuberculosis (1%) and malaria (11%) - and 16% went to other infectious and parasitic diseases. On average, 70% of external health aid went to infectious and parasitic diseases in 51 low- and middle-income countries. Of the $54.8 billion estimated disbursed for health in 2020, $13.7 billion (25%) was targeted toward the COVID-19 health response [6].

1.1.1 Current standards for diagnostic in clinical microbiology

The past few decades have seen a major revolution in the operation of microbial laboratories, driven by the development of molecular technologies and ways to make these accessible, namely amplification-based Polymerase Chain Reaction (PCR), matrix-assisted laser desorption/ionisation - time of flight (MALDI-TOF) and DNA-microarray-based hybridisation technology. These are used in conjunction with traditional techniques such as microscopy, culture, and serology. The application of these methods differs by suspected infection type: bacterial, viral, fungal or parasitic. For the purpose of this dissertation work, we will focus on bacterial (see Section 1.1.1.1) and viral infections (see Section 1.1.1.2).

1.1.1.1 Bacterial infections

For patients with bacterial infections, the crucial steps are (1) to grow an isolate from a specimen, (2) identify its species, and (3) determine its pathogenic potential and test its susceptibility to antimicrobial drugs [7]. Together, this information facilitates the specific and rational treatment of patients. For public health purposes, knowledge also needs to be gained about (4) the relatedness of the pathogen to other strains of the same species to investigate transmission routes and allow recognition of outbreaks [8] (see Figure 1.2).

The current gold standard for bacterial pathogen identification in diagnostic microbiology laboratories involves the isolation of the pathogen through culture followed by biochemical testing, a multi-step process that can take days to weeks before obtaining results, depending on the fastidiousness of the organism and if it can be cultured [9, 10]. Although culture allows the identification of a wide variety of organisms, some pathogens can escape routine due to strict metabolic necessities for growth or the requirement for specific biochemical tests needed for their identification. Furthermore, results will be obscured if a mixed culture is obtained, mainly if the cultures are obtained from sites with microbiota, such as the gut and the skin, increasing the risk of contamination by normal flora and leading to false results [10]. After successful growth in culture, Gram staining and MALDI-TOF mass spectrometry are often used for identification with good accuracy as long as the pathogen is presented in the coexisting database [11]. An alternate rapid identification method is PCR where nucleic acid fragments are detected through specific primers, being highly sensitive and specific, to the point where PCR may detect bacteria that are not viable after a patient has been treated for an infection and it is limited to the primer used [12]. Syndromic panels, an extension of PCR by using multiple primers (multiplex PCR) to simultaneously amplify nucleic acids from multiple targets in a single reaction, tried to address this issue by allowing for the identification of multiple bacteria and other important information such as the detection of antibiotic resistance or virulence genes [10].

Following identification, antibiotic-susceptibility testing is essential to guide clinicians

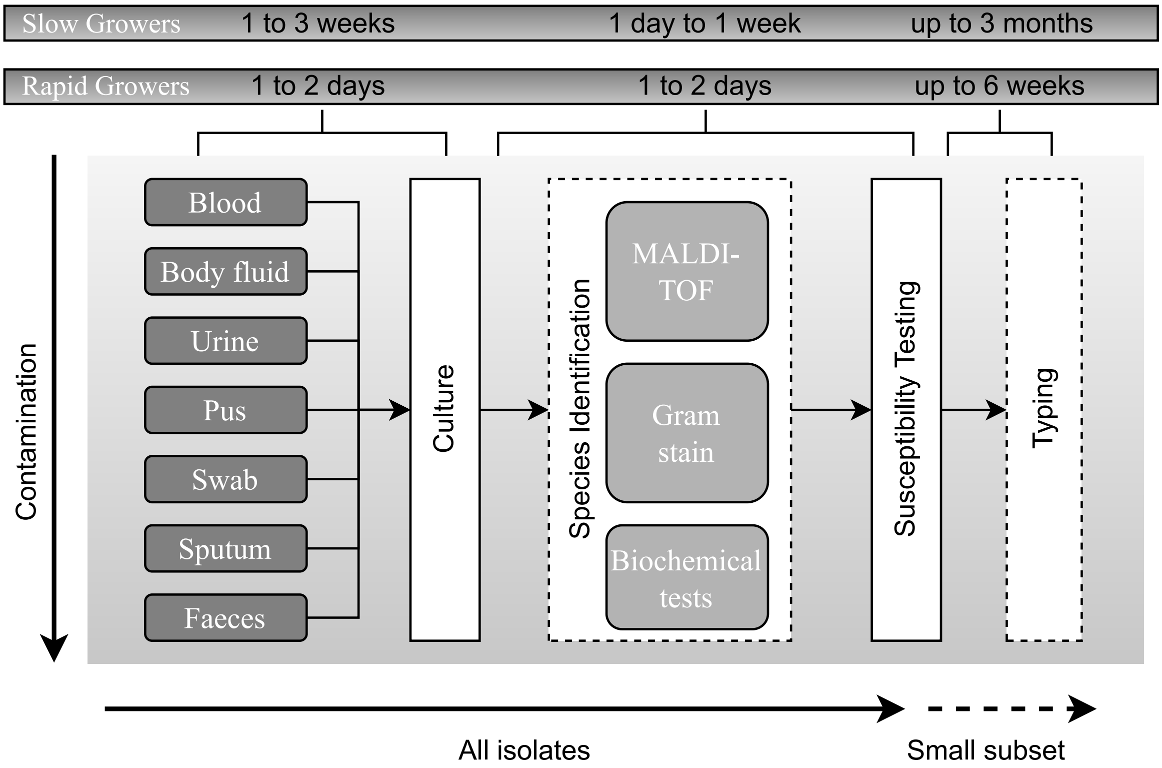


Figure 1.2: **Principles of current processing of bacterial pathogens.** Schematic representation of the current workflow for processing samples for bacterial pathogens is presented, with high complexity and a typical timescale of a few weeks to a few months. Samples that are likely to be normally sterile are often cultured on a rich medium that will support the growth of any culturable organism. Samples contaminated with colonising flora present a challenge for growing the infecting pathogen. Many types of culture media (referred to as selective media) are used to favour the growth of the suspected pathogen. Once an organism is growing, the likely pathogens are then processed through a complex pathway that has many contingencies to determine species and antimicrobial susceptibility. Broadly, there are two approaches. One approach uses MALDI-TOF for species identification prior to setting up susceptibility testing. The other uses Gram staining followed by biochemical testing to determine species; susceptibility testing is often set up simultaneously with doing biochemical tests. Lastly, depending on the species and perceived likelihood of an outbreak, a small subset of isolates may be chosen for further investigation using a wide range of typing tests. Adapted from [7].

in selecting an appropriate treatment. Conventional methods of bacterial resistance detection, such as disc diffusion, antimicrobial gradient strip, and broth microdilution, are widely used, but results cannot be obtained before 48 hours after receiving a sample, which can lead to prolonged use or overuse of broad-spectrum antibiotics [13]. Similarly to bacterial identification, MALDI-TOF and PCR have been increasingly adopted as solutions with shorter turnaround times, although no phenotypic information is recovered, nor information on the minimum inhibitory concentration (MIC) for a given antibiotic.

Choosing an appropriate bacterial typing technique for epidemiological studies depends on the available resources and the minimum intended resolution, ranging from DNA fingerprinting to multilocus sequence typing, Pulse Field Gel Electrophoresis (PFGE), and sequence-based typing (see section 1.2. A genomic approach to clinical microbiology) [8, 14]. DNA macrorestriction analysis by PFGE, which revolutionised precise separation of DNA fragments, became the most widely implemented DNA fingerprinting technique [14], becoming the golden standard for bacterial typing [15].

In the early 2000s, Multilocus Sequence Typing (MLST) was proposed as a portable, universal, and definitive method for characterising bacteria [16]. Instead of enzyme restriction of bacteria DNA, separation of restricted DNA bands using a PFGE chamber, followed by clonal assignment of bacteria based on banding patterns, MLST relies on the amplification through PCR sequences of internal fragments of housekeeping genes (usually 5 to 7), approximately 450-500 basepairs (bp) in size, followed by its sequence, usually my Sanger methods (see subsubsection 1.2.1.1. The first-generation of DNA sequencing). For each house-keeping gene, the different sequences present within a bacterial species are assigned as distinct alleles and, for each isolate, the alleles at each of the (usually) seven loci define the allelic profile or sequence type [17]. As with PFGE, different schemes, defining what housekeeping gene fragments are used, are available depending on the species. Unlike PFGE, the provision of freely accessible, curated databases of MLST nucleotide sequence data enables the direct comparison of bacterial isolates, providing the basis of a common language for bacterial typing [16]. So far, MLST schemes for more than 100 bacterial organisms have been published and made freely available[1], [18])

Depending on the organism identified, further and/or particular typing schemes can be applied. For *S. pneumoniae*, one of the pathogens listed in the WHO Global Priority Pathogens (GPP) list, the typing of the polysaccharide capsule, usually through Quellung reaction, is paramount for disease surveillance and evaluation of the pre- and post-pneumococcal vaccine, since the capsule, with over 90 serotypes reported, is the dominant surface structure of the organism and plays a critical role in virulence [19, 20]. For the *Salmonella* species, also in the GPP list, the serotype is usually determined by agglutination of the bacteria with specific antisera to identify variants of somatic (O) and flagella (H) antigens that, in various combinations, characterise more than 2600 reported serotypes [21].

1.1.1.2 Viral infections

Traditional approaches to the laboratory diagnosis of viral infections have been (1) direct detection in patient material of virions, viral antigens, or viral nucleic acids, (2) isolation of the virus in cultured cells, followed by identification of the isolate, and (3) detection and measurement of antibodies in patient serum (serology) [22]. Viral diagnostics is therefore generally organised into two primary categories, indirect and direct detection, depending on the method used (see Figure 1.3).

Indirect detection methods involve the propagation of virus particles through their introduction to a suitable host cell line (virus isolation), since viruses rely on host organisms to replicate. This is a relatively slow diagnostic method, sometimes taking weeks for the virus to propagate, usually followed by microscopy for its identification, or more commonly,

---

[1] `https://pubmlst.org/organisms`

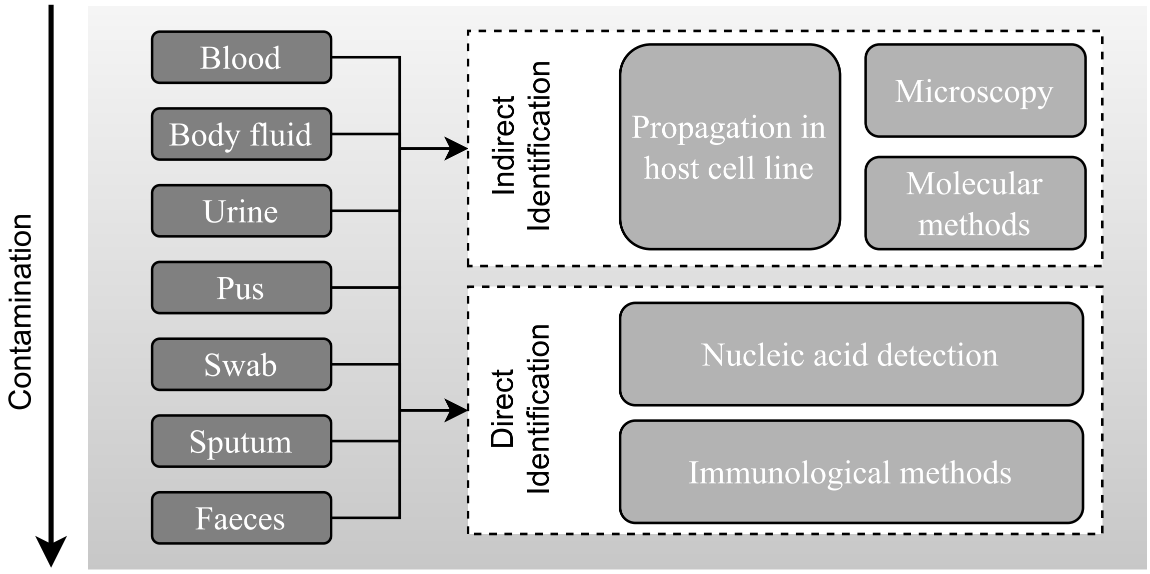


Figure 1.3: **Principles of current processing of viral pathogens.** Schematic representation of the current workflow for processing samples for viral pathogens is presented. Samples that are likely to be normally sterile are often cultured and isolated in suitable host cell lines (indirect identification). This supports the identification through microscopy of molecular methods, but the virus can take weeks to propagate. Direct identification is much faster, relying on nucleic acid detection of immunologic assays for the identification of the pathogen, without the need of virus propagation.

through molecular methods with an agent that detects a virus-associated protein, such as an antibody [23].

Direct detection methods negate the need for virus propagation, detecting the virus directly from the suspect source through nucleic acid and immunological methods. PCR and Reverse Transcription Polymerase Chain Reaction (RT-PCR) are widely applied methods for the detection of both DNA and Ribonucleic acid (RNA) viruses, respectively, driven by increased awareness of the clinical value of and demand for prompt information about viral loads, viral sequence data and potential antiviral resistance information [23]. Syndromic testing (see subsubsection 1.1.1.1. Bacterial infections) is now fully integrated into standard testing practises of many clinical laboratories [24]. The limitations of these assays include the absence of detection of off-target pathogens, a lack of full susceptibility information, cost, and false positive results. Real-Time Quantitative PCR (qPCR) remains the front line tool in aetiological diagnosis, measuring the production of the target amplicon throughout the reaction and providing quantitative results with high specificity and sensibility, albeit with a significant cost due to sophisticated apparatus despite high-throughput systems being widely established [23].

Immunoassays employ singular-epitope specificity antibodies as the primary means to detect viruses within a sample and provide a much more cost-efficient alternative to nucleic acid detection [23]. A major application is seroprevalence assays, an essential technique to identify patients who have been exposed to a virus (historical exposure), detect asymptomatic infection, or evaluate the efficacy of vaccines [25, 26]. Lateral Flow Immunoassays (LFIA) are widely used to detect virus-associated proteins directly from the source through antibodies labelled that binding to their cognate antigens, usually read by means of a colour change at a test line. In addition to being very cost-effective, LFIA have a turnaround time of minutes and the colour change can be observed with the naked eye, therefore facilitating rapid diagnosis, but their results are limited to semiquantitative and typically do not achieve sensitivity comparable to nucleic acid detection [23, 27, 28].

1.1.2 Surveillance and infection prevention in public health

Infectious disease surveillance is critical for improving population health, generating information that drives action not only in the management of infected patients but also in the prevention of new ones by identifying emerging health conditions that may have a significant impact by (1) describing the current burden and epidemiology of the disease, (2) to monitoring trends, and (3) identifying outbreaks and new pathogens [29, 30]. Public Health Surveillance Systems (PHSS) consist of the ongoing systematic collection, analysis and interpretation of data, and its integration with the timely dissemination of results to those who can carry out effective prevention and control activities [31].

Traditional PHSS can have different approaches based on the epidemiology and clinical presentation of the disease and the goals of surveillance. In passive surveillance systems, medical professionals in the community and health facilities report cases to the public health agency, which conducts data management and analysis once the data is received and communicates with the responsible entities. Globally, the WHO as described in the International Health Regulations what is notifiable by all countries, such as severe acute respiratory syndrome (SARS) and viral hemorrhagic fevers (Ebola, Lassa, Marburg), as well as guiding which public health measures should be implemented [32]. Active surveillance aims to detect every case, not relying on a reporting structure and can have many approaches from sentinel sites or networks of sites that capture cases of a given condition, such as respiratory tract infections, within a catchment population [30, 33]. The application of environmental surveillance methods, performed prospectively to detect pathogens prior to the recording of clinical cases or to monitor their abundance in the environment to assess the potential risk of disease, has been proven as a viable alternative, particularly in wastewater [34–37].

The emergence and reemergence of infectious diseases are closely linked to the biology and ecology of infectious agents, their hosts, and their vectors [38]. "One Health" is a collaborative and multi-disciplinary approach to designing and implementing programmes, policies, legislation and research in which multiple sectors communicate and work together to achieve better public health outcomes [39]. It recognises that people’s health is closely related to the health of animals and the shared environment, focussing on zoonotic and vector-borne diseases, antimicrobial resistance, food safety, food security, and environmental contamination [40]. This is crucial to (1) understanding the emergence and re-emergence of infectious and noncommunicable chronic diseases and (2) in creating innovative control strategies. A better understanding of the causes and consequences of certain human activities, lifestyles, and behaviours in ecosystems is crucial for a rigorous interpretation of disease dynamics and to drive public policies, but it requires breaking down the interdisciplinary barriers that still separate human and veterinary medicine from ecological, evolutionary, and environmental sciences [38].

1.2 A genomic approach to clinical microbiology

Since the publication of the first complete microbial genome, a quarter of a century ago, of the bacterium *Haemophilus influenzae* [41], genomics has transformed the field of microbiology, and in particular its clinical application (see Figure 1.4).

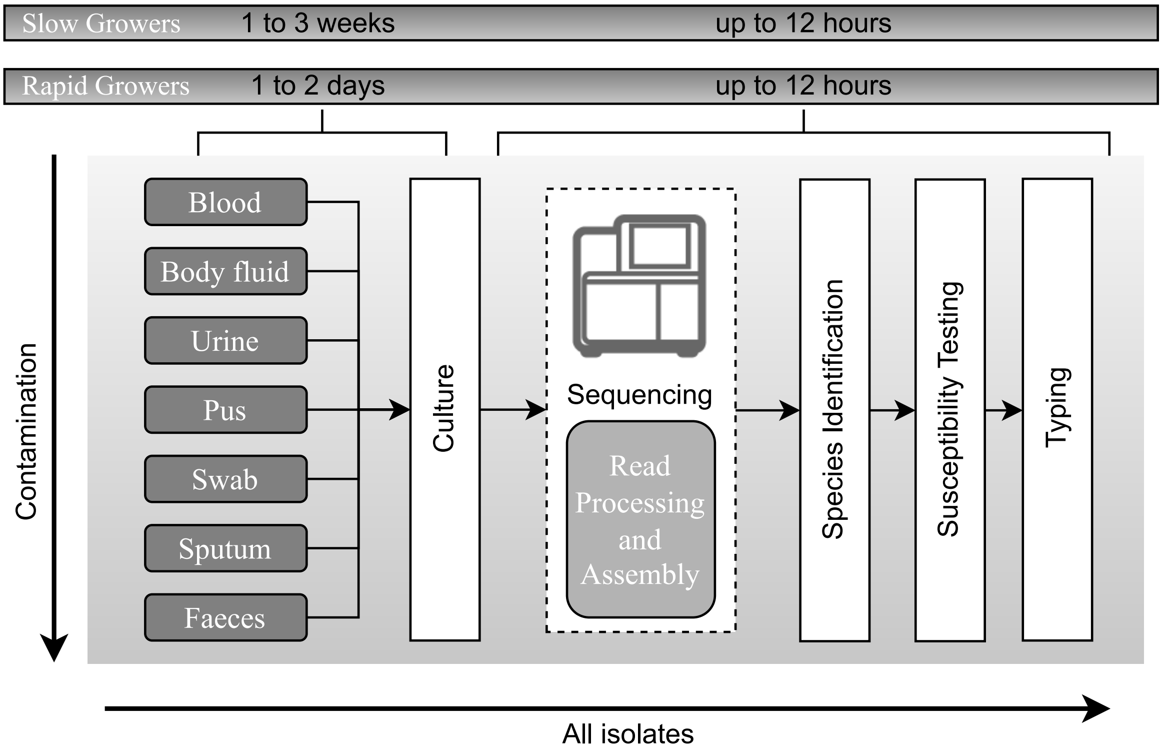


Figure 1.4: **Principles of current processing of bacterial pathogens based on whole genome sequencing.** Schematic representation of the workflow for processing samples for bacterial pathogens after the adoption of whole genome sequencing, with an expected timescale that could fit within a single day. The culture steps would be the same as currently used in a routine microbiology laboratory (see Figure 1.2). Once a likely pathogen is ready for sequencing, DNA is extracted, taking as little as 2 hours to prepare the DNA for sequencing. After sequencing, the main process for yielding information is computational. Automated sequence assembly algorithms are necessary for processing the raw sequence data, from which species, relationship to other isolates of the same species, antimicrobial resistance profile and virulence gene content can be assessed. All the results can also be used for outbreak detection and infectious diseases surveillance. Adapted from [7]

The paper describing the DNA-sequencing method with chain-terminating inhibitors used in the sequencing of the first microbial genome [42], earned the late Frederick Sanger his share of the 1980 Nobel Prize in Chemistry alongside Walter Gilbert, was, in 2014, the top fourth in the number of citations with over 60000, highlighting its impact in the field of biological sciences, and by extension medicine [43]. Currently, this number has increased to over 84000 according to PubMed Central® (PMC)[2][3]. Since its emergence, reductions in cost, technical advances in sequencing technologies, and new computational developments have made genomic sequencing one of the most influential tools in biomedical research, yielding unprecedented insights into microbial evolution and diversity, and the complexity of the genetic variation in both commensal and pathogenic microbes. The emerging application of genomic technologies in the clinic to combat infectious diseases is transforming clinical diagnostics and the detection and surveillance of outbreaks.

1.2.1 Twenty five years of microbial genome sequencing

Since the discovery of the structure of DNA [44], great strides have been made in understanding the complexity and diversity of genomes in health and disease. The development and commercialisation of high-throughput, massively parallel sequencing has democratised sequencing by offering individual laboratories, either in research or in health, access to the technology. Over the last quarter of a century, three main revolutions can be considered in genomic sequencing: the first, the second and the third generations of sequencing (see Figure 1.5).

1.2.1.1 The first-generation of DNA sequencing

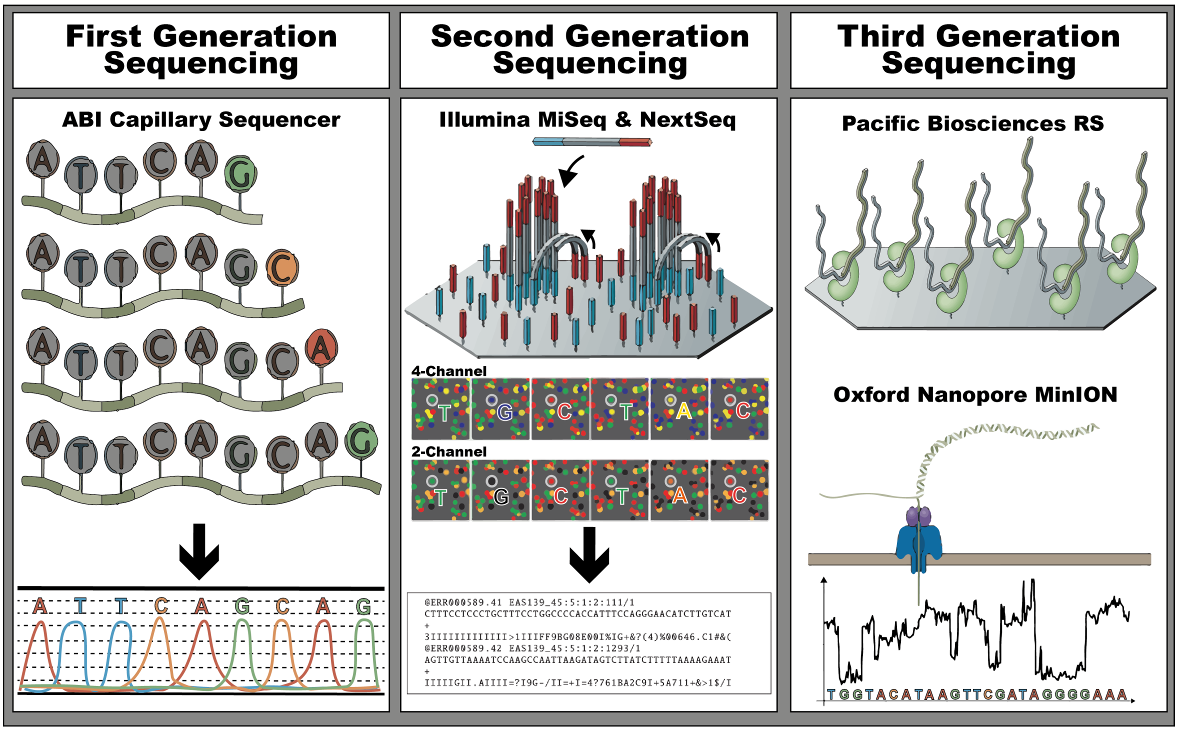
In the late 1980s, automated Sanger sequencing machines could sequence approximately 1,000 bases per day, having been applied in the 1990s to large bacterial genomes and the first unicellular and multicellular eukaryotic genomes [51]. The first genomes of pathogenic *Mycobacterium tuberculosis* [52], *Yersinia pestis* [53], *Escherichia coli* K-12 [54] were sequenced using this technology, requiring years of effort and significant budgets, but providing insights into the genomic complexity of these organisms. Some of the complete genome sequences produced during this era are still used today as high-quality references.

Simplistically, in first-generation DNA sequencing, also known as Sanger sequencing, a DNA polymerase is used to synthesise numerous copies of the sequence of interest using ddNTP in the reaction. At each nucleotide incorporation event, there is a chance that a ddNTP will be added and the growing DNA chain will terminate, resulting in a collection of DNA molecules of varying lengths [42, 45]. Modern Sanger sequencing uses fluorescently labelled ddNTP that allow the amplification step to be performed in a single reaction, resulting in a mixture of single-stranded DNA fragments of various lengths, each tagged at

---

[2] `https://pubmed.ncbi.nlm.nih.gov/`

[3] `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC431765/`



```
@ERR000589.41 EAS139_45:5:1:2:111/1
CTTTCCTCCCTGCTTTCCTGGCCCCACCATTTCCAGGGAACATCTTGTCAT
+
3IIIIIIIIIIIII>1IIIFF9BG08E00I%IG+&?(4)%00646.C1#&(
@ERR000589.42 EAS139_45:5:1:2:1293/1
AGTTGTTAAAATCCAAGCCAATTAAGATAGTCTTATCTTTTTAAAAGAAAT
+
IIIIIGII.AIIII=?I9G-/II=+I=4?761BA2C9I+5A711+&>1$/I
```

Figure 1.5: **The three revolutions in sequencing technology that have transformed the landscape of bacterial genome sequencing.** The first-generation, also known as Sanger sequencers, is represented by the ABI Capillary Sequencer (Applied Biosystems). During the sequencing reaction, at each nucleotide incorporation event, a fluorescently labelled ddNTP is incorporated, terminating the elongation of the DNA molecule. The resulting electropherogram for sequencing reaction is below and is read from left to right. The second-generation, also known as high-throughput sequencers, is represented by MiSeq, a 4-channel sequencer, and NextSeq, a 2-channel sequencer (Illumina), both sequencing by synthesis instruments. For both instruments, the loaded flowcell is sequenced in massive parallel reactions, with each nucleotide incorporation emitting a light signal that is captured and later basecalled into a fastq file, with an indication of the confidence of the call, presented below. In a 4-channel instrument, each nucleotide has its own marker (A: yellow, T: green, C: red, G: blue) but in a 2-channel instrument, only 2 markers exist (A: green plus red, T: green, C: red, G: no marker). These instruments allow the sequencing of both ends of the DNA fragment. Lastly, the third-generation, also known as long-read sequencers, is represented by the Pacific Bioscience BS sequencer and Oxford Nanopore MinION sequencer. In the first, immobilised polymerases in a SMRT Cell incorporating nucleotides with identifying fluorescent labels. In the latter, a nanopore embedded in a solid-state membrane causes a change in an ionic current across the membrane each time a nucleotide is pushed through the pore. This difference in potential is then used for basecalling. Adapted from [45–50]

one end with a fluorophore indicating the identity of the 3’ nucleotide that, after separation through capillary electrophoresis, the resulting electropherogram with four-colour fluorescence intensity can be interpreted by a base-calling software and producing 600–1000 bases of accurate sequence per reaction[45].

The first generation sequencing technology remains very useful for applications where high-throughput is not required due to its cost-effectiveness, relatively low sample load and accuracy of sequencing, even in repetitive genomic regions, although input DNA must consist of a relatively pure population of sequences [55]. One of the most common uses is thus individual sequencing reactions using a specific DNA primer on a specific template, such as MLST of bacterial genomes.

1.2.1.2 The second-generation of DNA sequencing

The release of the first truly high-throughput sequencing platform in the mid-2000s heralded a 50,000-fold drop in the cost of DNA sequencing in comparison with the first-generation technologies and led to the denomination of the second generation as next-generation sequencing (NGS) [47]. This trend has continued over the next two decades of continued development and improvement, allied to the emergence of benchtop sequencing platforms with high-throughput sequencing data and turnaround times of days, making it a standard in any microbiology and public health laboratories [46]. Second-generation sequencing methods can be grouped into two major categories: (1) sequencing by hybridisation and (2) sequencing by synthesis.

1.2.1.2.1 Sequencing by hybridisation

Sequencing by hybridisation, also known as sequencing by ligation, originally developed in the 1980s, relies on the binding of one strand of DNA to its complementary strand (hybridisation). By repeated hybridisation and washing cycles, it was possible to build larger contiguous sequence information, based on overlapping information from the probe hybridisation spot, being sensitive to even single-base mismatches when the hybrid region is short or if specialised mismatch detection proteins are present [55, 56]. Although widely implemented via DNA chips or microarrays, has largely been displaced by other methods, including sequencing by synthesis [47].

1.2.1.2.2 Sequencing by synthesis

Sequencing by synthesis methods is a further development of Sanger sequencing, without the ddNTP terminators, in combination with repeated cycles, run in parallel, of synthesis, imaging, and methods to incorporate additional nucleotides in the growing chain. All second-generation sequencing by synthesis approaches relies on a ‘library’ preparation using native or amplified DNA usually obtained by (1) DNA extraction, (2) DNA fragmentation and fragment size selection, and (3) ligation of adapters and optional barcodes to the ends of each fragment. This is generally followed by a step of DNA amplification. The resulting library is (4) loaded into a flow cell and (5) sequenced in massive parallel sequencing reactions [57]. Besides having much shorter read lengths than first-generation methods, with reads ranging from 45 to 300 bases,. These have an intrinsically higher error rate, the massively parallel sequencing of millions to billions of short DNA sequence reads allows the obtainment of millions of accurate sequences based on the identification of consensus (agreement) sequences [45, 47, 55].

Many of the approaches currently available for sequencing by synthesis methods have been described as cyclic array sequencing platforms, as they involve dispersal of target sequences across the surface of a two-dimensional array, followed by sequencing of those targets [45]. They can also be classified as single nucleotide addition or cyclic reversible termination or as single nucleotide addition [47].

The first relies on a single signal to mark the incorporation of a deoxynucleotide triphosphate (dNTP) into an elongating strand, avoiding the use of terminators. As a consequence, each of the four nucleotides must be added iteratively to a sequencing reaction to ensure that only one dNTP is responsible for the signal. The Roche 454 Life Sciences pyrosequencing device [4], was the first and most popular instrument implementing this technology, but discontinued since 2013 with support to the platform ceasing in 2016. This system distributes template-bound beads into a PicoTiterPlate along with beads containing an enzyme cocktail. As a dNTP is incorporated into a strand, an enzymatic cascade occurs, resulting in a bio-luminescence signal which is captured by a camera, which can be attributed to the incorporation of one or more identical dNTPs at a particular bead [47]. The ThermoFisher Ion Torrent system [5], released in 2010 and still available today, replaces the optical sensor, using instead H+ ions that are released as each dNTP is incorporated in the enzymatic cascade, and the consequential change in pH, to detect a signal [47]. Alongside the 454 pyrosequencing system, this system has difficulty in enumerating long repeats, additionally, the throughput of the method depends on the number of wells per chip, ranging from 10 megabases to 1000 megabases of 100 base reads in length, but with a very short run time (three hours) [45, 58].

The latter is defined by the use of terminator molecules that are similar to those used in the first-generation sequencing, preventing elongation of the DNA molecule, but unlike the first methods, it is reversible. To begin the process, a DNA template is primed by a sequence that is complementary to an adapter region, which will initiate polymerase binding to this double-stranded DNA region. During each cycle, a mixture of all four individually labelled and 3'-blocked dNTPs are added. After incorporation of a single dNTP into each elongating complementary strand, the unbound dNTPs are removed and the surface is imaged to identify which dNTP was incorporated at each cluster by optical capture. The fluorophore and blocking group can then be removed and a new cycle can begin [47]. The Illumina systems, which use this technology, account for the largest market share for sequencing instruments compared to other platforms[6], allowing paired-end sequencing and having the highest throughput (from 25 million reads for a MiSeq instrument to 1.2 billion reads for a NextSeq instrument[7]), with read lengths ranging from 45 to 300 bases in length with high accuracy, albeit with long running times (4 to 55 hours), rendering this technology a good choice for many sequencing applications where large read length is not required [45, 58, 59].

---

[4] `https://web.archive.org/web/20161226040638/http://454.com/`, snapshot from 26 December 2016

[5] `https://www.thermofisher.com/pt/en/home/brands/ion-torrent.html`

[6] `https://www.forbes.com/companies/illumina/?sh=774358a91aa6`

[7] `https://www.illumina.com/systems/sequencing-platforms.html`

1.2.1.3 The third-generation of DNA sequencing

Despite their wide adoption, second-generation methods require in the library preparation an enrichment or amplification step. These steps are time-consuming, introduce biases related to preferential capture or amplification of certain regions, and produce reads with relatively small size, making transversing repetitive genomic regions impossible if they are larger than the read length [45]. Third-generation sequencing technologies, also known as long-read sequencing or single-molecule sequencing, are characterised by the generation of ultra-long-reads, albeit at a much lower throughput than the second-generation [60]. They also have the potential to go beyond four-base sequencing to reveal genome-wide patterns of methylation and other chemical modifications that control the biology or the virulence of pathogens [61]. Currently, commercial long-read sequencing is supported by two companies: PacBio[8] and Oxford Nanopore Technologies (ONT)[9].

The basis of PacBio sequencers is known as SMRT, which takes place in single-use SMRT Cells. These contain multiple immobilised polymerases, which, after binding to an adaptor sequence, begin replication incorporating nucleotides with identifying fluorescent labels. The sequence of fluorescence pulses is recorded into a movie which is then converted into a nucleotide sequence. After the polymerase completes replication of one DNA strand, it continues to sequence the opposite adapter and second strand. As a result, multiple passes of the same template can be generated depending on the lifetime of the polymerase [46, 60]. This technology has accuracy comparable with the Illumina systems but requires a higher initial investment cost, are much larger machines in comparison with the benchtop counterparts, and have much lower throughput and longer library preparation protocols [60, 62].

ONT makes use of nanopores in small, portable single-molecule sequencing devices, capable of generating ultra-long sequences in real-time at a relatively low cost. Biological nanopores are embedded in solid-state membranes within disposable flow cells which, when a DNA strand passes through the pore driven by a motor protein, each nucleotide causes a change in an ionic current across the membrane, which is later base called [46, 60]. This process is free from fluorescence labels and amplification requirements, and after one strand is processed, the pore is available to sequence the next available strand. Sequence quality and length depend on the loaded library but are usually much lower than the alternative counterparts, and its throughput is dependent on the number and lifespan of the nanopore within the flowcell but is still much lower than the alternatives. Despite this, its portability, fast advances, and continued improvement of the flowcells make this a fast-adopted technology for long-read sequencing.

---

[8] `https://www.pacb.com/`

[9] `https://nanoporetech.com/`

1.2.2 DNA sequencing in clinical diagnosis and surveillance

WGS is becoming one of the most widely used applications of microbial genome sequencing. The major advantage of WGS is to yield all the available DNA information content on isolates in a single rapid step following culture (sequencing without culture will be discussed in the subsection 1.2.3. From genomics to metagenomics). In principle, after obtaining a pure culture, either bacterial (see subsubsection 1.1.1.1. Bacterial infections) or viral (see subsubsection 1.1.1.2. Viral infections), the data from sequencing contain all the information currently used for diagnostic and typing needs, and much more, thus opening the prospect for large-scale research into pathogen genotype-phenotype associations from routinely collected data [7]. The cost of producing massive amounts of information requires a new framework with expert handling and processing of computer-driven genomic information, as well as capable computational infrastructures (see Section 1.3), but through this technology, researchers and clinicians can obtain the most comprehensive view of genomic information and associated biological implications, transforming clinical diagnostics and the detection and surveillance of outbreaks. [47, 63, 64].

Targeted sequencing is also proving invaluable to clinical microbial and research, not only by allowing more individual samples to be sequenced within a single run, significantly reducing costs and the amount of data generated, but also, due to the smaller target size, obtaining results with very high confidence due to the high coverage obtained [47]. This has been particularly useful in viral genomics where sections, such as the capsid, or the complete viral genome can be selectively targeted directly from the suspected sample, offering a more time-effective method to achieve the same output as traditional nucleic acid amplification methods [23].

1.2.2.1 Sequencing in the routine laboratory workflow

WGS has been used in the routine laboratory workflow when typing of pathogens by a method having the highest possible discriminatory power is required either through Single Nucleotide Polymorphism (SNP) or core-genome/whole genome Multilocus Sequence Typing (cg/wg MLST) analysis, for example during hospital outbreaks [65].

The implementation of WGS in routine diagnostics requires several adaptations in the laboratory workflow, from the ‘wet’ laboratory part (extraction, library preparation, sequencing), to the ‘dry’ bioinformatics part where genomic data is analysed and its results interpreted by specialised personnel [66].

Currently, sequencing technologies are used in a case-by-case approach, with their adoption being much more present in a research setting than in a diagnostic one. Sequencing is mostly used after a diagnostic through the identification of the causative agent has already been performed. Although substantial advances have been made in reducing response time, most of the current systems do not yet generate enough data fast enough for a truly rapid response for it to be used in the clinical setting [47]. High-throughput DNA sequencing has found additional new applications in drug discovery and in functional genomics with, for example, SNP-based analysis to identify new drug targets [46].

Although the second-generation DNA sequencing methods have shed light on fundamental aspects of microbial ecology and function, they suffer from issues associated with short read length (see 1.2.1.2) and cannot reliably reconstruct long repeats because of uncertainties in mapping read, even when paired-end sequencing is used. Third-generation sequencing methods (see 1.2.1.3 The third-generation of DNA sequencing) have become increasingly used in microbiology, although their accuracy and low throughput make them difficult to implement in a clinical diagnostic setting.

1.2.2.2 Sequencing and genomic surveillance

Most notably, WGS has become a common tool in infection surveillance and prevention, allowing identification and tracking of pathogens, establishing transmission routes and outbreak control [67]. In bacterial infections, initiatives such as Pathogenwatch[10] offers a web-based platform for AMR analysis and phylogeny generation of *Campylobacter*, *Klebsiella*, *Neisseria gonorrhoeae*, *Staphylococcus aureus*, and *Salmonella Typhi* [68]. The Center for Genomic Epidemiology website[11] offers services for phylogenetic tree building and AMR prediction. Chewie Nomenclature Server[12] allows users to share genome-based gene-by-gene typing schemas and to maintain a common nomenclature, simplifying the comparison of results [69]. Enterobase[13] allows for the analysis and visualisation of genomic variation within enteric bacteria [70]. Microreact[14], from the same developers as Pathogenwatch, combines clustering, geographical and temporal data into an interactive visualisation with trees, maps, timelines and tables for a multitude of microorganisms, both bacterial and viral [71]. Particularly for viruses, GISAID[15] promotes the rapid sharing of data from all influenza viruses and the coronavirus causing COVID-19, including the genetic sequences and related clinical and epidemiological data [72]. ViPR[16] provides access to sequence records, gene and protein annotations, immune epitopes, 3D structures, host factor data, and other data types for over 14 viral families, including *Coronaviridae*, from which SARS-CoV-2 belongs to, and *Faviviridae*, the family of Dengue and Zika virus [73]. INSaFLU[17] supplies public health laboratories and influenza researchers with a web-based suite for effective and timely influenza and SARS-CoV-2 laboratory surveillance, identifying the type and

---

[10] `https://pathogen.watch/`

[11] `https://www.genomicepidemiology.org/`

[12] `https://chewbbaca.online/`

[13] `https://enterobase.warwick.ac.uk/`

[14] `https://microreact.org/`

[15] `https://www.gisaid.org/`

[16] `https://www.viprbrc.org/`

[17] `https://insaflu.insa.pt/`

subtype/lineage, detection of putative mixed infections and intra-host minor variants [74]. Nextrain[18] provide a continually-updated view of publicly available data alongside powerful analytic and visualisation tools to aid epidemiological understanding and improve outbreak response for 10 pathogens: Influenza, SARS-CoV-2, West Nile virus, Mumps, Zika, West African Ebola, Dengue, Measles, Enterovirus D68 and Tuberculosis [75].

In outbreak detection and surveillance, genetic sequencing techniques combined with epidemiological data have undoubtedly provided immeasurable insights regarding evolutionary relationships and transmission pathways in various environments [76, 77]. In a pandemic setting, this approach, although not novel, has been revolutionary, particularly in the COVID-19 setting.

In the 2009 swine-origin Influenza A H1N1 pandemic, the first complete genome was publicly available on the 25 of April 2009 (GenBank accession number FJ966079), about a month after records of increased flu activity in Mexico and 10 days after the first confirmed cases in California, United States of America [78, 79]. By the time the pandemic was declared, on 11 June 2009, [78] reported the origins and evolutionary genomics of the pandemic influenza A variant with a collection of 813 complete influenza genome sets, 17 of which belong to the new swine influenza viruses (GenBank accessions numbers GQ229259–GQ229378). The Middle East Respiratory Syndrome (MERS) pandemic, declared as such in 2015 [3], had its first publicly available sequence on 5 of July 2015 (GenBank accession number KT006149)[80], with a sequence from a camel, thought to be an intermediate host for the virus, available as early as 7 of March 2016 (GenBank accession number KU740200) [81, 82].

The SARS-CoV-2 has brought a new meaning to genomic surveillance, with the first sequence from a COVID-19 patient being made publicly available as early as 12 January 2020 from a case of respiratory disease from the Wuhan outbreak (GenBank accession number MN908947) [83]. At the date of the pandemic declaration by WHO, at 11 March 2020, over 400 complete SARS-CoV-2 sequences were deposited on GISAID[19], hitting over one million sequences in April 2021 [84]. Currently, over 8 million complete viral sequences are available at GISAID[20], being one of the most highly sequenced genomes of any organism on the planet. This richness in genomic information has been basal to identifying new variants of risk and new variants of concern with a myriad of different origins, identifying routes of transmission across borders, including the identification of "super-spreaders" events, and informing infection control measures [76, 77, 85].

---

[18] `https://nextstrain.org/`

[19] `http://web.archive.org/web/20200311053731/https://www.gisaid.org/`

[20] `https://www.gisaid.org/`

1.2.3 From genomics to metagenomics

Despite the increasing adoption of DNA sequencing methods in clinical microbiology, the sequencing of genetic material from a pure culture requires *a priori* knowledge of what to expect from a particular clinical sample or patient [86]. In most cases, this knowledge is enough to request the most appropriate test, such as syndromic panels or specific culture media, but this is not always the case. In recent years, there has been a growing interest in using metagenomics to deliver culture-independent approaches to microbial ecology, surveillance and diagnosis (see Figure 1.6)[46, 87]. Metagenomic DNA sequence allows detailed characterisation of pathogens in all kinds of samples originating from humans, animals, food and the environment, ligating the diagnostics to surveillance in a true "one health" fashion [88]. Unlike PCR or microarrays, it usually does not require primer or probe design, it can be easily multiplexed, and the specificity and selectivity of the sequencing can be adjusted computationally after acquiring the data [89] (see 1.3). While most molecular assays target only a limited number of pathogens, metagenomic approaches characterise all DNA or RNA present in a sample, enabling analysis of the entire microbiome as well as the human host genome or transcriptome in patient samples [90]. Whether or not it can entirely replace routine microbiology depends on several conditions and future developments, both technological and computational.

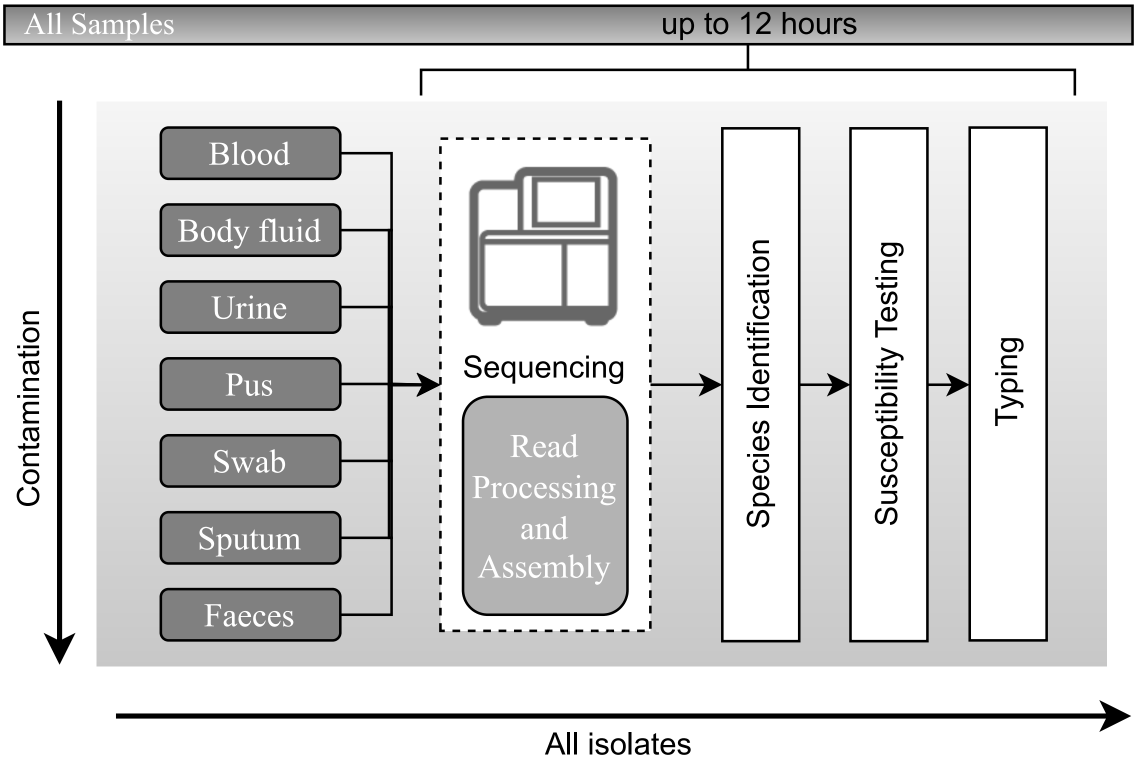


Figure 1.6: **Hypothetical workflow based on metagenomic sequencing.** Schematic representation of the hypothetical workflow for the direct processing of samples from suspected pathogen sources after adoption of metagenomic sequencing, with an expected timescale that could be accommodated in a single day. Adapted from [7].

Albeit lacking consensus in the field, metagenomics can be classified into two variants as proposed by [91]: (1) metaxonomics where marker genes ubiquitous in many taxa are targeted and sequenced, and (2) the untargeted "shotgun" sequencing of all microbial genomes present in a sample.

1.2.3.1 Metataxonomics and Targeted Metagenomics

Molecular barcoding approaches can be combined with second-generation high-throughput sequencing to achieve unprecedented depths of coverage in microbial community profiling, being defined as metataxonomics. For profiling bacterial species, the most popular approach is 16S ribossomal RNA (rRNA) gene sequencing, an 1500 base pair gene that encodes catalytic RNA that is part of the 30S ribosomal subunit. Traditionally, the variable regions of the 16S rRNA gene (V-regions) are targeted, or ranges thereof (V1-V2, V1-V3, V3-V4, V4, V4-V5, V6-V8, and V7-V9), and are specific to bacterial genus (96%) and for some, even species (87.5%), [92, 93]. Moreover, dedicated 16S databases that include near full-length sequences for a large number of strains and their taxonomic placements exist, such as RDP[21], Greengenes[22], silva[23] and NCBI's 16S ribosomal RNA project[24] [94–96]. The sequence of an unknown strain can be compared with the sequences in these databases after very closely related sequences are grouped into Operational Taxonomic Units (OTUs), an operational definition used to classify groups of closely related individuals. This allows the deduction of probable taxonomy, with the assumption that sequences of >95% identity represent the same genus, whereas sequences of >97% identity represent the same species [97].

Furthermore, it must necessarily account for intragenomic variation between 16S gene copies. Targeting 16S variable regions with short-read sequencing platforms cannot achieve the taxonomic resolution afforded by sequencing the entire gene and is limited by the database chosen [98]. The emergence of third generating sequencing technologies (see sub-subsection 1.2.1.3. The third-generation of DNA sequencing) allows for this limitation to be overcome but currently, only a fraction of the databases includes complete 16S rRNA sequences.

While viruses are an integral part of the microbiota, no universal viral marker genes are available to perform such taxonomic assignments. Amplification of whole viral genomes is possible and, in 2015, RNA extracted from whole blood, serum, re-suspended swabs and urine, after targeted amplification of the whole viral genome, proved invaluable in the track of the Ebola virus disease epidemic in West Africa, responsible for >11 thousand deaths, allowing for the characterisation of the infectious agent the determination of its evolutionary

---

[21] `http://rdp.cme.msu.edu/`

[22] `https://greengenes.secondgenome.com/`

[23] `https://www.arb-silva.de/`

[24] `https://www.ncbi.nlm.nih.gov/refseq/targetedloci/`

rate, signatures of host adaptation, identification and monitoring of diagnostic targets and responses to vaccines and treatments [99]. As an alternative, broad scope viral targeted sequence capture (TSC) panels offer depletion of background nucleic acids and improve the recovery of viral reads by targeting coding sequences from a multitude of viral genera, such as VirCapSeq-VERT Capture Panel[25] but do not guarantee the full recovery of the viral genome, and can present biases towards certain genera [100, 101].

1.2.3.2 Shotgun Metagenomics

SMg can offer relatively unbiased pathogen detection and characterisation. The capacity to detect all potential pathogens — bacteria, viruses, fungi and parasites — in a sample has great potential utility in the diagnosis of infectious disease [90], potentially able to provide genotyping, antimicrobial resistance and virulence profiling in a single methodological step. This comes with the cost of producing massive amounts of information that require expert handling and processing, as well as capable computational infrastructures [66, 102].

Clinical applications of SMg derive their roots from the use of microarrays (see subsection 1.1.1. Current standards for diagnostic in clinical microbiology), which it was successfully applied in in-depth microbiome analysis of different sites in the human body, it was the emergence of the second-generation sequencing technology and its high throughput of genomic data at a competitive price that made sequencing of all genomic content, DNA and/or RNA, in a clinical sample a viable possibility for diagnostics (see subsubsection 1.2.1.2. The second-generation of DNA sequencing The second-generation of DNA sequencing) [90, 103, 104]. The first reported case that demonstrated the utility of SMg was in 2014 with the clinical diagnosis of neuroleptospirosis in a 14-year-old immunodeficient and critically ill boy with meningoencephalitis by Wilson et al [105], prompting appropriate targeted antibiotic treatment and eventual recovery of the patient. In this case, traditional methods, including an invasive brain biopsy, failed to provide answers, until the shotgun sequencing of cerebrospinal fluid identified 475 of 3,063,784 sequence reads (0.016%) corresponding to leptospira, for which clinical assays were negative due to its very low abundance. Ever since many other reports of successful application of SMg in clinical metagenomics have been reported. but all in edge cases where traditional diagnostic methods have failed or as proof-of-concept [102, 106–109].

In public health microbiology, SMg combined with transmission network analysis allowed the investigation and quick action on the food supply of the 2013 outbreak of Shiga toxin-producing *Escherichia coli* (STEC) strain O104:H4 from faecal specimens obtained from patients [110]. A similar approach was followed in the detection of *Salmonella enterica* subsp. *enterica* serovar Heidelberg from faecal samples in two thought to be unrelated outbreaks in the United States of America, as well as the *in situ* abundance and level of

---

[25] `https://sequencing.roche.com/content/dam/rochesequence/worldwide/resources/brochure-vircapseq-vert-capture-panel-SEQ1000117.pdf`

intrapopulation diversity of the pathogen, and the possibility of co-infections with *Staphylococcus aureus*, overgrowth of commensal *Escherichia coli*, and significant shifts in the gut microbiome during infection relative to reference healthy samples [111]. More recently, shotgun metagenomic sequencing has evidenced alterations in the gut microbiota of a subset of COVID-19 patients that present the uncommon gastrointestinal (GI) symptoms, shedding a higher understanding of gut–lung axis affecting the progression of COVID-19 [112].

Clinical diagnostic applications have lagged behind research advances. A significant challenge with the shotgun metagenomic approach is the large variation in the pathogen load between patient samples, as evidenced in the studies presented. A low pathogen load and high contamination of host DNA or even the present microbiome may result in enough data to produce the high-resolution subtype needed to distinguish and cluster the cases that were caused by the same outbreak pathogen source, or, extremely, the undetection of the causative agent [90, 113]. Differential lysis of human host cells followed by degradation of background DNA has proven an effective method to reduce host contamination, but limitations include potential decreased sensitivity for microorganisms without cell walls, such as *Mycoplasma* spp. or parasites; a possible paradoxical increase in exogenous background contamination by use of additional reagent [114–116]. Additionally, it is often unclear whether a detected microorganism is a contaminant, coloniser or *bona fide* pathogen, and the lack of golden standards remains one of the biggest challenges when applying these methods in clinical microbiology for diagnosis.

In addition to negative controls, already a common practice in any sequencing assay and in particular in metataxonomics (see subsubsection 1.2.3.1. Metataxonomics and Targeted Metagenomics), positive controls can be a way to circumvent the lack of golden standards, either through the spike of the samples with a known amount of a specific DNA/RNA or though the sequencing of samples with known composition and abundance. Well-characterised reference standards and controls are needed to ensure the quality and stability of the SMg assay over time [90, 117]. Most available metagenomic reference materials are highly tailored to a specific application. For example, the ZymoBIOMICS Microbial Community Standard[26] is the first commercially available standard for microbiomics and metagenomics studies, providing a mock community with defined composition and abundance consisting of Gram-positive, Gram-negative and yeast. It is useful to determine the limit of detection of an assay, and the effectiveness and biases of a given protocol. Standards with a more limited spectrum of organisms are also available, such as the National Institute of Standards and Technology (NIST)[27] reference materials for mixed microbial DNA detection, which contain only bacteria. Thus, these materials may not apply to untargeted SMg analyses.

---

[26] `https://www.zymoresearch.com/collections/zymobiomics-microbial-community-standards`

[27] `https://www.nist.gov/`

1.3 The role of bioinformatics

As stated previously (see section 1.2. A genomic approach to clinical microbiology and subsection 1.2.3. From genomics to metagenomics), one of the biggest challenges when dealing with genomic, and in particular metagenomic, data is the lack of golden standards. This is also applicable to the bioinformatic analysis, required due to the amount of data produced by genomic sequencing technologies. This is currently one of the bottlenecks in the deployment of sequencing technology in clinical microbiology as there is no standard on how to deal with the increasing amount of data produced in a fit-for-purpose manner [118].

Bioinformatics is an interdisciplinary research field that applies methodologies from computer science, applied mathematics and statistics to the study of biological phenomena[118]. With the widespread use and continuous development of sequencing technologies, bioinformatics has become a cornerstone in modern clinical microbiology. Major efforts are being made on the standardisation and assessment of software for the analysis of genomic data, both commercial and open-source [102, 119–121].

1.3.1 From molecules to reads

In all sequencing technologies (see subsection 1.2.1. Twenty five years of microbial genome sequencing), many copies of the source DNA are randomly fragmented and sequenced. To these sequences, we refer to as reads. In the case of second-generation sequencing (see subsubsection 1.2.1.2. The second-generation of DNA sequencing The second-generation of DNA sequencing), one or both ends of the fragment can be sequenced. If a fragment is sequenced from one end, we refer to it as single-end sequencing. If a fragment is sequenced on both ends, spanning the entire fragment, it is called paired-end sequencing.

1.3.1.1 The FASTQ file

All sequencing technologies, regardless of generation, produce data in the same standard file format: the FASTQ, a text-based format for storing both a biological sequence (usually nucleotide sequence) and its corresponding quality scores [122]. Originally developed at the Wellcome Trust Sanger Institute, the FASTQ has emerged as a common file format for sharing sequencing read data (see 1.4). The FASTQ can be considered as an extension of the ‘FASTA sequence file format’, originally invented by Pearson et al [123], which includes just the sequence information. A FASTQ file normally uses four lines per sequence:

- **Line 1** begins with a ’@’ character and is followed by a sequence identifier and an optional description;

- **Line 2** is the raw sequence letters;
- **Line 3** begins with a '+' character and is optionally followed by the same sequence identifier (and any description) again;
- **Line 4** encodes the quality values for the sequence in Line 2 and must contain the same number of symbols as letters in the sequence.

In FASTQ both the sequence letter and quality score are each encoded with a single ASCII character for brevity. The quality of a sequence in a FASTQ file is represented by a quality value Q is an integer mapping of p, where p is the probability that the corresponding base call is incorrect (see Table 1.1). This is called the PHRED score [124] and is defined by the following equation:

$$\mathbf{Q}\text{PHRED} = -10 \times \log \mathbf{P} \tag{1.1}$$

The PHRED quality scores **Q** is defined as a property which is logarithmically related to the base-calling error probability **P**.

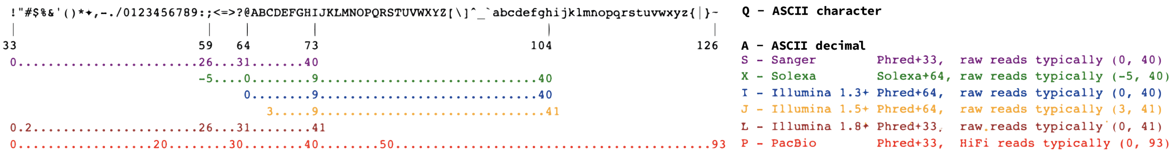
Table 1.1: **PHRED quality scores are logarithmically linked to error probabilities.** A PHRED Score of 20 indicates the likelihood of finding 1 incorrect base call among 100 bases. In other words, the precision of the base call is 99%. **Q** scores are classified as a property that is associated logarithmically with the probabilities of base calling error **P**.

Phred Quality Score	Probability of incorrect base call	Base call accuracy
10	1 in 10	90%
20	1 in 100	99%
30	1 in 1000	99.90%
40	1 in 10,000	99.99%
50	1 in 100,000	100.00%
60	1 in 1,000,000	100.00%

Since their introduction, PHRED scores have become the *de facto* standard for representing sequencing read base qualities [122]. Despite this convention, the encoding of the PHRED score can vary when translated to its ASCII representation in the FASTQ file format. For example, Sanger FASTQ files use ASCII 33–126 to encode PHRED qualities from 0 to 93 (that is, PHRED scores with an ASCII offset of 33). A full list of encoding is available in Figure 1.7.

1.3.1.2 FASTQ file simulation

With the lack of golden standards for metagenomic analysis, the use of simulated mock communities, with known composition, abundance, and genomic information, provides a



```
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz{|}~   Q - ASCII character
 |                         |    |        |                              |                     |
33                        59   64       73                            104                   126   A - ASCII decimal
 0........................26...31.......40                                                        S - Sanger        Phred+33,  raw reads typically (0, 40)
                          -5....0........9.............................40                         X - Solexa        Solexa+64, raw reads typically (-5, 40)
                                0........9.............................40                         I - Illumina 1.3+ Phred+64,  raw reads typically (0, 40)
                                   3.....9..............................41                        J - Illumina 1.5+ Phred+64,  raw reads typically (3, 41)
 0.2......................26...31........41                                                       L - Illumina 1.8+ Phred+33,  raw reads typically (0, 41)
 0..................20........30........40........50..........................................93  P - PacBio        Phred+33,  HiFi reads typically (0, 93)
```

Figure 1.7: **Range of FASTQ quality scores and their corresponding ASCII encoding.** For raw reads, the range of scores will depend on the technology and the base caller used. Starting in Illumina 1.8, the quality scores have returned to the use of the Sanger format (PHRED+33). For processed reads and long accurate reads, scores may be even higher with, for example, quality values of up to 93 observed in reads from PacBio HiFi reads.

ground truth against which success evaluations can be made. Given their standard structure and adoption, the generation of simulated FASTQ files from a reference or a set of references is very straightforward.

Multiple computational tools have been developed in recent years for the simulation of sequencing data, particularly for second and third-generation sequencing technologies, which could be used to compare existing and new bioinformatic analytical pipelines. [125] provides a comprehensive assessment of 23 different read-simulation tools, highlighting their distinct functionality, requirements, and potential applications, as well as providing a selection of suggestions for different simulation tools depending on their purpose. For *in silico* genomic and metagenomic sequence generation, a plethora of tools are available for first, second and third-generation reads (see Figure 1.8).

1.3.1.3 FASTQ quality assessment and quality control

Quality assessment and control is a basal step to any analysis and aims to (1) remove and/or filter low quality and low complexity reads, (2) trim adapters, and (3) remove host sequences from the samples' raw data. There are many tools available but the most commonly used are FastQC[28] (Babraham Bioinformatics) for quality control, followed by Trimmomatic [126], Cutadapt [127] or fastp [128] to trim and/or filter adaptors, low quality and low complexity sequences. For long-read sequencing, tools like NanoPlot and NanoStats [129], and Filtlong[29] can perform the equivalent quality assessment and control, adapter trimming and low quality trimming, respectively.

1.3.1.4 Direct taxonomic assignment and characterisation

A piece of important information that can be retrieved directly from the quality-controlled read data: (1) the identification and characterisation of the microbes present in a sample and (2) their relative abundance. Taxonomic classification methods can vary depending on the sequencing methodology used: pure culture, metataxonomics and amplicon

---

[28] `https://www.bioinformatics.babraham.ac.uk/projects/fastqc/`

[29] `https://github.com/rrwick/Filtlong/`

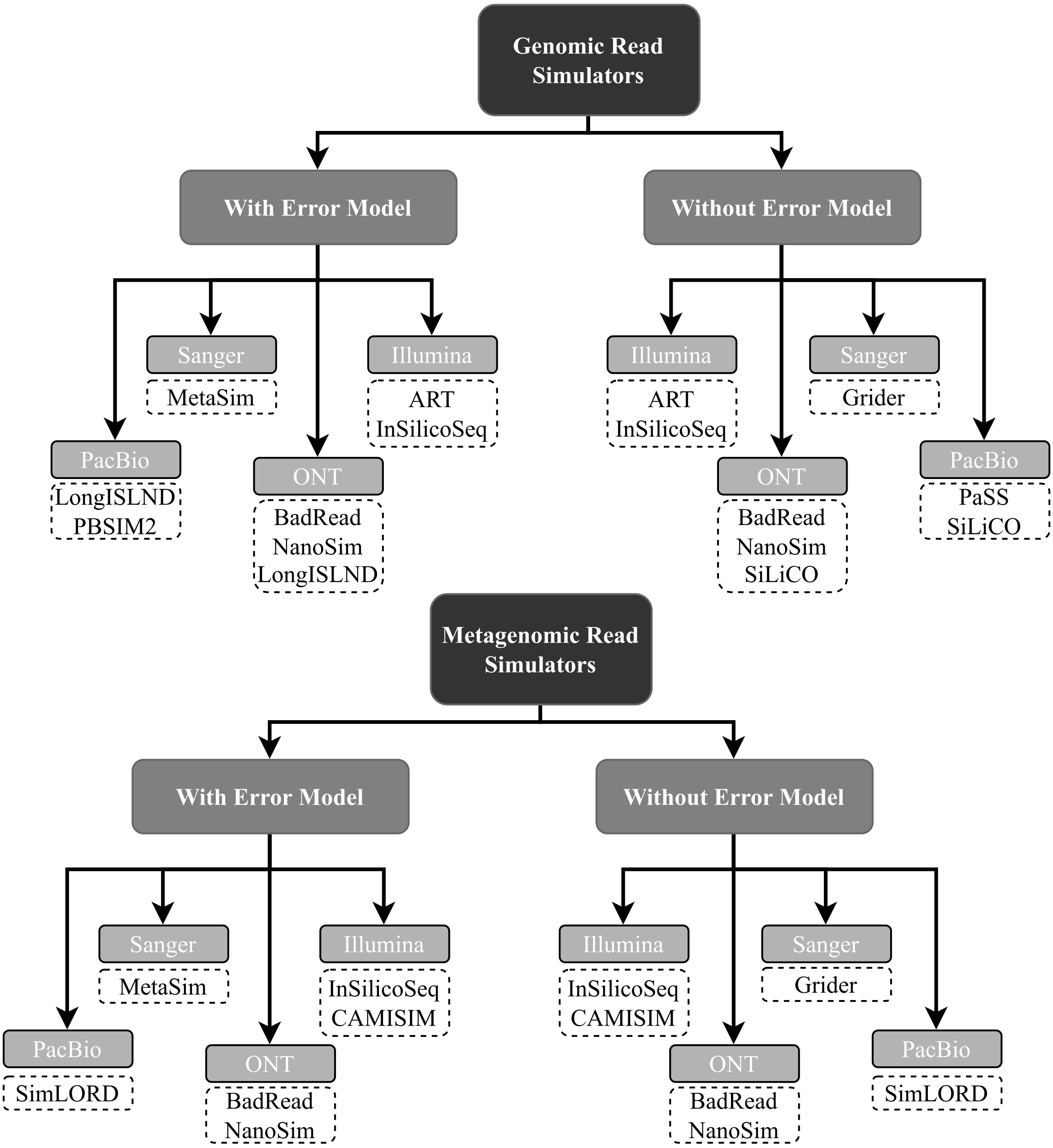


Figure 1.8: **Sequence simulators for genomic and metagenomic data.** For first generation sequencing, Metasim (`https://github.com/gwcbi/metagenomics_simulation`) and Grider (`https://sourceforge.net/projects/biogrinder/`) can generate mock genomic and metagenomic data, with and without error models, respectively. For Illumina data, ART (`https://www.niehs.nih.gov/research/resources/software/biostatistics/art/index.cfm`), InSilicoSeq (`https://github.com/HadrienG/InSilicoSeq`) and CAMISIM (`https://github.com/CAMI-challenge/CAMISIM`) represent options for in silico data generation. Due to their differences, the third-generation PacBio and Oxford Nanopore (ONT) have distinct software for in silico data generation. The first can be accomplished by LongISLND (`https://bioinform.github.io/longislnd/`) and PBSIM2 (`https://github.com/yukiteruono/pbsim2`) got genomic data, and SimLORD (`https://bitbucket.org/genomeinformatics/simlord/src`) for metagenomic data, with and without error model. The latter BadRead (`https://github.com/rrwick/Badread`) and NanoSim (`https://github.com/bcgsc/NanoSim`) can generate genomic and metagenomic *in silico* data, with and without error model. Additionally, for genomic data, LongISLND and SiLiCO (`https://github.com/ethanagb/SiLiCO`) generate data with and without error, respectively. Adapted from [125].

metagenomics, and shotgun metagenomics.

From pure culture, taxonomic identification of the read content of a sample is useful to assess contamination. Tools like Kraken2 [130, 131] and Braken [132] rely on a database to assign taxonomic labels to reads and are therefore biased to the contents of the database used. Various databases are available[30], varying in size and content (archaea, bacteria, viral, plasmid, human and eukaryotic pathogens) and therefore in sensitivity depending on the resources available and the purpose intended. Alternatively, there are options to create custom databases.

These tools are also extremely useful to assess the contents of a metagenomic sample. Alternatives such as Midas [133], Kaiju, [134], and MetaPhlAn2 [135] offer the same analysis as Kraken2 and Bracken using different algorithms, and with the disadvantage that they come prepackaged with their own databases, without the option to create a tailored database, limiting their applicability. Kaiju differs from the other tools by using a protein reference database, instead of nucleotide, but no pre-built version is available, requiring significant resources to build and index the database pre-use. Long-read data from third-generation sequencing technologies (see subsubsection 1.2.1.3. The third-generation of DNA sequencing) can be treated as single-end reads, and all mentioned tools accommodate the classification of single-end files.

1.3.2 From reads to genomes

Due to the limitations of current sequencing technologies (see subsection 1.2.1. Twenty five years of microbial genome sequencing), the order of the reads produced by these machines cannot be preserved. Therefore, to obtain the true original genomic sequence the process of "genome assembly" has to occur, where FASTQ files, containing the sequencing information, are converted into FASTA files with informative genomic sequences. Information can be inferred through genome annotation software, such as Prokka[31], identifying and labelling all relevant features in a genome sequence, such as predicted coding regions and their putative products, noncoding RNAs, signal peptides, and so on [136].

The term "draft genome" is commonly used because these sequencing technologies do not generate a single closed genome, particularly short-read such as in second generation sequencing (see subsubsection 1.2.1.2. The second-generation of DNA sequencing The second-generation of DNA sequencing) which need to be assembled into usually a series of sequences (contigs) that may cover up to 95% to 99% of the strain genome [118]. Long-read technologies (see subsubsection 1.2.1.3. The third-generation of DNA sequencing) allow for this value to reach 100%, effectively producing closed, complete genomes, notwithstanding that this value can sometimes overcome the 100% due to overlap [137].

---

[30] `https://benlangmead.github.io/aws-indexes/k2`

[31] `https://github.com/tseemann/prokka`

Assembling reads into contigs has many advantages, namely that longer sequences are more informative, allowing the consideration of whole genes or even gene clusters within a genome and understanding larger genetic variants and repeats. Additionally, it has the effect of removing most sequencing errors, though this can be at the expense of new assembly errors [138]. Two methods are used to obtain draft genomes: (1) through reference-guided sequence assembly, or (2), through *de novo* sequence assembly.

1.3.2.1 The FASTA file

In bioinformatics, the FASTA format is a text-based format to represent nucleotide or amino acid sequences using single-letter codes, preceded by a sequence name or any other information relative to the sequence. Similarly to FASTQ (see 1.3.1.1), it was developed by the Wellcome Trust Sanger Institute, the FASTQ has emerged as a common file format for sharing sequence data [123]. The FASTA file follows the following conformation:

- The **first line** of a FASTA file starts with a ">" (greater-than) symbol, signifying the comment portion;
- The **subsequent lines** containing the actual sequence itself represented in the standard IUB/IUPAC amino acid and nucleic acid codes [139], usually 80 characters in length.

A multiple sequence FASTA format can be obtained by concatenating several single sequence FASTA files in a common file (also known as multi-FASTA format). The extension of the file indicates the type of sequence (nucleotide or amino acid) present (see Table 1.2). For genomic data, the ".fasta", ".fa", ".fna", and ".ffn" are the most used, with the first two being generic and the last two specific for nucleic acid and coding regions of a genome.

Table 1.2: The standard filename extension for a text file containing FASTA formatted sequences.

Extension	Sequence	Definition
fasta, fa	generic FASTA	Any generic fasta file. See below for other common FASTA file extensions
fna	FASTA nucleic acid	Used generically to specify nucleic acids.
ffn	FASTA nucleotide of gene regions	Contains coding regions for a genome.
faa	FASTA amino acid	Contains amino acid sequences. A multiple protein fasta file can have the more specific extension mpfa.
frn	FASTA non-coding RNA	Contains non-coding RNA regions for a genome, in DNA alphabet e.g. tRNA, rRNA

1.3.2.2 Genomes through reference-guided sequence assembly

A reference-guided genome assembly uses an already sequenced reference genome to assemble a new genome, making use of the similarity between target and reference species to gain additional information, which often leads to a more complete and improved genome [140, 141]. This process is usually done through the mapping of the reads to a closely related reference sequence, and as more and more species get sequenced, the chances that a genome of the same or related species is already available, in which a significant proportion of the reads can be mapped, increase greatly. This process usually includes the following steps: (1) the reference genome has to be indexed, allowing compression of the input text while still permitting fast sub-string queries, (2) for each short-read several subsequences (seeds) are taken and searched to find their exact matches in the reference (candidate regions), (3) each short-read is then aligned to all corresponding candidate regions, and (4) the consensus sequence is computed in which the reference sequence is corrected when there is enough evidence of a difference based on the mapped reads, identifying the differences between it and the newly generated consensus sequence [142]. In addition to variants, the new consensus genome might have insertions or deletions with respect to the reference genome.

Besides the generation of a consensus sequence, the mapping of the reads to the reference sequence can be used to estimate sequence depth and breadth of coverage. Depth of coverage, often referred to simply as coverage, refers to the average number of times each nucleotide position in the strain's genome has a read that aligns to that position. Depending on the study goals, bacterial species, and the intended analyses, the optimal depth of coverage varies. In public repositories, most submissions have a depth of coverage ranging from 15 to 500 times [118]. The breadth of coverage is defined as the ratio of covered sequence on the reference by aligned reads.

1.3.2.3 Genomes through *de novo* sequence assembly

The *de novo* assembly refers to the bioinformatics process whereby reads are assembled into a draft genome using only the sequence information of the reads. Two methods are used to obtain draft genomes without the need of a reference genome: (1) through Overlap, Layout and Consensus, or (2) De Bruijn graph assembly (see Figure 1.9). The *de novo* assembly methods provide longer sequences that are more informative than shorter sequencing data and can provide a more complete picture of the microbial community in a given sample.

1.3.2.3.1 Overlap, Layout and Consensus assembly

First-generation sequencing technology (see subsubsection 1.2.1.1. The first-generation of DNA sequencing) produces far fewer reads than second-generation sequencing technology (see subsubsection 1.2.1.2. The second-generation of DNA sequencing The second-generation of DNA sequencing), but individual reads are longer (500 to 1000 bp). Assembly of Sanger data usually uses Overlap-Layout Consensus (OLC) approaches, in which:

- Overlaps are computed by comparing all reads to all other reads;
- Overlaps are grouped together to form contigs;

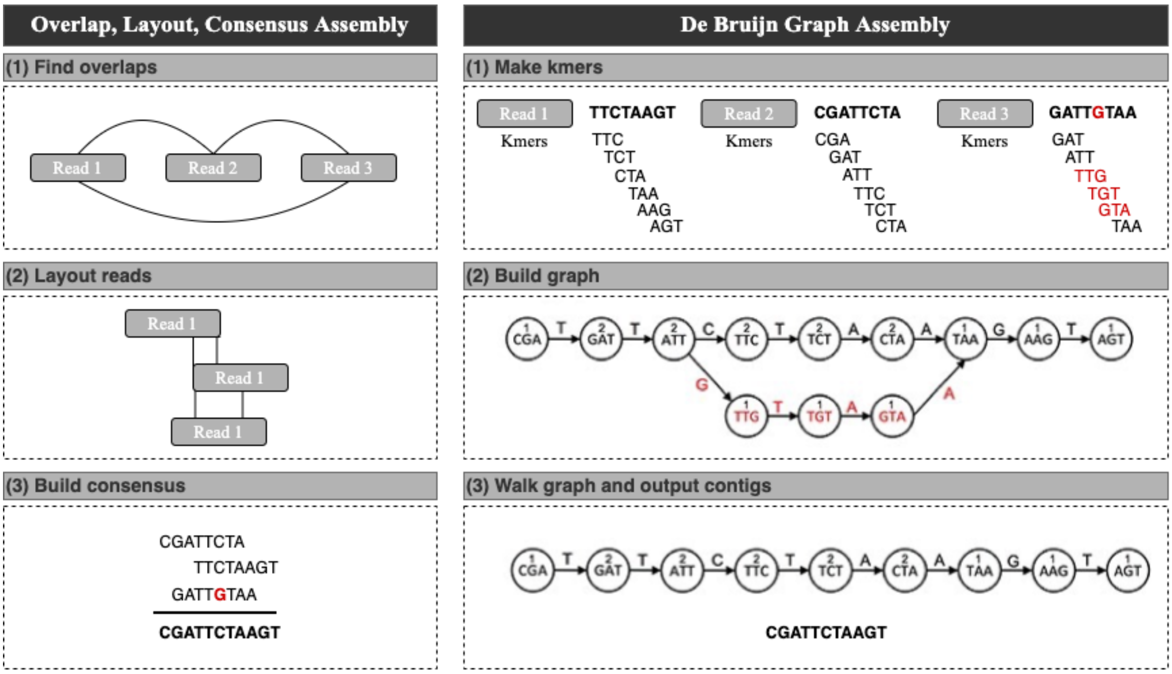


Figure 1.9: **Approaches to *de novo* genome assemble.** In Overlap, Layout, Consensus assembly, (1) overlaps are found between reads and an overlap graph constructed (edges indicate overlapping reads). (2) Reads are laid out into contigs based on the overlaps (lines indicate overlapping portions). (3) The most likely sequence is chosen to construct the consensus sequence. In the De Bruijn graph assembly, (1) reads are decomposed into kmers of a determined size by sliding a window of size k (in here of k=3) across the reads. (2) The kmers become vertices in the De Bruijn graph, with edges connecting overlapping kmers. Polymorphisms (red) form branches in the graph. A count is kept of how many times a kmer is seen, shown here as the numbers above kmers. (3) Contigs are built by walking the graph from the edge nodes. A variety of heuristics handle branches in the graphs—for example, low coverage paths, as shown here, may be ignored. Adapted from [138]

.

- A consensus contiguous sequence, or contig, is determined by picking the most likely nucleotides from the overlapping reads.

These types of assemblers were very popular in the early 2010s, with assemblers such as Celera[32], Genovo[33], xGenovo[34] and BBAP[35] having been widely used [143–146]. With the emergence of third-generation sequencing (see subsubsection 1.2.1.3. The third-generation of DNA sequencing The third-generation of DNA sequencing), OLC assemblers have been increasingly developed and adopted by the community to assembly long-read data. In the latest years, ra[36], raven[37] and canu[38], the latter being a a fork of the Celera Assembler, have become staples in the community, showing good reliability and amassing over 3000 citations [137, 147, 148].

---

[32] https://www.cbcb.umd.edu/software/celera-assembler
[33] https://cs.stanford.edu/genovo
[34] http://xgenovo.dna.bio.keio.ac.jp/
[35] http://homepage.ntu.edu.tw/~youylin/BBAP.html
[36] https://github.com/lbcb-sci/ra
[37] https://github.com/lbcb-sci/raven
[38] https://github.com/marbl/canu

1.3.2.3.2 De Bruijn graph assembly

In the De Bruijn assembly graph, reads are split into overlapping k-mers where nodes of the graph represent k-mers where:

- A directed edge from node N_a to node N_b indicates that N_b is next to N_a in a read;
- The number of nodes in the De Bruijn graph is theoretically the total number of identical k-mers in the genome;
- The weight on the edge indicates the number of times N_b is observed next to N_a in all reads.

Thus, the weight of an edge indicates the possibility that two k-mers appear after each other in the DNA sequence. A path in the graph where all edges have the highest weight is the most likely to be a part of the genome [142].

Most second-generation sequencing (see subsubsection 1.2.1.2. The second-generation of DNA sequencing The second-generation of DNA sequencing) assemblers, such as SPAdes[39] and SKESA[40], use a multiple k-mer De Bruijn graph, starting with the lowest size and iteratively adding k-mers of increasing length to connect the graph [149–151]. Older assemblers, such as Velvet[41], Ray[42] and SoapDeNovo2[43] use a single k-mer strategy for the De Bruijn graph construction [152–154].

1.3.2.4 Assembly quality assessment and quality control

The success of an assembly is evaluated in two steps: (1) globally, through intrinsic characteristics of the assembly itself, and (2) relative to a reference genome. The computation of the global metrics is performed through statistics inherent to the complete set of contigs assembled per sample, independent of the species/sample of origin. Commonly, these statistics include information on contig number, its median size and number ambiguous bases. The comparison with a reference sequence allows statistics such as the number of misassemblies, meaning contigs that do not reflect the structural organisation in the reference sequence, to be computed.

Assessment and evaluation of genome assemblies has been a relevant field ever since the emergence of the assembly process itself. The most widely adopted is QUAST[44], which can

---

[39] `https://github.com/ablab/spades/`

[40] `https://github.com/ncbi/SKESA/`

[41] `https://www.ebi.ac.uk/~zerbino/velvet/`

[42] `https://sourceforge.net/projects/denovoassembler/f`

[43] `https://sourceforge.net/projects/soapdenovo2/`

[44] `http://quast.sourceforge.net/quast`

evaluate assemblies both with a reference genome, as well as without a reference, producing many reports, summary tables and plots to help compare and assess assembly success [155], but alternatives, such as GenomeQC[45] exist [156].

1.3.3 Reproducibility, replicability and transparency

Computational algorithms have become an essential component of microbiome research, with great efforts by the scientific community to raise standards on the development and distribution of code. A lack of reproducibility in computational biology research can be attributed to many factors such as an incomplete or erroneous description of the software used, incomplete documentation on how to run an analysis, or failure to make available the relevant computer code needed [157]. As early as 1990, movements for reproducible research, with a special focus on computation-intensive scientific work, have arisen, brought on by the growing use of computational workflows for analysing data across a range of disciplines [158]

Despite the presented efforts, the effectiveness of computational reproducibility is still questionable. Stodden et al [159] reported that in 22 randomly selected publications whose results relied on the use of computational and data-enabled methods and were deemed to be reproducible (i.e. provided data and/or code), only 14% were straightforward to reproduce with minimal effort. Similar results have been observed in comparable studies [160–162]

Several steps can be implemented to ensure the transparency and reproducibility of the chosen bioinformatic workflow. Despite these efforts, sustainability and reproducibility are still major issues. The field of microbial bioinformatics is not yet widely adopted. The FAIR Principles, standing for Findability, Accessibility, Interoperability, and Reusability put specific emphasis on enhancing the ability to find and reuse not only data but also the algorithms, tools, and workflows that led to that data [163]. The FAIR guiding principles can be summarised as follows:

- **To be Findable:**
    - F1. (meta)data are assigned a globally unique and persistent identifier
    - F2. data are described with rich metadata (defined by R1 below)
    - F3. metadata clearly and explicitly include the identifier of the data it describes
    - F4. (meta)data are registered or indexed in a searchable resource
- **To be Accessible:**
    - A1. (meta)data are retrievable by their identifier using a standardised communications protocol

---

[45] `https://github.com/HuffordLab/GenomeQC`

    - A1.1 the protocol is open, free, and universally implementable
    - A1.2 the protocol allows for an authentication and authorisation procedure, where necessary
  - A2. metadata are accessible, even when the data are no longer available

- **To be Interoperable:**
  - I1. (meta)data use a formal, accessible, shared, and broadly applicable language for knowledge representation.
  - I2. (meta)data use vocabularies that follow FAIR principles
  - I3. (meta)data include qualified references to other (meta)data

- **To be Reusable:**
  - R1. meta(data) are richly described with a plurality of accurate and relevant attributes
  - R1.1. (meta)data are released with a clear and accessible data usage license
  - R1.2. (meta)data are associated with detailed provenance
  - R1.3. (meta)data meet domain-relevant community standards

Several steps have been recommended by experts to ensure the FAIR'ness of both software and data [164–167]. Favouring open-source tools, with clear documentation describing the methodology implemented and stating the version of the software used and which parameters were used, enables the comparison of results. This can be simplified by containerising all software tools with one of the many available solutions, such as Docker[46] or Singularity[47] [168]. The use of workflow managers, like nextflow[48], snakemake[49] or the Galaxy Project[50], will push reproducibility to the next level by taking advantage of the containerisation and scalability, enabling the workflow to be executed with the same parameters in the same conditions in a multitude of different environments [169–171].

When developing software, in the field of microbial bioinformatics, good software engineering practices are not yet widely adopted. An example of such is the widespread use of Version control systems (VCS), which has long been used to maintain code repositories in the software industry, are now finding new applications in science [164]. Git[51], a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency, provides a powerful way to track and compare versions, retrace errors, explore new approaches in a structured manner while maintaining

---

[46] `https://www.docker.com/`

[47] `https://sylabs.io/`

[48] `https://www.nextflow.io/`

[49] `https://snakemake.github.io/`

[50] `https://galaxyproject.org/`

[51] `https://git-scm.com/`

a full audit trail. Remote VCS hosting services, such as GitHub[52], allows for this functionality to be expanded by placing the software in a central location so that it can be accessed by multiple developers and users, facilitating collaboration and audibility. Another example is the use of continued validation through software testing. Modern software engineering advocates reliable software testing standards and best practices. Different approaches are employed: from unit testing to system testing, going from testing every individual component to testing a tool as a whole, verifying and demonstrating that the published code and data are working properly [172].

1.4 Bioinformatic Analysis for Metagenomics

As mentioned previously (see subsection 1.2.3. From genomics to metagenomics, Metagenomic shotgun sequencing circumvents the need for cultivation and, compared with metataxonomics, avoids biases from primer choice, enables the detection of organisms across all domains of life and *de novo* assembly of genomes and functional genome analyses. However, highly uneven sequencing depth of different organisms and low depth of coverage per species are drawbacks that limit taxa

1.4.0.1 Metataxonomics

Metataxonomics (see subsubsection 1.2.3.1. Metataxonomics and Targeted Metagenomics) is the most widely used technique for microbial diversity analysis [173], and due to its particularities, the analysis of this data is also very particular. Data analyses are mostly carried out through specialised pipelines that wrap and combine several tools, offering the possibility to follow a simple protocol with default configurations or choose between a plethora of different configurations to adjust for any particular needs. Quantitative Insights Into Microbial Ecology 2 (QUIIME2)[53] [174] has become the *de facto* tool for metataxonomic analysis as a framework with an ever-growing suite of plugins and intuitive data visualisation tools for the assessment of results. Mothur [175] and UPARSE [176] are also popular alternatives although resulting outputs differ significantly between pipelines despite using the same inputs having been reported by [177], with a magnitude that is comparable to differences in upstream sample treatment and sequencing procedures. A typical workflow starts with quality filtering, error correction and removal of chimeric sequences. These quality control steps are followed by either taxonomic assignment of reads or a clustering step where reads are gathered into OTUs given their sequence identity, followed by statistical analysis to assess differences between given groups. Taxonomic assignment methods classify query sequences based on the best hit found in reference databases of annotated sequences, being

---

[52] `https://github.com/`

[53] `https://qiime2.org/`

heavily dependent on the completeness of the reference databases (see subsubsection 1.2.3.1. Metataxonomics and Targeted Metagenomics). Classification is further limited by the lack of species annotation in most reference databases [178]. Alternatively, the same approach of direct taxonomic classification, without OTUs clustering, can be followed as with genomic and shotgun metagenomic data, given that the databases include rRNA sequences.

OTUs clustering methods can be categorised into: (1) computationally expensive hierarchical methods that cluster sequences based on a distance matrix measuring the difference between each pair of sequences, (2) less expensive heuristic methods cluster sequences into OTUs based on a pre-defined threshold, generally, with a sequence being selected as a seed and the rest of the sequences being analysed sequentially and added to existing or new clusters according to the defined threshold, and (3) model based clustering methods that do not rely on a pre-defined and fixed threshold, defining OTUs based on a soft threshold and carrying out the clustering process based on methods such as an unsupervised probabilistic Bayesian clustering algorithm [179]. These methods offer the possibility to cluster sequences based on criteria that do not depend on reference databases and are especially useful in less characterised microbial communities or with a high representation of uncultured microbes. Due to the assumptions made with this strategy, it is sensitive to under or overestimation of the number of OTUs in a sample as defining a threshold to accurately cluster sequences is difficult [178].

1.4.0.2 Shotgun metagenomics

A plethora of open-source tools are available specifically for shotgun metagenomic data, and several combinations of these tools can be used to characterise the causative agent in a patient's infection in a fraction of the time required by the traditional methods.

A major additional difficulty of shotgun metagenomic data is the overpowering quantities of host DNA that are often sequenced, making the microbial community sometimes close to undetectable [102]. The presence of contaminants, from the bench process to the pre-existing biota, and the cost associated with this methodology, are also major hinders to its applicability in the clinic. They account for major caveats and must be made aware of when analysing the data.

The basic strategies for analysing shotgun metagenomic data can be simplified in the scheme in Figure 1.10. One of the biggest challenges when doing metagenomic analysis is differentiating between colonisation and infection by successfully discriminating between a potential pathogen and background microbiota. In the latter, when analysing samples from presumably sterile sites, like cerebrospinal fluid and blood, it is safe to assume that all organisms found are of interest. In locations with a microbiota, the inclusion of negative controls is essential for the correct identification of contaminants in the taxonomic results, whether originated from the sample collection, handling or sequencing process. The use of spiked

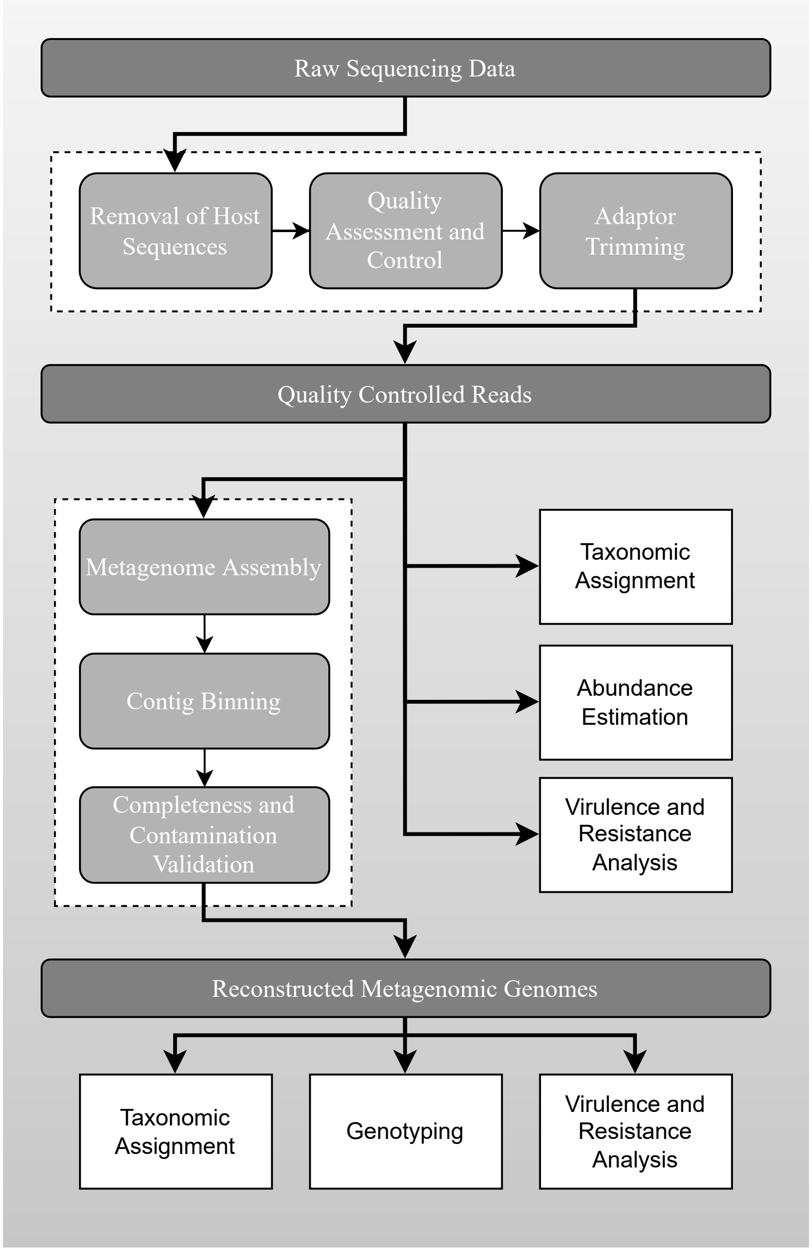


Figure 1.10: Typical bioinformatic analysis procedure for metagenomic data

metagenomic samples as positive control might guide the detection of possible pathogens by comparing relative abundance between the samples. These controls should be processed similarly to the samples and the taxonomic results should be filtered out from the final report.

As explored in subsection 1.3.2. From reads to genomes, longer sequences are more informative than shorter sequencing data, as the one obtained from second-generation sequencing (see subsubsection 1.2.1.2. The second-generation of DNA sequencing The second-generation of DNA sequencing), and can provide a more complete picture of the microbial community in a given sample. Several dedicated metagenomic assembly tools are available, such as metaSPAdes[54] and MegaHIT[55] [151, 180]. These tools, in comparison to single-cell data assemblers, are better at dealing with the combination of intra and intergenomic repeats and uneven sequencing coverage [181]. For third-generation sequencing, dedicated metagenomic assemblers have recently emerged, such as meta-flye[56] which expands on the original flye assembler by overcoming a k-mer selection limitation on low abundance species [182]. Nevertheless, the use of non-dedicated assemblers for metagenomics may come with the cost of wrongly interpreting variation as error, especially in samples that contained closely related species and the construction of chimeric sequences as traditional assemblers follow the basic principle that the coverage in a sample is constant [183].

The assembly-based approach requires the grouping of the different contigs into bins, ideally each collecting the sequences that belong to a microorganism present in the sample. The binning process can be taxonomy dependent, relying on a database to aggregate the sequences, or independent. The independent approach has the benefit of not relying on a database, but instead, it uses the composition of each sequence and coverage profiles to cluster together sequences that might belong to the same organism. These algorithms don't require prior knowledge about the genomes in a given sample, instead relying on features inherent to the sequences in the sample. Although most binning software can work with single metagenomic samples, most make use of differential coverage of multiple samples to improve the binning process [184]. It allows for the handling of complex ecosystems and might be crucial when analysing samples recovered from sites with complex microbiota. A comparison of five taxonomic independent and four taxonomic binning software by [121] revealed that, for taxonomic independent approaches, MaxBin2[57] had the highest completeness and purity in the bins obtained [185]. For taxonomic binning, working similarly to the direct taxonomic assignment of the sequencing data, PhyloPythiaS+[58] obtained better results in accuracy, completeness and purity, followed by Kraken[59] that still obtained decent results with the added benefit of very high speed of analysis, ease of use and inclusion of the pre-built databases [130, 186].

1.5 Aims of the Thesis

Shotgun metagenomic approaches, defined by the sequencing of random DNA fragments of microbial organisms directly from the biological sample, is a promising methodology to obtain very fast results for the identification of pathogens and their virulence and resistance properties directly from samples, without the need for culture. Standardisation of the method and validation of the statistical metrics used to analyse and report the data are of major

---

[54] `https://github.com/ablab/spades/`

[55] `https://github.com/voutcn/megahit/`

[56] `https://github.com/fenderglass/Flye/`

[57] `https://sourceforge.net/projects/maxbin2/`

[58] `https://github.com/algbioi/ppsp`

[59] `https://github.com/DerrickWood/kraken2/`

importance to get this approach to be accredited and used in clinical settings.

The main objective of this work is to evaluate the use of bioinformatics methods for the analysis of metagenomic data to allow the rapid identification, virulence analysis and antimicrobial susceptibility prediction of pathogens with clinical relevance. The main goals are:

- Evaluate the current impact and applicability of metagenomics genomics in medical microbiology, both in clinical and in surveillance and infection prevention settings;
- Develop novel methods and metrics to accurately identify and estimate the relative abundance of pathogens of interest through a hybrid approach of read mapping and *de novo* assembly methods;
- Standardise the process of metagenomic analysis, allowing the comparison of results obtained across domains and stakeholders
- Develop computationally efficient and robust frameworks that allow scientists and/or medical experts with limited programming experience to rapidly and easily query the abundance of specific taxa and genes across the samples of interest, obtaining simple and intuitive reports.

As a proof-of-concept, greater focus was given to clinically relevant taxa, such as the Dengue virus. All methodologies and tools developed were tested and validated on both real and simulated data.

1.6 References

[1] Theo Vos et al. “Global burden of 369 diseases and injuries in 204 countries and territories, 1990–2019: a systematic analysis for the Global Burden of Disease Study 2019”. en. In: *The Lancet* 396.10258 (Oct. 2020), pp. 1204–1222. ISSN: 01406736. DOI: `10.1016/S0140-6736(20)30925-9`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S0140673620309259` (visited on 05/16/2022).

[2] Hannah Ritchie et al. *Coronavirus Pandemic (COVID-19)*. 2020. URL: `https://ourworldindata.org/coronavirus` (visited on 01/28/2022).

[3] Jocelyne Piret and Guy Boivin. “Pandemics Throughout History”. In: *Frontiers in Microbiology* 11 (Jan. 2021), p. 631736. ISSN: 1664-302X. DOI: `10.3389/fmicb.2020.631736`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7874133/` (visited on 01/28/2022).

[4] World Health Organization. *Prioritization of pathogens to guide discovery, research and development of new antibiotics for drug-resistant bacterial infections, including tuberculosis*. Technical documents. World Health Organization, 2017, 87 p.

[5] World Health Organization. *Global expenditure on health: public spending on the rise?* en. Section: xi, 74 p. Geneva: World Health Organization, 2021. ISBN: 978-92-4-004121-9. URL: `https://apps.who.int/iris/handle/10665/350560` (visited on 02/01/2022).

[6] Angela E. Micah et al. “Tracking development assistance for health and for COVID-19: a review of development assistance, government, out-of-pocket, and other private spending on health for 204 countries and territories, 1990–2050”. English. In: *The Lancet* 398.10308 (Oct. 2021). Publisher: Elsevier, pp. 1317–1343. ISSN: 0140-6736, 1474-547X. DOI: `10.1016/S0140-6736(21)01258-7`. URL: `https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(21)01258-7/fulltext` (visited on 02/01/2022).

[7] Xavier Didelot et al. “Transforming clinical microbiology with bacterial genome sequencing”. en. In: *Nature Reviews Genetics* 13.9 (Sept. 2012), pp. 601–612. ISSN: 1471-0056, 1471-0064. DOI: `10.1038/nrg3226`. URL: `http://www.nature.com/articles/nrg3226` (visited on 01/28/2022).

[8] Betsy Foxman et al. “Choosing an appropriate bacterial typing technique for epidemiologic studies”. In: *Epidemiologic perspectives & innovations : EP+I* 2 (Nov. 2005), p. 10. ISSN: 1742-5573. DOI: `10.1186/1742-5573-2-10`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1308839/` (visited on 01/31/2022).

[9] Nurnabila Syafiqah Muhamad Rizal et al. “Advantages and Limitations of 16S rRNA Next-Generation Sequencing for Pathogen Identification in the Diagnostic Microbiology Laboratory: Perspectives from a Middle-Income Country”. en. In: *Diagnostics* 10.10 (Oct. 2020). Number: 10 Publisher: Multidisciplinary Digital Publishing

Institute, p. 816. ISSN: 2075-4418. DOI: `10.3390/diagnostics10100816`. URL: `https://www.mdpi.com/2075-4418/10/10/816` (visited on 02/04/2022).

[10] Christopher Giuliano, Chandni R. Patel, and Pramodini B. Kale-Pradhan. "A Guide to Bacterial Culture Identification And Results Interpretation". In: *Pharmacy and Therapeutics* 44.4 (Apr. 2019), pp. 192–200. ISSN: 1052-1372. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6428495/` (visited on 02/04/2022).

[11] Robin Patel. "MALDI-TOF MS for the Diagnosis of Infectious Diseases". In: *Clinical Chemistry* 61.1 (Jan. 2015), pp. 100–111. ISSN: 0009-9147. DOI: `10.1373/clinchem.2014.221770`. URL: `https://doi.org/10.1373/clinchem.2014.221770` (visited on 02/04/2022).

[12] Michelle H. Scerbo et al. "Beyond Blood Culture and Gram Stain Analysis: A Review of Molecular Techniques for the Early Detection of Bacteremia in Surgical Patients". eng. In: *Surgical Infections* 17.3 (June 2016), pp. 294–302. ISSN: 1557-8674. DOI: `10.1089/sur.2015.099`.

[13] M. Benkova, O. Soukup, and J. Marek. "Antimicrobial susceptibility testing: currently used methods and devices and the near future in clinical practice". en. In: *Journal of Applied Microbiology* 129.4 (2020). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/jam.14704, pp. 806–822. ISSN: 1365-2672. DOI: `10.1111/jam.14704`. URL: `https://onlinelibrary.wiley.com/doi/abs/10.1111/jam.14704` (visited on 02/04/2022).

[14] Franz Allerberger. "Molecular Typing in Public Health Laboratories: From an Academic Indulgence to an Infection Control Imperative". In: *Journal of Preventive Medicine and Public Health* 45.1 (Jan. 2012), pp. 1–7. ISSN: 1975-8375. DOI: `10.3961/jpmph.2012.45.1.1`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3278599/` (visited on 01/31/2022).

[15] Hui-min Neoh et al. "Pulsed-field gel electrophoresis (PFGE): A review of the "gold standard" for bacteria typing and current alternatives". en. In: *Infection, Genetics and Evolution* 74 (2019), p. 103935. ISSN: 1567-1348. DOI: `10.1016/j.meegid.2019.103935`. URL: `https://www.sciencedirect.com/science/article/pii/S156713481930156X` (visited on 01/31/2022).

[16] Martin C.J. Maiden. "Multilocus Sequence Typing of Bacteria". en. In: *Annual Review of Microbiology* 60.1 (Oct. 2006), pp. 561–588. ISSN: 0066-4227, 1545-3251. DOI: `10.1146/annurev.micro.59.030804.121325`. URL: `https://www.annualreviews.org/doi/10.1146/annurev.micro.59.030804.121325` (visited on 01/31/2022).

[17] Mette V. Larsen et al. "Multilocus Sequence Typing of Total-Genome-Sequenced Bacteria". en. In: *Journal of Clinical Microbiology* 50.4 (Apr. 2012), pp. 1355–1361. ISSN: 0095-1137, 1098-660X. DOI: `10.1128/JCM.06094-11`. URL: `https://journals.asm.org/doi/10.1128/JCM.06094-11` (visited on 01/31/2022).

[18] Keith A. Jolley, James E. Bray, and Martin C. J. Maiden. “Open-access bacterial population genomics: BIGSdb software, the PubMLST.org website and their applications”. In: *Wellcome Open Research* 3 (Sept. 2018), p. 124. ISSN: 2398-502X. DOI: `10.12688/wellcomeopenres.14826.1`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6192448/` (visited on 01/31/2022).

[19] Elita Jauneikaite et al. “Current methods for capsular typing of Streptococcus pneumoniae”. en. In: *Journal of Microbiological Methods* 113 (June 2015), pp. 41–49. ISSN: 0167-7012. DOI: `10.1016/j.mimet.2015.03.006`. URL: `https://www.sciencedirect.com/science/article/pii/S0167701215000858` (visited on 01/31/2022).

[20] James C. Paton and Claudia Trappetti. “Streptococcus pneumoniae Capsular Polysaccharide”. EN. In: *Microbiology Spectrum* (Apr. 2019). Publisher: ASM PressWashington, DC. DOI: `10.1128/microbiolspec.GPP3-0019-2018`. URL: `https://journals.asm.org/doi/abs/10.1128/microbiolspec.GPP3-0019-2018` (visited on 01/31/2022).

[21] Benjamin Diep et al. “Salmonella Serotyping; Comparison of the Traditional Method to a Microarray-Based Method and an in silico Platform Using Whole Genome Sequencing Data”. In: *Frontiers in Microbiology* 10 (2019). ISSN: 1664-302X. URL: `https://www.frontiersin.org/article/10.3389/fmicb.2019.02554` (visited on 01/31/2022).

[22] Christopher J. Burrell, Colin R. Howard, and Frederick A. Murphy. “Laboratory Diagnosis of Virus Diseases”. In: *Fenner and White’s Medical Virology* (2017), pp. 135–154. DOI: `10.1016/B978-0-12-375156-0.00010-2`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7149825/` (visited on 01/28/2022).

[23] A. Cassedy, A. Parle-McDermott, and R. O’Kennedy. “Virus Detection: A Review of the Current and Emerging Molecular and Immunological Methods”. In: *Frontiers in Molecular Biosciences* 8 (Apr. 2021), p. 637559. ISSN: 2296-889X. DOI: `10.3389/fmolb.2021.637559`. URL: `https://www.frontiersin.org/articles/10.3389/fmolb.2021.637559/full` (visited on 02/01/2022).

[24] Jennifer Dien Bard and Erin McElvania. “Panels and Syndromic Testing in Clinical Microbiology”. In: *Clinics in Laboratory Medicine* 40.4 (Dec. 2020), pp. 393–420. ISSN: 0272-2712. DOI: `10.1016/j.cll.2020.08.001`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7528880/` (visited on 02/04/2022).

[25] YuYen Chan et al. “Determining seropositivity—A review of approaches to define population seroprevalence when using multiplex bead assays to assess burden of tropical diseases”. en. In: *PLOS Neglected Tropical Diseases* 15.6 (June 2021). Publisher: Public Library of Science, e0009457. ISSN: 1935-2735. DOI: `10.1371/journal.pntd.0009457`. URL: `https://journals.plos.org/plosntds/article?id=10.1371/journal.pntd.0009457` (visited on 02/01/2022).

[26] Niklas Bobrovitz et al. “Global seroprevalence of SARS-CoV-2 antibodies: A systematic review and meta-analysis”. In: *PLoS ONE* 16.6 (June 2021), e0252617. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0252617. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8221784/ (visited on 02/01/2022).

[27] Katarzyna M. Koczula and Andrea Gallotta. “Lateral flow assays”. en. In: *Essays in Biochemistry* 60.1 (June 2016). Ed. by Pedro Estrela, pp. 111–120. ISSN: 0071-1365, 1744-1358. DOI: 10.1042/EBC20150012. URL: https://portlandpress.com/essaysbiochem/article/60/1/111/78237/Lateral-flow-assays (visited on 02/01/2022).

[28] Fabio Di Nardo et al. “Ten Years of Lateral Flow Immunoassay Technique Applications: Trends, Challenges and Future Perspectives”. en. In: *Sensors* 21.15 (Jan. 2021). Number: 15 Publisher: Multidisciplinary Digital Publishing Institute, p. 5185. ISSN: 1424-8220. DOI: 10.3390/s21155185. URL: https://www.mdpi.com/1424-8220/21/15/5185 (visited on 02/01/2022).

[29] Samuel L. Groseclose and David L. Buckeridge. “Public Health Surveillance Systems: Recent Advances in Their Use and Evaluation”. In: *Annual Review of Public Health* 38.1 (2017). _eprint: https://doi.org/10.1146/annurev-publhealth-031816-044348, pp. 57–79. DOI: 10.1146/annurev-publhealth-031816-044348. URL: https://doi.org/10.1146/annurev-publhealth-031816-044348 (visited on 02/07/2022).

[30] Jillian Murray and Adam L. Cohen. “Infectious Disease Surveillance”. In: *International Encyclopedia of Public Health* (2017), pp. 222–229. DOI: 10.1016/B978-0-12-803678-5.00517-8. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7149515/ (visited on 02/07/2022).

[31] Steven M. Teutsch. “Considerations in Planning a Surveillance System”. eng. In: *Principles & Practice of Public Health Surveillance*. 3rd ed. Oxford University Press, 2010. ISBN: 978-0-19-537292-2. DOI: 10.1093/acprof:oso/9780195372922.003.0002. URL: https://oxford.universitypressscholarship.com/10.1093/acprof:oso/9780195372922.001.0001/acprof-9780195372922-chapter-2 (visited on 02/07/2022).

[32] World Health Organization. *International Health Regulations (2005)*. en. Second edition. The Ukrainian version is published by the Center for Implementation of International Health Regulations, Ukraine. Geneva: World Health Organization, 2005. ISBN: 978-92-4-158041-0. URL: https://www.who.int/publications-detail-redirect/9789241580410 (visited on 02/07/2022).

[33] J. Melo-Cristino, Letícia Santos, and Mário Ramirez. “Estudo Viriato: Actualização de dados de susceptibilidade aos antimicrobianos de bactérias responsáveis por infecções respiratórias adquiridas na comunidade em Portugal em 2003 e 2004”. pt. In: *Revista Portuguesa de Pneumologia* 12.1 (Jan. 2006), pp. 9–30. ISSN: 0873-2159.

DOI: `10.1016/S0873-2159(15)30419-0`. URL: `https://www.sciencedirect.com/science/article/pii/S0873215915304190` (visited on 02/07/2022).

[34] Jason R Andrews et al. “Environmental Surveillance as a Tool for Identifying High-risk Settings for Typhoid Transmission”. In: *Clinical Infectious Diseases* 71.Supplement_2 (July 2020), S71–S78. ISSN: 1058-4838. DOI: `10.1093/cid/ciaa513`. URL: `https://doi.org/10.1093/cid/ciaa513` (visited on 02/07/2022).

[35] E. J. McWeeney. “Demonstration of the Typhoid Bacillus in Suspected Drinking Water by Parietti’s Method”. eng. In: *British Medical Journal* 1.1740 (May 1894), pp. 961–962. ISSN: 0007-1447. DOI: `10.1136/bmj.1.1740.961`.

[36] Stephen Baker et al. “Combined high-resolution genotyping and geospatial analysis reveals modes of endemic urban typhoid fever transmission”. eng. In: *Open Biology* 1.2 (Oct. 2011), p. 110008. ISSN: 2046-2441. DOI: `10.1098/rsob.110008`.

[37] David A. Larsen and Krista R. Wigginton. “Tracking COVID-19 with wastewater”. en. In: *Nature Biotechnology* 38.10 (Oct. 2020). Number: 10 Publisher: Nature Publishing Group, pp. 1151–1153. ISSN: 1546-1696. DOI: `10.1038/s41587-020-0690-1`. URL: `https://www.nature.com/articles/s41587-020-0690-1` (visited on 02/07/2022).

[38] Delphine Destoumieux-Garzón et al. “The One Health Concept: 10Years Old and a Long Road Ahead”. In: *Frontiers in Veterinary Science* 5 (Feb. 2018), p. 14. ISSN: 2297-1769. DOI: `10.3389/fvets.2018.00014`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5816263/` (visited on 03/08/2022).

[39] John S Mackenzie and Martyn Jeggo. “The One Health Approach—Why Is It So Important?” In: *Tropical Medicine and Infectious Disease* 4.2 (May 2019), p. 88. ISSN: 2414-6366. DOI: `10.3390/tropicalmed4020088`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6630404/` (visited on 02/07/2022).

[40] Sima Ernest Rugarabamu. *The One-Health Approach to Infectious Disease Outbreaks Control*. en. Publication Title: Current Perspectives on Viral Disease Outbreaks - Epidemiology, Detection and Control. IntechOpen, Sept. 2021. ISBN: 978-1-83881-911-8. DOI: `10.5772/intechopen.95759`. URL: `https://www.intechopen.com/chapters/75084` (visited on 02/07/2022).

[41] D. W. Hood et al. “DNA repeats identify novel virulence genes in Haemophilus influenzae.” en. In: *Proceedings of the National Academy of Sciences* 93.20 (Oct. 1996), pp. 11121–11125. ISSN: 0027-8424, 1091-6490. DOI: `10.1073/pnas.93.20.11121`. URL: `http://www.pnas.org/cgi/doi/10.1073/pnas.93.20.11121` (visited on 01/28/2022).

[42] F. Sanger, S. Nicklen, and A. R. Coulson. “DNA sequencing with chain-terminating inhibitors”. eng. In: *Proceedings of the National Academy of Sciences of the United States of America* 74.12 (Dec. 1977), pp. 5463–5467. ISSN: 0027-8424. DOI: `10.1073/pnas.74.12.5463`.

[43] Richard Van Noorden, Brendan Maher, and Regina Nuzzo. "The top 100 papers". en. In: *Nature News* 514.7524 (Oct. 2014). Cg_type: Nature News Section: News Feature, p. 550. DOI: `10.1038/514550a`. URL: `http://www.nature.com/news/the-top-100-papers-1.16224` (visited on 02/07/2022).

[44] J. D. Watson and F. H. C. Crick. "Molecular Structure of Nucleic Acids: A Structure for Deoxyribose Nucleic Acid". en. In: *Nature* 171.4356 (Apr. 1953). Number: 4356 Publisher: Nature Publishing Group, pp. 737–738. ISSN: 1476-4687. DOI: `10.1038/171737a0`. URL: `https://www.nature.com/articles/171737a0` (visited on 02/08/2022).

[45] Ian S. Hagemann. "Overview of Technical Aspects and Chemistries of Next-Generation Sequencing". en. In: *Clinical Genomics*. Elsevier, 2015, pp. 3–19. ISBN: 978-0-12-404748-8. DOI: `10.1016/B978-0-12-404748-8.00001-0`. URL: `https://linkinghub.elsevier.com/retrieve/pii/B9780124047488000010` (visited on 02/08/2022).

[46] Nicholas J. Loman and Mark J. Pallen. "Twenty years of bacterial genome sequencing". en. In: *Nature Reviews Microbiology* 13.12 (Dec. 2015). Number: 12 Publisher: Nature Publishing Group, pp. 787–794. ISSN: 1740-1534. DOI: `10.1038/nrmicro3565`. URL: `https://www.nature.com/articles/nrmicro3565` (visited on 02/08/2022).

[47] Sara Goodwin, John D. McPherson, and W. Richard McCombie. "Coming of age: ten years of next-generation sequencing technologies". en. In: *Nature Reviews Genetics* 17.6 (June 2016). Number: 6 Publisher: Nature Publishing Group, pp. 333–351. ISSN: 1471-0064. DOI: `10.1038/nrg.2016.49`. URL: `https://www.nature.com/articles/nrg.2016.49` (visited on 02/08/2022).

[48] Yunhao Wang et al. "Nanopore sequencing technology, bioinformatics and applications". en. In: *Nature Biotechnology* 39.11 (Nov. 2021). Number: 11 Publisher: Nature Publishing Group, pp. 1348–1365. ISSN: 1546-1696. DOI: `10.1038/s41587-021-01108-x`. URL: `https://www.nature.com/articles/s41587-021-01108-x` (visited on 03/01/2022).

[49] Michael L. Metzker. "Sequencing technologies — the next generation". en. In: *Nature Reviews Genetics* 11.1 (Jan. 2010), pp. 31–46. ISSN: 1471-0056, 1471-0064. DOI: `10.1038/nrg2626`. URL: `http://www.nature.com/articles/nrg2626` (visited on 03/01/2022).

[50] Liu Xu and Masahide Seki. "Recent advances in the detection of base modifications using the Nanopore sequencer". en. In: *Journal of Human Genetics* 65.1 (Jan. 2020). Number: 1 Publisher: Nature Publishing Group, pp. 25–33. ISSN: 1435-232X. DOI: `10.1038/s10038-019-0679-0`. URL: `https://www.nature.com/articles/s10038-019-0679-0` (visited on 03/01/2022).

[51] Linda Koch, Catherine Potenski, and Michelle Trenkmann. “Sequencing moves to the twenty-first century”. en. In: *Nature Research* (Feb. 2021). Bandiera_abtest: a Cg_type: Milestones Publisher: Nature Publishing Group. DOI: `10.1038/d42859-020-00100-w`. URL: `https://www.nature.com/articles/d42859-020-00100-w` (visited on 02/08/2022).

[52] S. T. Cole et al. “Deciphering the biology of Mycobacterium tuberculosis from the complete genome sequence”. en. In: *Nature* 393.6685 (June 1998). Number: 6685 Publisher: Nature Publishing Group, pp. 537–544. ISSN: 1476-4687. DOI: `10.1038/31159`. URL: `https://www.nature.com/articles/31159` (visited on 02/07/2022).

[53] J. Parkhill et al. “Genome sequence of Yersinia pestis, the causative agent of plague”. en. In: *Nature* 413.6855 (Oct. 2001). Number: 6855 Publisher: Nature Publishing Group, pp. 523–527. ISSN: 1476-4687. DOI: `10.1038/35097083`. URL: `https://www.nature.com/articles/35097083` (visited on 02/08/2022).

[54] Frederick R. Blattner et al. “The Complete Genome Sequence of *Escherichia coli* K-12”. en. In: *Science* 277.5331 (Sept. 1997), pp. 1453–1462. ISSN: 0036-8075, 1095-9203. DOI: `10.1126/science.277.5331.1453`. URL: `https://www.science.org/doi/10.1126/science.277.5331.1453` (visited on 02/08/2022).

[55] Barton E. Slatko, Andrew F. Gardner, and Frederick M. Ausubel. “Overview of Next Generation Sequencing Technologies”. In: *Current protocols in molecular biology* 122.1 (Apr. 2018), e59. ISSN: 1934-3639. DOI: `10.1002/cpmb.59`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6020069/` (visited on 02/08/2022).

[56] J.C. Detter et al. “Nucleic acid sequencing for characterizing infectious and/or novel agents in complex samples”. en. In: *Biological Identification*. Elsevier, 2014, pp. 3–53. ISBN: 978-0-85709-501-5. DOI: `10.1533/9780857099167.1.3`. URL: `https://linkinghub.elsevier.com/retrieve/pii/B9780857095015500015` (visited on 02/08/2022).

[57] Alice Maria Giani et al. “Long walk to genomics: History and current approaches to genome sequencing and assembly”. en. In: *Computational and Structural Biotechnology Journal* 18 (Jan. 2020), pp. 9–19. ISSN: 2001-0370. DOI: `10.1016/j.csbj.2019.11.002`. URL: `https://www.sciencedirect.com/science/article/pii/S2001037019303277` (visited on 02/08/2022).

[58] Nicholas J Loman et al. “Performance comparison of benchtop high-throughput sequencing platforms”. en. In: *Nature Biotechnology* 30.5 (May 2012), pp. 434–439. ISSN: 1087-0156, 1546-1696. DOI: `10.1038/nbt.2198`. URL: `http://www.nature.com/articles/nbt.2198` (visited on 02/14/2022).

[59] Anuj Kumar Gupta and U. D. Gupta. “Chapter 19 - Next Generation Sequencing and Its Applications”. en. In: *Animal Biotechnology*. Ed. by Ashish S. Verma and Anchal Singh. San Diego: Academic Press, Jan. 2014, pp. 345–367. ISBN: 978-0-12-416002-6. DOI: `10.1016/B978-0-12-416002-6.00019-5`. URL: `https:`

//www.sciencedirect.com/science/article/pii/B9780124160026000195 (visited on 02/14/2022).

[60] Minh Thuy Vi Hoang et al. "Long-Reads-Based Metagenomics in Clinical Diagnosis With a Special Focus on Fungal Infections". In: *Frontiers in Microbiology* 12 (2022). ISSN: 1664-302X. URL: https://www.frontiersin.org/article/10.3389/fmicb.2021.708550 (visited on 02/14/2022).

[61] Jonas Korlach and Stephen W Turner. "Going beyond five bases in DNA sequencing". en. In: *Current Opinion in Structural Biology*. Nucleic acids/Sequences and topology 22.3 (June 2012), pp. 251–261. ISSN: 0959-440X. DOI: 10.1016/j.sbi.2012.04.002. URL: https://www.sciencedirect.com/science/article/pii/S0959440X12000681 (visited on 02/14/2022).

[62] Aaron M. Wenger et al. "Accurate circular consensus long-read sequencing improves variant detection and assembly of a human genome". en. In: *Nature Biotechnology* 37.10 (Oct. 2019), pp. 1155–1162. ISSN: 1087-0156, 1546-1696. DOI: 10.1038/s41587-019-0217-9. URL: http://www.nature.com/articles/s41587-019-0217-9 (visited on 02/14/2022).

[63] Elizabeth T. Cirulli and David B. Goldstein. "Uncovering the roles of rare variants in common disease through whole-genome sequencing". en. In: *Nature Reviews Genetics* 11.6 (June 2010). Number: 6 Publisher: Nature Publishing Group, pp. 415–425. ISSN: 1471-0064. DOI: 10.1038/nrg2779. URL: https://www.nature.com/articles/nrg2779 (visited on 02/18/2022).

[64] Nature Reviews Genetics. "A genomic approach to microbiology". en. In: *Nature Reviews Genetics* 20.6 (June 2019), pp. 311–311. ISSN: 1471-0056, 1471-0064. DOI: 10.1038/s41576-019-0131-5. URL: http://www.nature.com/articles/s41576-019-0131-5 (visited on 01/26/2022).

[65] F. Tagini and G. Greub. "Bacterial genome sequencing in clinical microbiology: a pathogen-oriented review". In: *European Journal of Clinical Microbiology & Infectious Diseases* 36.11 (2017), pp. 2007–2020. ISSN: 0934-9723. DOI: 10.1007/s10096-017-3024-6. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5653721/ (visited on 02/08/2022).

[66] J. W. A. Rossen et al. "Practical issues in implementing whole-genome-sequencing in routine diagnostic microbiology". eng. In: *Clinical Microbiology and Infection: The Official Publication of the European Society of Clinical Microbiology and Infectious Diseases* 24.4 (Apr. 2018), pp. 355–360. ISSN: 1469-0691. DOI: 10.1016/j.cmi.2017.11.001.

[67] Stephanie W. Lo and Dorota Jamrozy. "Genomics and epidemiological surveillance". en. In: *Nature Reviews Microbiology* 18.9 (Sept. 2020). Number: 9 Publisher: Nature Publishing Group, pp. 478–478. ISSN: 1740-1534. DOI: 10.1038/s41579-020-0421-0. URL: https://www.nature.com/articles/s41579-020-0421-0 (visited on 02/18/2022).

[68] Ayorinde O. Afolayan et al. "Overcoming Data Bottlenecks in Genomic Pathogen Surveillance". eng. In: *Clinical Infectious Diseases: An Official Publication of the Infectious Diseases Society of America* 73.Supplement_4 (Dec. 2021), S267–S274. ISSN: 1537-6591. DOI: `10.1093/cid/ciab785`.

[69] Rafael Mamede et al. "Chewie Nomenclature Server (chewie-NS): a deployable nomenclature server for easy sharing of core and whole genome MLST schemas". In: *Nucleic Acids Research* 49.D1 (Jan. 2021), pp. D660–D666. ISSN: 0305-1048. DOI: `10.1093/nar/gkaa889`. URL: `https://doi.org/10.1093/nar/gkaa889` (visited on 02/18/2022).

[70] Zhemin Zhou et al. "The EnteroBase user's guide, with case studies on Salmonella transmissions, Yersinia pestis phylogeny, and Escherichia core genomic diversity". en. In: *Genome Research* 30.1 (Jan. 2020). Company: Cold Spring Harbor Laboratory Press Distributor: Cold Spring Harbor Laboratory Press Institution: Cold Spring Harbor Laboratory Press Label: Cold Spring Harbor Laboratory Press Publisher: Cold Spring Harbor Lab, pp. 138–152. ISSN: 1088-9051, 1549-5469. DOI: `10.1101/gr.251678.119`. URL: `https://genome.cshlp.org/content/30/1/138` (visited on 02/18/2022).

[71] Silvia Argimón et al. "Microreact: visualizing and sharing data for genomic epidemiology and phylogeography". In: *Microbial Genomics* 2.11 (). Publisher: Microbiology Society, e000093. ISSN: 2057-5858, DOI: `10.1099/mgen.0.000093`. URL: `https://www.microbiologyresearch.org/content/journal/mgen/10.1099/mgen.0.000093` (visited on 02/18/2022).

[72] Yuelong Shu and John McCauley. "GISAID: Global initiative on sharing all influenza data - from vision to reality". eng. In: *Euro Surveillance: Bulletin Europeen Sur Les Maladies Transmissibles = European Communicable Disease Bulletin* 22.13 (Mar. 2017), p. 30494. ISSN: 1560-7917. DOI: `10.2807/1560-7917.ES.2017.22.13.30494`.

[73] Brett E. Pickett et al. "Virus Pathogen Database and Analysis Resource (ViPR): A Comprehensive Bioinformatics Database and Analysis Resource for the Coronavirus Research Community". en. In: *Viruses* 4.11 (Nov. 2012). Number: 11 Publisher: Molecular Diversity Preservation International, pp. 3209–3226. ISSN: 1999-4915. DOI: `10.3390/v4113209`. URL: `https://www.mdpi.com/1999-4915/4/11/3209` (visited on 02/18/2022).

[74] Vítor Borges et al. "INSaFLU: an automated open web-based bioinformatics suite "from-reads" for influenza whole-genome-sequencing-based surveillance". In: *Genome Medicine* 10.1 (June 2018), p. 46. ISSN: 1756-994X. DOI: `10.1186/s13073-018-0555-0`. URL: `https://doi.org/10.1186/s13073-018-0555-0` (visited on 02/18/2022).

[75] James Hadfield et al. "Nextstrain: real-time tracking of pathogen evolution". In: *Bioinformatics* 34.23 (2018), pp. 4121–4123. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/bty407`. URL: `https://doi.org/10.1093/bioinformatics/bty407` (visited on 02/18/2022).

[76] Angela H. Beckett, Kate F. Cook, and Samuel C. Robson. "A pandemic in the age of next-generation sequencing". In: *The Biochemist* 43.6 (2021), pp. 10–15. ISSN: 0954-982X. DOI: `10.1042/bio_2021_187`. URL: `https://doi.org/10.1042/bio_2021_187` (visited on 02/23/2022).

[77] The Lancet. "Genomic sequencing in pandemics". English. In: *The Lancet* 397.10273 (Feb. 2021). Publisher: Elsevier, p. 445. ISSN: 0140-6736, 1474-547X. DOI: `10.1016/S0140-6736(21)00257-9`. URL: `https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(21)00257-9/fulltext` (visited on 02/23/2022).

[78] Gavin J. D. Smith et al. "Origins and evolutionary genomics of the 2009 swine-origin H1N1 influenza A epidemic". en. In: *Nature* 459.7250 (June 2009). Number: 7250 Publisher: Nature Publishing Group, pp. 1122–1125. ISSN: 1476-4687. DOI: `10.1038/nature08182`. URL: `https://www.nature.com/articles/nature08182` (visited on 02/23/2022).

[79] Novel Swine-Origin Influenza A (H1N1) Virus Investigation Team. "Emergence of a Novel Swine-Origin Influenza A (H1N1) Virus in Humans". In: *New England Journal of Medicine* 360.25 (June 2009). Publisher: Massachusetts Medical Society _eprint: https://doi.org/10.1056/NEJMoa0903810, pp. 2605–2615. ISSN: 0028-4793. DOI: `10.1056/NEJMoa0903810`. URL: `https://doi.org/10.1056/NEJMoa0903810` (visited on 02/23/2022).

[80] Roujian Lu et al. "Complete Genome Sequence of Middle East Respiratory Syndrome Coronavirus (MERS-CoV) from the First Imported MERS-CoV Case in China". In: *Genome Announcements* 3.4 (Aug. 2015), e00818–15. ISSN: 2169-8287. DOI: `10.1128/genomeA.00818-15`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4536671/` (visited on 02/23/2022).

[81] Ahmed Kandeil et al. "Complete Genome Sequence of Middle East Respiratory Syndrome Coronavirus Isolated from a Dromedary Camel in Egypt". In: *Genome Announcements* 4.2 (Apr. 2016), e00309–16. ISSN: 2169-8287. DOI: `10.1128/genomeA.00309-16`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4850855/` (visited on 02/23/2022).

[82] Badr M. Al-Shomrani et al. "Genomic Sequencing and Analysis of Eight Camel-Derived Middle East Respiratory Syndrome Coronavirus (MERS-CoV) Isolates in Saudi Arabia". In: *Viruses* 12.6 (June 2020), p. 611. ISSN: 1999-4915. DOI: `10.3390/v12060611`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7354450/` (visited on 02/23/2022).

[83] Fan Wu et al. “A new coronavirus associated with human respiratory disease in China”. en. In: *Nature* 579.7798 (Mar. 2020), pp. 265–269. ISSN: 0028-0836, 1476-4687. DOI: `10.1038/s41586-020-2008-3`. URL: `http://www.nature.com/articles/s41586-020-2008-3` (visited on 02/23/2022).

[84] Amy Maxmen. “One million coronavirus sequences: popular genome site hits mega milestone”. en. In: *Nature* 593.7857 (Apr. 2021). Bandiera_abtest: a Cg_type: News Number: 7857 Publisher: Nature Publishing Group Subject_term: SARS-CoV-2, Databases, Epidemiology, pp. 21–21. DOI: `10.1038/d41586-021-01069-w`. URL: `https://www.nature.com/articles/d41586-021-01069-w` (visited on 02/23/2022).

[85] Vítor Borges et al. “SARS-CoV-2 introductions and early dynamics of the epidemic in Portugal”. en. In: *Communications Medicine* 2.1 (Jan. 2022). Number: 1 Publisher: Nature Publishing Group, pp. 1–11. ISSN: 2730-664X. DOI: `10.1038/s43856-022-00072-0`. URL: `https://www.nature.com/articles/s43856-022-00072-0` (visited on 02/23/2022).

[86] Leonard Schuele et al. “Future potential of metagenomics in microbiology laboratories”. en. In: *Expert Review of Molecular Diagnostics* 21.12 (Dec. 2021), pp. 1273–1285. ISSN: 1473-7159, 1744-8352. DOI: `10.1080/14737159.2021.2001329`. URL: `https://www.tandfonline.com/doi/full/10.1080/14737159.2021.2001329` (visited on 05/20/2022).

[87] Nicholas J. Loman et al. “High-throughput bacterial genome sequencing: an embarrassment of choice, a world of opportunity”. en. In: *Nature Reviews Microbiology* 10.9 (Sept. 2012). Number: 9 Publisher: Nature Publishing Group, pp. 599–606. ISSN: 1740-1534. DOI: `10.1038/nrmicro2850`. URL: `https://www.nature.com/articles/nrmicro2850` (visited on 02/08/2022).

[88] J. W. A. Rossen, A. W. Friedrich, and J. Moran-Gilad. “& ESCMID Study Group for Genomic and Molecular Diagnostics (ESGMD). Practical issues in implementing whole-genome-sequencing in routine diagnostic microbiology”. In: *Clin. Microbiol. Infect.* 24 (2018). DOI: `10.1016/j.cmi.2017.11.001`. URL: `https://doi.org/10.1016/j.cmi.2017.11.001`.

[89] W. M. Dunne, L. F. Westblade, and B. Ford. “Next-generation and whole-genome sequencing in the diagnostic clinical microbiology laboratory”. en. In: *European Journal of Clinical Microbiology & Infectious Diseases* 31.8 (Aug. 2012), pp. 1719–1726. ISSN: 1435-4373. DOI: `10.1007/s10096-012-1641-7`. URL: `https://doi.org/10.1007/s10096-012-1641-7` (visited on 02/24/2022).

[90] Charles Y. Chiu and Steven A. Miller. “Clinical metagenomics”. en. In: *Nature Reviews Genetics* 20.6 (June 2019). Number: 6 Publisher: Nature Publishing Group, pp. 341–355. ISSN: 1471-0064. DOI: `10.1038/s41576-019-0113-7`. URL: `https://www.nature.com/articles/s41576-019-0113-7` (visited on 02/08/2022).

[91] Julian R. Marchesi and Jacques Ravel. “The vocabulary of microbiome research: a proposal”. In: *Microbiome* 3.1 (July 2015), p. 31. ISSN: 2049-2618. DOI: `10.1186/s40168-015-0094-5`. URL: `https://doi.org/10.1186/s40168-015-0094-5` (visited on 02/24/2022).

[92] Ramya Srinivasan et al. “Use of 16S rRNA Gene for Identification of a Broad Range of Clinically Relevant Bacterial Pathogens”. en. In: *PLOS ONE* 10.2 (June 2015). Publisher: Public Library of Science, e0117617. ISSN: 1932-6203. DOI: `10.1371/journal.pone.0117617`. URL: `https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0117617` (visited on 02/24/2022).

[93] Isabel Abellan-Schneyder et al. “Primer, Pipelines, Parameters: Issues in 16S rRNA Gene Sequencing”. EN. In: *mSphere* (Feb. 2021). Publisher: American Society for Microbiology 1752 N St., N.W., Washington, DC. DOI: `10.1128/mSphere.01202-20`. URL: `https://journals.asm.org/doi/abs/10.1128/mSphere.01202-20` (visited on 02/24/2022).

[94] J. R. Cole et al. “The Ribosomal Database Project: improved alignments and new tools for rRNA analysis”. In: *Nucleic Acids Research* 37.suppl_1 (Jan. 2009), pp. D141–D145. ISSN: 0305-1048. DOI: `10.1093/nar/gkn879`. URL: `https://doi.org/10.1093/nar/gkn879` (visited on 02/24/2022).

[95] T. Z. DeSantis et al. “Greengenes, a Chimera-Checked 16S rRNA Gene Database and Workbench Compatible with ARB”. In: *Applied and Environmental Microbiology* 72.7 (July 2006), pp. 5069–5072. ISSN: 0099-2240. DOI: `10.1128/AEM.03006-05`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1489311/` (visited on 02/24/2022).

[96] Elmar Pruesse et al. “SILVA: a comprehensive online resource for quality checked and aligned ribosomal RNA sequence data compatible with ARB”. In: *Nucleic Acids Research* 35.21 (Dec. 2007), pp. 7188–7196. ISSN: 0305-1048. DOI: `10.1093/nar/gkm864`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2175337/` (visited on 02/24/2022).

[97] Patrick D. Schloss and Jo Handelsman. “Introducing DOTUR, a Computer Program for Defining Operational Taxonomic Units and Estimating Species Richness”. In: *Applied and Environmental Microbiology* 71.3 (Mar. 2005). Publisher: American Society for Microbiology, pp. 1501–1506. DOI: `10.1128/AEM.71.3.1501-1506.2005`. URL: `https://journals.asm.org/doi/10.1128/AEM.71.3.1501-1506.2005` (visited on 02/24/2022).

[98] Jethro S. Johnson et al. “Evaluation of 16S rRNA gene sequencing for species and strain-level microbiome analysis”. en. In: *Nature Communications* 10.1 (Nov. 2019). Number: 1 Publisher: Nature Publishing Group, p. 5029. ISSN: 2041-1723. DOI: `10.1038/s41467-019-13036-1`. URL: `https://www.nature.com/articles/s41467-019-13036-1` (visited on 02/24/2022).

[99] Joshua Quick et al. “Real-time, portable genome sequencing for Ebola surveillance”. In: *Nature* 530.7589 (Feb. 2016), pp. 228–232. ISSN: 0028-0836. DOI: `10.1038/nature16996`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4817224/` (visited on 02/28/2022).

[100] Leonard Schuele et al. “Assessment of Viral Targeted Sequence Capture Using Nanopore Sequencing Directly from Clinical Samples”. en. In: *Viruses* 12.12 (Dec. 2020). Number: 12 Publisher: Multidisciplinary Digital Publishing Institute, p. 1358. ISSN: 1999-4915. DOI: `10.3390/v12121358`. URL: `https://www.mdpi.com/1999-4915/12/12/1358` (visited on 02/24/2022).

[101] Todd N. Wylie et al. “Enhanced virome sequencing using targeted sequence capture”. In: *Genome Research* 25.12 (Dec. 2015), pp. 1910–1920. ISSN: 1088-9051. DOI: `10.1101/gr.191049.115`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4665012/` (visited on 02/24/2022).

[102] Natacha Couto et al. “Critical steps in clinical shotgun metagenomics for the concomitant detection and typing of microbial pathogens”. en. In: *Scientific Reports* 8.1 (Dec. 2018), p. 13767. ISSN: 2045-2322. DOI: `10.1038/s41598-018-31873-w`. URL: `http://www.nature.com/articles/s41598-018-31873-w` (visited on 05/11/2022).

[103] Melissa B. Miller and Yi-Wei Tang. “Basic Concepts of Microarrays and Potential Applications in Clinical Microbiology”. In: *Clinical Microbiology Reviews* 22.4 (Oct. 2009), pp. 611–633. ISSN: 0893-8512. DOI: `10.1128/CMR.00019-09`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2772365/` (visited on 02/28/2022).

[104] Chana Palmer et al. “Rapid quantitative profiling of complex microbial populations”. In: *Nucleic Acids Research* 34.1 (2006), e5. ISSN: 0305-1048. DOI: `10.1093/nar/gnj007`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1326253/` (visited on 02/28/2022).

[105] Michael R. Wilson et al. “Actionable Diagnosis of Neuroleptospirosis by Next-Generation Sequencing”. In: *New England Journal of Medicine* 370.25 (June 2014). Publisher: Massachusetts Medical Society _eprint: https://doi.org/10.1056/NEJMoa1401268, pp. 2408–2417. ISSN: 0028-4793. DOI: `10.1056/NEJMoa1401268`. URL: `https://doi.org/10.1056/NEJMoa1401268` (visited on 02/28/2022).

[106] Prakhar Vijayvargiya et al. “Application of metagenomic shotgun sequencing to detect vector-borne pathogens in clinical blood samples”. en. In: *PLOS ONE* 14.10 (Feb. 2019). Publisher: Public Library of Science, e0222915. ISSN: 1932-6203. DOI: `10.1371/journal.pone.0222915`. URL: `https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0222915` (visited on 02/28/2022).

[107] Adriana Sanabria et al. "Shotgun-Metagenomics on Positive Blood Culture Bottles Inoculated With Prosthetic Joint Tissue: A Proof of Concept Study". In: *Frontiers in Microbiology* 11 (2020). ISSN: 1664-302X. URL: `https://www.frontiersin.org/article/10.3389/fmicb.2020.01687` (visited on 02/28/2022).

[108] Shota Hirakata et al. "The application of shotgun metagenomics to the diagnosis of granulomatous amoebic encephalitis due to Balamuthia mandrillaris: a case report". In: *BMC Neurology* 21.1 (2021), p. 392. ISSN: 1471-2377. DOI: `10.1186/s12883-021-02418-y`. URL: `https://doi.org/10.1186/s12883-021-02418-y` (visited on 02/28/2022).

[109] Wenfang He et al. "Case Report: Therapeutic Strategy With Delayed Debridement for Culture-Negative Invasive Group A Streptococcal Infections Diagnosed by Metagenomic Next-Generation Sequencing". In: *Frontiers in Public Health* 10 (2022). ISSN: 2296-2565. URL: `https://www.frontiersin.org/articles/10.3389/fpubh.2022.899077` (visited on 10/20/2022).

[110] Nicholas J. Loman et al. "A Culture-Independent Sequence-Based Metagenomics Approach to the Investigation of an Outbreak of Shiga-Toxigenic Escherichia coli O104:H4". In: *JAMA* 309.14 (2013), pp. 1502–1510. ISSN: 0098-7484. DOI: `10.1001/jama.2013.3231`. URL: `https://doi.org/10.1001/jama.2013.3231` (visited on 02/28/2022).

[111] Andrew D. Huang et al. "Metagenomics of Two Severe Foodborne Outbreaks Provides Diagnostic Signatures and Signs of Coinfection Not Attainable by Traditional Methods". In: *Applied and Environmental Microbiology* 83.3 (Jan. 2017), e02577–16. ISSN: 0099-2240. DOI: `10.1128/AEM.02577-16`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5244306/` (visited on 02/28/2022).

[112] Sijia Li et al. "Microbiome Profiling Using Shotgun Metagenomic Sequencing Identified Unique Microorganisms in COVID-19 Patients With Altered Gut Microbiota". In: *Frontiers in Microbiology* 12 (2021). ISSN: 1664-302X. URL: `https://www.frontiersin.org/article/10.3389/fmicb.2021.712081` (visited on 02/28/2022).

[113] Heather A. Carleton et al. "Metagenomic Approaches for Public Health Surveillance of Foodborne Infections: Opportunities and Challenges". In: *Foodborne Pathogens and Disease* 16.7 (July 2019). Publisher: Mary Ann Liebert, Inc., publishers, pp. 474–479. ISSN: 1535-3141. DOI: `10.1089/fpd.2019.2636`. URL: `https://www.liebertpub.com/doi/10.1089/fpd.2019.2636` (visited on 02/28/2022).

[114] Susannah J. Salter et al. "Reagent and laboratory contamination can critically impact sequence-based microbiome analyses". In: *BMC Biology* 12.1 (Nov. 2014), p. 87. ISSN: 1741-7007. DOI: `10.1186/s12915-014-0087-z`. URL: `https://doi.org/10.1186/s12915-014-0087-z` (visited on 02/28/2022).

[115] Dominic O'Neil, Heike Glowatz, and Martin Schlumpberger. "Ribosomal RNA Depletion for Efficient Use of RNA-Seq Capacity". en. In: *Current Protocols in Molecular Biology* 103.1 (2013). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/0471142727.mb0419s103, pp. 4.19.1–4.19.8. ISSN: 1934-3647. DOI: `10.1002/0471142727.mb0419s103`. URL: `https://onlinelibrary.wiley.com/doi/abs/10.1002/0471142727.mb0419s103` (visited on 02/28/2022).

[116] George R. Feehery et al. "A Method for Selectively Enriching Microbial DNA from Contaminating Vertebrate Host DNA". en. In: *PLoS ONE* 8.10 (2013). Publisher: Public Library of Science. DOI: `10.1371/journal.pone.0076096`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3810253/` (visited on 02/28/2022).

[117] Alexa B. R. McIntyre et al. "Comprehensive benchmarking and ensemble approaches for metagenomic classifiers". In: *Genome Biology* 18.1 (2017), p. 182. ISSN: 1474-760X. DOI: `10.1186/s13059-017-1299-7`. URL: `https://doi.org/10.1186/s13059-017-1299-7` (visited on 02/28/2022).

[118] J. A. Carriço et al. "A primer on microbial bioinformatics for nonbioinformaticians". en. In: *Clinical Microbiology and Infection* 24.4 (2018), pp. 342–349. ISSN: 1198-743X. DOI: `10.1016/j.cmi.2017.12.015`. URL: `https://www.sciencedirect.com/science/article/pii/S1198743X17307097` (visited on 02/18/2022).

[119] Alexandre Angers-Loustau et al. "The challenges of designing a benchmark strategy for bioinformatics pipelines in the identification of antimicrobial resistance determinants using next generation sequencing technologies". en. In: *F1000Research* 7 (Dec. 2018), p. 459. ISSN: 2046-1402. DOI: `10.12688/f1000research.14509.2`. URL: `https://f1000research.com/articles/7-459/v2` (visited on 03/25/2021).

[120] Bjorn Gruening et al. "Recommendations for the packaging and containerizing of bioinformatics software". en. In: *F1000Research* 7 (Mar. 2019), p. 742. ISSN: 2046-1402. DOI: `10.12688/f1000research.15140.2`. URL: `https://f1000research.com/articles/7-742/v2` (visited on 04/30/2022).

[121] Alexander Sczyrba et al. "Critical Assessment of Metagenome Interpretation—a benchmark of metagenomics software". en. In: *Nature Methods* 14.11 (Nov. 2017). Number: 11 Publisher: Nature Publishing Group, pp. 1063–1071. ISSN: 1548-7105. DOI: `10.1038/nmeth.4458`. URL: `https://www.nature.com/articles/nmeth.4458` (visited on 03/20/2022).

[122] Peter J. A. Cock et al. "The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants". In: *Nucleic Acids Research* 38.6 (Apr. 2010), pp. 1767–1771. ISSN: 0305-1048. DOI: `10.1093/nar/gkp1137`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2847217/` (visited on 03/02/2022).

[123] W R Pearson and D J Lipman. "Improved tools for biological sequence comparison." In: *Proceedings of the National Academy of Sciences of the United States of America* 85.8 (Apr. 1988), pp. 2444–2448. ISSN: 0027-8424. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC280013/` (visited on 03/02/2022).

[124] Brent Ewing and Phil Green. "Base-Calling of Automated Sequencer Traces Using Phred. II. Error Probabilities". en. In: *Genome Research* 8.3 (Mar. 1998). Company: Cold Spring Harbor Laboratory Press Distributor: Cold Spring Harbor Laboratory Press Institution: Cold Spring Harbor Laboratory Press Label: Cold Spring Harbor Laboratory Press Publisher: Cold Spring Harbor Lab, pp. 186–194. ISSN: 1088-9051, 1549-5469. DOI: `10.1101/gr.8.3.186`. URL: `https://genome.cshlp.org/content/8/3/186` (visited on 03/02/2022).

[125] Merly Escalona, Sara Rocha, and David Posada. "A comparison of tools for the simulation of genomic next-generation sequencing data". In: *Nature reviews. Genetics* 17.8 (Aug. 2016), pp. 459–469. ISSN: 1471-0056. DOI: `10.1038/nrg.2016.57`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5224698/` (visited on 03/03/2022).

[126] Anthony M. Bolger, Marc Lohse, and Bjoern Usadel. "Trimmomatic: a flexible trimmer for Illumina sequence data". In: *Bioinformatics* 30.15 (2014), pp. 2114–2120. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/btu170`. URL: `https://doi.org/10.1093/bioinformatics/btu170` (visited on 03/02/2022).

[127] Marcel Martin. "Cutadapt removes adapter sequences from high-throughput sequencing reads". In: *EMBnet.journal* 17.1 (May 2011), p. 10. ISSN: 2226-6089. DOI: `10.14806/ej.17.1.200`. URL: `http://journal.embnet.org/index.php/embnetjournal/article/view/200` (visited on 03/02/2022).

[128] Shifu Chen et al. "fastp: an ultra-fast all-in-one FASTQ preprocessor". In: *Bioinformatics* 34.17 (2018), pp. i884–i890. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/bty560`. URL: `https://doi.org/10.1093/bioinformatics/bty560` (visited on 03/02/2022).

[129] Wouter De Coster et al. "NanoPack: visualizing and processing long-read sequencing data". In: *Bioinformatics* 34.15 (2018), pp. 2666–2669. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/bty149`. URL: `https://doi.org/10.1093/bioinformatics/bty149` (visited on 03/02/2022).

[130] Derrick E Wood and Steven L Salzberg. "Kraken: ultrafast metagenomic sequence classification using exact alignments". en. In: *Genome Biology* 15.3 (2014), R46. ISSN: 1465-6906. DOI: `10.1186/gb-2014-15-3-r46`. URL: `http://genomebiology.biomedcentral.com/articles/10.1186/gb-2014-15-3-r46` (visited on 03/18/2022).

[131] Derrick E. Wood, Jennifer Lu, and Ben Langmead. “Improved metagenomic analysis with Kraken 2”. In: *Genome Biology* 20.1 (Nov. 2019), p. 257. ISSN: 1474-760X. DOI: `10.1186/s13059-019-1891-0`. URL: `https://doi.org/10.1186/s13059-019-1891-0` (visited on 03/03/2022).

[132] Jennifer Lu et al. “Bracken: estimating species abundance in metagenomics data”. en. In: *PeerJ Computer Science* 3 (Jan. 2017). Publisher: PeerJ Inc., e104. ISSN: 2376-5992. DOI: `10.7717/peerj-cs.104`. URL: `https://peerj.com/articles/cs-104` (visited on 03/03/2022).

[133] Stephen Nayfach et al. “An integrated metagenomics pipeline for strain profiling reveals novel patterns of bacterial transmission and biogeography”. eng. In: *Genome Research* 26.11 (Nov. 2016), pp. 1612–1625. ISSN: 1549-5469. DOI: `10.1101/gr.201863.115`.

[134] Peter Menzel, Kim Lee Ng, and Anders Krogh. “Fast and sensitive taxonomic classification for metagenomics with Kaiju”. en. In: *Nature Communications* 7.1 (Apr. 2016). Number: 1 Publisher: Nature Publishing Group, p. 11257. ISSN: 2041-1723. DOI: `10.1038/ncomms11257`. URL: `https://www.nature.com/articles/ncomms11257` (visited on 03/03/2022).

[135] Duy Tin Truong et al. “MetaPhlAn2 for enhanced metagenomic taxonomic profiling”. en. In: *Nature Methods* 12.10 (Oct. 2015). Number: 10 Publisher: Nature Publishing Group, pp. 902–903. ISSN: 1548-7105. DOI: `10.1038/nmeth.3589`. URL: `https://www.nature.com/articles/nmeth.3589` (visited on 03/03/2022).

[136] Torsten Seemann. “Prokka: rapid prokaryotic genome annotation”. eng. In: *Bioinformatics (Oxford, England)* 30.14 (July 2014), pp. 2068–2069. ISSN: 1367-4811. DOI: `10.1093/bioinformatics/btu153`.

[137] Ryan R. Wick and Kathryn E. Holt. “Benchmarking of long-read assemblers for prokaryote whole genome sequencing”. en. In: *F1000Research* 8 (Feb. 2021), p. 2138. ISSN: 2046-1402. DOI: `10.12688/f1000research.21782.4`. URL: `https://f1000research.com/articles/8-2138/v4` (visited on 03/25/2021).

[138] Martin Ayling, Matthew D Clark, and Richard M Leggett. “New approaches for metagenome assembly with short reads”. In: *Briefings in Bioinformatics* 21.2 (Mar. 2020), pp. 584–594. ISSN: 1477-4054. DOI: `10.1093/bib/bbz020`. URL: `https://doi.org/10.1093/bib/bbz020` (visited on 09/08/2022).

[139] e. “IUPAC-IUB Commission on Biochemical Nomenclature. Abbreviations and symbols for the description of the conformation of polypeptide chains. Tentative rules (1969)”. en. In: *Biochemistry* 9.18 (Sept. 1970), pp. 3471–3479. ISSN: 0006-2960, 1520-4995. DOI: `10.1021/bi00820a001`. URL: `https://pubs.acs.org/doi/abs/10.1021/bi00820a001` (visited on 03/28/2022).

[140] Tobias Rausch et al. “A consistency-based consensus algorithm for de novo and reference-guided sequence assembly of short reads”. In: *Bioinformatics* 25.9 (May 2009), pp. 1118–1124. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/btp131`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2732307/` (visited on 03/08/2022).

[141] Heidi E. L. Lischer and Kentaro K. Shimizu. “Reference-guided de novo assembly approach improves genome reconstruction for related species”. In: *BMC Bioinformatics* 18.1 (Nov. 2017), p. 474. ISSN: 1471-2105. DOI: `10.1186/s12859-017-1911-6`. URL: `https://doi.org/10.1186/s12859-017-1911-6` (visited on 03/08/2022).

[142] Arash Bayat et al. *Methods for De-novo Genome Assembly*. preprint. LIFE SCIENCES, June 2020. DOI: `10.20944/preprints202006.0324.v1`. URL: `https://www.preprints.org/manuscript/202006.0324/v1` (visited on 03/08/2022).

[143] E. W. Myers et al. “A whole-genome assembly of Drosophila”. eng. In: *Science (New York, N.Y.)* 287.5461 (Mar. 2000), pp. 2196–2204. ISSN: 0036-8075. DOI: `10.1126/science.287.5461.2196`.

[144] Jonathan Laserson, Vladimir Jojic, and Daphne Koller. “Genovo: De Novo Assembly for Metagenomes”. In: *Research in Computational Molecular Biology*. Ed. by David Hutchison et al. Vol. 6044. Series Title: Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 341–356. ISBN: 978-3-642-12682-6 978-3-642-12683-3. DOI: `10.1007/978-3-642-12683-3_22`. URL: `http://link.springer.com/10.1007/978-3-642-12683-3_22` (visited on 03/09/2022).

[145] Afiahayati, Kengo Sato, and Yasubumi Sakakibara. “An extended genovo metagenomic assembler by incorporating paired-end information”. en. In: *PeerJ* 1 (Oct. 2013). Publisher: PeerJ Inc., e196. ISSN: 2167-8359. DOI: `10.7717/peerj.196`. URL: `https://peerj.com/articles/196` (visited on 03/09/2022).

[146] You-Yu Lin et al. “De novo assembly of highly polymorphic metagenomic data using in situ generated reference sequences and a novel BLAST-based assembly pipeline”. In: *BMC Bioinformatics* 18.1 (2017), p. 223. ISSN: 1471-2105. DOI: `10.1186/s12859-017-1630-z`. URL: `https://doi.org/10.1186/s12859-017-1630-z` (visited on 03/09/2022).

[147] Robert Vaser and Mile Šikić. *Yet another de novo genome assembler*. en. preprint. Bioinformatics, May 2019. DOI: `10.1101/656306`. URL: `http://biorxiv.org/lookup/doi/10.1101/656306` (visited on 03/09/2022).

[148] Sergey Koren et al. “Canu: scalable and accurate long-read assembly via adaptive k-mer weighting and repeat separation”. en. In: *Genome Research* 27.5 (May 2017). Company: Cold Spring Harbor Laboratory Press Distributor: Cold Spring Harbor Laboratory Press Institution: Cold Spring Harbor Laboratory Press Label: Cold Spring Harbor Laboratory Press Publisher: Cold Spring Harbor Lab, pp. 722–

736. ISSN: 1088-9051, 1549-5469. DOI: 10.1101/gr.215087.116. URL: https://genome.cshlp.org/content/27/5/722 (visited on 03/09/2022).

[149] Anton Bankevich et al. “SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing”. en. In: *Journal of Computational Biology* 19.5 (May 2012), pp. 455–477. ISSN: 1066-5277, 1557-8666. DOI: 10.1089/cmb.2012.0021. URL: http://www.liebertpub.com/doi/10.1089/cmb.2012.0021 (visited on 03/18/2022).

[150] Alexandre Souvorov, Richa Agarwala, and David J. Lipman. “SKESA: strategic k-mer extension for scrupulous assemblies”. In: *Genome Biology* 19.1 (Oct. 2018), p. 153. ISSN: 1474-760X. DOI: 10.1186/s13059-018-1540-z. URL: https://doi.org/10.1186/s13059-018-1540-z (visited on 03/14/2022).

[151] Dinghua Li et al. “MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph”. In: *Bioinformatics* 31.10 (May 2015), pp. 1674–1676. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btv033. URL: https://doi.org/10.1093/bioinformatics/btv033 (visited on 03/14/2022).

[152] Daniel R. Zerbino and Ewan Birney. “Velvet: Algorithms for de novo short read assembly using de Bruijn graphs”. en. In: *Genome Research* 18.5 (May 2008). Company: Cold Spring Harbor Laboratory Press Distributor: Cold Spring Harbor Laboratory Press Institution: Cold Spring Harbor Laboratory Press Label: Cold Spring Harbor Laboratory Press Publisher: Cold Spring Harbor Lab, pp. 821–829. ISSN: 1088-9051, 1549-5469. DOI: 10.1101/gr.074492.107. URL: https://genome.cshlp.org/content/18/5/821 (visited on 03/14/2022).

[153] Sébastien Boisvert, François Laviolette, and Jacques Corbeil. “Ray: Simultaneous Assembly of Reads from a Mix of High-Throughput Sequencing Technologies”. en. In: *Journal of Computational Biology* 17.11 (Nov. 2010), pp. 1519–1533. ISSN: 1066-5277, 1557-8666. DOI: 10.1089/cmb.2009.0238. URL: http://www.liebertpub.com/doi/10.1089/cmb.2009.0238 (visited on 03/14/2022).

[154] Ruibang Luo et al. “SOAPdenovo2: an empirically improved memory-efficient short-read de novo assembler”. In: *GigaScience* 1.1 (Dec. 2012), pp. 2047–217X–1–18. ISSN: 2047-217X. DOI: 10.1186/2047-217X-1-18. URL: https://doi.org/10.1186/2047-217X-1-18 (visited on 03/14/2022).

[155] Alexey Gurevich et al. “QUAST: quality assessment tool for genome assemblies”. eng. In: *Bioinformatics (Oxford, England)* 29.8 (Apr. 2013), pp. 1072–1075. ISSN: 1367-4811. DOI: 10.1093/bioinformatics/btt086.

[156] Nancy Manchanda et al. “GenomeQC: a quality assessment tool for genome assemblies and gene structure annotations”. en. In: *BMC Genomics* 21.1 (Dec. 2020), p. 193. ISSN: 1471-2164. DOI: 10.1186/s12864-020-6568-2. URL: https://bmcgenomics.biomedcentral.com/articles/10.1186/s12864-020-6568-2 (visited on 08/11/2021).

[157] Jason A. Papin et al. “Improving reproducibility in computational biology research”. en. In: *PLOS Computational Biology* 16.5 (May 2020). Publisher: Public Library of Science, e1007881. ISSN: 1553-7358. DOI: `10.1371/journal.pcbi.1007881`. URL: `https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1007881` (visited on 04/01/2022).

[158] Jon F. Claerbout and Martin Karrenbach. “Electronic documents give reproducible research a new meaning”. en. In: *SEG Technical Program Expanded Abstracts 1992*. Society of Exploration Geophysicists, Jan. 1992, pp. 601–604. DOI: `10.1190/1.1822162`. URL: `http://library.seg.org/doi/abs/10.1190/1.1822162` (visited on 04/01/2022).

[159] Victoria Stodden, Jennifer Seiler, and Zhaokun Ma. “An empirical analysis of journal policy effectiveness for computational reproducibility”. en. In: *Proceedings of the National Academy of Sciences* 115.11 (Mar. 2018), pp. 2584–2589. ISSN: 0027-8424, 1091-6490. DOI: `10.1073/pnas.1708290115`. URL: `https://pnas.org/doi/full/10.1073/pnas.1708290115` (visited on 04/01/2022).

[160] Keith A. Baggerly and Kevin R. Coombes. “Deriving chemosensitivity from cell lines: Forensic bioinformatics and reproducible research in high-throughput biology”. In: *The Annals of Applied Statistics* 3.4 (Dec. 2009). ISSN: 1932-6157. DOI: `10.1214/09-AOAS291`. URL: `https://projecteuclid.org/journals/annals-of-applied-statistics/volume-3/issue-4/Deriving-chemosensitivity-from-cell-lines--Forensic-bioinformatics-and-reproducible/10.1214/09-AOAS291.full` (visited on 04/01/2022).

[161] Yang-Min Kim, Jean-Baptiste Poline, and Guillaume Dumas. “Experimenting with reproducibility: a case study of robustness in bioinformatics”. In: *GigaScience* 7.7 (July 2018), giy077. ISSN: 2047-217X. DOI: `10.1093/gigascience/giy077`. URL: `https://doi.org/10.1093/gigascience/giy077` (visited on 04/01/2022).

[162] Yunda Huang and Raphael Gottardo. “Comparability and reproducibility of biomedical data”. eng. In: *Briefings in Bioinformatics* 14.4 (July 2013), pp. 391–401. ISSN: 1477-4054. DOI: `10.1093/bib/bbs078`.

[163] Mark D. Wilkinson et al. “The FAIR Guiding Principles for scientific data management and stewardship”. en. In: *Scientific Data* 3.1 (Mar. 2016). Number: 1 Publisher: Nature Publishing Group, p. 160018. ISSN: 2052-4463. DOI: `10.1038/sdata.2016.18`. URL: `https://www.nature.com/articles/sdata201618` (visited on 04/01/2022).

[164] Karthik Ram. “Git can facilitate greater reproducibility and increased transparency in science”. en. In: *Source Code for Biology and Medicine* 8.1 (Dec. 2013), p. 7. ISSN: 1751-0473. DOI: `10.1186/1751-0473-8-7`. URL: `https://scfbm.biomedcentral.com/articles/10.1186/1751-0473-8-7` (visited on 04/01/2022).

[165] Stephen R. Piccolo and Michael B. Frampton. “Tools and techniques for computational reproducibility”. en. In: *GigaScience* 5.1 (Dec. 2016), p. 30. ISSN: 2047-217X. DOI: `10.1186/s13742-016-0135-4`. URL: `https://academic.oup.com/gigascience/article-lookup/doi/10.1186/s13742-016-0135-4` (visited on 04/01/2022).

[166] Carl Boettiger. “An introduction to Docker for reproducible research”. en. In: *ACM SIGOPS Operating Systems Review* 49.1 (Jan. 2015), pp. 71–79. ISSN: 0163-5980. DOI: `10.1145/2723872.2723882`. URL: `https://dl.acm.org/doi/10.1145/2723872.2723882` (visited on 04/01/2022).

[167] Geir Kjetil Sandve et al. “Ten Simple Rules for Reproducible Computational Research”. en. In: *PLOS Computational Biology* 9.10 (Oct. 2013). Publisher: Public Library of Science, e1003285. ISSN: 1553-7358. DOI: `10.1371/journal.pcbi.1003285`. URL: `https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003285` (visited on 04/01/2022).

[168] Gregory M. Kurtzer, Vanessa Sochat, and Michael W. Bauer. “Singularity: Scientific containers for mobility of compute”. en. In: *PLOS ONE* 12.5 (Nov. 2017). Publisher: Public Library of Science, e0177459. ISSN: 1932-6203. DOI: `10.1371/journal.pone.0177459`. URL: `https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0177459` (visited on 03/17/2022).

[169] Paolo Di Tommaso et al. “Nextflow enables reproducible computational workflows”. en. In: *Nature Biotechnology* 35.4 (Apr. 2017), pp. 316–319. ISSN: 1087-0156, 1546-1696. DOI: `10.1038/nbt.3820`. URL: `http://www.nature.com/articles/nbt.3820` (visited on 03/24/2021).

[170] Felix Mölder et al. “Sustainable data analysis with Snakemake”. In: *F1000Research* 10 (Apr. 2021), p. 33. ISSN: 2046-1402. DOI: `10.12688/f1000research.29032.2`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8114187/` (visited on 03/17/2022).

[171] Enis Afgan et al. “The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update”. In: *Nucleic Acids Research* 46.W1 (July 2018), W537–W544. ISSN: 0305-1048. DOI: `10.1093/nar/gky379`. URL: `https://doi.org/10.1093/nar/gky379` (visited on 03/17/2022).

[172] Matthew Krafczyk et al. “Scientific Tests and Continuous Integration Strategies to Enhance Reproducibility in the Scientific Software Context”. en. In: *Proceedings of the 2nd International Workshop on Practical Reproducible Evaluation of Computer Systems - P-RECS ’19*. Phoenix, AZ, USA: ACM Press, 2019, pp. 23–28. ISBN: 978-1-4503-6756-1. DOI: `10.1145/3322790.3330595`. URL: `http://dl.acm.org/citation.cfm?doid=3322790.3330595` (visited on 04/29/2022).

[173] Sarah K. Hilton et al. “Metataxonomic and Metagenomic Approaches vs. Culture-Based Techniques for Clinical Pathology”. In: *Frontiers in Microbiology* 7 (2016). ISSN: 1664-302X. URL: `https://www.frontiersin.org/article/10.3389/fmicb.2016.00484` (visited on 03/03/2022).

[174] Evan Bolyen et al. “Reproducible, interactive, scalable and extensible microbiome data science using QIIME 2”. en. In: *Nature Biotechnology* 37.8 (Aug. 2019). Number: 8 Publisher: Nature Publishing Group, pp. 852–857. ISSN: 1546-1696. DOI: `10.1038/s41587-019-0209-9`. URL: `https://www.nature.com/articles/s41587-019-0209-9` (visited on 03/03/2022).

[175] Patrick D. Schloss et al. “Introducing mothur: Open-Source, Platform-Independent, Community-Supported Software for Describing and Comparing Microbial Communities”. In: *Applied and Environmental Microbiology* 75.23 (Dec. 2009). Publisher: American Society for Microbiology, pp. 7537–7541. DOI: `10.1128/AEM.01541-09`. URL: `https://journals.asm.org/doi/10.1128/AEM.01541-09` (visited on 03/04/2022).

[176] Robert C. Edgar. “UPARSE: highly accurate OTU sequences from microbial amplicon reads”. en. In: *Nature Methods* 10.10 (Oct. 2013). Number: 10 Publisher: Nature Publishing Group, pp. 996–998. ISSN: 1548-7105. DOI: `10.1038/nmeth.2604`. URL: `https://www.nature.com/articles/nmeth.2604` (visited on 03/04/2022).

[177] Moira Marizzoni et al. “Comparison of Bioinformatics Pipelines and Operating Systems for the Analyses of 16S rRNA Gene Amplicon Sequences in Human Fecal Samples”. In: *Frontiers in Microbiology* 11 (2020). ISSN: 1664-302X. URL: `https://www.frontiersin.org/article/10.3389/fmicb.2020.01262` (visited on 03/04/2022).

[178] Sarah L. Westcott and Patrick D. Schloss. “De novo clustering methods outperform reference-based methods for assigning 16S rRNA gene sequences to operational taxonomic units”. In: *PeerJ* 3 (Dec. 2015), e1487. ISSN: 2167-8359. DOI: `10.7717/peerj.1487`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4675110/` (visited on 03/04/2022).

[179] Xiaolin Hao, Rui Jiang, and Ting Chen. “Clustering 16S rRNA for OTU prediction: a method of unsupervised Bayesian clustering”. en. In: *Bioinformatics* 27.5 (Mar. 2011), pp. 611–618. ISSN: 1460-2059, 1367-4803. DOI: `10.1093/bioinformatics/btq725`. URL: `https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btq725` (visited on 03/04/2022).

[180] Sergey Nurk et al. “metaSPAdes: a new versatile metagenomic assembler”. en. In: *Genome Research* 27.5 (May 2017), pp. 824–834. ISSN: 1088-9051, 1549-5469. DOI: `10.1101/gr.213959.116`. URL: `http://genome.cshlp.org/lookup/doi/10.1101/gr.213959.116` (visited on 03/25/2021).

[181] Nathan D Olson et al. “Metagenomic assembly through the lens of validation: recent advances in assessing and improving the quality of genomes assembled from metagenomes”. In: *Briefings in Bioinformatics* 20.4 (Aug. 2017), pp. 1140–1150. ISSN: 1467-5463. DOI: `10.1093/bib/bbx098`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6781575/` (visited on 03/17/2022).

[182] Mikhail Kolmogorov et al. “metaFlye: scalable long-read metagenome assembly using repeat graphs”. en. In: *Nature Methods* 17.11 (Nov. 2020). Number: 11 Publisher: Nature Publishing Group, pp. 1103–1110. ISSN: 1548-7105. DOI: `10.1038/s41592-020-00971-x`. URL: `https://www.nature.com/articles/s41592-020-00971-x` (visited on 03/20/2022).

[183] Hanno Teeling and Frank Oliver Glöckner. “Current opportunities and challenges in microbial metagenome analysis–a bioinformatic perspective”. eng. In: *Briefings in Bioinformatics* 13.6 (Nov. 2012), pp. 728–742. ISSN: 1477-4054. DOI: `10.1093/bib/bbs039`.

[184] Karel Sedlar, Kristyna Kupkova, and Ivo Provaznik. “Bioinformatics strategies for taxonomy independent binning and visualization of sequences in shotgun metagenomics”. eng. In: *Computational and Structural Biotechnology Journal* 15 (2017), pp. 48–55. ISSN: 2001-0370. DOI: `10.1016/j.csbj.2016.11.005`.

[185] Yu-Wei Wu, Blake A. Simmons, and Steven W. Singer. “MaxBin 2.0: an automated binning algorithm to recover genomes from multiple metagenomic datasets”. eng. In: *Bioinformatics (Oxford, England)* 32.4 (Feb. 2016), pp. 605–607. ISSN: 1367-4811. DOI: `10.1093/bioinformatics/btv638`.

[186] Ivan Gregor et al. “PhyloPythiaS+: a self-training method for the rapid reconstruction of low-ranking taxonomic bins from metagenomes”. en. In: *PeerJ* 4 (Feb. 2016). Publisher: PeerJ Inc., e1603. ISSN: 2167-8359. DOI: `10.7717/peerj.1603`. URL: `https://peerj.com/articles/1603` (visited on 03/20/2022).

Part I

Applying metagenomics in the clinical context

Chapter 2

Critical steps in clinical shotgun metagenomics for the concomitant detection and typing of microbial pathogens

2. CRITICAL STEPS IN CLINICAL SHOTGUN METAGENOMICS FOR THE CONCOMITANT DETECTION AND TYPING OF MICROBIAL PATHOGENS

This chapter is a reproduction of the following publication:

N. Couto, L. Schuele, E.C. Raangs, M. P. Machado, C. I. Mendes, T. F. Jesus, M. Chlebowicz, S. Rosema, M. Ramirez, J. A. Carriço, I. B. Autenrieth, A. W. Friedrich, S. Peter and J. W. Rossen. Critical steps in clinical shotgun metagenomics for the concomitant detection and typing of microbial pathogens. Sci Rep 8, 13767 (2018). DOI: `https://doi.org/10.1038/s41598-018-31873-w`

The supplementary information referred throughout the text can be consulted in this chapter before the section of references.

As mentioned in Chapter 1, section 1.2.3.2, SMg approaches have been a growing interest to deliver clinically relevant results without *a priori* knowledge of what to expect from a particular clinical sample or patient. The capacity to detect all potential pathogens in a sample has great potential utility in the diagnosis of infectious disease. However, it is unclear how the variety of available methods impacts the end results.

In this publication, SMg was applied to nine body fluid samples and one tissue sample from patients at the University Medical Center Groningen with varying degrees of contamination: one sample from peritoneal fluid, five from pus, two from synovial fluid of knees with a prosthesis, one from sputum and one from a bone biopsy. The results of microbial identification through whole genome sequencing (WGS) and SMg were compared to standard culture-based microbiological methods. In order to evaluate and compare the accuracy and reliability of the bioinformatics analyses in providing the closest results to culture and WGS of any cultured isolates, three different bioinformatic pipelines (two commercially and one freely available) were used. Most pathogens identified by culture were also identified through metagenomics, but substantial differences were noted between the taxonomic classification tools.

My contribution to this publication included the bioinformatics analysis of all the samples using a Unix-based approach. I performed quality assessment and quality control of the WGS and SMg data, the removal of host sequencing from the samples, and the taxonomic identification of the remaining reads in each sample through 3 different methods: MetaPhlan2, Kraken and MIDAS. Gene detection directly from the reads for bacterial typing was also performed using metaMLST, ReMatCh, Bowtie2 and Samtools. Finally, the reads were assembled using the SPAdes genome assembler, with and without metagenomic mode according to the sample being processed.

Critical steps in clinical shotgun metagenomics for the concomitant detection and typing of microbial pathogens

Natacha Couto[1], Leonard Schuele[1,2], Erwin C. Raangs[1], Miguel P. Machado[3], Catarina I. Mendes[1,3], Tiago F. Jesus[3], Monika Chlebowicz[1], Sigrid Rosema[1], Mário Ramirez[3], João A. Carriço[3], Ingo B. Autenrieth[2], Alex W. Friedrich[1], Silke Peter[2], John W. Rossen[1]

[1] University of Groningen, University Medical Center Groningen, Department of Medical Microbiology, Groningen, The Netherlands;

[2] Institute of Medical Microbiology and Hygiene, University of Tübingen, Germany;

[3] Instituto de Microbiologia, Instituto de Medicina Molecular, Faculdade de Medicina, Universidade de Lisboa, Portugal.

2.1 Abstract

High throughput sequencing has been proposed as a one-stop solution for diagnostics and molecular typing directly from patient samples, allowing timely and appropriate implementation of measures for treatment, infection prevention and control. However, it is unclear how the variety of available methods impacts the end results. We applied shotgun metagenomics on diverse types of patient samples using three different methods to deplete human DNA prior to DNA extraction. Libraries were prepared and sequenced with Illumina chemistry. Data was analysed using methods likely to be available in clinical microbiology laboratories using genomics. The results of microbial identification were compared to standard culture-based microbiological methods. On average, 75% of the reads were corresponded to human DNA, being a major determinant in the analysis outcome. None of the kits was clearly superior suggesting that the initial ratio between host and microbial DNA or other sample characteristics were the major determinants of the proportion of microbial reads. Most pathogens identified by culture were also identified through metagenomics, but substantial differences were noted between the taxonomic classification tools. In two cases the high number of human reads resulted in insufficient sequencing depth of bacterial DNA for identification. In three samples, we could infer the probable multilocus sequence type of the most abundant species. The tools and databases used for taxonomic classification and antimicrobial resistance identification had a key impact on the results, recommending that efforts need to be aimed at standardisation of the analysis methods if metagenomics is to be used routinely in clinical microbiology.

2.2 Introduction

Classical microbial culture is still considered the gold standard in medical microbiology. Several molecular detection techniques have been implemented but these are generally geared towards specific pathogens (e.g. specific RT-PCR or microarrays). Even when unbiased molecular approaches are used, such as 16S/18S rRNA gene sequencing, these do not provide all the information that can be obtained by culturing, e.g., antimicrobial susceptibility and molecular typing information. However, microbial culture is laborious and time-consuming and new methods are needed to replace it. Ideally, a single method should provide rapid identification and characterisation of clinically relevant pathogens directly from a sample in order to guide therapy, predict potential treatment failures and to reveal possible transmission events.

SMg is a culture-independent technique that provides valuable information not only at the identification level, but also at the level of molecular characterisation. Studies have shown that it has added value in terms of detection sensitivity and personalised treatment in clinical microbiology, when identifying bacteria [1, 2] or viruses [3]. Indeed Gyarmati et al., 2016 [4], used a sequence-based metagenomics approach directly from blood to detect non-culturable, difficult-to-culture and non-bacterial pathogens. The authors were able, through SMg, to detect viral and fungal pathogens together with bacteria, which had not been detected through classical microbiology. Additionally, SMg can be used for infection prevention, having the potential to identify transmission events directly from clinical samples [5]. For example, SMg was proven valuable for the identification of inter-host nucleotide variations occurring after direct transmission of noroviruses causing gastroenteritis [5]. Hasman and colleagues (2014) [1] were able to identify urinary pathogens directly from urine, as well as antimicrobial resistant genes compatible with the resistant phenotype determined through antimicrobial susceptibility testing. They also identified almost perfect phylogenetic matches between WGS data obtained by metagenomics and WGS of pure isolates.

Despite the promise of SMg of becoming a one-stop solution in clinical microbiology, SMg still has several challenges to overcome. One of the greatest challenges is the choice of the extraction and sequencing protocols, as well of the type of controls [6]. The extraction protocol should efficiently and specifically isolate microbial DNA/RNA, while removing the host DNA/RNA [7]. However, the variety of clinical samples used in the diagnosis of distinct types of infection (e.g. tissues versus fluids), poses a serious challenge for standardisation, an essential step if these methods are to be used by routine diagnostic laboratories. The sequencing protocol is also dependent on the pathogens of interest (e.g. bacteria versus viruses), sequencing strategy (DNA and/or RNA), required turnaround time, sequencing depth and error tolerance [6]. The use of defined controls is necessary for validation of each experiment and these should be adapted for every type of infection and sample type and should consist of a combination of known positive specimens, pathogen-negative patient specimens and pathogen-negative patient specimens spiked with live microorganisms or pure

DNA [6].

Another potential challenge are the metagenomics analysis tools. Recent studies have evaluated the different SMg sequence classification methods [8]. These use different methodologies for classification: sequence similarity-based methods, sequence composition-based methods and hybrid methods [8]. They differ not only in the algorithms for detecting the microorganisms present, but also in the databases used. This high variability leads to different results, not only at the microorganism classification level but also when evaluating the relative abundance of these pathogens [8]. A recent study evaluated the accuracy of 38 bioinformatics methods using both *in silico* and *in vitro* generated mock bacterial communities. Dozens to hundreds of species were falsely predicted by the most popular software, and no software clearly outperformed the others [8]. In the absence of studies comparing the outputs of different analysis methods in clinical samples, users may decide which methods to use based on personal experience with a given tool, availability of the tool in the laboratory or its ease of use. This poses a great challenge when providing reproducible results and creates uncertainty regarding the reliability of the information derived. This is a major barrier to the implementation of SMg approaches in routine clinical microbiology laboratories.

In this study, the aim was to identify the critical steps when using SMg for the identification and characterization of microbial pathogens directly from clinical specimens using methods that are likely to be available in clinical microbiology laboratories wanting to implement genomics for pathogen identification or molecular epidemiology studies. For this purpose, we used three human-DNA depletion kits and evaluated a diverse set of bioinformatics tools (commercial and non-commercial) in order to investigate how well they performed and what would the differences be in terms of taxonomic classification, antimicrobial resistance gene detection and typing directly from patient samples, bypassing culture.

2.3 Methods

2.3.1 Sample collection

Nine body fluid samples and one tissue sample entering the Medical Microbiology laboratory were selected for metagenomics sequencing. These included one sample from peritoneal fluid, five from pus (3 abscesses and 2 empyema), two from synovial fluid of knees with prosthesis, one from sputum and one from a bone biopsy (Table 2.1). All samples were stored at 4ºC for a variable period (2-10 days). The samples used for the present analyses were collected during routine diagnostics and infection prevention and control investigations. All procedures were carried out according to guidelines and regulations of University Medical Centre Groningen (UMCG) concerning the use of patient materials for the validation of clinical methods, which are in compliance with the guidelines of the Federation of Dutch

Medical Scientific Societies (FDMSS). Every patient entering the UMCG is informed that samples taken may be used for research and publication purposes, unless they indicate that they do not agree to it. This procedure has been approved by the Medical Ethical Committee of the UMCG. Informed consent was obtained from all individuals or their guardians prior to study participation. All samples were used after performing and completing a conventional microbiological diagnosis and were coded to protect patients' confidentiality. All experiments were performed in accordance with the guidelines of the Declaration of Helsinki and the institutional regulations.

2.3.2 Classic culturing and susceptibility testing

The samples were cultured following methods routinely used in our institution. Briefly, samples were streaked onto five plates (Mediaproducts BV, Groningen, The Netherlands) - blood agar (aerobic), chocolate agar (aerobic), McConkey agar (aerobic), Brucella agar (anaerobic) and Sabouraud Dextrose +AV (aerobic) - and incubated overnight under aerobic and anaerobic atmosphere at 37°C. The two pus samples were also plated onto Phenylethyl alcohol sheep blood agar (PEA), Kanamycin vancomycin laked blood (KVLB) agar and Bacteroides bile esculin (BBE) agar and incubated under anaerobic conditions overnight. The isolates recovered were subjected to susceptibility testing by Vitek 2 using either the AST-P559 (Gram-positive bacteria) or the AST-N344 (Gram-negative bacteria) card (bioMérieux, Marcy-l'Étoile, France) and identified by MALDI-TOF MS (Bruker Daltonik, Gmbh, Germany) using standard protocols.

Table 2.1: Characteristics of the samples and mapping of trimmed reads against a human genome hg19 (%) using CLC Genomics Workbench v10.0.1.

Sample	**Sample type**	**DNA extraction method**	**Total number of reads**	**Mapped reads against hg19**	**Unmapped reads**
Sample 1	Peritoneal fluid	Ultra-Deep Microbiome Prep (Molzym)	5892978	5,249,063 (89.2%)	632,951 (10.8%)
Sample 2	Pus (abscess)	Ultra-Deep Microbiome Prep (Molzym)	9603346	7,828.746 (81.6%)	1,770,558 (18.4%)
Sample 3	Synovial fluid	Ultra-Deep Microbiome Prep (Molzym)	8615810	8,254,594 (95.9%)	355,200 (4.1%)
Sample 4	Synovial fluid	Ultra-Deep Microbiome Prep (Molzym)	6078166	6,015,945 (99.0%)	61,099 (1.0%)
Sample 5	Pus (abscess)	Ultra-Deep Microbiome Prep (Molzym)	8368930	309,588 (3.7%)	8,052,272 (96.3%)
Sample 6	Pus (empyema)	QIAamp DNA Microbiome Kit (Qiagen)	2912802	2,877,066 (98.8%)	34,506 (1.1%)
Sample 7	Pus (empyema)	QIAamp DNA Microbiome Kit (Qiagen)	1486700	922,932 (62.2%)	561,772 (37.8%)
Sample 8	Bone biopsy	Micro-DXTM (Molzym)	6534866	229,149 (3.5%)	6,303,803 (96.5%)
Sample 9	Pus (abscess)	Micro-DXTM (Molzym)	6173132	6,081,612 (98.5%)	89,922 (1.5%)
Sample 10	Sputum	Micro-DXTM (Molzym)	7596836	7,337,832 (96.7%)	235,520 (3.3%)
Negative control	Water	QIAamp DNA Microbiome Kit (Qiagen)	1730738	1,706,861 (98.9%)	19,805 (1.2%)

2.3.3 DNA extraction, library preparation and sequencing

The DNA for metagenomic sequencing was isolated using the Ultra-Deep Microbiome Prep (Molzym Life Science, Bremen, Germany), Micro-Dx™kit (Molzym Life Science) or QIAamp DNA Microbiome Kit (Qiagen, Hilden, Germany) directly from the clinical samples and a negative control consisting of a mock sample of DNA and RNA free water (Table 2.1). These kits include human DNA depletion steps. The QIAamp DNA Microbiome Kit was used according to the manufacturer's protocol with an additional 5 min air-dry step before elution. For microbial lysis, a Precellys 24 homogeniser (Bertin, Montigny-le-Bretonneux, France) set to 3 times 30 seconds at 5000 rpm separated by 30 seconds was used. After extraction, DNA was quantified with the Qubit 2.0 (Life Technologies, ThermoFisher Scientific, Waltham, Massachusetts, EUA) and NanoDrop 2000 (ThermoFisher Scientific). The DNA quality was assessed using the Genomic DNA ScreenTape and Agilent 2200 TapeStation System (Agilent Technologies, California, United States of America). Isolated DNA was purified using Agencourt AMPure XP beads (Beckman Coulter, California, United States of America) according to the manufacturer's instructions, to eliminate small DNA fragments and chemical contaminants (e.g. benzonase). The DNA was then diluted to 0.2 ng/μl and 1 ng was used for the library preparation, using the Nextera XT Library Preparation kit (Illumina, California, United States of America), according to the manufacturer's protocol. Cluster generation and sequencing were performed with the MiSeq Reagent Kit v2 500-cycles Paired-End in a MiSeq instrument (Illumina). Samples were sequenced in batches of 5 samples on a single flow cell.

For the DNA extraction of bacterial isolates (when an isolate was recovered from culture), we used the UltraClean Microbial DNA Isolation Kit (Mo Bio), with some modifications. We started with solid cultures and resuspended a 10 μl-loopfull of culture directly into the tube with the microbeads and microbead solution. The library preparation, cluster generation and sequencing was performed as described above. Strains were sequenced in batches of 12 to 16 on a single flow cell.

2.3.4 Bioinformatics analyses

In order to evaluate and compare the accuracy and reliability of the bioinformatics analyses in providing the closest results to culture and WGS of any cultured isolates, three different pipelines (two commercially and one freely available) were used (Figure 2.1). Different tools to perform raw read quality control, filtering and trimming were used and reads were mapped against the human genome (hg19) before performing taxonomic classification. Reads mapping to hg19 where removed from the analysis to increase the efficiency of the bioinformatics tools. Typing (MLST), phylogenetic analysis, plasmid analysis, detection of antimicrobial resistance and virulence genes was performed. To determine the appropriateness of SMg as predictor of the WGS (chromosome and plasmids), SMg results obtained were compared with the results of WGS of any bacterial isolates obtained from culturing the sample.

All the parameters used in each approach are available in Supplementary Table 1 (see 2.8.1).

2.3.4.1 Unix-based approach

For the metagenomics data, read quality control and cleaning was performed using FastQC v0.11.5 and Trimmomatic v0.36, respectively, through the INNUca v2.6 pipeline[*], excluding assembly and polishing. Using a reference mapping approach against the human genome (UCSC hg19), human reads were discarded using Bowtie 2 v2.3.2 [9] and SAMtools v1.3.1 [10]. Those paired reads that did not map against the human genome were used in subsequent analyses. The bacterial species were identified through Kraken v0.10.5-beta [11] using the miniKraken database (pre-built 4 GB database constructed from complete bacterial, archaeal and viral genomes in RefSeq, as of Dec. 8, 2014), MIDAS [12] using the midas_db_v1.2 database (>30,000 bacterial reference genomes, as of May 9, 2018) and MetaPhlAn2 v2.0 [13] using the database provided by the tool ($\sim$13,500 bacterial and archaeal, $\sim$3,500 viral, and $\sim$110 eukaryotic reference genomes, as of May 9, 2018). The sequence type (ST) was obtained through metaMLST v1.1 [14] based on the metamlstDB_2017. Antimicrobial resistance genes were detected using ReMatCh v3.2[†], a read mapping tool that uses Bowtie 2 v2.3.2 [9] and the following rules for gene presence/absence: genes were considered present when $\geq 80\%$ of the reference sequence was covered and the sample sequence was $\geq 70\%$ identical to the one used as reference. For that, ResFinder database (2231 genes, downloaded on 29-06-2017) was used as reference and, due to the low coverage of microbial metagenomics samples, a minimal coverage depth of 1 read was set to consider a reference sequence position as covered (and therefore present in the sample data), as well as to perform base call (used for sequence identity determination). Finally, the assembly was accomplished through SPAdes v3.10.1 [15].

Plasmid detection was achieved by running the script PlasmidCoverage[‡], using the plasmid sequences downloaded from NCBI RefSeq (`ftp://ftp.ncbi.nlm.nih.gov/genomes/refseq/plasmid/`, as of May 11, 2017). The script uses Bowtie 2 v2.2.9 [9], to map the pre-processed input reads against the plasmid database (Bowtie2 index for all plasmid sequences). For Bowtie 2 we used the '-k' option, allowing each read to map to as many plasmid sequences as present in the NCBI RefSeq plasmid database (since plasmid sequences are modular) [16, 17]. Then, this pipeline used SAMtools v1.3.1 [10] to estimate the coverage for each position, and reported the length of plasmid sequence covered (in percentage) and average depth (mean number of reads mapped against a given position in each plasmid). Plasmids with less than 80% of its length covered were excluded from the

---

[*] `https://github.com/B-UMMI/INNUca/`

[†] `https://github.com/B-UMMI/ReMatCh/`

[‡] `https://github.com/tiagofilipe12/PlasmidCoverage`

final results in line with what has described elsewhere [18]. The pATLAS tool[§] was used to visualise which plasmids were present.

For the WGS reads of the bacterial isolates, the whole INNUca v2.6 pipeline was run, including SPAdes assembly and polishing. Plasmids were detected as mentioned previously.

2.3.4.2 Commercial-based approach

The fastq files containing the reads were uploaded into CLC Genomics Workbench v10.1.1, using the following options: Illumina import, paired-reads, paired-end (forward-reverse) and minimum distance of 1 and a maximum distance of 1000 (default). The trimming was performed using the default settings, except the quality trimming score limit was set to 0.01 and we added a Trim adapter list containing Illumina adapters. The mapping was performed with the Map Reads to Reference tool, using the hg19 genome as reference. The default settings were used with the addition of the collect un-mapped reads option. The *de novo* assembly tool was used for the assembly (even for the metagenomics reads) and, apart from the word size, which was changed to 29, all the settings were default. Two tools were used for the microbial identification, Taxonomic Profiling and Find Best Matches using K-mer Spectra (Microbial Genomics Module). In both, the bacterial and fungal databases were downloaded from NCBI RefSeq (with the Only Complete Genomes option turned off; minimum length 500,000 nucleotides) on 08-07-2017 (bacterial, 70,868 sequences) and 25-05-2017 (fungal, 377 sequences). The antimicrobial resistance genes were detected, based on the assembled contigs, using the Find Resistance tool (Microbial Genomics Module) and were initially only considered present when they were $\geq 70\%$ identical to the reference and $\geq 80\%$ of the sequence was covered. The analysis was also repeated using $\geq 40\%$ and $\geq 20\%$ of sequence coverage for comparison purposes. The database containing the antimicrobial resistance genes was downloaded directly to the software from ResFinder[¶] (downloaded on 05-07-2017, 2156 sequences). The MLST was determined through the Identify MLST tool (Microbial Genomics Module), using all MLST schemes available at PubMLST (04-03-2017). The same database used for plasmid detection in Unix, was used for mapping the reads in CLC Genomics Workbench. Again, plasmids with less than 80% of its length covered were excluded from the final results. For WGS reads we used the Trim Sequences tool and the assembly, antimicrobial resistance genes detection, and MLST determination were performed as before.

[§] `http://www.patlas.site/`

[¶] `https://cge.cbs.dtu.dk/services/data.php`

2.3.4.3 Web-based approaches

The fastq files containing the reads were uploaded into the BaseSpace[‖] website. First, the raw forward and reverse fastq reads were subjected to FASTQ Toolkit for adapter/quality trimming and length filtering with standard settings and length filtering adjusted to a minimum of 100 and a maximum of 500. The trimmed reads were then used as input for all the following processes. The available microorganism identification apps Kraken v1.0.0, MetaPhlAn v1.0.0 and GENIUS v.1.1.0 were used with the standard settings/parameters. SEAR was used to detect antimicrobial resistance genes, maintaining the standard settings except for the clustering stringency which was set to 0.98 and the annotation stringency was set to 40. The SPAdes Genome Assembler v3.9.0 app was run with the standard parameters for multi cell data type. For metagenomic datatype settings, the running mode was set to only assembly and careful mode was disabled.

The reads were uploaded into CosmosID[**] and Taxonomer[††] [19] directly without any quality trimming. We used the Full Analysis mode in Taxonomer.

2.3.4.4 wgMLST analyses

Typing was done by MLST and wgMLST analyses obtained using Ridom SeqSphere+ v4.0.1. The genomic data (assembled contigs) obtained from SMg was compared to the data obtained through WGS. Since no cg/wg MLST scheme was available for *Escherichia coli*, *Enterococcus faecalis*, *Ochrobactrum intermedium* and *Staphylococcus haemolyticus*, cgMLST and accessory genome schemes were constructed, using Ridom SeqSphere+ cgMLST Target Definer with the following parameters: a minimum length filter that removes all genes smaller than 50 bp; a start codon filter that discards all genes that contain no start codon at the beginning of the gene; a stop codon filter that discards all genes that contain no stop codon or more than one stop codon or that do not have the stop codon at the end of the gene; a homologous gene filter that discards all genes with fragments that occur in multiple copies within a genome (with identity of 90% and >100 bp overlap); and a gene overlap filter that discards the shorter gene from the cgMLST scheme if the two genes affected overlap >4 bp. The remaining genes were then used in a pairwise comparison using BLAST version 2.2.12 (parameters used were word size 11, mismatch penalty -1, match reward 1, gap open costs 5, and gap extension costs 2). All genes of the reference genome that were common in all query genomes with a sequence identity of $\geq$ 90% and 100% overlap and, with the default parameter stop codon percentage filter turned on, formed the final cgMLST scheme. The combination of all alleles in each strain formed an allelic profile that was used to generate minimum spanning trees using the parameter "pairwise ignore missing values" during distance calculation [20].

[‖] `https://basespace.illumina.com`

[**] `https://app.cosmosid.com/login`

[††] `https://www.taxonomer.com/`

2.3.4.5 Statistical analysis

The sensitivity and positive predictive value of each taxonomic classification method were determined. Classical culture and MALDI-TOF identifications were considered as the gold standard. The true positives were considered when the same bacterial species were identified by culture/MALDI-TOF and the taxonomic classification method. The false positives were detected when bacterial species different from those identified by culture/MALDI-TOF, were identified by the taxonomic classification method. The false negatives were determined when the bacterial species identified by culture/MALDI-TOF were not identified by the taxonomic classification method.

2.4 Results

2.4.1 Classical identification

Nine body fluid samples and one tissue sample from 9 different patients were sequenced, including one sample from peritoneal fluid, five from pus (3 abscesses and 2 empyemas), two from synovial fluid of knees with prosthesis, one from sputum and one from a bone biopsy (Table 2.1). In total 15 different isolates obtained from the 10 samples were considered of possible clinical significance and were selected for species identification and antimicrobial susceptibility testing during routine work up of the samples (Table 2.2,2.3 and 2.4). In samples 2 and 3, only one colony-forming unit (CFU) of *Escherichia coli* and *Staphylococcus epidermidis*, respectively, was detected after 48 hours of incubation. In samples 2 and 5, the anaerobic cultures were mixed to such an extent, that no further characterization of the colonies was performed, and the results were reported as anaerobic mixed culture.

Antimicrobial susceptibility testing, revealed three isolates to be fully susceptible, while the others were resistant to at least one antimicrobial. Two isolates, one *Staphylococcus haemolyticus* and one *S. epidermidis* were oxacillin-resistant and positive in the cefoxitin test (Vitek 2).

There was fungal growth in 2 samples (1 and 5) that included two Candida species (one *Candida glabrata* and one *Candida albicans*). The different bacterial and fungal species identified in each sample are shown in Tables 2.2, 2.3 and 2.4.

Table 2.2: Microorganisms identified by conventional methods, WGS and using shotgun metagenomics and the taxonomic classification methods in Unix.

Sample number	Culture result (CFU)[a]	Conventional identification (MALDI-TOF)	WGS-based identification	Shotgun metagenomics		
				Kraken[b]	MIDAS[c]	MetaPhlAn[c]
1	10^3 10^3 10	*E. faecium* *S. haemolyticus* *C. glabrata*	*E. faecium* *S. haemolyticus* -	*E. faecium* (34.6%) *S. haemolyticus* (10.1%) -	*E. faecium* (62.0%) *S. haemolyticus* (28.0%) -	*E. faecium* (66.6%) *S. haemolyticys* (27.7%) -
2	10^3 1 Not determined	*E. avium* *E. coli* Anaerobes	-# -# -#	Not identified* Not identified* Several species (29.5%)	Not identified* Not identified* Several species (100.0%)	Not identified* Not identified* Several species (100.0%)
3	1	*S. epidermidis*	-#	S. aureus (0.2%)	Not identified*	Not identified*
4	10^3	*S. aureus*	*S. aureus*	*S. aureus* (0.73%)	*S. aureus* (100%)	*S. aureus* (100%)
5	$\geq 10^5$ $\geq 10^5$ 10^3 10^3 Not determined 10	*E. coli* *K. oxytoca* *S. anginosus* *E. faecalis* Anaerobes *C. albicans*	*E. coli* *K. oxytoca* -# *E. faecalis* -# -#	*E. coli* (9.7%) *K. oxytoca* (0.5%) *S. anginosus* (0.07%) *E. faecalis* (0.3%) Several species (12.7%) -	*E. coli* (6.5%) *K. oxytoca* (0.3%) *S. anginosus* (0.01%) *E. faecalis* (0.9%) Several species (96.7%) -	*E. coli* (8.5%) *K. oxytoca* (0.3%) *Streptococcus spp.* (0.09%) *E. faecalis* (0.7%) Several species (90.4%) -
6	10^3	*E. faecium*	*E. faecium*	*E. faecium* (0.77%)	Not identified*	Not identified*
7	10^2	*S. aureus*	-#	*S. aureus* (82.9%)	*S. aureus* (100%)	*S. aureus* (100%)
8	10^3	*O. intermedium*	*O. intermedium*	*O. anthropi* (21.3%)	*O. intermedium* (99.4%)	*O. intermedium* (99.1%)
9	10^3	*S. aureus*	*S. aureus*	*S. aureus* (22.9%)	*S. aureus* (100%)	*S. aureus* (100%)
10	10^3	*S. marcescens*	-#	*S. marcescens* (64.7%)	*S. marcescens* (99.1%)	*S. marcescens* (100%)

[a] The number of colonies of a given species was estimated from the number of colonies with the same morphology on the same plate

[b] The relative abundance is calculated using total number of reads as denominator

[c] The relative abundance is calculated with the total number of classified reads as denominator

[d] miniKraken database was used

Although there was a laboratory identification, no isolates were available for WGS

* No reads matched that specific pathogen, not even at the genus level

Table 2.3: Microorganisms identified by conventional methods, WGS and using shotgun metagenomics and the taxonomic classification methods in CLC Genomics Workbench.

Sample number	Culture result (CFU)^{a}	Conventional identification (MALDI-TOF)	WGS-based identification	Shotgun metagenomics	
				Taxonomic Profiling (CLC)^{b}	Best match with K-mer spectra (CLC)^{c}
1	103	*E. faecium*	*E. faecium*	*E. faecium* (71%)	*E. faecium* (41.4%)
	10^3	*S. haemolyticus*	*S. haemolyticus*	*S. haemolyticus* (24%)	*S. haemolyticus* (13.8%)
	10	*C. glabrata*	-	*C. glabrata* (100%)	*C. glabrata* (0.5%)
2	10^3	*E. avium*	-$^{\#}$	Not identified*	Not identified*
	1	*E. coli*	-$^{\#}$	Not identified*	Not identified*
	Not determined	Anaerobes	-$^{\#}$	Several species (97%)	Several species (13.2%)
3	1	*S. epidermidis*	-#	Not identified*	*S. aureus* (4%)
4	10^3	*S. aureus*	*S. aureus*	Not identified*	*S. aureus* (9.7%)
5	$\geq 10^5$	*E. coli*	*E. coli*	*E. coli* (25%)	*E. coli* (11.5%)
	$\geq 10^5$	*K. oxytoca*	*K. oxytoca*	*K. michiganensis* (0.3%)	Not identified*
	10^3	*S. anginosus*	-$^{\#}$	Not identified*	Not identified*
	10^3	*E. faecalis*	E. faecalis	*E. faecalis* (2%)	*E. faecalis* (0.6%)
	Not determined	Anaerobes	-$^{\#}$	Several species (70.0%)	Not identified*
	10	*C. albicans*	-$^{\#}$	Not identified*	*C. albicans* (<0.05%)
6	10^3	*E. faecium*	*E. faecium*	Not identified*	*E. faecium* (4.0%)
7	10^2	*S. aureus*	-$^{\#}$	*S. aureus* (100%)	*S. aureus* (95.5%)
8	10^3	*O. intermedium*	*O. intermedium*	*O. intermedium* (86.0%)	*O. intermedium* (91.2%)
9	10^3	*S. aureus*	*S. aureus*	*S. aureus* (100%)	*S. aureus* (81.2%)
10	10^3	*S. marcescens*	-$^{\#}$	*S. marscescens* (100%)	*S. marcescens* (79.7%)

[a] The number of colonies of a given species was estimated from the number of colonies with the same morphology on the same plate

[b] The relative abundance is calculated using total number of reads as denominator

[c] The relative abundance is calculated with the total number of classified reads as denominator

[#] Although there was a laboratory identification, no isolates were available for WGS

[*] No reads matched that specific pathogen, not even at the genus level

Table 2.4: Microorganisms identified by conventional methods, WGS and using shotgun metagenomics and the taxonomic classification methods in webpages (BaseSpace, Taxonomer and CosmosID).

Sample number	**Culture result (CFU)**[a]	**Conventional identification (MALDI-TOF)**	**WGS-based identification**	**Shotgun metagenomics**				
				Genius (Basespace)[c]	**Kraken (Basespace)**[c,d]	**MetaPhlAn (Basespace)**[c]	**Taxonomer (Utah)**[b,e]	**Cosmos ID**[a]
1	10^3 10^3 10	*E. faecium* *S. haemolyticus* *C. glabrata*	*E. faecium* *S. haemolyticus* -	*E. faecium* (14.4%) *S. haemolyticus* (55.8%) -	*E. faecium* (25.0%) *S. haemolyticus* (20.1%) -	*E. faecium* (65.1%) *S. haemolyticys* (30.4%) -	*E. faecium* (22.9%) *S. haemolyticus* (20.1%) Not identified*	*E. faecium* (50.3%) *S. haemolyticus* (22.1%) *C. glabrata* (88.6%)
2	103 1 Not determined	*E. avium* *E. coli* Anaerobes	-# -# -#	Not identified* Not identified* Several species (94.0%)	Not identified* Not identified* Several species (27.0%)	Not identified* Not identified* Several species (54.2%)	Not identified* Not identified* Several species (14.2%)	Not identified* Not identified* Several species (100%)
3	1	*S. epidermidis*	-#	*S. aureus* (100%)	*S. aureus* (0.1%)	Not identified*	*S. pseudintermedius* (3.4%)	Not identified*
4	10^3	*S. aureus*	*S. aureus*	*S. aureus* (100%)	*S. aureus* (0.3%)	*S. aureus* (100%)	*S. aureus* (8.3%)	*S. aureus* (100%)
5	$\geq 10^5$ $\geq 10^5$ 10^3 10^3 Not determined 10	*E. coli* *K. oxytoca* *S. anginosus* *E. faecalis* Anaerobes *C. albicans*	*E. coli* *K. oxytoca* -# *E. faecalis* -# -#	*E. coli* (0.4%) Not identified* *S. anginosus* (0.03%) *E. faecalis* (0.8%) Several species (45.0%) -	*E. coli* (10.2%) *K. oxytoca* (0.5%) *S. anginosus* (0.4%) *E. faecalis* (0.3%) Several species (8.0%) -	*E. coli* (7.0%) *K. pneumoniae* (0.01%) *S. anginosus* (0.3%) *E. faecalis* (0.7%) Several species (89.1%) -	*E. coli* (3.6%) *K. michiganensis* (0.1%) *S. anginosus* (0.1%) *E. faecalis* (0.1%) Several species (60.3%) -	*E. coli* (7.6%) *K. oxytoca* (1.7%) *S. anginosus* (0.09%) *E. faecalis* (3.7%) Several species (86.2%) Not identified*
6	10^3	*E. faecium*	*E. faecium*	*E. faecium* (4.2%)	*E. faecium* (14.8%)	*E. faecium* (5.5%)	*E. faecium* (1.4%)	*E. faecium* (4.1%)
7	10^2	*S. aureus*	-#	*S. aureus* (100%)	*S. aureus* (93.8%)	*S. aureus* (100%)	*S. aureus* (14.2%)	*S. aureus* (100%)
8	10^3	*O. intermedium*	*O. intermedium*	*O. intermedium* (100%)	*O. nthropic* (88.9%)	*O. intermedium* (99.8%)	*O. intermedium* (13.1%)	*O. intermedium* (49.5%)
9	10^3	*S. aureus*	*S. aureus*	*S. aureus* (100%)	*S. aureus* (99.5%)	*S. aureus* (100%)	*S. aureus* (12.7%)	*S. aureus* (100%)
10	10^3	*S. marcescens*	-#	*S. marcescens* (32.5%)	*S. marcescens* (94.8%)	*Serratia* spp. (100%)	*S. marcescens* (1.4%)	S. *marscescens* (38.4%)

[a] The number of colonies of a given species was estimated from the number of colonies with the same morphology on the same plate

[b] The relative abundance is calculated using total number of reads as denominator

[c] The relative abundance is calculated with the total number of classified reads as denominator

[d] miniKraken database was used [e] Full Analysis mode was used

Although there was a laboratory identification, no isolates were available for WGS

* No reads matched that specific pathogen, not even at the genus level

2.4.2 Comparison of standard procedures and shotgun metagenomics for the identification of clinically relevant pathogens

The tools used for taxonomic classification are shown in Figure 2.1. The total number of reads and the total number of reads mapped against the human genome (hg19) varied between samples, ranging from 3.5% to 98.9% (Table 2.1). The abundance of human reads was not determined by the type of sample but was probably influenced by individual characteristics of each sample and the success of the methods used in depleting the human DNA. We identified the microorganisms present using different taxonomical methods, including three Unix-based tools (Kraken, Metaphlan2 and MIDAS), web-based tools including both commercial and freely available solutions (BaseSpace, Taxonomer and CosmosID) and one commercial approach having a graphical interface (CLC Genomics Workbench v10.0.1). The taxonomic classification results for each sample are presented in Tables 2.2, 2.3 and 2.4. In 8 samples, all the microorganisms identified by classical culture were also identified through metagenomics (using at least one method). In sample 2, two of the bacterial species identified by classical culture, i.e., *E. coli* and one *Enterococcus avium* were not identified through shotgun metagenomics and in sample 3 there was no concordance between the results of MALDI-TOF and the taxonomical classification methods at the species level (Tables 2.2, 2.3 and 2.4). We identified *Ochrobactrum intermedium* in the negative control, but in low amounts (1.0% of the reads mapped to the reference genome with the accession number NZ_ACQA01000002 and only 1.4% of the reference genome was covered). The sensitivity and positive predictive value of each classification method is shown in Table 2.5.

Table 2.5: Performance of the different taxonomic classification methods for each sample. Sensitivity and positive predictive value were calculated using culture/MALDI-TOF as standards

Method	**Total number of bacteria identified**[a]	**True positives**[a]	**False positives**	**False negatives**	**Sensitivity (%)**	**PPV (%)**
Culture/MALDI-TOF	9	9	0	0	1	1
MetaPhlAn (BaseSpace)	16	7	9	2	0.78	0.44
Genius (BaseSpace)	35	8	27	1	0.89	0.23
Kraken (BaseSpace)	959	7	952	2	0.78	0.01
Taxonomer (Full Analysis)	4649	8	4641	1	0.89	0
CosmosID	35	8	27	1	0.89	0.23
Taxonomic Profiling (CLC Genomics Workbench v10.0.1)	17	6	11	3	0.67	0.35
Best match K-mer spectra (CLC Genomics Workbench v10.0.1)	12	8	4	1	0.89	0.67
Kraken (Unix)	198	7	191	2	0.78	0.04
MetaPhlAn2 (Unix)	15	7	6	4	0.75	0.75
MIDAS (Unix)	34	7	26	2	0.88	0.5

[a] Excluding the samples with non-identified anaerobic bacteria (Samples 2 and 5)

Abbreviations: PPV – positive predictive value

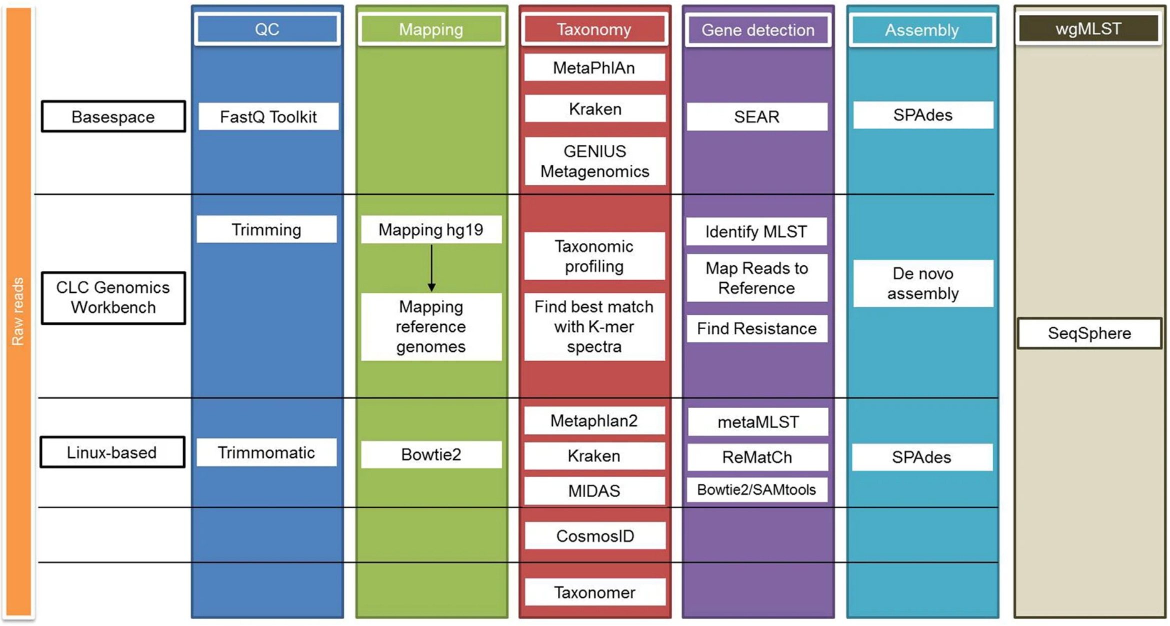


Figure 2.1: Scheme of the bioinformatic analysis of the metagenomics samples.

2.4.3 Determination of antimicrobial resistance

Metagenomics provides other sequence information in addition to pathogen detection. We determined the presence of antimicrobial-resistance genes in the SMg sequence data and compared the results with those obtained from WGS and phenotypic resistance testing (Table 2.6).

AMR genes found with CLC Genomics Workbench and ReMatCh in samples 1, 7 and 9 correlated well with phenotypic results. However, in the other 7 samples, not all antimicrobial resistance genes that could explain the phenotypic profile were identified. In addition, in samples 2, 5, 7 and 10, ReMatCh detected different resistance genes compared to those reported by CLC Genomics Workbench (Table 2.6). Some of these differences (genes *norA*, *blaSST*-1, *fusA*) were due to slight differences in the databases used, however, the other resistance genes were present in both databases. Interestingly, in two samples (samples 2 and 5), we were able to identify several antimicrobial resistance genes usually found in anaerobic bacteria. These were not reported by classical microbiology methods, probably because they were not considered relevant pathogens worthy of subsequent susceptibility study (mixed anaerobic culture).

Table 2.6: Antimicrobial resistance phenotypes and antimicrobial resistance genes detected using different approaches.

Sample number	**Conventional identification (MALDI-TOF)**	**Conventional susceptibility testing (VITEK 2)**[b]	**WGS**	**Shotgun metagenomics**	
			CLC Genomics Workbench	**ReMatCh (Unix)**	**CLC Genomics Workbencha**
1	E. faecium S. haemolyticus	LEV, ERY, CLI OXA, GEN, CIP, FOS, ERY, CLI	erm(B), msr(C), ant(6’)-Ia, aph(3’)-III, dfrG blaZ, mecA, ant(6’)-Ia, aph(3’)-III, aac(6’)-aph(2”), erm(C), mph(C), msr(A), dfrG	erm(B), msr(C), ant(6’)-Ia, aph(3’)-III, aac(6’)-aph(2”), blaZ, mecA, erm(C), mph(C), msr(A), dfrG	erm(B), msr(C), ant(6’)-Ia, aph(3’)-III, aac(6’)-aph(2”), blaZ, mecA, erm(C), mph(C), msr(A), dfrG
2	E. avium E. coli Anaerobes	DOX, CLI susceptible -	-# -# -#	Not detected Not detected catS, lnu(D), lsa(C), cepA-44, tet(Q)	Not detected Not detected catS, lnu(D), lsa(C), cepA-44, tet(Q), fusA
3	S. epidermidis	OXA, GEN, TEC, FUS, CIP, ERY, CLI	-#	Not detected	Not detected
4	S. aureus	PEN, ERY	blaZ, spc, erm(A)	Not detected	Not detected
5	E. coli K. oxytoca S. anginosus E. faecalis Anaerobes	susceptible AMX susceptible DOX, CLI -	-# blaOXY-1-3 -# tet(M), lsa(A) -#	- Not detected - tet(M) cfxA4, tet(Q)	- Not detected - tet(O) cfxA4, tet(Q)
6	E. faecium	PEN, AMX, CFX, IMP, GENhl, STRhl, LEV, ERY, CLI, AMP/SUL	erm(B), msr(C), ant(6’)-Ia, aph(3’)-III, aac(6’)-aph(2”), dfrG	Not detected	Not detected
7	S. aureus	PEN	blaZ	blaZ, norA	blaZ
8	O. intermedium	AMX, PIP/TAZ, CFX, CFT, CTZ, IMP, FOX, TOB, FOS, NIT, TMP	blaOCH-2	blaOCH-5	blaOCH-2
9	S. aureus	PEN	-#	blaZ	blaZ
10	S. marcescens	AMX, AMC, CFX, FOX, NIT, POL	-#	blaSST-1, tet(41), oqxB, aac(6’)-Ic	tet(41), oqxB, aac(6’)-Ic

[a] The analysis aborted when the script tried to connect to NCBI

[b] Only non-susceptibility is indicated.

Abbreviations: AMP/SUL, ampicillin/sulbactam; AMX, amoxicillin; AMC, amoxicillin/clavulanate; CFX, cefuroxime; FOS, fosfomycin; FOX, cefoxitin; CIP, ciprofloxacin; CLI, clindamycin; DOX, doxycycline; ERY, erythromycin; FUS, fusidic acid; GEN, gentamicin; GENhl, gentamicin high-level; LEV, levofloxacin; NIT, nitrofurantoin; PEN, penicillin; POL, polymyxin B; STRhl, streptomycin high-level; TEC, teicoplanin.

Table 2.7: Results of MLST using by whole genome sequencing and shotgun metagenomics

Sample number	**Conventional identification (MALDI-TOF)**	**WGS**	**Shotgun metagenomics**	
		CLC Genomics Workbench v10.1.1	**CLC Genomics Workbench v10.1.1**	**metaMLST (Unix-based)**
1	*E. faecium* *S. haemolyticus*	ST117 ST25	Not detected (6 alleles identified correctly) Not detected (3 alleles identified correctly)	ST117 Not detected
2	*E. avium* *E. coli* Anaerobes	-$^{\#}$ -$^{\#}$ -$^{\#}$	- Not detected -	- Not detected -
3	*S. epidermidis*	-$^{\#}$	Not detected	Not detected
4	*S. aureus*	ST30	Not detected	Not detected
5	*E. coli* *K. oxytoca* *S. anginosus* *E. faecalis* Anaerobes	ST141 ST40 -$^{\#}$ ST179 -$^{\#}$	ST141 Not detected - Not detected -	ST4508 Not detected - Not detected -$^{\#}$
6	*E. faecium*	ST117	Not detected	Not detected
7	*S. aureus*	ST30	ST30	ST667
8	*O. intermedium*	-	-	-
9	*S. aureus*	-$^{\#}$	Not detected	Not detected
10	*S. marcescens*	-$^{\#}$	-	-

Abbreviations: ST, sequence type

The SEAR app in BaseSpace (the only one available for antimicrobial resistance gene detection) crashed several times, although we performed the analysis repeatedly, using different parameters. We were only able to get results in 3 samples, with no resistance genes detected.

2.4.4 MLST and wgMLST analysis

In three cases when SMg data covered $\geq 93\%$ of the genome we were able to identify the ST, which corresponded to the one found using WGS of the isolated bacteria using CLC Genomics Workbench (n=2) and metaMLST (n=1). These results are summarized in Table 2.7. Assembled genomes and metagenomes, were compared by wgMLST analysis using Ridom SeqSphere+. Figure 2.2 shows examples of the allele difference between the genomes obtained through WGS versus the genomes obtained through shotgun metagenomics.

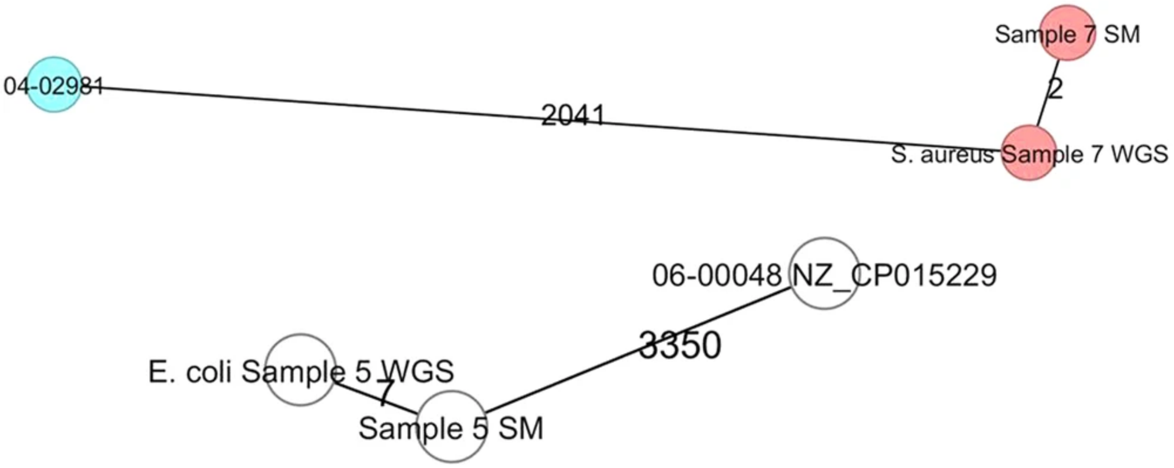


Figure 2.2: Minimum-spanning tree based on wgMLST allelic profiles of 2 S. aureus genomes and 2 E. coli genomes obtained through SM and WGS in comparison to reference strains 04-02981 (GenBank accession number NC_017340) and 06-00048 (NZ_CP015229), respectively. Each circle represents an allelic profile based on sequence analysis. The numbers on the connecting lines illustrate the numbers of target genes with differing alleles.

2.4.5 Characterisation of mobile genetic elements

Two different approaches, i.e. CLC Genomics Workbench and Bowtie2 were used to identify plasmids present in the sequence data. Both approaches used mapping of sequences against the same plasmid database. Since some plasmids present in the database are very similar and sequence reads may be mapped to more than one plasmid, we used the pATLAS tool, which provides an overview of the nodes (representing plasmid sequences) and links between plasmids (which connect similar plasmids), to enable the visualisation of the plasmids identified (Figure 2.3). A colour gradient indicates the sequence coverage of the plasmids. In most cases, the same plasmids were identified by both approaches, with some small differences in sequence coverage. When comparing the plasmids identified in the SMg dataset versus the WGS data, most of the plasmids were also detected in the isolates (an example is shown in Figure 2.4). However, some plasmids were not identified in any of the isolated bacteria and were probably residing in low-abundant species.

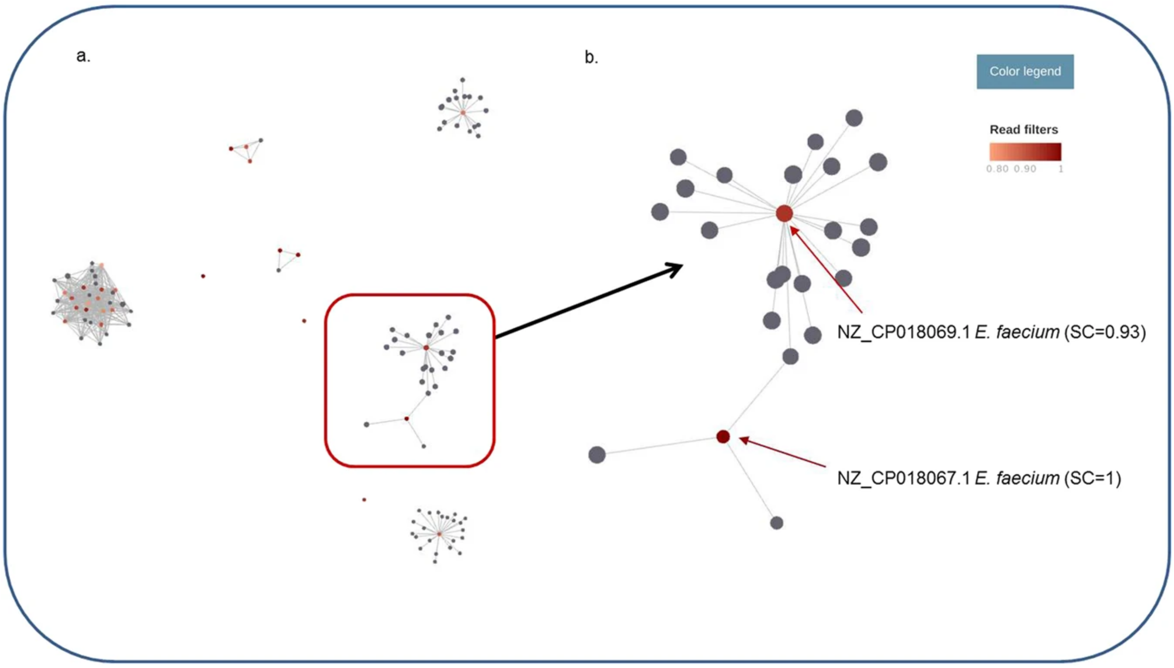


Figure 2.3: (a) Overview of the nodes (representing plasmid sequences) and links between plasmids (connecting similar plasmids) found in Sample 1 (SMg) using the pATLAS tool. (b) A closer look at one of the cloud of plasmids. The colour gradient in each cloud of plasmids represents the plasmid sequence coverage (SC), varying between 0-0.79 (grey) and 0.80-1 (red gradient).

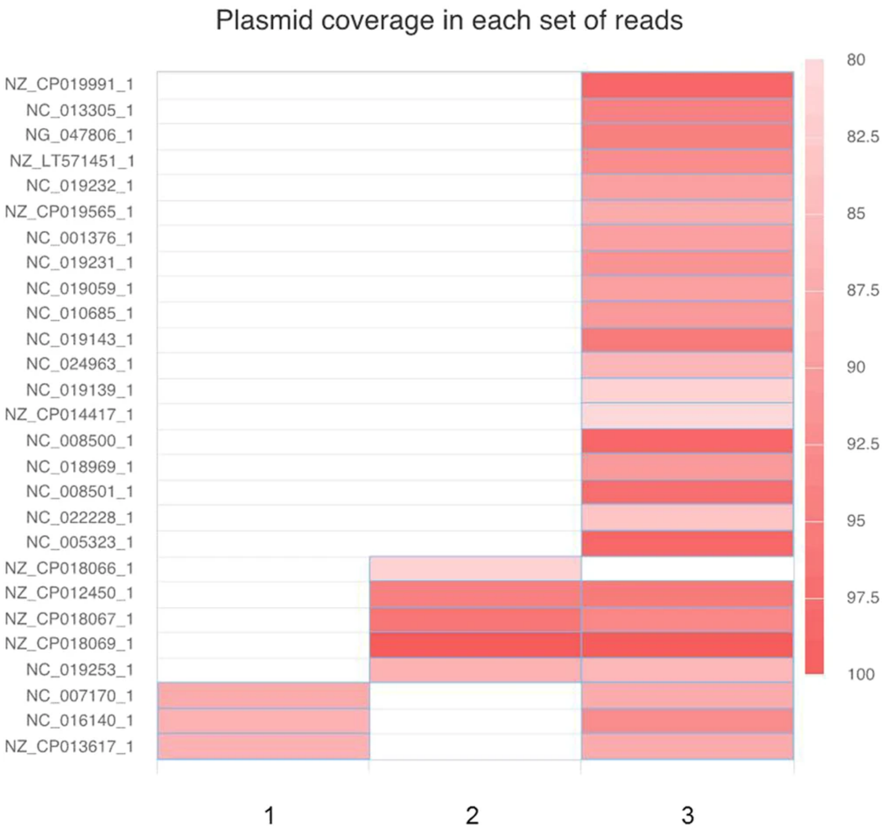


Figure 2.4: A heatmap comparing the identified plasmids using bowtie2 in *S. haemolyticus* WGS (1), *E. faecium* WGS (2) and in the SMg dataset (3) isolated from sample 1.

2.5 Discussion

This study evaluated the suitability of SMg for the microbiological diagnosis and (patho- and epi-) typing of microorganisms directly from real patient samples. The whole procedure took between 48-54 hours to complete, which is shorter than culture-based methods if one includes typing. However, the amount of information derived from SMg in most cases, did not overcome the necessity for pathogen isolation and subsequent (phenotypic and genotypic) typing, which can take up to 1-2 weeks (particularly in slow-growing organisms). Nevertheless, SMg can help guide antimicrobial therapy and be helpful in cases where there is a suspicion of transmission and there is a need to quickly determine the genetic relationship between pathogens, although the success of SMg in individual patient samples can be highly variable, as reported here.

Different bioinformatics pipelines were evaluated to identify potential differences between them and identify those which could provide the clinical microbiologist with the maximum of relevant and accurate information. In terms of microbial identification, in both Unix and web-based approaches we would recommend MetaPhlAn, since it has good sensitivity and a good positive predictive value (PPV). The find best match K-mer spectra tool should be used in the context of the CLC Genomics Workbench, since it had a higher sensitivity and PPV compared to the Taxonomic Profiling tool.

In a clinical setting, a combination of high sensitivity and high PPV of any new method is key. Popular software designed for bacterial identification, can predict dozens to hundreds of species in in vitro generated bacterial communities of known composition [8]. We observed the same when using Kraken and Taxonomer when comparing to culture-based methods. For both Kraken and Taxonomer, relative abundance cut-off values may be required to limit the number of species identified. However, which cut-off values should be used are a matter of debate, since in some cases, even if applying a cut-off value as low as 1.0% (comparable to what was found in the negative control) would have resulted in decreased sensitivity (e.g. the *Streptococcus anginosus* identified by culture in Sample 5 would have been disregarded). The methods that employ several parameters to infer microbial identification are superior, because they not only rely on the relative abundance of bacterial species, but also on the genome coverage and on the proportion of the genome that was covered. On the other hand, in some cases SMg may be more sensitive than culture in identifying pathogens, reflecting the higher sensitivity or the capacity to detect bacterial species which are non-culturable in the conditions used or that are no longer culturable, such as due to prior antimicrobial therapy. In such cases, other methods like 16S rDNA sequencing or the recently described 16S-23S rDNA sequencing method [21] may be used for discrepancy analyses. However, here we decided to use culture-based methods as the gold standard, since this is still the method of choice in clinical microbiology.

One limitation of this study was the exclusion of culture-negative samples and thus their inclusion would have affected the calculation of the specificity values. However, as mentioned above, culture-negative samples do not necessarily mean that the samples are pathogen-free, but it might only reflect the low sensitivity or capacity of culture-based methods to detect non-culturable bacterial species. As with other (molecular) methods, several controls should be included to validate the obtained results, including a negative control. In our negative control, we detected an *O. intermedium* strain, although with only 1.0% of the reads mapping to the reference genome and covering only 1.4% of the reference genome (accession number NZ_ACQA01000002). These results may be due to contamination during library preparation (e.g. sample-to-sample contamination prior to indexing), the result of sequencing artefacts (e.g. demultiplexing errors), or to incorrect classification during data analysis (e.g. highly similar regions) [3]. Our samples and sequencing libraries were handled in laminar flow cabinets; however, we cannot also exclude the possibility of contamination. Furthermore, the reagents used may also be or become contaminated with DNA leading the detection of these contaminating species, something that has been described previously [7]. This poses a challenge for interpretation, because some positive samples also had very low numbers of reads for some pathogens (< 1%). When approaching this limit of detection, small numbers of pathogen reads will be difficult to interpret, as they can represent true-positives with low abundance in the sample, or artefacts such as contamination during library preparation[3].

In terms of antimicrobial resistance gene detection, ReMatCh (Unix) and the CLC Genomics Workbench Find Resistance tool gave comparable results. Since ReMatCh (Unix) performs the analysis at the read level, while CLC Genomics Workbench performs it at the contig level, we suggest that both strategies should be employed in parallel when looking for antimicrobial resistance genes. It is also important to emphasise that the contig-level approach employed by CLC Genomics Workbench may give negative results if the sequence coverage is set to a high percentage (e.g. above 80%). This is due to the assembly method, which may split the antimicrobial resistance genes into different contigs, when the number of reads is too low. This phenomenon was observed in Sample 1, for the *aac(6')-aph(2")* gene, which was split into 3 different contigs, each part corresponding to less than 40% of the gene. Only when applying a cut-off value of *geq* 20% for sequence coverage could we identify all three parts of the gene, which in total corresponded to 89% of the entire sequence. Finally, it is important to point out that the ResFinder database (used here), and other databases, focus on acquired genes, not including chromosomal point mutations resulting in antimicrobial resistance. However, a recently developed tool, PointFinder, was added to ResFinder for the detection of chromosomal point mutations associated with antimicrobial resistance [22] and an updated database will be available soon.

Another challenge is to infer where these antimicrobial resistance genes are located (chromosome or plasmid). The study of mobile genetic elements, including plasmids, carrying antimicrobial resistance genes present in clinical samples is important to predict possible treatment failures and the spread of resistance within and across bacterial species. When performing bacterial isolation followed by WGS, information on polymicrobial infections may be lost. This is mainly driven by a bottleneck in culture, where some bacterial species are not isolated with standard work up protocols (frequently anaerobes and slow-growing organisms). The presence of antimicrobial resistance genes in plasmids of bacteria other than those isolated through culture poses a risk since they are not identified by conventional methods but could potentially be horizontally transmitted to pathogenic bacteria under the antimicrobial selective pressure of treatment. Antimicrobial administration may also select minority populations where these resistance determinants are found. Furthermore, the understanding of how plasmids are shared by different bacteria in a bacterial community (e.g. within an infection site or in the gut) can improve our understanding of how these elements disseminate across species and from patient to patient11. The SMg approach is clearly more efficient than culture in identifying the "cloud" of plasmids present in a given sample (Figure 2.4) and which can be potentially transferred to more pathogenic species generating problems of resistance, as was the case with the emerge of vancomycin resistance *S. aureus* [23].

Whole-genome sequencing has been used extensively for several purposes [24] and is considered to have the potential of playing an important role in clinical microbiology [25]. It is the ongoing goal of medical molecular microbiology to develop faster typing methods that can be used for outbreak surveillance. For this purpose, we assembled the metagenomics data and compared it with the assemblies given by WGS. Surprisingly, the assemblies provided by SPAdes in BaseSpace were closer to the assemblies provided by WGS. When comparing the genomes obtained through WGS and SMg, we could see that in 4 out of 8 bacterial isolates the number of different alleles was *leq* 7. This showed the potential of SMg to draw phylogenetic relationships from uncultured bacterial genomes, although more potentially limited than those obtained using WGS data from axenic cultures. As for the detection of resistance genes, a key limiting factor may be the number of bacterial reads, reflected in a lower genome coverage (e.g. samples 4 and 6). In these cases, we would have to either improve the human-DNA depletion step, improve the microbial enrichment or perform sequencing at a higher sequencing depth to have enough microbial reads to be able to get a more appropriate genome coverage. Yet, this last step will severely raise the sequencing costs, which might render the methodology unfeasible for routine application.

In this study, we evaluated the results of metagenomics pipelines using three different methods. CLC Genomics Workbench has advantages over the other methods. It does not require previous knowledge of Unix-based tools, it is arguably the most user-friendly and delivered reliable results for microbial identification and antimicrobial resistance gene detection. The downside was the assembly approaches, which provided lower wgMLST allele detection, when compared to the assemblies using SPAdes (BaseSpace and Unix). BaseSpace, the other commercial solution, on the other hand, provided only a few tools that can be used for metagenomics data. Furthermore, since Illumina did not develop the apps themselves, they offered no direct support. Contacting the developers (via email and posting on their forum) does not guarantee a solution to the issues in a time frame compatible with a routine clinical microbiology laboratory work. The dependence and no direct control over a third party to resolve software bugs and provide a stable platform illustrates a disadvantage of a cloud-based system like BaseSpace. Finally, the Unix-based pipeline complemented the data on antimicrobial resistance genes but did not offer better results in terms of microbial identification and MLST typing. However, many more freely available tools for this last purpose could have been used, potentially improving on the results obtained. Reference-guided assembly approaches, taking advantage of the species information derived in the first steps of our analysis pipelines, will deserve further study in the future since these may provide higher quality assemblies from metagenomics data. The main advantage of an open-source approach is its flexibility since it allows the user to choose the most adequate method for each desired outcome. There were several limitations to this study. First, the number of samples included was low and some of the bacterial isolates were not available for further WGS analysis. However, the extended data analyses performed in each sample limited the number of samples to be included. It is our intention to move forward with the most adequate pipelines for each purpose and apply them to additional patients' samples. Second, the samples differed greatly from each other. However, in our point of view, this was beneficial to the study, since it did not bias the analyses as it could have happened if only one type of sample had been used. Finally, we used three different extraction methods that could have influenced the final results. Yet, as can be seen in Table 2.1, the number of human reads differed between samples, even when using the same extraction kit. This suggests none of the kits is clearly superior to the others and that the ratio between host and microbial DNA or other individual sample characteristics will be the major determinants of the proportion of microbial reads.

In conclusion, this study showed the potential but also highlighted the problems of implementing shotgun metagenomics for the identification and typing of pathogens directly from clinical samples. Based on the results obtained here we can conclude that the tools and databases used for taxonomic classification and antimicrobial resistance will have a key impact on the results, cautioning about the comparison between studies using different methods and suggesting that efforts need to be directed towards standardisation of the analysis methods if SMg is to be used routinely in clinical microbiology.

2.6 Acknowledgements

We thank Peter Posma, Yvette Bisselink and Brigitte Dijkhuizen for excellent technical assistance. We thank Dr. Michael Lustig and colleagues from Molzym Life Science for helping with extraction protocols.

This project has received funding from the European Union’s Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement 713660.This work was partly supported by the INTERREG VA (202085) funded project EurHealth-1Health, part of a Dutch-German cross-border network supported by the European Commission, the Dutch Ministry of Health, Welfare and Sport (VWS), the Ministry of Economy, Innovation, Digitalisation and Energy of the German Federal State of North Rhine-Westphalia and the German Federal State of Lower Saxony.

2.7 Author contributions statement

N.C., J.A.C., M.R., S.P., I.A., A.W.F. and J.W.A conceived the experiment(s), N.C., L.S. and E.C.R. conducted the experiment(s), N.C., L.S., M.M., C.I.M., T.F.J., S.R., M.C., J.A.C. and M.R. analysed the results, N.C. and L.S. wrote the manuscript. All authors reviewed the manuscript.

2.8 Additional information

2.8.1 Accession codes

The paired-trimmed-un-mapped reads (hg19) generated for each sample have been submitted to SRA under project number SRP126380. The cgMLST schemes are deposited in figshare under the DOI:10.6084/m9.figshare.5679376

2.8.2 Competing financial interests

The authors declare that they have no conflict of interest.

2.9 Supplemental Material

Table 2.8: Supplementary table 1.

FastQ Toolkit v2.2.0	
Minimum read length	32
Sub-sampling	FALSE
Adapter trim stringency	0.9
Select respective adapters	TRUE
Quality trimming	FALSE
Poly-A/T Trimming	FALSE
Read Filtering	FALSE
Modify Reads	FALSE
Fix Format	FALSE
FastQC v1.0.0	
Kmer Size	5
Use Conatminant Filter	TRUE
Kraken Metagenomics v1.0.0	
Host Filter	TRUE RefSeqhg19
Classification Database	MiniKraken 20141208 (latest)
Filter Threshold	0
Metaphlan v1.0.0	
Sensitivity options for read-marker similarity (as descibed by BowTie2)	Very Sensitive
SPAdes Genome Assembler v3.9.0	
Running Mode	Error Correction & Assembly
Dataset type	Multi Cell
Careful Mode	Disable
k-mer lengths	Auto
SEAR: Antibiotic Resistance v1.0.0	
Read length cutoff (bases)	70
Read quality score cutoff	20
Read subtraction against E.coli reference genome (K12)?	No
Clustering stringency (express % as a decimal)	0.98
Annotation stringency (% length of reference ARG sequence mapped to by sequencing reads	40
GENIUS Metagenomics: Know Now v1.1.0	
Can't set any settings in BaseSpace	

Table 2.9: Supplementary table 2.

Trim Reads	
Trimmomatic v0.36 (INNUca v2.6 initial module)	
Quality trim	TRUE
Phred Quality limit	05:20
Trim adapter list	Illumina adapters
Remove 5’ terminal nucleotides	TRUE
Number of 5’ terminal nucleotides	3
Remove 3’ terminal nucleotides	TRUE
Number of 3’ terminal nucleotides	3
Discard short reads	TRUE
Minimum number of nucleotides in reads	55
Map Reads to Reference	
Bowtie2 v2.3.2	
References	Homo sapiens (hg19) index
Mode	end-to-end
Mode option	sensitive
Collect unmapped reads	FALSE
Taxonomic Classification	
Kraken v0.10.5-beta	
References	miniKraken database (Dec. 8, 2014)
K-mer length	35
MIDAS	
References	midas_db_v1.2 (May 9, 2018)
Word size for blast	28
Alignment coverage	0.75
MetaPhlAn2 v2.0	
References	default database (May 9, 2018)
Minimum total nucleotide length for the markers	2000
Quantile value for robust average	0.1
Statistical approach for converting marker abundances into clade abundances	clade global
Analysis type	profiling a metagenome in terms of relative abundance
Identify MLST	
metaMLST v1.1	
References	metamlstDB_2017
Bowtie2 mode	local
Bowtie2 mode option	very sensitive local
Collect unmapped reads	FALSE
Search for and report all alignment	TRUE
Find Resistance Genes	
ReMatCh v3.2	
References	ResFinder database (29-06-2017)
Minimum coverage to consider a position as present	1
Minimum coverage depth to perform a basecall	1
Minimum gene coverage (%)	80
Minimum gene identity (%)	70
***De Novo* Assembly**	
SPAdes v3.10.1	
Mode	careful
Error correction	FALSE
Read coverage cut-off value	2
List of K-mers	21,33,55,67,77
Plasmid Detection	
Bowtie2 v2.3.2	
References	NCBI RefSeq (May 11, 2017)
Mode	end-to-end
Mode option	sensitive
Collect unmapped reads	FALSE
Multiple alignment	TRUE

Table 2.10: Supplementary table 3.

Illumina	
Discard sequence names	FALSE
Discard quality scores	FALSE
Selected files	
Paired-end reads	TRUE
Read Orientation	Forward Reverse
minimum distance	1
maximum distance	1000
Remove failed reads	TRUE
Quality score	NCBI/Sanger or Illumina Pipeline 1.8 and later
MiSeq de-multiplexing	FALSE
Illumina trim	FALSE
Trim Reads	
Quality trim	TRUE
Quality limit	0.05
Ambiguous trim	TRUE
Ambiguous limit	2
Trim adapter list	Illumina adapters
Use colorspace	FALSE
Remove 5’ terminal nucleotides	FALSE
Number of 5’ terminal nucleotides	1
Remove 3’ terminal nucleotides	FALSE
Number of 3’ terminal nucleotides	1
Discard short reads	TRUE
Minimum number of nucleotides in reads	30
Discard long reads	FALSE
Maximum number of nucleotides in reads	1000
Map Reads to Reference	
References	Homo sapiens (hg19) sequence
Masking mode	No masking
Masking track	
Match score	1
Mismatch cost	2
Cost of insertions and deletions	Linear gap cost
Insertion cost	3
Deletion cost	3
Insertion open cost	6
Insertion extend cost	1
Deletion open cost	6
Deletion extend cost	1
Length fraction	0.5
Similarity fraction	0.8
Global alignment	FALSE
Color space alignment	TRUE
Color error cost	3
Auto-detect paired distances	TRUE
Non-specific match handling	Map randomly
Find Best Matches using K-mer Spectra	
References	NCBI references (2017-07-08)
K-mer length	16
Only index k-mers with prefix	ATGAC
Check for low quality and contamination	TRUE
Fraction of unmapped reads for quality check	0.1
***De Novo* Assembly**	
Mapping mode	Create simple contig sequences (fast)
Update contigs	TRUE
Mismatch cost	2

Table2.10- *Continued from previous page*

Insertion cost	3
Deletion cost	3
Colorspace error cost	3
Length fraction	0.5
Similarity fraction	0.8
Colorspace alignment	TRUE
Alignment mode	local
Match mode	random
Create list of un-mapped reads	FALSE
Automatic bubble size	TRUE
Bubble size	50
Automatic word size	TRUE
Word size	20
Minimum contig length	200
Guidance only reads	
Perform scaffolding	TRUE
Auto-detect paired distances	TRUE
Create report	TRUE
Find Resistance	
DB	Database for Find Resistance (2018-02-02)
Minimum identity %	70
Minimum length %	20
Filter overlaps	TRUE
Local Realignment	
Realign unaligned ends	TRUE
Multi-pass realignment	2
Guidance-variant track	
Maximum guidance-variant length	100
Force realignment to guidance-variants	FALSE
InDels and Structural Variants	
P-Value threshold	1.00E-04
Maximum number of mismatches	3
Ignore broken pairs	TRUE
Filter variants	FALSE
Minimum number of reads	2
Minimum relative consensus coverage	0
Minimum quality score	0
Restrict calling to target regions	
Local Realignment	
Realign unaligned ends	TRUE
Multi-pass realignment	2
Guidance-variant track	Defined by: InDels and Structural Variants (2)
Maximum guidance-variant length	100
Force realignment to guidance-variants	FALSE
Identify MLST Scheme from Genomes	
Schemes	PubMLST (04-03-2017)
Identify MLST	
Scheme	Defined by: Identify MLST Scheme from Genomes
Low coverage reported when below	

2.10 References

[1] Henrik Hasman et al. “Rapid Whole-Genome Sequencing for Detection and Characterization of Microorganisms Directly from Clinical Samples”. en. In: *Journal of Clinical Microbiology* 52.1 (Jan. 2014). Ed. by Y.-W. Tang, pp. 139–146. ISSN: 0095-1137, 1098-660X. DOI: `10.1128/JCM.02452-13`. URL: `https://journals.asm.org/doi/10.1128/JCM.02452-13` (visited on 03/18/2022).

[2] Matthias Willmann et al. “Antibiotic Selection Pressure Determination through Sequence-Based Metagenomics”. eng. In: *Antimicrobial Agents and Chemotherapy* 59.12 (Dec. 2015), pp. 7335–7345. ISSN: 1098-6596. DOI: `10.1128/AAC.01504-15`.

[3] Erin H. Graf et al. “Unbiased Detection of Respiratory Viruses by Use of RNA Sequencing-Based Metagenomics: a Systematic Comparison to a Commercial PCR Panel”. eng. In: *Journal of Clinical Microbiology* 54.4 (Apr. 2016), pp. 1000–1007. ISSN: 1098-660X. DOI: `10.1128/JCM.03060-15`.

[4] P. Gyarmati et al. “Metagenomic analysis of bloodstream infections in patients with acute leukemia and therapy-induced neutropenia”. eng. In: *Scientific Reports* 6 (Mar. 2016), p. 23532. ISSN: 2045-2322. DOI: `10.1038/srep23532`.

[5] Nathan D Olson et al. “Metagenomic assembly through the lens of validation: recent advances in assessing and improving the quality of genomes assembled from metagenomes”. In: *Briefings in Bioinformatics* 20.4 (Aug. 2017), pp. 1140–1150. ISSN: 1467-5463. DOI: `10.1093/bib/bbx098`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6781575/` (visited on 03/17/2022).

[6] Robert Schlaberg et al. “Validation of Metagenomic Next-Generation Sequencing Tests for Universal Pathogen Detection”. en. In: *Archives of Pathology & Laboratory Medicine* 141.6 (June 2017), pp. 776–786. ISSN: 0003-9985, 1543-2165. DOI: `10.5858/arpa.2016-0539-RA`. URL: `http://www.archivesofpathology.org/doi/10.5858/arpa.2016-0539-RA` (visited on 05/20/2022).

[7] Teresa L. Street et al. “Molecular Diagnosis of Orthopedic-Device-Related Infection Directly from Sonication Fluid by Metagenomic Sequencing”. en. In: *Journal of Clinical Microbiology* 55.8 (Aug. 2017). Ed. by Nathan A. Ledeboer, pp. 2334–2347. ISSN: 0095-1137, 1098-660X. DOI: `10.1128/JCM.00462-17`. URL: `https://journals.asm.org/doi/10.1128/JCM.00462-17` (visited on 03/18/2022).

[8] Michael A. Peabody et al. “Evaluation of shotgun metagenomics sequence classification methods using in silico and in vitro simulated communities”. en. In: *BMC Bioinformatics* 16.1 (Dec. 2015), p. 362. ISSN: 1471-2105. DOI: `10.1186/s12859-015-0788-5`. URL: `http://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-015-0788-5` (visited on 03/18/2022).

[9] Ben Langmead and Steven L Salzberg. “Fast gapped-read alignment with Bowtie 2”. en. In: *Nature Methods* 9.4 (Apr. 2012), pp. 357–359. ISSN: 1548-7091, 1548-7105. DOI: `10.1038/nmeth.1923`. URL: `http://www.nature.com/articles/nmeth.1923` (visited on 03/18/2022).

[10] Heng Li et al. “The Sequence Alignment/Map format and SAMtools”. eng. In: *Bioinformatics (Oxford, England)* 25.16 (Aug. 2009), pp. 2078–2079. ISSN: 1367-4811. DOI: `10.1093/bioinformatics/btp352`.

[11] Derrick E Wood and Steven L Salzberg. “Kraken: ultrafast metagenomic sequence classification using exact alignments”. en. In: *Genome Biology* 15.3 (2014), R46. ISSN: 1465-6906. DOI: `10.1186/gb-2014-15-3-r46`. URL: `http://genomebiology.biomedcentral.com/articles/10.1186/gb-2014-15-3-r46` (visited on 03/18/2022).

[12] Stephen Nayfach et al. “An integrated metagenomics pipeline for strain profiling reveals novel patterns of bacterial transmission and biogeography”. eng. In: *Genome Research* 26.11 (Nov. 2016), pp. 1612–1625. ISSN: 1549-5469. DOI: `10.1101/gr.201863.115`.

[13] Nicola Segata et al. “Metagenomic microbial community profiling using unique clade-specific marker genes”. en. In: *Nature Methods* 9.8 (Aug. 2012), pp. 811–814. ISSN: 1548-7091, 1548-7105. DOI: `10.1038/nmeth.2066`. URL: `http://www.nature.com/articles/nmeth.2066` (visited on 03/18/2022).

[14] Moreno Zolfo et al. “MetaMLST: multi-locus strain-level bacterial typing from metagenomic samples”. In: *Nucleic Acids Research* 45.2 (Jan. 2017), e7. ISSN: 0305-1048. DOI: `10.1093/nar/gkw837`. URL: `https://doi.org/10.1093/nar/gkw837` (visited on 03/18/2022).

[15] Anton Bankevich et al. “SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing”. en. In: *Journal of Computational Biology* 19.5 (May 2012), pp. 455–477. ISSN: 1066-5277, 1557-8666. DOI: `10.1089/cmb.2012.0021`. URL: `http://www.liebertpub.com/doi/10.1089/cmb.2012.0021` (visited on 03/18/2022).

[16] Chris Smillie et al. “Mobility of plasmids”. eng. In: *Microbiology and molecular biology reviews: MMBR* 74.3 (Sept. 2010), pp. 434–452. ISSN: 1098-5557. DOI: `10.1128/MMBR.00020-10`.

[17] Maria Pilar Garcillán-Barcia, Andrés Alvarado, and Fernando de la Cruz. “Identification of bacterial plasmids based on mobility and plasmid population biology”. eng. In: *FEMS microbiology reviews* 35.5 (Sept. 2011), pp. 936–956. ISSN: 1574-6976. DOI: `10.1111/j.1574-6976.2011.00291.x`.

[18] Tossawan Jitwasinkul et al. "Plasmid metagenomics reveals multiple antibiotic resistance gene classes among the gut microbiomes of hospitalised patients". en. In: *Journal of Global Antimicrobial Resistance* 6 (Sept. 2016), pp. 57–66. ISSN: 2213-7165. DOI: `10.1016/j.jgar.2016.03.001`. URL: `https://www.sciencedirect.com/science/article/pii/S2213716516300261` (visited on 03/18/2022).

[19] Steven Flygare et al. "Taxonomer: an interactive metagenomics analysis portal for universal pathogen detection and host mRNA expression profiling". In: *Genome Biology* 17.1 (May 2016), p. 111. ISSN: 1474-760X. DOI: `10.1186/s13059-016-0969-1`. URL: `https://doi.org/10.1186/s13059-016-0969-1` (visited on 03/19/2022).

[20] Werner Ruppitsch et al. "Defining and Evaluating a Core Genome Multilocus Sequence Typing Scheme for Whole-Genome Sequence-Based Typing of Listeria monocytogenes". eng. In: *Journal of Clinical Microbiology* 53.9 (Sept. 2015), pp. 2869–2876. ISSN: 1098-660X. DOI: `10.1128/JCM.01193-15`.

[21] Artur J. Sabat et al. "Targeted next-generation sequencing of the 16S-23S rRNA region for culture-independent bacterial identification - increased discrimination of closely related species". eng. In: *Scientific Reports* 7.1 (June 2017), p. 3434. ISSN: 2045-2322. DOI: `10.1038/s41598-017-03458-6`.

[22] Ea Zankari et al. "PointFinder: a novel web tool for WGS-based detection of antimicrobial resistance associated with chromosomal point mutations in bacterial pathogens". eng. In: *The Journal of Antimicrobial Chemotherapy* 72.10 (Oct. 2017), pp. 2764–2768. ISSN: 1460-2091. DOI: `10.1093/jac/dkx217`.

[23] José Melo-Cristino et al. "First case of infection with vancomycin-resistant Staphylococcus aureus in Europe". eng. In: *Lancet (London, England)* 382.9888 (July 2013), p. 205. ISSN: 1474-547X. DOI: `10.1016/S0140-6736(13)61219-2`.

[24] Ruud H. Deurenberg et al. "Application of next generation sequencing in clinical microbiology and infection prevention". eng. In: *Journal of Biotechnology* 243 (Feb. 2017), pp. 16–24. ISSN: 1873-4863. DOI: `10.1016/j.jbiotec.2016.12.022`.

[25] J. W. A. Rossen et al. "Practical issues in implementing whole-genome-sequencing in routine diagnostic microbiology". eng. In: *Clinical Microbiology and Infection: The Official Publication of the European Society of Clinical Microbiology and Infectious Diseases* 24.4 (Apr. 2018), pp. 355–360. ISSN: 1469-0691. DOI: `10.1016/j.cmi.2017.11.001`.

Chapter 3

Detection of a novel *mcr-5.4* gene variant in hospital tap water by shotgun metagenomic sequencing

3. DETECTION OF A NOVEL *MCR-5.4* GENE VARIANT IN HOSPITAL TAP WATER BY SHOTGUN METAGENOMIC SEQUENCING

This chapter is a reproduction of the following publication:

G. Fleres, N. Couto, L. Schuele, M. A. Chlebowicz, C. I. Mendes, L. W. M. van der Sluis, J. W. A. Rossen, A. W Friedrich, S. García-Cobos, Detection of a novel mcr-5.4 gene variant in hospital tap water by shotgun metagenomic sequencing, Journal of Antimicrobial Chemotherapy, Volume 74, Issue 12, December 2019, Pages 3626–3628, DOI: `https://doi.org/10.1093/jac/dkz363`

As referenced in Chapter 1, section 1.1.2, sequencing has become a common tool in surveillance and infection prevention, when combined with epidemiological data, have undoubtedly provided immeasurable insights regarding the identification of potential sources of pathogenicity and transmission pathways. SMg approaches, just like in a clinical setting, have been a growing interest to deliver relevant results without a priori knowledge of what to expect from a particular environmental sample.

In this publication, second (see Chapter 1, section 1.2.1.2) and third (see Chapter 1, section 1.2.1.3) generation sequencing SMg has been applied to eight concentrated water samples collected the University Medical Center Groningen. In one of the samples, the novel detection of an *mcr-5* gene, named *mcr-5.4*, is reported. To the best of our knowledge, this is the first time that this gene, a mobile colistin resistance (*mcr*) determinant, has been recovered from a hospital water environment, with analysis suggesting the order of *Pseudomonadales* as the most probable host.

My contribution to this publication included the bioinformatics analysis of the *mcr-5.4* carrying sample thorough hybrid assembly using metaSPAdes. The resulting assembled contigs were binned with the MaxBin2 tool and the bin having the sequence carrying the gene of interest was taxonomically characterised with Kraken2.

Detection of a novel *mcr-5.4* gene variant in hospital tap water by shotgun metagenomic sequencing

Giuseppe Fleres[1], Natacha Couto[1], Leonard Schuele[1], Monika A. Chlebowicz[1], Catarina I. Mendes[1], Luc W. M. van der Sluis[2], John W. A. Rossen[1], Alex W. Friedrich[1], Silvia García-Cobos[1]

[1] University of Groningen, University Medical Center Groningen, Department of Medical Microbiology, Groningen, The Netherlands;

[2] Center of Dentistry and Oral Hygiene, University Medical Center Groningen, 9712 CP Groningen, The Netherlands

3.1 Letter

Sir,

Colistin is considered a last-resort antibiotic for treating serious infections caused by MDR Gram-negative bacteria. The efficacy of this antibiotic is challenged by the emergence and global spread of mobile colistin resistance (*mcr*) determinants, which threaten human, animal and environmental health. The first mobile colistin resistance gene (*mcr-1*) was reported in 2015 and since then up to eight different variants have been described [1]. In 2017, Borowiak et al.[2] described a new transposon-associated phosphoethanolamine transferase mediating colistin resistance, named *mcr-5*, in d-tartrate-fermenting *Salmonella enterica* subsp. enterica serovar Paratyphi B isolated from poultry. The *mcr-5.3* variant has been recently reported in *Stenotrophomonas* spp. from sewage water [3]. Here we report for the first time (to the best of our knowledge) the detection of an *mcr-5* gene in a hospital water environment using short-read metagenomic sequencing (SRMseq) and subsequent characterization using long-read metagenomic sequencing (LRMseq) to reveal its genetic environment.

In June 2017, eight tap-water samples (900 mL) were collected at the University Medical Center Groningen. Water samples were filtered (0.2 lm) and after DNA extraction (PowerWater DNA Extraction Kit, QIAGEN), SRMseq was performed on a MiSeq instrument (500 cycles) (Illumina). Antibiotic resistance genes were identified in the metagenome assemblies (CLC Genomics Workbench v10.1.1, QIAGEN) using ABRicate-0.7 (https://github. com/tseemann/abricate) and applying the following thresholds: .70% identity and .80% coverage. One sample contained an *mcr*-type gene (5% sequencing depth), with the nucleotide change 313C.T (amino acid change F105L) with respect to the original *mcr-5.1* gene, which was designated *mcr-5.4* by NCBI (accession no. MK965519). This sample was selected for LRMseq; the DNA libraries were prepared using the Rapid PCR Barcoding Kit (SQK-RPB004) from Oxford Nanopore Technologies (ONT) and loaded into a FLO-MIN106 R9.4 flow cell. The run was performed on a MinION device (ONT) and it proceeded for 24 h. The data were basecalled using Albacore (https://github.com/rrwick/Basecallingcomparison) and further processed with Poretools [4] and Porechop (https://github.com/rrwick/Porechop). Trimmed reads from SRMseq and LRMseq were used for hybrid-assembly analysis by metaSPAdes-3.13.0 [5]. After a BLAST search using the hybrid contig containing the *mcr-5.4* gene, the plasmid pSE13-SA01718 (accession no. KY807921.1) was listed as one of the hits with the highest identity and we used it as a reference for genome comparison with the Artemis Comparison Tool (ACT) v1.0 [6]. The *mcr-5.4*-carrying contig from the hybrid assembly was annotated using PATRIC v3.5.27 [7]. Trimmed reads from SRMseq were used to investigate the bacterial composition by OneCodex [8]. Finally, in order to predict the bacterial host of the *mcr-5.4* gene, a contig-binning analysis of the hybrid-assembled metagenome was performed using MaxBin2 v2.2.4 (https://sourceforge.net/projects/maxbin2/), probability threshold 0.9 and minimum contig length 1000 bp. The resulting bin containing the *mcr-5.4* gene was selected for taxonomy classification using Kraken2 (https://github.com/DerrickWood/kraken2) (minikraken2 DB v1).

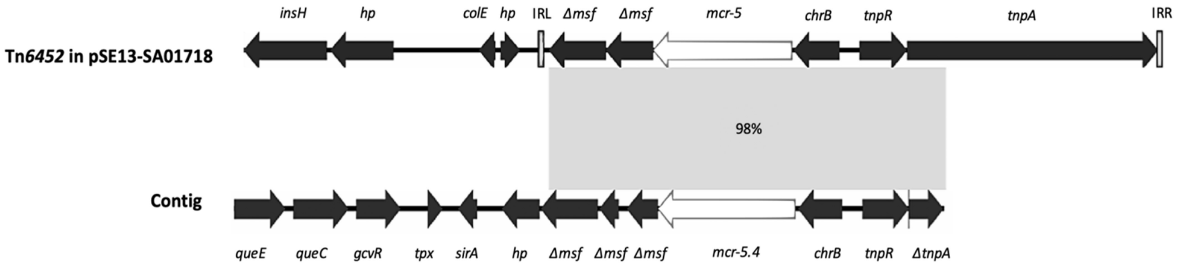


Figure 3.1: Comparative analysis of the genetic environment of *mcr-5* between the reference plasmid pSE13-SA01718 (accession no. KY807921.1) and the annotated hybrid metagenome contig (accession no. MK965519). The contig carrying the *mcr-5.4* gene consists of the following putative gene products: 7-carboxy-7-deazaguanine synthase (queE), 7-cyano-7-deazaguanine synthase (queC), glycine cleavage system transcriptional antiactivator GcvR (gcvR), thiol peroxidase (tpx), sulphurtransferase TusA family protein (sirA), hypothetical protein (hp), truncated MFS-type transporter (Δmsf), lipid A phosphoethanolamine transferase (*mcr-5.4*), ChrB domain protein (chrB), transposon resolvase (tnpR) and truncated transposon transposase (ΔtnpA). Areas with 98% identity between sequences are represented in light grey. Arrows indicate the position and direction of the genes. The transposon Tn6452 sequence in the reference plasmid pSE13-SA01718 is bounded by inverted repeats: IRL and IRR.

SRMseq showed the *mcr-5.4* gene detected in a contig of 2113 bp flanked by two truncated protein-coding sequences (CDSs), encoding the ChrB domain protein (involved in chromate resistance) and the Major Facilitator Superfamily (MFS) transporter. The hybrid-assembly analysis resulted in a contig of 8456 bp consisting of nine CDSs and four truncated CDSs (Figure 3.1). Comparative analysis of the genetic environment of the *mcr-5* gene, between the annotated hybrid metagenome contig and the reference plasmid pSE13-SA01718, showed a region of 4670 bp with 98% identity, corresponding to the backbone of the Tn6452 transposon (Figure 3.1). We observed three truncated CDSs for the MFS-type transporter in our contig instead of two as previously described in the reference sequence pSE13-SA01718. These differences did not appear to be due to sequencing errors when we checked the sequence MK965519, (i) using pilon (https://github.com/broadinstitute/pilon) to correct for errors in short-read sequencing data and (ii) using CLC Genomic Workbench to update the hybrid contig by mapping both long and short reads against the hybrid contig. We also observed a region of 3786 bp, with no identity either with the reference plasmid pSE13-SA01718 (Figure 3.1) or with any other sequence in the GenBank database.

Species previously described to harbour an *mcr-5* gene are *Escherichia coli*, *Pseudomonas aeruginosa*, *Salmonella enterica*, *Aeromonas hydrophila* and *Cupriavidus gilardii*. The bacterial composition analysis of the water sample using SRMseq showed the presence of *Pseudomonas* spp. (relative abundance: 0.004%), *Cupriavidus* spp. (relative abundance: 0.001%) and *Aeromonas* spp. (relative abundance: 0.0003%). The binning analysis produced a bin positive for the *mcr-5.4* gene consisting of 1336 contigs (genome size: 5 175 285 bp; genome completeness: 68.2%). This bin was taxonomically classified as bacteria (70.73%) and proteobacteria (64.90%), and from this the most abundant class was *Gammaproteobacteria* (37.20%) (order *Pseudomonadales*, 15.57%), followed by *Betaproteobacteria* (14.90%) (order *Burkholderiales*, 10.63%).

Colistin resistance determinants (*mcr*) have been rarely reported in water environments; *mcr-1* has been detected in both hospital sewage and in environmental water streams and *mcr-3* in environmental water [9, 10]. To the best of our knowledge, this is the first-time description of an *mcr-5* gene in an indoor and healthcare water environment. Despite the fact that the comparative analysis showed the hybrid contig covering a large region of Tn6452, neither the left inverted repeat (IRL) nor the right inverted repeat (IRR) have been found. In addition, the lack of the right transposon region does not allow us to search for other possible inverted repeats. Thus, it is not possible to conclude whether the described *mcr-5.4* gene is transferable or not. Taxonomic analysis suggested the order of *Pseudomonadales* as the most probable host of the *mcr-5.4* gene in the water sample. Further studies are needed to determine the frequency of this gene in hospital water and other water environments and to evaluate the potential risks for patients and healthcare workers.

3.2 Acknowledgements

We would like to thank Erwin C. Raangs for technical assistance.

3.3 Funding

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie SklodowskaCurie grant agreement 713660 (MSCA-COFUND-2015-DP 'Pronkjewail'), which includes in-kind contributions by commercial partners. None of the commercial partners had any influence on interpretation of reviewed data and conclusions drawn, or on drafting of the manuscript. This work was partly supported by the INTERREG VA (202085)-funded project EurHealth-1Health, part of a Dutch–German cross-border network supported by the European Commission, the Dutch Ministry of Health, Welfare and Sport (VWS), the Ministry of Economy, Innovation, Digitalization and Energy of the German Federal State of North RhineWestphalia and the German Federal State of Lower Saxony.

3.4 Transparency declarations

None to declare.

3.5 References

[1] Xiaoming Wang et al. “Emergence of a novel mobile colistin resistance gene, *mcr-8* , in NDM-producing *Klebsiella pneumoniae*”. en. In: *Emerging Microbes & Infections* 7.1 (Dec. 2018), pp. 1–9. ISSN: 2222-1751. DOI: `10.1038/s41426-018-0124-z`. URL: `https://www.tandfonline.com/doi/full/10.1038/s41426-018-0124-z` (visited on 01/24/2022).

[2] Maria Borowiak et al. “Identification of a novel transposon-associated phosphoethanolamine transferase gene, mcr-5, conferring colistin resistance in d-tartrate fermenting Salmonella enterica subsp. enterica serovar Paratyphi B”. en. In: *Journal of Antimicrobial Chemotherapy* 72.12 (Dec. 2017), pp. 3317–3324. ISSN: 0305-7453, 1460-2091. DOI: `10.1093/jac/dkx327`. URL: `https://academic.oup.com/jac/article/72/12/3317/4161410` (visited on 01/24/2022).

[3] Jun Li et al. “Co-Occurrence of Colistin and Meropenem Resistance Determinants in a *Stenotrophomonas* Strain Isolated from Sewage Water”. en. In: *Microbial Drug Resistance* 25.3 (Apr. 2019), pp. 317–325. ISSN: 1076-6294, 1931-8448. DOI: `10.1089/mdr.2018.0418`. URL: `https://www.liebertpub.com/doi/10.1089/mdr.2018.0418` (visited on 01/24/2022).

[4] N. J. Loman and A. R. Quinlan. “Poretools: a toolkit for analyzing nanopore sequence data”. en. In: *Bioinformatics* 30.23 (Dec. 2014), pp. 3399–3401. ISSN: 1367-4803, 1460-2059. DOI: `10.1093/bioinformatics/btu555`. URL: `https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btu555` (visited on 01/24/2022).

[5] Sergey Nurk et al. “metaSPAdes: a new versatile metagenomic assembler”. en. In: *Genome Research* 27.5 (May 2017), pp. 824–834. ISSN: 1088-9051, 1549-5469. DOI: `10.1101/gr.213959.116`. URL: `http://genome.cshlp.org/lookup/doi/10.1101/gr.213959.116` (visited on 03/25/2021).

[6] T. J. Carver et al. “ACT: the Artemis comparison tool”. en. In: *Bioinformatics* 21.16 (Aug. 2005), pp. 3422–3423. ISSN: 1367-4803, 1460-2059. DOI: `10.1093/bioinformatics/bti553`. URL: `https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/bti553` (visited on 01/24/2022).

[7] Alice R. Wattam et al. “Improvements to PATRIC, the all-bacterial Bioinformatics Database and Analysis Resource Center”. en. In: *Nucleic Acids Research* 45.D1 (Jan. 2017), pp. D535–D542. ISSN: 0305-1048, 1362-4962. DOI: `10.1093/nar/gkw1017`. URL: `https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/gkw1017` (visited on 01/24/2022).

[8] Samuel S Minot, Niklas Krumm, and Nicholas B Greenfield. *One Codex: A Sensitive and Accurate Data Platform for Genomic Microbial Identification*. en. preprint. Bioinformatics, Sept. 2015. DOI: `10.1101/027607`. URL: `http://biorxiv.org/lookup/doi/10.1101/027607` (visited on 01/24/2022).

[9] Feifei Zhao et al. “IncP Plasmid Carrying Colistin Resistance Gene *mcr-1* in Klebsiella pneumoniae from Hospital Sewage”. en. In: *Antimicrobial Agents and Chemotherapy* 61.2 (Feb. 2017). ISSN: 0066-4804, 1098-6596. DOI: `10.1128/AAC.02229-16`. URL: `https://journals.asm.org/doi/10.1128/AAC.02229-16` (visited on 01/24/2022).

[10] Hongmei Tuo et al. “The Prevalence of Colistin Resistant Strains and Antibiotic Resistance Gene Profiles in Funan River, China”. In: *Frontiers in Microbiology* 9 (Dec. 2018), p. 3094. ISSN: 1664-302X. DOI: `10.3389/fmicb.2018.03094`. URL: `https://www.frontiersin.org/article/10.3389/fmicb.2018.03094/full` (visited on 01/24/2022).

Chapter 4

DEN-IM: Dengue virus genotyping from shotgun and targeted metagenomics

This chapter is a reproduction of the following publication:

C. I. Mendes, E. Lizarazo, M. P. Machado, D. N. Silva, A. Tami, M. Ramirez, N. Couto, J. W. A. Rossen, J. A. Carriço, DEN-IM: dengue virus genotyping from amplicon and shotgun metagenomic sequencing. Microbial Genomics, Volume 6, Issue 3, March 2020. DOI: `https://doi.org/10.1099/mgen.0.000328`

The supplementary information referred throughout the text can be consulted in this chapter before the section of references.

Dengue virus (DENV) represents a public health threat and economic burden in affected countries. The risk of exposure to DENV is increasing, not only because of travel to endemic regions but also due to the broader dissemination of the mosquito vector, making the burden of dengue very significant.

The availability of genomic data is key to understanding viral evolution and dynamics and supporting improved control strategies. Currently, the use of second-generation sequencing technologies, which can be applied both directly to patient samples (shotgun metagenomics) and to PCR-amplified viral sequences (amplicon sequencing), is the most informative approach to monitor viral dissemination and genetic diversity by providing, in a single methodological step, identification and characterization of the whole viral genome at the nucleotide level. This makes DENV identification and characterization through genomic analysis by developing software where the lessons learned in Chapters 2 and 3 are applied.

We have developed DEN-IM, a one-stop, user-friendly, containerised and reproducible workflow for the analysis of Dengue virus short-read sequencing data from both amplicon and shotgun metagenomics approaches. DEN-IM was designed to perform a comprehensive analysis in order to generate either assemblies or consensus of full DENV coding sequences and to identify their serotype and genotype. DEN-IM can also detect all four DENV serotypes and the respective genotypes present in a spiked sample, raising the possibility that DEN-IM can play a role in the identification of co-infection cases whose prevalence is increasingly perceived in highly endemic areas.

My contribution to this publication included the design, implementation and optimisation of the DEN-IM workflow, including the creation of the Docker containers for all dependencies. Two databases, one comprising 3830 DENV sequences for the retrieval of the reads of interest from the input samples, and a second comprising 161 sequences representing the genetic diversity of all DENV sero and genotypes were constructed by me. Additionally, I've also written the manuscript.

DEN-IM: dengue virus genotyping from amplicon and shotgun metagenomic sequencing

Catarina I. Mendes[1,2,*], Erley Lizarazo[2,*], Miguel P. Machado[1], Diogo N. Silva[1], Adiana Tami[2], Mário Ramirez[1], Natacha Couto[2], John W. A. Rossen[2] and João A. Carriço[1]

[1]Instituto de Microbiologia, Instituto de Medicina Molecular, Faculdade de Medicina, Universidade de Lisboa, Lisboa, Portugal

[2]University of Groningen, University Medical Center Groningen, Department of Medical Microbiology and Infection Prevention, Groningen, The Netherlands

[*]Contributed equally

4.1 Abstract

Dengue virus (DENV) represents a public health and economic burden in affected countries. The availability of genomic data is key to understanding viral evolution and dynamics, supporting improved control strategies. Currently, the use of High Throughput Sequencing (HTS) technologies, which can be applied both directly to patient samples (shotgun metagenomics) and PCR amplified viral sequences (targeted metagenomics), is the most informative approach to monitor the viral dissemination and genetic diversity.

Despite many advantages, these technologies require bioinformatics expertise and appropriate infrastructure for the analysis and interpretation of the resulting data. In addition, the many software solutions available can hamper reproducibility and comparison of results. Here we present DEN-IM, a one-stop, user-friendly, containerised and reproducible workflow for the analysis of DENV sequencing data, both from shotgun and targeted metagenomics approaches. It is able to infer the DENV coding sequence (CDS), identify the serotype and genotype, and generate a phylogenetic tree. It can easily be run on any UNIX-like system, from local machines to high-performance computing clusters, performing a comprehensive analysis without the requirement of extensive bioinformatics expertise.

Using DEN-IM, we successfully analysed two DENV datasets. The first comprised 25 shotgun metagenomic sequencing samples of variable serotype and genotype, including an in vitro spiked sample containing the four known serotypes. The second dataset consisted of 106 targeted metagenomic sequences of DENV 3 genotype III where DEN-IM allowed detection of the intra-genotype diversity. The DEN-IM workflow, parameters and execution configuration files, and documentation are freely available at `https://github.com/B-UMMI/DEN-IM`.

4.1.1 Keywords

dengue virus, surveillance, metagenomics, reproducibility, workflow, containerization, scalability

4.2 Author Notes

All supporting data, code and protocols have been provided within the article or through supplementary data files.

Metagenomic sequencing data available under BioProject PRJNA474413. DEN-IM reports for the analysed datasets are available in Figshare under https://doi.org/10.6084/m9.figshare.11316599.v1. Phylogeny inference trees for the dengue virus typing database available in Figshare at https://doi.org/10.6084/m9.figshare.11316599.v1. The supplemental material is available in Figshare at https://doi.org/10.6084/m9.figshare.11316599.v1. DEN-IM's source code and documentation available at https://github.com/B-UMMI/DEN-IM.

4.3 Data Summary

1. The supplemental material and tables are available at Figshare under https://doi.org/10.6084/m9.figshare.9963812
2. The 106 DENV-3 targeted metagenomics paired-end short-read datasets are available under BioProject PRJNA394021. The 25 shotgun metagenomics dataset is available under BioProject PRJNA474413. The accession number for all the samples in the shotgun metagenomics dataset are available in the Supplementary material
3. The accession numbers for the 41 samples, belonging to zika virus, chikungunya virus and yellow fever virus shotgun and targeted metagenomic datasets are available in the Supplementary material.
4. DEN-IM reports for the analysed datasets are available at Figshare (https://doi.org/10.6084/m9.figshare.9318851).
5. Phylogeny inference trees for the dengue virus typing database available at Figshare (https://doi.org/10.6084/m9.figshare.9331826).
6. Code for the DEN-IM workflow is available at `https://github.com/B-UMMI/DEN-IM` and documentation, including step-by-step tutorials, is available at `https://github.com/B-UMMI/DEN-IM/wiki`.

4.4 Impact Statement

The risk of exposure to DENV is increasing not only by travelling to endemic regions, but also due to the broader dissemination of the mosquito, making the burden of dengue very significant.

The decreasing costs and wider availability of HTS makes it an ideal technology to monitor DENV's transmission. Metagenomics approaches decrease the time to obtain nearly complete DENV sequences without the need for time-consuming viral culture through the direct processing and sequencing of patient samples. A ready to use bioinformatics workflow, enabling the reproducible analysis of DENV, is therefore particularly relevant for the development of a straightforward HTS workflow.

DEN-IM was designed to perform a comprehensive analysis in order to generate either assemblies or consensus of full DENV CDSs and to identify their serotype and genotype. DEN-IM can also detect all four DENV genotypes present in a spiked sample, raising the possibility that DEN-IM can play a role in the identification of co-infection cases whose prevalence is increasingly appreciated in highly endemic areas. Although being ready-to-use, the DEN-IM workflow can be easily customised to the user's needs.

DEN-IM enables reproducible and collaborative research, being accessible to a wide group of researchers regardless of their computational expertise and resources available.

4.5 Introduction

The Dengue virus (DENV), a single-stranded positive-sense RNA virus belonging to the Flavivirus genus, is one of the most prevalent arboviruses and is mainly concentrated in tropical and subtropical regions. Infection with DENV results in symptoms ranging from mild fever to haemorrhagic fever and shock syndrome [1]. Transmission to humans occurs through the bite of Aedes mosquitoes, namely Aedes aegypti and Aedes albopictus [2]. In 2010, it was predicted that the burden of dengue disease reached 390 million cases/year worldwide [3]. The high morbidity and mortality of dengue makes it the arbovirus with the highest clinical significance [4]. DENV is a significant public health challenge in countries where the infection is endemic due to the high health and economic burden. Despite the emergence of novel therapies and ecological strategies to control the mosquito vector, there are still important knowledge gaps in the virus biology and its epidemiology [2].

The viral genome of $\sim$11,000 nucleotides, consists of a CDS of approximately 10.2 Kb that is translated into a single polyprotein encoding three structural proteins (capsid - C, premembrane - prM, envelope - E) and seven non-structural proteins (NS1, NS2A, NS2B, NS3, NS4A, NS4B and NS5). Additionally, the genome contains two Non-Coding Regions

(NCRs) at their 5' and 3' ends [5].

DENV can be classified into four serotypes (1, 2, 3 and 4), differing from each other from 25% to 40% at the amino acid level. They are further classified into genotypes that vary by up to 3% at the amino acid level [2]. The DENV-1 serotype comprises five genotypes (I-V), DENV-2 groups six (I-VI, also named American, Cosmopolitan, Asian-American, Asian II, Asian I and Sylvatic), DENV-3 four (I-III and V), and DENV-4 also four (I-IV).

Although real-time reverse transcription polymerase chain reaction (RT-PCR) will probably remain the front line in Dengue etiological diagnosis, the implementation of a surveillance system relying on HTS technologies allows the simultaneous identification and characterization by serotyping and genotyping of DENV cases at the nucleotide level in a single methodological step. Due to the high sensitivity of these technologies, previous studies showed that viral sequences can be directly obtained from patient sera using a shotgun metagenomics approach [6]. Alternatively, HTS can be used in a targeted metagenomics approach in which a PCR step is used to pre-amplify viral sequences before sequencing. In recent years, HTS has been successfully used as a tool for identification of DENV directly from clinical samples [6, 7]. This also allows the rapid identification of the serotype and genotype important for disease management as the genotype may be associated with disease outcome [8].

Several initiatives aim to facilitate the identification of the DENV serotype and genotype from HTS data. The Genome Detective project (https://www.genomedetective.com/) offers an online Dengue Typing Tool (https://www.genomedetective.com/app/typingtool/dengue/) [9] relying on BLAST and phylogenetic methods in order to identify the closest serotype and genotype, but it requires as input assembled genomes in FASTA format. The same project also offers the Genome Detective Typing Tool (https://www.genomedetective.com/app/typingtool/virus/) [10] identifying viruses present in a sample. Additionally, there are several tools available for viral read identification and assembly, such as VIP [11], virusTAP [12] and drVM [13], but none performs genotyping of the identified reads.

We developed DEN-IM as a ready-to-use, one-stop, reproducible bioinformatic analysis workflow for the processing and phylogenetic analysis of DENV using paired-end raw HTS data. DEN-IM is implemented in Nextflow [14], a workflow manager software that uses Docker (https://www.docker.com) containers with pre-installed software for all the workflow tools. The DEN-IM workflow, as well as parameters and documentation, are available at `https://github.com/B-UMMI/DEN-IM`.

4.6 The DEN-IM Workflow

DEN-IM is a user-friendly automated workflow enabling the analysis of shotgun or targeted metagenomics data for the identification, serotyping, genotyping, and phylogenetic analysis of DENV, as represented in Figure 4.1, accepting as input raw paired-end sequencing data (FASTQ files) and informing the user with an interactive and comprehensive HTML report (Supplementary Figure 4.5), as well as providing output files of the whole pipeline.

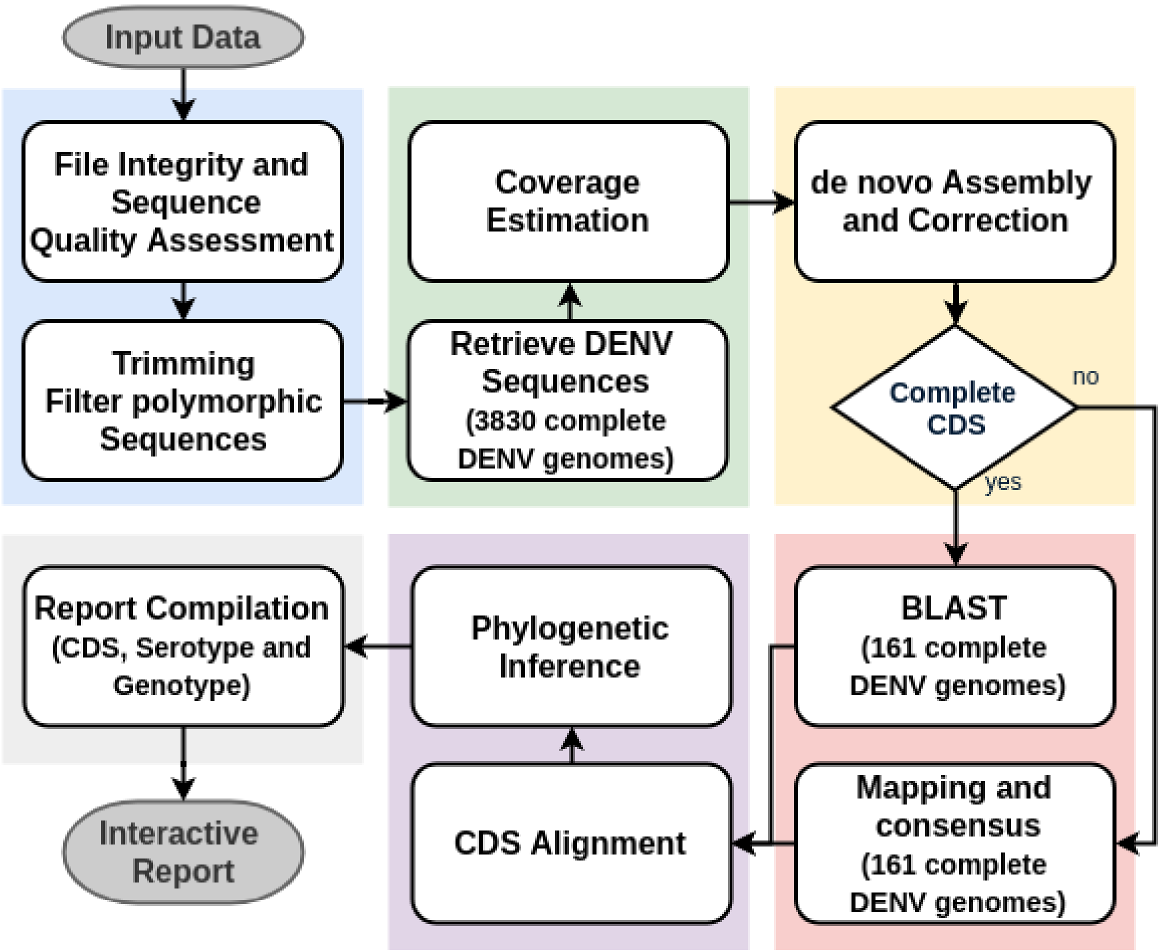


Figure 4.1: The DEN-IM workflow separated into five different components. The raw sequencing reads are provided as input to the first block (in blue), responsible for quality control and elimination of low-quality reads and sequences. After successful preprocessing of the reads, these enter the second block (green) for retrieval of the DENV reads using the mapping database of 3858 complete DENV genomes as a reference. This block also provides an initial estimate of the sequencing depth. After the de novo assembly and assembly correction block (yellow), the CDSs are retrieved and then classified with the reduced-complexity DENV typing database containing 161 sequences representing the known diversity of DENV serotypes and genotypes (red). If a complete CDS fails to be assembled, the reads are mapped against the DENV typing database and a consensus sequence is obtained for classification and phylogenetic inference. All CDSs are aligned and compared in a phylogenetic analysis (purple). Lastly, a report is compiled (grey) with the results of all the blocks of the workflow.

It is implemented in Nextflow, a workflow management system that allows the effortless deployment and execution of complex distributed computational workflows in any UNIX-based system, from local machines to high-performance computing (HPC) with a container engine installation, such as Docker (https://www.docker.com/), Shifter [15] or Singularity [16]. DEN-IM integrates Docker containerised images, compatible with other container engines, for all the tools necessary for its execution, ensuring reproducibility and the tracking of both software code and version, regardless of the operating system used.

Users can customise the workflow execution either by using command line options or by modifying the simple plain-text configuration files. To make the execution of the workflow as simple as possible, a set of default parameters and directives is provided. An exhaustive description of each parameter is available as Supplementary material (see 4.12.2).

The local installation of the DEN-IM workflow, including the docker containers with all the tools needed and the curated DENV database, requires 15 Gigabytes (Gb) of free disk space. The minimum requirements to execute the workflow are at least 5 Gb of memory and 4 CPUs. The disk space required for execution depends greatly on the size of the input data, but for the datasets used in this article, DEN-IM generates approximately 5 Gb of data per Gb input data. DEN-IM workflow can be divided into the following components:

4.6.0.1 Quality Control and Trimming

The Quality Control (QC) and Trimming block starts with a process to verify the integrity of the input data. If the sequencing files are corrupted, the execution of the analysis of that sample is terminated. The sequences are then processed by FastQC (https://www.bioinformatics.babraham.ac.uk/projects/fastqc/, version 0.11.7) to determine the quality of the individual base pairs of the raw data files. The low-quality bases and adapter sequences are trimmed by Trimmomatic [17] (version 0.36). In addition, paired-end reads with a read length shorter than 55 nucleotides after trimming are removed from further analyses. Lastly, the low complexity sequences, containing over 50% of poly-A, poly-N or poly-T nucleotides, are filtered out of the raw data using PrinSeq [17] (version 0.10.4).

4.6.0.2 Retrieval of DENV sequences

In the second step, DENV sequences are selected from the sample using Bowtie2 [18] (version 2.2.9) and Samtools [18] (version 1.4.1). As a reference we provide the DENV mapping database, a curated DENV database composed of 3830 complete DENV genomes. An in-depth description of this database is available as Supplementary material (see 4.12.1). A permissive approach is followed by allowing for mates to be kept in the sample even when only one read maps to the database in order to keep as many DENV derived reads as possible. The output of this block is a set of processed reads of putative DENV origin.

4.6.0.3 Assembly

DEN-IM applies a two-assembler approach to generate assemblies of the DENV CDS. To obtain a high confidence assembly, the processed reads are first de novo assembled with SPAdes [19] (version 3.12.0). If the full CDS fails to be assembled into a single contig, the data is re-assembled with the MEGAHIT assembler [20] (version 1.1.3), a more permissive assembler developed to retrieve longer sequences from metagenomics data. The resulting assemblies are corrected with Pilon [21] (version 1.22) after mapping the processed reads to the assemblies with Bowtie2.

If more than one complete CDS is present in a sample, each of the sequences will follow the rest of the DEN-IM workflow independently. If no full CDS is assembled neither with SPAdes nor with MEGAHIT, the processed reads are passed on to the next module for consensus generation by mapping, effectively constituting DEN-IM's two-pronged approach using both assemblers and mapping.

4.6.0.4 Typing

For each DENV complete CDS, the serotype and genotype is determined with the Seq_Typing tool (`https://github.com/B-UMMI/seq_typing`, version 2.0) [22] using BLAST [23] and the custom Typing database of DENV containing 161 complete sequences (see 4.12.1). The tool determines which reference sequence is more closely related to the query based on the identity and length of the sequence covered, returning the serotype and genotype of the reference sequence

If a complete CDS fails to be obtained through the assembly process, the processed reads are mapped against the same DENV typing database, with Bowtie2, using the Seq_Typing tool, with similar criteria for coverage and identity to those used with the BLAST approach. If a type is determined, the consensus sequence obtained follows through to the next step in the workflow. Otherwise, the sample is classified as Non-Typable and its process terminated.

4.6.0.5 Phylogeny

All DENV complete CDSs and consensus sequences analysed in a workflow execution are aligned with MAFFT [24] (version 7.402). By default, or if the number of samples analysed is less than 4, four representative sequences for each DENV serotype (1 to 4) from NCBI are also included in the alignment. The NCBI references included are NC_001477.1 (DENV-1), NC_001474.2 (DENV-2), NC_001475.2 (DENV-3) and NC_002640.1 (DENV-4). The closest reference sequence to each analysed sample in the DENV typing database to each analysed sample can also be retrieved and included in the alignment. With the resulting alignment, a Maximum Likelihood tree is constructed with RaXML [25] (version 8.2.11).

4.6.0.6 Output and Report

The output files of all tools in DEN-IM's workflow are stored in the 'results' folder in the directory of DEN-IM's execution, as well as the execution log file DEN-IM and for each component.

The HTML report (Supplementary Figure 4.5), stored in the 'pipeline_results' directory contains all results divided into four sections: report overview, tables, charts and phylogenetic tree. The report overview and all tables allow for selection, filtering and highlighting of particular samples in the analysis. All tables have information on if a sample failed or passed the quality control metrics highlighted by green, yellow or red signs for pass, warning and fail messages, respectively.

The *in silico* typing table contains the results of the serotype and genotype of each CDS analysed, as well as identity, coverage and GenBank ID of the closest reference in the DENV typing database. The quality control table shows information regarding the number of raw base pairs and number of reads in the raw input files and the percentage of trimmed reads. The mapping table includes the results for the mapping of the trimmed reads to the DENV mapping database, including the overall alignment rate, and an estimation of the sequence depth including only the DENV reads. For the assembly statistics table, the number of CDSs in each sample, the number of contigs and the number of assembled base pairs generated by either SPAdes or MEGAHIT assemblers is included. The number of contigs and assembled base pairs after correction with Pilon is also presented in the table. The assembled contig size distribution scatter plot is available in the chart section, showing the contig size distribution for the Pilon corrected assembled CDSs.

Lastly, a phylogenetic tree is included, rooted at midpoint for visualisation purposes, and with each tip coloured according to the genotyping results. If the option to retrieve the closest typing reference is selected, these sequences are also included in the tree with respective typing metadata. The tree can be displayed in several conformations provided by Phylocanvas JavaScript library (`http://phylocanvas.net`, version 2.8.1) and it is possible to zoom in or collapse selected branches. The support bootstrap values of the branches can be displayed, and the tree can be exported as a Newick tree file or as a PNG image.

4.7 Software comparison

DEN-IM offers a core assembly functionality, leveraging a de novo and consensus assembly approach, to obtain a full CDS sequence to perform geno- and serotyping, followed by phylogenetic positioning of the samples analysed. This results in a phylogenetic tree showing the genotyping results, presented in an HTML file.

There are several alternative tools, both command line and online based, capable of identifying DENV reads and performing assembly (Table 4.1). VIP and drVM are both stand-alone pipelines, like DEN-IM, and several components overlap with DEN-IM's but the retrieval of viral sequences is not targeted for DENV, and no serotyping and genotyping is performed. VIP performs a phylogenetic analysis against the reference database. VirusTAP is a web server for the identification of viral reads using the ViPR and IRD databases, or alternatively with the RefSeq Virus database. GenomeDetective is also a web service that provides two tools, one for the assembly of viral sequences from raw data (Virus tool) and another for serotyping and genotyping of DENV fasta sequences (Dengue Typing tool). Both tools need to be run consecutively, with the Virus Tool providing a link to redirect to the Dengue Typing tool when a DENV sequence is identified.

Table 4.1: DEN-IM's workflow comparison with different tools for the identification and genotyping of DENV from sequencing data.

Tool	**Quality Control**	**DENV Sequence Retrieval**	**Assembly**	**Typing**	**Phylogeny**	**Report**
DEN-IM	✓	✓	✓	✓	✓	✓(one report with all samples analysed)
VIP	✓	✓[1]	✓	X	✓	✓
VirusTAP	✓	✓[1]	✓	X	X	✓(web-based, one per sample, downloadable)
drVM	✓	✓[1]	✓	X	X	X
GenomeDetective Virus Tool	✓	X	✓	X	X	✓(web-based, one per sample)
GenomeDetective Dengue Typing Tool	X	X	X	✓[2]	X	✓(web-based, one per sample)

[1] Targeted for viral sequences, but not specific for DENV

[2] Sequence file can be received from GenomeDetective Virus Tool, as well as independently uploaded

Of all the tools listed in Table 4.1, only Genome Detective offers a tool to determine the DENV sero- and genotype from a fasta sequence, but the need to run their virus identification tool prior to obtain a sequence from the raw sequencing data increases the time to obtain a typing result, especially when a large number of sequences needs to be analysed. Moreover, these tools are not open source, so we are unable to compare the methodology used with our own. Additionally, there might be privacy issues in submitting data to external services, like VirusTAP and GenomeDetective, especially when handling metagenomics data that contain human sequences subjected to strict privacy laws in most countries. Therefore, a stand-alone tool is preferable for these analyses since these can be run in secure local environments. DEN-IM's main advantage when compared to web-based platforms is the ability to analyse batches of samples in a scalable manner, obtaining a report summarizing all the samples analysed and a phylogeny analysis of all DENV CDSs recovered.

4.8 Results

To evaluate the DEN-IM workflow performance, we analysed three datasets, one containing shotgun metagenomics sequencing data of patient samples (see Table 4.2), a second with amplicon sequencing data, a set with 106 paired-end samples obtained from Parameswaran et al [26] and another set with 78 single-end samples available under BioProject PRJNA321963, and a third dataset of publicly available sequences, both from amplicon and shotgun metagenomics, containing 45 chikungunya virus (CHIKV) samples, 66 zika virus (ZKV), and 21 yellow fever virus (YFV) samples (see Table 4.3). All analyses were executed with the default resources and parameters (available at `https://github.com/B-UMMI/DEN-IM`). In the shotgun metagenomics and the single-end amplicon sequencing datasets the closest typing reference in the final tree and the NCBI DENV references for each serotype were included in the phylogenetic analysis. The resulting reports for each dataset are available on Figshare at `https://doi.org/10.6084/m9.figshare.9318851`.

4.8.0.1 Shotgun metagenomics dataset

We analysed a dataset containing 22 shotgun metagenomics paired-end short-read Illumina sequencing samples from positive dengue cases, one positive control (purified from a DENV culture), one negative control (blank), and an in vitro spiked sample containing the 4 DENV serotypes (see 4.12.3). On average, each sample took 7 minutes to analyse. A total of 75 CPU hours were used to analyse the 25 samples, with a total of 17 Gb in size. This analysis resulted in 69 Gb of data. The negative control and the 92-1001 sample had no reads after trimming and filtering of low complexity reads, therefore they were removed from further analysis (see 4.4). When mapping to the DENV mapping database, the percentage of DENV reads in the 21 clinical samples, positive control and spiked sample passing QC ranged from 0.01% (sample UCUG0186) to 85.38% (sample Positive Control - PC). After coverage depth estimation, the analysis of the samples 91-0115 and UCUG0186 was terminated due to a low proportion of DENV reads (0.05% and 0.01% respectively). Therefore, they failed to meet the threshold criterion of having an estimated depth of coverage of $\geq$10x (estimated coverages of 3.17x and 5.65x, respectively). Sequence data of sample 91-0106 contained only 960 DENV reads (0.03%) but these were successfully assembled into a CDS with an estimated depth of coverage of 14.71x.

In the assembly module, the remaining 19 samples, the spiked sample and the PC were assembled with DEN-IM's two assembler approach. Twenty-four full CDS were assembled (see 4.6), even in samples originally having DENV read content as low as 0.03% of the total reads. Sixteen samples, including the spiked sample and the positive control, were assembled in the first step with the SPAdes assembler, and five in the second with the MEGAHIT assembler. In the spiked sample, all four CDSs were successfully assembled and recovered.

Serotype and genotype were successfully determined for the 24 DENV CDSs by BLAST (see 4.6). The most common were serotype 2 genotype III (Asian American) and serotype 4 genotype II, with 8 samples each (33%), followed by serotype 3 genotype III (n=5, 21%), serotype 1 genotype V (n=2, 8%) and serotype 2 genotype V (Asian I) (n=1, 4%). All CDSs recovered and the respective closest reference genome in the typing database were aligned and a maximum likelihood phylogenetic tree was obtained to visualise the relationship between the samples (Figure 4.2). There was a perfect concordance between the results of serotyping and genotyping and the major groups in the tree. Four distinct CDSs were assembled for the spiked sample that resulted in different coverages of each serotype CDS (2032x times coverage for DENV-2, 229x coverage for DENV-1, 76x coverage for DENV-3 and 30x times coverage for DENV-4), in accordance with the ranking order of the real-time RT-PCR results (see 4.12.3).

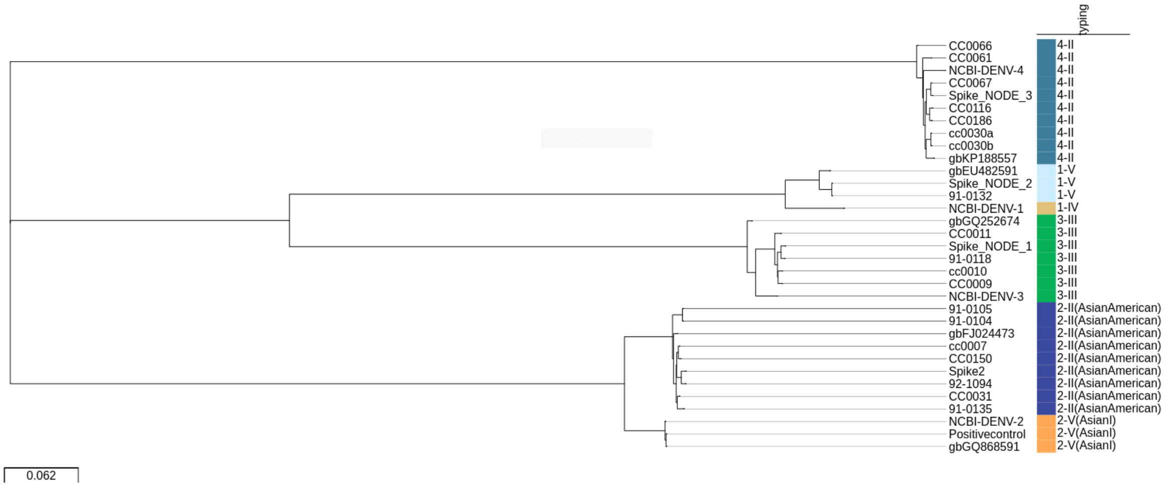


Figure 4.2: Phylogenetic reconstruction of the shotgun metagenomic dataset. Maximum Likelihood tree in the DEN-IM report for the 24 complete CDSs (n=21 samples) obtained with the metagenomics dataset, the respective closest references in the typing database (identified by their GenBank ID), and the NCBI DENV references for each serotype (NCBI-DENV-1: NC_001477.1, NCBI-DENV-2: NC_001474.2, NCBI-DENV-3: NC_001475.2, NCBI-DENV-4: NC_002640.1). The tree is midpoint rooted for visualisation purposes and the scale represents average substitutions per site. The colours depict the DENV genotyping results.

4.8.0.2 The Amplicon Sequencing Dataset

To validate DEN-IM's performance in a amplicon sequencing approach, a dataset of 106 paired-end HTS samples of PCR products using primers targeting DENV-3 (27) were analysed (see 4.12.4). On average, each sample took 5 minutes to analyse. The 106 samples, with 51 Gb in size, took 3622 CPU hours to be analysed, resulting in 424 Gb of data.

No samples failed the quality control block (see Table 4.5). The proportion of DENV reads ranged from 24.72% (SRR5821236) to 99.81% (SRR5821254) of the total processed reads. The samples with less than 70% DENV DNA were taxonomically profiled with Kraken2 (28) and the minikraken2_v2 database (`ftp://ftp.ccb.jhu.edu/pub/data/`

`kraken2_dbs/minikraken2_v2_8GB_201904_UPDATE.tgz`) and the source of contamination was determined to have come largely from Human DNA (see Table 4.6).

Of the 106 samples, 43 (41%) managed to assemble a complete CDS sequence (see Table 4.5) whereas a mapping approach was used for the remaining 63 samples (60%) and a consensus CDS was generated. For the assembled CDSs, all but one were assembled with MEGAHIT after not producing a full CDS with SPAdes. Moreover, pronounced variation on the size of the assembled contigs is evident in the contig size distribution plot (see 4.7).

All 106 CDSs recovered belonged to serotype 3 genotype III. Despite the same classification, the maximum likelihood tree indicates that there is detectable genetic diversity within the dataset (486 SNPs in 10237 nucleotides) (Figure 4.3).

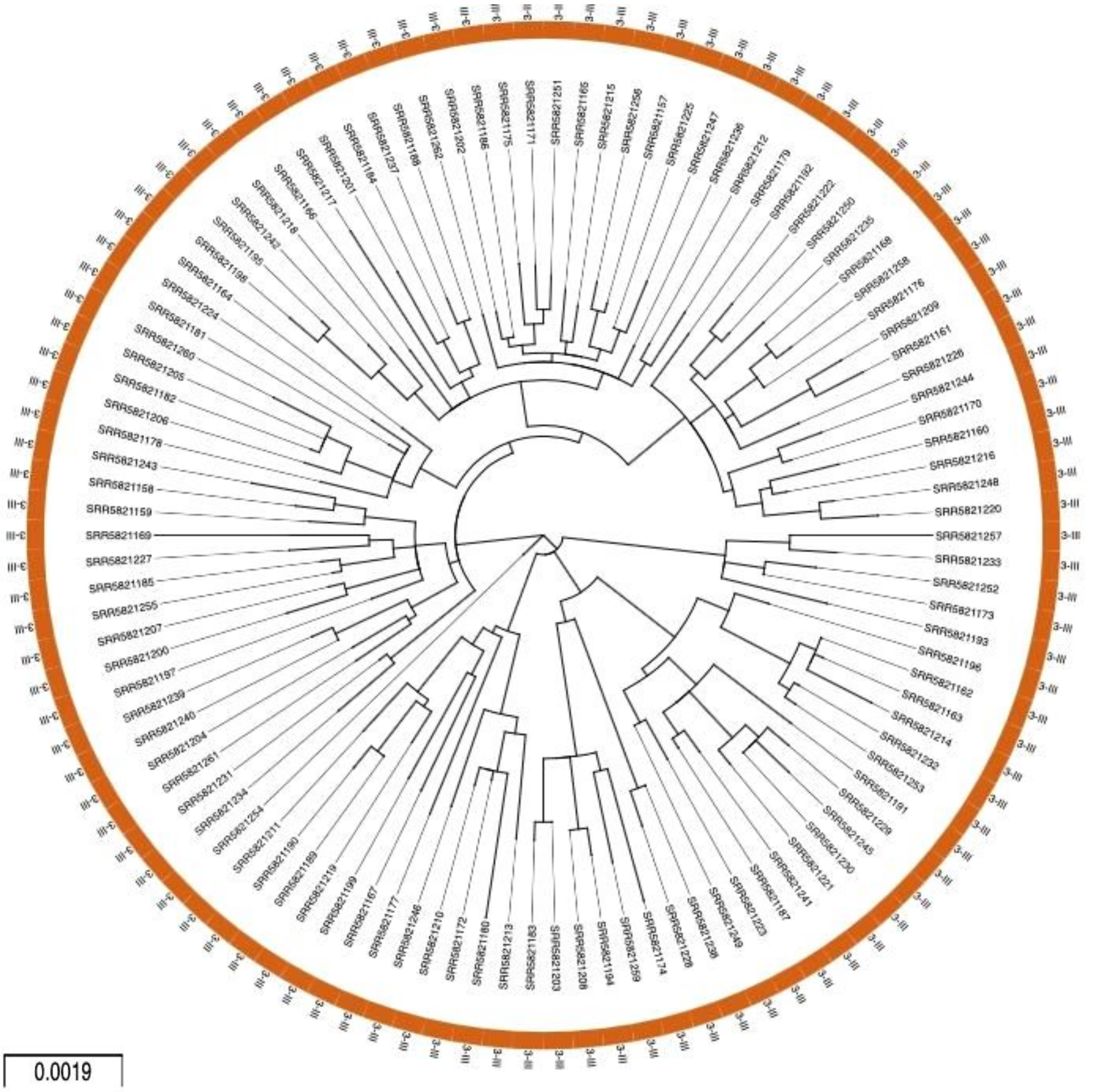


Figure 4.3: Phylogenetic reconstruction of the paired-end targeted metagenomic dataset. Maximum likelihood circular tree in the DEN-IM report for the 106 complete CDSs obtained with the targeted metagenomics dataset (n=106). All samples belong to serotype 3 genotype III. The scale represents average substitutions per site.

A second amplicon dataset, containing 78 DENV-1 single-end samples recovered from different Aedes aegypti isofemale hosts were analysed (see 4.12.4). On average, each sample took 3 minutes to analyse. The 78 samples, with 19 Gb in size, took 278 CPU hours to be analysed, resulting in 203 Gb of data.

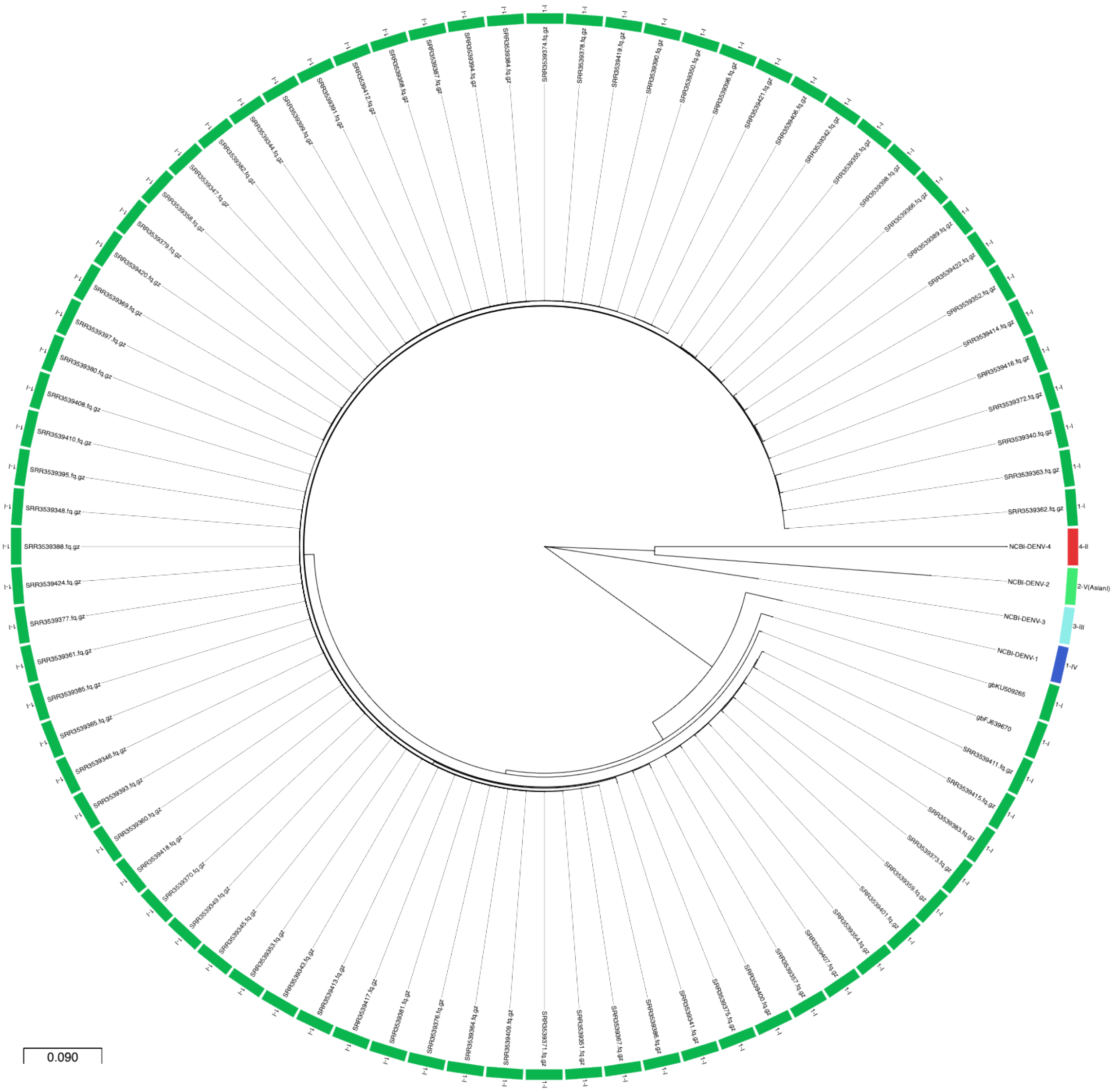


Figure 4.4: Phylogenetic reconstruction of the single-end targeted metagenomic dataset. Maximum likelihood circular tree in the DEN-IM report for the 78 complete CDSs obtained with the targeted metagenomics dataset (n=78) and the NCBI DENV references for each serotype (NCBI-DENV-1: NC_001477.1, NCBI-DENV-2: NC_001474.2, NCBI-DENV-3: NC_001475.2, NCBI-DENV-4: NC_002640.1). All samples belong to serotype 1 genotype I. The scale represents average substitutions per site.

No samples failed the quality control block and the proportion of DENV reads ranged from 59% (SRR3539343) to 96% (SRR3539408) of the total processed reads (see Table 4.7). Of the 78 samples, 53 (68%) assembled a complete CDS sequence and in the remaining 25 (32%) the complete CDS was obtained through mapping. All CDSs recovered, the respective closest reference genome in the typing database and NCBI's references for each DENV serotype were aligned and a maximum likelihood phylogenetic tree was obtained (Figure 4.4). All 78 samples belong to serotype 1 genotype I and, similarly to the previous dataset of

106 samples, there was detectable genetic diversity within the dataset (651 SNPs in 10808 nucleotides excluding reference sequences).

4.8.0.3 The Non-DENV Arbovirus Dataset

In order to evaluate DEN-IM's specificity to DENV sequences, a third dataset of publicly available sequences of arbovirus other than DENV, both from amplicon and shotgun metagenomics, was analysed containing 45 CHIKV samples, 66 ZKV, and 21 YFV samples (see Table 4.3). All 132 samples failed DEN-IM's workflow, 16 due to not enough sequencing data remaining after quality trimming, and the remaining 116 due to very low estimated coverage of the DENV genome (less than 0.01x), as expected.

4.9 Conclusion

We have successfully analysed two DENV datasets, one comprising 25 shotgun metagenomics sequencing samples and a second of 106 paired-end and 78 single-end targeted metagenomics samples.

In the first dataset, we recovered 24 CDSs from 19 clinical samples, including a spiked sample and a positive control that were correctly serotyped and genotyped. Besides the negative control, 3 samples did not return typing information due to failing quality checks.

The proportion of DENV reads in the metagenomics samples was highly variable. This may reflect the viral load in patients in which DENV was detected by real-time RT-PCR. In the spiked sample, containing 4 distinct DENV serotypes, all four were correctly detected despite not being present in equal concentrations, highlighting the potential of the DEN-IM workflow to accurately detect and recover multiple DENV genomes from samples with DENV co-infection, even if the serotypes are present in low abundance. Indeed, recent studies from areas of high endemicity suggest that co-infection with multiple DENV serotypes may frequently occur [27, 28] and the co-circulation of different DENV strains of the same serotype, but distinct genotypes, in these areas [27] raises the possibility of simultaneous infection with more than one genotype.

When analysing the 106 paired-end targeted metagenomics dataset, only 43 CDS samples were de novo assembled. For the remaining 63 samples, consensus sequences were obtained through mapping. In all samples DENV 3-III was correctly identified. Similar results were obtained for the 78 single-end samples where 53 CDS were de novo assembled, and 25 consensus sequences were obtained through mapping. All samples were identified as DENV-1 I. These two datasets demonstrate the success of DEN-IM's two-pronged approach of combining assembler and mapping. DEN-IM's specificity was shown when it found no false positive results when analysing a dataset containing arboviruses other than DENV.

DEN-IM is built with modularity and containerisation as keystones, leveraging the parallelization of processes and guaranteeing reproducible analyses across platforms. The modular design allows for new modules to be easily added and tools that become outdated to be easily updated, ensuring DEN-IM's sustainability. The software versions are also described in the Nextflow script and configuration files, and in the dockerfiles for each container, allowing the traceability of each step of data processing.

Being developed in Nextflow, DEN-IM runs on any UNIX-like system and provides out-of-the-box support for several job schedulers (e.g., PBS, SGE, SLURM) and integration with containerised software like Docker or Singularity. While it has been developed to be ready to use by non-experts, not requiring any software installation or parameter tuning, it can still be easily customised through the configuration files.

The interactive HTML reports (see 4.5) provide an intuitive platform for data exploration, allowing the user to highlight specific samples, filter and re-order the data tables, and export the plots as needed.

Together with the workflow and software containers, a database containing 3858 complete DENV genomes for DENV sequence retrieval and a subset database with 161 curated DENV genomes for serotyping and genotyping are provided. While constructing these databases, the obstacles reported by Cuypers et al [29] were apparent, namely the lack of formal definition of a DENV genotype and the lack of a standardised classification procedure that could assign sequences to a previously defined genotypic/sub-genotypic clade [29]. Discrepancies between the phylogenetic relationship and the genotype assignment were frequent and, throughout this study, the classification of some strains within the ViPR database [30] was updated. As suggested previously [29], further evaluation of the DENV classification will benefit future research and investigation into the population dynamics of this virus. Our typing approach was designed to use the currently accepted DENV classification. However, DEN-IM can be easily modified if a new DENV classification system is to be established in the future.

DEN-IM provides a user-friendly workflow that makes it possible to analyse short-read raw sequencing data from shotgun or targeted metagenomics for the presence, typing and phylogenetic analysis of DENV. The use of containerised workflows, together with shareable reports, will allow an easier comparison of results globally, promoting collaborations that can benefit the populations where DENV is endemic. The DEN-IM source code is freely available in the DEN-IM GitHub repository (`https://github.com/B-UMMI/DEN-IM`), which includes a wiki with full documentation and easy to follow instructions.

4.10 Author Statements

4.10.1 Authors and contributions

C.I.M., E.L., N.C., M.R., J.A.C. and J.W.A.R. designed the workflow. C.I.M implemented and optimised the workflow, created the Docker containers, and wrote the manuscript. M.P.M. implemented the DENV genotyping module in the workflow and D.N.S. contributed to the development of DEN-IM's HTML report. E.L., A. T., and N.C. provided the shotgun metagenomics data used to test and validate the workflow and wrote the manuscript. A.T., N.C., M.R., J.A.C. and J.W.A.R. critically revised the article. All authors read, commented on, and approved the final manuscript.

4.10.2 Conflict of interest

The authors declare that they have no competing interests.

4.10.3 Funding information

C.I.M. was supported by the Fundação para a Ciência e Tecnologia (grant SFRH/BD/129483/2017). Erley Lizarazo received the Abel Tasman Talent Program grant from the UMCG, University of Groningen, Groningen, The Netherlands. This work was partly supported by the ONEIDA project (LISBOA-01-0145-FEDER-016417) co-funded by FEEI–Fundos Europeus Estruturais e de Investimento from Programa Operacional Regional Lisboa 2020 and by national funds from FCT–Fundação para a Ciência e a Tecnologia and by UID/BIM/50005/2019, project funded by Fundação para a Ciência e a Tecnologia (FCT)/ Ministério da Ciência, Tecnologia e Ensino Superior (MCTES) through Fundos do Orçamento de Estado.

4.10.4 Ethical approval

This study followed international standards for the ethical conduct of research involving human subjects. Data and sample collection was carried out within the DENVEN and IDAMS (International Research Consortium on Dengue Risk Assessment, Management and Surveillance) projects. The study was approved by the Ethics Review Committee of the Biomedical Research Institute, Carabobo University (Aval Bioetico #CBIIB(UC)-014 and CBIIB-(UC)-2013-1), Maracay, Venezuela; the Ethics, Bioethics and Biodiversity Committee (CEBioBio) of the National Foundation for Science, Technology and Innovation

(FONACIT) of the Ministry of Science, Technology and Innovation, Caracas, Venezuela; the regional Health authorities of Aragua state (CORPOSALUD Aragua) and Carabobo State (INSALUD); and by the Ethics Committee of the Medical Faculty of Heidelberg University and the Oxford University Tropical Research Ethics Committee.

4.10.5 Consent for publication

All individuals, or a parent or legal guardian if under 16 years of age, whose sample and data were collected have given consent to participate in the study.

4.10.6 Acknowledgements

The authors would like to thank Tiago F. Jesus and Bruno Ribeiro-Gonçalves for their invaluable help with the Nextflow implementation. We would also like to thank Erwin C. Raangs from the UMCG for his assistance in the sequencing of the shotgun metagenomics dataset. Additionally, the authors thank Lize Cuypers, Krystof Theys, Pieter Libin and Gilberto Santiago for their discussions on DENV nomenclature and classification. This work was done in collaboration with the ESCMID Study Group on Molecular and Genomic Diagnostics (ESGMD), Basel, Switzerland.

4.11 Data Bibliography

- Catarina Inês Mendes. DEN-IM supplemental material and tables are deposited at Figshare with DOI 10.6084/m9.figshare.9963812 (https://doi.org/10.6084/m9.figshare.9963812.v3).
- Catarina Inês Mendes. DEN-IM reports for the analysed datasets tables are deposited at Figshare with DOI 0.6084/m9.figshare.9318851 (https://doi.org/10.6084/m9.figshare.9318851).
- Catarina Inês Mendes. Phylogeny inference trees for the dengue virus typing database are deposited at Figshare with DOI 10.6084/m9.figshare.9331826 (https://doi.org/10.6084/m9.figshare.9331826).
- Catarina Inês Mendes. Code for the DEN-IM workflow (https://github.com/B-UMMI/DEN-IM).

4.12 Supplementary Material

4.12.1 Dengue virus reference databases

We have compiled a database of 3858 complete DENV genomes obtained from the NIAID Virus Pathogen Database and Analysis Resource (ViPR) in October 2019 [30] (`http://www.viprbrc.org/`).The sequences were distributed unevenly throughout the four DENV serotypes, with DENV-1 being the most represented with 1636 sequences (42.72%), followed by DENV-2 with 1067 sequences (27.86%), DENV-3 with 807 sequences (21.07%), and DENV-4 with 320 sequences (8.36%). The selection criteria for the search were as follows: a) complete genome sequence only, b) human or mosquito host, c) collection year (1950-2018). Data available from all countries was included and duplicated sequences were removed and only the sequences with sub-type data were kept. A representative of DENV serotype 1 genotype III was introduced (EF457905, recovered from monkey) as no representatives were available with the search criteria used. This genotype is sylvatic and considered extinct [31, 32]. Additionally, any sample with IUPAC codes in the sequence provided were excluded.

In order to recover the maximum number of DENV reads from the input HTS data in the first mapping step (Figure 2.1), we maintained the database with the 3858 complete DENV genomes to retain as much diversity as possible. This database is referred as DENV mapping database and is available on GitHub at `https://github.com/B-UMMI/DEN-IM/blob/master/ref/DENV_MAPPING_V3.fasta`.

For typing purposes, overly similar sequences in the collection were removed from the database by clustering the sequences in each serotype at 98% nucleotide similarity with CD-HIT [33], leaving 161 representative sequences of all described DENV serotypes and genotypes, with 46 DENV-1 sequences (Table 4.8), 63 DENV-2 (Table 4.9), 25 DENV-3 (Tables 4.10) and 27 DENV-4 (Table 4.11). This database is referred as DENV typing database and is available on GitHub at `https://github.com/B-UMMI/DEN-IM/blob/master/ref/DENV_TYPING_V3.fasta`. This step is necessary to speed up the classification step for genotyping.

Phylogenetic analysis of typing collection was performed by aligning the full reference genomes with MAFFT [24], in auto mode and with automatic sequence orientation adjustment. A phylogenetic tree was inferred with RAxML (version 8.12.11) [25] using the GTR-Γ substitution model and 500 times bootstrap. Additionally, the same analysis was performed with the envelope protein (E) only, as this region has been used traditionally for sero- and genotyping [34–40], and continues to be the standard in many laboratories for genotyping. The resulting trees are available as supplemental material (Figures 4.8 to 4.11) and on Figshare (`https://10.6084/m9.figshare.9331826`).

The sequence JF459993 from the DENV-1 collection, as of April 2019, was annotated in ViPR as belonging to genotype IV, but in our analysis, it clustered within genotype I clade (Figure 4.8). The classification of DENV-1 I was also obtained from GenomeDetective Dengue Subtyping Tool (`https://www.genomedetective.com/app/typingtool/dengue/`), so we proceeded to alter the annotation of this particular sample (Table 4.7). In order to harmonise dengue nomenclature, the system uses Roman-numeric labels to identify the genotype, with the exception of Serotype 2 (Table 4.5), which used both Roman-numeric and geographic origin due to the widespread adoption of the latter.

4.12.2 Workflow parameters

The short-read data is passed as input through the "–fastq" parameter, that by default is set to match all files in the "fastq" folder that match the pattern "*_R1,2*". Both paired and single-end sequencing data can be passed through with the "–fastq" parameter, as defined by the pattern used.

In the process to verify the integrity of the short-read raw sequencing data, the integrity of the input files is assessed by attempting to decompress and read the files. An estimation of the depth of coverage is also performed. By default, the input size ("—genomeSize") is set to 0.012 Mb and the minimum coverage depth ("—minCoverage") is set to 10. If any input file is found to be corrupt, its progression in the workflow is aborted.

In the FastQC and Trimmomatic module, FastQC (https://www.bioinformatics.babraham.ac.uk/projects/fastqc/) is run with the parameters "–extract –nogroup –format fastq". FastQC will inform Trimmomatic [41] on how many bases to trim from the 3'and 5' ends of the raw reads. By default, Trimmomatic uses the default set of Illumina adapters provided with the workflow but this behaviour can be overwritten with the "—adapters" parameter. The additional Trimmomatic parameters "-–trimSlidingWindow", "—trimLeading", "—trimTrailing" and "—trimMinLength"can all be set to different values.

The removal of low complexity sequences is done with PrinSeq [17] using a custom parameter ("–pattern"), which by default is set to the value "A 50%; T 50%; N 50%", removing sequences whose content is at least half composed of a polymeric sequence (A, T or N).

To retrieve the reads that map to the DENV reference database, Bowtie2 [18] is run with default parameters with the DENV mapping database as a reference. For paired-end data, the reads and their mates that map to the reference are retrieved with "samtools view -buh -F 12" and "samtools fastq" commands. In single-end reads, all mapped reads are retrieved with "samtools view -buh -F 4" and "samtools fastq". The DENV mapping database can be altered with the "—reference" parameter, or alternatively, a Bowtie2 index can be provided with the "—index" parameter. This allows for the workflow to work with other databases obtained through public and owned DENV genomes. The coverage estimation step is performed on the retrieved DENV reads with the same parameters are the first estimation ("—genomeSize=0.012" and "—minCoverage=10").

In the assembly process, the retrieved DENV reads are firstly assembled with SPAdes Genome Assembler [19] with the options "–careful –only-assembler –cov-cutoff". The coverage cut-off if dictated by the "—spadesMinCoverage" and "—spadesMinKmerCoverage" parameters, set to 2 by default. If the assembly with SPAdes fails to produce a contig equal or greater than the value defined in the "–minimumContigSize" parameter (default of 10000), the data is re-assembled with the MEGAHIT assembler [20] with default parameters. By default, the k-mers to be used in the assembly in both tools ("–spadesKmers" and "–megahitKmers") are automatically determined depending on the read size. If the maximum read length is equal or greater than 175 nucleotides, the assembly is done with the k-mers "55, 77, 99, 113, 127", otherwise the k-mers "21, 33, 55, 67, 77" are used.

To correct the assemblies produced, the Pilon tool [21] is run after mapping the QC'ed reads back to the assembly with Bowtie2 and "samtools sort". This process also verifies the coverage and the number of contigs produced in the assembly. The behaviour can be altered with the parameters "–minAssemblyCoverage", "–AMaxContigs" and "–genomeSize", set to "auto", 1000 and 0.01 Mb by default. The first parameter, when set to 'auto', the minimum assembly coverage for each contig required is set to the 1/3 of the assembly mean coverage or to a minimum of 10x. The ratio of contig number per genome MB is calculated based on the genome size estimation for the samples. The contigs larger than the value defined in the "–size" parameter (default of 10000 nucleotides) are considered to be complete CDSs and follow the rest to the workflow independently. If no complete CDS is recovered, the QC'ed read data is passed to the mapping to module that does the DENV typing database and consensus generation.

The serotyping and genotyping are performed with the Seq_Typing tool [22] with the command "seq_typing.py assembly" or "seq_typing.py reads", using as reference the provided curated DENV typing database. It is possible to retrieve the genomes of the closest references and include them in the downstream analysis by changing the "–get_reference" option to "true". By default, this is not included in the analysis.

The CDSs, and the reference sequences if requested, are aligned with the MAFFT tool [24] with the options "–adjustdirection –auto". By default, four representative sequences for each DENV serotype (1 to 4) from NCBI is also included in the alignment. This option can be turned off by changing the value of "–includeNCBI" to "false". If the number of sequences in the alignment is less than 4 these are automatically added.

A maximum likelihood phylogenetic tree is obtained with the RaXML tool [25] with the options "-p 12345 -f -a". Additionally, and by default, the substitution model ("–substitutionModel") is set to "GTRGAMMA", the bootstrap is set to 500 ("–bootstrap") and the seed to "12345" ("–seedNumber").

4.12.3 Shotgun Metagenomics Sequencing Data

Samples of plasma (n=9) and serum samples (n=13) from confirmed dengue symptomatic patients were collected in Venezuela between 2010-2015 (Table S2) (see Availability of supporting materials). DENV positivity was confirmed by either RT-qPCR [42] or nested RT-PCR [36].

As a positive control sample, the supernatant of a viral culture containing DENV-2 strain 16681 was used. The negative control sample consisted of DNA- and RNA-free water (Sigma-Aldrich, St. Louis, MO, USA).

A spiked sample was produced consisting of a mixture of four 5 µl of cDNA isolated from clinical samples including all DENV serotypes (DENV-1 to -4). The viral cDNA for these samples was not in equal concentration and the viral copy number in the clinical samples was assessed by RT-PCR [36]. The results were as follow: DENV-2 with 1070000 copies/µl, DENV-1 with 117830 copies/µl, DENV-3 with 44300 copies/µl and DENV-4 with 6600 copies/µl.

The cDNA libraries were generated using either the NEBNext® RNA First and Second strand modules and the Nextera XT DNA library preparation kit (NXT), or the TruSeq RNA V2 library preparation kit (TS). The libraries were sequenced in MiSeq and NextSeq instruments using 300-cycles v2 paired-end cartridges.

The DEN-IM workflow was executed with the raw sequencing data using the default parameters and resources in an HPC cluster with 300 Cores/600 Threads of Processing Power and 3 TB RAM divided through 15 computational nodes, 9 with 254 GB Ram and 6 with 126GB RAM.

4.12.4 Amplicon Sequencing Data

The accession numbers for the 106 DENV-3 paired-end amplicon sequencing paired-end short-read datasets are available under BioProject PRJNA394021. The accession numbers for the 78 DENV-1 amplicon sequencing single-end short-read datasets are available under BioProject PRJNA321963. The Run Accession IDs for both sets were obtained with NCBI's RunSelector and the raw data was downloaded with the GetSeqENA tool (https://github.com/B-UMMI/getSeqENA).

The DEN-IM workflow was executed with the raw sequencing data with default parameters and resources in the same HPC cluster as the shotgun metagenomics dataset.

4.12.5 Non-DENV Arbovirus Data

The accession numbers for the 132 samples, belonging to zika virus (ZKV), chikungunya virus (CHIKV) and yellow fever virus (YFV) amplicon and metagenomic datasets are available as supplemental material (Table S4). As with the amplicon sequencing dataset, the list of Run Accession IDs was obtained with NCBI's RunSelector and the raw data was downloaded with the GetSeqENA tool (`https://github.com/B-UMMI/getSeqENA`).

The DEN-IM workflow was executed with default parameters and resources in the same HPC cluster as the amplicon and shotgun metagenomics datasets.

4.12.6 Supplemental Tables

Table 4.2: Collection date, serotype confirmation and run accession identifier for the metagenomic sequencing dataset.

Sample	**Collection Date**	**Source**	**Serotype (qPCR)**	**Serotype**	**Genotype**	**Run Accession**
91-0104	21/9/2015	plasma	2	2	III(AsianAmerican)	SRR8842525
91-0105	22/9/2015	plasma	2	2	III(AsianAmerican)	SRR7252349
91-0115	30/9/2015	plasma	3	-	-	SRR7252368
91-0118	5/10/2015	plasma	3	3	III	SRR7252362
91-0132	19/10/2015	plasma	1	1	V	SRR8883926
91-0135	27/10/2015	plasma	2	2	III(AsianAmerican)	SRR9004764
92-1001	2/10/2015	plasma	1	-	-	SRR7252337
92-1094	16/10/2015	plasma	2	2	III(AsianAmerican)	SRR8842524
CC0007	31/8/2010	serum	2	2	III(AsianAmerican)	SRR7252354
CC0009	31/8/2010	serum	3	3	III	SRR8842527
CC0010	27/8/2010	serum	3	3	III	SRR7252358
CC0011	27/8/2010	serum	3	3	III	SRR8842526
CC0030a	1/9/2010	serum	4	4	II	SRR7252356
CC0030b	1/9/2010	serum	4	4	II	SRR7252355
CC0031	2/9/2010	serum	2	2	III(AsianAmerican)	SRR8842521
CC0061	20/1/2011	serum	4	4	II	SRR8842520
CC0066	11/10/2011	serum	4	4	II	SRR8842523
CC0067	18/10/2011	serum	4	4	II	SRR8842522
CC0116	29/3/2012	serum	4	4	II	SRR8842519
CC0150	9/5/2012	serum	2	2	III(AsianAmerican)	SRR8842518
CC0186	17/7/2012	serum	4	4	II	SRR9004763
UCUG0186	30/8/2010	serum	4	4	II	SRR8842528
Negative Control	-	-	-	-	-	SRR8842530
Positive Control	-	-	2	2	V(AsianI)	SRR8886136
Spiked sample	-	-	1,2,3,4	1,2,3,4	V,III(Asian American),III,II	SRR8842529

Table 4.3: Run accession ID, BioProject SRA Study ID, source and organism present for each sample of the negative control dataset (ZKV – zika virus, CHIKV – chikungunya virus, YFV – yellow fever virus).

Run ID	Bioproject	SRA Study	Source	Organism
SRR8031152	PRJNA494391	SRP163225	Shotgun Metagenomic	ZKV
SRR8062732	PRJNA494391	SRP163225	Shotgun Metagenomic	ZKV
SRR8031153	PRJNA494391	SRP163225	Shotgun Metagenomic	ZKV
SRR8063606	PRJNA494391	SRP163225	Shotgun Metagenomic	ZKV
SRR8063603	PRJNA494391	SRP163225	Shotgun Metagenomic	ZKV
SRR8063605	PRJNA494391	SRP163225	Shotgun Metagenomic	ZKV
SRR8031155	PRJNA494391	SRP163225	Shotgun Metagenomic	ZKV
SRR8031154	PRJNA494391	SRP163225	Shotgun Metagenomic	ZKV
SRR8063604	PRJNA494391	SRP163225	Shotgun Metagenomic	ZKV
SRR8062733	PRJNA494391	SRP163225	Shotgun Metagenomic	ZKV
SRR7985391	PRJNA494391	SRP163225	Shotgun Metagenomic	ZKV
SRR7985394	PRJNA494391	SRP163225	Shotgun Metagenomic	ZKV
SRR7985620	PRJNA494391	SRP163225	Shotgun Metagenomic	CHIKV
SRR7985390	PRJNA494391	SRP163225	Shotgun Metagenomic	ZKV
SRR7985392	PRJNA494391	SRP163225	Shotgun Metagenomic	ZKV
SRR7985621	PRJNA494391	SRP163225	Shotgun Metagenomic	CHIKV
SRR5179639	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179637	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179646	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR7985389	PRJNA494391	SRP163225	Shotgun Metagenomic	ZKV
SRR7985622	PRJNA494391	SRP163225	Shotgun Metagenomic	CHIKV
SRR7985619	PRJNA494391	SRP163225	Shotgun Metagenomic	CHIKV
SRR5179667	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179653	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR7985393	PRJNA494391	SRP163225	Shotgun Metagenomic	ZKV
SRR5179638	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179636	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179666	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179650	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179649	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179643	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179635	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179645	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179642	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179644	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179647	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179641	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179640	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179652	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179648	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR5179651	PRJNA361543	SRP096859	Amplicon Metagenomics	YFV
SRR9020503	PRJNA541092	SRP195668	Amplicon Metagenomics	CHIKV
SRR9020505	PRJNA541093	SRP195669	Amplicon Metagenomics	CHIKV
SRR9020506	PRJNA541094	SRP195670	Amplicon Metagenomics	CHIKV
SRR9020509	PRJNA541095	SRP195671	Amplicon Metagenomics	CHIKV
SRR9020511	PRJNA541096	SRP195672	Amplicon Metagenomics	CHIKV
SRR9020513	PRJNA541097	SRP195673	Amplicon Metagenomics	CHIKV
SRR9020514	PRJNA541098	SRP195674	Amplicon Metagenomics	CHIKV
SRR9020516	PRJNA541099	SRP195675	Amplicon Metagenomics	CHIKV
SRR9020518	PRJNA541100	SRP195676	Amplicon Metagenomics	CHIKV
SRR9020520	PRJNA541101	SRP195677	Amplicon Metagenomics	CHIKV
SRR9020521	PRJNA541102	SRP195678	Amplicon Metagenomics	CHIKV
SRR9020523	PRJNA541103	SRP195679	Amplicon Metagenomics	CHIKV
SRR9020525	PRJNA541104	SRP195680	Amplicon Metagenomics	CHIKV
SRR9020527	PRJNA541105	SRP195681	Amplicon Metagenomics	CHIKV
SRR9020529	PRJNA541106	SRP195682	Amplicon Metagenomics	CHIKV
SRR9020530	PRJNA541107	SRP195683	Amplicon Metagenomics	CHIKV
SRR9020532	PRJNA541108	SRP195684	Amplicon Metagenomics	CHIKV

Continue on next page

Table4.3- *Continued from previous page*

Run ID	**Bioproject**	**SRA Study**	**Source**	**Organism**
SRR9020534	PRJNA541109	SRP195685	Amplicon Metagenomics	CHIKV
SRR9020537	PRJNA541110	SRP195686	Amplicon Metagenomics	CHIKV
SRR9020539	PRJNA541111	SRP195687	Amplicon Metagenomics	CHIKV
SRR9020541	PRJNA541112	SRP195688	Amplicon Metagenomics	CHIKV
SRR9020542	PRJNA541113	SRP195689	Amplicon Metagenomics	CHIKV
SRR9020504	PRJNA541114	SRP195690	Amplicon Metagenomics	CHIKV
SRR9020507	PRJNA541115	SRP195691	Amplicon Metagenomics	CHIKV
SRR9020508	PRJNA541116	SRP195692	Amplicon Metagenomics	CHIKV
SRR9020510	PRJNA541117	SRP195693	Amplicon Metagenomics	CHIKV
SRR9020512	PRJNA541118	SRP195694	Amplicon Metagenomics	CHIKV
SRR9020515	PRJNA541119	SRP195695	Amplicon Metagenomics	CHIKV
SRR9020517	PRJNA541120	SRP195696	Amplicon Metagenomics	CHIKV
SRR9020519	PRJNA541121	SRP195697	Amplicon Metagenomics	CHIKV
SRR9020522	PRJNA541122	SRP195698	Amplicon Metagenomics	CHIKV
SRR9020524	PRJNA541123	SRP195699	Amplicon Metagenomics	CHIKV
SRR9020526	PRJNA541124	SRP195700	Amplicon Metagenomics	CHIKV
SRR9020528	PRJNA541125	SRP195701	Amplicon Metagenomics	CHIKV
SRR9020531	PRJNA541126	SRP195702	Amplicon Metagenomics	CHIKV
SRR9020533	PRJNA541127	SRP195703	Amplicon Metagenomics	CHIKV
SRR9020535	PRJNA541128	SRP195704	Amplicon Metagenomics	CHIKV
SRR9020536	PRJNA541129	SRP195705	Amplicon Metagenomics	CHIKV
SRR9020538	PRJNA541130	SRP195706	Amplicon Metagenomics	CHIKV
SRR9020540	PRJNA541131	SRP195707	Amplicon Metagenomics	CHIKV
SRR7369225	PRJNA47661	SRP150883	Shotgun Metagenomic	ZKV
SRR7369226	PRJNA47661	SRP150883	Shotgun Metagenomic	ZKV
SRR6505781	PRJNA431343	SRP131290	Shotgun Metagenomic	ZKV
SRR8260975	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8260976	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8260977	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8260978	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8260979	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8260980	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261322	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261325	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261326	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261329	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261330	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261331	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261332	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261333	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261335	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261336	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261338	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261341	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261342	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261343	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261345	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261346	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261347	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261348	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261352	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261353	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261354	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261355	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261356	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261359	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261360	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261361	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261362	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261364	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV

Continue on next page

Table4.3- *Continued from previous page*

Run ID	Bioproject	SRA Study	Source	Organism
SRR8261365	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261366	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261367	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261369	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261402	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261404	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261407	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261411	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261412	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261413	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261415	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261416	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV
SRR8261417	PRJNA314889	SRP162155	Shotgun Metagenomic	ZKV

Table 4.4: Number of raw base pairs, overall alignment rate against the DENV mapping database, estimated coverage depths and serotype and genotype for 25 shotgun metagenomics sequencing samples.

Sample	Raw Megabases	% DENV Reads	Estimated coverage depth (times)	Serotype	Genotype
91-0104	2193.71	12.46	5944.67	2	III (AsianAmerican)
91-0105	191.37	4.01	495.97	2	III (AsianAmerican)
91-0115	179.24	0.05	3.74	-	-
91-0118	195.27	1.69	86.53	3	III
91-0132	378.21	20.02	4698.12	1	V
91-0135	91.71	21.45	1287.52	2	III (AsianAmerican)
92-1001 a)	163.44	-	-	-	-
92-1094	1197.92	8.48	4032.21	2	III (AsianAmerican)
CC0007	252.97	3.79	383.77	2	III (AsianAmerican)
CC0009	2055.13	9.48	8226.27	3	III
CC0010	368.64	5.68	1197.58	3	III
CC0011	924.69	8.38	3016.17	3	III
CC0030a	261.12	52.52	2914.87	4	II
CC0030b	399.04	10.51	677.96	4	II
CC0031	1572.1	68.91	52318.33	2	III (AsianAmerican)
CC0061	1262.83	8.97	5120.4	4	II
CC0066	1087.45	2.8	569.7	4	II
CC0067	1022.06	5.55	2548.84	4	II
CC0116	773.31	6.72	2313.99	4	II
CC0150	1403.69	17.41	12065.81	2	III (AsianAmerican)
CC0186	671.78	0.03	14.71	4	II
UCUG0186 b)	1116.67	0.01	5.65	-	-
Negative Control a)	163.67	-	-	-	-
Positive Control	443.93	85.38	19362.07	2	V (Asian I)
				3	III
Spike	1518.93	41.7	22289.98	1	V
				2	III (AsianAmerican)
				4	II

a) Failed quality control - No sequence data after quality trimming.

b) Failed quality control - Low sequence depth (<10x).

Table 4.5: Number of raw base pairs, overall alignment rate, in percentage, for the mapping against the DENV database, number of ORFs recovered, and respective serotype and genotype for 106 paired-end amplicon sequencing samples.

Sample	Raw Megabases	% DENV DNA	CDS Assembly	Serotype	Genotype
SRR5821157	439.35	82.56	consensus	3	III
SRR5821158	77.34	85.19	consensus	3	III
SRR5821159	68.00	91.11	consensus	3	III
SRR5821160	119.54	97.77	consensus	3	III
SRR5821161	53.40	92.76	consensus	3	III
SRR5821162	49.59	99.39	consensus	3	III
SRR5821163	66.43	97.78	consensus	3	III
SRR5821164	69.96	99.18	consensus	3	III
SRR5821165	75.48	98.38	consensus	3	III
SRR5821166	38.99	62.03	de novo	3	III
SRR5821167	73.15	49.19	de novo	3	III
SRR5821168	49.59	99.63	consensus	3	III
SRR5821169	119.39	99.74	de novo	3	III
SRR5821170	61.45	99.09	consensus	3	III
SRR5821171	61.63	98.92	consensus	3	III
SRR5821172	69.86	98.96	de novo	3	III
SRR5821173	80.37	97.59	de novo	3	III
SRR5821174	37.58	76.69	de novo	3	III
SRR5821175	112.70	75.55	de novo	3	III
SRR5821176	139.34	99.03	de novo	3	III
SRR5821177	41.19	44.56	de novo	3	III
SRR5821178	59.03	81.06	de novo	3	III
SRR5821179	95.59	84.7	de novo	3	III
SRR5821180	48.75	98.15	consensus	3	III
SRR5821181	64.45	99.3	consensus	3	III
SRR5821182	64.40	98.88	consensus	3	III
SRR5821183	115.14	95.61	consensus	3	III
SRR5821184	170.72	94.11	de novo	3	III
SRR5821185	181.75	98.19	de novo	3	III
SRR5821186	246.98	96.4	de novo	3	III
SRR5821187	55.62	99.74	consensus	3	III
SRR5821188	70.95	99.39	consensus	3	III
SRR5821189	82.61	99.27	de novo	3	III
SRR5821190	138.58	98.81	consensus	3	III
SRR5821191	59.92	99.72	de novo	3	III
SRR5821192	40.53	36.88	consensus	3	III
SRR5821193	92.08	98.9	de novo	3	III
SRR5821194	58.69	98.53	consensus	3	III
SRR5821195	127.80	99.64	consensus	3	III
SRR5821196	59.30	86.62	de novo	3	III
SRR5821197	87.78	99.47	de novo	3	III
SRR5821198	185.55	99.72	de novo	3	III
SRR5821199	83.55	99.62	consensus	3	III
SRR5821200	85.52	99.5	consensus	3	III
SRR5821201	129.77	94.6	consensus	3	III
SRR5821202	56.60	99.81	consensus	3	III
SRR5821203	80.28	99.22	consensus	3	III
SRR5821204	68.46	95.52	de novo	3	III
SRR5821205	44.45	98.53	consensus	3	III
SRR5821206	43.67	97.88	consensus	3	III
SRR5821207	78.93	99.22	de novo	3	III
SRR5821208	87.45	97.72	consensus	3	III
SRR5821209	73.40	94.16	de novo	3	III
SRR5821210	55.86	91.35	de novo	3	III
SRR5821211	75.53	85.6	consensus	3	III
SRR5821212	98.89	99.09	de novo	3	III
SRR5821213	84.85	95.03	de novo	3	III

Continue on next page

Table4.5- *Continued from previous page*

Sample	**Raw Megabases**	**% DENV DNA**	**CDS Assembly**	**Serotype**	**Genotype**
SRR5821214	15.33	96.28	de novo	3	III
SRR5821215	13.08	96.74	consensus	3	III
SRR5821216	45.07	98.85	de novo	3	III
SRR5821217	161.65	88.94	consensus	3	III
SRR5821218	51.09	95.29	consensus	3	III
SRR5821219	84.68	99.1	de novo	3	III
SRR5821220	88.26	82.64	de novo	3	III
SRR5821221	64.76	86.62	de novo	3	III
SRR5821222	93.47	97.48	consensus	3	III
SRR5821223	86.50	98.99	de novo	3	III
SRR5821224	73.31	26.43	consensus	3	III
SRR5821225	68.85	98.43	consensus	3	III
SRR5821226	67.75	96.67	consensus	3	III
SRR5821227	32.56	99.54	de novo	3	III
SRR5821228	38.73	86.68	consensus	3	III
SRR5821229	77.18	99.69	consensus	3	III
SRR5821230	175.73	99.58	de novo	3	III
SRR5821231	100.82	99.58	de novo	3	III
SRR5821232	86.89	99.47	consensus	3	III
SRR5821233	270.15	99.56	consensus	3	III
SRR5821234	76.07	99.75	consensus	3	III
SRR5821235	32.78	79.78	consensus	3	III
SRR5821236	80.19	24.72	de novo	3	III
SRR5821237	50.59	97.38	consensus	3	III
SRR5821238	63.56	97.63	de novo	3	III
SRR5821239	29.66	41.15	consensus	3	III
SRR5821240	62.61	94.64	de novo	3	III
SRR5821241	17.52	98.03	consensus	3	III
SRR5821242	58.86	99.25	consensus	3	III
SRR5821243	50.08	93.56	consensus	3	III
SRR5821244	32.67	99.09	consensus	3	III
SRR5821245	64.96	99.77	consensus	3	III
SRR5821246	104.11	90.14	consensus	3	III
SRR5821247	98.64	99.73	consensus	3	III
SRR5821248	129.28	90.73	consensus	3	III
SRR5821249	45.76	93.13	de novo	3	III
SRR5821250	72.54	98.88	de novo	3	III
SRR5821251	115.85	97.7	consensus	3	III
SRR5821252	60.76	94	consensus	3	III
SRR5821253	64.45	99.66	consensus	3	III
SRR5821254	0.27	98.12	consensus	3	III
SRR5821255	62.53	99.55	de novo	3	III
SRR5821256	54.57	99.58	consensus	3	III
SRR5821257	34.90	99.53	de novo	3	III
SRR5821258	68.64	99.6	consensus	3	III
SRR5821259	73.04	98.8	consensus	3	III
SRR5821260	54.60	99.14	consensus	3	III
SRR5821261	55.54	95.5	de novo	3	III
SRR5821262	106.05	91.78	consensus	3	III

Table 4.6: Taxonomic profiling results for the amplicon sequencing samples with less than 70% DENV DNA.

Sample	Bowtie2	Kraken2 (minikraken2_v2 DB)		
	DENV (%)	Unclassified	Homo sapiens	DENV (%)
SRR5821236	24.72	5.47	71.61	19.63
SRR5821224	26.43	7.01	71.06	19.58
SRR5821192	36.88	8.12	61.78	28.73
SRR5821239	41.15	8.29	56.43	33.84
SRR5821167	49.19	14.79	50.16	34.38
SRR5821166	62.03	13.72	37.77	47.97

Table 4.7: Number of raw base pairs, overall alignment rate, in percentage, for the mapping against the DENV database, number of ORFs recovered, and respective serotype and genotype for 78 single-end amplicon sequencing samples.

Sample	Raw Megabases	% DENV DNA	CDS Assembly	Serotype	Genotype
SRR3539340	330365175	83.7	consensus	I	1
SRR3539341	317977866	66.56	consensus	I	1
SRR3539342	406075245	74.2	consensus	I	1
SRR3539343	302220886	59.24	de novo	I	1
SRR3539344	424801129	83.21	de novo	I	1
SRR3539345	345821429	92.58	de novo	I	1
SRR3539346	411918039	90.92	de novo	I	1
SRR3539347	411031278	90.92	de novo	I	1
SRR3539348	469139944	92.45	de novo	I	1
SRR3539349	537372466	90.77	de novo	I	1
SRR3539350	401844325	90.32	de novo	I	1
SRR3539351	401993816	89.76	de novo	I	1
SRR3539352	357846693	88.48	consensus	I	1
SRR3539353	412322289	82.94	de novo	I	1
SRR3539354	398022772	86.07	consensus	I	1
SRR3539355	388552807	92.43	consensus	I	1
SRR3539357	351745878	88.51	consensus	I	1
SRR3539358	398098393	70.74	de novo	I	1
SRR3539359	479640173	91.59	consensus	I	1
SRR3539360	374570187	75.78	de novo	I	1
SRR3539361	370202077	72.56	de novo	I	1
SRR3539362	402201658	83.22	consensus	I	1
SRR3539363	467055595	75.85	consensus	I	1
SRR3539364	312321789	65.93	de novo	I	1
SRR3539365	253871159	88.37	de novo	I	1
SRR3539366	246292055	82.12	consensus	I	1
SRR3539367	228721211	86.98	de novo	I	1
SRR3539368	253255975	89.84	de novo	I	1
SRR3539369	254904463	91.39	de novo	I	1
SRR3539370	256094646	89.77	de novo	I	1
SRR3539371	266981417	93.77	de novo	I	1
SRR3539372	195098066	82.5	consensus	I	1
SRR3539373	237636237	84.54	consensus	I	1
SRR3539374	202624880	91.98	de novo	I	1
SRR3539375	399641302	87.95	consensus	I	1
SRR3539376	209424800	92.58	de novo	I	1
SRR3539377	278160288	89.75	de novo	I	1
SRR3539378	328706147	87.33	de novo	I	1
SRR3539379	370640534	88.93	de novo	I	1
SRR3539380	313475971	66.56	de novo	I	1

Continue on next page

Table4.7- *Continued from previous page*

Sample	**Raw Megabases**	**% DENV DNA**	**CDS Assembly**	**Serotype**	**Genotype**
SRR3539381	327213068	89.39	de novo	I	1
SRR3539382	295317021	78.49	de novo	I	1
SRR3539383	335941236	81.98	consensus	I	1
SRR3539384	383785104	90.79	de novo	I	1
SRR3539385	330006204	88.86	de novo	I	1
SRR3539386	454412182	87.3	de novo	I	1
SRR3539387	321847824	92.31	de novo	I	1
SRR3539388	354652844	92.31	de novo	I	1
SRR3539389	345354321	88.38	consensus	I	1
SRR3539390	412365081	84.83	de novo	I	1
SRR3539391	374367976	84.5	de novo	I	1
SRR3539393	428999734	82.5	de novo	I	1
SRR3539394	323218873	91.91	de novo	I	1
SRR3539395	375283202	91.7	de novo	I	1
SRR3539396	434756338	84.96	de novo	I	1
SRR3539397	361928373	93.07	de novo	I	1
SRR3539398	462599218	80.7	consensus	I	1
SRR3539399	379115053	86.74	de novo	I	1
SRR3539400	404747525	93.34	consensus	I	1
SRR3539401	327849624	94.64	consensus	I	1
SRR3539406	209992112	78.25	de novo	I	1
SRR3539407	370290249	91.86	consensus	I	1
SRR3539408	191269315	95.58	de novo	I	1
SRR3539409	398058055	91.69	de novo	I	1
SRR3539410	393229460	94.48	de novo	I	1
SRR3539411	387469496	93.21	consensus	I	1
SRR3539412	53752250	82.32	de novo	I	1
SRR3539413	347547808	85.47	de novo	I	1
SRR3539414	355980530	84.16	consensus	I	1
SRR3539415	364109410	92.25	consensus	I	1
SRR3539416	341121914	86.11	consensus	I	1
SRR3539417	339098553	84.5	de novo	I	1
SRR3539418	332640627	85.66	de novo	I	1
SRR3539419	360466242	85.65	de novo	I	1
SRR3539420	415554748	84.23	de novo	I	1
SRR3539421	322411348	93.42	de novo	I	1
SRR3539422	387614239	82.17	consensus	I	1
SRR3539424	446656613	84.25	de novo	I	1

Table 4.8: Representative sequences of serotype 1 diversity in the Dengue Virus Typing Database.

Sample	**ViPR Classification**	**Origin**	**Collection Year**
EU482591	DENV-1 V	USA	2006
KU509254	DENV-1 V	Venezuela	2011
MF004384	DENV-1 V	France	2014
GU131956	DENV-1 V	Mexico	2006
AF311956	DENV-1 V	Brazil	1997
FJ205874	DENV-1 V	USA	1995
FJ478457	DENV-1 V	USA	1996
EU482567	DENV-1 V	USA	1998
DQ285559	DENV-1 V	Reunion	2004
JN903578	DENV-1 V	India	2007
KP188548	DENV-1 V	Brazil	2013
JQ922544	DENV-1 V	India	1963
KX380796	DENV-1 V	Singapore	2012

Continue on next page

Table4.8- *Continued from previous page*

Sample	ViPR Classification	Origin	Collection Year
JQ922548	DENV-1 V	India	2005
KP406801	DENV-1 V	South Korea	2004
DQ285562	DENV-1 V	Comoros	1993
JQ922546	DENV-1 V	India	1971
EF457905	DENV-1 III	Malaysia	1972
AF180818	DENV-1 II	Unknown	Unknown
JQ922547	DENV-1 II	Thailand	1960
KY496855	DENV-1 IV	Taiwan	2016
LC128301	DENV-1 IV	Philippines	2016
KX951689	DENV-1 IV	Taiwan	2004
KC762653	DENV-1 IV	Indonesia	2008
KU509261	DENV-1 IV	Indonesia	2010
AB189121	DENV-1 IV	Indonesia	1998
KC762620	DENV-1 IV	Indonesia	2007
EU863650	DENV-1 IV	Chile	2002
AB195673	DENV-1 IV	Japan	2003
AB204803	DENV-1 IV	Japan	2004
JF459993	DENV-1 I	Myanmar	2002
KT827371	DENV-1 I	China	2014
KX620454	DENV-1 I	China	2014
FJ639670	DENV-1 I	Cambodia	2001
KU509250	DENV-1 I	Thailand	2012
KJ755855	DENV-1 I	India	2013
GU131678	DENV-1 I	Viet Nam	2008
KU509265	DENV-1 I	Unknown	2012
KF955446	DENV-1 I	Viet Nam	2008
JF937615	DENV-1 I	Viet Nam	2008
FJ639678	DENV-1 I	Cambodia	2003
EU660395	DENV-1 I	Viet Nam	2007
AB608789	DENV-1 I	Taiwan	1994
GQ868636	DENV-1 I	Cambodia	2008
KY586539	DENV-1 I	Thailand	1995
KU509258	DENV-1 I	Eritrea	2010

Table 4.9: Representative sequences of serotype 2 diversity in the Dengue Virus Typing Database.

Sample	ViPR Classification	Origin	Collection Year
HQ705624	DENV-2 III (AsianAmerican)	Nicaragua	2009
KY977454	DENV-2 III (AsianAmerican)	Panama	2011
KY474330	DENV-2 III (AsianAmerican)	Ecuador	2014
FJ024473	DENV-2 III (AsianAmerican)	Colombia	2005
JX669476	DENV-2 III (AsianAmerican)	Brazil	2010
JN819419	DENV-2 III (AsianAmerican)	Brazil	2000
KF955364	DENV-2 III (AsianAmerican)	Puerto Rico	2006
JX669480	DENV-2 III (AsianAmerican)	Brazil	1995
FJ639699	DENV-2 III (AsianAmerican)	Cambodia	2002
EU482449	DENV-2 III (AsianAmerican)	Viet Nam	2006
EU482778	DENV-2 III (AsianAmerican)	Viet Nam	2003
KY586692	DENV-2 V (AsianI)	Thailand	2001
KY586679	DENV-2 V (AsianI)	Thailand	2001
KY586571	DENV-2 V (AsianI)	Thailand	2006
KY586572	DENV-2 V (AsianI)	Thailand	2006
EU726767	DENV-2 V (AsianI)	Thailand	1994
GQ868591	DENV-2 V (AsianI)	Thailand	1964
KF704356	DENV-2 IV (AsianII)	Cuba	1981

Continue on next page

Table4.9- *Continued from previous page*

Sample	**ViPR Classification**	**Origin**	**Collection Year**
JQ922552	DENV-2 I (American)	India	1960
KJ918750	DENV-2 I (American)	India	2007
JQ922553	DENV-2 I (American)	India	1980
GQ868592	DENV-2 I (American)	Colombia	1986
JX966379	DENV-2 I (American)	Mexico	1994
GQ398257	DENV-2 I (American)	Indonesia	1977
KY923048	DENV-2 VI (Sylvatic)	Malaysia	2015
JF260983	DENV-2 VI (Sylvatic)	Spain	2009
KY937189	DENV-2 II (Cosmopolitan)	China	2015
KY937188	DENV-2 II (Cosmopolitan)	China	2015
KY937187	DENV-2 II (Cosmopolitan)	China	2015
JQ955624	DENV-2 II (Cosmopolitan)	India	2011
KU509271	DENV-2 II (Cosmopolitan)	India	2006
KF041232	DENV-2 II (Cosmopolitan)	Pakistan	2011
JQ922551	DENV-2 II (Cosmopolitan)	India	2005
JX475906	DENV-2 II (Cosmopolitan)	India	2009
MG779194	DENV-2 II (Cosmopolitan)	Kenya	2017
FJ882602	DENV-2 II (Cosmopolitan)	Sri Lanka	1996
EU056810	DENV-2 II (Cosmopolitan)	Burkina Faso	1983
KY627763	DENV-2 II (Cosmopolitan)	Burkina Faso	2016
KM279515	DENV-2 II (Cosmopolitan)	Singapore	2011
KX452015	DENV-2 II (Cosmopolitan)	Malaysia	2014
KC762662	DENV-2 II (Cosmopolitan)	Indonesia	2007
KU509270	DENV-2 II (Cosmopolitan)	Unknown	2012
KP012546	DENV-2 II (Cosmopolitan)	China	2014
KX452034	DENV-2 II (Cosmopolitan)	Malaysia	2014
KX452048	DENV-2 II (Cosmopolitan)	Malaysia	2014
KX452044	DENV-2 II (Cosmopolitan)	Malaysia	2014
HM488257	DENV-2 II (Cosmopolitan)	Guam	2001
KU509277	DENV-2 II (Cosmopolitan)	Philippines	2010
KU509269	DENV-2 II (Cosmopolitan)	Philippines	2009
KU509274	DENV-2 II (Cosmopolitan)	Philippines	2010
GQ398263	DENV-2 II (Cosmopolitan)	Indonesia	1975

Table 4.10: Representative sequences of serotype 3 diversity in the Dengue Virus Typing Database.

Sample	**ViPR Classification**	**Origin**	**Collection Year**
KF954946	DENV-3-III	China	2013
JQ922557	DENV-3 III	India	2005
KU509286	DENV-3 III	India	2011
EU687233	DENV-3 III	USA	2002
GQ252674	DENV-3 III	Sri Lanka	1997
FJ882573	DENV-3 III	Sri Lanka	1993
GQ199887	DENV-3 III	Sri Lanka	1983
JQ922555	DENV-3 III	India	1966
HM631854	DENV-3 II	Cambodia	2008
KY586703	DENV-3 II	Thailand	2006
KU509280	DENV-3 II	Thailand	2011
FJ744730	DENV-3 II	Thailand	2001
KY586814	DENV-3 II	Thailand	2006
DQ863638	DENV-3 II	Thailand	1973
KC762684	DENV-3 I	Indonesia	2007
KY863456	DENV-3 I	Indonesia	2016
KC762691	DENV-3 I	Indonesia	2008
KC762692	DENV-3 I	Indonesia	2010

Continue on next page

Table4.10- *Continued from previous page*

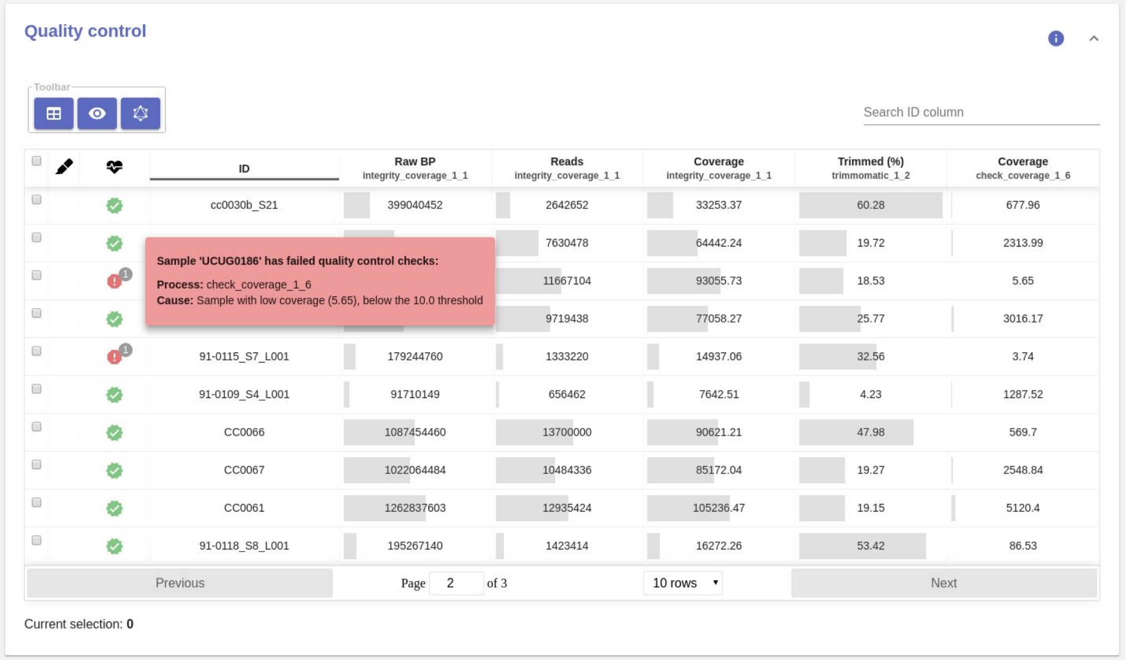
Sample	ViPR Classification	Origin	Collection Year
KY794787	DENV-3 I	Papua New Guinea	2007
MF004386	DENV-3 I	Malaysia	2012
AB189128	DENV-3 I	Indonesia	1998
KU509279	DENV-3 I	Philippines	2008
FJ898455	DENV-3 I	Cook Islands	1991
KU725666	DENV-3 V	Unkown	Unknown

Table 4.11: Representative sequences of serotype 4 diversity in the Dengue Virus Typing Database.

Sample	ViPR Classification	Origin	Collection Year
MG601754	DENV-4 I	China	2013
KY586839	DENV-4 I	Thailand	1995
KT026308	DENV-4 I	Thailand	2011
JN638572	DENV-4 I	Cambodia	2008
KY586942	DENV-4 I	Thailand	2006
KP792537	DENV-4 I	Singapore	2011
MG272273	DENV-4 I	India	2016
MG272272	DENV-4 I	India	2016
KU509287	DENV-4 I	India	2009
JQ922559	DENV-4 I	India	1979
GQ868594	DENV-4 I	Philippines	1956
JQ922558	DENV-4 I	India	1962
KU523872	DENV-4 II	Indonesia	2015
KP723482	DENV-4 II	China	2010
JX024757	DENV-4 II	Singapore	2010
KC762695	DENV-4 II	Indonesia	2007
JQ915088	DENV-4 II	New Caledonia	2009
GQ398256	DENV-4 II	Singapore	2005
KP188557	DENV-4 II	Brazil	2012
KY474335	DENV-4 II	Ecuador	2014
KT276273	DENV-4 II	Haiti	2014
KF907503	DENV-4 II	Senegal	1953
KY586945	DENV-4 III	Thailand	1998
JF262779	DENV-4 IV	Malaysia	1975

4.12.7 Supplemental Figures

a)



☐			ID	Raw BP integrity_coverage_1_1	Reads integrity_coverage_1_1	Coverage integrity_coverage_1_1	Trimmed (%) trimmomatic_1_2	Coverage check_coverage_1_6
☐		✓	cc0030b_S21	399040452	2642652	33253.37	60.28	677.96
☐		✓			7630478	64442.24	19.72	2313.99
☐		! 1			11667104	93055.73	18.53	5.65
☐		✓			9719438	77058.27	25.77	3016.17
☐		! 1	91-0115_S7_L001	179244760	1333220	14937.06	32.56	3.74
☐		✓	91-0109_S4_L001	91710149	656462	7642.51	4.23	1287.52
☐		✓	CC0066	1087454460	13700000	90621.21	47.98	569.7
☐		✓	CC0067	1022064484	10484336	85172.04	19.27	2548.84
☐		✓	CC0061	1262837603	12935424	105236.47	19.15	5120.4
☐		✓	91-0118_S8_L001	195267140	1423414	16272.26	53.42	86.53

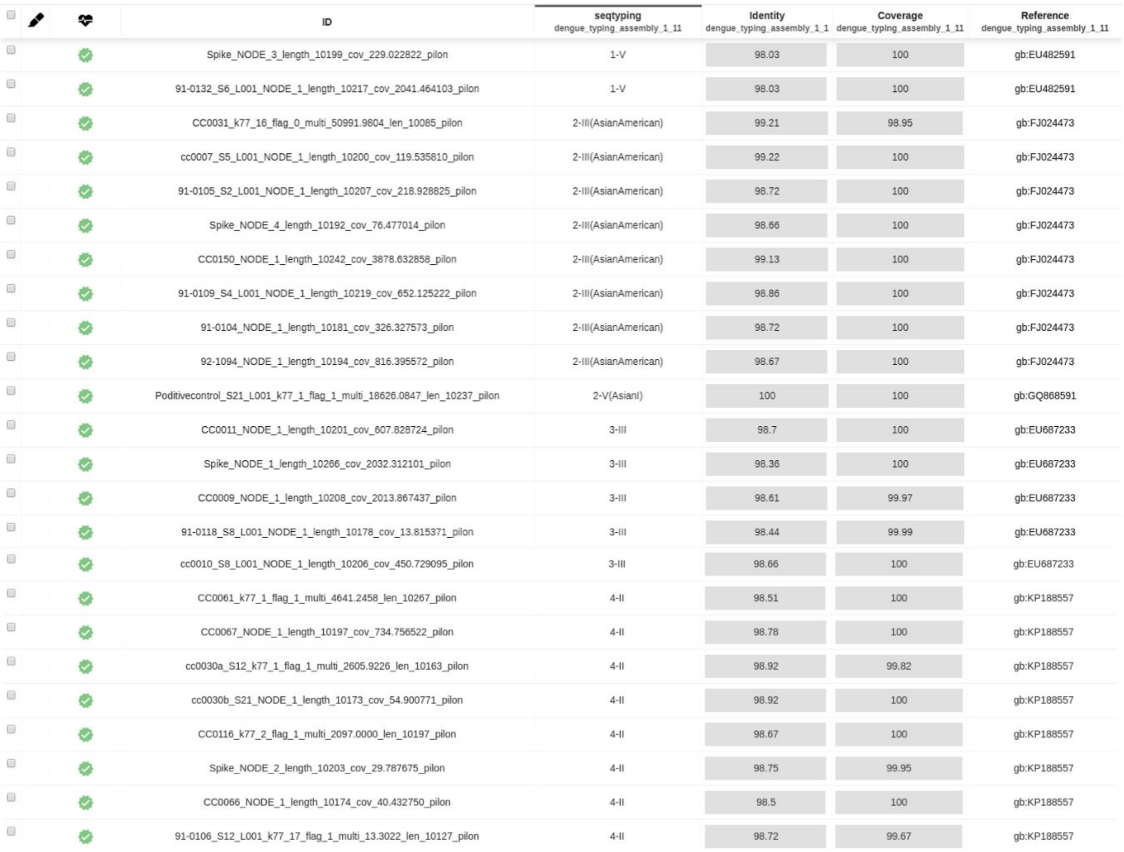
Sample 'UCUG0186' has failed quality control checks:

Process: check_coverage_1_6
Cause: Sample with low coverage (5.65), below the 10.0 threshold

Previous Page 2 of 3 10 rows Next

Current selection: **0**

b)



☐			ID	seqtyping dengue_typing_assembly_1_11	Identity dengue_typing_assembly_1_1	Coverage dengue_typing_assembly_1_11	Reference dengue_typing_assembly_1_11
☐		✓	Spike_NODE_3_length_10199_cov_229.022822_pilon	1-V	98.03	100	gb:EU482591
☐		✓	91-0132_S6_L001_NODE_1_length_10217_cov_2041.464103_pilon	1-V	98.03	100	gb:EU482591
☐		✓	CC0031_k77_16_flag_0_multi_50991.9804_len_10085_pilon	2-III(AsianAmerican)	99.21	98.95	gb:FJ024473
☐		✓	cc0007_S5_L001_NODE_1_length_10200_cov_119.535810_pilon	2-III(AsianAmerican)	99.22	100	gb:FJ024473
☐		✓	91-0105_S2_L001_NODE_1_length_10207_cov_218.928825_pilon	2-III(AsianAmerican)	98.72	100	gb:FJ024473
☐		✓	Spike_NODE_4_length_10192_cov_76.477014_pilon	2-III(AsianAmerican)	98.66	100	gb:FJ024473
☐		✓	CC0150_NODE_1_length_10242_cov_3878.632858_pilon	2-III(AsianAmerican)	99.13	100	gb:FJ024473
☐		✓	91-0109_S4_L001_NODE_1_length_10219_cov_652.125222_pilon	2-III(AsianAmerican)	98.86	100	gb:FJ024473
☐		✓	91-0104_NODE_1_length_10181_cov_326.327573_pilon	2-III(AsianAmerican)	98.72	100	gb:FJ024473
☐		✓	92-1094_NODE_1_length_10194_cov_816.395572_pilon	2-III(AsianAmerican)	98.67	100	gb:FJ024473
☐		✓	Poditivecontrol_S21_L001_k77_1_flag_1_multi_18626.0847_len_10237_pilon	2-V(AsianI)	100	100	gb:GQ868591
☐		✓	CC0011_NODE_1_length_10201_cov_607.828724_pilon	3-III	98.7	100	gb:EU687233
☐		✓	Spike_NODE_1_length_10266_cov_2032.312101_pilon	3-III	98.36	100	gb:EU687233
☐		✓	CC0009_NODE_1_length_10208_cov_2013.867437_pilon	3-III	98.61	99.97	gb:EU687233
☐		✓	91-0118_S8_L001_NODE_1_length_10178_cov_13.815371_pilon	3-III	98.44	99.99	gb:EU687233
☐		✓	cc0010_S8_L001_NODE_1_length_10206_cov_450.729095_pilon	3-III	98.66	100	gb:EU687233
☐		✓	CC0061_k77_1_flag_1_multi_4641.2458_len_10267_pilon	4-II	98.51	100	gb:KP188557
☐		✓	CC0067_NODE_1_length_10197_cov_734.756522_pilon	4-II	98.78	100	gb:KP188557
☐		✓	cc0030a_S12_k77_1_flag_1_multi_2605.9226_len_10163_pilon	4-II	98.92	99.82	gb:KP188557
☐		✓	cc0030b_S21_NODE_1_length_10173_cov_54.900771_pilon	4-II	98.92	100	gb:KP188557
☐		✓	CC0116_k77_2_flag_1_multi_2097.0000_len_10197_pilon	4-II	98.67	100	gb:KP188557
☐		✓	Spike_NODE_2_length_10203_cov_29.787675_pilon	4-II	98.75	99.95	gb:KP188557
☐		✓	CC0066_NODE_1_length_10174_cov_40.432750_pilon	4-II	98.5	100	gb:KP188557
☐		✓	91-0106_S12_L001_k77_17_flag_1_multi_13.3022_len_10127_pilon	4-II	98.72	99.67	gb:KP188557

Figure 4.5: DEN-IM report tables. a) DEN-IM's quality control report containing information of the number of base-pairs and the number of reads for the analysed samples, the estimated coverage depth before and after mapping, and the percentage of reads in the input data that were trimmed. b) DEN-IM's typing report for 24 CDSs recovered from the metagenomic dataset. The ID contains the CDS contig name, the typing result for serotype-genotype, the values for identity and coverage, and the GenBank ID of the closest reference in the Typing Database containing 161 complete DENV genomes.

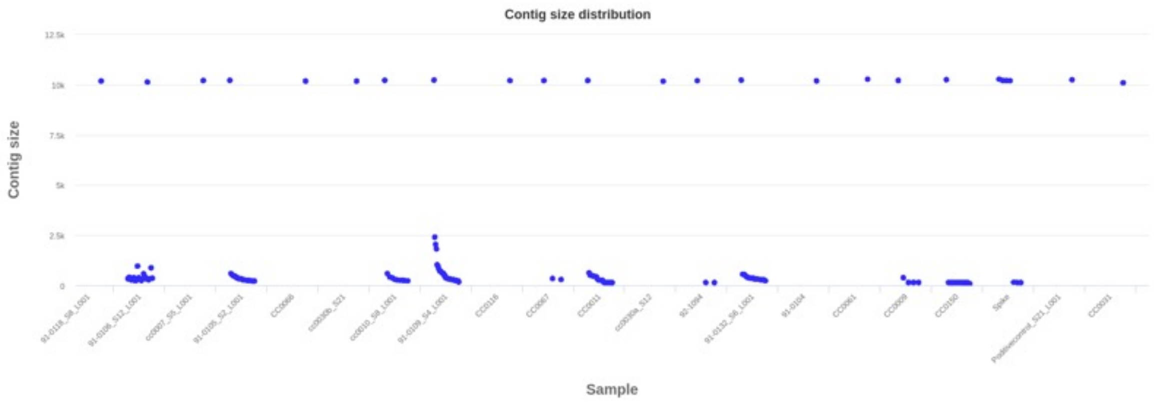


Figure 4.6: Contig size distribution for the shotgun metagenomics sequencing dataset. Each dot depicts an assembled DENV contig. Above the 10Kb are full CDS of DENV.

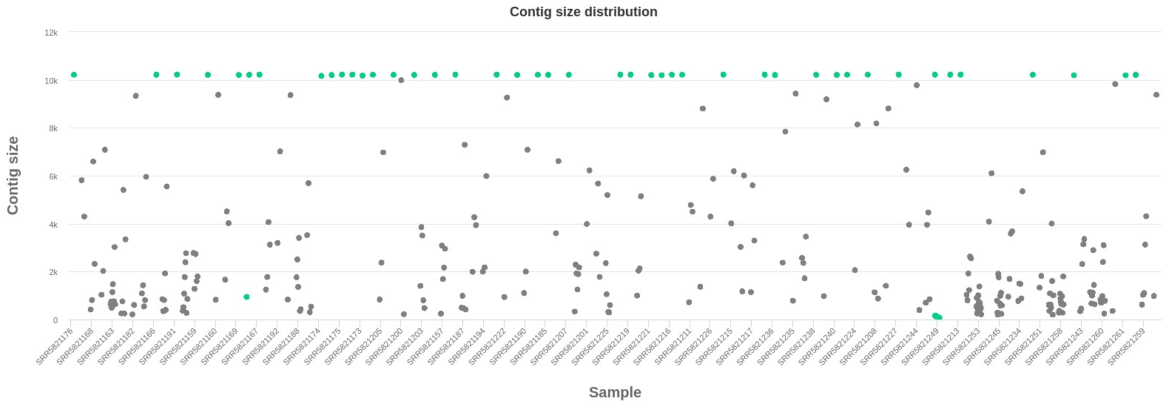


Figure 4.7: Contig size distribution of the amplicon sequencing dataset with 106 paired-end samples. Each dot depicts an assembled DENV contig. Above the 10Kb are full CDS of DENV. Contigs belonging from samples that assembled a complete DENV CDS are highlighted in green, whereas the remaining are coloured in grey.

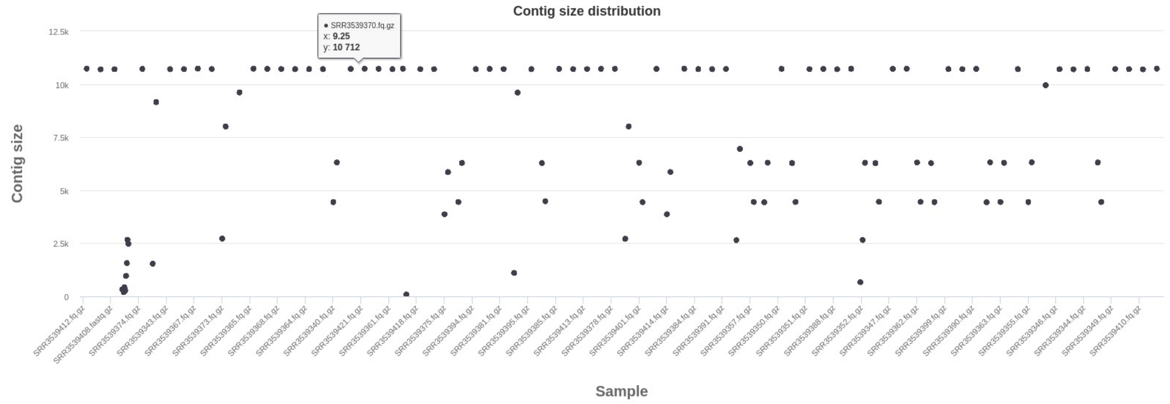


Figure 4.8: Contig size distribution of the amplicon sequencing dataset with 78 single-end samples. Each dot depicts an assembled DENV contig. Above the 10Kb are full CDS of DENV.

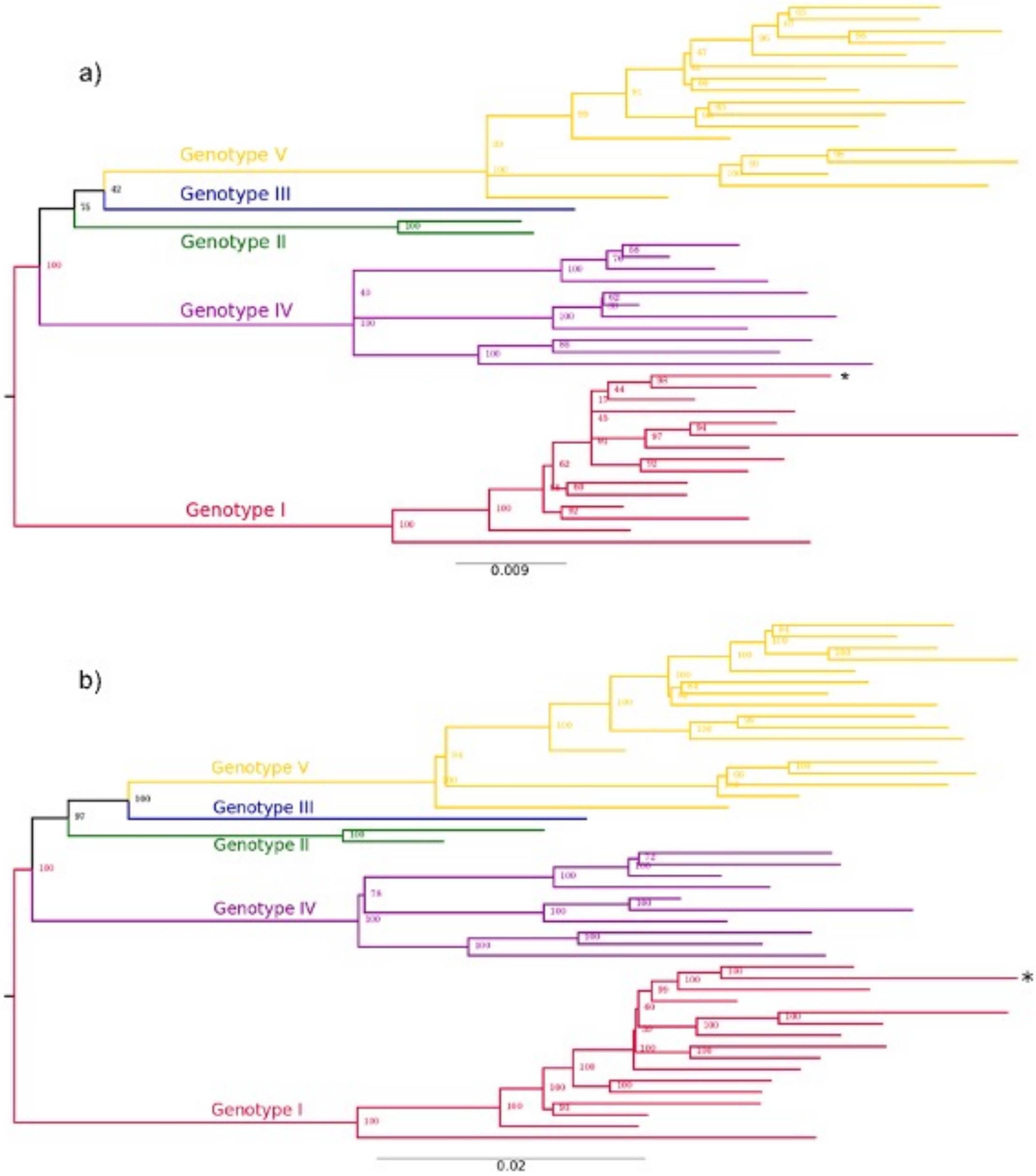


Figure 4.9: Maximum Likelihood inference of the multiple sequence alignment of the 46 DENV-1 complete genomes in the typing dataset, with a) envelope region and b) whole genome sequence. 1635 complete DENV-1 genomes were clustered at 98% nucleotide identity and the representative genomes were aligned with MAFFT. A maximum likelihood tree was inferred with RAxML. The tree is coloured according to genotype (red: genotype I; green: genotype II; blue: genotype III; purple: genotype IV). The sample JF459993, marked with a star, is currently annotated in ViPR as belonging to genotype IV but, given to the good phylogenetic support, it was re-classified as belonging to the genotype I.

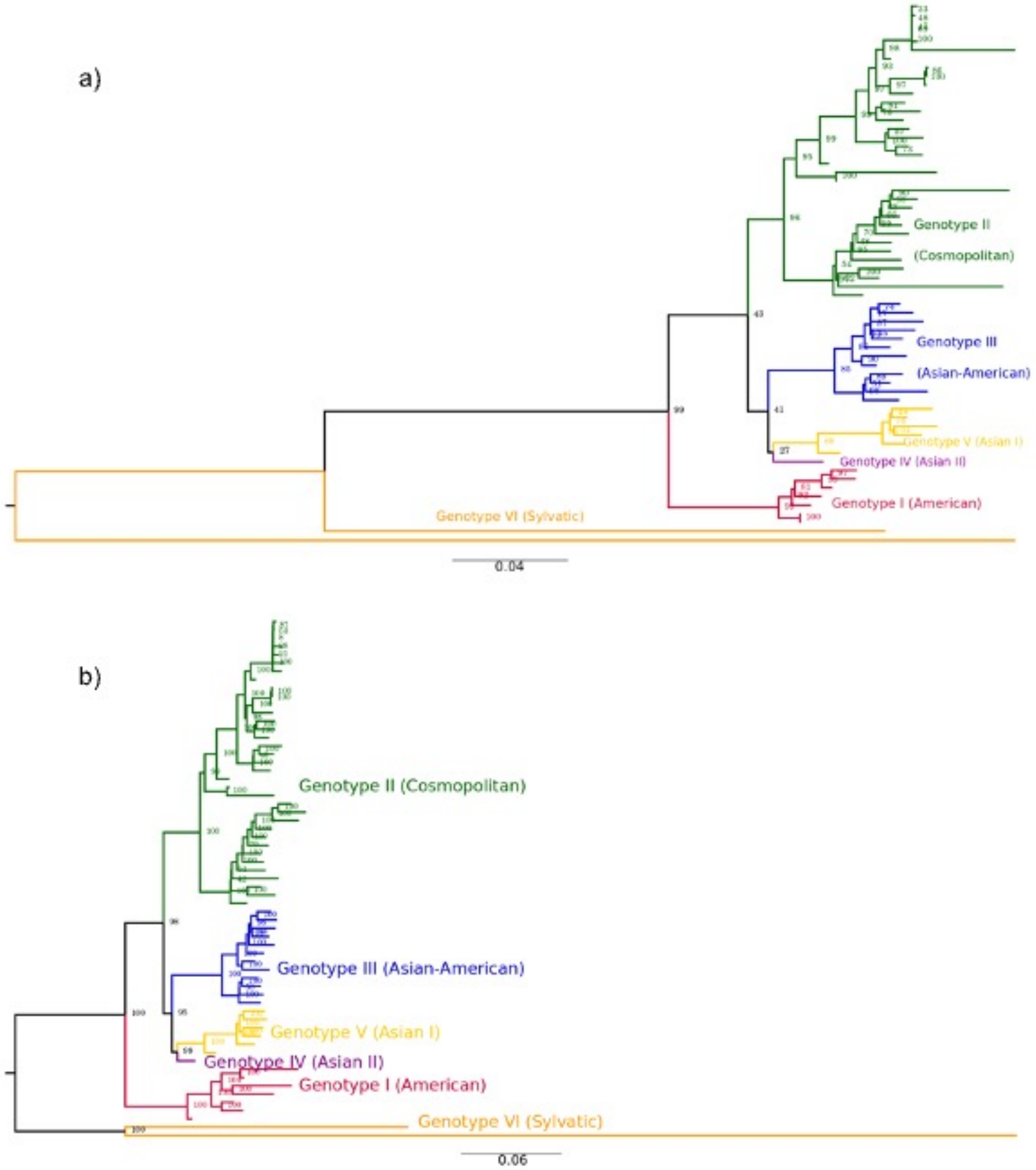


Figure 4.10: Maximum Likelihood inference of the multiple sequence alignment of the 63 DENV-2 complete genomes in the typing dataset, with a) envelope region and b) whole genome sequence. 1067 complete DENV-1 genomes were clustered at 98% nucleotide identity and the representative genomes were aligned with MAFFT. A maximum likelihood tree was inferred with RAxML. The tree is coloured according to genotype (red: genotype I; green: genotype II; blue: genotype III; purple: genotype IV).

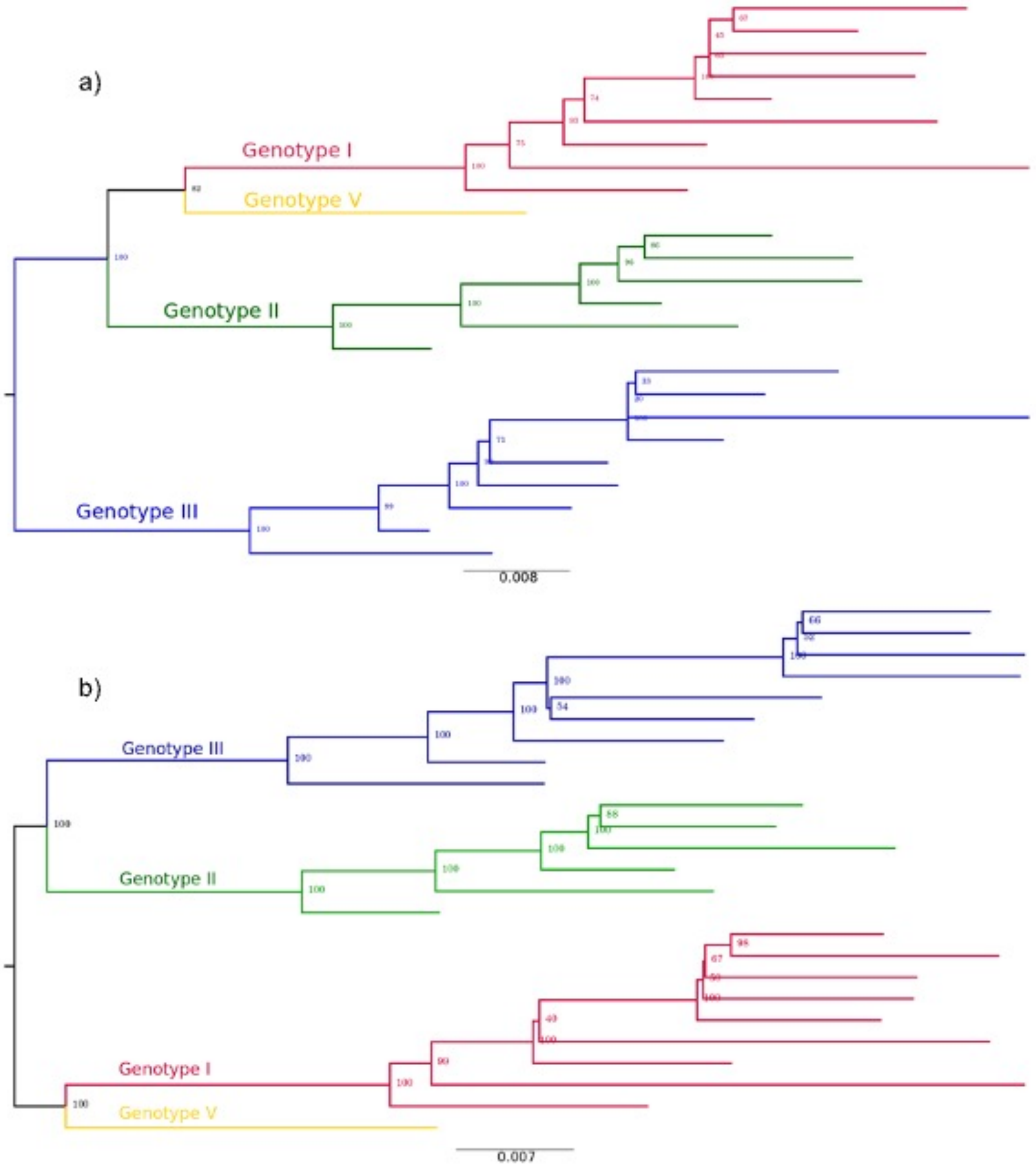


Figure 4.11: Maximum Likelihood inference of the multiple sequence alignment of the 25 DENV-3 complete genomes in the typing dataset, with a) envelope region and b) whole genome sequence. 807 complete DENV-3 genomes were clustered at 98% nucleotide identity and the representative genomes were aligned with MAFFT. A maximum likelihood tree was inferred with RAxML. The tree is coloured according to genotype (red: genotype I; green: genotype II; blue: genotype III; purple: genotype IV).

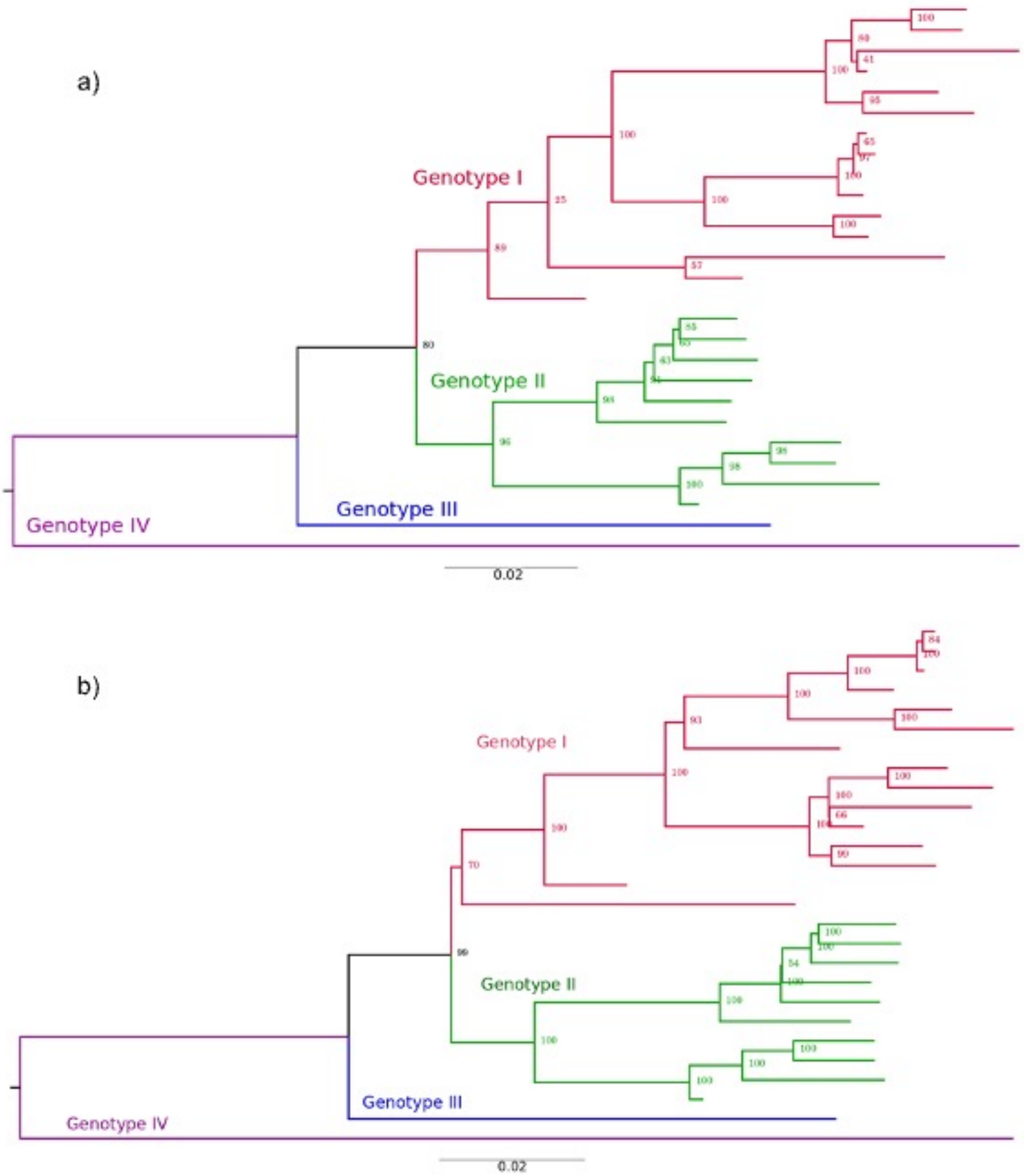


Figure 4.12: Maximum Likelihood inference of the multiple sequence alignment of the 27 DENV-4 complete genomes in the typing dataset, with a) envelope region and b) whole genome sequence. 320 complete DENV-4 genomes were clustered at 98% nucleotide identity and the representative genomes were aligned with MAFFT. A maximum likelihood tree was inferred with RAxML. The tree is coloured according to genotype (red: genotype I; green: genotype II; blue: genotype III; purple: genotype IV).

4.13 References

[1] World Health Organization. *Dengue: Guidelines for Diagnosis, Treatment, Prevention and Control*. en. Google-Books-ID: dlc0YSIyGYwC. World Health Organization, 2009. ISBN: 978-92-4-154787-1.

[2] Michael S. Diamond and Theodore C. Pierson. "Molecular Insight into Dengue Virus Pathogenesis and Its Implications for Disease Control". English. In: *Cell* 162.3 (July 2015). Publisher: Elsevier, pp. 488–492. ISSN: 0092-8674, 1097-4172. DOI: `10.1016/j.cell.2015.07.005`. URL: `https://www.cell.com/cell/abstract/S0092-8674(15)00842-9` (visited on 01/20/2021).

[3] Samir Bhatt et al. "The global distribution and burden of dengue". eng. In: *Nature* 496.7446 (Apr. 2013), pp. 504–507. ISSN: 1476-4687. DOI: `10.1038/nature12060`.

[4] José Lourenço et al. "Challenges in dengue research: A computational perspective". en. In: *Evolutionary Applications* 11.4 (2018). _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/eva.12554, pp. 516–533. ISSN: 1752-4571. DOI: `10.1111/eva.12554`. URL: `https://onlinelibrary.wiley.com/doi/abs/10.1111/eva.12554` (visited on 01/24/2022).

[5] Katrin C. Leitmeyer et al. "Dengue Virus Structural Differences That Correlate with Pathogenesis". en. In: *Journal of Virology* 73.6 (June 1999), pp. 4738–4747. ISSN: 0022-538X, 1098-5514. DOI: `10.1128/JVI.73.6.4738-4747.1999`. URL: `https://journals.asm.org/doi/10.1128/JVI.73.6.4738-4747.1999` (visited on 03/29/2022).

[6] Nathan L. Yozwiak et al. "Virus Identification in Unknown Tropical Febrile Illness Cases Using Deep Sequencing". en. In: *PLoS Neglected Tropical Diseases* 6.2 (Feb. 2012). Ed. by Rebeca Rico-Hesse, e1485. ISSN: 1935-2735. DOI: `10.1371/journal.pntd.0001485`. URL: `https://dx.plos.org/10.1371/journal.pntd.0001485` (visited on 06/19/2021).

[7] Chun Kiat Lee et al. "Clinical use of targeted high-throughput whole-genome sequencing for a dengue virus variant". en. In: *Clinical Chemistry and Laboratory Medicine (CCLM)* 55.9 (Sept. 2017). Publisher: De Gruyter, e209–e212. ISSN: 1437-4331. DOI: `10.1515/cclm-2016-0660`. URL: `https://www.degruyter.com/document/doi/10.1515/cclm-2016-0660/html` (visited on 01/24/2022).

[8] Zareen Fatima et al. "Serotype and genotype analysis of dengue virus by sequencing followed by phylogenetic analysis using samples from three mini outbreaks-2007-2009 in Pakistan". In: *BMC Microbiology* 11.1 (2011), p. 200. ISSN: 1471-2180. DOI: `10.1186/1471-2180-11-200`. URL: `https://doi.org/10.1186/1471-2180-11-200` (visited on 01/24/2022).

[9] Vagner Fonseca et al. "A computational method for the identification of Dengue, Zika and Chikungunya virus species and genotypes". en. In: *PLOS Neglected Tropical Diseases* 13.5 (2019). Publisher: Public Library of Science, e0007231. ISSN: 1935-2735. DOI: `10.1371/journal.pntd.0007231`. URL: `https://journals.plos.org/plosntds/article?id=10.1371/journal.pntd.0007231` (visited on 01/20/2021).

[10] Michael Vilsker et al. "Genome Detective: an automated system for virus identification from high-throughput sequencing data". In: *Bioinformatics* 35.5 (Mar. 2019), pp. 871–873. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/bty695`. URL: `https://doi.org/10.1093/bioinformatics/bty695` (visited on 01/24/2022).

[11] Yang Li et al. "VIP: an integrated pipeline for metagenomics of virus identification and discovery". en. In: *Scientific Reports* 6.1 (Mar. 2016). Bandiera_abtest: a Cc_license_type: cc_by Cg_type: Nature Research Journals Number: 1 Primary_atype: Research Publisher: Nature Publishing Group Subject_term: Computational biology and bioinformatics;Microbiology Subject_term_id: computational-biology-and-bioinformatics;microbiology, p. 23774. ISSN: 2045-2322. DOI: `10.1038/srep23774`. URL: `https://www.nature.com/articles/srep23774` (visited on 01/24/2022).

[12] Akifumi Yamashita, Tsuyoshi Sekizuka, and Makoto Kuroda. "VirusTAP: Viral Genome-Targeted Assembly Pipeline". In: *Frontiers in Microbiology* 7 (2016). ISSN: 1664-302X. URL: `https://www.frontiersin.org/article/10.3389/fmicb.2016.00032` (visited on 01/24/2022).

[13] Hsin-Hung Lin and Yu-Chieh Liao. "drVM: a new tool for efficient genome assembly of known eukaryotic viruses from metagenomes". en. In: *GigaScience* 6.2 (Feb. 2017). ISSN: 2047-217X. DOI: `10.1093/gigascience/gix003`. URL: `https://academic.oup.com/gigascience/article/doi/10.1093/gigascience/gix003/2929394` (visited on 03/29/2022).

[14] Paolo Di Tommaso et al. "Nextflow enables reproducible computational workflows". en. In: *Nature Biotechnology* 35.4 (Apr. 2017), pp. 316–319. ISSN: 1087-0156, 1546-1696. DOI: `10.1038/nbt.3820`. URL: `http://www.nature.com/articles/nbt.3820` (visited on 03/24/2021).

[15] Lisa Gerhardt et al. "Shifter: Containers for HPC". In: *Journal of Physics: Conference Series* 898 (Oct. 2017), p. 082021. ISSN: 1742-6588, 1742-6596. DOI: `10.1088/1742-6596/898/8/082021`. URL: `https://iopscience.iop.org/article/10.1088/1742-6596/898/8/082021` (visited on 03/24/2021).

[16] Gregory M. Kurtzer, Vanessa Sochat, and Michael W. Bauer. "Singularity: Scientific containers for mobility of compute". en. In: *PLOS ONE* 12.5 (Nov. 2017). Publisher: Public Library of Science, e0177459. ISSN: 1932-6203. DOI: `10.1371/journal.pone.0177459`. URL: `https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0177459` (visited on 03/17/2022).

[17] Robert Schmieder and Robert Edwards. “Quality control and preprocessing of metagenomic datasets”. In: *Bioinformatics* 27.6 (Mar. 2011), pp. 863–864. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/btr026`. URL: `https://doi.org/10.1093/bioinformatics/btr026` (visited on 01/24/2022).

[18] Ben Langmead and Steven L Salzberg. “Fast gapped-read alignment with Bowtie 2”. en. In: *Nature Methods* 9.4 (Apr. 2012), pp. 357–359. ISSN: 1548-7091, 1548-7105. DOI: `10.1038/nmeth.1923`. URL: `http://www.nature.com/articles/nmeth.1923` (visited on 03/18/2022).

[19] Anton Bankevich et al. “SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing”. en. In: *Journal of Computational Biology* 19.5 (May 2012), pp. 455–477. ISSN: 1066-5277, 1557-8666. DOI: `10.1089/cmb.2012.0021`. URL: `http://www.liebertpub.com/doi/10.1089/cmb.2012.0021` (visited on 03/18/2022).

[20] Dinghua Li et al. “MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph”. In: *Bioinformatics* 31.10 (May 2015), pp. 1674–1676. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/btv033`. URL: `https://doi.org/10.1093/bioinformatics/btv033` (visited on 03/14/2022).

[21] Bruce J. Walker et al. “Pilon: An Integrated Tool for Comprehensive Microbial Variant Detection and Genome Assembly Improvement”. en. In: *PLOS ONE* 9.11 (Nov. 2014). Publisher: Public Library of Science, e112963. ISSN: 1932-6203. DOI: `10.1371/journal.pone.0112963`. URL: `https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0112963` (visited on 01/24/2022).

[22] Miguel Paulo Machado et al. “Epidemiological Surveillance and Typing Methods to Track Antibiotic Resistant Strains Using High Throughput Sequencing”. en. In: *Antibiotics: Methods and Protocols*. Ed. by Peter Sass. Methods in Molecular Biology. New York, NY: Springer, 2017, pp. 331–356. ISBN: 978-1-4939-6634-9. DOI: `10.1007/978-1-4939-6634-9_20`. URL: `https://doi.org/10.1007/978-1-4939-6634-9_20` (visited on 01/24/2022).

[23] S. Altschul. “Gapped BLAST and PSI-BLAST: a new generation of protein database search programs”. In: *Nucleic Acids Research* 25.17 (Sept. 1997), pp. 3389–3402. ISSN: 13624962. DOI: `10.1093/nar/25.17.3389`. URL: `https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/25.17.3389` (visited on 03/29/2022).

[24] Tsukasa Nakamura et al. “Parallelization of MAFFT for large-scale multiple sequence alignments”. In: *Bioinformatics* 34.14 (July 2018), pp. 2490–2492. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/bty121`. URL: `https://doi.org/10.1093/bioinformatics/bty121` (visited on 01/20/2021).

[25] Alexandros Stamatakis. “RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies”. In: *Bioinformatics* 30.9 (2014), pp. 1312–1313. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/btu033`. URL: `https://doi.org/10.1093/bioinformatics/btu033` (visited on 01/24/2022).

[26] Poornima Parameswaran et al. “Intrahost Selection Pressures Drive Rapid Dengue Virus Microevolution in Acute Human Infections”. en. In: *Cell Host & Microbe* 22.3 (Sept. 2017), 400–410.e5. ISSN: 19313128. DOI: `10.1016/j.chom.2017.08.003`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S1931312817303359` (visited on 06/19/2021).

[27] Paula Eillanny Silva Marinho et al. “Meningitis Associated with Simultaneous Infection by Multiple Dengue Virus Serotypes in Children, Brazil - Volume 23, Number 1—January 2017 - Emerging Infectious Diseases journal - CDC”. en-us. In: (). DOI: `10.3201/eid2301.160817`. URL: `https://wwwnc.cdc.gov/eid/article/23/1/16-0817_article` (visited on 01/24/2022).

[28] Manchala Nageswar Reddy et al. “Occurrence of concurrent infections with multiple serotypes of dengue viruses during 2013–2015 in northern Kerala, India”. en. In: *PeerJ* 5 (Mar. 2017), e2970. ISSN: 2167-8359. DOI: `10.7717/peerj.2970`. URL: `https://peerj.com/articles/2970` (visited on 03/30/2022).

[29] Lize Cuypers et al. “Time to Harmonize Dengue Nomenclature and Classification”. en. In: *Viruses* 10.10 (Oct. 2018), p. 569. ISSN: 1999-4915. DOI: `10.3390/v10100569`. URL: `http://www.mdpi.com/1999-4915/10/10/569` (visited on 06/19/2021).

[30] Brett E. Pickett et al. “Virus Pathogen Database and Analysis Resource (ViPR): A Comprehensive Bioinformatics Database and Analysis Resource for the Coronavirus Research Community”. en. In: *Viruses* 4.11 (Nov. 2012). Number: 11 Publisher: Molecular Diversity Preservation International, pp. 3209–3226. ISSN: 1999-4915. DOI: `10.3390/v4113209`. URL: `https://www.mdpi.com/1999-4915/4/11/3209` (visited on 02/18/2022).

[31] Christian Julián Villabona-Arenas and Paolo Marinho de Andrade Zanotto. “Worldwide spread of Dengue virus type 1”. eng. In: *PloS One* 8.5 (2013), e62649. ISSN: 1932-6203. DOI: `10.1371/journal.pone.0062649`.

[32] Nikos Vasilakis and Scott C. Weaver. “The history and evolution of human dengue emergence”. eng. In: *Advances in Virus Research* 72 (2008), pp. 1–76. ISSN: 0065-3527. DOI: `10.1016/S0065-3527(08)00401-6`.

[33] Weizhong Li and Adam Godzik. “Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences”. eng. In: *Bioinformatics (Oxford, England)* 22.13 (July 2006), pp. 1658–1659. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/btl158`.

[34] R. Rico-Hesse. “Molecular evolution and distribution of dengue viruses type 1 and 2 in nature”. eng. In: *Virology* 174.2 (Feb. 1990), pp. 479–493. ISSN: 0042-6822. DOI: `10.1016/0042-6822(90)90102-w`.

[35] Rebeca Rico-Hesse. “Microevolution and virulence of dengue viruses”. eng. In: *Advances in Virus Research* 59 (2003), pp. 315–341. ISSN: 0065-3527. DOI: `10.1016/s0065-3527(03)59009-1`.

[36] R. S. Lanciotti et al. “Rapid detection and typing of dengue viruses from clinical samples by using reverse transcriptase-polymerase chain reaction”. eng. In: *Journal of Clinical Microbiology* 30.3 (Mar. 1992), pp. 545–551. ISSN: 0095-1137. DOI: `10.1128/jcm.30.3.545-551.1992`.

[37] R. S. Lanciotti, D. J. Gubler, and D. W. Trent. “Molecular evolution and phylogeny of dengue-4 viruses”. eng. In: *The Journal of General Virology* 78 (Pt 9) (Sept. 1997), pp. 2279–2284. ISSN: 0022-1317. DOI: `10.1099/0022-1317-78-9-2279`.

[38] Chonticha Klungthong et al. “The molecular epidemiology of dengue virus serotype 4 in Bangkok, Thailand”. eng. In: *Virology* 329.1 (Nov. 2004), pp. 168–179. ISSN: 0042-6822. DOI: `10.1016/j.virol.2004.08.003`.

[39] Chunlin Zhang et al. “Clade replacements in dengue virus serotypes 1 and 3 are associated with changing serotype prevalence”. eng. In: *Journal of Virology* 79.24 (Dec. 2005), pp. 15123–15130. ISSN: 0022-538X. DOI: `10.1128/JVI.79.24.15123-15130.2005`.

[40] Chunlin Zhang et al. “Structure and age of genetic diversity of dengue virus type 2 in Thailand”. eng. In: *The Journal of General Virology* 87.Pt 4 (Apr. 2006), pp. 873–883. ISSN: 0022-1317. DOI: `10.1099/vir.0.81486-0`.

[41] Anthony M. Bolger, Marc Lohse, and Bjoern Usadel. “Trimmomatic: a flexible trimmer for Illumina sequence data”. In: *Bioinformatics* 30.15 (2014), pp. 2114–2120. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/btu170`. URL: `https://doi.org/10.1093/bioinformatics/btu170` (visited on 03/02/2022).

[42] Gilberto A. Santiago et al. “Analytical and Clinical Performance of the CDC Real Time RT-PCR Assay for Detection and Typing of Dengue Virus”. In: *PLoS Neglected Tropical Diseases* 7.7 (July 2013), e2311. ISSN: 1935-2727. DOI: `10.1371/journal.pntd.0002311`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3708876/` (visited on 04/11/2022).

Part II

Impact of *de novo* assemblers in metagenomics

Chapter 5

LMAS: Last Metagenomic Assembler Standing

5. LMAS: LAST METAGENOMIC ASSEMBLER STANDING

This chapter is a reproduction of the following manuscript:

C. I. Mendes, P. Vila-Cerqueira, Y. Motro, J. Moran-Gilad, J. A. Carriço, M. Ramirez. LMAS: Evaluating metagenomic short de novo assembly methods through defined communities. GigaScience, Volume 12, 2023, giac122, `https://doi.org/10.1093/gigascience/giac122`

The supplementary information referred throughout the text can be consulted in this chapter before the section of references.

Short-read SMg can offer comprehensive microbial detection and characterisation of complex clinical samples. The *de novo* assembly of raw sequence data is key in metagenomic analysis, yielding longer sequences that offer contextual information and afford a more complete picture of the microbial community. The assembly process is the bedrock and may constitute a major bottleneck in obtaining trustworthy, reproducible results.

In this chapter, we present LMAS, an automated workflow developed as a flexible platform to allow users to evaluate traditional and metagenomic dedicated prokaryotic de novo assembly software performance given known standard communities. Its implementation in Nextflow ensures the transparency and reproducibility of the results obtained and the use of Docker containers provides further flexibility. The results are presented in an interactive HTML report where global and reference-specific performance metrics can be explored. Currently, 12 assemblers still being maintained are implemented in LMAS, with the possibility of expansion as novel algorithms are developed.

To showcase LMAS we initially used a test dataset of eight bacterial genomes and four plasmids of the ZymoBIOMICS Microbial Community Standards with linear and logarithmic distribution and found that k-mer De Bruijn graph assemblers outperformed the alternative approaches but came with a greater computational cost. Furthermore, assemblers branded as metagenomic specific did not consistently outperform other genomic assemblers in metagenomic samples. Some assemblers still in use, such as ABySS, MetaHipmer2, minia and VelvetOptimiser, showed significant performance problems and their usability may be limited, particularly when assembling complex samples.

To test assembler performance with an even more complex dataset, we used the 12-strain BMock12 community standard. This sample includes a non-even distribution of species and several closely related sets: two replicons of Halomonas sp. (ANIb=0.98), three replicons of the Micromonospora genus (average ANIb=0.85) and two replicons of Marinobacter sp (ANIb=0.78). Furthermore, and to represent a mock community trying to reproduce an existing microbiome, the NIBSC Gut DNA Reference Gut-Mix-RR and Gut-Mix-HiLo community standards were analysed, consisting of 20 common gut microbiome strains in an even and staggered composition respectively.

The performance of each assembler varied depending on the species of interest and its abundance in the sample, with less abundant species presenting a significant challenge for all assemblers. No assembler stood out as an undisputed all-purpose choice for short-read metagenomic prokaryote genome assembly, highlighting that efforts are still needed to further improve metagenomic assembly performance. Our results also suggest that sample complexity and a particular interest in some sample components may affect assembler choice. Using LMAS could help users in their choice of assembler for their specific purpose. As such, we believe that this manuscript is appropriate for publication in Microbiome as a Software article.

My contribution to this publication included the design, implementation and optimisation of the LMAS workflow, including the creation of the Docker containers for all dependencies. I performed the data analysis and comparison of assemblers included in LMAS with ZymoBIOMICS Microbial Community Standards, both evenly and logarithmically distributed samples, BMock12 and IBSC Gut DNA Reference Gut-Mix-RR and Gut-Mix-HiLo community standards. Additionally, I've also written the manuscript.

LMAS: Evaluating metagenomic short de novo assembly methods through defined communities

Catarina I. Mendes[1], ∗, Pedro Vila-Cerqueira[1]∗, Yair Motro[2], Jacob Moran-Gilad[2], João A. Carriço[1] Mário Ramirez[1],

[1]Instituto de Microbiologia, Instituto de Medicina Molecular, Faculdade de Medicina, Universidade de Lisboa, Lisboa, Portugal

[2]Faculty of Health Sciences, Ben-Gurion University of the Negev, Beer-Sheva, Israel

5.1 Abstract

Background The de novo assembly of raw sequence data is key in metagenomic analysis. It allows recovering draft genomes from a pool of mixed raw reads, yielding longer sequences that offer contextual information and provide a more complete picture of the microbial community.

Results To better compare de novo assemblers for metagenomic analysis, LMAS was developed as a flexible platform allowing users to evaluate assembler performance given known standard communities. Overall, in our test datasets, k-mer De Bruijn graph assemblers outperformed the alternative approaches but came with a greater computational cost. Furthermore, assemblers branded as metagenomic specific did not consistently outperform other genomic assemblers in metagenomic samples. Some assemblers still in use, such as ABySS, MetaHipmer2, minia and VelvetOptimiser, perform relatively poorly and should be used with caution when assembling complex samples.

Conclusions The choice of a de novo assembler depends on the computational resources available, the replicon of interest, and the major goals of the analysis. No single assembler appeared an ideal choice for short-read metagenomic prokaryote replicon assembly, each showing specific strengths. The choice of metagenomic assembler should be guided by user requirements and characteristics of the sample of interest, and LMAS provides an interactive evaluation platform for this purpose.

5.1.0.1 Keywords

Shotgun Metagenomics, de novo assembly, benchmark, draft genome quality, simulation

5.2 Background

Short-read shotgun metagenomics has the potential to offer comprehensive microbial detection and characterisation of complex clinical or environmental samples. Despite becoming an increasingly used approach, it comes at the cost of producing massive amounts of data that require expert handling and processing, as well as adequate computational resources. The *de novo* assembly process is key when analysing metagenomic data since it allows recovering contigs representing the replicons present in the sample, be it prokaryotic chromosomes, plasmids or viruses, from a pool of mixed raw reads. These contigs are longer sequences that offer better contextual information than reads alone and provide a more complete picture of the microbial community than the species composition. Despite efforts for the development, standardisation and assessment of software for metagenomic analysis, both commercial and open-source [1–5], the de novo assembly process still represents a critical point in these analyses.

The assembly of draft genomes has become a central step when analysing pure bacterial cultures, for instance allowing genomic comparisons through single nucleotide SNPs or gene-by-gene methods, such as core-genome Multilocus Sequence Typing (cgMLST). De Bruijn graphs (dBg) algorithms are currently the most widely used approach in modern assembly software. dBg handles unresolvable repeats by essentially fragmenting the sequence, that is, forming multiple contigs for each of the possibly contiguous sequences present in the sample. Additionally, the inherent heterogeneity of complex samples, potentially containing a multitude of replicons, could make traditional genome assemblers, implementing optimisations based on the assumption of having a single genome in the sample, not suitable for metagenomics [6].

Several dedicated metagenomic assembly tools for short-read data are available [6]. These tools are generally assumed to perform better when dealing with complex samples having a combination of intragenomic and intergenomic repeats and uneven and low coverage sequencing depths of some of the replicons [7]. Not using dedicated metagenomic assemblers was suggested to come with the cost of generating artificial variation and chimeric contigs, especially in samples that contain closely related species [8]. However, to our knowledge, no formal comparison has been done looking at increased accuracy or gains in contiguity of assemblies obtained with metagenomic assemblers versus traditional assemblers.

With an ever-increasing range of both traditional and metagenomic assemblers becoming available, choosing the best performing tool can be an arduous and time-consuming task since the choice may vary depending on the purpose of the analysis, organism of interest, complexity of the sample and computational infrastructure available. Additionally, the evaluation of the resulting contigs is not straightforward since one metric is not sufficient to classify an assembly, particularly with complex samples [7, 9]. Despite several de novo assembly validation methods relying on features of the created contigs themselves, such as QUAST [10], being useful in identifying inconsistencies indicative of potential assembly errors, the use of reference-based validation methods offer the possibility of a more complete evaluation of accuracy and are particularly important to benchmark attempts to reconstruct communities. MetaQUAST [9], a modification of QUAST, extends the original software by performing assembly evaluation based on aligning contigs to a reference, which can be provided or inferred by the software, and reports, in addition to the standard metrics for single genomes reported by QUAST, the number of interspecies translocations and the number of possibly misassembled contigs.

The use of mock communities, with known composition, abundance and genomic information, provides a ground truth against which the success of the assembly of a complex sample can be evaluated. Such mock communities facilitate the identification of misassemblies, such as chimeric sequences generated from the improper combination of two distinct replicons, indels or single nucleotide variants improperly created by the assembler. On the other hand, the comparison of the performance of two assemblers is only possible if the input data is the same and if the same evaluation metrics are applied [3].

To tackle these challenges, we developed LMAS (Last Metagenomic Assembler Standing), an automated workflow to enable the benchmarking of traditional and metagenomic prokaryotic de novo assembly software using defined mock communities. The results of LMAS are presented in an interactive HTML report where selected global and reference replicon-specific performance metrics can be explored. The mock communities can be provided by the user to better reflect the samples of interest. New assemblers can be added with minimal changes to the pipeline so that LMAS can be expanded to include novel algorithms as they are developed. The portability and ease of use of LMAS are intended to allow provide users with a continuous benchmarking platform to easily evaluate the performance of assemblers, in mock communities mimicking as closely as possible their samples of interest.

5.3 Implementation

5.3.1 Workflow overview

LMAS is a user-friendly automated workflow enabling the benchmarking of traditional and metagenomic prokaryotic de novo assembly software using defined mock communities. LMAS was implemented in Nextflow [11] to provide flexibility and ensure the transparency and reproducibility of the results. LMAS relies on the use of Docker [12] containers for each assembler, allowing versions to be tracked and changed easily.

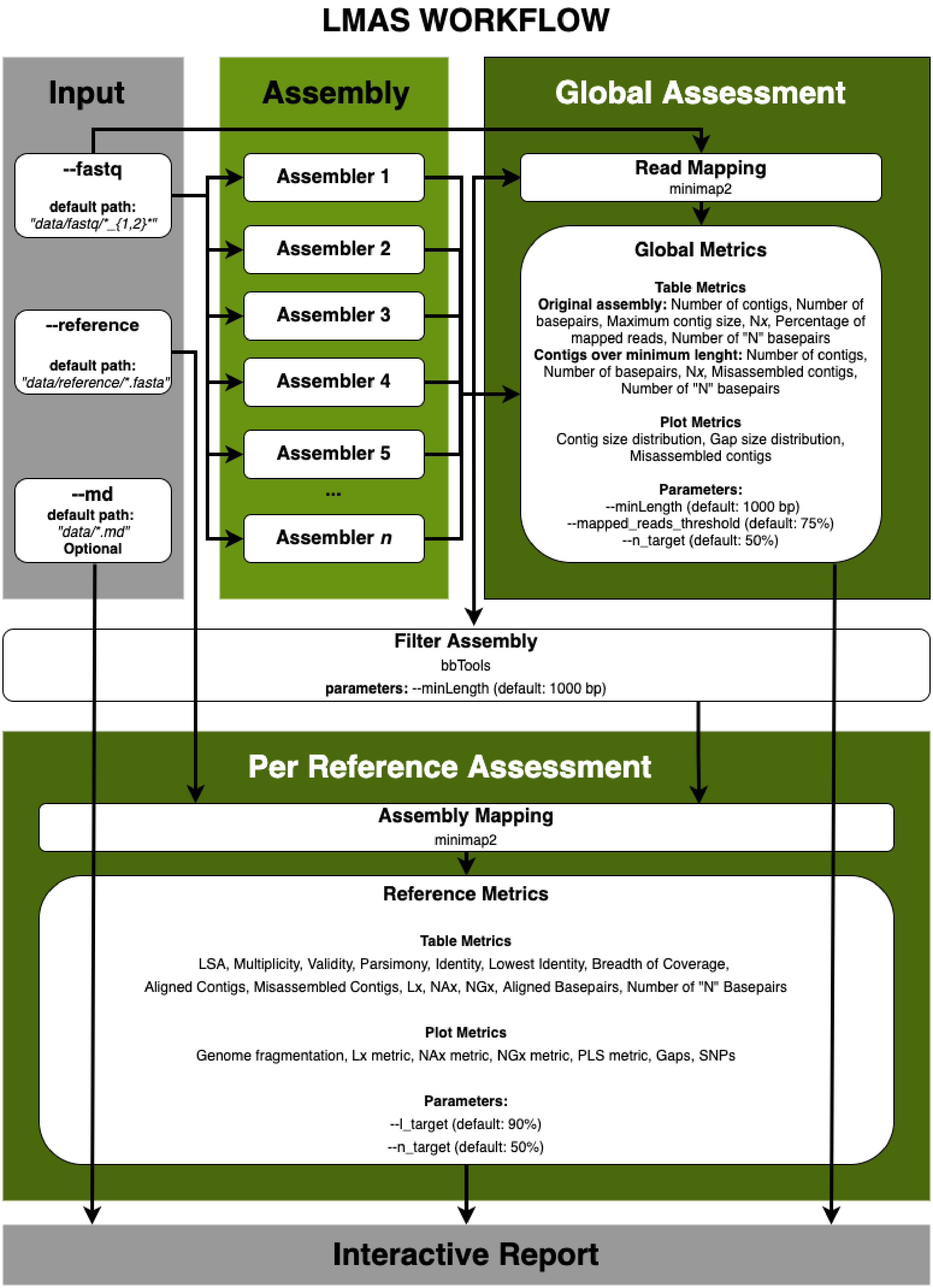


Figure 5.1: The LMAS workflow. The input sequencing data is assembled in parallel, resources permitting, by the set of assemblers included in LMAS. The resulting contigs are processed and the global quality assessment is performed. After filtering for the user-defined minimum contig size, the remaining sequences are mapped against the provided reference and the resulting information is processed to evaluate assembly quality by replicon in the reference file. All results, and optional text information describing the samples, are grouped in the LMAS report.

5.3.2 Installation and Usage

LMAS can be installed through Bioconda [13] or Github [14], with detailed instructions available in the documentation [15]. LMAS requires as inputs the complete reference replicons (genomes, plasmids or any other replicons present) and short-read paired-end raw data. All complete references (linear replicons) should be provided in a single file. This raw data can be either obtained in silico by creating simulated reads from the reference replicons or sequencing mock communities of known composition. Optionally, information on the input samples in a markdown file can be provided to be presented in the report.

A step-by-step execution tutorial is available at [16]. Users can customise the workflow execution either by using command-line options or by modifying the simple plain-text configuration files. To make the execution of the workflow as simple as possible, a set of default parameters and directives is provided. A complete description of each parameter is available in Supplemental Material (see Supplemental Material, Workflow parameters), as well as in the documentation [17]. The results are presented in an interactive HTML report, stored in the “report” folder in the directory of LMAS’ execution. The output files of all assemblers and quality assessment processing scripts in the workflow are stored in the “results” folder, in the same location.

5.3.3 Supported Assemblers and selection criteria

A collection of de novo assembly tools was compiled, including OLC and De Bruijn graphs (dBg) assembly algorithms, the latter including both single k-mer and multiple k-mer value approaches, and hybrid assemblers implementing both algorithms, including both genomic and metagenomic assemblers (Supplemental Table 5.2).Of these, 11 assemblers were selected based on the date of last update (at least 2015) and are implemented in LMAS: ABySS [18] (version 2.3.1), GATB Minia Pipeline [19] (commit hash 9d56f42) , IDBA-UD [20] (version 1.1.3), MEGAHIT [21] (version 1.2.9), MetaHipMer2 [22] (version 2.0.0.65-gaad446d-dirty-AddGtest), metaSPAdes [23] (version 3.15.3), minia [24] (version 3.2.6), SKESA [25] (version 2.5.0), SPAdes [26] (version 3.15.3), Unicycler [27] (version 0.4.9) and VelvetOptimiser [28] (commit hash 092bdee) (Table 5.1). The execution commands for each assembler are available as Supplemental Material (see 5.8.2) and in the documentation [29]. New assemblers can be added with minimal changes to the pipeline so that LMAS can be expanded as novel algorithms are developed. A template is available to facilitate their integration and a step-by-step guide is included in the documentation [30]. The only two requirements for the addition of a new assembler are the execution command for the assembler for paired-end short-read data and a Nextflow-compatible container with the assembler and any dependencies.

Table 5.1: Prokaryotic de novo assemblers integrated into LMAS.

Assembler	Type	Algorithm
GATBMiniaPipeline	Metagenomic	Multiple k-mer De Bruijn graph
IDBA-UD	Metagenomic	Multiple k-mer De Bruijn graph
MEGAHIT	Metagenomic	Multiple k-mer De Bruijn graph
MetaHipMer2	Metagenomic	Multiple k-mer De Bruijn graph
metaSPAdes	Metagenomic	Multiple k-mer De Bruijn graph
ABySS	Genomic	Single k-mer De Bruijn graph
MINIA	Genomic	Single k-mer De Bruijn graph
SKESA	Genomic	Multiple k-mer De Bruijn graph
SPAdes	Genomic	Multiple k-mer De Bruijn graph
Unicycler	Genomic	Multiple k-mer De Bruijn graph
VelvetOptimizer	Genomic	Single k-mer De Bruijn graph

5.3.4 Assembly Quality Metrics

The success of an assembly is evaluated in two steps: globally (see 5.3.4.1) and relative to each of the replicons present in the sample (see 5.3.4.2). In both, the tabular presentation in the reports allows the comparison of exact values between assemblers, and the interactive plots allow a more intuitive overview and easy exploration of results. In addition to the assembly success metrics, computational resource statistics are registered for each assembler (see 5.8.5.3).

5.3.4.1 Global Metrics

The computation of the global metrics is performed through statistics inherent to the complete set of contigs assembled per sample, independent of the species/sample of origin. The metrics are presented, in tabular form, for the complete set of contigs and those filtered for a minimum length, and also graphically for the contigs filtered for a minimum length. The statistics include information on contig number, size and ambiguous bases; and the proportion of reads mapping to the created contigs. Two statistics are a consolidation of per reference metrics: misassemblies (i.e. contigs that do not reflect the structural organisation in the reference replicons); and the overall size of gaps in all reference replicons not covered by any contig. A more detailed description of all global metrics is available in Supplemental Material (see 5.8.5.1).

5.3.4.2 Per Reference Metrics

For the computation of the reference-based metrics, only the Filtered Set (FS) contigs are considered, for each reference replicon in the sample. These contigs are the ones exceeding the user-defined minimum sequence length, filtered using BBTools (version 38.44). After this initial step, the contigs are mapped to the reference replicons with minimap2 [31] (version 2.22). The metrics are computed through custom python code (see 5.8.4) for each replicon in the file provided as input. A detailed description of all reference-based metrics is available in Supplemental Material (see 5.8.5.2).

In addition to the statistics shared with the global metrics, LMAS also calculates the number of mismatches relative to each reference, the COMPASS [32] metrics and two new metrics we propose: Longest single alignment (LSA) and Pls.

LSA represents the fraction of the longest single alignment between a contig and the reference, relative to the reference length. The Pls, or Phred-like score, is a scoring function based on the identity of each aligned contig to the reference replicon. Similarly to the Phred quality score [33], a measure of the quality of the identification of the bases by sequencing, the Pls measures the quality of the assembly of a contig. The formula of Pls is similar to the Phred score formula but uses as the error function the identity of the base in the contig to that of the reference replicon. The formula to obtain the Pls metric per contig is Equation 5.1.

$$Phred(E) = \begin{cases} -log(E) \times 10 & \text{if } 0 < E \leq 1 \\ 60 & \text{if } E = 0 \end{cases} \quad \text{where } E = 1 - \text{Identity} \tag{5.1}$$

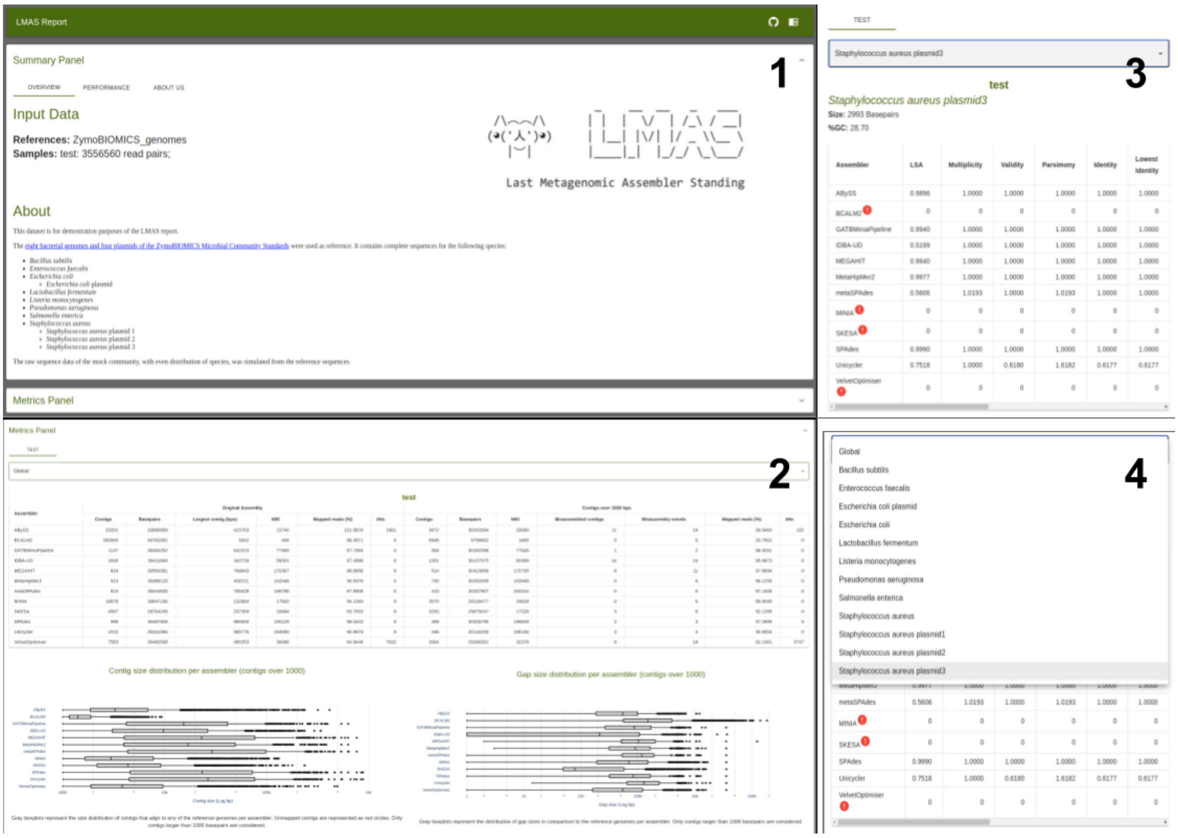
5.3.5 The LMAS Report

The LMAS results are presented in an interactive HTML. The LMAS report is composed of two main panels: a top summary panel with information on input samples (provided by the user) and the resources used during LMAS' execution, and a bottom panel where selected global and reference specific assembly metrics can be explored for each sample. LMAS constructs the HTML file after workflow completion, storing it in the "reports" folder. The report data can be easily shared between users and requires only a browser for visualisation.

5.3.5.1 Summary Panel

The top panel of the report contains information on the input samples and overall performance of the assemblers in LMAS, divided into three tabs: Overview, Performance and About us. On the top right corner of the report, direct links to LMAS' source repository and documentation are provided.

- *Overview*: This tab contains information on the input data, including the name and number of reads of the raw sequencing data, and the name of the reference file. Additional information provided by the user about the community used as input is also



Input Data

References: ZymoBIOMICS_genomes
Samples: test: 3556560 read pairs;

About

This dataset is for demonstration purposes of the LMAS report.

The eight bacterial genomes and four plasmids of the ZymoBIOMICS Microbial Community Standards were used as reference. It contains complete sequences for the following species:

- *Bacillus subtilis*
- *Enterococcus faecalis*
- *Escherichia coli*
  - *Escherichia coli* plasmid
- *Lactobacillus fermentum*
- *Listeria monocytogenes*
- *Pseudomonas aeruginosa*
- *Salmonella enterica*
- *Staphylococcus aureus*
  - *Staphylococcus aureus* plasmid 1
  - *Staphylococcus aureus* plasmid 2
  - *Staphylococcus aureus* plasmid 3

The raw sequence data of the mock community, with even distribution of species, was simulated from the reference sequences.

Staphylococcus aureus plasmid3

test

Staphylococcus aureus plasmid3
Size: 2993 Basepairs
%GC: 28.70

Assembler	LSA	Multiplicity	Validity	Parsimony	Identity	Lowest identity
ABySS	0.9896	1.0000	1.0000	1.0000	1.0000	1.0000
BCALM2	0	0	0	0	0	0
GATBMiniaPipeline	0.9940	1.0000	1.0000	1.0000	1.0000	1.0000
IDBA-UD	0.5199	1.0000	1.0000	1.0000	1.0000	1.0000
MEGAHIT	0.9940	1.0000	1.0000	1.0000	1.0000	1.0000
MetaHipMer2	0.9977	1.0000	1.0000	1.0000	1.0000	1.0000
metaSPAdes	0.5606	1.0193	1.0000	1.0193	1.0000	1.0000
MINIA	0	0	0	0	0	0
SKESA	0	0	0	0	0	0
SPAdes	0.9990	1.0000	1.0000	1.0000	1.0000	1.0000
Unicycler	0.7518	1.0000	0.6180	1.6182	0.6177	0.6177
VelvetOptimiser	0	0	0	0	0	0

Global
Bacillus subtilis
Enterococcus faecalis
Escherichia coli plasmid
Escherichia coli
Lactobacillus fermentum
Listeria monocytogenes
Pseudomonas aeruginosa
Salmonella enterica
Staphylococcus aureus
Staphylococcus aureus plasmid1
Staphylococcus aureus plasmid2
Staphylococcus aureus plasmid3

Figure 5.2: The LMAS report. All results, and optional text information describing the samples, are grouped in the LMAS report, an interactive and responsive HTML file, for exploration in any browser. Links for LMAS source code and documentation are available in the top right corner of the report. 1) The summary panel of the LMAS report contains information on the input reference sequences and raw sequencing data samples (provided by the user), and the overall computational performance of the assemblers in LMAS. 2) The LMAS metric panel contains the explorable global and reference specific performance metrics per input raw sequencing data sample. The tabular presentation allows direct comparison of exact values between assemblies, and the interactive plots allow for an intuitive overview and easy exploration of results. 3) If an assembler fails to produce an assembly, or fails to assemble sequences that map to the reference replicon, it is marked in the table with a red warning sign. 4) The global or reference replicon specific metrics can be accessed for each sample in the dropdown menu.

presented here.

- *Performance*: This tab contains a table with information on the version, the containers used and computational performance metrics for each assembler in LMAS.
- *About us*: This tab contains information on the LMAS GitHub repositories and the LMAS development team.

5.3.5.2 Metrics Panel

The bottom portion of the report contains the explorable global and reference specific performance metrics per input raw sequencing data sample. Each sample has its own tab and the global or reference replicon specific metrics can be accessed in the dropdown menu.

5.3.5.2.1 Global Metrics

A table displays the global assembly metrics computed for the complete and FS contigs. If an assembler fails to produce an assembly, it is marked on the table with a red warning sign. The global metric plots are interactive, allow zooming in on particular areas and provide extra information as hover text boxes. The plots can be saved as PNG in whatever view the user selects.

5.3.5.2.2 Per Reference Metrics

Similarly to the global assembly metrics, a table displays the computed set of reference restricted metrics for the FS contigs. If an assembler fails to produce sequences that align to the reference, these are marked in the table with a red warning sign. Information on the expected reference replicon length and the GC content is calculated from the input files and reported above the table. The per-reference metric plots are also interactive, allowing the same type of operations as the global metric plots.

5.3.6 Comparison with other assembly evaluation software programs

The assessment and evaluation of genome assemblies has been a relevant field ever since the emergence of the assembly process itself, and therefore many solutions have been proposed [3, 7, 9, 10, 34–37]. The Critical Assessment of Metagenome Interpretation (CAMI) proposed a set of recommendations and best practices for benchmarking in microbiome research [38]. These recommendations include the reporting of computational performance, which may condition the choice of software by the users, such as runtime, disk space and memory consumption, also reported by LMAS (see 5.8.5). As also suggested by CAMI, LMAS tracks the exact program version and command-line calls through its implementation in Nextflow. Moreover, using containerised assemblers and being easily installable through Bioconda, LMAS facilitates deployment in diverse user machines. Unlike the CAMI tutorial, in which users are asked to download and install the necessary tools, in LMAS everything is provided in a one-stop reproducible workflow that effortlessly handles all pre-processing, assembly, post-processing, traceability and report production steps, freeing users to focus on providing relevant samples for analysis and interpreting the results in view of the intended applications.

Concerning software for assembly quality assessment currently available, the most widely adopted is QUAST [10], or when dealing with metagenomic data, its extension metaQUAST [34], which was also adopted by the CAMI challenges [3,5] [3, 5] and suggested in the CAMI Tutorial [38]. Although several features of these tools overlap with

LMAS' quality assessment components, these differ from LMAS in the sense that they are not a single step workflow allowing a traceable and reproducible assembly of mock communities. Unlike QUAST and metaQUAST, whose purpose is to evaluate assemblies, the purpose of LMAS is to allow users to evaluate assembler performance for a given sample of interest. Supplementary Table 5.3 shows the comparison of the output and computed assembly quality metrics generated by LMAS, QUAST and metaQUAST.

5.4 Results and Discussion

To illustrate the use of LMAS and evaluate the performance of the chosen assemblers we initially used the eight bacterial genomes and four plasmids of the ZymoBIOMICS Microbial Community Standards as reference. As input we used the raw sequence reads of mock communities with an even and logarithmic distribution of species, from real sequencing runs [39] and simulated read datasets, with and without error, matching the distribution of species in each sample [40]. Our dataset is composed of samples ENN (in silico generated evenly distributed without error), EMS (in silico generated evenly distributed with Illumina MiSeq error model), ERR2984773 (evenly distributed real Illumina MiSeq sample), LNN (in silico generated logarithmically distributed without error), LHS (in silico generated logarithmically distributed with Illumina HiSeq error model) and ERR2935805 (logarithmically distributed real Illumina HiSeq sample) (see Supplemental Table 5.4). Detailed information about the generation of the input samples is available as Supplemental Material (see 5.8.7, Supplemental Table 5.5). To evaluate the reproducibility of an assembler performance, the LMAS workflow was run three times for all samples using default parameters, and the resulting data was processed for each sample (see Chapter 5.8.7.5) Supplementary Table 5.6 to Table 5.11 present an overview of the average global performance per assembler for each sample in LMAS.

To test assembler performance with an even more complex dataset, we used the 12 strain BMock community standard (accession SRX4901583, real Illumina HiSeq 2500 sample) [41]. This sample includes a non-even distribution of species, with the most abundant replicon having 3093x coverage (*Muricauda* sp. ES.050) and the lowest only 0.1x coverage (*Micromonospora coxensis*) (Supplementary Table 5.25). For the sake of a less resource-intense evaluation we downsampled to have 20% of the reads available in the original sample and further processed only these. The main challenges of this data set are possibly to assemble the genomes of the three *Micromonospora* spp. and the two *Halomonas* spp. strains, which have an ANIb >0.84 and 0.98, respectively (Supplementary Table 5.26). The two *Marinobacter* spp. have an ANIb of 0.78.

To represent more realistic samples of a human microbiome study, the Gut-Mix-RR and Gut-Mix-HiLo standards, including 20 species known to be present in the human gut, were used as reference (accessions SRR11487941 and SRR11487935, respectively, both real Illumina MiSeq samples) [42]. For the Gut-Mix-RR, the abundance of each bacterial genome is relatively even (maximum of 66x, an average of 22.32x and a minimum of 5.99x; Supplementary Table 5.29). The Gut-Mix-HiLo has an uneven abundance of species (maximum of 115x, an average of 20.45x and a minimum of 0.34x; Supplementary Table 5.29). The genomes in these mock communities are fairly diverse (average ANIb=0.67, Supplementary Table 5.30), with the two subspecies of *Bifidobacterium longum* (ANIb=0.95) possibly being the most challenging. It is worth noting that only draft genomes are available for eight of the strains, including one of the *Bifidobacterium longum* subspecies (Supplementary Table 5.29). The *Roseburia hominis* and *Roseburia intestinalis* are the closest related closed replicons (ANIb>0.77) in this sample.

5.4.1 Some assemblers perform poorly

Of the 12 de novo prokaryotic assemblers included in LMAS, 4 stand out as having an overall poor performance in the ZymoBIOMICS Microbial Community Standards dataset: ABySS, MetaHipmer2, minia and VelvetOptimiser. Both ABySS and MetaHipmer2 performed inconsistently with differing resource requirements for the same sample in different runs, namely run time and memory allocation (see 5.8.7.6, Supplemental Figure 5.10). Moreover, ABySS failed to produce an assembly for sample ERR2984773 for 1 of the runs (see Supplementary Table 5.8) and for sample LHS in any of the 3 runs in the time limit of 3 days (see Supplementary Table 5.10), and MetaHipmer2 failed to produce an assembly for samples LNN and LHS in all 3 runs (see Supplementary Tables 5.9-5.10). VelvetOptimiser generated the highest number of inconsistent contigs across the 3 LMAS runs (Figure 5.3, Supplementary Table 5.12), with 1.69% of the total contigs created present in only 1 or 2 runs. Although not as extreme as VelvetOptimiser, ABySS (0.52%), minia (0.14%), GATBMiniaPipeline (0.32%), MetaHipMer2 (0.11%) and IDBA-UD (0.08%) also showed inconsistencies in contig size.

Regarding the quality assessment of the assemblies produced (Figure 5.4, Supplementary Table 5.13), ABySS and minia are the only single k-mer dBg assemblers in the collection and were found to mostly underperform relative to their multiple k-mer dBg counterparts, as reported previously [3, 43–45], generally resulting in more fragmented assemblies, although there were significant differences in performance across samples. Among multiple k-mer assemblers, VelvetOptimiser frequently produced a very high number of contigs of very small size (over 99% of the contigs not surpassing the minimum length of 1,000 bp) and therefore a low N50 (an average of 29,768 bp versus a global average of 84,114 bp) (Supplementary Tables 5.6-5.11). Additionally, ABySS and VelvetOptimizer produced contigs with a very large number of Ns, with an average of 1,019 and 3,035 uncalled bases per assembly, respectively. MetaHipMer2, although having overall average metrics in the two evenly distributed mock samples (ENN and EMS, Supplementary Tables 5.6-5.7) where it was able to run successfully, it severely underperformed in the real samples (ERR2984773 and ERR2935805,

Figure 5.3: Assembly reproducibility. Inconsistent contigs produced per assembler over 3 LMAS runs. The distribution of contig sizes, in basepairs, consistently present in all three LMAS runs are indicated in the grey boxplots for each assembler. If an assembler produced a contig only present in two of the runs (as determined by its size), its size is indicated in teal. If a contig is present in a single run, it is represented in red.

Supplementary Tables 5.8 and 5.11). Generally, the performance scores of the assemblers decreased considerably for the real samples in comparison with the simulated ones, either with or without error, underscoring the importance of using mock samples instead of simulated reads to evaluate assembler performance. High utilisation of the reads in the dataset is observed for most assemblers, with on average at least 90% of the reads mapping back to the assembly, except for ABySS, MetaHipMer2 and VelvetOptimiser whose values are in the range of 46-79%. Despite an overall good performance, SPAdes produced the highest number of misassembled contigs in the logarithmically distributed sample, with an average of 98 and a maximum of 572 (sample ERR2935805, Supplementary Table 5.11), in comparison to the global average of 11 misassembled contigs for all assemblers across all samples. However, this behaviour was not consistent across samples, with the evenly distributed sample showing similar misassembled contigs between SPAdes and other assemblers, similarly to the other mock samples tested (see below).

Due to their poor performance discussed above, the following assemblers have not been included in subsequent analyses: ABySS, MetaHipmer2, minia and VelvetOptimiser.

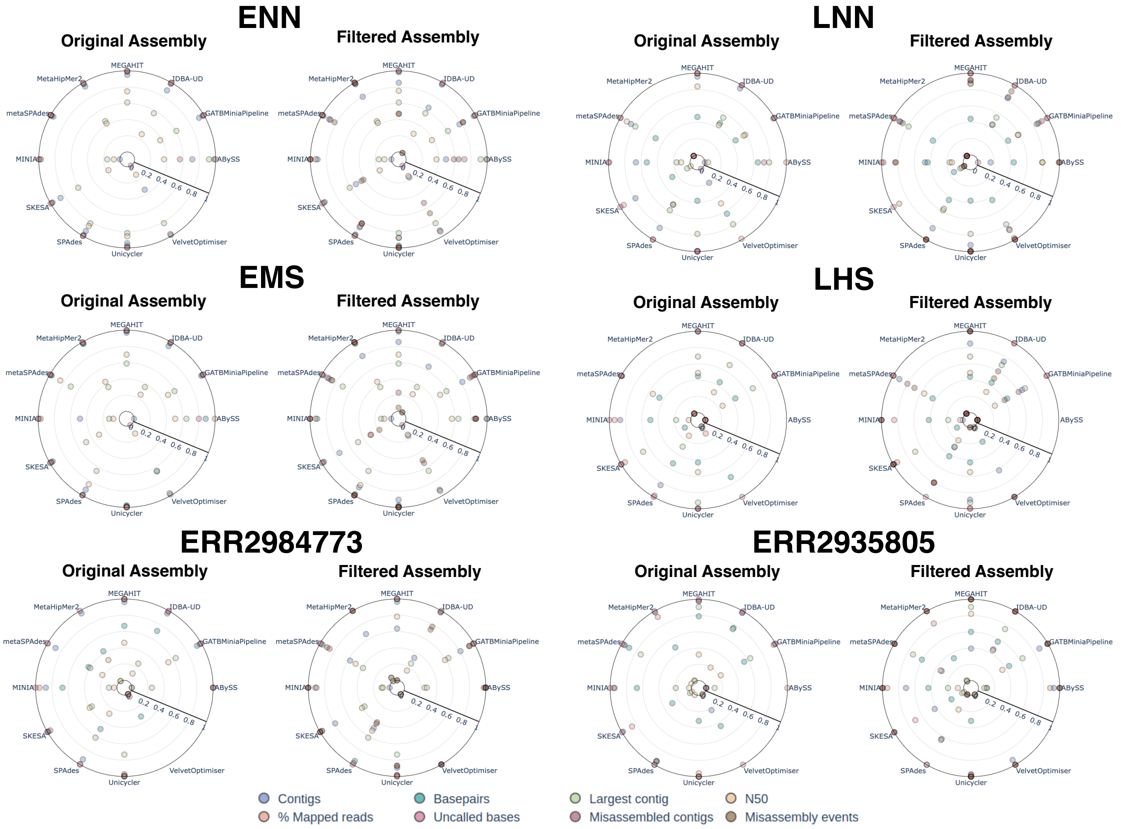


Figure 5.4: Assembler performance for the ZymoBIOMICS Microbial Community Standards dataset. For each sample in the dataset, the best score of each assembler in the 3 LMAS runs was selected. The results for each global assembly metric was normalised, with 1 representing the best result, and 0 the worst. For the original assembly, the following metrics are presented: number of contigs produced (in blue), number of basepairs produced (in teal), the size of the largest contig assembled (in green), N50 (in yellow), percentage of mapped reads to the assembly (in orange) and uncalled bases (in red). For the filtered assembly, the additional metrics are presented: number of misassembled contigs (in purple) and number of misassembly events (in brown).

5.4.2 Metagenomic dedicated assemblers do not outperform genomic assemblers

After excluding the poorly performing assemblers, LMAS includes 3 genomic (SKESA, SPAdes and Unicycler) and 4 labelled as metagenomic specific (GATBMiniaPipeline, IDBA-UD, MEGAHIT and metaSPAdes) de novo prokaryotic assemblers, all implementing multiple k-mer dBg algorithms. As observed in Figure 5.5, Supplementary Table 5.14 and Supplemental Figure 5.11, there were very significant differences between the best and the worst performing assemblers of each type for the ZymoBIOMICS Microbial Community Standards dataset, with this difference being more pronounced for metagenomic assemblers. The best performing assemblers of each type behaved frequently quite similarly, and the differences between them tended to be attenuated after filtering for contigs <1 kbp. Still, for the linearly distributed samples (ENN, EMS and ERR2984773), the overall worst performers tended to be metagenomic assemblers. In contrast, for the logarithmically distributed samples (LNN, LHS and ERR2935805) the opposite was observed, with genomic assemblers tending to be the worst-performing (Figure 5.5). For the logarithmically distributed samples, the number of basepairs recovered is significantly lower than expected from their composition for both genomic and metagenomic assemblers, particularly after filtering (Supplementary Table 5.14), as contigs representing the less abundant species are not recovered by either type of assemblers (see Assembler performance is influenced by replicon abundance in the sample). For this dataset, the fact that an assembler is branded as genomic or metagenomic does not translate into better or worse performance in dealing with these complex samples, but rather characteristics of the individual assemblers themselves determine their performance.

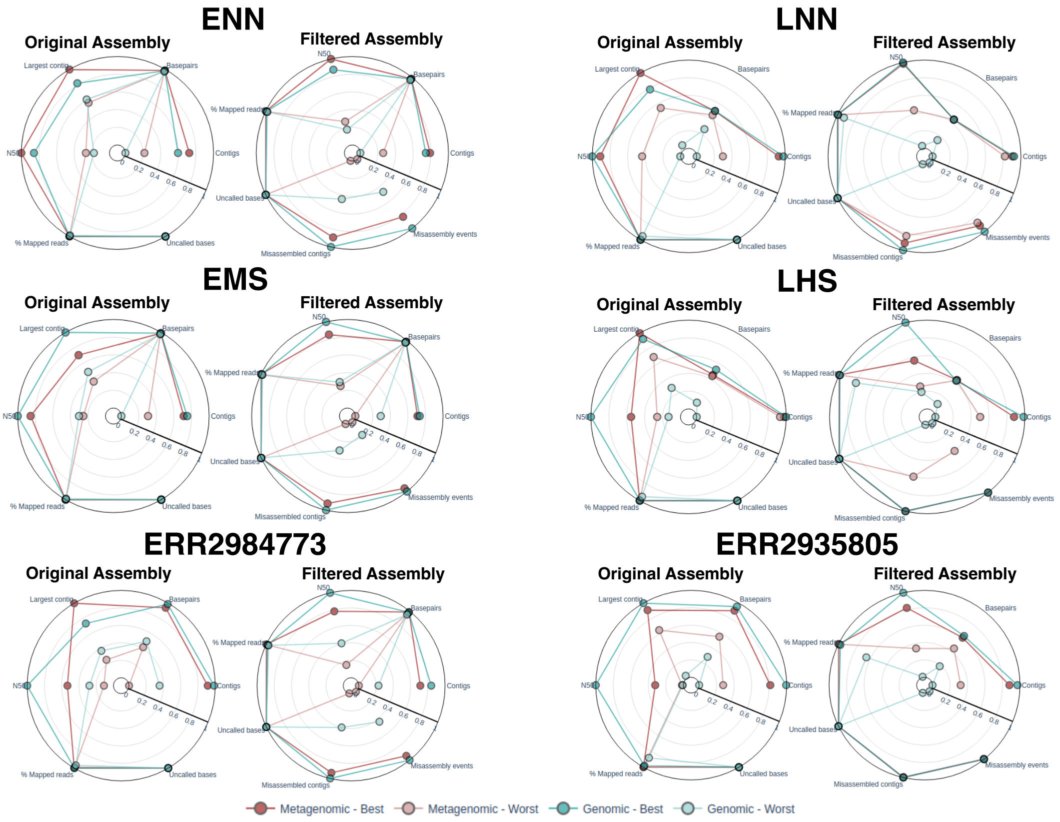


Figure 5.5: Performance of genomic and metagenomic assemblers for the ZymoBIOMICS Microbial Community Standards dataset. For each sample in the dataset and for the 3 runs, the best and worst scores for each assembler category were selected: genomic (in blue) and metagenomic (in red). The results for each global assembly metric were normalised, with 1 representing the best result, and 0 the worst. For the original assembly, the following metrics are presented: number of contigs produced, number of basepairs produced, the size of the largest contig assembled, N50, percentage of mapped reads to the assembly and uncalled bases. For the filtered assembly, the additional metrics are presented: number of misassembled contigs and number of misassembly events.

In the BMock standard, similarly to the ZymoBIOMICS standard, no significant difference was observed between the genomic and metagenomic assemblers, particularly after filtering for contigs <1 kbp (Supplemental Table 5.27). Both *Marinobacter* replicons (2615840697 and 2616644829; 448x and 135x coverage, respectively) were successfully recovered by all genomic and metagenomic assemblers with >0.87 breadth of coverage (Supplementary Table 5.28). The two most abundant *Micromonospora* replicons (2623620557 and 2623620567; 15x and 18x coverage, respectively) were also recovered to a breadth of coverage >0.95 by most assemblers, except the genomic assemblers SKESA and Unicycler (Supplementary Table 5.28). All assemblers, with the exception of SKESA and GATB-MiniaPipeline, recovered both Halomonas replicons (2623620617 and 2623620618) with a breadth of coverage of >0.84 (Supplementary Table 5.28). When considering this set of closely related replicons metagenomic assemblers also did not perform consistently better than genomic assemblers in the number of misassembled contigs or SNPs relative to the reference genome. Among the metagenomic assemblers, GATBMiniaPipeline and IDBA-UD performed particularly well but with values close to those of the two best genomic assemblers (SPAdes and Unicycler). IDBA-UB performed significantly worse with the *Micromonospora* replicons, possibly because of their lower coverage. As could have been expected, the number of SNPs in the lower coverage *Halomonas* replicon was consistently higher than in the one with higher coverage (1.6-fold to 85.0-fold depending on the assembler), despite their relatively modest (12.6%) difference in estimated coverage in the sample and very significant depth of coverage (>500x) (Supplementary Tables 5.25 and 5.28). However, when comparing the number of SNPs in the *Marinobacter* replicons, this relationship is reversed, with the replicon with higher coverage (448x) having more SNPs relative to the reference than the one with lower coverage (135x) (Supplementary Table 5.28). This indicates that other factors, such as characteristics of the replicon, the actual representation in the sample and the closeness to other replicons in the sample may influence the performance of assemblers. It is also interesting to see that the contigs generated by themselves do not allow the distinction of closely related strains, since the number of SNPs relative to the reference genomes (if the assembler is able to cover >0.79 of the genome), are in the ranges of 315-9,915 and 9,773-70,196 for the higher and lower depth of coverage *Halomonas* replicons, respectively.

For the Gut-Mix-RR and Gut-HiLo-RR mock communities, the same pattern was observed as with the other mock communities, with the differences between the metagenomic and genomic assemblers being attenuated after filtering for contigs <1 kbp (Supplemental Table 5.31 and **??**). Particularly for the evenly distributed Gut-Mix-RR sample, when considering the subset of *Roseburia* spp., the replicon with the lowest coverage (12x for *R. intestinalis* versus 18x for *R. hominis*) had a consistently higher number of SNPs, with the exception of the SKESA assemblies, where the opposite was observed. This is similar to what was observed for the *Halomonas* replicons in the BMock12 community standard.

5.4.3 Success is not straightforward

Several factors contribute to suboptimal performance of the assembly process, from DNA isolation and library preparation protocol; sequencing technology, depth and read length; to possible contamination and inherent characteristics of the sample composition.

5.4.3.1 Assembler performance is influenced by species

For the eight bacterial genomes present in the ZymoBIOMICS Microbial Community Standards dataset samples, even in those with an even distribution of the genomes (ENN, EMS and ERR2984773), variations in the assembly metrics were observed (Figure 5.6, Supplemental Figures 5.12-5.14, Supplemental Tables 5.15-5.17). For all samples in the dataset, the genomes are recovered almost completely, with all replicons being >90% represented in the resulting assemblies. *Lactobacillus fermentum* is the least represented genome (92.2%-94.9%). Most replicon sequences are recovered in <100 contigs, except for *Pseudomonas aeruginosa*, *Escherichia coli* and *Salmonella enterica*, and not considering IDBA-UB, which frequently produces a larger number of contigs when compared to other assemblers. However, in other mock samples these worse performance of IDBA-UD in terms of number of contigs is not so clear. The absolute values of other metrics of assembly quality, such as LSA, misassembly events or uncalled bases, are also different between bacterial genomes (Supplemental Tables 5.15-5.17). The fact that *S. enterica* is a closely related species to *E. coli*, with high level of genetic similarity (ANIb >0.8, Supplemental Table 5.24), could have created difficulties for resolving the assemblies in a mixed sample and lead to the lower coverage observed, the higher number of contigs and the increased number of misassembled contigs identified in these species in some samples. However, in the case of the larger number of contigs of *P. aeruginosa*, no related species are present in the sample and these possibly reflect intrinsic properties of the replicon such as a high number of prophages integrated in the bacterial genome [46]. Similarly, replicon characteristics could be behind the lower breadth of coverage consistently observed in *L. fermentum* assemblies.

In the BMock12 and the Gut-Mix samples, which have pairs of much more similar replicons, it is true that the closely related replicons do have higher numbers of contigs. However, it is possible that the properties of the individual replicons also have an impact on the number of contigs generated by the assemblers (Supplemental Figure S8). Another potential confounder which was not explored is the length of the reads, with Miseq samples having 300bp reads and HiSeq samples having 150bp reads.

5.4.3.2 Longer contigs have higher confidence

The Pls metric, which measures the error rate of a contig relative to the reference, shows that for every replicon, longer contigs have higher Pls (Figure 5.7). This could justify the option of filtering an assembly by length, even beyond the 1000 bp minimum contig size implemented by default in LMAS. Not only are we eliminating shorter, less informative contigs in terms of genetic context, but these are also the ones most likely to contain errors relative to the reference sequence.

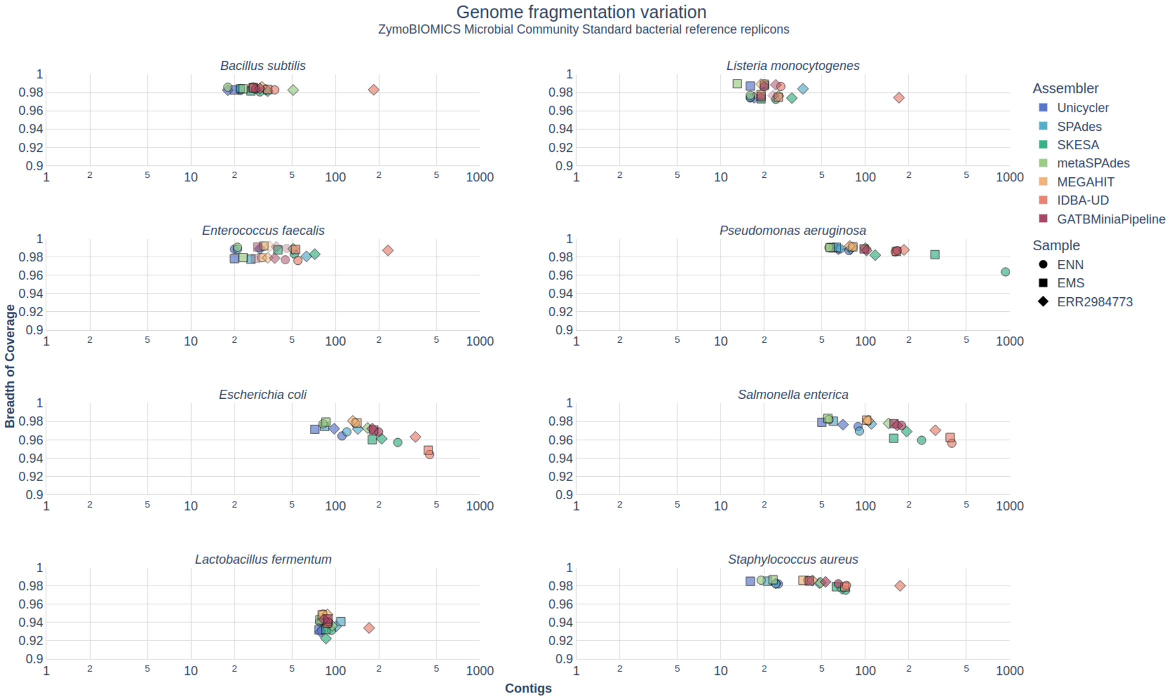


Figure 5.6: Genome fragmentation for each reference replicon of the ZimoBIOMICS community standards dataset for the evenly distributed samples. Genome fragmentation for the 3 LMAS runs is represented by the number of contigs and breadth of coverage of the reference per assembler for the evenly distributed samples: ENN (evenly distributed without error model, identified by a circle), EMS (evenly distributed with Illumina MiSeq error model, identified by a square) and ERR2984773 (real Illumina MiSeq sample, identified by a diamond). Each assembler is identified with the following colour scheme - dark blue: Unicycler, light blue: SPAdes, dark green: SKESA, light green: metaSPAdes, yellow: MEGAHIT, orange: IDBA-UD, red: GATB-MiniaPipeline.

5.4.3.3 Longer contigs have higher confidence

Some genomic regions in several replicons are consistently a challenge for all assemblers. As observed in Figure 5.8, all genomes of the ZymoBIOMICS Microbial Community Standards dataset present certain regions that fail to assemble for all tools in all runs, even those generating high-quality draft assemblies. Of all seven assemblers considered, only GATB-MiniaPipeline, MEGAHIT and IDBA-UD showed inconsistency in the gaps produced over the 3 LMAS runs (Supplemental Table 5.18), as expected from producing variable sets of contigs. The regions consistently missing for all assemblers in all runs are rich in repetitive elements, such as rRNA and tRNA coding sequences and mobile genetic elements (Supplemental Table 5.19), with larger gaps corresponding to tandem sets of these elements. This reflects an intrinsic limitation of short-read sequencing since the length of a read pair is not enough to bridge across the repetitive element, preventing the generation of contigs representing these regions. This is something that could be addressed by the use of long-read sequencing technologies. Despite this, some assemblers are able to produce contigs that represent some of these large tandem regions, such as MEGAHIT and SKESA for *E. faecalis*, and IDBA-UD, MEGAHIT and metaSPADES for *L. monocytogenes*, but such performance is not consistent for all reference replicons. For instance, SKESA fails to assemble two large

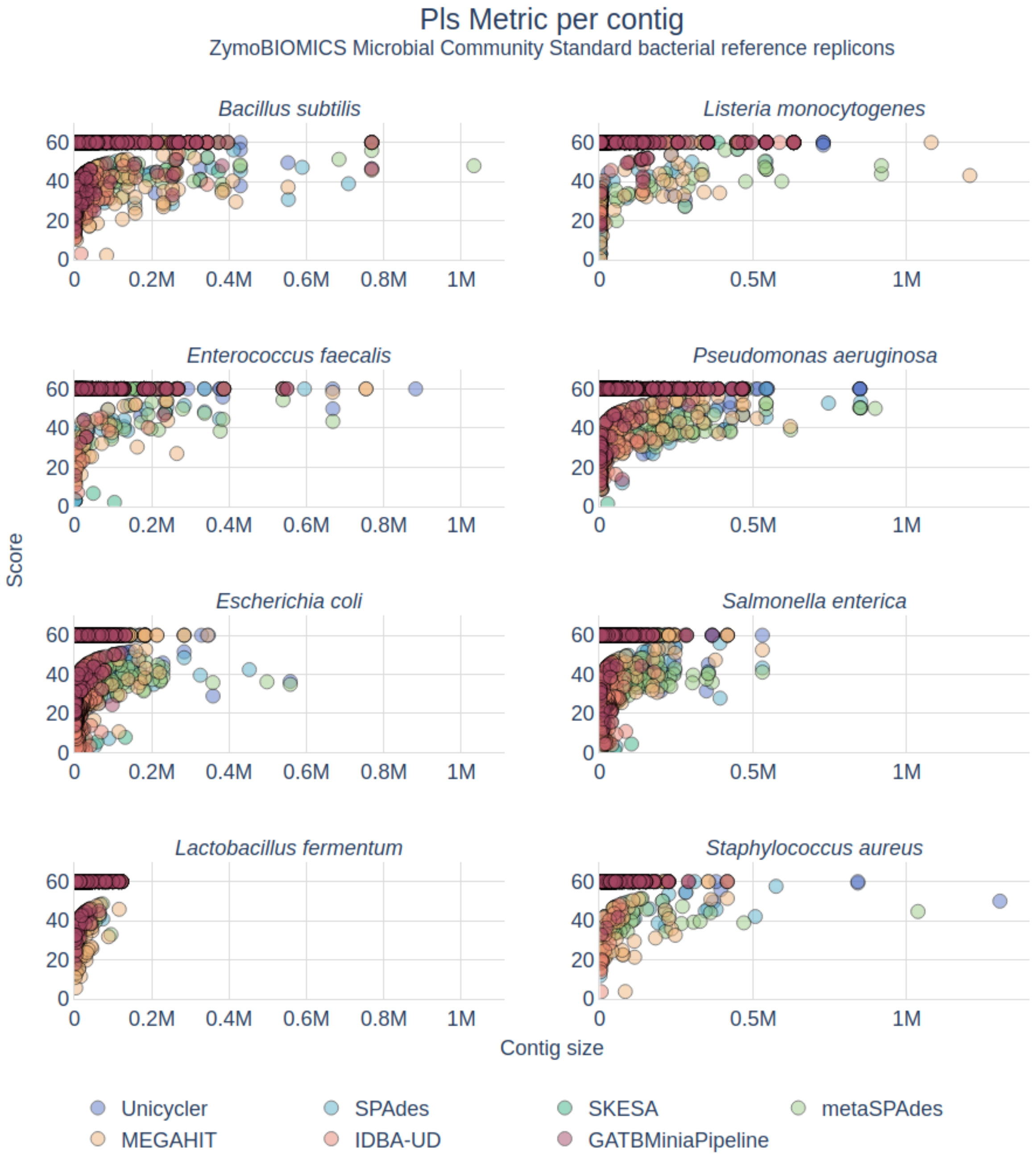


Figure 5.7: : Phred-like score (Pls) per contig for each reference replicon of the ZimoBIOMICS community standards datasets. The Pls score was calculated for each unique contig produced by each assembler in 3 LMAS runs and is represented in relation to its contig size. Each contig is coloured according to the assembler with the following colour scheme - dark blue: Unicycler, light blue: SPAdes, dark green: SKESA, light green: metaSPAdes, yellow: MEGAHIT, orange: IDBA-UD, red: GATBMiniaPipeline.

regions of the *S. enterica* genome that all other assemblers successfully cover.

For the BMock12 community standard, the same pattern of consistency of difficult regions across replicons can be observed for all replicons, with the exception of the lowest abundance replicons (see Supplemental Figure S9). Interestingly, the two closely related *Halomonas* replicons present a very dissimilar gap pattern, and with a high number of gaps (n=2789 and n=2702) distributed throughout the replicon sequence, which possibly reflects

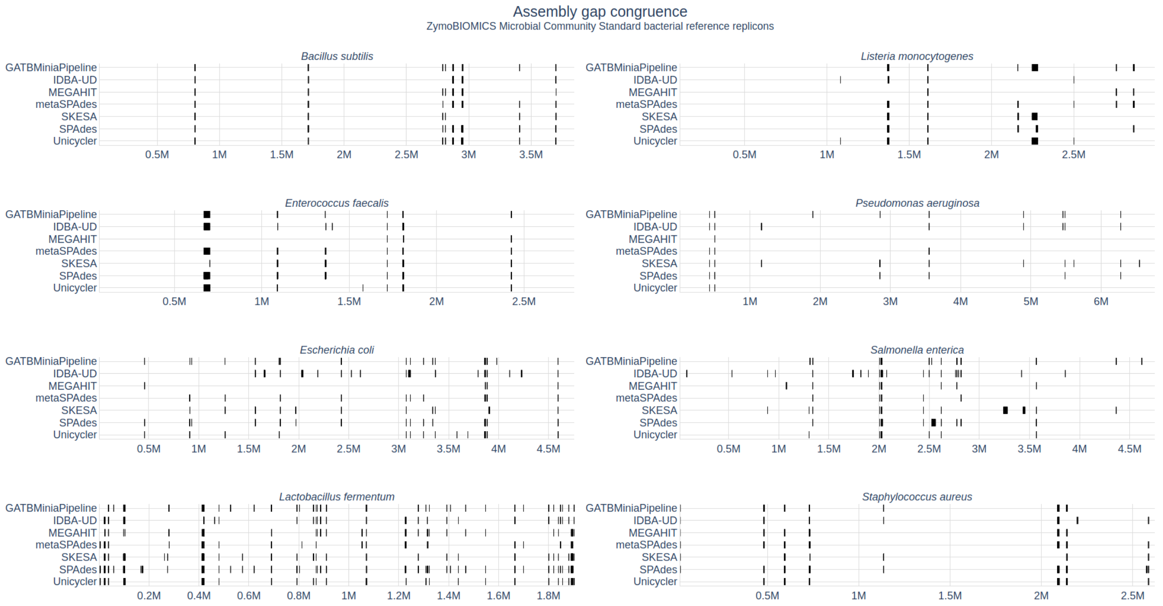


Figure 5.8: Location of gaps in comparison to the reference sequence, per assembler, for each reference replicon of the ZimoBIOMICS community standards datasets. The resulting plot contains the consistent gaps obtained from a three LMAS run for the evenly distributed dataset (ENN, EMS and ERR2984773) for GATB-MiniaPipeline, IDBA-UD, MEGAHIT, metaSPAdes, SKESA, SPAdes and Unicycler assemblers.

the difficulty of assembling closely related replicons in the same sample.

5.4.3.4 Assembler performance is influenced by replicon abundance in the sample

The logarithmically distributed samples (LNN, LHS and ERR2935805) of the ZymoBIOMICS community standard dataset showed greater variation in the assembly success metrics than the evenly distributed samples (Supplementary Table 5.9-5.11), reflecting the difficulty of recovering sequences of the lowest abundant replicons. For the three replicons with an estimated depth of coverage >15x, a similar pattern is observed in logarithmically distributed samples as in evenly distributed samples, albeit with greater dispersion in the number of contigs generated and with a markedly decreased breath of coverage for some assemblers and samples in the logarithmically distributed samples (Figure 5.6 and Supplementary Figure 5.15). Almost no contigs >1000 bp were retrieved for replicons with an estimated depth of coverage of <2x resulting in a very low breadth of coverage (<1%) (Supplementary Table 5.5, Supplementary Table 5.23). This leads to a severe underrepresentation of the diversity of the community in the generated contigs, particularly of plasmid sequences due to their smaller length and abundance as was described previously [3, 43, 47]. This happens despite the greater sequencing depth of these samples versus those with an even distribution (>5-fold difference in the number of reads).

For the BMock12 community standard, the very low replicon abundance (*Micromonospora coxensis*, 2623620609, 0.02x coverage), fails to assemble in all tools (Supplemental Figure S9, Supplemental table S24). The *Micromonospora echinaurantiaca*

(2623620557, 14.9x coverage) and *Micromonospora echinofusca* (2623620567, 18.2x coverage) fail to assemble with SKESA and Unicycler, and the *Propionibacteriaceae* replicon (2615840646, 31.9x coverage) fails to be assembled with SKESA. In the Gut-Mix-RR standards there are several replicons with <1x to 20x depth of coverage. Significant breath of coverage (>0.7) was obtained with the contigs created by most assemblers, including successful assemblies of some of the replicons by SKESA and Unicycler that had failed with higher coverage replicons in the BMock12 standard. Taking together these data and that of the ZymoBIOMICS standard, these suggest that it is hard to establish a universal breakpoint at which each of the assemblers is able to generate high breadth of coverage contigs, with the actual genome of interest and the composition of the sample possibly playing a role. Nevertheless, coverages >15x result in high breadth of coverage contigs, albeit in the lower range with many contigs and a significant number of SNPs.

5.5 Conclusions

The purpose of LMAS is to empower users to test assembler performance in meaningful conditions for their experimental setup and objectives. Suitable mock communities, reproducing the users' samples of interest, can be used as a gold standard to evaluate assembler performance. To illustrate LMAS' functionalities we analysed three well-known samples used in several studies. Although the eight species ZymoBIOMICS Microbial Community Standards might not be representative of the metagenomic complexity of the samples of interest of most researchers,we hoped that its relative simplicity meant that the results shown would represent a best-case scenario, since as sample complexity increases so do the challenges to assembler performance. However, the results of the BMock12 and Gut-Mix community standards suggest that the actual genome of interest and community composition play an important part in the results of individual assemblers.

Our results showed significant differences in both global and reference-dependent assembly quality metrics generated by each de novo assembler. The performance of each assembler varied depending on the species of interest and its abundance in the sample, with less abundant species presenting a significant challenge for all assemblers. The fact that an assembler is branded as specific for metagenomics does not guarantee a better performance in metagenomic samples, with assemblers used for genomic assembly outperforming the worst metagenomic assembler tested. Even when considering communities with very similar replicons, the overall performance of metagenomic assemblers was not consistently better than that of genomic assemblers. The results also indicate that the recovery of assemblies allowing strain-level discrimination at the SNP level is highly unlikely based solely on the assembler generated contigs. The following assemblers showed significant performance problems and their usability may be limited, at least with the default parameters we used: ABySS, MetaHipmer2, minia and VelvetOptimiser.

The choice of de novo assembler depends greatly on the computational resources available, the species of interest, its representation in the sample, and, possibly, the composition of the community in the sample. In our testing with any of the mock communities, no assembler stood out as an undisputed all-purpose choice for short-read metagenomic prokaryotic genome assembly, with different assemblers showing specific strengths. Users would thus benefit from analysing the results of sequencing mock communities (ideally) or of artificially generated reads simulating their samples of interest to guide their choice of assembler. LMAS was developed to be an easy to use and flexible tool for this purpose. From the results that we obtained with various mock communities, the following assemblers performed consistently well (presented in alphabetical order): MEGAHIT, metaSPAdes, SKESA, SPAdes and Unicycler. From our assessment, we conclude that these assemblers are the most likely candidates to perform well in other complex samples.

LMAS was built with modularity and containerization as keystones, leveraging the parallelization of processes and guaranteeing reproducibility across platforms. The modular design allows for new assemblers to be easily added and existing assemblers to be easily updated, allowing LMAS to function as a continuous benchmarking platform and ensuring its future relevance as improvements in assembly software are proposed. LMAS will also support evaluating the gains of any cumulative improvements to existing assemblers using the same benchmark set adapted to a specific project or goal. Such reproducibility, capacity to easily add assemblers of interest not included in the current version and flexibility for future extensions are important principles in computational method benchmarking. Moreover, users may compare software performance against mock communities of special interest, depending on their operational focus. Moreover, by lowering the barriers to perform comparisons between assemblers, LMAS will encourage users may compare software performance against mock communities of special interest, depending on their operational focus.

The interactive report provides an intuitive platform for data exploration, allowing the user to easily sift through global and reference specific performance metrics for each sample, as well as providing information on the assemblers executed to allow traceability of the results. Producing an extensive, metric rich report allows users interested in different aspects of assembler performance to make informed decisions, particularly when choosing among the top-performing assemblers, which show only minor differences.

LMAS applies several well-known assembly metrics and proposes two more: LSA, which represents the fraction of the longest single alignment between a contig and the reference, and Pls, a scoring function based on the identity of each aligned contig to the reference replicon. The entire set of assembly quality metrics used in LMAS allows not only the assessment of quality based on statistics inherent to a set of assembled contigs but also a comparison to a ground truth provided through the use of samples of known composition and reference sequences. The LMAS report provides an interactive and intuitive platform for the exploration of these results, allowing users to easily test assemblers in mock samples with species composition and distribution relevant for their own studies.

Although computationally intensive due to the complex nature of the de novo assembly process, LMAS is the only software integrating assembly and its evaluation into a single pipeline, guaranteeing the same conditions are met for all tools. With LMAS, it is now possible to evaluate which de novo assembler produces the most relevant results for a given community of interest. The LMAS workflow is open-source and its code and documentation are available at `https://github.com/B-UMMI/LMAS` and `https://lmas.readthedocs.io/` respectively.

5.6 Availability of supporting source code and requirements

Project name: LMAS

Project home page: https://github.com/B-UMMI/LMAS

Operating system(s): UNIX-like systems. Programming languages: Nextflow, Python, Bash, Javascript

Other requirements: Java version 8 or highest. Docker/Singularity/Shifter

License: GNU GPL v3

RRID: SCR_022251

5.7 Declarations

5.7.1 Ethics approval and consent to participate

Not applicable.

5.7.2 Consent for publication

Not applicable.

5.7.3 Availability of data and material

The datasets analysed during the current study are available in the Zenodo repository, under `https://doi.org/10.5281/zenodo.4588969`. All supplemental material is available in the Zenodo repository, under `https://doi.org/10.5281/zenodo.6623457`. Likewise, all figures in the current manuscript are available in their original format in the Zenodo repository, under `https://doi.org/10.5281/zenodo.6783042`. Real sequencing data of the ZymoBIOMICS Microbial Community Standards is available under accessions ERR2984773 and ERR2935805 [39]. All data generated or analysed during this study are included in this published article, its supplementary information files and the data analysis repository located at [48]. Additionally, the reports for the ZymoBIOMICS Microbial Community Standard, BMock12 Community Standard and NIBSC Gut DNA Reference are available at `https://doi.org/10.5281/zenodo.7088960`, `https://doi.org/10.5281/zenodo.7092431` and `https://doi.org/10.5281/zenodo.7092693` respectively.

5.7.4 Competing interests

MR received honoraria for serving on the speakers' bureau of Pfizer and Merck Sharp and Dohme and for participating in expert panels of GlaxoSmithKline and Merck Sharp and Dohme. The other authors declare that they have no competing interests.

5.7.5 Funding

C.I.M. was supported by the Fundação para a Ciência e Tecnologia (grant SFRH/BD/129483/2017).

5.7.6 Author's contributions

C.I.M., M.R. designed the workflow. C.I.M implemented and optimised the workflow, created the Docker containers, generated mock shotgun metagenomics data used to test and validate the workflow, contributed to the development of the HTML report and analysed the data. C.I.M. and M.R. wrote the manuscript. P.V.C. contributed to the development of the HTML report. M.R., J.A.C. Y.M, and J.M.G critically revised the manuscript. All authors read, commented on, and approved the final manuscript.

5.7.7 Acknowledgements

The authors would like to thank Rafael Mamede for his contribution to the implementation and commentary on the several interactive plots implemented throughout the LMAS report. The authors would also like to thank Nabil Fareed-Alikan for his insightful commentary on the interpretation of the results reported in this manuscript, and Anthony Underwood and Robert A. Petit III for their assistance in building the LMAS Nextflow workflow. The author would also like to thank Samuel Nicholls, Joshua Quick, Shuiquan Tang and Nicholas Loman for publicly providing the sequencing data for the ZymoBIOMICS Microbial Community Standards. Likewise, the authors would like to thank Volkan Sevim, Juna Lee, Robert Egan, Alicia Clum, Hope Hundley, Janey Lee, R. Craig Everroad, Angela M. Detweiler, Brad M. Bebout, Jennifer Pett-Ridge, Markus Göker, Alison E. Murray, Stephen R. Lindemann, Hans-Peter Klenk, Ronan O'Malley, Matthew Zane, Jan-Fang Cheng, Alex Copeland, Christopher Daum, Esther Singer Tanja Woyke for providing the sequencing data for the BMock12 Commynity Standard, and to Gregory C. A. Amos, Alastair Logan, Saba Anwar, Martin Fritzsche, Ryan Mate, Thomas Bleazard Sjoerd Rijpkema for providing the sequencing data of the NIBSC Gut DNA Reference Gut-Mix-RR and Gut-Mix-HiLo Community Standard.

5.8 Supplemental Materials

5.8.1 Workflow parameters

In LMAS, a set of default parameters is provided but these can be altered, either by passing the new value when executing the workflow or by editing the "params.config" file in the "configs" folder. There are three main parameters in LMAS: "reference", "fastq" and "md". The short-read data is passed as input through the "–fastq" parameter, which by default is set to match all files in the "data/fastq" folder that match the pattern "*_R1,2*". The reference sequences in a single file can be passed with the "–reference" parameter, matching by default fasta files (with the pattern "*.fasta") in the "data/reference" folder. Although not mandatory, text information, in a markdown file, on input samples can be passed to LMAS to be presented in the report with the "–md" parameter. By default, this is matched to the "*.md" pattern in the "data" folder.

Several options are available to alter the behaviour of the assemblers incorporated in LMAS, namely to alter the values of the k-mer for each assembly iteration, as detailed in the documentation [**empty citation**]. By default, these values reflect the corresponding default settings of the assemblers. Additionally, each assembler can be skipped from the workflow, and the resources for the execution, such as CPUs, memory and time limit, can be altered for all assembly processes. For the assembly quality assessment performed by LMAS, the following parameters are provided and can be adjusted:

- **“–minLength”:** Value for minimum contig length, in basepairs. By default, this value is set to 1000 basepairs;
- **“–mapped_reads_threshold”:** Value for the minimum percentage of a read aligning to the contig to be considered as mapped. By default, this value is set to 75%;
- **“–n_target”:** Target value for the N, NA and NG metrics, ranging from 0 to 100%. By default, this value is set to 50%;
- **“–l_target”:** Target value for the L metric, ranging from 0 to 100%. By default, this value is set to 90%;

5.8.2 Short-read de novo assemblers

We have compiled a collection of de novo assembly tools, including OLC and dBg assembly algorithms, with both single k-mer and multiple k-mer value approaches, and hybrid assemblers (Supplemental Table 5.2). The collection includes both genomic and metagenomic assemblers, developed explicitly to handle metagenomic datasets. The dates of the last release correspond to the ones available in the preparation of this manuscript.

5.8.2.1 Selection Criteria

Only open-source tools, with clear documentation describing the methodology implemented, were considered. The collection of tools was ordered by the date of the last update, and a Docker container [49] for the top 11 assemblers was created with the latest released version, with the version used as the tag. In the case of tools where a versioned release is not available, the container was created with the latest version in the default branch of the source repository, using the date of the last update as the tag. The PANDAseq [50] assembler was excluded due to execution errors.

5.8.2.2 Assemblers in LMAS

Assemblers benchmarked in LMAS, in alphabetical order:

5.8.2.2.1 ABySS

The ABySS assembler [18] is a de novo sequence assembler intended for short paired-end reads and genomes of all sizes. It follows the model of minia, wherein a probabilistic Bloom filter representation is used to encode the de single k-mer size Bruijn graph, reducing memory requirements for de novo assembly. The code is open-source and available at [51]. The following command is used: “abyss-pe name=’$sample_id’k=$KmerSize B=$BloomSize in=’$fastq”, where “$sample_id” contains the identifier of the sample, contains a list of the input read files, “$sample_id” the identifier of the sample, “$KmerSize” the length of the nodes of the graph (by default set to 96), “$BloomSize” the size, in Gb, of the bloom filter (by default set to 2 GB), and “$fastq” the forward and reverse fastq files.

5.8.2.2.2 GATB-Minia Pipeline

GATB-Minia is an assembly pipeline, still unpublished, that consists of Bloocoo [8] for error correction, minia 3 [24] for contigs assembly, which is based on the BCALM2 assembler [52], and BESST [53] for scaffolding. It was developed to extend the minia assembler to use the dBg algorithm with multiple k-mer values and to explicitly handle metagenomic data. The code is open-source and available at [19]. The following command is used: “gatb -1 $fastq_pair[0] -2 $fastq_pair[1] –kmer-sizes $kmer_list -o $sample_id”, where $fastq_pair[0] contains the forward-facing reads, $fastq_pair[1] the reverse-facing reads, $kmer_list the list of values for length of the nodes of the dBg (by default set to 21,61,101,141,181), and “$sample_id” the identifier of the sample.

5.8.2.2.3 IDBA-UD

IDBA-UD [20] is a dBg graph assembler for assembling reads from single-cell sequencing or metagenomic sequencing technologies with uneven sequencing depths. It employs multiple depth relative thresholds to remove erroneous k-mers in both low-depth and high-depth regions. The technique of local assembly with paired-end information is used to solve the branch problem of low-depth short repeat regions. To speed up the process, an error correction step is conducted to correct reads of high-depth regions that can be aligned to high confidence contigs. The code is open-source and available at [54]. The following command is used: “idba_ud -l $fasta_reads_single”, where $fasta_reads_single contains the combined sequence data converted to FASTA format reads with “reformat.sh” from BBtools [55].

5.8.2.2.4 MEGAHIT

MEGAHIT [21] is a de novo assembler for large and complex metagenomics datasets. It makes use of the succinct dBg, with a multiple k-mer size strategy. In each iteration, MEGAHIT cleans potentially erroneous edges by removing tips, merging bubbles and removing low local coverage edges, especially useful for metagenomics which suffers from non-uniform sequencing depths. The code is open-source and available at [56]. The following command is used: “megahit -o megahit –k-list $kmers -1 $fastq_pair[0] -2 $fastq_pair[1]”, where $kmers contains the list of values for length of the nodes of the dBg (by default set to 21,29,39,59,79,99,119,141), $fastq_pair[0] contains the forward-facing reads, and $fastq_pair[1] the reverse-facing reads.

5.8.2.2.5 MetaHipMer2

MetaHipMer2 [22] is a multiple k-mer size dBg de novo metagenome short-read assembler built to run efficiently on both single servers and on multi-node supercomputers, where it can scale up to coassemble terabase-sized metagenomes. The code is open-source and available at [57]. The following command is used: “mhm2.py -k $kmers -r $fasta_reads_single -s 0”, where $kmers contains the list of values for length of the nodes of the dBg (by default set to “21,33,55,77,99”), where $fasta_reads_single contains the combined sequence data converted to FASTA format reads with “reformat.sh” from BBtools [55]. The “-s 0” option skips the scaffolding step.

5.8.2.2.6 metaSPAdes

SPAdes [19] started as a tool aiming to resolve uneven coverage in single-cell genome data, with metaSPAdes [23] later released building a specific metagenomic pipeline on top of SPAdes. It uses multiple k-mer sizes of dBg, starting with the lowest kmer size and adding hypothetical k-mers to connect the assembly graph. The code is open-source and available at [58]. The following command is used: “metaspades.py –only-assembler -k $kmers -1 $fastq_pair[0] -2 $fastq_pair[1]”, where $kmers contains the list of values for length of the nodes of the dBg (by default set to “auto”), $fastq_pair[0] contains the forward-facing reads, and $fastq_pair[1] the reverse-facing reads.

5.8.2.2.7 minia

Minia [24] performs the assembly on a data structure based on unitigs produced by the BCALM [52] software and using graph simplifications that are heavily inspired by the SPAdes assembler [26]. Minia is a short-read traditional assembler based on dBg graph using a single k-mer length. The code is open-source and available at [59]. The following command is used: “minia -in $list_reads -out $sample_id”, where “$list_reads” contains a list of the input read files and “$sample_id” the identifier of the sample.

5.8.2.2.8 SKESA

SKESA [25] is a de novo sequence read assembler that is based on dBg and uses conservative heuristics. It is designed to create breaks at repeat regions in the genome, creating shorter assemblies but with greater sequence quality. It tries to obtain good contiguity by using multiple k-mers longer than mate length and up to insert size. The code is open-source and available at `https://github.com/ncbi/SKESA`. The following command is used: "skesa –use_paired_ends –contigs_out $sample_id –fastq $fastq_pair[0] $fastq_pair[1]", where "$sample_id" refers to the identifier of the sample, $fastq_pair[0] contains the forward-facing reads, and $fastq_pair[1] the reverse-facing reads.

5.8.2.2.9 SPAdes

SPAdes [26] is an assembly tool aiming to resolve uneven coverage in single-cell genome data through multiple k-mer sizes of dBgs. It starts with the smallest k-mer size and adds hypothetical k-mers to connect the graph. The code is open-source and available at [58]. The following command is used: "spades.py –only-assembler -k $kmers -1 $fastq_pair[0] -2 $fastq_pair[1] ", where $kmers contains the list of values for length of the nodes of the dBg (by default set to "auto"), $fastq_pair[0] contains the forward-facing reads, and $fastq_pair[1] the reverse-facing reads.

5.8.2.2.10 UNICYCLER

Unicycler [27] is an assembly pipeline for bacterial genomes that can do long-read assembly, hybrid assembly and short-read assembly. When assembling Illumina-only read sets, it functions as a SPAdes-optimiser, using a dBg algorithm with multiple k-mer values. The code is open-source and available at [60]]. The following command is used: "unicycler -o . –no_correct –no_pilon -1 $fastq_pair[0] -2 $fastq_pair[1]", where $fastq_pair[0] contains the forward-facing reads, and $fastq_pair[1] the reverse-facing reads.

5.8.2.2.11 VELVETOPTIMIZER

This optimising pipeline of the Velvet assembler [61] is still unpublished but extends the original tool by performing several dBg assemblies with variable k-mer sizes. It searches a supplied hash value range for the optimum, estimates the expected coverage and then searches for the optimum coverage cutoff. It uses Velvet's internal mechanism for estimating insert lengths for paired-end libraries. The code is open-source and available at [28]. The following command is used: “VelvetOptimiser.pl -v -s $velvetoptimizer_hashs -e $velvetoptimizer_hashe -f ’-shortPaired -fastq.gz -separate $fastq_pair[0] $fastq_pair[1]’”, where $velvetoptimizer_hashs is the lower end of the hash value range that the optimiser will search for the optimum (default: 19), $velvetoptimizer_hashe is the upper end of the hash value range that the optimiser will search for the optimum (default: 31), $fastq_pair[0] contains the forward-facing reads, and $fastq_pair[1] the reverse-facing reads.

5.8.3 Misassembly detection

For the detection of misassembly events in the assemblies, the assembled sequences are first filtered for a minimum sequence length with BBTools [55] (version 38.44), as defined in the parameters, using the following command: “reformat.sh in=$assembly out=filtered_$assembly minlength=$minLen”, where $assembly contains the file with the assembled sequences and $minLen the value of the minimum sequence length allowed.

The filtered assembled sequences are mapped against the tripled reference replicons, ensuring that the assembled contigs can fully align regardless of their starting position relative to that of the provided reference sequence. This is done with minimap2 [31] (version 2.22) with the following parameters: “minimap2 –cs -N 0 -t -r 10000 -g 10000 -x asm20 –eqx”.

5.8.4 Assembly filtering and mapping

The assembled sequences are first filtered for a minimum sequence length with BBTools [14] (version 38.44), as defined in the parameters, using the following command: “reformat.sh in=$assembly out=filtered_$assembly minlength=$minLen”, where $assembly contains the file with the assembled sequences and $minLen the value of the minimum sequence length allowed.

The filtered assembled sequences are mapped against the tripled reference replicons, as explained above, with minimap2 [31] (version 2.22) with the following parameters: “minimap2 –cs -N 0 -t -r 10000 -g 10000 -x asm20 –eqx”.

5.8.5 LMAS Metrics

The following metrics are computed by the LMAS workflow, globally for characteristics intrinsic to the assembled contigs, and relative to the replicons present in the sample.

5.8.5.1 Global Metrics

5.8.5.1.1 General contig information

The following metrics are computed and presented in tabular form:

- **Contigs:** The total number of contigs in the assembly;
- **Basepairs:** The total number of bases in the assembly;
- **Maximum sequence length:** The length of the largest contig in the assembly;
- **Number of 'N's:** Number of uncalled bases;
- **Mapped reads:** Percentage of mapped reads to the assembly;

For each plot, the following metrics are presented:

- **Contig size distribution per assembler:** For each assembler in LMAS, a boxplot is computed representing the size distribution of contigs that align to any of the reference replicons. The unmapped contigs, if present, are represented in a red scatterplot overlapping the boxplot.
- **Gap size distribution per assembler:** For each assembler in LMAS, a boxplot is computed representing the distribution of gap sizes. Gaps are calculated after aligning all contigs to the reference replicons. All gaps 1 basepair in length are considered.

5.8.5.1.2 Contiguity

The following metrics are computed and presented in tabular form:

- **Nx (where $0 < x \leq 100$):** Length for which the collection of all contigs of that length or longer in an assembly covers at least a given percentage of the total length of the assembly

5.8.5.1.3 Misassemblies

A misassembly event is defined as a continuously assembled contig being broken into multiple non-collinear blocks when mapping to the reference replicons, i.e. the contig produced by the assembler does not preserve the exact synteny observed in the reference replicon. This may reflect the addition or deletion of sequence stretches or the shuffling of sequence blocks relative to the reference replicons. For a large insertion or deletion to be considered it must be $\geq$50 basepairs in length [62]. This metric is computed for the filtered set of contigs, i.e. those of length above a user-specified minimum size and mapping to the reference replicons (see 5.8.3). The misassemblies are processed with custom python code.

The following misassembly types are identified:

- **Chimera:** a contig has two or more sequence blocks mapping to different reference replicons;
- **Insertion:** a sequence block ($\geq$50 basepairs) which is not present in any of the reference replicons has been introduced into the contig by the assembly process;
- **Deletion:** a sequence block ($\geq$50 basepairs) of the reference replicon is missing from the contig created by the assembly process;
- **Inversion:** a contig has at least two sequence blocks mapping to the same replicon but reversed end to end, i.e. one of the blocks maps to the sense strand and the other to the antisense strand in the reference replicon while both are in the same strand in the contig, or vice-versa;
- **Rearrangement:** a contig has at least two sequence blocks mapping to the same replicon, in the same orientation, in a different order than in the reference sequence;
- **Translocation:** a contig has at least two sequence blocks abutting in the contig but mapping non-collinearly (over 1000 base pairs apart) in the reference replicon;
- **Duplication:** a sequence block of a contig maps at least twice to the reference replicon in different alignment blocks;
- **Inconsistency:** a contig has at least two sequence blocks abutting in the contig but fails to be classified in any of the previous categories.

Figure 5.9 provides a visual description of the detected misassemblies. The following metric is computed and presented in tabular form:

- **Misassembled contigs:** Number of contigs with misassembly events
- **Misassembly events:** Total number of misassemblies in the contigs

In the plot, the metrics are presented for the filtered set of contigs:

- **Misassembled contigs:** Scatter plot for misassembled contigs per assembler, the size of the misassembled contigs, and the number of blocks created by the misassembly in the contig. The distribution of contig size for all misassembled contigs is represented in a boxplot. Information on the misassembly is presented as a hover text for each misassembly event.

5.8.5.2 Per Reference Metrics

5.8.5.2.1 General contig information

The following metrics are computed and presented in tabular form:

- **Contigs:** The total number of contigs in the assembly that align to the reference replicon;
- **Basepairs:** The total number of bases in the assembly that align to the reference replicon;
- **Number of 'N's:** Number of uncalled bases (N's) in the contigs that align to the reference replicon.

5.8.5.2.2 COMPASS

A measure of the quality of a replicon assembly can be considered the proportion of the reference covered by the contigs, i.e. the breadth of coverage of the reference replicon. The COMPASS metrics [9] complement our view of the quality of the assembly with other metrics such as how much redundancy is there in the assembly or the parsimony of the contigs relative to the reference. COMPASS is composed of the following metrics, presented in tabular form:

- **Breadth of Coverage:** Ratio of covered sequence on the reference by aligned contigs;
- **Multiplicity:** Ratio of the length of the alignable assembled sequence to covered sequence on the reference;
- **Validity:** Ratio of the length of the alignable assembled sequence to total basepairs in the aligned contigs;
- **Parsimony:** Cost of the assembly (multiplicity over validity);

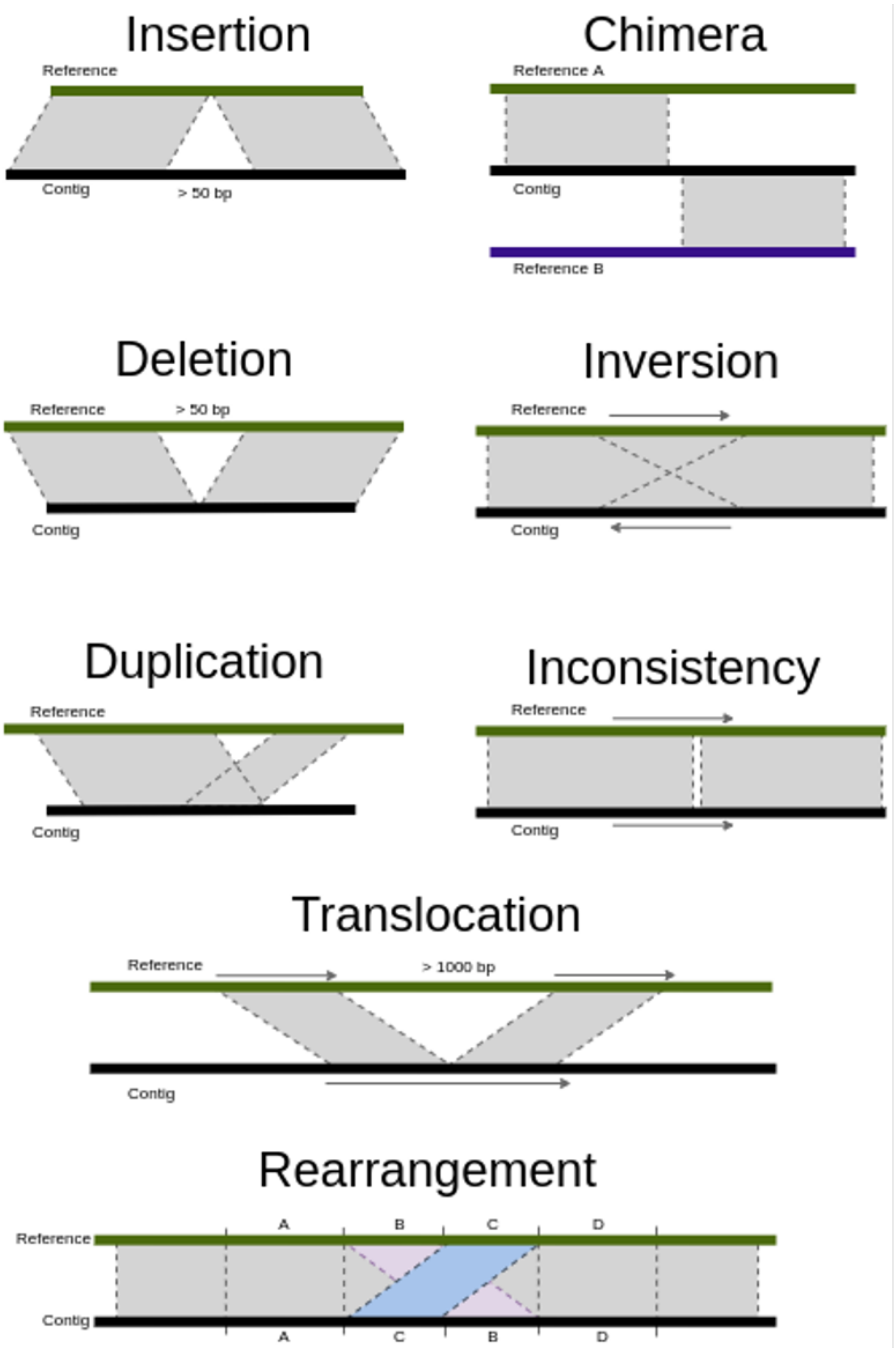


Figure 5.9: LMAS misassembly classification. Misassembled contigs are classified into 6 main categories: chimera, insertion, deletion, inversion, rearrangement, translocation and duplication, according to the mapping orientation, the distance between blocks in the contig and the mapping coordinates in the reference replicon. If a contig is classified as being chimeric, no further classification is performed. The other categories are classified independently of each other, with combinations being possible, to better reflect the differences in comparison to the reference. If a contig is broken into multiple sequence blocks but fails to be classified in any of the previous categories, it is reported as being inconsistent

Additionally, the Breadth of Coverage metric is displayed graphically:

- **Genome Fragmentation:** Scatter plot representing the number of contigs per breadth of coverage of the reference, per assembler.

5.8.5.2.3 Contiguity

To supplement the traditional NA and NG contiguity metrics implemented in QUAST [10], we define the LSA metric as the longest single alignment between the assembly and the reference replicon, relative to the reference replicon length, as proposed previously [63]. This provides a simpler picture of assembly quality as lower contiguity immediately suggests a higher fragmentation, missing sequences or more misassemblies. The following metrics are presented in tabular form:

- **LSA:** longest single alignment between the assembly and the reference, relative to the reference length;
- **NAx (where $0 < x \leq 100$):** Length for which the collection of aligned contigs of that length or longer in an assembly covers at least a given percentage of the total length of the reference replicon;
- **NGx (where $0 < x \leq 100$):** Length for which the collection of aligned contigs of that length or longer covers at least a given percentage of the sequence of the reference.
- **Lx (where $0 < x \leq 100$):** Minimal number of contigs that cover x % of the sequence of the reference;

The NAx, NGx and Lx metrics are presented graphically in a line plot for each value of x, where x represents the percentage of the sequence of the reference, ranging from 0 to 100, per assembler.

5.8.5.2.4 Identity

The identity is defined as the number of exact matches between the contigs and the reference replicon, relative to the reference replicon length. The following metrics are presented in tabular form:

- **Identity:** Ratio of identical basepairs in all aligned contigs to the reference;

- **Lowest identity:** Identity of the lowest scoring contig to the reference;

For each plot, the metrics are presented for the contigs filtered for a minimum length that align with the reference replicon.

- **Pls Metric:** Scatter plot for the Phred-like score per contig, per assembler;
- **Gaps:** Location of gaps in comparison to the reference sequence, per assembler, with the cumulative number of gaps per position in the reference. Gaps with 1 basepair or more in length are considered;
- **SNPs:** Location of substitutions in comparison to the reference sequence, per assembler, with the indication of the substitution type and coordinate in the reference. Additionally, the cumulative number of SNPs per position in the reference is presented.

5.8.5.2.5 Misassembly

Similar to what is performed in Global Metrics, this metric is computed for the filtered set of contigs. An aligned contig is considered misassembled when broken into multiple blocks when mapping to the linear reference replicon. Chimeric contigs aligning to more than one reference replicon are counted in each reference individually.

- **Misassembled contigs:** Number of aligned contigs that contain a misassembly event;
- **Misassembly events:** Total number of misassemblies in the aligned contigs;

Additionally, the following information is shown graphically:

- **Misassemblies:** Location of the alignment blocks of misassembled contigs in comparison to the reference sequence, per assembler, with the cumulative number of basepairs in the alignment blocks per position in the reference.

5.8.5.3 Computational Performance Metrics

Different software, implementing distinct de novo assembly algorithms, have distinct computational requirements. As such, computational statistics are registered for each assembler. The following metrics are presented in tabular form:

- **Avg Time:** Average run-time formatted as “hour:minute:second”;

- **CPU/Hour:** Average amount of time, in hours, of CPU usage by an assembler. CPU load obtained from the number of CPUs and their usage percentage;
- **Max Memory (GB):** Maximum peak memory usage by the assembler;
- **Average Read (GB):** Average data size read from disk by the assembler;
- **Average Write (GB):** Average data size written to disk by the assembler.

Additionally, for reproducibility and traceability purposes, the following information is also registered for each assembler in the table:

- **Version:** Version of the assembler captured from stdout;
- **Container:** Full tag of the container used to run the assembler, with a link to the container location in Docker Hub [49].

5.8.6 LMAS Report

LMAS comes pre-packaged with the JS source code for the interactive report, available in the resources/ folder. The source code for the report is available in the LMAS.js repository [64]. It was built with the JavaScript frameworks React [65] (version 16.8.0) and Material-UI [66] (version 4.11.00). All interactive charts were rendered with the graph visualisation library Plotly.js [67] (version 1.57.1) through its React component, react-plotly [65](version 2.5.0).

5.8.7 ZymoBIOMICS microbial community standards

The “get_data.sh” bash script file provided with LMAS downloads the ZymoBIOMICS Microbial Community Standard data and saves it in the “data” folder, in conformation with the default parameters. The simulated samples and all reference replicons saved in a singular multi-sequence fasta are publicly available in Zenodo under the DOI `https://doi.org/10.5281/zenodo.4588969`.

5.8.7.1 Reference Sequences

The complete bacterial genomes and plasmid sequences for the Microbial Community Standards were obtained from ZymoBIOMICS’ Amazon Simple Storage Service, available at `https://s3.amazonaws.com/zymo-files/BioPool/ZymoBIOMICS.STD.refseq.v2.zip`.

For the analysis of LMAS results, the complete ZymoBIOMICS' reference genomes were annotated with PROKKA [68] (version 1.14.5), using the species-specific database for each reference sequence when available. The number of tRNA, rRNA and mobile element coding genes is available in Supplemental Table 5.20. Pairwise comparisons among the set of reference replicons were conducted by calculating the Average Nucleotide Identity (ANI) through BLASTn (version 2.12.0) [69] using pyani [70] (version 0.2.11) [71]. The results are available in Supplemental Table 5.24.

5.8.7.2 Real Sequencing Data

The real paired-end Illumina sequencing data for the ZymoBIOMICS Microbial Community Standards, both evenly and logarithmic distributed, was obtained from the PRJEB29504 study accession [39]. The evenly distributed community standard, containing 8.5 million read pairs, is available under the ERR2984773 accession, and the logarithmically distributed sample, containing 47.5 million read pairs, is available under the accession ERR2935805.

5.8.7.3 Mock Sequencing Data

A set of simulated samples were generated from the genomes in the ZymoBIOMICS standard through the InSilicoSeq sequence simulator (version 1.5.2) [40], including both even and logarithmic distribution, with and without Illumina error model. The error model was obtained from each corresponding real sample depending on the distribution and used to generate the mock data with matching characteristics, including read number and abundance of species in the community (Supplemental Table 5.5).

5.8.7.4 Mock Sequencing Data

The taxonomic composition of the ZymoBIOMICS standard samples, both real and mocks was determined through Kraken2 [72] using the Standard Database (`https://genome-idx.s3.amazonaws.com/kraken/k2_standard_20210517.tar.gz`). The following command was used: "kraken2 –output $sample.kraken –report $sample.kraken_report –memory-mapping –paired –gzip-compressed $fastq_pair[0] $fastq_pair[1]" where $sample is the sample name, $fastq_pair[0] contains the forward-facing reads, and $fastq_pair[1] the reverse-facing reads.

The processing of the kraken reports was performed through custom python code [48] where all the percentage of reads that matched for the species in the dataset were saved, as well as the percentage of unclassified reads. For the *Lactobacillus fermentum*, as in the Standard Kraken database no general Species level classification is available, the percentage of reads was calculated as the sum of all reads aligning to one of the *L. fermentum* subspecies.

The rest of the reads that were classified as any other species were saved conjunctively as "Other". Supplemental Table 5.21 contains the percentage of classified reads for each of the species in the community, as well as "other" and unclassified reads.

5.8.7.5 Assessment of Assembly Success

The complete set of results for 3 LMAS runs for the raw sequence reads of mock communities with an even and logarithmic distribution of species, from real sequencing runs [39] and simulated read datasets, with and without error, matching the intended distribution of species in each sample for the eight bacterial genomes and four plasmids of the ZymoBIOMICS Microbial Community Standards as reference is available in Supplemental Table 5.22 and 5.23. For the assessment of the assembly success for each sample, the different metrics for all LMAS runs were combined and descriptive statistics, such as the average value, standard deviation, minimum and maximum, were obtained through Python's Pandas describe function [73, 74]. Both global and reference based for each assembler, each reference replicon and each sample (ENN - evenly distributed without error model; EMS - evenly distributed with Illumina MiSeq error model; ERR2984773 - real evenly distributed Illumina MiSeq sample, LNN - logarithmically distributed without error model; LHS - logarithmically distributed with Illumina HiSeq error model; ERR2935805 - real logarithmically distributed Illumina HiSeq sample). For descriptive statistics on several assembler by each assembly type (genomic or metagenomic) and each assembler algorithm (single or multiple k-mer), the use of median was preferred due to its higher robustness against outliers and the high range of the distribution of the results. Plotly [67] was used to compute the graphs aggregating the results obtained. The jupyter notebooks [75] with the data processing and all resulting files are available at [48].

The top result of each assembler for each sample was selected, based on the following criteria:

- For the number of uncalled bases, number of misassembled contigs and number of misassembly events, the lower the value, the better, with the exception of 0 for the number of contigs;
- For the percentage of mapped reads and N50, the higher the value, the better;
- The number of basepairs, the best results was the one closest to the target value of the number of basepairs in the reference replicons;

For reference-specific metrics, in addition to the ones stated above when applicable (Number of contigs produced, number of uncalled bases, number of misassembled contigs and number of misassembly events), the following criteria were used:

- For the L90 metric, the lower value was better, with the exception of 0;
- For LSA, NA, NG, breadth of coverage, identity and lowest identity, the higher the value the better;
- For multiplicity, parsimony and validity, the closer to 1, the better;

To obtain the worst value in each metric, the opposite criteria were used. The normalised score for each metric was obtained from the best result for each assembler in each sample through Equation 5.2. For the assessment of assembler consistency, each contig for each assembler was considered the same as its size was exactly the same in each LMAS run.

$$\begin{cases} 1 - \frac{x}{min(X)} & \text{if maximum value is best} \\ \frac{x}{min(X)} & \text{if minimum value is best} \\ 1 - \frac{|x-T|}{T} & \text{if target value is best} \end{cases} \tag{5.2}$$

Where x is the given value of a metric for an assembler, X the list of values for that metric for all assemblers, and T the target value.

5.8.7.6 Resource requirements differ greatly

Regarding computational resources, there is a disparity in usage for the evenly and logarithmically distributed samples (Figure 5.10), with the latter having more resource-intensive requirements possibly due mostly to a higher number of reads. The resource usage also varied greatly by assembler, with multiple k-mer dBG (SPAdes, metaSPAdes, MEGAHIT, SKESA, IDBA-UD, GATBMiniaPipeline and Unicycler) having overall higher resource usage. ABySS performance was inconsistent, having reached a maximum of 1412 CPU hours to produce an assembly (sample ERR2984773), resulting in a run time of 35.52 hours. MetaHipmer2 was the assembler with the highest memory usage, reaching a maximum of 68.7 GB.

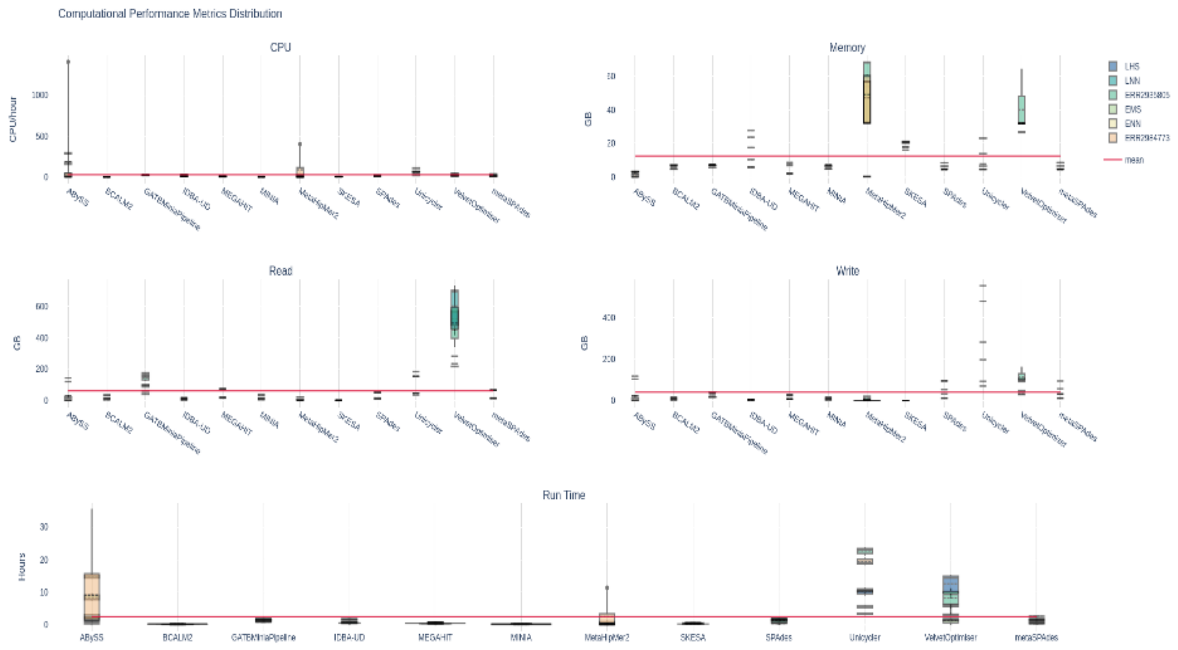


Figure 5.10: Computational resources used by each assembler for the evenly and logarithmically distributed samples. Each plot describes the distribution of resource consumption for 3 LMAS runs for the ZymoBIOMICS microbial community standard dataset for the following metrics: A) CPU/hour, B) Maximum memory in GB; C) Data written to disk in GB; D) Data read from dist in GB; E) Run time in hours. The mean for all samples and all assemblers is indicated in red. The samples are indicated as follows: ENN: dark blue, EMS: teal, ERR2984773: green, LNN: light green, LHS: yellow, ERR2935805: light orange.

Table 5.2: Tools available for the de novo assembly of prokaryotic genomes. For each tool, its publication is indicated, if available, as well as the assembly algorithm implemented if it was developed explicitly to handle metagenomic datasets. The tools are ordered by the date of the last update, with the source code indicated when available. The tools incorporated in LMAS are indicated as such.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S1.xlsx`

Table 5.3: Comparison of metrics and features of LMAS with QUAST and MetaQUAST.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S2.xlsx`

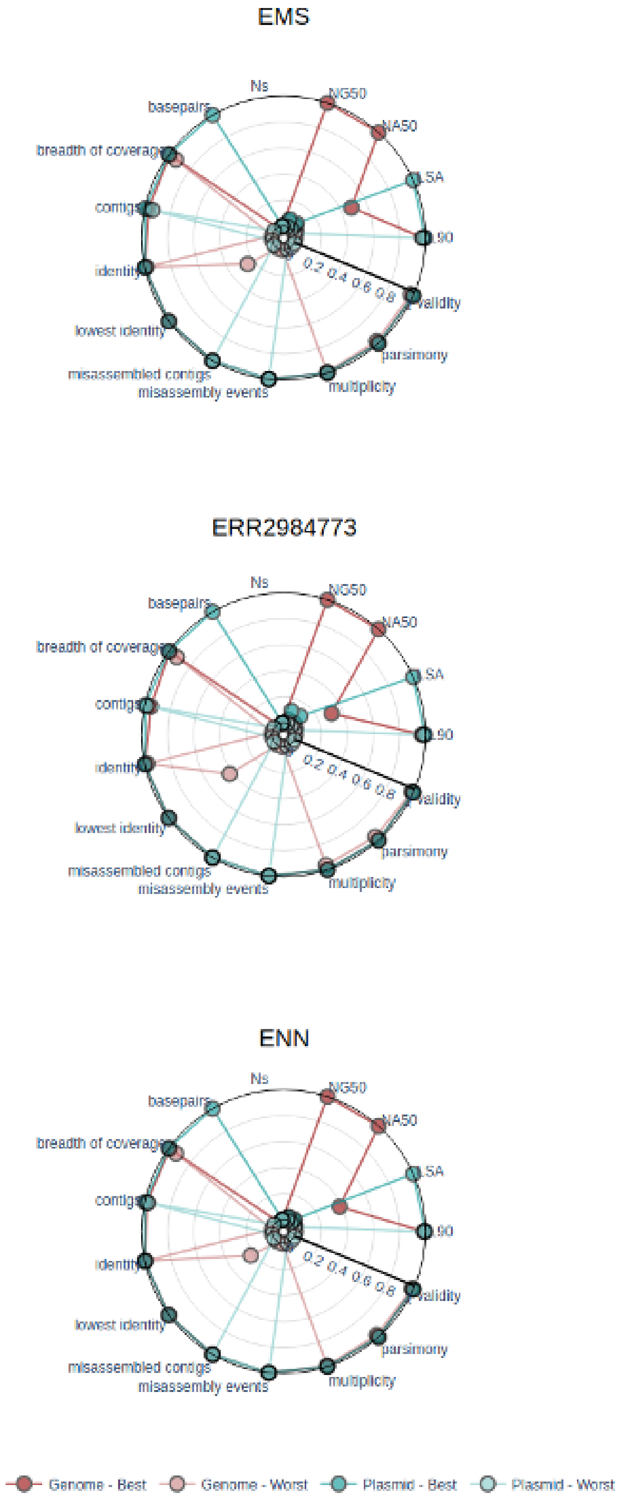


Figure 5.11: Performance per reference of genomic and metagenomic assemblers for the evenly distributed samples in the ZymoBIOMICS Microbial Community Standards dataset. For each sample in the dataset and for the 3 runs, the best and worst scores for each assembler category were selected: genomic (in blue) and metagenomic (in red). The results for each global assembly metric was normalised, with 1 representing the best result, and 0 the worst.

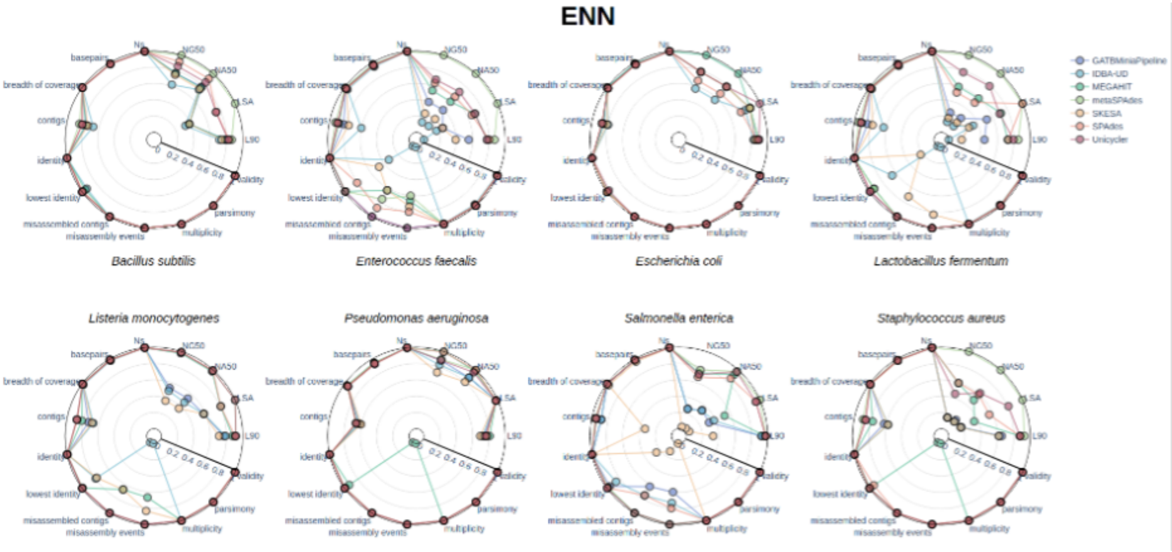


Figure 5.12: Assembler performance per reference for the ZymoBIOMICS Microbial Community Standards dataset for sample ENN. The best score for each assembler was selected for 3 LMAS runs. The results for each global assembly metric was normalised, with 1 representing the best result, and 0 the worst. The following assemblers are represented: GATBMiniaPipeline: dark blue, IDBA-UD: light blue, MEGAHIT: dark green, metaSPAdes: light green, SKESA: yellow, SPAdes: orange, Unicycler: red.

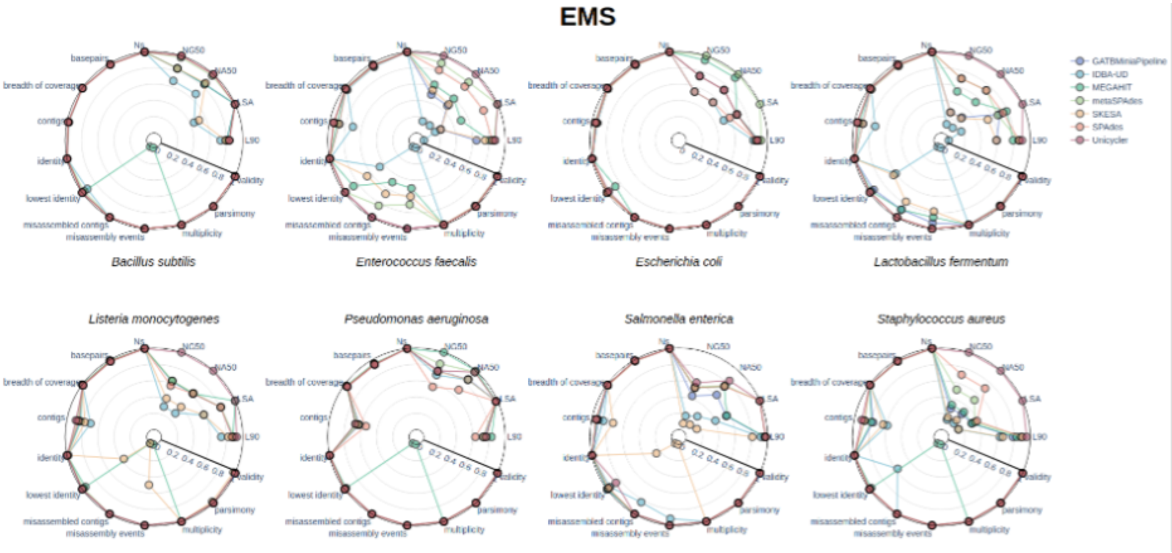


Figure 5.13: Assembler performance per reference for the ZymoBIOMICS Microbial Community Standards dataset for sample EMS. The best score for each assembler was selected for 3 LMAS runs. The results for each global assembly metric was normalised, with 1 representing the best result, and 0 the worst. The following assemblers are represented: GATBMiniaPipeline: dark blue, IDBA-UD: light blue, MEGAHIT: dark green, metaSPAdes: light green, SKESA: yellow, SPAdes: orange, Unicycler: red.

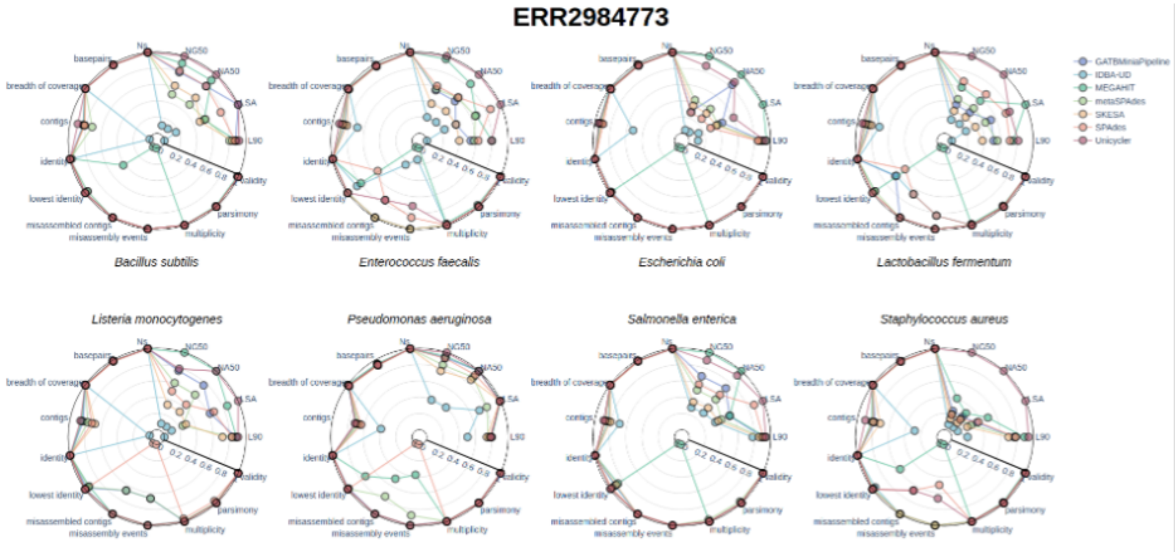


Figure 5.14: Assembler performance per reference for the ZymoBIOMICS Microbial Community Standards dataset for sample ERR2984773. The best score for each assembler was selected for 3 LMAS runs. The results for each global assembly metric was normalised, with 1 representing the best result, and 0 the worst. The following assemblers are represented: GATBMiniaPipeline: dark blue, IDBA-UD: light blue, MEGAHIT: dark green, metaSPAdes: light green, SKESA: yellow, SPAdes: orange, Unicycler: red.

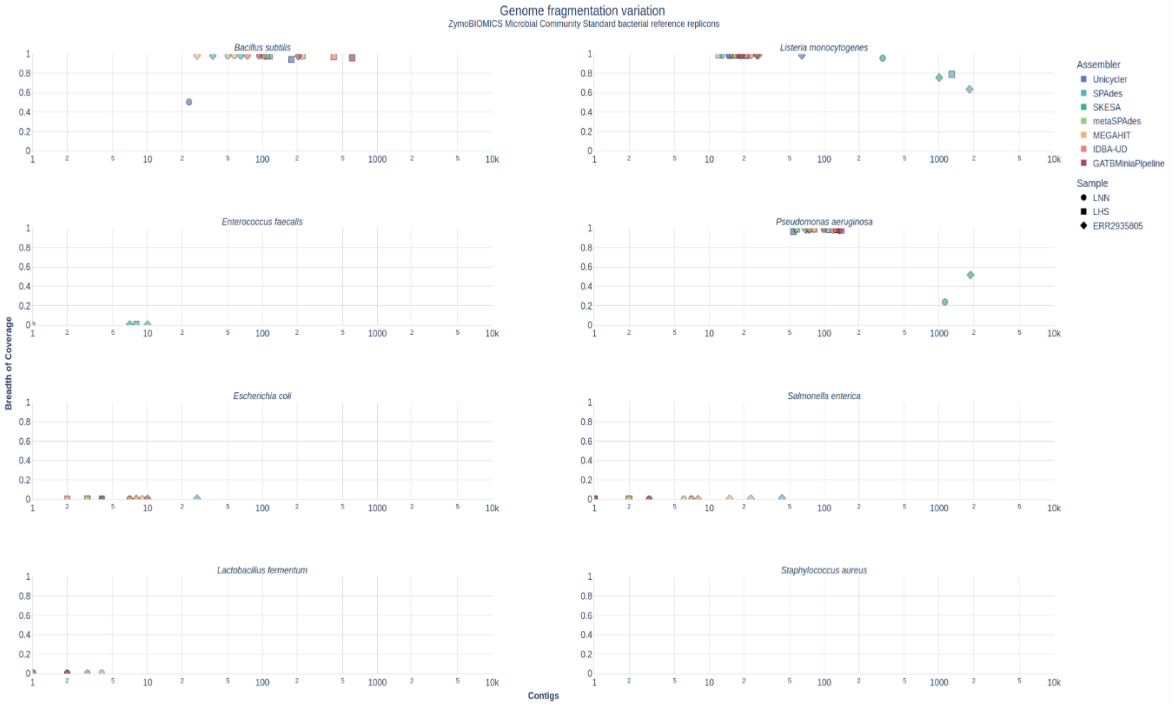


Figure 5.15: Genome fragmentation for each reference replicon of the ZimoBIOMICS community standards dataset for the logarithmically distributed samples. Genome fragmentation for the 3 LMAS runs is represented by the number of contigs and breadth of coverage of the reference per assembler for the logarithmically distributed samples: LNN (logarithmically distributed without error model, identified by a circle), LHS (logarithmically distributed with Illumina HiSeq error model, identified by a square) and ERR2935805 (real Illumina HiSeq sample, identified by a diamond). Each assembler is identified with the following colour scheme - dark blue: Unicycler, light blue: SPAdes, dark green: SKESA, light green: metaSPAdes, yellow: MEGAHIT, orange: IDBA-UD, red: GATBMiniaPipeline.

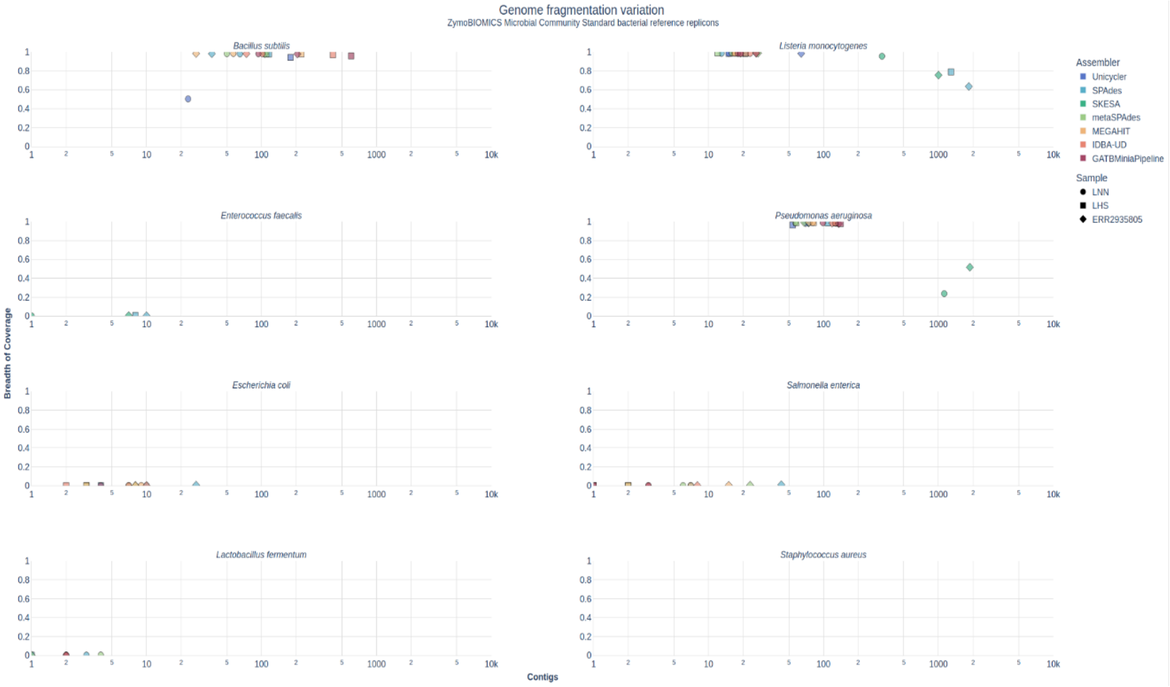


Figure 5.16: Genome fragmentation for each reference replicon of the BMock12 community standards dataset sample. Genome fragmentation is represented by the number of contigs and breadth of coverage of the reference per assembler. Each assembler is identified with the following colour scheme - dark blue: Unicycler, light blue: SPAdes, dark green: SKESA, light green: metaSPAdes, yellow: MEGAHIT, orange: IDBA-UD, red: GATBMiniaPipeline. Each reference replicon is identified by its IMG Taxon ID: 2615840527, Muricauda sp; 2615840533, Thioclava sp; 2615840601, Cohaesibacter sp; 2615840646, Propionibacteriaceae bacterium, 2615840697, Marinobacter sp LV10R510-8; 2616644829, Marinobacter sp LV10MA510-1; 2617270709, Psychrobacter sp; 2623620557, Micromonospora echinaurantiaca; 2623620567, Micromonospora echinofusca; 2623620609, Micromonospora coxensis; 2623620617, Halomonas sp. HL-4; and 2623620618, Halomonas sp. HL-93.

Table 5.4: The ZymoBIOMICS Microbial Community Standard datasets. Set of raw sequence reads used as input in LMAS of mock communities with an even and logarithmic distribution of species, from real sequencing runs and simulated read datasets, with and without error, matching the intended distribution of species in ZymoBIOMICS Microbial Community Standard.

Sample Name	Distribution	Error Model	Read Length (bp)	Read Pairs (M)
ENN	Even	None	150	8,5
EMS	Even	Illumina MiSeq	150	8,5
ERR2984773	Even	Real MiSeq Sample	150	8,5
LNN	Log	None	100	47,5
LHS	Log	Illumina HiSeq	100	47,5
ERR2935805	Log	Real HiSeq Sample	100	47,5

Table 5.5: Microbial composition of the ZymoBIOMICS microbial community standard dataset with Even and Logarithmic distribution of species. Theoretical microbial composition of the standards, and the corresponding number of reads generated for each replicon.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S4.xlsx`

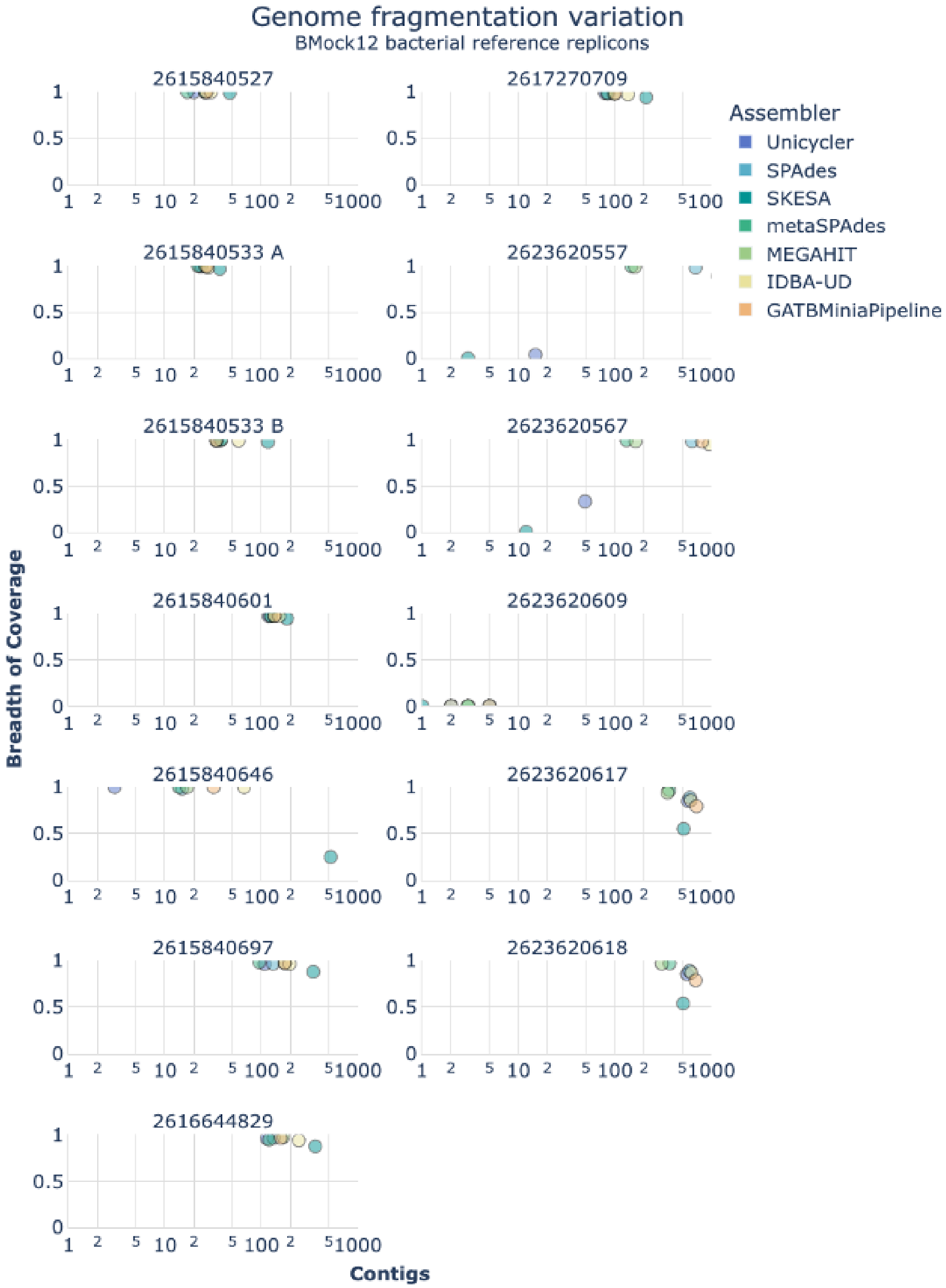


Figure 5.17: Location of gaps in comparison to the reference sequence, per assembler, for each reference replicon of the BMock12 community standards datasets. The resulting plot contains the gaps obtained for GATBMiniaPipeline, IDBA-UD, MEGAHIT, metaSPAdes, SKESA, SPAdes and Unicycler assemblers.

Table 5.6: Global quality metrics variation in three LMAS runs for sample ENN per assembler. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, are presented for each metric for each assembler.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S5.xlsx`

Table 5.7: Global quality metrics variation in three LMAS runs for sample EMS per assembler. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, are presented for each metric for each assembler.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S6.xlsx`

Table 5.8: Global quality metrics variation in three LMAS runs for sample ERR2984773 per assembler. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, are presented for each metric for each assembler.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S7.xlsx`

Table 5.9: Global quality metrics variation in three LMAS runs for sample LNN per assembler. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, are presented for each metric for each assembler.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S8.xlsx`

Table 5.10: Global quality metrics variation in three LMAS runs for sample LHS per assembler. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, are presented for each metric for each assembler.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S9.xlsx`

Table 5.11: Global quality metrics variation in three LMAS runs for sample ERR2935805 per assembler. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, are presented for each metric for each assembler.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S10.xlsx`

Table 5.12: Inconsistent contigs produced by the assemblers in 3 LMAS runs. For each assembler, the total number of contigs produced over the 3 runs of the LMAS workflow is indicated, as well as the contigs present in only two and a single run.

Assembler	Total Contigs	Contigs present in 3 out of 3 runs	Contigs present in 2 out of 3 runs	Contigs present in 1 out of 3 runs
ABySS	7638	2505	50	23
BCALM2	1962	654	0	0
GATBMiniaPipeline	14321	4753	25	12
IDBA-UD	15383	5115	16	6
MEGAHIT	5440	1812	1	2
MetaHipMer2	2358	772	16	10
metaSPAdes	6462	2154	0	0
minia	6427	2106	41	27
SKESA	11187	3729	0	0
SPAdes	8631	2877	0	0
Unicycler	5598	1866	0	0
VelvetOptimiser	2315	553	112	432

Table 5.13: Global assembly metrics for single and multiple k-mer dBg assemblers. The median and the minimum and maximum values obtained are presented for each metric for all samples in 3 runs of LMAS. Single k-mer bBg assemblers: ABySS, BCALM2 and minia. Multiple k-mer bBg assembler: GATBMiniaPipeline, IDBA-UD, MEGAHIT, MetaHipMer2, metaSPAdes, SKESA, SPAdes, Unicycler and VelverOptimiser.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S12.xlsx`

Table 5.14: Global assembly metrics for genomic and metagenomic multiple k-mer dBg assemblers. The median and the minimum and maximum values obtained are presented for each metric for all samples in 3 runs of LMAS. Genomic assemblers: SKESA, SPAdes and Unicycler. Metagenomic assemblers: GATBMiniaPipeline, IDBA-UD, MEGAHIT and metaSPAdes.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S13.xlsx`

Table 5.15: Per reference quality metrics variation in three LMAS s for sample ENN per assembler of the ZymoBIOMICS microbial community standard dataset. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, are presented for each metric for each assembler.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S14.xlsx`

Table 5.16: Per reference quality metrics variation in three LMAS s for sample EMS per assembler of the ZymoBIOMICS microbial community standard dataset. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, are presented for each metric for each assembler.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S15.xlsx`

Table 5.17: Per reference quality metrics variation in three LMAS s for sample ERR2984773 per assembler of the ZymoBIOMICS microbial community standard dataset. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, are presented for each metric for each assembler.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S16.xlsx`

Table 5.18: Inconsistent gaps produced by the assemblers in 3 LMAS runs. For each assembler, the total number of gaps consistently produced in relation to the reference replicons over the 3 runs of the LMAS workflow is indicated, as well as gaps present in only two and a single run.

Assembler	Gaps present in 3 out of 3 runs	Gaps present in 2 out of 3 runs	Gaps present in 1 out of 3 runs
GATBMiniaPipeline	3108	278	157
IDBA-UD	4458	68	40
MEGAHIT	1017	122	89
metaSPAdes	1185	0	0
SKESA	6681	0	0
SPAdes	2433	0	0
Unicycler	1512	0	0

Table 5.19: Annotation of consistent gaps produced by the assemblers in 3 LMAS runs.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S18.xlsx`

Table 5.20: Number of tRNA and rRNA coding sequencing, and mobile elements in ZymoBIOMICS microbial community standard reference replicons. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, are presented for each metric for each assembler.

Reference	tRNA	rRNA	Mobile elements
Bacillus subtilis	86	31	17
Enterococcus faecalis	62	12	7
Escherichia coli	89	22	85
Lactobacillus fermentum	59	15	80
Listeria monocytogenes	67	18	3
Pseudomonas aeruginosa	78	12	19
Salmonella enterica	88	22	39
Staphylococcus aureus	60	19	7

Table 5.21: Taxonomic classification of the ZymoBIOMICS microbial community standard dataset. The classification was performed with Kraken2, using the Standard Database. The results are presented as the percentage of classified reads for the 8 bacterial species in the community, as well as unclassified reads and the group of reads that are classified as species not contained in the community standard.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S20.xlsx`

Table 5.22: Global assembly metrics for dBg assemblers with single and multiple k-mer algorithms.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S21.xlsx`

Table 5.23: Reference based quality metrics in three LMAS runs for the ZimoBIOMICS community standards dataset.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S22.xlsx`

Table 5.24: Pairwise comparisons of the ZymoBIOMICS microbial community standard reference replicons. All pairwise comparisons among the set of genomes were conducted using Average Nucleotide Identity through BLAST as a proxy for DNA-DNA hybridisation.

Table available for download at `https://zenodo.org/record/6623458/files/LMAS%20Supplemental%20Material%20-%20Tables_Table%20S23.xlsx`

Table 5.25: Microbial composition of the BMock12 microbial community standard dataset.

Table available for download at `https://zenodo.org/record/7129554/files/LMAS%20Supplemental%20Material%20-%20Tables%20-%20revision_Table%20S24.xlsx`

Table 5.26: Pairwise comparisons of the BMock12 microbial community standard reference replicons.

Table available for download at `https://zenodo.org/record/7129554/files/LMAS%20Supplemental%20Material%20-%20Tables%20-%20revision_Table%20S25.xlsx`

Table 5.27: Global quality metrics for the BMock12 sample SRX4901583 per assembler.

Table available for download at `https://zenodo.org/record/7129554/files/LMAS%20Supplemental%20Material%20-%20Tables%20-%20revision_Table%20S26.xlsx`

Table 5.28: Per reference quality metrics for the BMock12 sample SRX4901583 per assembler.

Table available for download at `https://zenodo.org/record/7129554/files/LMAS%20Supplemental%20Material%20-%20Tables%20-%20revision_Table%20S27.xlsx`

Table 5.29: Microbial composition of the Gut-Mix-RR microbial community standard dataset.

Table available for download at `https://zenodo.org/record/7129554/files/LMAS%20Supplemental%20Material%20-%20Tables%20-%20revision_Table%20S28.xlsx`

Table 5.30: Pairwise comparisons of the Gut-Mix microbial community standard reference replicons.

Table available for download at `https://zenodo.org/record/7129554/files/LMAS%20Supplemental%20Material%20-%20Tables%20-%20revision_Table%20S29.xlsx`

Table 5.31: Global quality metrics for the Gut-Mix-RR sample SRR11487941 per assembler.

Table available for download at `https://zenodo.org/record/7129554/files/LMAS%20Supplemental%20Material%20-%20Tables%20-%20revision_Table%20S30.xlsx`

5.9 References

[1] Alexandre Angers-Loustau et al. “The challenges of designing a benchmark strategy for bioinformatics pipelines in the identification of antimicrobial resistance determinants using next generation sequencing technologies”. en. In: *F1000Research* 7 (Dec. 2018), p. 459. ISSN: 2046-1402. DOI: `10.12688/f1000research.14509.2`. URL: `https://f1000research.com/articles/7-459/v2` (visited on 03/25/2021).

[2] Bjorn Gruening et al. “Recommendations for the packaging and containerizing of bioinformatics software”. en. In: *F1000Research* 7 (Mar. 2019), p. 742. ISSN: 2046-1402. DOI: `10.12688/f1000research.15140.2`. URL: `https://f1000research.com/articles/7-742/v2` (visited on 04/30/2022).

[3] Alexander Sczyrba et al. “Critical Assessment of Metagenome Interpretation—a benchmark of metagenomics software”. en. In: *Nature Methods* 14.11 (Nov. 2017). Number: 11 Publisher: Nature Publishing Group, pp. 1063–1071. ISSN: 1548-7105. DOI: `10.1038/nmeth.4458`. URL: `https://www.nature.com/articles/nmeth.4458` (visited on 03/20/2022).

[4] Natacha Couto et al. “Critical steps in clinical shotgun metagenomics for the concomitant detection and typing of microbial pathogens”. en. In: *Scientific Reports* 8.1 (Dec. 2018), p. 13767. ISSN: 2045-2322. DOI: `10.1038/s41598-018-31873-w`. URL: `http://www.nature.com/articles/s41598-018-31873-w` (visited on 05/11/2022).

[5] F. Meyer et al. *Critical Assessment of Metagenome Interpretation - the second round of challenges*. en. preprint. Bioinformatics, July 2021. DOI: `10.1101/2021.07.12.451567`. URL: `http://biorxiv.org/lookup/doi/10.1101/2021.07.12.451567` (visited on 10/25/2021).

[6] Martin Ayling, Matthew D Clark, and Richard M Leggett. “New approaches for metagenome assembly with short reads”. In: *Briefings in Bioinformatics* 21.2 (Mar. 2020), pp. 584–594. ISSN: 1477-4054. DOI: `10.1093/bib/bbz020`. URL: `https://doi.org/10.1093/bib/bbz020` (visited on 09/08/2022).

[7] Nathan D Olson et al. “Metagenomic assembly through the lens of validation: recent advances in assessing and improving the quality of genomes assembled from metagenomes”. en. In: *Briefings in Bioinformatics* 20.4 (July 2019), pp. 1140–1150. ISSN: 1467-5463, 1477-4054. DOI: `10.1093/bib/bbx098`. URL: `https://academic.oup.com/bib/article/20/4/1140/4075034` (visited on 03/25/2021).

[8] Hanno Teeling and Frank Oliver Glöckner. “Current opportunities and challenges in microbial metagenome analysis–a bioinformatic perspective”. eng. In: *Briefings in Bioinformatics* 13.6 (Nov. 2012), pp. 728–742. ISSN: 1477-4054. DOI: `10.1093/bib/bbs039`.

[9] Keith R Bradnam et al. "Assemblathon 2: evaluating de novo methods of genome assembly in three vertebrate species". en. In: *GigaScience* 2.1 (Dec. 2013), p. 10. ISSN: 2047-217X. DOI: `10.1186/2047-217X-2-10`. URL: `https://academic.oup.com/gigascience/article-lookup/doi/10.1186/2047-217X-2-10` (visited on 04/13/2021).

[10] Alexey Gurevich et al. "QUAST: quality assessment tool for genome assemblies". eng. In: *Bioinformatics (Oxford, England)* 29.8 (Apr. 2013), pp. 1072–1075. ISSN: 1367-4811. DOI: `10.1093/bioinformatics/btt086`.

[11] Paolo Di Tommaso et al. "Nextflow enables reproducible computational workflows". en. In: *Nature Biotechnology* 35.4 (Apr. 2017), pp. 316–319. ISSN: 1087-0156, 1546-1696. DOI: `10.1038/nbt.3820`. URL: `http://www.nature.com/articles/nbt.3820` (visited on 03/24/2021).

[12] Dirk Merkel. "Docker: Lightweight Linux Containers for Consistent Development and Deployment". In: *Linux J.* 2014.239 (Mar. 2014). Place: Houston, TX Publisher: Belltown Media. ISSN: 1075-3583.

[13] *Lmas :: Anaconda.org*. URL: `https://anaconda.org/bioconda/lmas` (visited on 04/06/2022).

[14] Ines Mendes, Pedro Vila-Cerqueira, and Mario Ramirez. *LMAS: Last (Meta)genomic Assembler Standing*. original-date: 2020-10-20T15:03:12Z. July 2021. URL: `https://github.com/B-UMMI/LMAS` (visited on 04/04/2022).

[15] *Installation — LMAS 0.1 documentation*. URL: `https://lmas.readthedocs.io/en/latest/getting_started/installation.html` (visited on 04/04/2022).

[16] *Basic Usage — LMAS 0.1 documentation*. URL: `https://lmas.readthedocs.io/en/latest/user/basic_usage.html` (visited on 04/04/2022).

[17] *Parameters — LMAS 0.1 documentation*. URL: `https://lmas.readthedocs.io/en/latest/user/parameters.html` (visited on 04/04/2022).

[18] Shaun D. Jackman et al. "ABySS 2.0: resource-efficient assembly of large genomes using a Bloom filter". en. In: *Genome Research* 27.5 (May 2017), pp. 768–777. ISSN: 1088-9051, 1549-5469. DOI: `10.1101/gr.214346.116`. URL: `http://genome.cshlp.org/lookup/doi/10.1101/gr.214346.116` (visited on 08/28/2021).

[19] *GATB/gatb-minia-pipeline*. original-date: 2015-09-10T13:03:09Z. Mar. 2022. URL: `https://github.com/GATB/gatb-minia-pipeline` (visited on 04/04/2022).

[20] Y. Peng et al. "IDBA-UD: a de novo assembler for single-cell and metagenomic sequencing data with highly uneven depth". en. In: *Bioinformatics* 28.11 (June 2012), pp. 1420–1428. ISSN: 1367-4803, 1460-2059. DOI: `10.1093/bioinformatics/bts174`. URL: `https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/bts174` (visited on 04/08/2021).

[21] Dinghua Li et al. "MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph". In: *Bioinformatics* 31.10 (May 2015), pp. 1674–1676. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/btv033`. URL: `https://doi.org/10.1093/bioinformatics/btv033` (visited on 03/14/2022).

[22] Evangelos Georganas et al. "Extreme Scale De Novo Metagenome Assembly". In: *SC18: International Conference for High Performance Computing, Networking, Storage and Analysis*. Dallas, TX, USA: IEEE, Nov. 2018, pp. 122–134. ISBN: 978-1-5386-8384-2. DOI: `10.1109/SC.2018.00013`. URL: `https://ieeexplore.ieee.org/document/8665813/` (visited on 09/27/2021).

[23] Sergey Nurk et al. "metaSPAdes: a new versatile metagenomic assembler". en. In: *Genome Research* 27.5 (May 2017), pp. 824–834. ISSN: 1088-9051, 1549-5469. DOI: `10.1101/gr.213959.116`. URL: `http://genome.cshlp.org/lookup/doi/10.1101/gr.213959.116` (visited on 03/25/2021).

[24] Rayan Chikhi and Guillaume Rizk. "Space-efficient and exact de Bruijn graph representation based on a Bloom filter". en. In: *Algorithms for Molecular Biology* 8.1 (Jan. 2013), p. 22. ISSN: 1748-7188. DOI: `10.1186/1748-7188-8-22`. URL: `https://almob.biomedcentral.com/articles/10.1186/1748-7188-8-22` (visited on 04/08/2021).

[25] Alexandre Souvorov, Richa Agarwala, and David J. Lipman. "SKESA: strategic k-mer extension for scrupulous assemblies". In: *Genome Biology* 19.1 (Oct. 2018), p. 153. ISSN: 1474-760X. DOI: `10.1186/s13059-018-1540-z`. URL: `https://doi.org/10.1186/s13059-018-1540-z` (visited on 03/14/2022).

[26] Anton Bankevich et al. "SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing". en. In: *Journal of Computational Biology* 19.5 (May 2012), pp. 455–477. ISSN: 1066-5277, 1557-8666. DOI: `10.1089/cmb.2012.0021`. URL: `http://www.liebertpub.com/doi/10.1089/cmb.2012.0021` (visited on 03/18/2022).

[27] Ryan R. Wick et al. "Unicycler: Resolving bacterial genome assemblies from short and long sequencing reads". en. In: *PLOS Computational Biology* 13.6 (June 2017). Ed. by Adam M. Phillippy, e1005595. ISSN: 1553-7358. DOI: `10.1371/journal.pcbi.1005595`. URL: `https://dx.plos.org/10.1371/journal.pcbi.1005595` (visited on 04/08/2021).

[28] Torsten Seemann. *VelvetOptimiser: automate your Velvet assemblies*. original-date: 2012-06-06T05:54:27Z. Nov. 2021. URL: `https://github.com/tseemann/VelvetOptimiser` (visited on 04/04/2022).

[29] *Short-Read (Meta)Genomic Assemblers — LMAS 0.1 documentation*. URL: `https://lmas.readthedocs.io/en/latest/user/assemblers.html` (visited on 04/04/2022).

[30] *Add Assembler Process — LMAS 0.1 documentation.* URL: `https://lmas.readthedocs.io/en/latest/dev/add_process.html` (visited on 04/04/2022).

[31] Heng Li. "Minimap2: pairwise alignment for nucleotide sequences". In: *Bioinformatics* 34.18 (2018), pp. 3094–3100. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/bty191`. URL: `https://doi.org/10.1093/bioinformatics/bty191` (visited on 03/02/2022).

[32] Dent Earl et al. "Assemblathon 1: a competitive assessment of de novo short read assembly methods". eng. In: *Genome Research* 21.12 (Dec. 2011), pp. 2224–2241. ISSN: 1549-5469. DOI: `10.1101/gr.126599.111`.

[33] Brent Ewing and Phil Green. "Base-Calling of Automated Sequencer Traces Using Phred. II. Error Probabilities". en. In: *Genome Research* 8.3 (Mar. 1998). Company: Cold Spring Harbor Laboratory Press Distributor: Cold Spring Harbor Laboratory Press Institution: Cold Spring Harbor Laboratory Press Label: Cold Spring Harbor Laboratory Press Publisher: Cold Spring Harbor Lab, pp. 186–194. ISSN: 1088-9051, 1549-5469. DOI: `10.1101/gr.8.3.186`. URL: `https://genome.cshlp.org/content/8/3/186` (visited on 03/02/2022).

[34] Alla Mikheenko, Vladislav Saveliev, and Alexey Gurevich. "MetaQUAST: evaluation of metagenome assemblies". eng. In: *Bioinformatics (Oxford, England)* 32.7 (Apr. 2016), pp. 1088–1090. ISSN: 1367-4811. DOI: `10.1093/bioinformatics/btv697`.

[35] Nancy Manchanda et al. "GenomeQC: a quality assessment tool for genome assemblies and gene structure annotations". en. In: *BMC Genomics* 21.1 (Dec. 2020), p. 193. ISSN: 1471-2164. DOI: `10.1186/s12864-020-6568-2`. URL: `https://bmcgenomics.biomedcentral.com/articles/10.1186/s12864-020-6568-2` (visited on 08/11/2021).

[36] Stephen Meader et al. "Genome assembly quality: assessment and improvement using the neutral indel model". eng. In: *Genome Research* 20.5 (May 2010), pp. 675–684. ISSN: 1549-5469. DOI: `10.1101/gr.096966.109`.

[37] Richard Challis et al. "BlobToolKit – Interactive Quality Assessment of Genome Assemblies". en. In: *G3 Genes|Genomes|Genetics* 10.4 (Apr. 2020), pp. 1361–1374. ISSN: 2160-1836. DOI: `10.1534/g3.119.400908`. URL: `https://academic.oup.com/g3journal/article/10/4/1361/6026202` (visited on 08/11/2021).

[38] Fernando Meyer et al. "Tutorial: assessing metagenomics software with the CAMI benchmarking toolkit". en. In: *Nature Protocols* 16.4 (Apr. 2021). Number: 4 Publisher: Nature Publishing Group, pp. 1785–1801. ISSN: 1750-2799. DOI: `10.1038/s41596-020-00480-3`. URL: `https://www.nature.com/articles/s41596-020-00480-3` (visited on 04/06/2022).

[39] Samuel M Nicholls et al. “Ultra-deep, long-read nanopore sequencing of mock microbial community standards”. en. In: *GigaScience* 8.5 (May 2019), giz043. ISSN: 2047-217X. DOI: `10.1093/gigascience/giz043`. URL: `https://academic.oup.com/gigascience/article/doi/10.1093/gigascience/giz043/5486468` (visited on 04/27/2021).

[40] Hadrien Gourlé et al. “Simulating Illumina metagenomic data with InSilicoSeq”. en. In: *Bioinformatics* 35.3 (Feb. 2019). Ed. by John Hancock, pp. 521–522. ISSN: 1367-4803, 1460-2059. DOI: `10.1093/bioinformatics/bty630`. URL: `https://academic.oup.com/bioinformatics/article/35/3/521/5055123` (visited on 04/27/2021).

[41] Volkan Sevim et al. “Shotgun metagenome data of a defined mock community using Oxford Nanopore, PacBio and Illumina technologies”. en. In: *Scientific Data* 6.1 (Nov. 2019). Number: 1 Publisher: Nature Publishing Group, p. 285. ISSN: 2052-4463. DOI: `10.1038/s41597-019-0287-z`. URL: `https://www.nature.com/articles/s41597-019-0287-z` (visited on 06/30/2022).

[42] Gregory C. A. Amos et al. “Developing standards for the microbiome field”. In: *Microbiome* 8.1 (June 2020), p. 98. ISSN: 2049-2618. DOI: `10.1186/s40168-020-00856-3`. URL: `https://doi.org/10.1186/s40168-020-00856-3` (visited on 06/30/2022).

[43] Fernando Meyer et al. “Critical Assessment of Metagenome Interpretation: the second round of challenges”. en. In: *Nature Methods* 19.4 (Apr. 2022), pp. 429–440. ISSN: 1548-7091, 1548-7105. DOI: `10.1038/s41592-022-01431-4`. URL: `https://www.nature.com/articles/s41592-022-01431-4` (visited on 05/11/2022).

[44] Basil Britto Xavier et al. “Employing whole genome mapping for optimal de novo assembly of bacterial genomes”. In: *BMC Research Notes* 7.1 (July 2014), p. 484. ISSN: 1756-0500. DOI: `10.1186/1756-0500-7-484`. URL: `https://doi.org/10.1186/1756-0500-7-484` (visited on 08/31/2022).

[45] Kanak Mahadik et al. “Scalable Genome Assembly through Parallel de Bruijn Graph Construction for Multiple k-mers”. en. In: *Scientific Reports* 9.1 (Oct. 2019). Number: 1 Publisher: Nature Publishing Group, p. 14882. ISSN: 2045-2322. DOI: `10.1038/s41598-019-51284-9`. URL: `https://www.nature.com/articles/s41598-019-51284-9` (visited on 08/31/2022).

[46] Genevieve Johnson et al. “Complete Genome Sequence of a Pseudomonas aeruginosa Isolate from a Kidney Stone”. In: *Microbiology Resource Announcements* 8.38 (Sept. 2019), e01073–19. ISSN: 2576-098X. DOI: `10.1128/MRA.01073-19`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6753288/` (visited on 09/09/2022).

[47] Adrian Fritz et al. "CAMISIM: simulating metagenomes and microbial communities". In: *Microbiome* 7.1 (Feb. 2019), p. 17. ISSN: 2049-2618. DOI: 10.1186/s40168-019-0633-6. URL: https://doi.org/10.1186/s40168-019-0633-6 (visited on 06/24/2022).

[48] *LMAS Manuscript Analysis*. original-date: 2021-05-03T12:58:08Z. Jan. 2022. URL: https://github.com/B-UMMI/LMAS_Manuscript_Analysis (visited on 04/04/2022).

[49] *Docker Hub Container Image Library | App Containerization*. URL: https://hub.docker.com/ (visited on 04/04/2022).

[50] Andre P Masella et al. "PANDAseq: paired-end assembler for illumina sequences". en. In: *BMC Bioinformatics* 13.1 (2012), p. 31. ISSN: 1471-2105. DOI: 10.1186/1471-2105-13-31. URL: http://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-13-31 (visited on 04/23/2021).

[51] *ABySS*. original-date: 2012-03-19T23:13:39Z. Mar. 2022. URL: https://github.com/bcgsc/abyss (visited on 04/04/2022).

[52] Rayan Chikhi, Antoine Limasset, and Paul Medvedev. "Compacting de Bruijn graphs from sequencing data quickly and in low memory". en. In: *Bioinformatics* 32.12 (June 2016), pp. i201–i208. ISSN: 1367-4803, 1460-2059. DOI: 10.1093/bioinformatics/btw279. URL: https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btw279 (visited on 04/08/2021).

[53] Kristoffer Sahlin et al. "BESST - Efficient scaffolding of large fragmented assemblies". en. In: *BMC Bioinformatics* 15.1 (Dec. 2014), p. 281. ISSN: 1471-2105. DOI: 10.1186/1471-2105-15-281. URL: https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-15-281 (visited on 06/10/2021).

[54] Yu Peng. *loneknightpy/idba*. original-date: 2014-06-20T16:09:28Z. Mar. 2022. URL: https://github.com/loneknightpy/idba (visited on 04/04/2022).

[55] Brian Bushnell, Jonathan Rood, and Esther Singer. "BBMerge – Accurate paired shotgun read merging via overlap". en. In: *PLOS ONE* 12.10 (Oct. 2017). Ed. by Patrick Jon Biggs, e0185056. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0185056. URL: https://dx.plos.org/10.1371/journal.pone.0185056 (visited on 06/10/2021).

[56] Dinghua Li. *MEGAHIT*. original-date: 2014-09-25T10:29:18Z. Mar. 2022. URL: https://github.com/voutcn/megahit (visited on 04/04/2022).

[57] *berkeleylab / mhm2 — Bitbucket*. URL: https://bitbucket.org/berkeleylab/mhm2/src/master/ (visited on 04/04/2022).

[58] *SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing | Journal of Computational Biology*. URL: `https://www.liebertpub.com/doi/10.1089/cmb.2012.0021` (visited on 01/24/2022).

[59] *Minia*. original-date: 2016-04-07T09:27:59Z. Mar. 2022. URL: `https://github.com/GATB/minia` (visited on 04/04/2022).

[60] Ryan Wick. *Unicycler*. original-date: 2016-03-14T06:57:02Z. Apr. 2022. URL: `https://github.com/rrwick/Unicycler` (visited on 04/04/2022).

[61] Daniel R. Zerbino and Ewan Birney. "Velvet: Algorithms for de novo short read assembly using de Bruijn graphs". en. In: *Genome Research* 18.5 (May 2008). Company: Cold Spring Harbor Laboratory Press Distributor: Cold Spring Harbor Laboratory Press Institution: Cold Spring Harbor Laboratory Press Label: Cold Spring Harbor Laboratory Press Publisher: Cold Spring Harbor Lab, pp. 821–829. ISSN: 1088-9051, 1549-5469. DOI: `10.1101/gr.074492.107`. URL: `https://genome.cshlp.org/content/18/5/821` (visited on 03/14/2022).

[62] Shunichi Kosugi et al. "Comprehensive evaluation of structural variation detection algorithms for whole genome sequencing". en. In: *Genome Biology* 20.1 (Dec. 2019), p. 117. ISSN: 1474-760X. DOI: `10.1186/s13059-019-1720-5`. URL: `https://genomebiology.biomedcentral.com/articles/10.1186/s13059-019-1720-5` (visited on 05/10/2021).

[63] Ryan R. Wick and Kathryn E. Holt. "Benchmarking of long-read assemblers for prokaryote whole genome sequencing". en. In: *F1000Research* 8 (Feb. 2021), p. 2138. ISSN: 2046-1402. DOI: `10.12688/f1000research.21782.4`. URL: `https://f1000research.com/articles/8-2138/v4` (visited on 03/25/2021).

[64] *LMAS Report*. original-date: 2020-11-20T14:06:53Z. Nov. 2021. URL: `https://github.com/B-UMMI/LMAS.js` (visited on 04/04/2022).

[65] *React*. URL: `https://plotly.com/javascript/react/` (visited on 04/04/2022).

[66] *MUI: The React component library you always wanted*. pt. URL: `https://mui.com/pt/` (visited on 04/04/2022).

[67] *Plotly*. URL: `https://plotly.com/javascript/` (visited on 04/04/2022).

[68] Torsten Seemann. "Prokka: rapid prokaryotic genome annotation". eng. In: *Bioinformatics (Oxford, England)* 30.14 (July 2014), pp. 2068–2069. ISSN: 1367-4811. DOI: `10.1093/bioinformatics/btu153`.

[69] Christiam Camacho et al. "BLAST+: architecture and applications". en. In: *BMC Bioinformatics* 10.1 (2009), p. 421. ISSN: 1471-2105. DOI: `10.1186/1471-2105-10-421`. URL: `http://www.biomedcentral.com/1471-2105/10/421` (visited on 08/09/2021).

[70] Leighton Pritchard. *pyani*. original-date: 2014-12-30T18:48:01Z. Mar. 2022. URL: `https://github.com/widdowquinn/pyani` (visited on 04/04/2022).

[71] Leighton Pritchard et al. “Genomics and taxonomy in diagnostics for food security: soft-rotting enterobacterial plant pathogens”. en. In: *Analytical Methods* 8.1 (2016), pp. 12–24. ISSN: 1759-9660, 1759-9679. DOI: `10.1039/C5AY02550H`. URL: `http://xlink.rsc.org/?DOI=C5AY02550H` (visited on 12/20/2021).

[72] Derrick E. Wood, Jennifer Lu, and Ben Langmead. “Improved metagenomic analysis with Kraken 2”. In: *Genome Biology* 20.1 (Nov. 2019), p. 257. ISSN: 1474-760X. DOI: `10.1186/s13059-019-1891-0`. URL: `https://doi.org/10.1186/s13059-019-1891-0` (visited on 03/03/2022).

[73] *pandas - Python Data Analysis Library*. URL: `https://pandas.pydata.org/` (visited on 04/04/2022).

[74] *pandas.DataFrame.describe — pandas 1.4.2 documentation*. URL: `https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.describe.html` (visited on 04/04/2022).

[75] *Project Jupyter*. Library Catalog: jupyter.org. URL: `https://www.jupyter.org` (visited on 07/21/2020).

Part III

Challenges of data availability in metagenomics and beyond

Chapter 6

hAMRonization: Enhancing antimicrobial resistance prediction using PHA4GE standards and specification

This chapter is a reproduction of the following manuscript to be submitted for publication in eLife:

C. I. Mendes, E. Griffiths, A. Manuele, D. Fornika, S. Tausch, T. Le-Viet, J. Phelan, C. Meehan, A. Raphenya, B. Alcock, E. Culp, A. McArthur, M. Feldgarden, G. Tyson, M. Galas, J. Campos, A. Witney, D. Aanensen, A. Black, L. Katz, P. Oluniyi, I. Olawoye, R. Timme, A. T. Vasconcelos, A. Page, D. MacCannell, F. Maguire. hAMRonization: Enhancing antimicrobial resistance prediction using PHA4GE standards and specification.

As discussed in Chapters 1 2 and 3, SMg can offer relatively unbiased pathogen detection and characterisation, potentially able to provide antimicrobial resistance and virulence profiling in a single methodological step. The rise of AMR poses a major threat to human health worldwide.

Despite the relative standardisation of the process to acquire genomic data, either through WGS or SMg, there's a myriad of different tools available to perform *in silico* AMR detection, with some using their own reference database and each one generating a unique, non-standardised report of the genes or variants that can possibly confer resistance in a given sample. This is a huge barrier to the comparison of results and the modularity of tools within bioinformatic workflows.

In this chapter, we present a standardised output specification for the reporting of genes or variants potentially conferring AMR. This specification aims to standardise the reporting of these results, regardless of the tool and/or database used. To ease its utilisation, we've packaged this specification into hAMRonization, a command-line utility that is able to aggregate results from a wide variety of AMR detection tools, both species-agnostic and species-specific, providing a unified report tabular form, JSON or through an interactive HTML file that can be opened within the browser for navigable data exploration.

We aim that with hAMRonization, stakeholders, such as clinical practitioners, will be more efficiently informed of potential AMR genes of variants in a given dataset containing genomic information. Ultimately, the possibility of using multiple AMR detection tools, with multiple AMR databases, increases evidence, which increases confidence in the results obtained. Likewise, differences in databases can be evidenced for better interpretability.

My contribution to this publication included the development of the antimicrobial detection contextual data specification package, including its conversion and availability in a machine-applicable JSON format. It also includes the design and implementation of the hAMRonization package, results analysis, manuscript production and editing.

hAMRonization: Enhancing antimicrobial resistance prediction using PHA4GE standards and specification

Catarina I. Mendes[1,*], Emma J. Griffiths[2,*], Alex Manuele[3], Dan Fornika[4], Simon H. Tausch[5], Thanh Le-Viet [6], Amogelang R. Raphenya[7], Brian Alcock[7], Elizabeth Culp[7], Andrew G. McArthur[7], Michael Feldgarden[8], Gregory H. Tyson[9], Marcelo Galas[10], Josefina Campos[11], Adam A. Witney[12], David M. Aanensen[13,14], Allison Black[15], Emma Hodcroft[16,17], Lee S. Katz[18,19], Paul E. Oluniyi[20,21], Idowu B. Olawoye[20,21], Ruth E. Timme[22], Ana Tereza R. Vasconcelos[23], Andrew J. Page[6], Duncan R. MacCannell[19], Finlay Maguire[3] on behalf of the Public Health Alliance for Genomic Epidemiology (PHA4GE) consortium Data Structures Working Group

[1]Instituto de Microbiologia, Instituto de Medicina Molecular, Faculdade de Medicina, Universidade de Lisboa, Lisboa, Portugal

[2] Faculty of Health Sciences, Simon Fraser University, Burnaby, BC, Canada

[3] Faculty of Computer Science, Dalhousie University, NS, Canada

[4] British Columbia Centre for Disease Control, Vancouver, BC, Canada

[5] Department of Biological Safety, German Federal Institute for Risk Assessment, 10589 Berlin, Germany

[6] Quadram Institute, Norwich, UK

[7] Department of Biochemistry and Biomedical Sciences and the Michael G. DeGroote Institute for Infectious Disease Research, McMaster University; Hamilton; Ontario; Canada

[8] National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health; Bethesda; Maryland; USA

[9] Center for Veterinary Medicine, U.S. Food and Drug Administration; Laurel; Maryland; USA

[10] Communicable Diseases and Environmental Determinants of Health, Pan American Health Organization; Washington DC; USA

[11] INEI-ANLIS "Dr Carlos G. Malbrán"; Buenos Aires; Argentina

[12] Institute for Infection and Immunity, St George's, University of London; London; UK

[13] Centre for Genomic Pathogen Surveillance, Wellcome Genome Campus; Cambridge; UK;

[14] The Big Data Institute, Li Ka Shing Centre for Health Information and Discovery,

6. HAMRONIZATION: ENHANCING ANTIMICROBIAL RESISTANCE PREDICTION USING PHA4GE STANDARDS AND SPECIFICATION

Nuffield Department of Medicine, University of Oxford; Oxford; UK

[15] Department of Epidemiology, University of Washington; Washington; USA.

[16] Biozentrum, University of Basel, Basel, Switzerland

[17] Swiss Institute of Bioinformatics; Lausanne; Switzerland

[18] Center for Food Safety, University of Georgia; Georgia; USA;

[19] National Center for Emerging and Zoonotic Infectious Diseases, Centers for Disease Control and Prevention; Georgia; USA

[20] African Center of Excellence for Genomics of Infectious Diseases (ACEGID), Redeemer's University, Ede; Osun State; Nigeria;

[21] Department of Biological Sciences, College of Natural Sciences, Redeemer's University; Ede; Osun State; Nigeria

[22] Center for Food Safety and Applied Nutrition, U.S. Food and Drug Administration; College Park; MD; USA

[23] Bioinformatics Laboratory National Laboratory of Scientific Computation LNCC/MCTI; Rio de Janeiro; Brazil

* Contributed equally

6.1 Abstract

The detection of antimicrobial resistance (AMR) directly from genomic or metagenomic data has become a standard procedure in public health, with a large number of different bioinformatic tools currently available to perform this task. These tools, although implementing similar principles, differ in supported inputs, search algorithms, parameterisation, and underlying reference databases. Each of these tools generates a report of detected AMR genes or variants in a distinct, non-standard, format. This is a huge barrier to the comparison of results and the modularity of tools within bioinformatic workflows.

The Public Health Alliance for Genomic Epidemiology (PHA4GE) (`https://pha4ge.org`) has developed a standardized output specification for the bioinformatic detection of AMR from genomes. hAMRonization, a python package and command-line utility, implements PHA4GE's AMR specification to combine the outputs of disparate antimicrobial resistance gene detection tools into a single unified format. hAMRonization can be easily extended, currently supporting 18 different tools, both species-agnostic and species-specific, for the detection of genes and/or variants conferring AMR. The harmonized re-

ports are available in tabular form, JSON or through an interactive HTML file that can be opened within the browser for navigable data exploration. The hAMRonization and underlying specification are open-source and freely available through PyPI, conda and GitHub (`https://github.com/pha4ge/hAMRonization`).

6.1.0.1 Keywords

interoperability; antimicrobial resistance; public health; workflows

6.2 Introduction

Antimicrobial resistance (AMR) represents a present and growing public health crisis with a global impact. Multidrug resistance is increasing in a broad range of pathogens [1]; combined with low rates of antimicrobial drug discovery [2] this represents a threat to human and animal health [3]. National and international action plans e.g., [3–6], have identified several strategies to mitigate the risk of AMR such as rapid diagnosis of the AMR determinants present within a clinical sample, improved surveillance of AMR, and gaining a better understanding of the mechanisms of environmental AMR transmission.

Diagnostic and public health surveillance analyses are increasingly performed using genomic and metagenomic data [7]. Therefore, accurate identification of genes or variants from genomic data which are predicted to confer resistance to antimicrobials is critical for monitoring and attempting to mitigate the spread of AMR. This work is being performed all over the world by public health agencies, clinicians, industry, and academic researchers. Given the scale of this problem and the number of stakeholders involved, many bioinformatics tools have been developed that are dedicated to the task of AMR gene and variant detection [7–9]. There are at least 18 active (as of 2022) open-source command-line tools designed to identify the presence of genes and/or variants associated with AMR.

These tools include several developed to work with a specific primary database, e.g., the Resistance Gene Identifier (RGI) for the Comprehensive Antibiotic Resistance Database (CARD) [10], AMRFinderPlus and the National Center for Biotechnology Information (NCBI) Pathogen Detection Reference Gene catalogue [11], and ResFinder and KmerResistance [12] for the ResFinder database [13]. Other tools exist that use merged forms of existing databases such as ResFams [14], AMRplusplus [15], and DeepARG [16]. Finally, there are AMR gene identification tools that provide a database-agnostic approach using novel algorithms (e.g., GROOT [17], ARIBA [18]), or a different interface (e.g., ABRicate [19], sraX [20]).

These tools all have different strengths and weaknesses exist attributable to varied underlying databases, search algorithms, and default parameterisations. Based on the specific requirements of a given AMR analysis context one tool may be better suited to the analytical workflow than another. Unfortunately, all these tools also generate differently formatted outputs using divergent terminology. This poses a significant challenge to the effective integration, modularity, and comparison of AMR gene detection methods. The consequence of this is that it is difficult for researchers and public health experts to systematically evaluate the suitability of different tools in their workflow. With the limited examples of these comparisons largely reliant on the development of custom ad-hoc tool-specific parsers, e.g., [11, 18]), or has skipped the tools entirely and directly compared the underlying algorithms e.g., [21]. Even in cases where benchmarking has been performed, the disparate output formats mean it requires significant work to modify a workflow to use a different AMR detection tool. This greatly limits modularity and flexibility to integrate new tools into existing analyses or repurpose them to new or changing requirements. Given the recent calls to mitigate these issues in public health genomic epidemiology [22], it is critical that data structures and methods are developed which enable tool-agnostic, robust parsing, manipulation, and transformation of AMR gene detection results.

To this end, we compared and consolidated the outputs of existing AMR gene detection tools to develop the hAMRonization specification. This is a standardised set of recommended and mandatory output terms and labels for AMR gene detection tools, such as “Gene Name”, “% Coverage (breadth)”, and “Drug Class”. The outputs for all currently maintained, general purpose, AMR gene detection tools can be directly converted to this unified specification. Furthermore, this conversion to a common AMR gene detection output specification can be performed automatically using a companion library of biopython-compatible parsers.

6.3 Results

6.3.1 The PHA4GE hAMRonization Detection Specification

The PHA4GE hAMRonization Detection specification was designed to harmonize information prioritised by public health practitioners for genomic surveillance, outbreak investigations and research with direct public health impacts. Fields from various widely used AMR detection tools were compared, and a set of 36 fields were selected to capture information regarding the software and database provenance, information about reference genes and genomes, as well as mutations and resistance genes detected in sequences of interest. The field labels are standardised using terms sourced or contributed to open-source ontologies. Ontologies are sets of controlled vocabulary arranged in a hierarchy, in which terms are linked using logical relationships and the meanings of the terms are disambiguated by the assignment of unique and persistent identifiers. The positions of detected mutations with respect to reference genes/genomes/proteins are specified using the machine-readable Human

Table 6.1: hAMRonization: AMR Gene Detection Output Specification.

Table available for download at `https://zenodo.org/record/7286376/files/PHA4GE%20AMR%20Gene%20%26%20Variant%20Specification%20-%202022.xlsx`

Genome Variation Society (HGVS) notation system, a sequence variant nomenclature that has been used previously for encoding tuberculosis drug resistance mutations (den Dunnen et al., 2016). To translate HGVS notations into clinician-friendly, human-readable language, a simplified interpretation of mutation coordinates is also provided within the specification. A list of the fields in the specification, along with their ontology identifiers, definitions, expected value types, examples of use, and additional guidance for implementation, are presented in Table 6.1. It should be noted that the guidance is meant to provide recommendations for developers and may contain suggestions that go beyond the immediate aims of the specification.

6.3.2 hAMRonization Data Transformation, Parsing Tools, and Harmonized Reports

To enable automated data transformations from variable tool outputs to the PHA4GE hAMRonization specification, fields from 18 gene and mutation detection tools (i.e. ABRicate, AMRFinderPlus, AMRPlusPlus, ARIBA, C-SSTAR, DeepARG, GROOT, KmerResistance, ResFams, ResFinder (includes PointFinder), SraX, SRST2, and StarAMR) were mapped to the specification. It's worth noting that not all tools provide values for all specification fields. Also, many tools provide outputs beyond what is captured by the specification.

Parsers were then constructed to automate the transformation and movement of outputs into the appropriate fields in the hAMRonization reports (Figure 6.1). Parser code is open source, and all code and installation instructions are available at `https://github.com/pha4ge/hAMRonization`. hAMRonization can be executed from the command-line, and alternatively, can be installed and used in Galaxy via the Galaxy tool shed. hAMRonization provides both tabular and interactive reporting formats. The interactive report provides the user with an HTML file that can be opened within the browser for navigable data exploration.

6.3.3 hAMRonization Snakemake Pipeline

The developed hAMRonization snakemake pipeline (see 6.5.3) was used to run two sets of data through a selection of AMR detection tools, using various AMR gene databases.

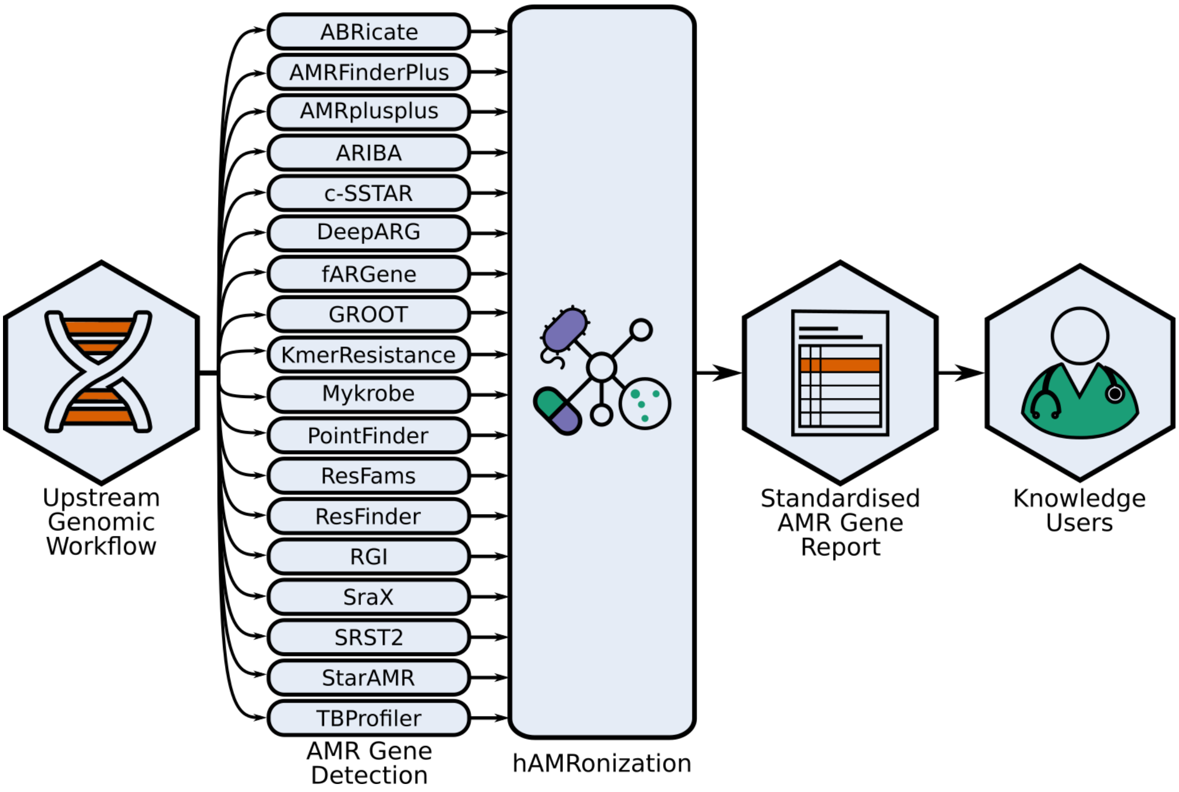


Figure 6.1: Overview of hAMRonization within standard AMR genomics/metagenomics workflows. The hAMRonization module and command-line utility tools allow the creation of a single unified output report from any of 18 established disparate AMR gene and/or variant prediction tools. This facilitates users such as national public health reference laboratories and academic institutions to compare results across tools and change their workflow to different tools without having to develop custom code. It also means that the communication of results to downstream knowledge users, such as infection control prevention clinicians, can be done in a consistent standardised manner regardless of which AMR prediction tool was used in the genomic/metagenomic analysis.

6.3.4 Piloting hAMRonization in Real-World Public Health Settings

To test the utility and value of the hAMRonization specification and its associated tools for AMR genomic surveillance and data sharing, pilot projects were launched in laboratories across three partner teams, which included Cambodia/Australia, Nigeria (seven laboratories across the AMR surveillance network), and Malaysia/Argentina via a PHA4GE subgrant competition. PHA4GE also engaged eight labs from seven countries in Latin America via the ReLAVRA AMR surveillance network (a network associated with the Pan American Health Organization (PAHO), a regional office of the World Health Organization). Participants were issued with a short document outlining the purpose of the software, installation instructions and prompting questions for feedback on their experience. Participants were also asked to share harmonized data among their networks where previously sharing was difficult due to the variability of tool outputs. As hAMRonization and its associated materials are written in English, the documentation had to first be translated into Spanish by PAHO prior to distribution in Latin American countries.

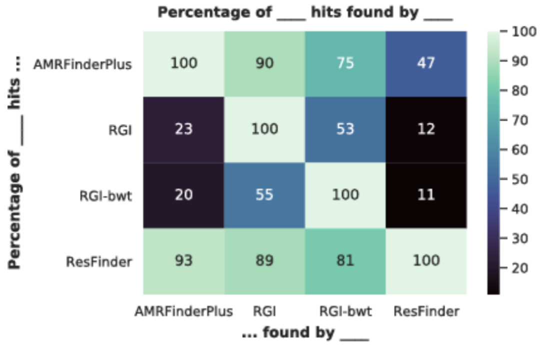
Subgrant participants used hAMRonization to analyse data from hospital settings and/or national surveillance projects involving a variety of organisms e.g. *Salmonella paratyphi*, Methicillin-Resistant *Staphylococcus aureus* (MRSA). Feedback included the need for improved definitions of fields and guidance on required inputs, the need for certain fixes for circumstances causing hAMRonization crashes, and the need for more clinician-friendly installation and execution instructions. Some participants also suggested that the inclusion of sequence contextual data would be helpful for analysis (i.e. hospital name and country of sample collection, ward (medical vs surgical), collection date, specimen type (e.g. blood, urine), latitude/longitude of sample collection). In addition to the testing exercise performed, the Nigeria Team (led by the University of Ibadan) integrated hARMonization into their NextflowTower platform for AMR bioinformatics analysis training workshops. To better enable their clinician colleagues without bioinformatics expertise to use the hAMRonization, the Malaysia/Argentina Team (led by the National University of Malaysia) developed a Google Colaboratory (a Google Notebook that enables code to be executed from a Browser) and has distributed it to their network partners.

The ReLAVRA pilot included National Reference Laboratories from Argentina, Brasil, Chile, Colombia, Costa Rica, México and Paraguay. Laboratories and were provided with a dataset of 20 *Shigella sonnei* genomes from a previous Latin America regional project analysed with 5 different AMR detection tools: ABRIcate, AMRFinderPlus, ARIBA, SRST2, StarAMR. Participants received the outputs of the AMR detection tools together with the instructions for installing and running hAMRonization. When participants completed the exercise, they were asked to complete the structured poll. Twelve responses from the seven participating nations were received, and a summary report was prepared by ReLAVRA and provided to PHA4GE. Most of the participants were bioinformaticians and laboratory technicians. For bioinformaticians, the installation instructions were easy to follow, however, laboratory technicians without a bioinformatics background expressed difficulties largely with the dependencies installation. Participants agreed that all harmonised output fields provided relevant information for AMR detection but the most useful were the fields that cover the AMR gene detected and the confidence of that result. Respondents were concerned about the importance of having knowledge of AMR and genomic data analysis for the correct interpretation of the results, especially with regard to decision-makers in clinical settings. An example related to this concern was that when detecting *blaTEM-1* gene by AMRFinderPlus, the field “antimicrobial agent” indicates "Beta-lactam", but this type of drug includes a wide variety of antibiotics, from amoxicillin to extended-spectrum cephalosporins (ESC) and carbapenems. However, *TEM-1* is an extended-spectrum beta-lactamase that does not confer resistance to ESC or carbapenems. While this concern arises from the output of AMRFinderPlus rather than hAMRonization, this challenge is a barrier to the implementation of genomics for AMR surveillance in clinical settings and should be addressed. PHA4GE is cataloguing such critical challenges and will work with the community to solve them. In terms of output summary format, most of the participants preferred the tabular report, citing that a single file combining all the relevant information was more convenient and amenable to processing with different downstream software. Some participants preferred citing its user-friendly visual interface was helpful for comparing which tools identified the same gene.

While differences in gene nomenclature arise from variability in reference databases, users expressed that harmonization of gene names across different reference databases would be extremely helpful (especially those produced by ARIBA). Fields that require knowledge to interpret should not be included e.g. “antimicrobial agent”. While hAMRonization does not generate the tool-specific error warnings that users sometimes encounter while running the tools, it would be useful to provide explanations of these warnings in the user guide.

While differences in tool algorithms are outside of the scope of hAMRonization, they result in discrepancies in results which are highlighted by hAMRonization e.g. the absence of a result because the gene was not present in the sequence of interest or due to its absence in the database or due to differences in the search algorithms used by each AMR tool. Users suggested that providing some explanation of the differences in tool algorithms would be helpful for interpretation.

Overall, all users found the tool very useful for doing regional genomic surveillance and sharing results with different laboratories that use different AMR detection tools across networks. Technical issues such as fixes for crashing, improved definitions, and instructions for inputs were addressed. Ways to address bigger picture requests such as the harmonization of gene nomenclature, tool-specific issues, and the inclusion of contextual data (sample metadata as well as epidemiological, clinical and laboratory data) are under discussion by PHA4GE.



	AMRFinderPlus	RGI	RGI-bwt	ResFinder
AMRFinderPlus	100	90	75	47
RGI	23	100	53	12
RGI-bwt	20	55	100	11
ResFinder	93	89	81	100

RGI: BLASTp + CARD
RGI bwt (beta): Bowtie2 + CARD
AMRFinderPlus: BLASTx + BARRG DB
ResFinder: BLASTn + ResFinder DB

	RGI	RGI bwt	AMR	Res
Total # unique gene names	140	223	70	46

Figure 6.2: Unique AMR genes idenfied by RGI, RGI bwt, AMRFInderPlus and ResFinder. The heatmap represents the percenetage of AMR genes found by the combination of the four tools. Although a very large percentage of the ResFinder and AMRFinderPlus detected genes are identified by all tools, RGI and in particular RGI bwt identify a very large subset of AMR genes not present in the othe datasets.

6.3.5 hAMRonization and Research Analysis and Biological Insights

To test the utility and value of the hAMRonization specification and its associated tools for biological research, PHA4GE engaged partners from the research community. hAMRonization was used to benchmark tools for the detection of AMR genes in *Klebsiella pneumoniae* by researchers at McMaster University (Hamilton, Canada), and was also used to compare *Pseudomonas aeruginosa* Sequence Types associated with a Metallo-Carbapenemases resistance outbreak in the United Kingdom by researchers at the University of London (London, UK). hAMRonization challenges identified during the pilots were also addressed during an AMR hackathon organised by PHA4GE, CLIMB-Big Data, and the Joint Programming Initiative on Antimicrobial Resistance (JPIAMR).

6.3.5.1 *Klebsiella pneumoniae* and tool benchmarking

A dataset of 89 Klebsiella pneumoniae were analysed using RGI, relying on BLASTp and the CARD database, RGI btw, relying on Bowtie2 and the CARD database, AMRFinderPlus which uses BLASTx and the BARRG database, and ResFinder with BLASTn and the ResFinder database. The aim was to determine the number of unique AMR genes identified, with a large variation having been identified across the four tools (Figure 6.2). RGI bwt and RGI identified the highest number of hits (n=223 and n=140 respectively), albeit using different identification methods both use the same database. AMRFinderPlus and Resfinder identified an order of magnitude less AMR genes (n=70 and n=46 respectively), even though both rely on BLAST, as does RGI. This highlights the impact of the database chosen when assessing AMR genes.

6.3.5.2 Characterizing *Pseudomonas aeruginosa* Sequence Types associated with Metallo-Carbapenemase resistance in an outbreak

A set of 87 *Pseudomonas aeruginosa* sequences (ST 111) associated with a Metallo-Carbapenemases resistance outbreak in the United Kingdom [23], were analysed using ABRIcate, AMRFinderplus, C-SSTAR, Resfinder, StarAMR with default parameters and databases. The resulting outputs were combined using hAMRonization (Figure 6.3). Most isolates (n=73) carried the *VIM-2* gene, conferring the observed resistance phenotype. Some genes, such as *crpP* and *sul1*, are repeatedly identified by most of the tools. Differences in gene nomenclature in different databases complicate comparisons e.g. the same gene was also seen with different spelling, such as *blaVIM-2* being called a *blaVIM*. Nomenclature issues aside, harmonised outputs from different tools served to increase the body of evidence (and confidence in results) indicating the role of different AMR determinants in the outbreak.

6.3.5.3 hAMRonization and Hackathon-based Community Development

A hackathon jointly organized by PHA4GE, CLIMB-Big Data, and JPIAMR was held in October 2021 (`https://github.com/AMR-Hackathon-2021`). The aim of the hackathon was to improve upon currently available bioinformatics tools and methods for the AMR community, with a special focus on antimicrobial resistance in bacteria. A hackathon is an event in which a large number of people from the community meet to engage in collaborative computer programming. Project ideas are pitched, and participants form working groups based on the projects proposed. PHA4GE representatives pitched challenges identified during the hAMRonization pilot implementations as hackathon projects - such as the need for a clinician-friendly interpretation of HGVS notation and the need for harmonization of gene nomenclature across reference databases. As a result of the collaborative work during the hackathon, the hAMRonization specification was improved by the development of a layman's translation of HGVS notation specifying the coordinates of mutations identified within genes, proteins and genomic sequences of interest in relation to their position to a reference sequence (led by Dr Connor Mehan, University of Blah). Furthermore, the use of an ontology-based tool for harmonizing gene nomenclature called CHARMdb was explored and tested in hARMonization (led by Dr Anthony Underwood, previously of Public health England, currently at Broken String Biosciences Ltd). While CHARMdb is not yet fully integrated into hAMRonization, we anticipate its use in future development. These achievements are examples of the important work that can be done when scientists are brought together to use their creativity and expertise to solve problems at community-driven hackathons.

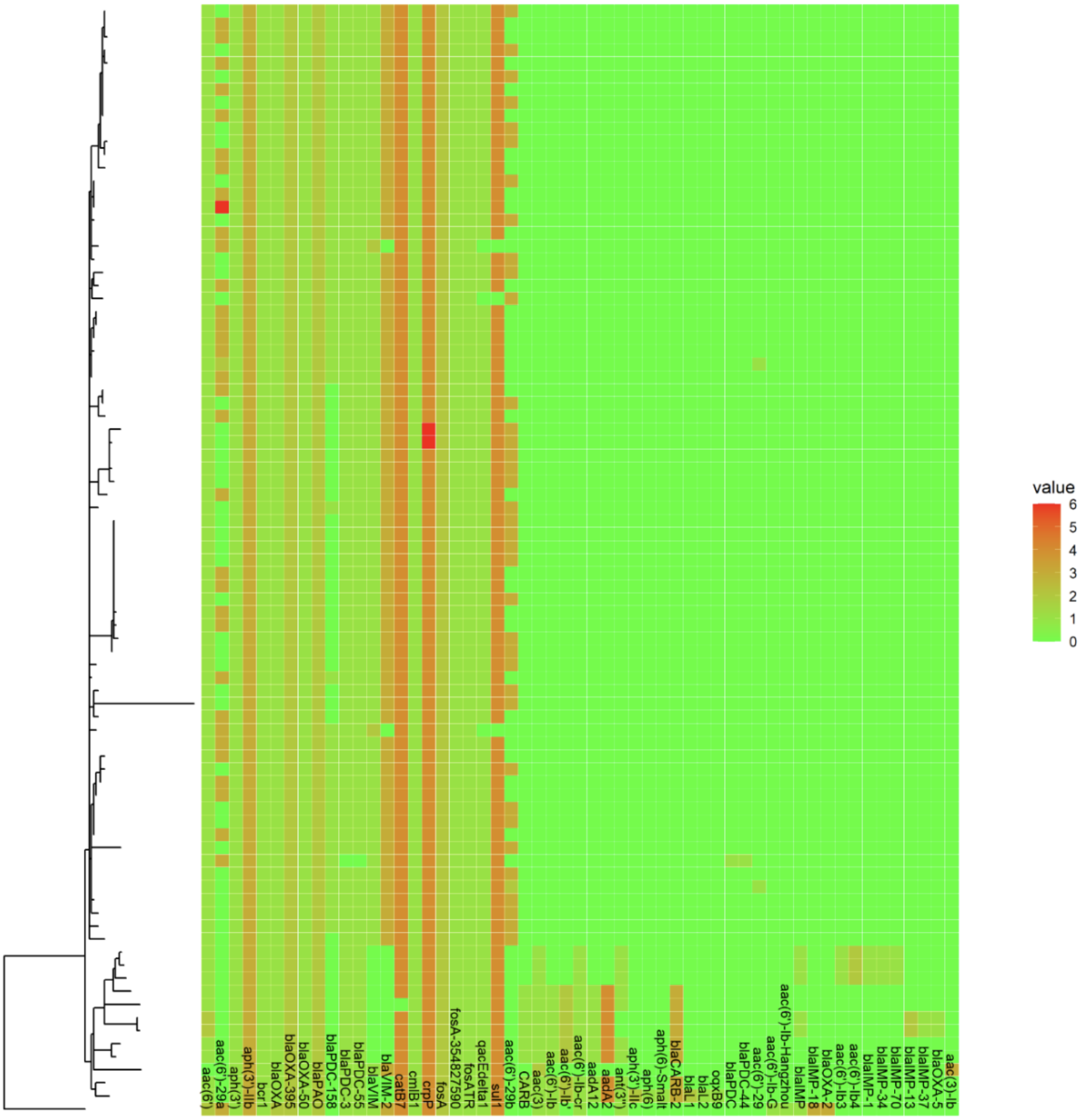


Figure 6.3: Heatmap of the number of predictions made by the AMR gene detection tools (abricate, amrfinderplus, csstar, resfinder, staramr) that find each resistance gene, overlaid on a tree from those isolates.

6.4 Discussion

The lack of standardisation in the reporting of AMR gene and mutation detection greatly hinders the comparison of results across the public health sector. The myriad of options available for this purpose highlights a grave interoperability problem. In this work, we have identified and prioritised data elements capturing AMR detection information most pertinent for public health decision-making. We have developed a standardised specification for harmonising variable AMR detection tool outputs, implementing open source, community-based ontologies as well as standardised formats and nomenclatures. We have also operationalised the specification through the development of tools (parsers and workflows) and harmonised reports, available as a package called hAMRonization. We have also tested the hAMRonization in different public health and research settings. The results of this work have shown that hARMonization enables the dissemination of results to stakeholders in a single consistent format, allowing not only the comparison of tools and databases, but the validation of results through multiple detection algorithms.

The developed standardized data specification and python package and command line utility improve data harmonization and interoperability, allowing the comparison of results not only across multiple AMR detection algorithms but also AMR reference databases. hAMRonization also improves the capture of provenance information, such as the versions of software and databases used in analyses, which can help improve the reproducibility of results.

It should be noted that while hAMRonization has been applied to AMR detection, many data elements are use case agnostic and so can be repurposed for the detection of genes and mutations associated with other phenotypes such as virulence, pathogenicity, transmissibility, heavy metal resistance, mobility factors etc. Extension of the specification through the adding of other optional higher-order annotation fields such as virulence factor type (“adherence”, “toxin” etc) or metal type, could help to create modularity, different types of detection streams, and feed directly into other downstream analysis. A limitation of hAMRonization currently is that while individual determinants contributing to resistance can be detected however there is no support for highlighting the presence of multi-component resistance mechanisms such as glycopeptide resistance clusters [24] or efflux pumps [25] in this specification.

One of the most substantial challenges with any attempt to improve interoperability by introducing new standards/specifications is their uptake. We have attempted to minimise the barriers to uptake by operationalizing it with a set of automated data transformation tools. For each of the AMR detection tools, a set of “mappings” or instructions to convert their outputs to the hAMRonization specification has been provided along with parsers to automate this process. Ideally, developers will adopt this specification and structure their tool’s output accordingly in the future. This would obviate the need for future mappings and parsers.

However, even if this does not occur, the nature of the specification means that only minimal effort would be required to convert tabular results from any novel AMR gene detection tool. Another benefit of such a community-based consensus standard for AMR detection is that it provides much-needed standards for industrial partners developing instruments and devices. A challenge within public health laboratories is the lack of interoperability between different phenotype characterization instruments and lab information management systems. Providing community-vetted, standardised specifications for genomics applications enables vendors to develop tools and other resources that are interoperable and add value (for example, an AMR typing platform conforming to the hAMRonization standard could produce results that could more easily plug into a LIMS of other downstream software more seamlessly).

The implementation of hAMRonization in real-world settings has enabled us to more clearly catalogue and articulate public health challenges outside of the scope of the specification and tools, that impact the use of hAMRonization. Lessons learned to include the fact that the harmonization of data for genomic surveillance comes with additional needs outside of standards and technical implementation, including improved support for the interpretation and evaluation of harmonised data. Also, in harmonising the outputs of tools using different computational approaches, the strengths and weaknesses of the tools need to be made more transparent to users as they become removed from working directly with the individual tools themselves. Future work will focus on these areas, as well as better harmonising gene nomenclature across reference databases.

In summary, through the uptake of hAMRonization data standard and associated tools, we aim to improve the comparability of results, and the interoperability of workflows in the public health microbial genomics ecosystem. This will lead to better AMR surveillance, more effective genomic diagnostics, and ultimately improved human and animal health.

6.5 Materials and Methods

6.5.1 PHA4GE hAMRonization Specification

To compare and unify the broad set of output fields across individual AMR gene detection tools it is necessary to create a common language that can be used to describe them all. This takes the form of a standardised set of output labels and ontological identifiers (linked to standardised definitions). Individual fields from the output of each tool can then be mapped to a standardised label. For example, the accession of the contig a given AMR gene was detected on is listed as “Contig” in RGI, “contig_name” in ResFinder, and "contig id" in AMRFinderPlus. Therefore, our specification includes a standardised “Contig ID” field which captures all of these values across tools.

This common language, or specification, was created through an iterative expert-led process which consisted of curating and comparing output formats from 18 open-source, species-agnostic AMR gene detection tools, with evidence of active use in the literature. This included consultation with an international set of national public health groups, tool developers, and AMR genomics experts through the PHA4GE initiative. In addition to identifying terms compatible with the range of outputs, this work also involved the identification of a core set of essential contextual data for use of AMR detection in health or research contexts. This includes items such as the input filename, name and version of the AMR database used, and the name and version of the AMR gene detection tool. Once the prioritised data elements were selected, definitions and implementation guidance for the standardised fields were developed. Appropriate ontology terms matching the desired data elements were sourced from different OBO Foundry ontologies. Ontologies within the OBO Foundry library are developed using common principles and practices designed to increase interoperability across domains of knowledge. Terms in the draft AMR specification for which ontology concepts already existed were adjusted to align with community standards. Terms which did not already exist were submitted to appropriate source ontologies. Ontologies are meant to represent “universal truth” as much as possible, and not just use case-specific nuances. As such, additional guidance was provided for AMR detection-specific implementation. Fields, definitions, specification-specific guidance, value types, and examples of use can be found in the hAMRonization specification available on GitHub. In total, the specification consists of 36 fields sourced/harmonised with the Ontology for Biological Investigations (OBI), the Antimicrobial Resistance Ontology (ARO), the Genomic Epidemiology Ontology (GenEpiO), and the Chemical Entities of Biological Interest Ontology (ChEBI). The finalised hAMRonization specification was then encoded in JSON format to enable automated computer validation of hAMRonized output reports and deposited within the hAMRonization git repository.

6.5.2 hAMRonization Package

Mappings from the output fields in each of the 18 tools to the hAMRonization specification were then encoded in a python package (github.com/pha4ge/hAMRonization). This package was developed following the modular approach of biopython [26] and provides an easy-to-use command line tool to enable automatic conversion of output files from each of these tools to a standardised report following the hAMRonization specification. Where appropriate and when tools don’t provide the full set of essential core metadata will prompt for this information to ensure high-quality reproducible analyses. This tool allows the collation of results from multiple tools and samples into a single standardised report to aid comparison and analysis. These reports can optionally be generated as spreadsheets (CSV), JSON, and an interactive navigable HTML format to maximise utility and compatibility with downstream processes. Finally, this package also provides a python module to enable direct integration of the functions into other python software. To ensure practical installation in as broad a range of computing infrastructures across varied public health, academic, and industry research settings this tool is open source and is freely available via GitHub, PyPI, bioconda, docker, and the galaxy toolshed.

6.5.3 hAMRonization Proof of Concept Workflow

As a proof-of-concept, a snakemake pipeline [27] was created which automatically runs arbitrary microbial genomic data through a set of species-agnostic AMR gene detection tools, supported by hAMRonization, and generates a single report for each. This set is composed by abricate (version 1.0.1), amrfinderplus [11] (version 3.10.1), amrplusplus [15] (v1.0.0), ariba [18](version 2.14.6), c-SSTAR [28] (version 1.1.01), groot [17] (version 1.1.2), kmer-resistance [12]((version 2.2.0), resfams [14] via hmmer (v3.3.2), resfinder [29] (version 3.2), rgi [10] (version 5.1.1), srax [20] (version 1.5), srst2 [30] (version 0.2.0) and staramr [13] (version 0.7.2).In order to guarantee the reproducibility of the results obtained, it integrates fixed versions of the tools implemented in the pipeline from conda. The pipeline is available at `https://github.com/pha4ge/hAMRonization_workflow`.

6.6 Acknowledgements

Contributors to hAMRonization on GH. Participants of AMR hackathon organised by PHA4GE, CLIMB-Big Data, and the Joint Programming Initiative on Antimicrobial Resistance (JPIAMR)

6.7 Disclaimer

Nothing to declare.

6.8 Funding statements

All authors have read this manuscript and consented to its publication. We wish to thank the Bill and Melinda Gates Foundation for supporting the establishment and work of the PHA4GE consortium. CIM was supported by the Fundação para a Ciência e Tecnologia (grant SFRH/BD/129483/2017). Work by EJG was funded by a Genome Canada Bioinformatics and Computational Biology 2017 Grant (286GET) and a 2018 Canadian Institutes of Health Research (CIHR) Project Grant (APP356925). FM is supported by a Donald Hill Family Fellowship.

6.9 Supplemental Materials

Table 6.2: Scope, functionality, inputs and outputs of widely used AMR gene prediction tools.

Table available for download at `https://zenodo.org/record/7286376/files/PHA4GE%20AMR%20Gene%20%26%20Variant%20Specification%20-%202022.xlsx`

Table 6.3: PHA4GE AMR Gene Detection Output Specification Package resources.

Table available for download at `https://zenodo.org/record/7286376/files/PHA4GE%20AMR%20Gene%20%26%20Variant%20Specification%20-%202022.xlsx`

6.10 References

[1] Bilal Aslam et al. “Antibiotic resistance: a rundown of a global crisis”. In: *Infection and Drug Resistance* 11 (Oct. 2018), pp. 1645–1658. ISSN: 1178-6973. DOI: 10.2147/IDR.S173867. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6188119/ (visited on 07/21/2020).

[2] Eric D. Brown and Gerard D. Wright. “Antibacterial drug discovery in the resistance era”. en. In: *Nature* 529.7586 (Jan. 2016). Number: 7586 Publisher: Nature Publishing Group, pp. 336–343. ISSN: 1476-4687. DOI: 10.1038/nature17042. URL: https://www.nature.com/articles/nature17042 (visited on 07/20/2020).

[3] WHO. *WHO | Global action plan on antimicrobial resistance*. Tech. rep. Publisher: World Health Organization. 2015. URL: http://www.who.int/antimicrobial-resistance/publications/global-action-plan/en/ (visited on 07/20/2020).

[4] Jim O’Neill. *Antimicrobial Resistance: Tackling a Crisis for the Future Health and Wealth of Nations*. Tech. rep. 2014.

[5] Jim O’Neill. *Tackling Drug-Resistance Infections Globally: Final Report and Recommendations*. Tech. rep. 2016.

[6] Public Health Agency of Canada. *Antimicrobial Resistance and Use in Canada: A Federal Framework for Action*. eng. policies. Last Modified: 2014-11-13. Oct. 2014. URL: https://www.canada.ca/en/public-health/services/antibiotic-antimicrobial-resistance/antimicrobial-resistance-use-canada-federal-framework-action.html (visited on 07/20/2020).

[7] Andrew G. McArthur and Kara K. Tsang. “Antimicrobial resistance surveillance in the genomic age”. eng. In: *Annals of the New York Academy of Sciences* 1388.1 (2017), pp. 78–91. ISSN: 1749-6632. DOI: 10.1111/nyas.13289.

[8] Manish Boolchandani, Alaric W. D’Souza, and Gautam Dantas. “Sequencing-based methods and resources to study antimicrobial resistance”. en. In: *Nature Reviews Genetics* (Mar. 2019). ISSN: 1471-0056, 1471-0064. DOI: 10.1038/s41576-019-0108-4. URL: http://www.nature.com/articles/s41576-019-0108-4 (visited on 07/20/2020).

[9] Rene S. Hendriksen et al. “Using Genomics to Track Global Antimicrobial Resistance”. In: *Frontiers in Public Health* 7 (Sept. 2019). ISSN: 2296-2565. DOI: 10.3389/fpubh.2019.00242. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6737581/ (visited on 07/20/2020).

[10] Brian P Alcock et al. “CARD 2020: antibiotic resistome surveillance with the comprehensive antibiotic resistance database”. In: *Nucleic Acids Research* 48.D1 (Jan. 2020), pp. D517–D525. ISSN: 0305-1048. DOI: 10.1093/nar/gkz935. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7145624/ (visited on 07/20/2020).

[11] Michael Feldgarden et al. “Validating the AMRFinder Tool and Resistance Gene Database by Using Antimicrobial Resistance Genotype-Phenotype Correlations in a Collection of Isolates”. In: *Antimicrobial Agents and Chemotherapy* 63.11 (Oct. 2019). ISSN: 0066-4804. DOI: `10.1128/AAC.00483-19`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6811410/` (visited on 07/20/2020).

[12] Philip T. L. C. Clausen et al. “Benchmarking of methods for identification of antimicrobial resistance genes in bacterial whole genome data”. en. In: *Journal of Antimicrobial Chemotherapy* 71.9 (Sept. 2016). Publisher: Oxford Academic, pp. 2484–2488. ISSN: 0305-7453. DOI: `10.1093/jac/dkw184`. URL: `https://academic.oup.com/jac/article/71/9/2484/2238319` (visited on 07/21/2020).

[13] E. Zankari et al. “Identification of acquired antimicrobial resistance genes”. en. In: *Journal of Antimicrobial Chemotherapy* 67.11 (Nov. 2012), pp. 2640–2644. ISSN: 0305-7453, 1460-2091. DOI: `10.1093/jac/dks261`. URL: `https://academic.oup.com/jac/article-lookup/doi/10.1093/jac/dks261` (visited on 06/02/2021).

[14] Molly K Gibson, Kevin J Forsberg, and Gautam Dantas. “Improved annotation of antibiotic resistance determinants reveals microbial resistomes cluster by ecology”. In: *The ISME Journal* 9.1 (Jan. 2015), pp. 207–216. ISSN: 1751-7362. DOI: `10.1038/ismej.2014.106`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4274418/` (visited on 07/21/2020).

[15] Enrique Doster et al. “MEGARes 2.0: a database for classification of antimicrobial drug, biocide and metal resistance determinants in metagenomic sequence data”. en. In: *Nucleic Acids Research* 48.D1 (Jan. 2020). Publisher: Oxford Academic, pp. D561–D569. ISSN: 0305-1048. DOI: `10.1093/nar/gkz1010`. URL: `https://academic.oup.com/nar/article/48/D1/D561/5624973` (visited on 07/21/2020).

[16] Gustavo Arango-Argoty et al. “DeepARG: a deep learning approach for predicting antibiotic resistance genes from metagenomic data”. In: *Microbiome* 6.1 (Feb. 2018), p. 23. ISSN: 2049-2618. DOI: `10.1186/s40168-018-0401-z`. URL: `https://doi.org/10.1186/s40168-018-0401-z` (visited on 07/21/2020).

[17] Will P M Rowe and Martyn D Winn. “Indexed variation graphs for efficient and accurate resistome profiling”. en. In: *Bioinformatics* 34.21 (Nov. 2018). Ed. by John Hancock, pp. 3601–3608. ISSN: 1367-4803, 1460-2059. DOI: `10.1093/bioinformatics/bty387`. URL: `https://academic.oup.com/bioinformatics/article/34/21/3601/4995843` (visited on 06/02/2021).

[18] Martin Hunt et al. “ARIBA: rapid antimicrobial resistance genotyping directly from sequencing reads”. en. In: *Microbial Genomics* 3.10 (Oct. 2017). ISSN: 2057-5858. DOI: `10.1099/mgen.0.000131`. URL: `https://www.microbiologyresearch.org/content/journal/mgen/10.1099/mgen.0.000131` (visited on 06/02/2021).

[19] Torsten Seeman. *Abricate*. 2020.

[20] Leonardo G. Panunzi. “sraX: A Novel Comprehensive Resistome Analysis Tool”. In: *Frontiers in Microbiology* 11 (Feb. 2020), p. 52. ISSN: 1664-302X. DOI: `10.3389/fmicb.2020.00052`. URL: `https://www.frontiersin.org/article/10.3389/fmicb.2020.00052/full` (visited on 06/02/2021).

[21] C. McCall and I. Xagoraraki. “Comparative study of sequence aligners for detecting antibiotic resistance in bacterial metagenomes”. eng. In: *Letters in Applied Microbiology* 66.3 (Mar. 2018), pp. 162–168. ISSN: 1472-765X. DOI: `10.1111/lam.12842`.

[22] Allison Black et al. “Ten recommendations for supporting open pathogen genomic analysis in public health”. en. In: *Nature Medicine* 26.6 (June 2020). Number: 6 Publisher: Nature Publishing Group, pp. 832–841. ISSN: 1546-170X. DOI: `10.1038/s41591-020-0935-z`. URL: `https://www.nature.com/articles/s41591-020-0935-z` (visited on 07/20/2020).

[23] A.A. Witney et al. “Genome sequencing and characterization of an extensively drug-resistant sequence type 111 serotype O12 hospital outbreak strain of Pseudomonas aeruginosa”. en. In: *Clinical Microbiology and Infection* 20.10 (Oct. 2014), O609–O618. ISSN: 1198743X. DOI: `10.1111/1469-0691.12528`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S1198743X14653945` (visited on 06/02/2021).

[24] Patrice Courvalin. “Vancomycin resistance in gram-positive cocci”. eng. In: *Clinical Infectious Diseases: An Official Publication of the Infectious Diseases Society of America* 42 Suppl 1 (Jan. 2006), S25–34. ISSN: 1537-6591. DOI: `10.1086/491711`.

[25] H. Okusu, D. Ma, and H. Nikaido. “AcrAB efflux pump plays a major role in the antibiotic resistance phenotype of Escherichia coli multiple-antibiotic-resistance (Mar) mutants”. eng. In: *Journal of Bacteriology* 178.1 (Jan. 1996), pp. 306–308. ISSN: 0021-9193. DOI: `10.1128/jb.178.1.306-308.1996`.

[26] P. J. A. Cock et al. “Biopython: freely available Python tools for computational molecular biology and bioinformatics”. en. In: *Bioinformatics* 25.11 (June 2009), pp. 1422–1423. ISSN: 1367-4803, 1460-2059. DOI: `10.1093/bioinformatics/btp163`. URL: `https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/btp163` (visited on 06/02/2021).

[27] J. Koster and S. Rahmann. “Snakemake–a scalable bioinformatics workflow engine”. en. In: *Bioinformatics* 28.19 (Oct. 2012), pp. 2520–2522. ISSN: 1367-4803, 1460-2059. DOI: `10.1093/bioinformatics/bts480`. URL: `https://academic.oup.com/bioinformatics/article-lookup/doi/10.1093/bioinformatics/bts480` (visited on 07/22/2020).

[28] Tom J. B. de Man and Brandi M. Limbago. “SSTAR, a Stand-Alone Easy-To-Use Antimicrobial Resistance Gene Predictor”. en. In: *mSphere* 1.1 (Feb. 2016). Ed. by Paul Dunman. ISSN: 2379-5042. DOI: `10.1128/mSphere.00050-15`. URL: `https://journals.asm.org/doi/10.1128/mSphere.00050-15` (visited on 06/02/2021).

[29] Valeria Bortolaia et al. “ResFinder 4.0 for predictions of phenotypes from genotypes”. en. In: *Journal of Antimicrobial Chemotherapy* 75.12 (Dec. 2020), pp. 3491–3500. ISSN: 0305-7453, 1460-2091. DOI: `10.1093/jac/dkaa345`. URL: `https://academic.oup.com/jac/article/75/12/3491/5890997` (visited on 06/02/2021).

[30] Michael Inouye et al. “SRST2: Rapid genomic surveillance for public health and hospital microbiology labs”. In: *Genome Medicine* 6.11 (Nov. 2014), p. 90. ISSN: 1756-994X. DOI: `10.1186/s13073-014-0090-6`. URL: `https://doi.org/10.1186/s13073-014-0090-6` (visited on 08/06/2020).

Chapter 7

Future-proofing and maximising the utility of metadata: The PHA4GE SARS-CoV-2 contextual data specification package

This chapter is a reproduction of the following publication:

E. J. Griffiths, R. E. Timme, C. I. Mendes, A. J. Page, N. Alikhan, D. Fornika, F. Maguire, J. Campos, D. Park, I. B. Olawoye, P. E. Oluniyi, D. Anderson, A. Christoffels, A. G. da Silva, R. Cameron, D. Dooley, L. S. Katz, A. Black, I. Karsch-Mizrachi, T. Barrett, A. Johnston, T. R. Connor, S. M. Nicholls, A. A. Witney, G. H. Tyson, S. H. Tausch, A. R. Raphenya, B. Alcock, D. M. Aanensen, E. Hodcroft, W. W. L. Hsiao, A. T. R. Vasconcelos, D. R. MacCannell on behalf of the Public Health Alliance for Genomic Epidemiology (PHA4GE) consortium. Future-proofing and maximizing the utility of metadata: The PHA4GE SARS-CoV-2 contextual data specification package. GigaScience, Volume 11, giac003 (2022). DOI: `https://doi.org/10.1093/gigascience/giac003`

As mentioned in Chapter 1, the SARS-CoV-2 has brought a new meaning to genomic surveillance, with currently, over 8 million complete viral sequences available at GISAID[*], being one of the most highly sequenced genomes of any organism on the planet. This richness in genomic information has been basal to identifying new variants of risk and new variants of concern with a myriad of different origins, identifying routes of transmission across borders, including the identification of "super-spreaders" events, and informing infection control measures.

Despite this richness in genomic information, the same is not observed for the metadata that accompanies it. As described in Chapter 7, a standardised output specification was conceived, this time applied to SARS-CoV-2 contextual data based on harmonisable, publicly available community standards. This is implemented through a collection template, as well as a variety of protocols and tools to support both the harmonisation and submission of

---

[*] `https://www.gisaid.org/`

sequence data and contextual information to public biorepositories.

My contribution to this publication included the development of the SARS-CoV-2 contextual data specification package, including its conversion and availability in a machine-applicable JSON format. I've also maintained the public repository where the data specification package is hosted[†] Additionally, I've also contributed to the manuscript production and editing.

[†] `https://github.com/pha4ge/SARS-CoV-2-Contextual-Data-Specification/`

Future-proofing and maximizing the utility of metadata: The PHA4GE SARS-CoV-2 contextual data specification package

Emma J. Griffiths[1], Ruth E. Timme[2], Catarina I. Mende[3], Andrew J. Page[4], Nabil-Fareed Alikha[4], Dan Fornika[5], Finlay Maguire[6], Josefina Campos[7], Daniel Park[8], Idowu B. Olawoy[9,10], Paul E. Oluniy[9,10], Dominique Anderson[11], Alan Christoffel[11], Anders Gonçalves da Silva[12], Rhiannon Cameron[1], Damion Dooley[1], Lee S. Katz[13], Allison Black[14], Ilene Karsch-Mizrach[15], Tanya Barret[15], Anjanette Johnston[15], Thomas R. Connor[16,17], Samuel M. Nicholls[18], Adam A. Witney[19], Gregory H. Tyson[20], Simon H. Tausch[21], Amogelang R. Raphenya[22], Brian Alcock[22], David M. Aanensen[23,24], Emma Hodcrof[25], William W. L. Hsiao[1,5,26], Ana Tereza R. Vasconcelos[27], Duncan R. MacCannel[28], on behalf of the Public Health Alliance for Genomic Epidemiology (PHA4GE) consortium

[1] Faculty of Health Sciences, Simon Fraser University, Burnaby, British Columbia,Canada;

[2] Center for Food Safety and Applied Nutrition, U.S. Food and Drug Administration, College Park, MD,USA;

[3] Instituto de Microbiologia, Instituto de Medicina Molecular, Faculdade de Medicina, Universidade de Lisboa, Portugal;

[4] Quadram Institute Bioscience, Norwich, Norfolk, UK;

[5] BC Centre for Disease Control Public Health Laboratory, Vancouver, Canada;

[6] Faculty of Computer Science, Dalhousie University, Halifax, Canada;

[7] INEI-ANLIS "Dr Carlos G. Malbrán", Buenos Aires, Argentina;

[8] The Broad Institute of MIT and Harvard, Cambridge, MA, USA;

[9] African Center of Excellence for Genomics of Infectious Diseases (ACEGID), Redeemer's University, Ede, Osun State, Nigeria;

[10] Department of Biological Sciences, College of Natural Sciences, Redeemer's University, Ede, Osun State, Nigeria;

[11] South African Medical Research Council Bioinformatics Unit, South African National Bioinformatics Institute, University of the Western Cape, Bellville, South Africa;

[12] Microbiological Diagnostic Unit Public Health Laboratory, The Peter Doherty Institute for Infection and Immunity, The University of Melbourne, Melbourne, Victoria, Australia;

[13] Center for Food Safety, University of Georgia, Georgia, USA;

[14] Department of Epidemiology, University of Washington, Washington, USA;

[15] National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health, Bethesda, MD, USA;

[16] Organisms and Environment Division, School of Biosciences, Cardiff University, Cardiff, Wales, UK;

[17] Public Health Wales, University Hospital of Wales, Cardiff, UK;

[18] University of Birmingham, Birmingham, UK;

[19] Institute for Infection and Immunity, St George's, University of London, London, UK;

[20] Center for Veterinary Medicine, U.S. Food and Drug Administration, Laurel, Maryland, USA;

[21] Department of Biological Safety, German Federal Institute for Risk Assessment, Berlin, Germany;

[22] Department of Biochemistry and Biomedical Sciences and the Michael G. DeGroote Institute forInfectious Disease Research, McMaster University, Hamilton, Ontario, Canada;

[23] Centre for Genomic Pathogen Surveillance, Wellcome Genome Campus, Cambridge, UK;

[24] The Big Data Institute, Li Ka Shing Centre for Health Information and Discovery, Nuffield Department of Medicine, University of Oxford, Oxford, UK;

[25] Biozentrum, University of Basel, Basel, Switzerland & Swiss Institute of Bioinformatics, Lausanne,Switzerland;

[26]Department of Pathology and Laboratory Medicine, University of British Columbia, Vancouver, Canada;

[27] Bioinformatics Laboratory National Laboratory of Scientific Computation LNCC/MCTI, Rio de Janeiro, Brazil;

[28] National Center for Emerging and Zoonotic Infectious Diseases, Centers for Disease Control and Prevention, Georgia, USA

7.1 Abstract

The Public Health Alliance for Genomic Epidemiology (PHA4GE) (https://pha4ge.org) is a global coalition that is actively working to establish consensus standards, document and share best practices, improve the availability of critical bioinformatics tools and resources, and advocate for greater openness, interoperability, accessibility and reproducibility in public health microbial bioinformatics. In the face of the current pandemic, PHA4GE has identified a need for a fit-for-purpose, open-source SARS-CoV-2 contextual data standard. As such, we have developed a SARS-CoV-2 contextual data specification package based on harmonisable, publicly available community standards. The specification can be implemented via a collection template, as well as an array of protocols and tools to support both the harmonisation and submission of sequence data and contextual information to public biorepositories. Well-structured, rich contextual data adds value, promotes reuse, and enables aggregation and integration of disparate data sets. Adoption of the proposed standard and practices will better enable interoperability between datasets and systems, improve the consistency and utility of generated data, and ultimately facilitate novel insights and discoveries in SARS-CoV-2 and COVID-19. The package is now supported by the National Center for Biotechnology (NCBI)'s BioSample database.

7.2 Findings

7.2.1 The importance of contextual data for interpreting SARS-CoV-2 sequences

First identified in late 2019 in Wuhan, China, the SARS-CoV-2 virus has now spread to virtually every country and territory in the world, resulting in millions of confirmed cases, and deaths, globally [1, 2]. Understanding, monitoring and preventing transmission, as well as the development of vaccines and effective therapeutic options, have been primary goals of the public health response to SARS-CoV-2.

Tracking the spread and evolution of the virus at global, national and local scales has been aided by the analysis of viral genome sequence data alongside SARS-CoV-2 epidemiology. Large scale sequencing efforts are often formalised as consortia across the world, including the COG-UK in the UK [3], SPHERES in the USA [4], CanCOGeN in Canada [5], Latin American Genomics SARS-CoV-2 Network [6, 7], 2019nCoVR in China [8], the South Africa NGS Genomic Surveillance Network [9], AusTrakka in Australia and New Zealand [10], and INSACOG in India [11]. In addition to these initiatives, many agencies, universities and hospital laboratories around the world are also sequencing and sharing sequence data at an unprecedented pace. Deposition of these sequences into public repositories such as the Global Initiative on Sharing All Influenza Data (GISAID) and the International Nucleotide Sequence Database Collaboration (INSDC) has enabled rapid global sharing of data [12, 13]. At the time of writing, 174 countries had undertaken open sequencing initiatives (GISAID accessed 2021-06-23) depositing 2,057,675 sequences which are being reused and analysed on a massive scale. The open data sharing paradigm has had tremendous success in the genomic epidemiology of foodborne pathogens [14, 15], and has the potential to reveal a deeper understanding of SARS-CoV-2 origin, pathogenicity, and basic biology when submissions from environmental samples and wild hosts are included alongside human clinical samples [16].

SARS-CoV-2 sequencing, analysis, and open sharing have played a crucial role in a number of developments during the pandemic, such as dispelling misinformation about the origins of the virus [17], the identification and surveillance of variants of concern [18], [19], the improvement of diagnostic performance and rapid testing [20–22], and the development of vaccines which are currently being distributed in the largest global vaccination program the world has ever seen [23]. Viral genomic sequences are also being used to understand transmission and reinfection events [24] as well to monitor the prevalence and diversity of lineages during different exposure events and in different settings e.g. animal reservoirs [25], long-term care facilities [26–28], healthcare and other work sites [29–33], conferences and other public gatherings [34], as well before and after public health responses (e.g. border controls and travel restrictions, lockdowns and quarantines, vaccination, etc.), through successive waves of infections [35–46]. However, it is critical to note that public health sequence data is of limited value without accompanying contextual metadata.

Contextual data consists of sample metadata (e.g., collection date, sample type, geographical location of sample collection), as well as laboratory (e.g., date and location testing, cycle threshold (CT) values), clinical outcomes (e.g., hospitalisation, death, recovery), epidemiological (e.g., age, gender, exposures, vaccination status) and methods (e.g., sampling, sequencing, bioinformatics) that enable the interpretation sequence data (e.g., previous examples). High-quality contextual data is also crucial for quality control. For example, detecting systematic batch effect errors related to certain sequencing centres and methods can help evaluate which variants represent real, circulating viruses, as opposed to artefacts of sample handling or sequencing which may arise due to different aspects experimental design, laboratory procedures, bioinformatics processing, and applied quality control thresholds [47–49].

Good data stewardship practices are not only critical for auditability and reproducibility, but for posterity - documenting critical information about samples, methods, risk factors and outcomes etc., can help future-proof information used to build a roadmap for dealing with future public health crises. Contextual data, however, is often collected on a project-specific basis according to local needs and reporting requirements which results in the collection of different data types at different levels of granularity, with different meanings and implicit bias of variables and attributes. Furthermore, the information is often collected as free text, or if structured, according to organisation or initiative-specific data dictionaries, using different fields, terms, formats, abbreviations, and jargon.

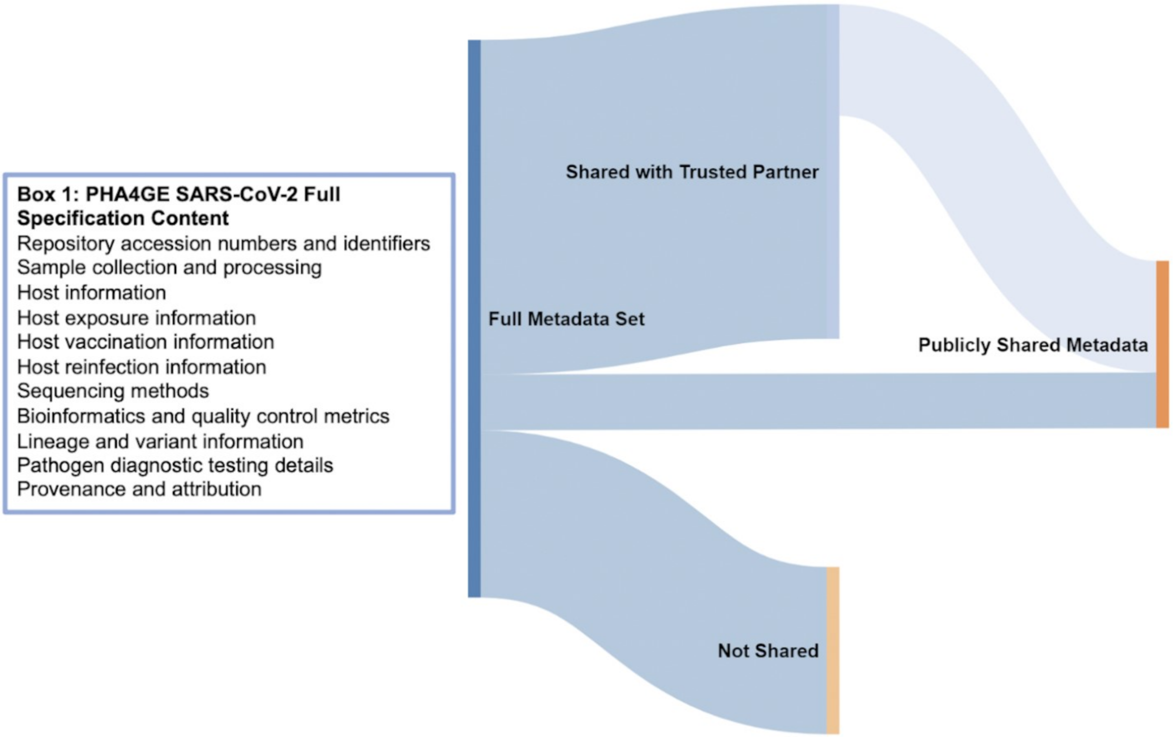
The variability in the way information is encoded in private databases tends to propagate to public repositories, which makes the information more difficult to interpret and to use. There are different existing standards that can be used to structure contextual data, like minimum information checklists (MIxS [50], MIGS [51], the NIAID/BRC Project and Sample Application Standard [52]) and various interoperable ontologies (OBO Foundry [53]), which make information easier to aggregate and reuse for different types of analyses. However, these attribute packages and metadata standards developed by different organisations are usually scoped to cover as many use cases and pathogens as possible, and as such, can include fields of information not applicable to SARS-CoV-2, or that may be subject to privacy concerns, or exclude fields commonly used in public health surveillance and investigations. As different types of contextual data are subject to different ethical, practical and privacy concerns, not all components of existing standards are immediately or widely collectable and shareable. As a result, the range of generic metadata standards being applied to SARS-CoV-2 data presents challenges for data harmonization [54] and analysis critical for fighting the disease and ending the pandemic.

In light of these challenges, PHA4GE has identified a need for a fit-for-purpose, open-source SARS-CoV-2 contextual data specification which can be used to consistently structure information as part of good data management practices and for data sharing with trusted partners and/or public repositories. The specification was developed by consensus among domain experts, and incorporates existing community standards with an emphasis on SARS-CoV-2 public health needs and ensuring privacy while maximising information content and interoperability across datasets and databases to better enable analyses to fight COVID-19. The specification package also contains a number of accompanying materials such as standard operating procedures, tools, a reference guide, and repository submission protocols (protocols.io) to help put the standard into practice.

7.2.2 SARS-CoV-2 Contextual Data Specification: The Framework

The purpose of the PHA4GE SARS-CoV-2 specification is to provide a mechanism for consistent structure, collection and formatting of fields and values containing SARS-CoV-2 contextual data pertaining to clinical, animal, and environmental samples. We emphasise that the purpose of this specification is not to force data sharing, but rather to provide a framework to structure data consistently across disparate laboratory and epidemiological databases so that they can be harmonised for different uses (Figure 7.1). Data sharing is just one use case and can involve sharing between divisions within a single agency, sharing between partners based on memorandums of understanding, or submission to public repositories.

The PHA4GE SARS-CoV-2 contextual data specification was created through broad consultation with representatives from public health laboratories, research institutes and universities in 11 countries (Argentina, Australia, Brazil, Canada, Germany, Nigeria, Portugal, South Africa, Switzerland, the United Kingdom, the United States of America) who are involved with the SARS-CoV-2 genome sequencing and analysis efforts at various scales. Based on this consultation and consensus, the specification contains different fields covering a wide array of data types described in Box 1 (Figure 7.1). The specification attempts to harmonise different data standards (INSDC, GISAID, MIxS, MIGS, Sample Application Standard) by reusing fields or mapping to fields, as much as possible. As PHA4GE embraces FAIR data stewardship principles (Findability, Accessibility, Interoperability and Reuse of digital assets), we strived to implement FAIR principles in the design and implementation of the specification for data management and data sharing. At their core, these principles emphasise machine-actionability and consistency of data, and are critical for dealing with the volume and complexity of genomic sequence and contextual data. Principles of FAIR data stewardship that have been implemented include improving machine-actionability of data by using a formal, accessible, shared, and broadly applicable language for knowledge representation, reusing existing standards and ontology-based vocabulary to increase interoperability, providing a data usage license, capturing data provenance, and making all resources open, free and widely accessible.



Box 1: PHA4GE SARS-CoV-2 Full Specification Content
Repository accession numbers and identifiers
Sample collection and processing
Host information
Host exposure information
Host vaccination information
Host reinfection information
Sequencing methods
Bioinformatics and quality control metrics
Lineage and variant information
Pathogen diagnostic testing details
Provenance and attribution

Figure 7.1: Contextual data flow. Contextual data can be captured and structured using the PHA4GE specification so that they can be more easily harmonised across different data sources and providers. Different subsets of the harmonised data can be (i) shared with public repositories, e.g., GISAID and INSDC; (ii) shared with trusted partners, e.g., national sequencing consortia, public health partners; and (iii) kept private and retained locally with the potential for sharing in the future for particular surveillance or research activities. While fields have been colour-coded in the template to indicate whether they are considered "required," "strongly recommended," or "optional," how the specification is implemented and whether any of the data are shared is ultimately at the discretion of the user. Box 1 describes the information types covered in the full specification.

The versioned specification is available as a contextual data collection template (.xlsx) and in machine-amenable JSON format from GitHub (v 3.0.0 - https://github.com/pha4ge/SARS-CoV-2-Contextual-Data-Specification) [55]. The collection template also offers standardised terms for a number of fields in the form of pick lists. The fields are colour-coded to indicate required (yellow), strongly recommended (purple) or optional status (white). Fields useful for surveillance were prioritised as required. Formats for data elements like dates are also prescribed according to international standards (e.g., dates should be formatted according to ISO8601).

The template is also supported by several materials such as term and field-level Reference Guides (available as tabs in the collection template Excel workbook), which provides definitions, data entry guidance and examples of usage [55]. The field-level Reference Guide also provides mapping of PHA4GE fields to existing contextual data standards, highlighting public health and SARS-CoV-2-specific fields that were missing as well as fields in those other standards that were considered out of scope.

The Open Biological and Biomedical Ontology (OBO) Foundry is a community of researchers that use a prescribed set of principles and practices to develop a wide range of interoperable ontologies focused on the life sciences [56]. Fields and terms in the specification have been mapped to existing OBO Foundry ontology terms, and where required, new ontology terms have been developed and are being made available in different application and domain-specific ontologies within The Foundry (see Table 7.1 for a list of source ontologies). As of version 3.0.0 and beyond, terms in pick lists provided in the collection template are presented with corresponding ontology identifiers in the format “Label [ontology ID]” e.g., Blood [UBERON:0000178]. Axioms and additional cross references to ontologies and existing standards are actively being developed in collaboration with community developers. We anticipate that our contributions to these freely available, open-source resources will be of use to the COVID-19 research community.

Protocols have also been created and are openly available on protocols.io [57], including a curation Standard Operating Procedure (SOP) containing instructions for using the collection template as well as guidance for a number of privacy and practical concerns. A series of versioned SARS-CoV-2 sequence and contextual data submission protocols and accompanying instructional videos for how to prepare submissions and navigate through the various submission portals for GISAID, NCBI and EMBL-EBI are also provided.

A mapping file indicating which PHA4GE fields correspond to contextual data elements recommended by the World Health Organization has been provided to help data providers comply with international guidance [58]. This mapping file also includes tabs indicating which PHA4GE fields correspond to those found in different repository submission forms to facilitate data transformations for submissions. Such transformations can be automated using a contextual data harmonization application called the DataHarmonizer [59]. PHA4GE has worked with the developers of the DataHarmonizer to offer the PHA4GE standard as a

Table 7.1: Ontologies implemented in the PHA4GE SARS-CoV-2 specification.

Ontology[1]	**Link**
BRENDA Tissue Ontology (BTO)	https://obofoundry.org/ontology/bto.html
Cell Line Ontology (CLO)	https://obofoundry.org/ontology/clo.html
Environmental conditions, treatments and exposures ontology (ECTO)	https://obofoundry.org/ontology/ecto.html
Environment Ontology (ENVO)	https://obofoundry.org/ontology/envo.html
Food Ontology (FoodOn)	https://obofoundry.org/ontology/foodon.html
Gazetteer Ontology (GAZ)	https://obofoundry.org/ontology/gaz.html
Gender, Sex, and Sexual Orientation Ontology (GSSO)	https://obofoundry.org/ontology/gsso.html
Genomic Epidemiology Ontology (GenEpiO)	https://obofoundry.org/ontology/genepio.html
Genomics Cohorts Knowledge Ontology (GECKO)	https://obofoundry.org/ontology/gecko.html
Human Disease Ontology (DOID)	https://obofoundry.org/ontology/doid.html
Human Phenotype Ontology (HP)	https://obofoundry.org/ontology/hp.html
Mammalian Phenotype Ontology (MP)	https://obofoundry.org/ontology/mp.html
Measurement Method Ontology (MMO)	https://obofoundry.org/ontology/mmo.html
Mondo Disease Ontology (MONDO)	https://obofoundry.org/ontology/mondo.html
Mouse Pathology Ontology (MPATH)	https://obofoundry.org/ontology/mpath.html
National Cancer Institute Thesaurus (NCIT)	https://obofoundry.org/ontology/ncit.html
NCBI Taxonomy Ontology (NCBITaxon)	https://obofoundry.org/ontology/ncbitaxon.html
Neuro Behaviour Ontology (NBO)	https://obofoundry.org/ontology/nbo.html
Ontology for Biomedical Investigations (OBI)	https://obofoundry.org/ontology/obi.html
Ontology of Medically Related Social Entities (OMRSE)	https://obofoundry.org/ontology/omrse.html
Population and Community Ontology (PCO)	https://obofoundry.org/ontology/pco.html
UBERON Multi-species Anatomy Ontology (UBERON)	https://obofoundry.org/ontology/uberon.html
Unit Ontology (UO)	https://obofoundry.org/ontology/uo.html
Vaccine Ontology (VO)	https://obofoundry.org/ontology/vo.html

[1] Vocabulary for fields and terms in the specification have been sourced or mapped to OBO Foundry domain and application ontologies, which are highlighted in this list. New fields and terms for which there were no existing equivalents have been developed and submitted to these ontologies, expanding these community resources

template in the tool (Gill et al, in preparation). Users can standardize and validate entered data and export it as GISAID and NCBI-ready submission forms (BioSample, SRA, GenBank and GenBank source modifier forms). It should be noted that other excellent contextual data transformation tools have been developed by the community, such as METAGENOTE, multiSub, and a GISAID-to-ENA conversion script [60–62].

A table outlining the different specification package materials can be found in Table 7.2.

7.2.3 Getting Started - How To Use The Standard

In designing the specification we first considered the goals of data collection and harmonization. Consulted stakeholders felt that the primary priority of standardising data should be improved support for SARS-CoV-2 genomic surveillance activities and the submission of sequence data and minimal metadata to public repositories. The two most important attributes for tracking transmission from pathogen genomic data are temporal information describing when a sample was collected and spatial information describing where a virus was sampled.

Comparisons of minimal contextual data requirements across different national sequenc-

Table 7.2: Resources that form the PHA4GE SARS-CoV-2 contextual data specification package

Resource[1]	**Description**
Collection template and controlled vocabulary pick lists	Spreadsheet-based collection form containing different fields (identifiers and accessions, sample collection and processing, host information, host exposure, vaccination and reinfection information, lineage and variant information, sequencing, bioinformatics and QC metrics, diagnostic testing information, author acknowledgements). Fields are colour-coded to indicate required, recommended or optional status. Many fields offer pick lists of controlled vocabulary. Vocabulary lists are also available in a separate tab.
Reference guides	Field and term definitions, guidance, and examples are provided as separate tabs in the collection template .xlsx file (see Term Reference Guide and Field Reference Guide).
Curation protocol on protocols.io	Step-by-step instructions for using the collection template are provided in a standard operating procedure (SOP). Ethical, practical, and privacy considerations are also discussed. Examples and instructions for structuring sample descriptions as well as sourcing additional standardized terms (outside those provided in pick lists) are also discussed.
Mapping file of PHA4GE fields to metadata standards	PHA4GE fields are mapped to existing metadata standards such as the Sample Application Standard, MIxS 5.0, and the MIGS Virus Host-associated attribute package. Mappings are available in the Reference guide tab. Mappings highlight which fields of these standards are considered useful for SARS-CoV-2 public health surveillance and investigations, and which fields are considered out of scope.
Mapping of PHA4GE fields to WHO metadata recommendations	PHA4GE fields are mapped to corresponding contextual data elements recommended by the World Health Organization.
Mapping file of PHA4GE fields to EMBL-EBI, NCBI and GISAID submission requirements	Many PHA4GE fields have been sourced from public repository submission requirements. The different repositories have different requirements and field names. Repository submission fields have been mapped to PHA4GE fields to demonstrate equivalencies and divergences.
Data submission protocol (NCBI) on protocols.io	The SARS-CoV-2 submission protocol for NCBI provides step-by-step instructions and recommendations aimed at improving interoperability and consistency of submitted data.
Data submission protocol (EMBL-EBI) on protocols.io	The SARS-CoV-2 submission protocol for ENA provides step-by-step instructions and recommendations aimed at improving interoperability and consistency of submitted data.
Data submission protocol (GISAID) on protocols.io	The SARS-CoV-2 submission protocol for GISAID provides step-by-step instructions and recommendations aimed at improving interoperability and consistency of submitted data.
JSON structure of PHA4GE specification	A JSON structure of the PHA4GE specification has been provided for easier integration into software applications.
PHA4GE template in the DataHarmonizer	Javascript application enabling standardized data entry, validation and export of contextual data as submission-ready forms for GISAID and NCBI. The SOP for using the software can be found at https://github.com/Public-Health-Bioinformatics/DataHarmonizer/wiki/PHA4GE-SARS-CoV-2-Template

[1] There are a number of resources that form the PHA4GE SARS-CoV-2 contextual data specification package which are described in the table. The package has been compiled to support user implementation and data sharing, with integration into workflows and new software applications in mind.

ing efforts, as well as submission requirements for INSDC and GISAID databases, yielded a minimal set of 14 fields which have been annotated as “required” in the specification (colour-coded yellow in the collection template). The required fields, corresponding definitions, and guidance notes are described in Table 7.3. A number of other fields have been annotated as “strongly recommended” (colour-coded purple in the collection template) for capturing sample collection and processing methods, critical epidemiological information about the host, and acknowledging scientific contributions. Fields colour-coded white are considered optional.

Table 7.3: Minimal (required) contextual data fields. Through consultation and consensus, fourteen fields were prioritized for SARS-CoV-2 surveillance, which are considered required in the specification. Field names, definitions, and guidance are presented.

Field Name	**Definition**	**Guidance**
specimen collector sample ID	The user-defined name for the sample.	Every Sample ID from a single submitter must be unique. It can have any format, but we suggest that you make it concise, unique and consistent within your lab, and as informative as possible.
sample collected by	The name of the agency that collected the original sample.	The name of the agency should be written out in full, (with minor exceptions) and consistent across multiple submissions.
sequence submitted by	The name of the agency that generated the sequence.	The name of the agency should be written out in full, (with minor exceptions) and be consistent across multiple submissions.
sample collection date	The date on which the sample was collected.	Record the collection date accurately in the template. Required granularity includes year, month and day. Before sharing this data, ensure this date is not considered identifiable information. If this date is considered identifiable, it is acceptable to add "jitter" to the collection date by adding or subtracting calendar days. Do not change the collection date in your original records. Alternatively,” received date” may be used as a substitute in the data you share. The date should be provided in ISO 8601 standard format "YYYY-MM-DD".
geo_loc name (country)	Country of origin of the sample.	Provide the country name from the pick list in the template.
geo_loc name (state/province/region)	State/province/region of origin of the sample.	Provide the state/province/region name from the GAZ geography ontology. Search for geography terms here: https://www.ebi.ac.uk/ols/ontologies/gaz
organism	Taxonomic name of the organism.	Use “Severe acute respiratory syndrome coronavirus 2”
isolate	Identifier of the specific isolate.	This identifier should be an unique, indexed, alpha-numeric ID within your laboratory. If submitted to the INSDC, the "isolate" name is propagated throughout different databases. As such, structure the "isolate" name to be ICTV/INSDC compliant in the following format: "SARS-CoV-2/host/country/sampleID/date"

Continue on next page

Table7.3- *Continued from previous page*

Field Name	**Definition**	**Guidance**
host (scientific name)	The taxonomic, or scientific name of the host.	Common name or scientific name are required if there was a host. Scientific name examples e.g., Homo sapiens. Select a value from the pick list. If the sample was environmental, put "not applicable".
host disease	The name of the disease experienced by the host.	This field is only required if there was a host. If the host was a human select COVID-19 from the pick list. If the host was asymptomatic, this can be recorded under “host health state details”. "COVID-19" should still be provided if the patient is asymptomatic. If the host is not human, and the disease state is not known or the host appears healthy, put “not applicable”.
purpose of sequencing	The reason that the sample was sequenced.	The reason why a sample was originally collected may differ from the reason why it was selected for sequencing. The reason a sample was sequenced may provide information about potential biases in sequencing strategy. Provide the purpose of sequencing from the picklist in the template. The reason for sample collection should be indicated in the "purpose of sampling" field.
sequencing instrument	The model of the sequencing instrument used.	Select a sequencing instrument from the picklist provided in the template.
consensus sequence software name	The name of software used to generate the consensus sequence.	Provide the name of the software used to generate the consensus sequence.
consensus sequence software version	The version of the software used to generate the consensus sequence.	Provide the version of the software used to generate the consensus sequence.

As many contextual data fields are stored in different locations and databases (e.g., LIMS, epidemiology case report forms and databases), a benefit of implementing the PHA4GE collection template is that it enables the capture of these different pieces of information in one place. The collection template also offers picklists for a variety of fields e.g., a curated INSDC country list for “geo_loc name (country)”, the standardised name of the virus under the “organism” field (i.e., Severe acute respiratory coronavirus 2), and a multitude of standardised terms for sample types (anatomical materials and sites, environmental materials and sites, collection devices and methods). The “purpose of sequencing” field provides standardised tags which can be used to highlight sampling strategy criteria (e.g., baseline surveillance (random sampling), targeted sequencing (non-random sampling), which are very important for understanding bias when interpreting patterns in sequence data. The picklists provided are neither exhaustive, nor comprehensive, but have been curated from current literature representing active sampling and surveillance activities.

If a pick list is missing standardised terms of interest, the reference guide also provides links to different ontology look-up services enabling users to identify additional standardised terms. The reference guide provides definitions for the fields, additional guidance regarding the structure of the values in the field, and any suggestions for addressing issues pertaining to privacy and identifiability. The curation SOP provides users with step-by-step instructions for populating the template, looking up standardised terms, and how best to structure sample descriptions. The SOP also highlights a number of ethical, practical, and privacy considerations for data sharing.

7.2.4 Implementation of the PHA4GE specification around the world

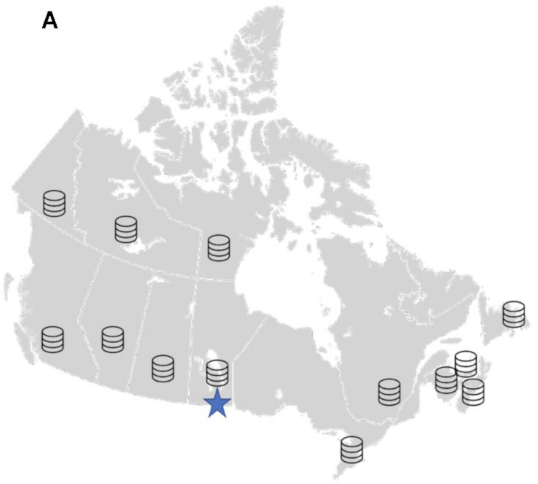
The amount of, and manner in which the specification is implemented is ultimately at the discretion of the user. To date, versions of the specification are being implemented in the CanCOGeN (Canada) and SPHERES (USA) SARS-CoV-2 sequencing initiatives, the AusTrakka (Australia and New Zealand) data sharing platform [1–3], by the Global Emerging Pathogens Treatment Consortium (Africa) [63], the African Centre of Excellence for Genomics of Infectious Diseases (ACEGID) in Nigeria [64], the Baobab LIMS [65] at the South African National Bioinformatics Institute (SANBI) [66], and the Latin American Genomics Network [67].

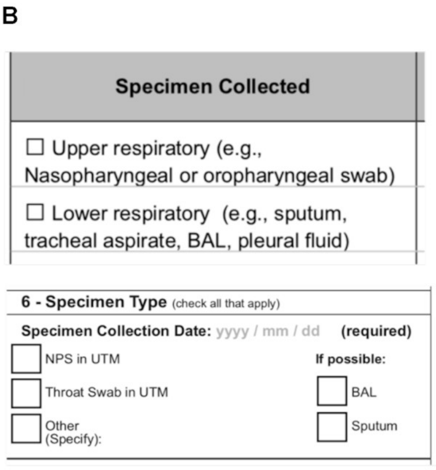
Canada is implementing a version of the PHA4GE specification to harmonise contextual data across all data providers for national SARS-CoV-2 surveillance [5]. Harmonised contextual information is provided by different jurisdictions, and stored in the national genomics surveillance database at the Public Health Agency of Canada's National Microbiology Laboratory. A worked example is provided to demonstrate how free text information can be structured according to the specification, and how subsets of the contextual data can be shared according to jurisdictional policies (Figure 7.2).

While the primary use case of the specification is for clinical sequencing, the sample collection fields have been developed to enable capture of information for a wide range of sample types, including environmental samples (e.g., swabs of hospital equipment and patient rooms, wastewater samples) and non-human hosts (e.g., wildlife, agricultural animal samples).

7.2.5 Submitting Data to Public Sequence Repositories

Most existing SARS-CoV-2 sequences have only been deposited in GISAID, with a proportion of submitters also depositing matching raw read data in the INSDC (i.e., National Center for Biotechnology Information (NCBI), European Molecular Biology Laboratory-European Bioinformatics Institute (EMBL-EBI) and DNA Data Bank of Japan (DDBJ)). While consensus genomes are widely deposited and used for public surveillance purposes, raw read data is critical for comparing methods, assessing reproducibility, as well as identi-





Specimen Collected

☐ Upper respiratory (e.g., Nasopharyngeal or oropharyngeal swab)

☐ Lower respiratory (e.g., sputum, tracheal aspirate, BAL, pleural fluid)

6 - Specimen Type (check all that apply)

Specimen Collection Date: yyyy / mm / dd **(required)**

☐ NPS in UTM

☐ Throat Swab in UTM

☐ Other (Specify):

If possible:

☐ BAL

☐ Sputum

C

Anonymised Example Data:

Province A screens travelers as part of an international border testing program. A sample is selected by the lab team (Johnny Bloggs, Bhav Singh, Tina Lee) for sequencing as part of the **travel surveillance** program. The sample, collected **on November 13 2020**, is a **nasal swab** from an individual who **shares a household with a known case** that **recently traveled to country X**. The individual was **asymptomatic**. Diagnostic RT-qPCR testing based on the E gene yielded a **CT value of 23**. The individual was a **43 year old female**. The lab implements the **ARTIC protocol** to perform amplicon sequencing on an **Illumina MiSeq** using the **primer scheme described by Freed et al (2020), and ncov-tools** for bioinformatic processing and analysis.

Standardized Contextual Data:

specimen collector sample ID: provA_12345
sample collected by: Province A Public Health Lab
sample collector contact email: provlabA@lab.ca
sequence submitted by: Province A Public Health Lab
sequence submitter contact email: provlabA@lab.ca
geo_loc_name (country): Canada [GAZ:00002560]
geo_loc_name (state/province/region): Province A
sample collection date: 2020-11-13
anatomical site: Nasopharynx (NP) [UBERON:0001728]
collection device: Swab [GENEPIO:0100027]
purpose of sampling: Diagnostic testing [GENEPIO:0100002]
purpose of sequencing: International travel surveillance [GENEPIO:0100014]
host (scientific name): Severe acute respiratory syndrome coronavirus 2 [NCBITaxon:2697049]
host disease: COVID-19 [MONDO:0100096]
host age: 43
host age bin: 40 – 49 [GENEPIO:0100053]
host age unit: year [UO:0000036]
host gender: Female [NCIT:C46110]
host health state: Asymptomatic [NCIT:C3833]
exposure setting: Contact with known COVID-19 case [GENEPIO:0100184]
sequencing instrument: Illumina MiSeq [GENEPIO:0100125]
sequencing protocol name: ARTIC protocol
amplicon pcr primer scheme: Freed at al (2020)
amplicon size: 1200 bp
raw sequence data processing method: https://github.com/jts/ncov-tools
dehosting method: https://github.com/jts/ncov-tools
consensus sequence software name: Freebayes
consensus sequence software version: 1.3.2
gene name 1: E gene (orf4) [GENEPIO:0100151]
diagnostic pcr Ct value 1: 23
authors: Johnny Bloggs, Bhav Singh, Tina Lee

Figure 7.2: The PHA4GE specification is being implemented in CanCOGeN to harmonise contextual data across jurisdictions. (A) CanCOGeN is Canada's SARS-CoV-2 national genomic surveillance initiative. Canada has a decentralised health system, with one federal and 13 provincial/territorial public health jurisdictions. Provinces/Territories have authority over how data are collected, stored, and shared. Every Canadian public health jurisdiction uses different collection instruments (e.g., case report forms), different data management systems, and different pipelines and software to perform bioinformatic analyses. Provinces/Territories share sequencing data and accompanying contextual data with the National Microbiology Lab's national SARS-CoV-2 genomics database (starred) according to a version of the PHA4GE specification for national surveillance activities. (B) Excerpts from two different province-specific case collection forms. Sample type information is collected in data collection instruments using different fields, different terms, at different levels of granularity, using abbreviations and formats. BAL: bronchoalveolar lavage; NPS: nasopharyngeal swab; UTM: universal transport medium. (C) An anonymised example of how the standard consistently structures contextual information and how it is being used for data sharing. The contextual data specification provides a wide variety of fields and pick lists of terms. In the example, the full set of standardised information shown would be shared by the province with the national database. Standardised information in boldface would be shared with public repositories; however select data elements (underscored) would be withheld according to jurisdictional data sharing policies. The specification enables users to harmonise and integrate data provenance, sampling strategy criteria, epidemiological information, and methods.

fying minor variants. Linkage of contextual data to consensus sequences as well as raw data in public repositories is vital.

Within the INSDC, the contextual data is stored as accessioned BioSamples [68] with a consistent set of attribute names and standardised values. BioSamples add value, promote reuse, and enable interoperability of data submitted from laboratories that may only be connected by following the same metadata standard. The INSDC databases have until recently provided a generic pathogen metadata template for the BioSample that is heavily utilised for bacterial genomic surveillance [69]. GISAID uses a different format and data structure for associating metadata primarily for influenza surveillance and now extended to include SARS-CoV-2. The ENA provides a virus metadata checklist (ENA virus pathogen reporting standard checklist) developed as part of the COMPARE project [70], which is very similar to the GISAID submission requirements.

Building off of these existing standards, a metadata specification for SARS-CoV-2 genomic surveillance was developed that is broad enough for internal laboratory use while providing mechanisms for mapping/transforming standardised contextual data for public release to INSDC and GISAID. Recently, PHA4GE worked with NCBI to develop a dedicated SARS-CoV-2 BioSample submission package in the NCBI Submission Portal, which incorporates many fields from the PHA4GE standard [71]. The Genomics Standards Consortium will also align its forthcoming “MIxS for SARS-CoV-2” package with this specification. EMBL-EBI will also offer the PHA4GE standard to submitters as one of its validated checklists. Taken together, the PHA4GE specification has already had widespread impact on contextual information data structures around the world.

The detailed mapping of PHA4GE fields to public repository submission requirements, as well as guidance and advice, are available as supporting documents (see Table 7.1). We have also provided detailed protocols for data submission to the three participating repositories, GenBank/SRA (NCBI), ENA (EMBL-EBI), and GISAID. An overview of how the PHA4GE specification is integrated into public repository submissions is presented in Figure 7.3. PHA4GE recommendations for FAIR SARS-CoV-2 data submissions are as follows:

1. submit raw sequencing data and assembled/consensus genomes to INSDC and GISAID when permitted by jurisdictional data sharing policies
2. create a BioSample record when submitting to the INSDC using the PHA4GE guidance, populating the mandatory and recommended fields where possible
3. curate public records (sequence data and contextual data), updating them when subsequent information becomes available or retracting if/when records become untrustworthy.

The specification has been used to submit standardised contextual data to different repositories by labs and sequencing initiatives globally. A selection of accession numbers for

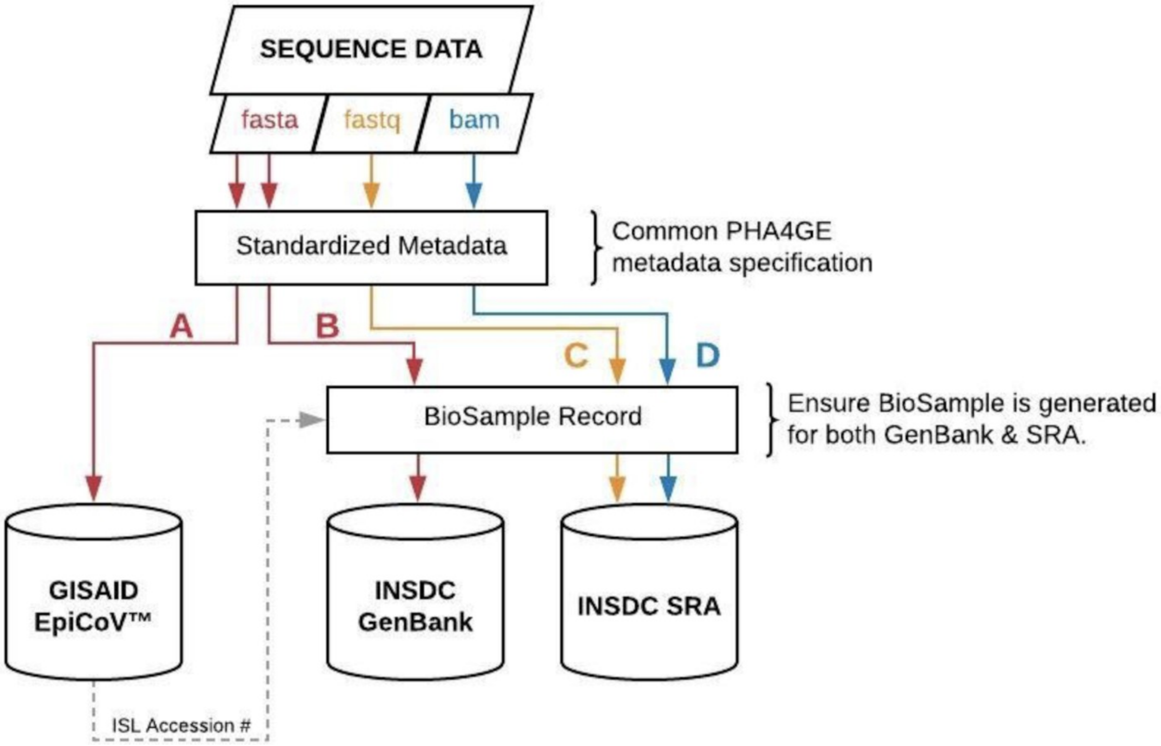


Figure 7.3: Overview of how the PHA4GE SARS-CoV-2 contextual data specification can be integrated into public repository submission. The PHA4GE collection template provides a one-stop shop for different data types that are important for global surveillance. The protocols provided as part of the specification package describe how PHA4GE fields can be mapped to different repository submission forms. Consensus sequences (FASTA), accompanied by a subset of PHA4GE fields, can be submitted to the GISAID EpiCoV database (A). Consensus sequences (FASTA) (B) as well as raw/processed data (FASTQ, BAM) (C, D) can be submitted to INSDC databases (e.g., GenBank, SRA) with different subsets of PHA4GE fields as part of a BioSample record. BioSamples are propagated throughout INSDC databases.

submissions to different repositories is provided below (Table 7.4).

7.3 Conclusion

The collective response to the SARS-CoV-2 pandemic has resulted in an unprecedented deployment of genomic surveillance worldwide, bringing together public health agencies, academic research institutions, and industry partners. This unified action provides opportunities to more effectively understand and respond to the pandemic. Yet it also provides an enormous challenge, as realising the full potential of this opportunity will require standardisation and harmonization of data collection across these partners. With our SARS-CoV-2 metadata specification we have endeavoured to create a mechanism for promoting consistent, standardised contextual data collection that can be applied broadly. We envision that given the increased uptake, this specification will improve the consistency of collected data, making information reusable by agencies as they continue working towards an increased understanding of SARS-CoV-2 epidemiology and biology, and harmonising them such that community-based data sharing efforts are not excessively burdened. We anticipate that the

Table 7.4: A selection of accession numbers of harmonised contextual data records submitted to different public repositories.

Data Contributor	Repository Name	Accession Number
African Centre of Excellence for Genomics of Infectious Diseases (Nigeria)	GISAID	EPI_ISL_1035827 EPI_ISL_1035826 EPI_ISL_1035825
COVID-19 Genomic Surveillance Regional Network (Latin America)	GISAID	EPI_ISL_2158821 EPI_ISL_2158802 EPI_ISL_2158810
COVID-19 Genomic Surveillance Regional Network (Latin America)	EMBL-EBI	SAMEA8968916
Rhode Island Department of Health/Broad Institute (SPHERES)	NCBI	SAMN18306978
Massachusetts General Hospital/Broad Institute (SPHERES)	NCBI	SAMN18309294
Flow Health/Broad Institute (SPHERES)	NCBI	SAMN18308763
New Brunswick Diagnostic Virology Reference Center/Public Health Agency of Canada (CanCOGeN)	NCBI	SAMN16784832
Toronto Invasive Bacterial Diseases Network/McMaster University (CanCOGeN)	NCBI	SAMN17505317
Bat coronavirus phylogeography- Université de La Réunion, UMR Processus Infectieux en Milieu Insulaire Tropical (PIMIT) and Field Museum of Natural History	NCBI	SAMN20400589 SAMN20400588

experience and lessons learned creating the specification package for SARS-CoV-2 will better enable the rapid development and deployment of pathogen-specific standards for public health pathogen genomic surveillance in the future.

7.4 Methods

The PHA4GE SARS-CoV-2 data specification was developed by first comparing existing metadata standards (e.g., MIxS/MIGS, the NIAID/BRC Sample Application Standard), various sequence repository submission requirements (e.g., GISAID, INSDC), as well as national and international case report forms.

A gap analysis was performed to identify SARS-CoV-2 public health surveillance data elements that were missing in available standards. Fields in existing standards that were deemed to be out of scope were excluded from the specification. Terms for pick lists were sourced from public health documents, the literature, and when available, various interoperable ontologies (OBO Foundry). The fields and terms from the gap analysis were structured in the collection template (.xlsx). Field definitions, guidance for use, examples and mappings to various standards were developed as part of the Reference Guides provided in separate tabs in the template workbook. Vocabulary lists were also provided in a separate tab in the template workbook to enable validation, and to enable users to add terms to pick lists as needed, according to instructions provided in the curation SOP. The specification was also encoded as a JavaScript Object Notation (JSON) file.

The specification was reviewed by public health, bioinformaticians and data standards experts from different public health agencies, research institutes and sequencing consortia and adapted according to feedback. Upon request by community members, versioned protocols for public repository submission were created and deposited in protocols.io.

The first version of the specification was made publicly available in August 2020 with a CC-BY 4.0 International attribution license. Iterative improvements were made to a development branch of the specification over the next 10 months as the pandemic evolved, and in response to user feedback and requests. The second major release (2.0) was made publicly available in May 2021. A third major release (3.0) including ontology mappings and the term-level reference guide was made publicly available in December 2021. The PHA4GE template was incorporated into the contextual data harmonization, validation and transformation tool called The DataHarmonizer through a collaborative effort with the Hsiao Public Health Bioinformatics Laboratory (Simon Fraser University). Details regarding DataHarmonizer development can be found elsewhere (manuscript in preparation).

7.5 Availability and Requirements

The software used in this study is available on GitHub.

Project name: SARS-CoV-2-Contextual-Data-Specification

Project home page: https://github.com/pha4ge/SARS-CoV-2-Contextual-Data-Specification

Operating system: Platform independent

Programming language: Not applicable

Other requirements: xlsx-compatible spreadsheet software

License: CC-BY 4.0 International

RRID: SCR_021378

biotools: pha4ge_sars-cov-2_contextual_data_specification

7.6 Declarations

7.6.1 Ethics approval and consent to participate

Not applicable.

7.6.2 Consent for publication

Not applicable.

7.6.3 Competing interests

The authors declare that they have no competing interests.

7.7 Funding

We wish to thank the Bill Melinda Gates Foundation for supporting the establishment and work of the PHA4GE consortium. AJP and NFA gratefully acknowledge the support of the Biotechnology and Biological Sciences Research Council (BBSRC); and were supported by the Quadram Institute Bioscience BBSRC funded Core Capability Grant (project number BB/CCG1860/1) and the BBSRC Institute Strategic Programme Microbes in the Food Chain BB/R012504/1 and its constituent project BBS/E/F/000PR10352. FM was supported by a Donald Hill Family Fellowship in Computer Science. CIM was supported by the Fundação para a Ciência e Tecnologia (grant SFRH/BD/129483/2017). Work by EJG, RC, DD, and WWLH was funded by a Genome Canada Bioinformatics and Computational Biology 2017 Grant 286GET and a Genome Canada CanCOGeN grant E09CMA. The work of IKM, TB, and AJ was supported by the Intramural Research Program of the National Library of Medicine, National Institutes of Health.

7.8 Authors' contributions

EJG: Conceptualization, Methodology, Investigation, Software, Visualization, Writing - Original Draft Preparation, Validation, Supervision; RET: Methodology, Investigation, Software, Validation, Writing - Original Draft Preparation; CIM: Methodology, Software, Writing - Review & Editing; AJP: Methodology, Writing - Original Draft Preparation; NFA:

Methodology, Software, Validation, Writing - Original Draft Preparation; DF: Methodology, Software; JC: Validation, Writing - Review & Editing; DP: Validation, Writing - Review & Editing; IDB: Validation, Writing - Review Editing; DA: Software, Validation, Writing - Review & Editing; AC: Writing - Review & Editing; AGS: Software, Validation, Writing - Review & Editing; RC: Software, Validation; DD: Software, Validation; LSK: Validation, Writing - Review & Editing; AB: Methodology, Writing - Original Draft Preparation; IKM: Software, Validation, Writing - Review & Editing; TB: Software, Validation, Writing - Review & Editing; AJ: Software, Validation, Writing - Review & Editing; TRC: Validation, Writing - Review & Editing; SMN: Validation, Writing - Review & Editing; AAW: Writing - Review & Editing; PEO: Writing - Review & Editing; GHT: Writing - Review & Editing; SHT: Writing - Review & Editing; ARR: Writing - Review & Editing; BA: Writing - Review & Editing; DAM: Writing - Review & Editing; EH: Writing - Review & Editing; WWLH: Writing - Review & Editing; ATRV: Writing - Review & Editing; DRM: Conceptualization, Methodology, Visualization, Writing - Review & Editing, Funding Acquisition.

7.9 Acknowledgements

The authors would like to thank the US Center for Disease Control and Prevention's Technical Outreach and Assistance for States Team (TOAST) for their feedback, support, and assistance in disseminating the PHA4GE specification package among US public health networks.

7.10 References

[1] World Health Organization. *Coronavirus disease (COVID-19) – World Health Organization*. en. URL: `https://www.who.int/emergencies/diseases/novel-coronavirus-2019` (visited on 04/22/2022).

[2] Ensheng Dong, Hongru Du, and Lauren Gardner. “An interactive web-based dashboard to track COVID-19 in real time”. eng. In: *The Lancet. Infectious Diseases* 20.5 (May 2020), pp. 533–534. ISSN: 1474-4457. DOI: `10.1016/S1473-3099(20)30120-1`.

[3] COVID-19 Genomics UK (COG-UK) consortiumcontact@cogconsortium.uk. “An integrated national scale SARS-CoV-2 genomic surveillance network”. eng. In: *The Lancet. Microbe* 1.3 (July 2020), e99–e100. ISSN: 2666-5247. DOI: `10.1016/S2666-5247(20)30054-9`.

[4] CDC. *Cases, Data, and Surveillance*. en-us. Feb. 2020. URL: `https://www.cdc.gov/coronavirus/2019-ncov/variants/spheres.html` (visited on 04/22/2022).

[5] CanCOGeN (Genome Canada). *CanCOGeN | Genome Canada*. URL: `https://www.genomecanada.ca/en/cancogen` (visited on 04/22/2022).

[6] Pan American Health Organization and World Health Organization. *Laboratory Guidelines for the Detection and Diagnosis of COVID-19 Virus Infection - PAHO/WHO | Pan American Health Organization*. en. URL: `https://www.paho.org/en/documents/laboratory-guidelines-detection-and-diagnosis-covid-19-virus-infection` (visited on 04/22/2022).

[7] Darlan S. Candido et al. “Evolution and epidemic spread of SARS-CoV-2 in Brazil”. eng. In: *Science (New York, N.Y.)* 369.6508 (Sept. 2020), pp. 1255–1260. ISSN: 1095-9203. DOI: `10.1126/science.abd2161`.

[8] Wen-Ming Zhao et al. “The 2019 novel coronavirus resource”. eng. In: *Yi Chuan = Hereditas* 42.2 (Feb. 2020), pp. 212–221. ISSN: 0253-9772. DOI: `10.16288/j.yczz.20-030`.

[9] Network for Genomic Surveillance South Africa. *NGS-SA: Network for Genomic Surveillance South Africa*. URL: `http://www.krisp.org.za/ngs-sa/ngs-sa_network_for_genomic_surveillance_south_africa/` (visited on 04/22/2022).

[10] Communicable Diseases Genomics Network. *AusTrakka*. en-US. URL: `https://www.cdgn.org.au/austrakka` (visited on 04/22/2022).

[11] Government of India. *Indian SARS-CoV-2 Genomic Consortia (INSACOG)*. URL: `https://dbtindia.gov.in/index.php` (visited on 04/22/2022).

[12] Yuelong Shu and John McCauley. “GISAID: Global initiative on sharing all influenza data - from vision to reality”. eng. In: *Euro Surveillance: Bulletin Europeen Sur Les Maladies Transmissibles = European Communicable Disease Bulletin* 22.13 (Mar. 2017), p. 30494. ISSN: 1560-7917. DOI: `10.2807/1560-7917.ES.2017.22.13.30494`.

[13] Ilene Karsch-Mizrachi et al. “The international nucleotide sequence database collaboration”. eng. In: *Nucleic Acids Research* 46.D1 (Jan. 2018), pp. D48–D51. ISSN: 1362-4962. DOI: `10.1093/nar/gkx1097`.

[14] Marc W. Allard et al. “Practical Value of Food Pathogen Traceability through Building a Whole-Genome Sequencing Network and Database”. eng. In: *Journal of Clinical Microbiology* 54.8 (Aug. 2016), pp. 1975–1983. ISSN: 1098-660X. DOI: `10.1128/JCM.00081-16`.

[15] Kristy A. Kubota et al. “PulseNet and the Changing Paradigm of Laboratory-Based Surveillance for Foodborne Diseases”. eng. In: *Public Health Reports (Washington, D.C.: 1974)* 134.2_suppl (Dec. 2019), 22S–28S. ISSN: 1468-2877. DOI: `10.1177/0033354919881650`.

[16] Joseph A. Cook et al. “Integrating Biodiversity Infrastructure into Pathogen Discovery and Mitigation of Emerging Infectious Diseases”. eng. In: *Bioscience* 70.6 (July 2020), pp. 531–534. ISSN: 0006-3568. DOI: `10.1093/biosci/biaa064`.

[17] Kristian G. Andersen et al. “The proximal origin of SARS-CoV-2”. en. In: *Nature Medicine* 26.4 (Apr. 2020). Number: 4 Publisher: Nature Publishing Group, pp. 450–452. ISSN: 1546-170X. DOI: `10.1038/s41591-020-0820-9`. URL: `https://www.nature.com/articles/s41591-020-0820-9` (visited on 04/26/2022).

[18] Ravindra K. Gupta. “Will SARS-CoV-2 variants of concern affect the promise of vaccines?” en. In: *Nature Reviews Immunology* 21.6 (June 2021). Number: 6 Publisher: Nature Publishing Group, pp. 340–341. ISSN: 1474-1741. DOI: `10.1038/s41577-021-00556-5`. URL: `https://www.nature.com/articles/s41577-021-00556-5` (visited on 04/26/2022).

[19] Public Health England. *SARS-CoV-2 variants of concern and variants under investigation - Technical briefing 16*. en.

[20] Los Alamos National Laboratory. *In silico evaluation of diagnostic assays*. URL: `https://covid19.edgebioinformatics.org/#/assayValidation` (visited on 04/26/2022).

[21] Kevin S. Kuchinski et al. “Mutations in emerging variant of concern lineages disrupt genomic sequencing of SARS-CoV-2 clinical specimens”. en. In: *International Journal of Infectious Diseases* 114 (Jan. 2022), pp. 51–54. ISSN: 12019712. DOI: `10.1016/j.ijid.2021.10.050`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S1201971221008389` (visited on 04/26/2022).

[22] Anurup Ganguli et al. “Rapid isothermal amplification and portable detection system for SARS-CoV-2”. en. In: *Proceedings of the National Academy of Sciences* 117.37 (Sept. 2020), pp. 22727–22735. ISSN: 0027-8424, 1091-6490. DOI: `10.1073/pnas.2014739117`. URL: `https://pnas.org/doi/full/10.1073/pnas.2014739117` (visited on 04/26/2022).

[23] World Health Organization. *COVID-19 vaccine tracker and landscape*. en. URL: `https://www.who.int/publications/m/item/draft-landscape-of-covid-19-candidate-vaccines` (visited on 04/26/2022).

[24] Richard L. Tillett et al. “Genomic evidence for reinfection with SARS-CoV-2: a case study”. English. In: *The Lancet Infectious Diseases* 21.1 (Jan. 2021). Publisher: Elsevier, pp. 52–58. ISSN: 1473-3099, 1474-4457. DOI: `10.1016/S1473-3099(20)30764-7`. URL: `https://www.thelancet.com/journals/laninf/article/PIIS1473-3099(20)30764-7/fulltext` (visited on 04/26/2022).

[25] Bas B. Oude Munnink et al. “Transmission of SARS-CoV-2 on mink farms between humans and mink and back to humans”. eng. In: *Science (New York, N.Y.)* 371.6525 (Jan. 2021), pp. 172–177. ISSN: 1095-9203. DOI: `10.1126/science.abe5901`.

[26] Chih-Cheng Lai et al. “COVID-19 in long-term care facilities: An upcoming threat that cannot be ignored”. In: *Journal of Microbiology, Immunology, and Infection* 53.3 (June 2020), pp. 444–446. ISSN: 1684-1182. DOI: `10.1016/j.jmii.2020.04.008`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7153522/` (visited on 04/26/2022).

[27] Dinesh Aggarwal et al. *The role of genomics in understanding COVID-19 outbreaks in long term care facilities*. preprint. Open Science Framework, Nov. 2020. DOI: `10.31219/osf.io/7y9rk`. URL: `https://osf.io/7y9rk` (visited on 04/26/2022).

[28] Michelle Murti et al. “Investigation of a severe SARS-CoV-2 outbreak in a long-term care home early in the pandemic”. en. In: *CMAJ* 193.19 (May 2021). Publisher: CMAJ Section: Research, E681–E688. ISSN: 0820-3946, 1488-2329. DOI: `10.1503/cmaj.202485`. URL: `https://www.cmaj.ca/content/193/19/E681` (visited on 04/26/2022).

[29] Jonathan W. Dyal et al. “COVID-19 Among Workers in Meat and Poultry Processing Facilities — 19 States, April 2020”. In: *MMWR. Morbidity and Mortality Weekly Report* 69.18 (May 2020). ISSN: 0149-2195, 1545-861X. DOI: `10.15585/mmwr.mm6918e3`. URL: `http://www.cdc.gov/mmwr/volumes/69/wr/mm6918e3.htm?s_cid=mm6918e3_w` (visited on 04/26/2022).

[30] Thomas Günther et al. “SARS-CoV-2 outbreak investigation in a German meat processing plant”. en. In: *EMBO Molecular Medicine* 12.12 (Dec. 2020). ISSN: 1757-4676, 1757-4684. DOI: `10.15252/emmm.202013296`. URL: `https://onlinelibrary.wiley.com/doi/10.15252/emmm.202013296` (visited on 04/26/2022).

[31] Joanne Taylor et al. “Serial Testing for SARS-CoV-2 and Virus Whole Genome Sequencing Inform Infection Risk at Two Skilled Nursing Facilities with COVID-19 Outbreaks - Minnesota, April-June 2020”. eng. In: *MMWR. Morbidity and mortality weekly report* 69.37 (Sept. 2020), pp. 1288–1295. ISSN: 1545-861X. DOI: `10.15585/mmwr.mm6937a3`.

[32] Daniela Loconsole et al. “Investigation of an outbreak of symptomatic SARS-CoV-2 VOC 202012/01-lineage B.1.1.7 infection in healthcare workers, Italy”. en. In: *Clinical Microbiology and Infection* 27.8 (Aug. 2021), 1174.e1–1174.e4. ISSN: 1198743X. DOI: `10.1016/j.cmi.2021.05.007`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S1198743X21002287` (visited on 04/26/2022).

[33] Dan Frampton et al. “Genomic characteristics and clinical effect of the emergent SARS-CoV-2 B.1.1.7 lineage in London, UK: a whole-genome sequencing and hospital-based cohort study”. en. In: *The Lancet Infectious Diseases* 21.9 (Sept. 2021), pp. 1246–1256. ISSN: 14733099. DOI: `10.1016/S1473-3099(21)00170-5`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S1473309921001705` (visited on 04/26/2022).

[34] Ana da Silva Filipe et al. “Genomic epidemiology reveals multiple introductions of SARS-CoV-2 from mainland Europe into Scotland”. eng. In: *Nature Microbiology* 6.1 (Jan. 2021), pp. 112–122. ISSN: 2058-5276. DOI: `10.1038/s41564-020-00838-z`.

[35] Bas B. Oude Munnink et al. “Rapid SARS-CoV-2 whole-genome sequencing and analysis for informed public health decision-making in the Netherlands”. en. In: *Nature Medicine* 26.9 (Sept. 2020). Number: 9 Publisher: Nature Publishing Group, pp. 1405–1410. ISSN: 1546-170X. DOI: `10.1038/s41591-020-0997-y`. URL: `https://www.nature.com/articles/s41591-020-0997-y` (visited on 04/26/2022).

[36] Joseph R. Fauver et al. “Coast-to-Coast Spread of SARS-CoV-2 during the Early Epidemic in the United States”. eng. In: *Cell* 181.5 (May 2020), 990–996.e5. ISSN: 1097-4172. DOI: `10.1016/j.cell.2020.04.021`.

[37] Edward S. Knock et al. “Key epidemiological drivers and impact of interventions in the 2020 SARS-CoV-2 epidemic in England”. eng. In: *Science Translational Medicine* 13.602 (July 2021), eabg4262. ISSN: 1946-6242. DOI: `10.1126/scitranslmed.abg4262`.

[38] Courtney R. Lane et al. “Genomics-informed responses in the elimination of COVID-19 in Victoria, Australia: an observational, genomic epidemiological study”. eng. In: *The Lancet. Public Health* 6.8 (Aug. 2021), e547–e556. ISSN: 2468-2667. DOI: `10.1016/S2468-2667(21)00133-X`.

[39] Louis du Plessis et al. “Establishment and lineage dynamics of the SARS-CoV-2 epidemic in the UK”. eng. In: *Science (New York, N.Y.)* 371.6530 (Feb. 2021), pp. 708–712. ISSN: 1095-9203. DOI: `10.1126/science.abf2946`.

[40] George Githinji et al. *Tracking the introduction and spread of SARS-CoV-2 in coastal Kenya*. en. Tech. rep. Type: article. medRxiv, Oct. 2020, p. 2020.10.05.20206730. DOI: `10.1101/2020.10.05.20206730`. URL: `https://www.medrxiv.org/content/10.1101/2020.10.05.20206730v1` (visited on 04/27/2022).

[41] Luke W. Meredith et al. "Rapid implementation of SARS-CoV-2 sequencing to investigate cases of health-care associated COVID-19: a prospective genomic surveillance study". eng. In: *The Lancet. Infectious Diseases* 20.11 (Nov. 2020), pp. 1263–1272. ISSN: 1474-4457. DOI: `10.1016/S1473-3099(20)30562-4`.

[42] Wenjuan Zhang et al. "Analysis of Genomic Characteristics and Transmission Routes of Patients With Confirmed SARS-CoV-2 in Southern California During the Early Stage of the US COVID-19 Pandemic". eng. In: *JAMA network open* 3.10 (Oct. 2020), e2024191. ISSN: 2574-3805. DOI: `10.1001/jamanetworkopen.2020.24191`.

[43] S. Wesley Long et al. "Molecular Architecture of Early Dissemination and Massive Second Wave of the SARS-CoV-2 Virus in a Major Metropolitan Area". eng. In: *mBio* 11.6 (Oct. 2020), e02707–20. ISSN: 2150-7511. DOI: `10.1128/mBio.02707-20`.

[44] Jemma L. Geoghegan et al. "Genomic epidemiology reveals transmission patterns and dynamics of SARS-CoV-2 in Aotearoa New Zealand". eng. In: *Nature Communications* 11.1 (Dec. 2020), p. 6351. ISSN: 2041-1723. DOI: `10.1038/s41467-020-20235-8`.

[45] Torsten Seemann et al. "Tracking the COVID-19 pandemic in Australia using genomics". eng. In: *Nature Communications* 11.1 (Sept. 2020), p. 4376. ISSN: 2041-1723. DOI: `10.1038/s41467-020-18314-x`.

[46] Angela McLaughlin et al. *Early and ongoing importations of SARS-CoV-2 in Canada*. en. Tech. rep. Type: article. medRxiv, Apr. 2021, p. 2021.04.09.21255131. DOI: `10.1101/2021.04.09.21255131`. URL: `https://www.medrxiv.org/content/10.1101/2021.04.09.21255131v1` (visited on 04/27/2022).

[47] Nicola De Maio et al. *Issues with SARS-CoV-2 sequencing data - SARS-CoV-2 coronavirus / nCoV-2019 Genomic Epidemiology*. en. May 2020. URL: `https://virological.org/t/issues-with-sars-cov-2-sequencing-data/473` (visited on 04/27/2022).

[48] Mikhail Rayko and Aleksey Komissarov. *Quality control of low-frequency variants in SARS-CoV-2 genomes*. en. preprint. Genomics, Apr. 2020. DOI: `10.1101/2020.04.26.062422`. URL: `http://biorxiv.org/lookup/doi/10.1101/2020.04.26.062422` (visited on 04/27/2022).

[49] Leo L. M. Poon et al. "Recurrent mutations associated with isolation and passage of SARS coronavirus in cells from non-human primates". eng. In: *Journal of Medical Virology* 76.4 (Aug. 2005), pp. 435–440. ISSN: 0146-6615. DOI: `10.1002/jmv.20379`.

[50] Pelin Yilmaz et al. "Minimum information about a marker gene sequence (MIMARKS) and minimum information about any (x) sequence (MIxS) specifications". eng. In: *Nature Biotechnology* 29.5 (May 2011), pp. 415–420. ISSN: 1546-1696. DOI: `10.1038/nbt.1823`.

[51] Dawn Field et al. "The minimum information about a genome sequence (MIGS) specification". eng. In: *Nature Biotechnology* 26.5 (May 2008), pp. 541–547. ISSN: 1546-1696. DOI: `10.1038/nbt1360`.

[52] Vivien G. Dugan et al. "Standardized metadata for human pathogen/vector genomic sequences". eng. In: *PloS One* 9.6 (2014), e99979. ISSN: 1932-6203. DOI: `10.1371/journal.pone.0099979`.

[53] Barry Smith et al. "The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration". eng. In: *Nature Biotechnology* 25.11 (Nov. 2007), pp. 1251–1255. ISSN: 1087-0156. DOI: `10.1038/nbt1346`.

[54] Lynn M. Schriml et al. "COVID-19 pandemic reveals the peril of ignoring metadata standards". eng. In: *Scientific Data* 7.1 (June 2020), p. 188. ISSN: 2052-4463. DOI: `10.1038/s41597-020-0524-5`.

[55] Public Health Alliance for Genomic Epidemiology. *SARS-CoV-2-Contextual-Data-Specification: Collection template and associated materials for SARS-CoV-2 metadata*. URL: `https://github.com/pha4ge/SARS-CoV-2-Contextual-Data-Specification` (visited on 04/27/2022).

[56] The OBO Foundry. *The OBO Foundry*. URL: `http://www.obofoundry.org/` (visited on 04/26/2022).

[57] Public Health Alliance for Genomic Epidemiology. *PHA4GE - research workspace on protocols.io*. en. URL: `https://www.protocols.io/workspaces/pha4ge` (visited on 04/27/2022).

[58] World Health Organization. *Guidance for surveillance of SARS-CoV-2 variants: Interim guidance, 9 August 2021*. en. URL: `https://www.who.int/publications-detail-redirect/WHO_2019-nCoV_surveillance_variants` (visited on 04/27/2022).

[59] Hsiao Public Health Bioinformatics Lab. *DataHarmonizer*. original-date: 2020-05-17T16:17:49Z. Apr. 2022. URL: `https://github.com/cidgoh/DataHarmonizer` (visited on 04/27/2022).

[60] *METAGENOTE: Home*. URL: `https://metagenote.niaid.nih.gov/` (visited on 04/27/2022).

[61] Maximilian Haeussler. *multiSub*. original-date: 2021-04-02T15:17:30Z. Apr. 2022. URL: `https://github.com/maximilianh/multiSub` (visited on 04/27/2022).

[62] *ena-content-dataflow/scripts/gisaid_to_ena at master · enasequence/ena-content-dataflow*. en. URL: `https://github.com/enasequence/ena-content-dataflow` (visited on 04/27/2022).

[63] *GET Africa – ONE AFRICA, ONE HEALTH, ONE DESTINY*. en. URL: `https://www.getafrica.org/` (visited on 04/27/2022).

[64] *Acegid – Welcome Online*. en-US. URL: `https://acegid.org/` (visited on 04/27/2022).

[65] *Welcome to Baobab LIMS*. en-US. URL: `https://baobablims.org/` (visited on 04/27/2022).

[66] *SANBI – South African National Bioinformatics Institute – The South African National Bioinformatics Institute delivers biomedical discovery appropriate to both international and African context. Researchers at SANBI perform the highest level of research and provide excellence in education.* en-US. URL: `https://www.sanbi.ac.za/` (visited on 04/27/2022).

[67] *COVID-19 Genomic Surveillance Regional Network - PAHO/WHO | Pan American Health Organization*. URL: `https://www.paho.org/en/topics/influenza-and-other-respiratory-viruses/covid-19-genomic-surveillance-regional-network` (visited on 04/27/2022).

[68] Tanya Barrett et al. “BioProject and BioSample databases at NCBI: facilitating capture and organization of metadata”. eng. In: *Nucleic Acids Research* 40.Database issue (Jan. 2012), pp. D57–63. ISSN: 1362-4962. DOI: `10.1093/nar/gkr1163`.

[69] *Home - Pathogen Detection - NCBI*. URL: `https://www.ncbi.nlm.nih.gov/pathogens/` (visited on 04/27/2022).

[70] Compare. *Home - Compare Europe*. en. URL: `https://www.compare-europe.eu/` (visited on 04/27/2022).

[71] NCBI Staff. *A dedicated SARS-CoV-2 BioSample submission package in the NCBI Submission Portal*. en-US. May 2021. URL: `https://ncbiinsights.ncbi.nlm.nih.gov/2021/05/11/sars-cov-2-biosample-submission-package/` (visited on 04/27/2022).

Part IV

Crowdsourcing for software robustness in metagenomics and beyond

Chapter 8

Software testing in microbial bioinformatics: a call to action

This chapter is a reproduction of the following publication:

B. C. L. van der Putten, C. I. Mendes, B. M. Talbot, J. de Korne-Elenbaas, R. Mamede, P. Vila-Cerqueira, L. P. Coelho, C. A. Gulvik, L. S. Katz and The ASM NGS 2020 Hackathon participants. Software testing in microbial bioinformatics: a call to action. Microbial Genomics, Volume 8, Issue 3, March 2022. DOI: `https://doi.org/10.1099/mgen.0.000790`

Computational algorithms have become an essential component of microbiome research, with great efforts by the scientific community to raise standards on the development and distribution of code. Despite these efforts, sustainability and reproducibility are major issues since continued validation through software testing is still not a widely adopted practice.

In the field of microbial bioinformatics, good software engineering practices are not yet widely adopted. Many microbial bioinformaticians start out as (micro)biologists and subsequently learn how to code. Without abundant formal training, a lot of education about good software engineering practices comes down to an exchange of information within the microbial bioinformatics community.

Here, we report seven recommendations that help researchers implement software testing in microbial bioinformatics. These recommendations are: Establish software needs and testing goals; Use appropriate input test files; Use an easy-to-follow language format to implement testing; Try to automate testing; Test across multiple computational setups; and Encourage others to test your software.

We propose collaborative software testing as an opportunity to continuously engage software users, developers, and students to unify scientific work across domains. As automated software testing remains underused in scientific software, our set of recommendations not only ensures appropriate effort can be invested into producing high-quality and robust software but also increases engagement in its sustainability.

We have developed these recommendations based on our experience from a collaborative hackathon organised prior to the American Society for Microbiology Next Generation Sequencing (ASM NGS) 2020 conference. We also present a repository hosting examples and guidelines for testing, available from `https://github.com/microbinfie-hackathon2020/CSIS`.

My contribution to this publication included the development of the seven recommendations here presented, including examples of software testing. Additionally, I have also contributed to the manuscript production and editing.

Software testing in microbial bioinformatics: a call to action

Boas C. L. van der Putten[1,2,*], Catarina I. Mendes[3,*], Brooke M. Talbot[4], Jolinda de Korne-Elenbaas[1,5], Rafael Mamede[3], Pedro Vila-Cerqueira[3], Luis P. Coelho[6,7], Christopher A. Gulvik[8], Lee S. Katz[9,10] and The ASM NGS 2020 Hackathon participants

[1]Department of Medical Microbiology, Amsterdam UMC, University of Amsterdam, the Netherlands

[2]Department of Global Health, Amsterdam Institute for Global Health and Development, Amsterdam UMC, University of Amsterdam, the Netherlandss

[3]Instituto de Microbiologia, Instituto de Medicina Molecular, Faculdade de Medicina, Universidade de Lisboa, Lisboa, Portugal

[4]Department of Biological and Biomedical Sciences, Emory University, Atlanta, GA, USA

[5]Department of Infectious Diseases, Public Health Laboratory, Public Health Service of Amsterdam, the Netherlands

[6]Institute of Science and Technology for Brain-Inspired Intelligence, Fudan University, PR China

[7]Key Laboratory of Computational Neuroscience and Brain-Inspired Intelligence, PR China

[8]Bacterial Special Pathogens Branch, Division of High-Consequence Pathogens and Pathology, Centers for Disease Control and Prevention, Atlanta, GA, USA

[9]Center for Food Safety, University of Georgia, Griffin, GA, USA

[10]Enteric Diseases Laboratory Branch, Division of Foodborne, Waterborne, and Environmental Diseases, Centers for Disease Control and Prevention, Atlanta, GA, USA

[*]Contributed equally

8.1 Abstract

Computational algorithms have become an essential component of research, with great efforts by the scientific community to raise standards on the development and distribution of code. Despite these efforts, sustainability and reproducibility are major issues since continued validation through software testing is still not a widely adopted practice. Here, we

report seven recommendations that help researchers implement software testing in microbial bioinformatics. We have developed these recommendations based on our experience from a collaborative hackathon organised prior to the American Society for Microbiology Next Generation Sequencing (ASM NGS) 2020 conference. We also present a repository hosting examples and guidelines for testing, available from `https://github.com/microbinfie-hackathon2020/CSIS`.

8.2 Impact Statement

In the field of microbial bioinformatics, good software engineering practices are not yet widely adopted. Many microbial bioinformaticians start out as (micro)biologists and subsequently learn how to code. Without abundant formal training, a lot of education about good software engineering practices comes down to an exchange of information within the microbial bioinformatics community. This paper serves as a resource that could help microbial bioinformaticians get started with software testing if they have not had formal training.

8.3 Background

Computational algorithms, software, and workflows have enhanced the breadth and depth of microbiological research and expanded the capacity of infectious disease surveillance in public health practice. Scientists now have a wealth of bioinformatic tools for addressing pertinent questions quickly and keeping pace with the availability of larger and more complex biological datasets. Despite these advances, we are finding ourselves in a crisis of computational reproducibility [1].

Modern software engineering advocates reliable software testing standards and best practices. Different approaches are employed: from unit testing to system testing [2], going from testing every individual component to testing a tool as a whole (Fig. 8.1). The extent of testing is a balance between the resources available and increasing sustainability and reproducibility. Continuous Integration (CI), where code changes are frequently integrated and assertion of the new code's correctness before integration is often automatedly performed through tests, provides a robust approach for ensuring the reproducibility of scientific results without requiring human interaction. Comprehensive testing of scientific software might prevent computational errors which subsequently lead to erroneous results and retractions [3, 4]. However, the role of testing extends beyond that, as it also provides a way to measure software coverage, and therefore its robustness, allowing for reported issues to be converted into testable actions (regression tests), and the expansion and refactoring of existing code without compromising its function.

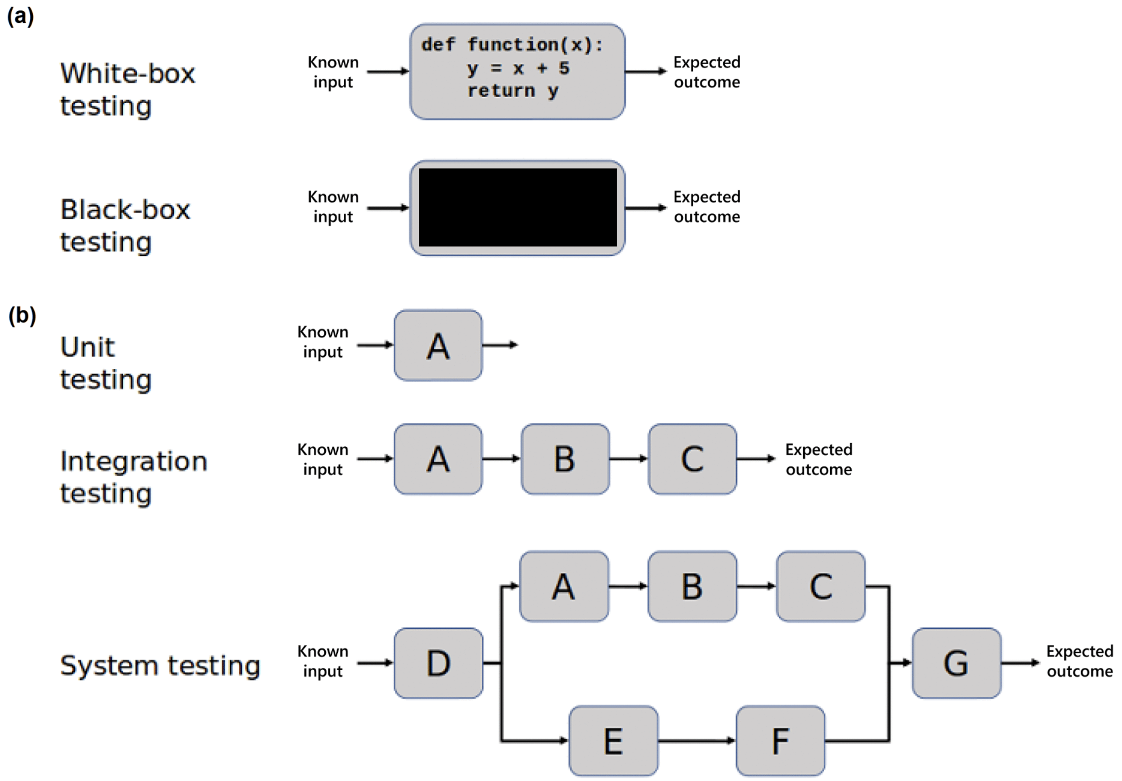


Figure 8.1: Testing strategies. (a) White-box vs. black-box testing. In white-box testing, the tester knows the underlying code and structure of the software, where the tester does not know this in black-box testing. Note that this distinction is not strictly dichotomous and is considered less useful nowadays (b) Unit vs. integration vs. system testing. When software comprises several modules, it is possible to test each single module (unit testing), groups of related modules (integration testing) or all modules (system testing). Note that the terms white-box testing and unit testing are sometimes used interchangeably but relate to different concepts

Software testing among peers across fields aligns with previous efforts of hackathons to create a more unified and informed bioinformatics software community [5]. In this context, we hosted a cooperative hackathon prior to the ASM NGS conference in 2020, demonstrating that the microbial bioinformatics community can contribute to software sustainability using a collaborative platform (Table **??**). From this experience, we would like to propose collaborative software testing as an opportunity to continuously engage software users, developers, and students to unify scientific work across domains. We have outlined the following recommendations for ensuring software sustainability through testing and offer a repository of automated test knowledge and examples at the Code Safety Inspection Service (CSIS) repository on GitHub (`https://github.com/microbinfie-hackathon2020/CSIS`).

8.4 Recommendations

Based on our experiences from the ASM NGS 2020 hackathon, we developed seven recommendations that can be followed during software development.

8.4.1 Establish software needs and testing goals

Manually testing the functionality of a tool is feasible in early development but can become laborious as the software matures. Developers may establish software needs and testing goals during the planning and designing stages to ensure an efficient testing structure. Table 8.1 provides an overview of testing methodologies and can serve as a guide to developers that aim to implement testing practises. A minimal test set could address the validation of core components or the programme as a whole (system testing) and gradually progress toward verification of key functions which can accommodate code changes over time (unit testing, Figure 8.1). Ideally, testing should be implemented from the early stages of software development (test-driven development). Defining the scope of testing is important before developing tests. For pipeline development, testing of each individual component can be laborious and can be expedited if those components already implement testing of their own. Testing of the pipeline itself should take priority.

8.4.2 Input test files: the good, the bad, and the ugly

When testing, it is important to include test files with known expected outcomes for a successful run. However, it is equally important to include files or other inputs on which the tool is expected to fail. For example, some tools should recognize and report an empty input file or a wrong input format. Therefore, the test dataset should be small enough to be easily deployed (see recommendation 4) but as large as necessary to cover all intended test cases.

Table 8.1: Overview of testing approaches. Software testing can be separated into three types: installation, functionality and destructive. Each component is described, followed by an example on a real-life application on Software X, a hypothetical nucleotide sequence annotation tool

Name	**Description**	**Example**
	Installation testing: can the software be invoked on different setups?	
Installation testing	Can the software be installed on different platforms?	Test whether Software X can be installed using apt-get, pip, conda and from source.
Configuration testing	With which dependencies can the software be used?	Test whether Software X can be used with different versions of blast+.
Implementation testing	Do different implementations work similarly enough?	Test whether Software X works the same between the standalone and webserver versions.
Compatibility testing	Are newer versions compatible with previous input/output?	Test whether Software X can be used with older versions of the UniProtKB database.
Static testing	Is the source code syntactically correct?	Check whether all opening braces have corresponding closing braces or whether code is indented correctly in Software X.
	Standard functionality testing: does the software do what it should in daily use?	
Use case testing	Can the software do what it is supposed to do regularly?	Test whether Software X can annotate different FASTA files: with spaces in the header, without a header, an empty file, with spaces in the sequence, with unknown characters in the sequences, et cetera.
Workflow testing	Can the software successfully traverse each path in the analysis?	Test whether Software X works in different modes (using fast mode or using one dependency over the other).
Sanity testing	Can the software be invoked without errors?	Test whether Software X works correctly without flags, or when checking dependencies or displaying help info.
	Destructive testing: what makes the software fail?	
Mutation testing	How do the current tests handle harmful alterations to the software?	Test whether changing a single addition to a subtraction within Software X causes the test suite to fail.
Load testing	At what input size does the software fail?	Test whether Software X can annotate a small plasmid (10 kbp), a medium-size genome (2 Mbp) or an unrealistically large genome for a prokaryote (1 Gbp).
Fault injection	Does the software fail if faults are introduced and how is this handled?	Test whether Software X fails if nonsense functions are introduced in the gene calling code.

Data provenance should be disclosed, either if it's from real data or originated in silico. Typically, a small test data is packaged with the software. Examples of valid and invalid file formats are available through the BioJulia project (`https://github.com/BioJulia/BioFmtSpecimens`). The nf-core project (`https://nf-co.re/`) provides a repository with test data for a myriad of cases (`https://github.com/nf-core/test-datasets`).

8.4.3 Use an established framework to implement testing

Understanding the test workflow can not only ensure continued software development but also the integrity of the project for developers and users. Testing frameworks improve test development and efficiency. Examples include unittest (`https://docs.python.org/3/library/unittest.html`) or pytest (`https://docs.pytest.org/en/stable/`) for Python, and testthat (`https://testthat.r-lib.org/`) for R, testing interfaces such as TAP (`http://testanything.org/`), or built-in test attributes such as in Rust. Although many tests can be implemented using a combination of frameworks, personal preferences (e.g. amount of boilerplate code required) might drive your choice. Additionally, in Github Actions the formulas of each test block can be explicitly stated using the standardised and easy-to-follow YAML (`https://yaml.org/`, Fig. **??**, available in the online version of this article), already adopted by most continuous integration platforms (recommendation #4). For containerised software, testing considerations differ slightly and have been covered previously by Gruening et al. (2019) [6].

8.4.4 Testing is good, automated testing is better

When designing tests, planning for automation saves development time. Whether your tests are small or comprehensive, automatic triggering of tests will help reduce your workload. Many platforms trigger tests automatically based on a set of user-defined conditions. Platforms such as GitHub Actions (`https://github.com/features/actions`) and GitLab CI (`https://about.gitlab.com/stages-devops-lifecycle/continuous-integration`) offer straightforward automated testing of code seamlessly upon deployment. A typical workflow, consisting of a minimal testing framework (see recommendation #1 and #3) and a small test dataset (see recommendation #2), can then be directly integrated within your project hosted on a version control system, such as GitHub (`https://github.com/`), and directly integrated with a continuous integration provider, such as GitHub Actions in GitHub. Testing considerations for containerised software has been covered previously by Gruening et al. (2019) [6].

8.4.5 Ensure portability by testing on several platforms

The result of an automated test in the context of one computational workspace does not ensure the same result will be obtained in a different setup. It is important to ensure your software can be installed and used across supported platforms. One way to ensure this is to test on different environments, with varying dependency versions (e.g. multiple Python versions, instead of only the most recent one). Developers can gain increased benefits of testing if tests are run on different setups automatically (see recommendation #4 and Fig. 8.2).

8.4.6 Showcase the tests

For prospective users, it is good to know whether you have tested your software and, if so, which tests you have included. This can be done by displaying a badge [7] (see `https://github.com/microbinfie-hackathon2020/CSIS/blob/main/README.md#example-software-testing`), or linking to your defined testing strategy e.g. a GitHub Actions YAML, (see recommendation #2, Fig. 8.2). Documenting the testing goal and process enables end-users to easily check tool functionality and the level of testing [8].

It may be helpful to contact the authors, directly or through issues in the code repository, whose software you have tested to share successful outcomes or if you encountered abnormal behaviour or component failures. An external perspective can be useful to find bugs that the authors are unaware of. A set of issue templates for various situations is available in the CSIS repository on GitHub (`https://github.com/microbinfie-hackathon2020/CSIS/tree/main/templates`).

8.4.7 Encourage others to test your software

Software testing can be crowdsourced, as showcased by the ASM NGS 2020 hackathon. Software suites such as Pangolin (`https://github.com/cov-lineages/pangolin`) [9] and chewBBACA (`https://github.com/B-UMMI/chewBBACA`) [10] have implemented automated testing developed during the hackathon. For developers, crowdsourcing offers the benefits of fresh eyes on your software. Feedback and contributions from users can expedite the implementation of software testing practices. It also contributes to software sustainability by creating community buy-in, which ultimately helps the software maintainers keep pace with dependency changes and identify current user needs.

8.5 Conclusions

Testing is a critical aspect of scientific software development, but automated software testing remains underused in scientific software. In this hackathon, we demonstrated the usefulness of testing and developed a set of recommendations that should improve the development of tests. We also demonstrated the feasibility of producing test suites for already-established microbial bioinformatics software (Table S1).

8.6 Funding information

C.I.M. was supported by the Fundação para a Ciência e Tecnologia (grant SFRH/BD/129483/2017). L.P.C. was partially supported by Shanghai Municipal Science and Technology Major Project (2018SHZDZX01) and ZJLab. R. M. was supported by the Fundação para a Ciência e Tecnologia (grant 2020.08493.BD).

8.7 Acknowledgements

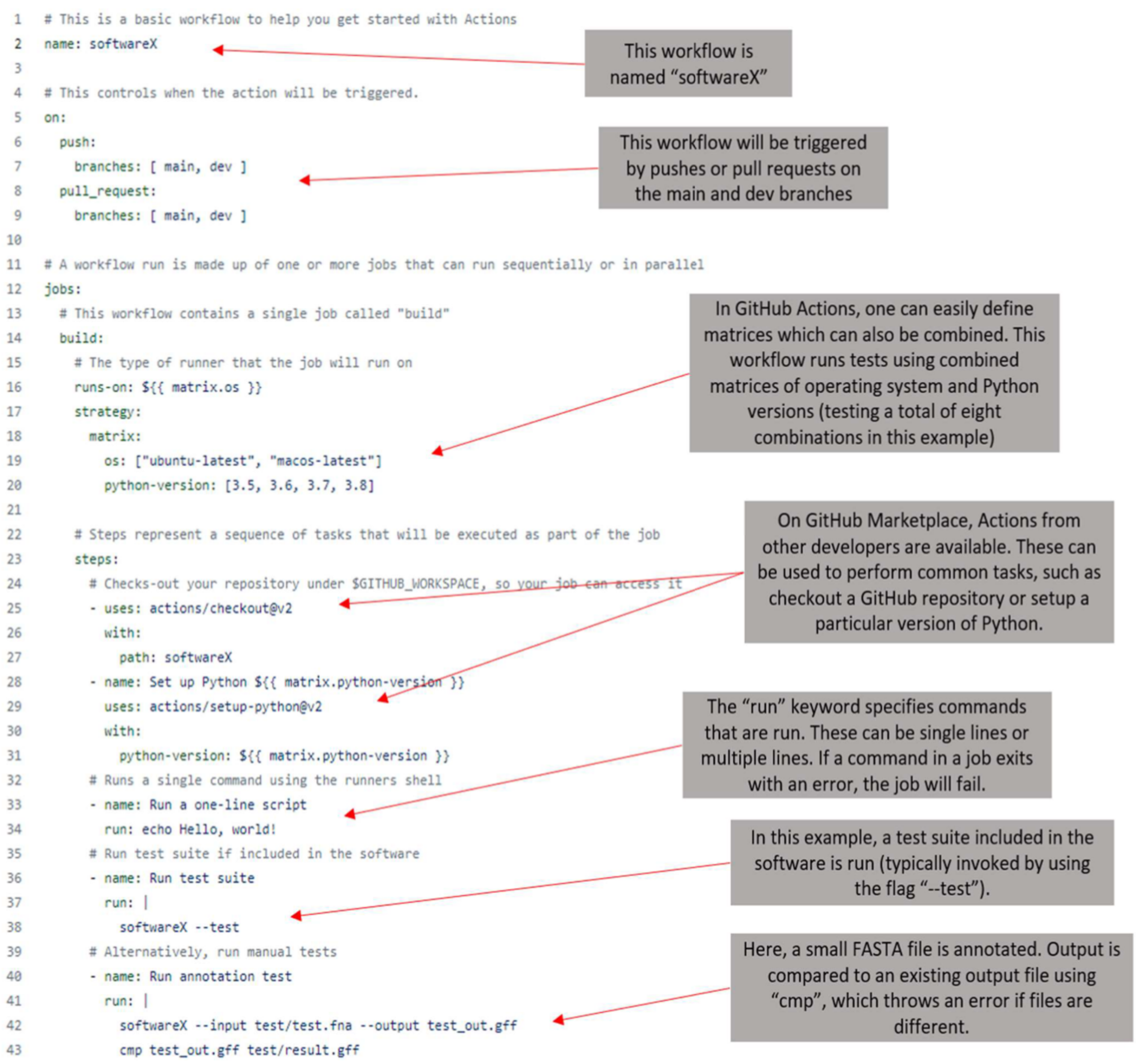
The findings and conclusions in this report are those of the authors and do not necessarily represent the official position of the Centers for Disease Control and Prevention (CDC). The mention of company names or products does not constitute an endorsement by the CDC.

8.8 Author contributions

In addition to the authors, the following participants were responsible for automating tests for bioinformatic tools and contributing a community resource for identifying software that can pass unit tests, available at `https://github.com/microbinfie-hackathon2020/CSIS`. Participants are listed alphabetically: Áine O'Toole, Amit Yadav, Justin Payne, Mario Ramirez, Peter van Heusden, Robert A. Petit III, Verity Hill, Yvette Unoarumhi.

8.9 Conflicts of interest

The authors declare that there are no conflicts of interest.



```yaml
# This is a basic workflow to help you get started with Actions
name: softwareX

# This controls when the action will be triggered.
on:
  push:
    branches: [ main, dev ]
  pull_request:
    branches: [ main, dev ]

# A workflow run is made up of one or more jobs that can run sequentially or in parallel
jobs:
  # This workflow contains a single job called "build"
  build:
    # The type of runner that the job will run on
    runs-on: ${{ matrix.os }}
    strategy:
      matrix:
        os: ["ubuntu-latest", "macos-latest"]
        python-version: [3.5, 3.6, 3.7, 3.8]

    # Steps represent a sequence of tasks that will be executed as part of the job
    steps:
      # Checks-out your repository under $GITHUB_WORKSPACE, so your job can access it
      - uses: actions/checkout@v2
        with:
          path: softwareX
      - name: Set up Python ${{ matrix.python-version }}
        uses: actions/setup-python@v2
        with:
          python-version: ${{ matrix.python-version }}
      # Runs a single command using the runners shell
      - name: Run a one-line script
        run: echo Hello, world!
      # Run test suite if included in the software
      - name: Run test suite
        run: |
          softwareX --test
      # Alternatively, run manual tests
      - name: Run annotation test
        run: |
          softwareX --input test/test.fna --output test_out.gff
          cmp test_out.gff test/result.gff
```

This workflow is named “softwareX”

This workflow will be triggered by pushes or pull requests on the main and dev branches

In GitHub Actions, one can easily define matrices which can also be combined. This workflow runs tests using combined matrices of operating system and Python versions (testing a total of eight combinations in this example)

On GitHub Marketplace, Actions from other developers are available. These can be used to perform common tasks, such as checkout a GitHub repository or setup a particular version of Python.

The “run” keyword specifies commands that are run. These can be single lines or multiple lines. If a command in a job exits with an error, the job will fail.

In this example, a test suite included in the software is run (typically invoked by using the flag “--test”).

Here, a small FASTA file is annotated. Output is compared to an existing output file using “cmp”, which throws an error if files are different.

Figure 8.2: Example YAML file for a GitHub Actions workflow.

8.10 Supplemental Material

Table 8.2: Software tested during the ASM NGS 2020 hackathon

Software Name	Software (URL)	Test File (URL)	Literature Citation (DOI)
BUSCO	https://gitlab.com/ezlab/busco	https://github.com/microbinfie-hackathon2020/CSIS/blob/main/.github/workflows/busco.yml	10.1093/bioinformatics/btv351
Centrifuge	https://github.com/DaehwanKimLab/centrifuge	https://github.com/microbinfie-hackathon2020/CSIS/blob/main/.github/workflows/centrifuge.yml	10.1101/gr.210641.116
CheckM	https://github.com/Ecogenomics/CheckM	https://github.com/microbinfie-hackathon2020/CSIS/blob/main/.github/workflows/checkm.yml	10.1101/gr.186072.114
chewBBACA	https://github.com/B-UMMI/chewBBACA	https://github.com/B-UMMI/chewBBACA/blob/master/.github/workflows/chewbbaca.yml	10.1099/mgen.0.000166
CSIS	https://github.com/microbinfie-hackathon2020/CSIS	https://github.com/microbinfie-hackathon2020/CSIS/blob/main/.github/workflows/CSIS.yml	this manuscript
Genotyphi	https://github.com/katholt/genotyphi	https://github.com/microbinfie-hackathon2020/CSIS/blob/main/.github/workflows/genotyphi.yml	10.1038/ncomms12827
Kraken	https://github.com/DerrickWood/kraken	https://github.com/microbinfie-hackathon2020/CSIS/blob/main/.github/workflows/kraken.yml	10.1186/gb-2014-15-3-r46
Kraken2	https://github.com/DerrickWood/kraken2	https://github.com/microbinfie-hackathon2020/CSIS/blob/main/.github/workflows/kraken2.yml	10.1186/s13059-019-1891-0
KrakenUniq	https://github.com/fbreitwieser/krakenuniq	https://github.com/microbinfie-hackathon2020/CSIS/blob/main/.github/workflows/krakenuniq.yml	10.1186/s13059-018-1568-0
Pangolin	https://github.com/cov-lineages/pangolin	https://github.com/microbinfie-hackathon2020/CSIS/blob/main/.github/workflows/pangolin.yml	absent
Prokka	https://github.com/tseeman/prokka	https://github.com/microbinfie-hackathon2020/CSIS/blob/main/.github/workflows/prokka.yml	10.1093/bioinformatics/btu153
Quast	https://github.com/ablab/quast	https://github.com/microbinfie-hackathon2020/CSIS/blob/main/.github/workflows/quast.yml	10.1093/bioinformatics/btt086
Shovill	https://github.com/tseemann/shovill	https://github.com/microbinfie-hackathon2020/CSIS/blob/main/.github/workflows/shovill.yml	absent
SKESA	https://github.com/ncbi/SKESA	https://github.com/microbinfie-hackathon2020/CSIS/blob/main/.github/workflows/skesa.yml	10.1186/s13059-018-1540-z
Trycycler	https://github.com/rrwick/Trycycler	https://github.com/microbinfie-hackathon2020/CSIS/blob/main/.github/workflows/trycycler.yml	10.5281/zenodo.4430941
Unicycler	https://github.com/rrwick/Unicycler	https://github.com/microbinfie-hackathon2020/CSIS/blob/main/.github/workflows/unicycler.yml	10.1371/journal.pcbi.1005595

8.11 References

[1] Victoria Stodden, Jennifer Seiler, and Zhaokun Ma. "An empirical analysis of journal policy effectiveness for computational reproducibility". en. In: *Proceedings of the National Academy of Sciences* 115.11 (Mar. 2018), pp. 2584–2589. ISSN: 0027-8424, 1091-6490. DOI: `10.1073/pnas.1708290115`. URL: `https://pnas.org/doi/full/10.1073/pnas.1708290115` (visited on 04/01/2022).

[2] Matthew Krafczyk et al. "Scientific Tests and Continuous Integration Strategies to Enhance Reproducibility in the Scientific Software Context". en. In: *Proceedings of the 2nd International Workshop on Practical Reproducible Evaluation of Computer Systems - P-RECS '19*. Phoenix, AZ, USA: ACM Press, 2019, pp. 23–28. ISBN: 978-1-4503-6756-1. DOI: `10.1145/3322790.3330595`. URL: `http://dl.acm.org/citation.cfm?doid=3322790.3330595` (visited on 04/29/2022).

[3] Geoffrey Chang et al. "Retraction". en. In: *Science* 314.5807 (Dec. 2006), pp. 1875–1875. ISSN: 0036-8075, 1095-9203. DOI: `10.1126/science.314.5807.1875b`. URL: `https://www.science.org/doi/10.1126/science.314.5807.1875b` (visited on 04/29/2022).

[4] Barry G. Hall and Stephen J. Salipante. "Retraction: Measures of Clade Confidence Do Not Correlate with Accuracy of Phylogenetic Trees". en. In: *PLoS Computational Biology* 3.7 (2007), e158. ISSN: 1553-734X, 1553-7358. DOI: `10.1371/journal.pcbi.0030158`. URL: `https://dx.plos.org/10.1371/journal.pcbi.0030158` (visited on 04/29/2022).

[5] Ben Busby et al. *Closing gaps between open software and public data in a hackathon setting: User-centered software prototyping*. en. Tech. rep. 5:672. Type: article. F1000Research, May 2016. DOI: `10.12688/f1000research.8382.2`. URL: `https://f1000research.com/articles/5-672` (visited on 04/29/2022).

[6] Bjorn Gruening et al. "Recommendations for the packaging and containerizing of bioinformatics software". en. In: *F1000Research* 7 (Mar. 2019), p. 742. ISSN: 2046-1402. DOI: `10.12688/f1000research.15140.2`. URL: `https://f1000research.com/articles/7-742/v2` (visited on 04/30/2022).

[7] Asher Trockman et al. "Adding sparkle to social coding: an empirical study of repository badges in the *npm* ecosystem". In: *Proceedings of the 40th International Conference on Software Engineering*. ICSE '18. New York, NY, USA: Association for Computing Machinery, 2018, pp. 511–522. ISBN: 978-1-4503-5638-1. DOI: `10.1145/3180155.3180209`. URL: `https://doi.org/10.1145/3180155.3180209` (visited on 04/30/2022).

[8] Mehran Karimzadeh and Michael M Hoffman. "Top considerations for creating bioinformatics software documentation". In: *Briefings in Bioinformatics* 19.4 (July

2018), pp. 693–699. ISSN: 1477-4054. DOI: 10.1093/bib/bbw134. URL: https://doi.org/10.1093/bib/bbw134 (visited on 04/30/2022).

[9] Áine O'Toole et al. "Assignment of epidemiological lineages in an emerging pandemic using the pangolin tool". In: *Virus Evolution* 7.2 (Dec. 2021), veab064. ISSN: 2057-1577. DOI: 10.1093/ve/veab064. URL: https://doi.org/10.1093/ve/veab064 (visited on 04/30/2022).

[10] Mickael Silva et al. "chewBBACA: A complete suite for gene-by-gene schema creation and strain identification". In: *Microbial Genomics* 4.3 (). Publisher: Microbiology Society, e000166. ISSN: 2057-5858, DOI: 10.1099/mgen.0.000166. URL: https://www.microbiologyresearch.org/content/journal/mgen/10.1099/mgen.0.000166 (visited on 04/30/2022).

Chapter 9

General Discussion

The rise of life lost due to microbial pathogens, particularly when associated with the surge of AMR, poses a significant threat to human health around the world. Optimising the diagnostic process is crucial in managing infectious diseases [1]. Currently, the golden standard for clinical microbiology are culture, antimicrobial susceptibility testing, PCR, including syndromic multiplex testing, and serology. Sequencing, when applied, is usually limited to 16S for prokaryotic pathogen identification [2].

In this thesis, we have evaluated the use of bioinformatics methods for the analysis of metagenomic data to allow the rapid identification, virulence analysis and antimicrobial susceptibility prediction of pathogens with clinical relevance, from both diagnostic and surveillance settings. With the widespread use and continuous development of sequencing technologies, bioinformatics has become a cornerstone in modern clinical microbiology. As mentioned previously, the lack of golden standards severely hinders the applicability of bioinformatic methods [3–7].

Metagenomics, and in particular SMg, has emerged as a promising approach for diagnosis from clinical samples and surveillance of organisms of interest from the environment [8–11]. A single metagenomics analysis has the potential to detect common, rare and novel pathogens, and provide a broad overall picture of the microbial content present in a sample. Despite this, it is often unclear whether a given detected microorganism is a contaminant, coloniser or *bona fide* pathogen.

Several limitations have been identified that, in their current form, curb the applicability of these methods in both clinical and public health microbiology.

9.1 Limitation to the application of metagenomics in clinical microbiology

9.1.1 Limitations of sequencing technologies

While the application of genomics in clinical microbiology has been increasing, the translation of genetic information remains challenging. Recent advances in DNA sequencing have expanded their application as a diagnostic tool, but limitations still prevail. After over a quarter of a century of development and maturation, several technologies are available to be used both in research and in the clinic, from the first generation DNA chain termination sequencing to third generation long-read sequencing. Despite this, the main benefit of the current clinical microbiology testing paradigm, in comparison with genomic approaches, is that it allows for cost-effective negative results [2]. The initial capital cost of setting up a genomics-capable facility for the use of metagenomics is considerable [2]. Several factors can reduce the sensitivity and specificity of these methods. Sequencing only one type of molecule, either DNA or RNA, can lead to the missing of sample components such as RNA viruses or non-replicating DNA viruses, although sequencing all nucleic acids present increases the overall cost per sample [10]. Additionally to the sample composition, the specimen volume, collection method, transport and sequencing method can affect SMg sensitivity, with multiple possible approaches to the wet lab work which require optimisation [12]. Regarding the sequencing technologies available, each has its own pitfalls.

First generation sequencing technologies require the input DNA to consist of a pure population of sequences, as each molecule will contribute to the final electropherogram, as it is a superposition of all of the input molecules [13]. As such, it cannot be applied to metagenomic methodologies.

Second generation sequencing so far represent the most popular technology applied in metagenomics [10, 14–16]. Second-generation methods require library preparation including an enrichment or amplification step [13], a time-consuming, bias-inducing procedure that is propagated to the resulting data. Another limitation is the size of the outputted sequences that, despite the massive throughput of some of the machines available, requiring a very small DNA input load to produce up to billions of sequences [15], ranges from 45 to 300 bases in length [17, 18]. This is simply not enough to transverse the most repetitive genomic regions, and severely limits the sensitivity of the methodology as the source organism of the short sequences is hard to determine both by mapping and *de novo* assembly methods. Furthermore, turnaround times range from several hours to full days, not ideal for a timely diagnosis and reporting, and require batching of samples to be run cost-effectively which, depending on the instrument, can range from a few samples to a few dozen, adding to the turnaround time and lowering the sensitivity of pathogen detection [2, 10]. In Chapter 2, the analysis through SMg of 10 samples took between 48-54 hours to complete, which is shorter than culture-based methods if one includes typing. Due to the sequencing depth required to capture the lowest abundant microorganisms in a sample, only the instruments with the highest throughput are recommended for metagenomic sequencing [1]. This, however, is not cost-effective in routine diagnostics where samples need to be immediately processed [9]. Also, as discussed in Chapter 2, when applying sample batching, second generation sequencing-related phenomena such as index hopping (also named index switching) or crosstalk (also called sample bleeding) can introduce false-positive results.

Third generation sequencing technologies have emerged as a viable alternative to second generation methods as longer sequences offer more contextual information and do not have the limitation of bias-inducing pre-sequencing PCR library preparation, as each molecule is sequenced directly [15, 18]. This results in a lower bias, but a much lower throughput of data and a much higher baseline sample input requirement [19]. Long-reads also have the advantage of resolving structural variations and variants in repetitive regions, which are poorly resolved by short-reads and are often excluded in bioinformatics analysis. In particular, ONT sequencing has emerged as an attractive platform for clinical laboratories to adopt due to its low cost and rapid turnaround time, being able to provide results in almost real-time [12]. However, compared to other platforms [19], this method still has problems with base-calling accuracy. Although it can produce reads of up to 2 megabases in length, its main drawbacks have been a low throughput of sequence data and a high error rate (approximately 10%). In Chapter 3 it was shown that even when hybrid assembly with both Illumina and ONT data is employed, leveraging second and third generation methodologies, complete genomic sequences, particularly chimeric ones such as plasmids, are still not fully recovered. Regardless, rapid advancements are being achieved with new, more accurate flowcells and better base calling software [20].

9.1.2 Limitations of host sequence depletion

The unbiased nature of SMg allows the sequencing of the nucleic acid of all microbes (including commensal) and the host. Nearly all DNA and RNA content in most clinical samples is host (human) derived, further lowering the sensitivity of potential pathogen detection [19]. As a result, the presence of an overwhelming amount of host DNA or RNA is one of the most important problems to be addressed in SMg. As observed in Chapter 2, the number of human reads differed between the 10 samples selected for SMg, even when using the same extraction kit and including a human DNA depletion step. This highlights that the ratio between host and microbial DNA is a major determinant of the proportion, and therefore the detection of microbial reads.

There are depletion steps that aim to decrease the relative proportion of human host background sequences through capture probes, lysis and deoxyribonuclease and/or ribonuclease

---

[1] `https://www.illumina.com/systems/sequencing-platforms.html`

treatment [19]. Theoretically, the microbial proportion of the sample is protected within viral capsids and microbial cell walls, but alterations to the microbial composition can still occur. Alternatively, the host sequences can be removed after sequencing, as performed in Chapter 4. In this Chapter, the reads of interest were captured through a mapping approach to a large collection of reference genomes. Still, the opposite methodology can be applied, where the contaminant host sequences are removed (in their majority) by mapping to a human reference genome. This method is not as cost-efficient as host DNA depletion in the bench, as a greater proportion of background sequences are sequenced, but the community composition remains, theoretically, unaltered.

9.1.3 Limitations of the bioinformatic analysis

In addition to the variability of wet lab protocols, the bioinformatics for data handling and interpretation can be resource intensive. Bioinformatic analysis requires highly trained staff, valid analysis tools, including the reference database, good computational infrastructure, and the creation of standardised procedures [12]. In routine settings, automation and standardisation of the analysis are significant for the reliability of the diagnostic and surveillance test results. Computational pipelines for the analysis of SMg data have unique challenges and requirements, from host depletion and taxonomic classification, to metagenomic *de novo* assembly, binning of Metagenomic Assembled Genome (MAG)s and strain characterisation. Additionally, as observed in Chapters 2 and 3, each sequencing platform, from second (Illumina) and third generation (ONT), requires their own data processing steps and quality control metrics as the tools that have been developed in the research community for short-read data are not feasible for long-read data. Data interpretation adds an additional level of complexity.

For *in silico* host depletion, only the sequences that align to a human reference genome will be removed, not necessarily removing the totality of human sequences and, in the lens of patient identity protection, leaving the most identifiable sequences behind. In the same principle, when performing taxonomic characterisation, only sequences present in the database used for sequence identification will be identified, with rare pathogens or emerging strains of pathogens not being reported. This is a limitation of DEN-IM, presented in Chapter 4, where the latter approach is implemented. In Chapter 2, different taxonomic classification tools produced different results for the same sample. Additionally, reference databases are often biased towards certain organisms, as well as certain pathogens that are important to distinguish are similar genetically, with current methods not having enough resolution to do so [19]. Lastly, contamination with flora and/or reagents should be accounted for as it can limit specificity, hence the importance of using and analysing controls alongside samples.

The quality control of the SMg reads is in all similar to any genomics workflow, an essential prerequisite that involves quality trimming and sequencing adaptor removal. In opposition, the assembly step, where reads are stitched into longer fragments, referred to as contigs, is usually tailored to SMg analysis. These contigs are longer sequences that offer better contextual information than reads alone and provide a more complete picture of the microbial community than just the species composition and are usually followed by reconstructing the individual genes and species. In Chapter 4 both metagenomic and traditional genomic assemblers are employed in a two-pronged approach to guarantee the highest chance of success when recovering full DENV genomes from the metagenomic samples. As discussed in Chapter 5, several dedicated metagenomic assembly tools for short-read data are available, generally assumed to perform better when dealing with the complex SMg samples. Despite this assumption, assemblers branded as metagenomic specific did not consistently outperform other genomic assemblers in the metagenomic samples analysed. Additionally, the performance of each assembler varied depending on the species of interest and its abundance in the sample, with less abundant species presenting a significant challenge for all assemblers.

Other areas of bioinformatics not directly covered in this thesis, such as contig binning, also provide their own set of challenges. In order to reconstruct genomes using heterogeneous sequencing data, contig grouping based on an individual genome of origin is done, either by supervised methods, relying on taxonomic information in a database, or unsupervised clustering. The first, like taxonomic assignment, is limited to what is available in the database used, whereas the latter, despite not requiring *a priori* knowledge, tends to be very computationally expensive and with very variable accuracy [21].

There is no standard method for interpreting SMg results. Chapters 2 and 5 highlight how different bioinformatics approaches, and different tools for the same approach, can affect the overall interpretation of the results. In particular, substantial differences were noted between the taxonomic classification tools in Chapter 2, with the results being highly dependent on the tools, and especially the database that was chosen for the analysis greatly impacts its applicability in a clinical setting.

9.2 Better standards in metagenomics for clinical microbiology

9.2.1 The need for better assessment

The constant changes in versions and/or the discontinuation of a bioinformatics tool complicates the standardisation of data analysis. In routine settings, automation and standardisation of the analysis are significant for the reliability of the diagnostic test results. With the lack of proper benchmarks to validate what approach is to be followed, no hope of standardisation can be achieved.

9.2.1.1 Performing proper benchmarking of software

In Chapter 2, the suitability of SMg for the microbiological diagnosis was assessed, with particular emphasis on the bioinformatic process. In total, 3 different bioinformatics pipelines were evaluated to identify those which could provide a clinical microbiologist with the maximum of relevant and accurate information: an open-source Unix-based approach, and two commercial alternatives, Basespace and CLC Genomics Workbench. No approach outperformed the other, and very disparate results were obtained for each one. One of the limitations of this study was that, although several approaches were used, no comprehensive assessment of the tools available for each step of the workflow was performed (quality control, removal of host sequences, taxonomic identification, gene detection and *de novo* assembly). Currently, no open-source, user-friendly, one-stop, comprehensive metagenomics toolkit with a visual user interface for shotgun metagenomics analysis is available, but efforts are being made to change this paradigm. Alternatives, such as Anvi'o [22] for shotgun metagenomics, and Qiime2 [23] for metataxonomics, leverage the individual components in a streamlined manner, but require command-line familiarity. Still, each individual component of such software should be individually evaluated.

In Chapter 4, where ultimately a two-pronged approach was followed including both a genomic and a metagenomic *de novo* assembler. This arose from the need to recover complete CDS sequences, where none of the alternatives on their own was able to perform consistently. Despite several benchmarking studies being available [7, 24–29], the assemblers benchmarked and the datasets used are not consistent, usually limited to a couple of different tools that do not reflect the current landscape of possible *de novo* assembly software. These benchmarks are also usually coupled with the release of a new tool, therefore often skewed towards certain results and do not offer an unbiased assessment.

In Chapter 5, 11 *de novo* assemblers were benchmarked with the same metagenomic samples, selected based on the date of the last update to the software and with a cut-off in 2015. No assembler stood out as an undisputed all-purpose choice for short-read metagenomic prokaryote *de novo* genome assembly, highlighting that efforts are still needed to further improve metagenomic assembler performance.

Despite these efforts, just a small subset of the common bioinformatics methods applied to SMg have been assessed. Regarding the binning process, disparate results have been observed [7, 21, 30, 31], highlighting that the need for proper benchmark cannot be limited to the tools itself, but also the standardisation of the data used for the benchmark. Similarly, the same has been observed for taxonomic assignment [7, 32, 33].

9.2.1.2 The use of mock communities

Chapter 5 shows that suitable mock communities, reproducing the users' samples of interest, can be used as a gold standard to evaluate tool performance. Several well-characterised mock communities are currently available to be used by the community when assessing and benchmarking software suitable for SMg analysis. Chapter 6 features the ZymoBIOMICS Microbial Community Standard, composed of eight bacterial species and two fungi, and is available commercially, with reference sequences made publicly available, with both even and logarithmically distribution of species[2]. Still, ZymoBIOMICS Microbial Community Standards might not be representative of the metagenomic complexity of the samples of interest of most researchers, its relative simplicity means that the results shown probably represent a best-case scenario. Furthermore, in the logarithmically distributed sample, six replicons have a coverage lower than one time, rendering it unrecoverable through assembly and possibly mapping. To address this, two other community standards were analysed: the BMock12 [34] and the NIBSC Gut DNA Reference [35].

The BMock12 community is composed of twelve bacterial species, spanning eight genera. This community dataset includes several closely related strains: two replicons of *Halomonas* sp., three replicons of the *Micromonospora* and two replicons of *Marinobacter* sp. The challenges of this data set are to assemble the genomes of the two Marinobacter and the two Halomonas strains, which come at 85% and 99% ANIb, respectively. Samples were made available of the sequencing products of this community both in second (Illumina) and third-generation sequencing (PacBio and Oxford Nanopore), expanding its utility for the benchmarking of both short-read, long-read and hybrid software. With worth noting that the molarity of each genome in the mock community is not even throughout, resulting in the extreme overrepresentation of some taxa in relation to others. This can have the benefit of providing a form of a limit of detection assessment, depending on the type of analysis performed. Still, besides the challenges that this dataset present, is still a rather simplistic picture of a metagenomic sample.

The NIBSC Gut DNA Reference community dataset, with the evenly distributed Gut-Mix-RR sample and its uneven distribution of species counterpart Gut-HiLo-RR, spans five phyla, thirteen families, sixteen genera and nineteen species, comprising twenty strains that aim to emulate the healthy human gut microbiota. The strains were obtained from Leibniz Institute DSMZ-German Collection of Microorganisms and Cell Cultures GmbH. The genomes in the mock community are fairly diverse, with the challenging genomes being the two *Bifidobacterium longum* subspecies (ANIb=0.95). It's worth denoting that no complete genome is available for eight of the genomes, including one of the *Bifidobacterium longum* subspecies. This makes the assessment of results challenging, if not impossible, as no source of absolute truth is available.

---

[2] `https://www.zymoresearch.com/collections/zymobiomics-microbial-community-standards`

For the replication of more complex communities, the MICROBIOME Community of Special Interest[3] makes available, through their Critical Assessment of Metagenome Interpretation (CAMI) initiative[4], several datasets varying in complexity, replicating several communities such as human microbiome and the rhizosphere [7, 29]. Unfortunately, sources of truth are not made available, including the complete genomes of the community profile, and assessment of results is difficult, if not impossible. Despite their popularity, no information is provided in the manuscripts or in the supplemental materials about the source of the genomes in each dataset. It is only mentioned that the community composition was designed according to undisclosed specified criteria. Additionally, only assemblies are made available, and not the source references for the generation of the mock datasets. These assemblies are, unfortunately, often the result of a pooled assembly and are extremely fragmented. This makes this dataset not suitable for benchmark studies, which require defined mock communities as a source of truth.

The Fishing in the Soup dataset [36] is a mock community containing DNA of foodborne bacteria. The standard consists of 34 equimolarly pooled bacterial DNAs belonging to 17 genera and 30 species, elected based on their incidence in foodstuff and pathogenicity. Despite its diversity, not all reference sequences are available, and the used reference sequences are not the ones corresponding to the sequenced strains in some cases. Although two sets of the same species replicons are present (*Bacillus cereus*, *Salmonella enterica*, *Clostridium perfringens*), their ANIb is lower than the BMock12 *halomonas* replicons, and no independent references for different strains of the same species are provided.

Similarly to the ZymoBIOMICS Microbial Community Standard, communities with good metadata and reference genomes publicly available are available for acquisition and sequencing from American Type Culture Collection (ATCC). The ATCC 10 Strain Even Mix Genomic Material MSA-1000[5] and 20 Strain Even Mix Genomic Material MSA-1002[6] consists of, respectively, 10 and 20, fully sequenced, characterised, and authenticated ATCC Genuine Cultures mixed evenly, selected based on pathogen relevance. It also provides Gut Microbiome Genomic Mix MSA-1006[7] composed of an even mixture of genomic DNA prepared from 12 fully sequenced, characterised, and authenticated bacterial species observed in normal and atypical gut microbial communities. Just like ZymoBIOMICS Microbial Community Standards, the simplicity of the ATCC mock communities might not be representative of the metagenomic complexity of the samples of interest to most researchers, representing simplistic scenarios with limited usability.

No ideal solution is currently available in the research community or the market, highlighting the need for better, more diverse and well-described community standards representing realistic scenarios. Chapter 5 the results of the ZymoBIOMICS Microbial Commu-

---

[3] `https://www.microbiome-cosi.org/`

[4] `https://data.cami-challenge.org/`

[5] `https://www.atcc.org/products/msa-1000`

[6] `https://www.atcc.org/products/msa-1002`

[7] `https://www.atcc.org/products/msa-1006`

nity Standard, BMock12 and NIBSC Gut DNA community standards suggest that the actual genome of interest and community composition play an important part in the results of individual assemblers. Particularly, The performance of each assembler varied depending on the species of interest and its abundance in the sample. For this reason, users should analyse the results of sequencing mock communities (ideally) or artificially generated reads simulating their samples of interest to guide their downstream metagenomic analysis and tool selection.

9.2.2 The need for better reproducibility

The analysis of biological data is driven by the development of a myriad of open-source software tools, each carrying out a specialised step, that then can be chained together to generate new knowledge and insights. However, as with any complex system, susceptibility to variation can cause the entire system to collapse. Such variation can be the use of different operating systems, the availability of computational resources, and ambiguities with tool versioning and documentation [37, 38].

One of the biggest challenges when dealing with metagenomic data is the lack of golden standards, although major efforts are being made on the standardisation and assessment of software, both commercial and open source [4–7]. A plethora of free-to-use, open-source tools are available specifically for metagenomic data, both short and long-read data, and several combinations of these tools can be used to characterise the causative agent in a patient's infection in a fraction of the time required by the traditional methods. Unfortunately, as observed in Chapter 5, massive differences in success and performance were observed in software built for the same purpose. In this case, the *de novo* assembly of metagenomic genomic data. There are several steps that can be implemented to ensure the transparency and reproducibility of the chosen workflow for metagenomic analysis, regardless of the tools chosen.

In Chapters 4 and 5, one of the main key objectives was to a fully reproducible, scalable workflow for one, the analysis and characterization of DENV directly from metagenomic samples, and two, evaluate the metagenomic *de novo* assembly process through the benchmark of the most commonly used tools. Chapter 6 focused on the development of a software utility for the harmonization of the output of several AMR detection tools. For that, several steps were implemented. Below there are several strategies that have been implemented throughout this work to ensure the reproducibility of the results obtained.

9.2.2.1 The need for clear documentation

The most fundamental strategy for enabling the reproducibility of software, and the results obtained thereafter, is through clear documentation of both the software and the process followed. This includes not only a clear explanation of how the software works, and what is expected as input and output, but also the inclusion of a "quick start" guide to validate tool installation and behaviour. Additionally, the provision of an example dataset is often invaluable to have a "ground truth" to compare the obtained results against, but this will be covered more in-depth in the 9.2.2.5 section.

The narratives presented in the documentation, both in the manuscripts describing the use of a given software tool, or in the location where the software is hosted, help the community understand how to re-execute a given analysis, and more importantly, how to apply the same process to their own data. This requires that the steps followed are written in enough detail to be able to be retraced by others. Despite these efforts, documentation alone is not enough to ensure reproducibility as computer configurations can differ greatly, from different operative systems to the libraries available in the system itself (see below).

As important as the overall narrative describing a tool, clear documentation within the tool itself, added during its development, represents an important step for readability, and therefore accountability, of the code produced. This also represents an important requirement for the maintenance of the tool long-term, even after the original developer or team of developers is no longer actively maintaining the software.

9.2.2.2 The need for easy software distribution

Favouring open-source tools, with clear documentation describing the methodology implemented, and stating the version of the software used and which parameters were used enables the comparison of results. This requires good management of the software installation, and its dependencies, for this information to be trackable and distributed to the community. Bioinformatics software comes in a variety of programming languages and requires diverse installation methods. This heterogeneity makes the management of software complicated, error-prone, and time-consuming. To reproduce the same software on different machines, full control of the computational environment is required, which usually requires administrative privileges that are often not available. It has been described that as much as 28% of all resources are currently not accessible through the original Uniform Resource Locator (URL) published in the paper they first appeared in [38]. Although several journals have clear requirements for the sharing of code and data, there are currently no effective requirements to promote installability and long-term archival stability of software tools [38], strongly hindering the effectiveness of the tool distribution and usability.

To solve this issue several options have risen throughout time. For language-specific package managers, PyPi[8], the Python's Package Indexer, or CRAN[9], the Comprehensive R Archive Network, follow the principle that precompiled binary distributions are hosted in a centralised service. This makes the sharing of software easier as a determined version of a

---

[8] `https://pypi.org/`

[9] `https://cran.r-project.org/`

package can be downloaded and executed by the user without the need for compilation from the source. Still, the installation is system-wide, and conflicts can arise.

Efforts, such as Conda package manager[10], and in particular Bioconda [39], have become a popular means to overcome these challenges for all major operating systems. Package managers normalise software installations across ecosystems by describing each software with a human-readable "recipe" that defines its dependencies, as well as a simple "build script" that performs the steps necessary to build and install the software. These installations happen in an isolated environment, obliviating the reliance on system-wide access privileges. This way, the software can exist in an isolated form, allowing the control of software versions and the versions of its dependencies without conflicts arising if a given package gets updated. Additionally, the current status of an environment can be stored in a file and distributed to collaborators.

Container software, like Docker[11] or Singularity [40], came to simplify this process by containerising all the software tools and required run environments. These tools are changing the way scientists and researchers develop, deploy and exchange scientific software. Containers constitute lightweight software components and libraries that can be quickly packaged, are designed to run anywhere, and are useful and essential tools to leverage bioinformatics software reproducibility [6, 41]. Their principle is that a container is a standard unit of software that packages up code and all its dependencies so the application runs quickly and reliably from one computing environment to another. In comparison with alternatives such as traditional Virtual Machines, containers are much more lightweight, running directly in the host operating system, in an isolated manner, while containing all requirements of software needed to run it, such as the code, system tools and system libraries. Their biggest advantage is that a software container will run the same regardless of the environment it is deployed as it runs as an isolated process in user space. Still, one of the main disadvantages is that when a container is created, it's not possible to backtrack its creation, therefore the exact dependencies installed are unknown. It becomes a problem of "it simply works" without having the ability to know exactly how, so although reproducible, containers are not transparent.

9.2.2.3 The need for Workflow managers

The use of workflow managers, like nextflow [42], Snakemake [43] or the Galaxy Project [44], pushes reproducibility to the next level, enabling the workflow to be executed with the exact same parameters in the same conditions in a multitude of different environments [37]. Bioinformatic workflows managers offer integration with containers and package managers while providing portability, reproducibility, automatic resource management and native cloud support, not being limited to running in a local environment.

---

[10] `https://conda.io`

[11] `https://www.docker.com/`

Traditional computational pipelines are usually developed using custom scripts, being highly coupled to their local compute infrastructure. Furthermore, they lack the option to resume failed steps, and often lack parameter tracking, tool versioning and require manual installation, making them difficult to share and maintain, greatly hindering reproducibility. Workflow managers, being reactive workflow frameworks, allow the creation of pipelines with asynchronous, and therefore implicitly parallelised data streams. In the case of Nextflow, it provides its own domain language for building a pipeline, which can cause the learning curve too steep, but allows the incorporation of existing tools and scripts written in other languages, offering the possibility to effortlessly port tools over and minimising refactoring. Although the creation of workflow pipelines is designed for bioinformaticians familiar with programming, its execution is for everyone.

Another advantage of workflow managers is the automation of input parameters and software tool version tracking, often generating reports with detailed information regarding its execution. Re-entrancy allows users to run a pipeline from its last successfully executed step, rather than from the beginning, in the case of disruption. They also process the intermediate files generated during an execution of a pipeline automatically. This is a true paradigm shift where monolithic pipelines are replaced with flexible alternatives, simplifying the implementation of robust and complex analysis while providing additional features that help optimise resource management and reproducibility.

9.2.2.4 The need for Version Control

Even though the use of container software coupled with workflow managers, these approaches are not, *per se*, versionable. Version control systems, such as Git [45], were designed not only to solve the problem of multiple people working on the same code (collaboration) but also on how to store versions properly. Through Git, a stage of a project can be saved, alongside a message, which allows backtracking to previous versions, and auditability to track what changes happened when, and by whom.

Remote repositories, such as GitHub[12], the single largest host for Git repositories, are the central point of collaboration for millions of developers and projects. A large percentage of all Git repositories are hosted on GitHub, and many open-source projects use it for Git hosting, issue tracking, code review, and code dissemination. This benefits the computational biology community immensely, effectively guaranteeing permanent access to published scientific URL [38], either directly or through an archived version.

---

[12] `https://github.com/`

9.2.2.5 The need for Open Integration Testing

As explored in chapter 8, the use of containers, workflow managers and proper version control do not guarantee the correct executability of a given pipeline. Testing is a critical aspect of scientific software development, either through the traditional technique approach where the code is tested after its development or through test-driven development approach where a suite of tests is first created and the code developed iteratively to solve the tests [46]. Despite this, and as discussed in Chapter 8, automated software testing remains underused in scientific software, an example of good software engineering principles not yet widely adopted by the community.

When writing or maintaining software, implementing software tests can greatly help to prevent errors and facilitate code development. In Chapter 8, we defined a set of seven recommendations for researchers in microbial bioinformatics looking to develop software tests. Good software engineering practices are sometimes lacking in the field of microbial bioinformatics as many researchers receive little or no formal training in software engineering. The established set of recommendations aims to aid researchers planning to implement software tests alongside their software development practices. Ultimately, the integration of automatic software testing with version control systems, workflow managers and container software is a good strategy to ensure the reproducibility of code and analysis in metagenomics for clinical microbiology.

9.2.3 The need for better Interpretability

Understanding how to report complex genomic test results to stakeholders who may have varying familiarity with genomics, including clinicians, laboratorians, epidemiologists, and researchers, is critical to the successful and sustainable implementation of metagenomics in clinical microbiology. One of the main focuses of chapters 4 and 5 was the production of interactive, responsive and intuitive reports for the developed workflows. The interactive HTML reports developed provide an intuitive platform for data exploration, allowing the user to highlight specific samples, filter and re-order the data tables, and export the plots as needed.

In chapter 4 the report provided at the end of the workflow execution contains all results divided into four sections: report overview, tables, charts and phylogenetic tree. The report overview and all tables allow for the selection, filtering and highlighting of particular samples in the analysis. All tables have information on if a sample failed or passed the quality control metrics highlighted by green, yellow or red signs for a pass, warning and fail messages, respectively

In chapter 5, the interactive report provides an intuitive platform for data exploration, allowing the user to easily sift through global and reference specific performance metrics for each sample, as well as providing information on the assemblers executed to allow traceability of the results. Producing an extensive, metric-rich report allows users interested in different aspects of assembler performance to make informed decisions, particularly when choosing among the top-performing assemblers, which show only minor differences.

Design Study Methodology, a human-centred approach drawn from the information visualisation domain, to redesign an existing clinical report has been proposed previously [47], integrating quantitative and qualitative feedback for representing complex microbial genomic data. We further extend this by providing interactive HTML reports, possible to be shared as a traditional report, but with the advantage of allowing a more in-depth exploration of the results through explorable tables and plots.

9.2.4 The need for better Interoperability

Despite this richness in genomic information, the same is not observed for the metadata that accompanies it. Chapters 6 and 7 propose two metadata specification schemas, the first focused on the reporting of AMR from genomic data, and the second a minimum metadata standard for the sharing of SARS-CoV-2 genomic information in public repositories, the latter already adopted by NCBI Submission Portal.

Similarly to what has been observed in the SARS-CoV-2 response, where an unprecedented deployment of genomic surveillance worldwide has been observed, the application of metagenomic to clinical microbiology will require the standardisation of procedures, one of them being the metadata collected and how the results are displayed to stakeholders.

Contextual data consists of sample metadata (e.g., collection date, sample type, geographical location of sample collection), as well as laboratory (e.g., date and location testing, cycle threshold (CT) values), clinical outcomes (e.g., hospitalisation, death, recovery), epidemiological (e.g., age, gender, exposures, vaccination status) and methods (e.g., sampling, sequencing, bioinformatics) that enable the interpretation sequence data (e.g., previous examples). High-quality contextual data is also crucial for quality control. For example, detecting systematic batch effect errors related to certain sequencing centres and methods can help evaluate which variants represent real, circulating viruses, as opposed to artefacts of sample handling or sequencing which may arise due to different aspects of experimental design, laboratory procedures, bioinformatics processing, and applied quality control thresholds [48–50].

Good data stewardship practices are not only critical for auditability and reproducibility, but for posterity and can help future-proof information used to inform a diagnosis or deal with a public health crisis. Contextual data, however, is often collected on a project-specific basis according to local needs and reporting requirements which results in the collection of different data types at different levels of granularity, with different meanings and implicit bias of variables and attributes. Furthermore, the information is often collected as free text, or if structured, according to an organisation or initiative-specific data dictionaries, using different fields, terms, formats, abbreviations, and jargon.

As discussed in chapter 7, minimum information checklists (MIxS [51], MIGS [52], the NIAID/BRC Project and Sample Application Standard [53]) and various interoperable ontologies (OBO Foundry [54]), make information easier to aggregate and reuse. As such, a fit-for-purpose contextual data schema, taking into consideration ethical, practical and privacy concerns, is necessary for not only the reporting of metagenomic results but also its dissemination through, for example, public genomic repositories. This aids not only the inference of information for diagnosis but also public health efforts for the identification of potential outbreaks.

9.2.5 The need for Crowdsourcing

The term ‘crowdsourcing’ is used to describe systems for accomplishing directed work that requires human intelligence. These human-powered systems are built to solve discrete tasks with clear endpoints [55]. Crowdsourcing for better standards represents a viable way to adopt better practices for the use of metagenomics in clinical microbiology.

Different forms of crowdsourcing were successfully applied to address key problems in bioinformatics [55–57]. Chapter 8 represents a proof of concept based on our experiences with developing software tests during a hackathon organised prior to the ASM Conference on Rapid Applied Microbial Next-Generation Sequencing and Bioinformatic Pipelines (ASM NGS) in 2020. In the end, we defined a set of seven recommendations for researchers in microbial bioinformatics looking to develop software tests. Although limited to software testing, it represents a viable form of engaging the community in order to uphold new standards.

In the application of metagenomics to clinical diagnostics, we have discussed major pitfalls that require not only expert human input but also wide adoption by the community. Crowdsourcing generally begins where automation fails, and this is where initiatives such as hackathons represent a fundamental change in the way that scientific work is performed and distributed within the community. Recalling the original definition, crowdsourcing is a shift from work done in-house to work done in the open by anyone that is able. This means not only that we can often solve more problems more efficiently, but also that different people are solving them. Ultimately, is through crowdsourcing that the fastest and most efficient way can be applied to change golden standards, adopt new paradigms and ensure that the necessary conditions are widely spread.

9.3 References

[1] Theo Vos et al. “Global burden of 369 diseases and injuries in 204 countries and territories, 1990–2019: a systematic analysis for the Global Burden of Disease Study 2019”. en. In: *The Lancet* 396.10258 (Oct. 2020), pp. 1204–1222. ISSN: 01406736. DOI: 10.1016/S0140-6736(20)30925-9. URL: https://linkinghub.elsevier.com/retrieve/pii/S0140673620309259 (visited on 05/16/2022).

[2] Alexander L. Greninger. “The challenge of diagnostic metagenomics”. In: *Expert Review of Molecular Diagnostics* 18.7 (July 2018). Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/14737159.2018.1487292, pp. 605–615. ISSN: 1473-7159. DOI: 10.1080/14737159.2018.1487292. URL: https://doi.org/10.1080/14737159.2018.1487292 (visited on 05/23/2022).

[3] J. A. Carriço et al. “A primer on microbial bioinformatics for nonbioinformaticians”. en. In: *Clinical Microbiology and Infection* 24.4 (2018), pp. 342–349. ISSN: 1198-743X. DOI: 10.1016/j.cmi.2017.12.015. URL: https://www.sciencedirect.com/science/article/pii/S1198743X17307097 (visited on 02/18/2022).

[4] Natacha Couto et al. “Critical steps in clinical shotgun metagenomics for the concomitant detection and typing of microbial pathogens”. en. In: *Scientific Reports* 8.1 (Dec. 2018), p. 13767. ISSN: 2045-2322. DOI: 10.1038/s41598-018-31873-w. URL: http://www.nature.com/articles/s41598-018-31873-w (visited on 05/11/2022).

[5] Alexandre Angers-Loustau et al. “The challenges of designing a benchmark strategy for bioinformatics pipelines in the identification of antimicrobial resistance determinants using next generation sequencing technologies”. en. In: *F1000Research* 7 (Dec. 2018), p. 459. ISSN: 2046-1402. DOI: 10.12688/f1000research.14509.2. URL: https://f1000research.com/articles/7-459/v2 (visited on 03/25/2021).

[6] Bjorn Gruening et al. “Recommendations for the packaging and containerizing of bioinformatics software”. en. In: *F1000Research* 7 (Mar. 2019), p. 742. ISSN: 2046-1402. DOI: 10.12688/f1000research.15140.2. URL: https://f1000research.com/articles/7-742/v2 (visited on 04/30/2022).

[7] Alexander Sczyrba et al. “Critical Assessment of Metagenome Interpretation—a benchmark of metagenomics software”. en. In: *Nature Methods* 14.11 (Nov. 2017). Number: 11 Publisher: Nature Publishing Group, pp. 1063–1071. ISSN: 1548-7105. DOI: 10.1038/nmeth.4458. URL: https://www.nature.com/articles/nmeth.4458 (visited on 03/20/2022).

[8] Nicholas J. Loman et al. “A Culture-Independent Sequence-Based Metagenomics Approach to the Investigation of an Outbreak of Shiga-Toxigenic Escherichia coli O104:H4”. In: *JAMA* 309.14 (2013), pp. 1502–1510. ISSN: 0098-7484. DOI: 10.

1001/jama.2013.3231. URL: https://doi.org/10.1001/jama.2013.3231 (visited on 02/28/2022).

[9] J. W. A. Rossen, A. W. Friedrich, and J. Moran-Gilad. "& ESCMID Study Group for Genomic and Molecular Diagnostics (ESGMD). Practical issues in implementing whole-genome-sequencing in routine diagnostic microbiology". In: *Clin. Microbiol. Infect.* 24 (2018). DOI: 10.1016/j.cmi.2017.11.001. URL: https://doi.org/10.1016/j.cmi.2017.11.001.

[10] Leonard Schuele et al. "Future potential of metagenomics in microbiology laboratories". en. In: *Expert Review of Molecular Diagnostics* 21.12 (Dec. 2021), pp. 1273–1285. ISSN: 1473-7159, 1744-8352. DOI: 10.1080/14737159.2021.2001329. URL: https://www.tandfonline.com/doi/full/10.1080/14737159.2021.2001329 (visited on 05/20/2022).

[11] Charles Y. Chiu and Steven A. Miller. "Clinical metagenomics". en. In: *Nature Reviews Genetics* 20.6 (June 2019). Number: 6 Publisher: Nature Publishing Group, pp. 341–355. ISSN: 1471-0064. DOI: 10.1038/s41576-019-0113-7. URL: https://www.nature.com/articles/s41576-019-0113-7 (visited on 02/08/2022).

[12] Lauren M. Petersen et al. "Third-Generation Sequencing in the Clinical Laboratory: Exploring the Advantages and Challenges of Nanopore Sequencing". In: *Journal of Clinical Microbiology* 58.1 (Dec. 2019), e01315–19. ISSN: 0095-1137. DOI: 10.1128/JCM.01315-19. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6935936/ (visited on 05/23/2022).

[13] Ian S. Hagemann. "Overview of Technical Aspects and Chemistries of Next-Generation Sequencing". en. In: *Clinical Genomics*. Elsevier, 2015, pp. 3–19. ISBN: 978-0-12-404748-8. DOI: 10.1016/B978-0-12-404748-8.00001-0. URL: https://linkinghub.elsevier.com/retrieve/pii/B9780124047488000010 (visited on 02/08/2022).

[14] J. W. A. Rossen et al. "Practical issues in implementing whole-genome-sequencing in routine diagnostic microbiology". eng. In: *Clinical Microbiology and Infection: The Official Publication of the European Society of Clinical Microbiology and Infectious Diseases* 24.4 (Apr. 2018), pp. 355–360. ISSN: 1469-0691. DOI: 10.1016/j.cmi.2017.11.001.

[15] Nicholas J. Loman and Mark J. Pallen. "Twenty years of bacterial genome sequencing". en. In: *Nature Reviews Microbiology* 13.12 (Dec. 2015). Number: 12 Publisher: Nature Publishing Group, pp. 787–794. ISSN: 1740-1534. DOI: 10.1038/nrmicro3565. URL: https://www.nature.com/articles/nrmicro3565 (visited on 02/08/2022).

[16] Nicholas J. Loman et al. "High-throughput bacterial genome sequencing: an embarrassment of choice, a world of opportunity". en. In: *Nature Reviews Microbiology* 10.9 (Sept. 2012). Number: 9 Publisher: Nature Publishing Group, pp. 599–606.

ISSN: 1740-1534. DOI: 10.1038/nrmicro2850. URL: https://www.nature.com/articles/nrmicro2850 (visited on 02/08/2022).

[17] Nicholas J Loman et al. "Performance comparison of benchtop high-throughput sequencing platforms". en. In: *Nature Biotechnology* 30.5 (May 2012), pp. 434–439. ISSN: 1087-0156, 1546-1696. DOI: 10.1038/nbt.2198. URL: http://www.nature.com/articles/nbt.2198 (visited on 02/14/2022).

[18] Şule Ari and Muzaffer Arikan. "Next-Generation Sequencing: Advantages, Disadvantages, and Future". en. In: *Plant Omics: Trends and Applications*. Ed. by Khalid Rehman Hakeem, Hüseyin Tombuloğlu, and Güzin Tombuloğlu. Cham: Springer International Publishing, 2016, pp. 109–135. ISBN: 978-3-319-31703-8. DOI: 10.1007/978-3-319-31703-8_5. URL: https://doi.org/10.1007/978-3-319-31703-8_5 (visited on 05/18/2022).

[19] Wei Gu, Steve Miller, and Charles Y. Chiu. "Clinical Metagenomic Next-Generation Sequencing for Pathogen Detection". In: *Annual review of pathology* 14 (Jan. 2019), pp. 319–338. ISSN: 1553-4006. DOI: 10.1146/annurev-pathmechdis-012418-012751. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6345613/ (visited on 05/18/2022).

[20] Mantas Sereika et al. *Oxford Nanopore R10.4 long-read sequencing enables near-perfect bacterial genomes from pure cultures and metagenomes without short-read or reference polishing*. en. preprint. Microbiology, Oct. 2021. DOI: 10.1101/2021.10.27.466057. URL: http://biorxiv.org/lookup/doi/10.1101/2021.10.27.466057 (visited on 05/23/2022).

[21] Richa Bharti and Dominik G Grimm. "Current challenges and best-practice protocols for microbiome analysis". In: *Briefings in Bioinformatics* 22.1 (Jan. 2021), pp. 178–193. ISSN: 1477-4054. DOI: 10.1093/bib/bbz155. URL: https://doi.org/10.1093/bib/bbz155 (visited on 05/25/2022).

[22] A. Murat Eren et al. "Anvi'o: an advanced analysis and visualization platform for 'omics data". en. In: *PeerJ* 3 (Oct. 2015). Publisher: PeerJ Inc., e1319. ISSN: 2167-8359. DOI: 10.7717/peerj.1319. URL: https://peerj.com/articles/1319 (visited on 06/01/2022).

[23] Evan Bolyen et al. "Reproducible, interactive, scalable and extensible microbiome data science using QIIME 2". en. In: *Nature Biotechnology* 37.8 (Aug. 2019). Number: 8 Publisher: Nature Publishing Group, pp. 852–857. ISSN: 1546-1696. DOI: 10.1038/s41587-019-0209-9. URL: https://www.nature.com/articles/s41587-019-0209-9 (visited on 03/03/2022).

[24] Jason R. Miller, Sergey Koren, and Granger Sutton. "Assembly algorithms for next-generation sequencing data". en. In: *Genomics* 95.6 (June 2010), pp. 315–327. ISSN: 08887543. DOI: 10.1016/j.ygeno.2010.03.001. URL: https://linkinghub.elsevier.com/retrieve/pii/S0888754310000492 (visited on 03/25/2021).

[25] Dent Earl et al. "Assemblathon 1: a competitive assessment of de novo short read assembly methods". eng. In: *Genome Research* 21.12 (Dec. 2011), pp. 2224–2241. ISSN: 1549-5469. DOI: `10.1101/gr.126599.111`.

[26] Keith R Bradnam et al. "Assemblathon 2: evaluating de novo methods of genome assembly in three vertebrate species". en. In: *GigaScience* 2.1 (Dec. 2013), p. 10. ISSN: 2047-217X. DOI: `10.1186/2047-217X-2-10`. URL: `https://academic.oup.com/gigascience/article-lookup/doi/10.1186/2047-217X-2-10` (visited on 04/13/2021).

[27] Xingyu Liao et al. "Current challenges and solutions of de novo assembly". en. In: *Quantitative Biology* 7.2 (June 2019), pp. 90–109. ISSN: 2095-4689, 2095-4697. DOI: `10.1007/s40484-019-0166-9`. URL: `http://link.springer.com/10.1007/s40484-019-0166-9` (visited on 03/25/2021).

[28] Zhao Chen, David L. Erickson, and Jianghong Meng. "Benchmarking hybrid assembly approaches for genomic analyses of bacterial pathogens using Illumina and Oxford Nanopore sequencing". en. In: *BMC Genomics* 21.1 (Dec. 2020), p. 631. ISSN: 1471-2164. DOI: `10.1186/s12864-020-07041-8`. URL: `https://bmcgenomics.biomedcentral.com/articles/10.1186/s12864-020-07041-8` (visited on 08/28/2021).

[29] Fernando Meyer et al. "Critical Assessment of Metagenome Interpretation: the second round of challenges". en. In: *Nature Methods* 19.4 (Apr. 2022), pp. 429–440. ISSN: 1548-7091, 1548-7105. DOI: `10.1038/s41592-022-01431-4`. URL: `https://www.nature.com/articles/s41592-022-01431-4` (visited on 05/11/2022).

[30] Yi Yue et al. "Evaluating metagenomics tools for genome binning with real metagenomic datasets and CAMI datasets". In: *BMC Bioinformatics* 21.1 (July 2020), p. 334. ISSN: 1471-2105. DOI: `10.1186/s12859-020-03667-3`. URL: `https://doi.org/10.1186/s12859-020-03667-3` (visited on 05/25/2022).

[31] Chao Yang et al. "A review of computational tools for generating metagenome-assembled genomes from metagenomic sequencing data". en. In: *Computational and Structural Biotechnology Journal* 19 (Jan. 2021), pp. 6301–6314. ISSN: 2001-0370. DOI: `10.1016/j.csbj.2021.11.028`. URL: `https://www.sciencedirect.com/science/article/pii/S2001037021004931` (visited on 05/25/2022).

[32] Simon H. Ye et al. "Benchmarking Metagenomics Tools for Taxonomic Classification". en. In: *Cell* 178.4 (Aug. 2019), pp. 779–794. ISSN: 0092-8674. DOI: `10.1016/j.cell.2019.07.010`. URL: `https://www.sciencedirect.com/science/article/pii/S0092867419307755` (visited on 05/25/2022).

[33] Javier Tamames, Marta Cobo-Simón, and Fernando Puente-Sánchez. "Assessing the performance of different approaches for functional and taxonomic annotation of metagenomes". In: *BMC Genomics* 20.1 (Dec. 2019), p. 960. ISSN: 1471-2164. DOI: `10.1186/s12864-019-6289-6`. URL: `https://doi.org/10.1186/s12864-019-6289-6` (visited on 05/25/2022).

[34] Volkan Sevim et al. “Shotgun metagenome data of a defined mock community using Oxford Nanopore, PacBio and Illumina technologies”. en. In: *Scientific Data* 6.1 (Nov. 2019). Number: 1 Publisher: Nature Publishing Group, p. 285. ISSN: 2052-4463. DOI: `10.1038/s41597-019-0287-z`. URL: `https://www.nature.com/articles/s41597-019-0287-z` (visited on 06/30/2022).

[35] Gregory C. A. Amos et al. “Developing standards for the microbiome field”. In: *Microbiome* 8.1 (June 2020), p. 98. ISSN: 2049-2618. DOI: `10.1186/s40168-020-00856-3`. URL: `https://doi.org/10.1186/s40168-020-00856-3` (visited on 06/30/2022).

[36] Josephine Grützke et al. “Fishing in the Soup – Pathogen Detection in Food Safety Using Metabarcoding and Metagenomic Sequencing”. In: *Frontiers in Microbiology* 10 (2019). ISSN: 1664-302X. URL: `https://www.frontiersin.org/articles/10.3389/fmicb.2019.01805` (visited on 09/01/2022).

[37] Laura Wratten, Andreas Wilm, and Jonathan Göke. “Reproducible, scalable, and shareable analysis pipelines with bioinformatics workflow managers”. en. In: *Nature Methods* 18.10 (Oct. 2021). Number: 10 Publisher: Nature Publishing Group, pp. 1161–1168. ISSN: 1548-7105. DOI: `10.1038/s41592-021-01254-9`. URL: `https://www.nature.com/articles/s41592-021-01254-9` (visited on 05/26/2022).

[38] Serghei Mangul et al. “Challenges and recommendations to improve the installability and archival stability of omics computational tools”. In: *PLoS Biology* 17.6 (June 2019), e3000333. ISSN: 1544-9173. DOI: `10.1371/journal.pbio.3000333`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6605654/` (visited on 09/05/2022).

[39] Björn Grüning et al. “Bioconda: sustainable and comprehensive software distribution for the life sciences”. en. In: *Nature Methods* 15.7 (July 2018). Number: 7 Publisher: Nature Publishing Group, pp. 475–476. ISSN: 1548-7105. DOI: `10.1038/s41592-018-0046-7`. URL: `https://www.nature.com/articles/s41592-018-0046-7` (visited on 09/02/2022).

[40] Gregory M. Kurtzer, Vanessa Sochat, and Michael W. Bauer. “Singularity: Scientific containers for mobility of compute”. en. In: *PLOS ONE* 12.5 (Nov. 2017). Publisher: Public Library of Science, e0177459. ISSN: 1932-6203. DOI: `10.1371/journal.pone.0177459`. URL: `https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0177459` (visited on 03/17/2022).

[41] Carl Boettiger. “An introduction to Docker for reproducible research”. en. In: *ACM SIGOPS Operating Systems Review* 49.1 (Jan. 2015), pp. 71–79. ISSN: 0163-5980. DOI: `10.1145/2723872.2723882`. URL: `https://dl.acm.org/doi/10.1145/2723872.2723882` (visited on 04/01/2022).

[42] Paolo Di Tommaso et al. “Nextflow enables reproducible computational workflows”. en. In: *Nature Biotechnology* 35.4 (Apr. 2017), pp. 316–319. ISSN: 1087-0156, 1546-1696. DOI: `10.1038/nbt.3820`. URL: `http://www.nature.com/articles/nbt.3820` (visited on 03/24/2021).

[43] Johannes Köster and Sven Rahmann. “Snakemake—a scalable bioinformatics workflow engine”. In: *Bioinformatics* 28.19 (Oct. 2012), pp. 2520–2522. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/bts480`. URL: `https://doi.org/10.1093/bioinformatics/bts480` (visited on 10/17/2022).

[44] Enis Afgan et al. “The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update”. In: *Nucleic Acids Research* 46.W1 (July 2018), W537–W544. ISSN: 0305-1048. DOI: `10.1093/nar/gky379`. URL: `https://doi.org/10.1093/nar/gky379` (visited on 03/17/2022).

[45] Scott Chacon and Ben Straub. *Pro git*. Apress, 2014.

[46] Matthew Krafczyk et al. “Scientific Tests and Continuous Integration Strategies to Enhance Reproducibility in the Scientific Software Context”. en. In: *Proceedings of the 2nd International Workshop on Practical Reproducible Evaluation of Computer Systems - P-RECS ’19*. Phoenix, AZ, USA: ACM Press, 2019, pp. 23–28. ISBN: 978-1-4503-6756-1. DOI: `10.1145/3322790.3330595`. URL: `http://dl.acm.org/citation.cfm?doid=3322790.3330595` (visited on 04/29/2022).

[47] Anamaria Crisan et al. “Evidence-based design and evaluation of a whole genome sequencing clinical report for the reference microbiology laboratory”. In: *PeerJ* 6 (Jan. 2018), e4218. ISSN: 2167-8359. DOI: `10.7717/peerj.4218`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5767084/` (visited on 06/03/2022).

[48] Nicola De Maio et al. *Issues with SARS-CoV-2 sequencing data - SARS-CoV-2 coronavirus / nCoV-2019 Genomic Epidemiology*. en. May 2020. URL: `https://virological.org/t/issues-with-sars-cov-2-sequencing-data/473` (visited on 04/27/2022).

[49] Mikhail Rayko and Aleksey Komissarov. *Quality control of low-frequency variants in SARS-CoV-2 genomes*. en. preprint. Genomics, Apr. 2020. DOI: `10.1101/2020.04.26.062422`. URL: `http://biorxiv.org/lookup/doi/10.1101/2020.04.26.062422` (visited on 04/27/2022).

[50] Leo L. M. Poon et al. “Recurrent mutations associated with isolation and passage of SARS coronavirus in cells from non-human primates”. eng. In: *Journal of Medical Virology* 76.4 (Aug. 2005), pp. 435–440. ISSN: 0146-6615. DOI: `10.1002/jmv.20379`.

[51] Pelin Yilmaz et al. “Minimum information about a marker gene sequence (MIMARKS) and minimum information about any (x) sequence (MIxS) specifications”. eng. In: *Nature Biotechnology* 29.5 (May 2011), pp. 415–420. ISSN: 1546-1696. DOI: `10.1038/nbt.1823`.

[52] Dawn Field et al. “The minimum information about a genome sequence (MIGS) specification”. eng. In: *Nature Biotechnology* 26.5 (May 2008), pp. 541–547. ISSN: 1546-1696. DOI: `10.1038/nbt1360`.

[53] Vivien G. Dugan et al. “Standardized metadata for human pathogen/vector genomic sequences”. eng. In: *PloS One* 9.6 (2014), e99979. ISSN: 1932-6203. DOI: `10.1371/journal.pone.0099979`.

[54] Barry Smith et al. “The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration”. eng. In: *Nature Biotechnology* 25.11 (Nov. 2007), pp. 1251–1255. ISSN: 1087-0156. DOI: `10.1038/nbt1346`.

[55] Benjamin M. Good and Andrew I. Su. “Crowdsourcing for bioinformatics”. In: *Bioinformatics* 29.16 (Aug. 2013), pp. 1925–1933. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/btt333`. URL: `https://doi.org/10.1093/bioinformatics/btt333` (visited on 06/05/2022).

[56] Ryan Connor et al. “NCBI’s Virus Discovery Hackathon: Engaging Research Communities to Identify Cloud Infrastructure Requirements”. In: *Genes* 10.9 (Sept. 2019), p. 714. ISSN: 2073-4425. DOI: `10.3390/genes10090714`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6771016/` (visited on 06/05/2022).

[57] Jeaime Powell et al. “Organizing online hackathons for newcomers to a scientific community – Lessons learned from two events”. In: *Sixth Annual International Conference on Game Jams, Hackathons, and Game Creation Events*. ICGJ 2021. New York, NY, USA: Association for Computing Machinery, 2021, pp. 78–82. ISBN: 978-1-4503-8417-9. DOI: `10.1145/3472688.3472700`. URL: `https://doi.org/10.1145/3472688.3472700` (visited on 06/05/2022).

Chapter 10

Conclusion

The work I hereby present aims to evaluate the use of bioinformatics methods for the analysis of metagenomic data to allow the rapid identification, virulence analysis and antimicrobial susceptibility prediction of pathogens with clinical relevance. SMg still presents as a promising methodology to obtain very fast results for the identification of pathogens and their virulence and resistance properties directly from samples, without the need for culture. To be accredited and used in clinical settings, this approach must be standardised and the statistical metrics used to analyse and report the data must be validated.

The impact and applicability of SMg in clinical microbiology, including both a diagnosis and surveillance and infection prevention, has been assessed, with the unique challenges of both highlighted. For diagnostics, the biggest drawback of this methodology is the extremely cost-ineffective negative results, allied with the very high sensitivity that translates into several false-positive results. It has been used as a last-resort diagnostic technique [1–4], with great success when classical approaches fail to detect the causative agent of a disease. Chapter 2 aimed to evaluate SMg approaches in a collection of varied samples, with particular interest in the bioinformatic pipeline applied, and how it affected the results obtained in a potential diagnosis setting. Most pathogens identified by culture were also identified through metagenomics, but substantial differences were noted between the taxonomic classification tools. In surveillance and infection prevention, SMg has also been successfully applied [5–7], with programs existing relying on this methodology for the monitoring of emerging or of global interest pathogens [8]. Chapter 3, the novel detection of an *mcr*-5 gene, named *mcr*-5.4, is reported, from a concentrated water sample.

The accurate identification of pathogens of interest from SMg remains one of the biggest challenges when analysing this type of data. A hybrid approach of read mapping and *de novo* assembly methods sometimes proved the only way to successfully recover sequences of interest with enough quality for genome reconstruction. In Chapter 4, both approaches were employed to provide, in a single methodological step, identification and characterization of a whole viral genome at the nucleotide level. Furthermore, Chapter 5 highlighted that no assembler stood out as an undisputed all-purpose choice for short-read metagenomic prokaryote genome assembly, hence efforts are still needed to further improve metagenomic assembly performance.

A strong focus on the standardisation and reproducibility of the results obtained, with the employment of new technologies to do so, such as container software and workflow managers, is of the uttermost necessity for the SMg data analysis solutions. Transparency, scalability, and ease of installation are key-stones, regardless of the tools chosen. The solutions adopted throughout this work, such the use of docker, nextflow and conda, allied to clear and easy to follow documentation, aim to lower the barrier of entry when performing detailed analysis that are complex and computationally expensive in their nature. But most importantly, the production of intuitive, responsive and easy-to-follow reports, allowing the summary of key results, as well as the detailed exploration of the resulting data, by stakeholders, be it bioinformatic personnel or experts in the given area of expertise, represent the single most important contribution to lowering the barrier between who produces the data and who has the capacity to make informed decisions based on that data.

As public health laboratories expand their genomic sequencing and bioinformatics capacity for the surveillance of different pathogens, labs must carry out robust validation, training, and optimisation of wet- and dry-lab procedures. Despite this richness in genomic information, the same is not observed for the contextual information that accompanies it. This contextual information, composed of metadata Standardisation of harmonisable, publicly available community standards is ultimately what will allow the transmission of information through various stakeholders, containing the minimum information required for it to be understandable and actionable across domains.

10.1 Future perspectives

Despite the results presented in this work, the future of SMg still looks promising. Regardless, for it to pass from the last resort solution when traditional methods fail to a standard in clinical microbiology, some issues still need to be addressed.

One of the biggest hinders to the application of SMg in the clinic is the significant cost of negative results, allied with a very high sensitivity but not as high specificity, being very prone to contamination leading to false positive results. Therefore, part of the work to improve this technology passes through the improvement in the bench. Host depletion steps still have a long way to produce satisfactory results for diagnosis purposes, without altering the proportions of organisms in a community. Although traditional methods require a very significant amount of man-hours, sequencing requires a different, more specialised set of skills to obtain quality results. Additionally, the time gained in the results attainment is offset by the increased cost.

Regarding bioinformatic analysis of SMg data, the lack of golden standards presents the biggest hindrance. But as new methods keep being developed, fuelled by the extreme and exciting interest of the community for this methodology, and more importantly, as current methods continue to be validated, and their results openly shared, in a faindable and FAIRer way, we slowly but surely walk towards the standardisation of methodologies.

I'm as excited with the potential of SMg for clinical microbioly, both in the diagnostics and public health and surveillance branches, as I was when I first started this project. The potentials are staggering, albeit currently it's still very much an experimental and/or last resort approach. As methodologies continue to improve, sequencing will become ubiquitous in every diagnostics and public health laboratory. The transition between academic applications to being a viable solution in hospitals and health reference laboratories might be slow, or at least slower than what was once expected, but it's an inevitability that we should not only be prepared for but also excited about.

10.2 References

[1] Wenfang He et al. "Case Report: Therapeutic Strategy With Delayed Debridement for Culture-Negative Invasive Group A Streptococcal Infections Diagnosed by Metagenomic Next-Generation Sequencing". In: *Frontiers in Public Health* 10 (2022). ISSN: 2296-2565. URL: `https://www.frontiersin.org/articles/10.3389/fpubh.2022.899077` (visited on 10/20/2022).

[2] Prakhar Vijayvargiya et al. "Application of metagenomic shotgun sequencing to detect vector-borne pathogens in clinical blood samples". en. In: *PLOS ONE* 14.10 (Feb. 2019). Publisher: Public Library of Science, e0222915. ISSN: 1932-6203. DOI: `10.1371/journal.pone.0222915`. URL: `https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0222915` (visited on 02/28/2022).

[3] Adriana Sanabria et al. "Shotgun-Metagenomics on Positive Blood Culture Bottles Inoculated With Prosthetic Joint Tissue: A Proof of Concept Study". In: *Frontiers in Microbiology* 11 (2020). ISSN: 1664-302X. URL: `https://www.frontiersin.org/article/10.3389/fmicb.2020.01687` (visited on 02/28/2022).

[4] Shota Hirakata et al. "The application of shotgun metagenomics to the diagnosis of granulomatous amoebic encephalitis due to Balamuthia mandrillaris: a case report". In: *BMC Neurology* 21.1 (2021), p. 392. ISSN: 1471-2377. DOI: `10.1186/s12883-021-02418-y`. URL: `https://doi.org/10.1186/s12883-021-02418-y` (visited on 02/28/2022).

[5] Nicholas J. Loman et al. "A Culture-Independent Sequence-Based Metagenomics Approach to the Investigation of an Outbreak of Shiga-Toxigenic Escherichia coli O104:H4". In: *JAMA* 309.14 (2013), pp. 1502–1510. ISSN: 0098-7484. DOI: `10.1001/jama.2013.3231`. URL: `https://doi.org/10.1001/jama.2013.3231` (visited on 02/28/2022).

[6] Andrew D. Huang et al. "Metagenomics of Two Severe Foodborne Outbreaks Provides Diagnostic Signatures and Signs of Coinfection Not Attainable by Traditional Methods". In: *Applied and Environmental Microbiology* 83.3 (Jan. 2017), e02577–16. ISSN: 0099-2240. DOI: `10.1128/AEM.02577-16`. URL: `https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5244306/` (visited on 02/28/2022).

[7] Sijia Li et al. "Microbiome Profiling Using Shotgun Metagenomic Sequencing Identified Unique Microorganisms in COVID-19 Patients With Altered Gut Microbiota". In: *Frontiers in Microbiology* 12 (2021). ISSN: 1664-302X. URL: `https://www.frontiersin.org/article/10.3389/fmicb.2021.712081` (visited on 02/28/2022).

[8] Karrie K. K. Ko, Kern Rei Chng, and Niranjan Nagarajan. “Metagenomics-enabled microbial surveillance”. en. In: *Nature Microbiology* 7.4 (Apr. 2022). Number: 4 Publisher: Nature Publishing Group, pp. 486–496. ISSN: 2058-5276. DOI: `10.1038/s41564-022-01089-w`. URL: `https://www.nature.com/articles/s41564-022-01089-w` (visited on 10/20/2022).

Chapter 11

Appendix

www.nature.com/scientificreports

SCIENTIFIC REPORTS

Corrected: Author Correction

OPEN

Received: 12 January 2018
Accepted: 28 August 2018

Critical steps in clinical shotgun metagenomics for the concomitant detection and typing of microbial pathogens

Natacha Couto[1], Leonard Schuele[1,2], Erwin C. Raangs[1], Miguel P. Machado[3], Catarina I. Mendes[1,3], Tiago F. Jesus[3], Monika Chlebowicz[1], Sigrid Rosema[1], Mário Ramirez[3], João A. Carriço[3], Ingo B. Autenrieth[2], Alex W. Friedrich[1], Silke Peter[2] & John W. Rossen[1]

High throughput sequencing has been proposed as a one-stop solution for diagnostics and molecular typing directly from patient samples, allowing timely and appropriate implementation of measures for treatment, infection prevention and control. However, it is unclear how the variety of available methods impacts the end results. We applied shotgun metagenomics on diverse types of patient samples using three different methods to deplete human DNA prior to DNA extraction. Libraries were prepared and sequenced with Illumina chemistry. Data was analyzed using methods likely to be available in clinical microbiology laboratories using genomics. The results of microbial identification were compared to standard culture-based microbiological methods. On average, 75% of the reads corresponded to human DNA, being a major determinant in the analysis outcome. None of the kits was clearly superior suggesting that the initial ratio between host and microbial DNA or other sample characteristics were the major determinants of the proportion of microbial reads. Most pathogens identified by culture were also identified through metagenomics, but substantial differences were noted between the taxonomic classification tools. In two cases the high number of human reads resulted in insufficient sequencing depth of bacterial DNA for identification. In three samples, we could infer the probable multilocus sequence type of the most abundant species. The tools and databases used for taxonomic classification and antimicrobial resistance identification had a key impact on the results, recommending that efforts need to be aimed at standardization of the analysis methods if metagenomics is to be used routinely in clinical microbiology.

Classical microbial culture is still considered the gold standard in medical microbiology. Several molecular detection techniques have been implemented but these are generally geared towards specific pathogens (e.g. specific RT-PCR or microarrays). Even when unbiased molecular approaches are used, such as 16 S/18 S rRNA gene sequencing, these do not provide all the information that can be obtained by culturing, e.g., antimicrobial susceptibility and molecular typing information. However, microbial culture is laborious and time-consuming and new methods are needed to replace it. Ideally, a single method should provide rapid identification and characterization of clinically relevant pathogens directly from a sample in order to guide therapy, predict potential treatment failures and to reveal possible transmission events.

Shotgun metagenomics (SMg) is a culture-independent technique that provides valuable information not only at the identification level, but also at the level of molecular characterization. Studies have shown that it has added value in terms of detection sensitivity and personalized treatment in clinical microbiology, when identifying bacteria[1,2] or viruses[3]. Indeed Gyarmati *et al.*, 2016[4], used a sequence-based metagenomics approach directly

[1]University of Groningen, University Medical Center Groningen, Department of Medical Microbiology, Groningen, The Netherlands. [2]Institute of Medical Microbiology and Hygiene, University of Tübingen, Tübingen, Germany. [3]Instituto de Microbiologia, Instituto de Medicina Molecular, Faculdade de Medicina, Universidade de Lisboa, Lisbon, Portugal. Correspondence and requests for materials should be addressed to N.C. (email: n.monge.gomes.do.couto@umcg.nl)

	Sample 1	**Sample 2**	**Sample 3**	**Sample 4**	**Sample 5**	**Sample 6**	**Sample 7**	**Sample 8**	**Sample 9**	**Sample 10**	**Negative control**
Sample type	Peritoneal fluid	Pus (abscess)	Synovial fluid	Synovial fluid	Pus (abscess)	Pus (empyema)	Pus (empyema)	Bone biopsy	Pus (abscess)	Sputum	Water
DNA extraction method	Ultra-Deep Microbiome Prep (Molzym)	Ultra-Deep Microbiome Prep (Molzym)	Ultra-Deep Microbiome Prep (Molzym)	Ultra-Deep Microbiome Prep (Molzym)	Ultra-Deep Microbiome Prep (Molzym)	QIAamp DNA Microbiome Kit (Qiagen)	QIAamp DNA Microbiome Kit (Qiagen)	Micro-DX$^{\mathrm{TM}}$ (Molzym)	Micro-DX$^{\mathrm{TM}}$ (Molzym)	Micro-DX$^{\mathrm{TM}}$ (Molzym)	QIAamp DNA Microbiome Kit (Qiagen)
Total number of reads	5,892,978	9,603,346	8,615,810	6,078,166	8,368,930	2,912,802	1,486,700	6,534,866	6,173,132	7,596,836	1,730,738
Mapped reads against hg19	5,249,063 (89.2%)	7,828.746 (81.6%)	8,254,594 (95.9%)	6,015,945 (99.0%)	309,588 (3.7%)	2,877,066 (98.8%)	922,932 (62.2%)	229,149 (3.5%)	6,081,612 (98.5%)	7,337,832 (96.7%)	1,706,861 (98.9%)
Unmapped reads	632,951 (10.8%)	1,770,558 (18.4%)	355,200 (4.1%)	61,099 (1.0%)	8,052,272 (96.3%)	34,506 (1.1%)	561,772 (37.8%)	6,303,803 (96.5%)	89,922 (1.5%)	235,520 (3.3%)	19,805 (1.2%)

Table 1. Characteristics of the samples and mapping of trimmed reads against a human genome hg19 (%) using CLC Genomics Workbench v10.0.1.

from blood to detect non-culturable, difficult-to-culture and non-bacterial pathogens. The authors were able, through SMg, to detect viral and fungal pathogens together with bacteria, which had not been detected through classical microbiology. Additionally, SMg can be used for infection prevention, having the potential to identify transmission events directly from clinical samples[5]. For example, SMg was proven valuable for the identification of inter-host nucleotide variations occurring after direct transmission of noroviruses causing gastroenteritis[5]. Hasman and colleagues (2014)[1] were able to identify urinary pathogens directly from urine, as well as antimicrobial resistant genes compatible with the resistant phenotype determined through antimicrobial susceptibility testing. They also identified almost perfect phylogenetic matches between whole-genome sequence (WGS) data obtained by metagenomics and WGS of pure isolates.

Despite the promise of SMg of becoming a one-stop solution in clinical microbiology, SMg still has several challenges to overcome. One of the greatest challenges is the choice of the extraction and sequencing protocols, as well of the type of controls[6]. The extraction protocol should efficiently and specifically isolate microbial DNA/RNA, while removing the host DNA/RNA[7]. However, the variety of clinical samples used in the diagnosis of distinct types of infection (e.g. tissues versus fluids), poses a serious challenge for standardization, an essential step if these methods are to be used by routine diagnostic laboratories. The sequencing protocol is also dependent on the pathogens of interest (e.g. bacteria versus viruses), sequencing strategy (DNA and/or RNA), required turnaround time, sequencing depth and error tolerance[6]. The use of defined controls is necessary for validation of each experiment and these should be adapted for every type of infection and sample type and should consist of a combination of known positive specimens, pathogen-negative patient specimens and pathogen-negative patient specimens spiked with live microorganisms or pure DNA[6].

Another potential challenge are the metagenomics analysis tools. Recent studies have evaluated the different SMg sequence classification methods[8]. These use different methodologies for classification: sequence similarity-based methods, sequence composition-based methods and hybrid methods[8]. They differ not only in the algorithms for detecting the microorganisms present, but also in the databases used. This high variability leads to different results, not only at the microorganism classification level but also when evaluating the relative abundance of these pathogens[8]. A recent study evaluated the accuracy of 38 bioinformatics methods using both *in silico* and *in vitro* generated mock bacterial communities. Dozens to hundreds of species were falsely predicted by the most popular software, and no software clearly outperformed the others[8]. In the absence of studies comparing the outputs of different analysis methods in clinical samples, users may decide which methods to use based on personal experience with a given tool, availability of the tool in the laboratory or its ease of use. This poses a great challenge when providing reproducible results and creates uncertainty regarding the reliability of the information derived. This is a major barrier to the implementation of SMg approaches in routine clinical microbiology laboratories.

In this study, the aim was to identify the critical steps when using SMg for the identification and characterization of microbial pathogens directly from clinical specimens using methods that are likely to be available in clinical microbiology laboratories wanting to implement genomics for pathogen identification or molecular epidemiology studies. For this purpose, we used three human-DNA depletion kits and evaluated a diverse set of bioinformatics tools (commercial and non-commercial) in order to investigate how well they performed and what the differences would be in terms of taxonomic classification, antimicrobial resistance gene detection and typing directly from patient samples, bypassing culture.

Results

Classical identification. Nine body fluid samples and one tissue sample from 9 different patients were sequenced, including one sample from peritoneal fluid, five from pus (3 abscesses and 2 empyemas), two from synovial fluid of knees with prosthesis, one from sputum and one from a bone biopsy (Table 1). In total 15 different isolates obtained from the 10 samples were considered of possible clinical significance and were selected for species identification and antimicrobial susceptibility testing during routine work up of the samples (Tables 2, 3 and 4). In samples 2 and 3, only one colony-forming unit (CFU) of *Escherichia coli* and *Staphylococcus*

Sample number	Culture result (CFU)[a]	Conventional identification (MALDI-TOF)	WGS-based identification	Shotgun metagenomics		
				Kraken[b]	MIDAS[c]	MetaPhlAn[c]
1	10^3 10^3 10	*E. faecium* *S. haemolyticus* *C. glabrata*	*E. faecium* *S. haemolyticus* —	*E. faecium* (34.6%) *S. haemolyticus* (10.1%) —	*E. faecium* (62.0%) *S. haemolyticus* (28.0%) —	*E. faecium* (66.6%) *S. haemolyticys* (27.7%) —
2	10^3 1 Not determined	*E. avium* *E. coli* Anaerobes	—# —# —#	Not identified* Not identified* Several species (29.5%)	Not identified* Not identified* Several species (100.0%)	Not identified* Not identified* Several species (100.0%)
3	1	*S. epidermidis*	—#	*S. aureus* (0.2%)	Not identified*	Not identified*
4	10^3	*S. aureus*	*S. aureus*	*S. aureus* (0.73%)	*S. aureus* (100%)	*S. aureus* (100%)
5	$\geq 10^5 \geq 10^5$ 10^3 10^3 Not determined 10	*E. coli* *K. oxytoca* *S. anginosus* *E. faecalis* Anaerobes *C. albicans*	*E. coli* *K. oxytoca* —# *E. faecalis* —# —#	*E. coli* (9.7%) *K. oxytoca* (0.5%) *S. anginosus* (0.07%) *E. faecalis* (0.3%) Several species (12.7%) —	*E. coli* (6.5%) *K. oxytoca* (0.3%) *S. anginosus* (0.01%) *E. faecalis* (0.9%) Several species (96.7%) —	*E. coli* (8.5%) *K. oxytoca* (0.3%) *Streptococcus* spp. (0.09%) *E. faecalis* (0.7%) Several species (90.4%) —
6	10^3	*E. faecium*	*E. faecium*	*E. faecium* (0.77%)	Not identified*	Not identified*
7	10^2	*S. aureus*	—#	*S. aureus* (82.9%)	*S. aureus* (100%)	*S. aureus* (100%)
8	10^3	*O. intermedium*	*O. intermedium*	*O. anthropi* (21.3%)	*O. intermedium* (99.4%)	*O. intermedium* (99.1%)
9	10^3	*S. aureus*	*S. aureus*	*S. aureus* (22.9%)	*S. aureus* (100%)	*S. aureus* (100%)
10	10^3	*S. marcescens*	—#	*S. marcescens* (64.7%)	*S. marcescens* (99.1%)	*S. marcescens* (100%)

Table 2. Microorganisms identified by conventional methods, WGS and using shotgun metagenomics and the taxonomic classification methods in Unix. [a]The number of colonies of a given species was estimated from the number of colonies with the same morphology on the same plate; [b]The relative abundance is calculated using total number of reads as denominator; [c]The relative abundance is calculated with the total number of classified reads as denominator; [d]miniKraken database was used; #Although there was a laboratory identification, no isolates were available for WGS; *No reads matched that specific pathogen, not even at the genus level.

epidermidis, respectively, was detected after 48 hours of incubation. In samples 2 and 5, the anaerobic cultures were mixed to such an extent, that no further characterization of the colonies was performed, and the results were reported as anaerobic mixed culture.

Antimicrobial susceptibility testing, revealed three isolates to be fully susceptible, while the others were resistant to at least one antimicrobial. Two isolates, one *Staphylococcus haemolyticus* and one *S. epidermidis* were oxacillin-resistant and positive in the cefoxitin test (Vitek 2).

There was fungal growth in 2 samples (1 and 5) that included two *Candida* species (one *Candida glabrata* and one *Candida albicans*). The different bacterial and fungal species identified in each sample are shown in Tables 2, 3 and 4.

Comparison of standard procedures and shotgun metagenomics for the identification of clinically relevant pathogens.

The tools used for taxonomic classification are shown in Fig. 1. The total number of reads and the total number of reads mapped against the human genome (hg19) varied between samples, ranging from 3.5% to 98.9% (Table 1). The abundance of human reads was not determined by the type of sample but was probably influenced by individual characteristics of each sample and the success of the methods used in depleting the human DNA. We identified the microorganisms present using different taxonomical methods, including three Unix-based tools (Kraken, Metaphlan2 and MIDAS), web-based tools including both commercial and freely available solutions (BaseSpace, Taxonomer and CosmosID) and one commercial approach having a graphical interface (CLC Genomics Workbench v10.0.1). The taxonomic classification results for each sample are presented in Tables 2, 3 and 4. In 8 samples, all the microorganisms identified by classical culture were also identified through metagenomics (using at least one method). In sample 2, two of the bacterial species identified by classical culture, i.e., *E. coli* and one *Enterococcus avium* were not identified through shotgun metagenomics and in sample 3 there was no concordance between the results of MALDI-TOF and the taxonomical classification methods at the species level (Tables 2, 3 and 4). We identified *Ochrobactrum intermedium* in the negative control, but in low amounts (1.0% of the reads mapped to the reference genome with the accession number NZ_ACQA01000002 and only 1.4% of the reference genome was covered). The sensitivity and positive predictive value of each classification method is shown in Table 5.

Determination of antimicrobial resistance.

Metagenomics provides other sequence information in addition to pathogen detection. We determined the presence of antimicrobial-resistance genes in the SMg sequence data and compared the results with those obtained from WGS and phenotypic resistance testing (Table 6).

Antimicrobial resistance genes found with CLC Genomics Workbench and ReMatCh in samples 1, 7 and 9 correlated well with phenotypic results. However, in the other 7 samples, not all antimicrobial resistance genes that could explain the phenotypic profile were identified. In addition, in samples 2, 5, 7 and 10, ReMatCh detected different resistance genes compared to those reported by CLC Genomics Workbench (Table 6). Some of these differences (genes *norA*, *bla*SST-1, *fusA*) were due to slight differences in the databases used, however, the other resistance genes were present in both databases. Interestingly, in two samples (samples 2 and 5), we were able to identify several

Sample number	Culture result (CFU)[a]	Conventional identification (MALDI-TOF)	WGS-based identification	Shotgun metagenomics	
				Taxonomic Profiling (CLC)[b]	**Best match with K-mer spectra (CLC)[c]**
1	10^3 10^3 10	*E. faecium* *S. haemolyticus* *C. glabrata*	*E. faecium* *S. haemolyticus* —	*E. faecium* (71%) *S. haemolyticus* (24%) *C. glabrata* (100%)	*E. faecium* (41.4%) *S. haemolyticus* (13.8%) *C. glabrata* (0.5%)
2	10^3 1 Not determined	*E. avium* *E. coli* Anaerobes	—[#] —[#] —[#]	Not identified* Not identified* Several species (97%)	Not identified* Not identified* Several species (13.2%)
3	1	*S. epidermidis*	—[#]	Not identified*	*S. aureus* (4%)
4	10^3	*S. aureus*	*S. aureus*	Not identified*	*S. aureus* (9.7%)
5	$\geq 10^5 \geq 10^5$ 10^3 10^3 Not determined 10	*E. coli* *K. oxytoca* *S. anginosus* *E. faecalis* Anaerobes *C. albicans*	*E. coli* *K. oxytoca* —[#] *E. faecalis* —[#] —[#]	*E. coli* (25%) *K. michiganensis* (0.3%) Not identified* *E. faecalis* (2%) Several species (70.0%) Not identified*	*E. coli* (11.5%) Not identified* Not identified* *E. faecalis* (0.6%) Not identified* *C. albicans* (<0.05%)
6	10^3	*E. faecium*	*E. faecium*	Not identified*	*E. faecium* (4.0%)
7	10^2	*S. aureus*	—[#]	*S. aureus* (100%)	*S. aureus* (95.5%)
8	10^3	*O. intermedium*	*O. intermedium*	*O. intermedium* (86.0%)	*O. intermedium* (91.2%)
9	10^3	*S. aureus*	*S. aureus*	*S. aureus* (100%)	*S. aureus* (81.2%)
10	10^3	*S. marcescens*	—[#]	*S. marscescens* (100%)	*S. marcescens* (79.7%)

Table 3. Microorganisms identified by conventional methods, WGS and using shotgun metagenomics and the taxonomic classification methods in CLC Genomics Workbench. [a]The number of colonies of a given species was estimated from the number of colonies with the same morphology on the same plate; [b]The relative abundance is calculated with the total number of classified reads as denominator; [c]Based on the Output Quality Report; [#]Although there was a laboratory identification, no isolates were available for WGS; *No reads matched that specific pathogen, not even at the genus level.

antimicrobial resistance genes usually found in anaerobic bacteria. These were not reported by classical microbiology methods, probably because they were not considered relevant pathogens worthy of subsequent susceptibility study (mixed anaerobic culture).

The SEAR app in BaseSpace (the only one available for antimicrobial resistance gene detection) crashed several times, although we performed the analysis repeatedly, using different parameters. We were only able to get results in 3 samples, with no resistance genes detected.

MLST and wgMLST analysis. In three cases when SMg data covered ≥93% of the genome we were able to identify the ST, which corresponded to the one found using WGS of the isolated bacteria using CLC Genomics Workbench (n = 2) and metaMLST (n = 1). These results are summarized in Table 7. Assembled genomes and metagenomes, were compared by wgMLST analysis using Ridom SeqSphere+. Figure 2 shows examples of the allele difference between the genomes obtained through WGS versus the genomes obtained through shotgun metagenomics.

Characterization of mobile genetic elements. Two different approaches, i.e. CLC Genomics Workbench and Bowtie2 were used to identify plasmids present in the sequence data. Both approaches used mapping of sequences against the same plasmid database. Since some plasmids present in the database are very similar and sequence reads may be mapped to more than one plasmid, we used the pATLAS tool, which provides an overview of the nodes (representing plasmid sequences) and links between plasmids (which connect similar plasmids), to enable the visualization of the plasmids identified (Fig. 3). A color gradient indicates the sequence coverage of the plasmids. In most cases, the same plasmids were identified by both approaches, with some small differences in sequence coverage. When comparing the plasmids identified in the SMg dataset versus the WGS data, most of the plasmids were also detected in the isolates (an example is shown in Fig. 4). However, some plasmids were not identified in any of the isolated bacteria and were probably residing in low-abundant species.

Discussion

This study evaluated the suitability of SMg for the microbiological diagnosis and (patho- and epi-) typing of microorganisms directly from real patient samples. The whole procedure took between 48–54 hours to complete, which is shorter than culture-based methods if one includes typing. However, the amount of information derived from SMg in most cases, did not overcome the necessity for pathogen isolation and subsequent (phenotypic and genotypic) typing, which can take up to 1–2 weeks (particularly in slow-growing organisms). Nevertheless, SMg can help guide antimicrobial therapy and can be helpful in cases where there is a suspicion of transmission and there is a need to quickly determine the genetic relationship between pathogens, although the success of SMg in individual patient samples can be highly variable, as reported here.

Different bioinformatics pipelines were evaluated to identify potential differences between them and identify those which could provide the clinical microbiologist with the maximum of relevant and accurate information. In terms of microbial identification, in both Unix and web-based approaches we would recommend MetaPhlAn, since it has good sensitivity and a good positive predictive value (PPV). The Find Best Match K-mer Spectra tool

				Shotgun metagenomics				
Sample number	**Culture result (CFU)[a]**	**Conventional identification (MALDI-TOF)**	**WGS-based identification**	**Genius (Basespace)[c]**	**Kraken (Basespace)[c,d]**	**MetaPhlAn (Basespace)[c]**	**Taxonomer (Utah)[b,e]**	**Cosmos ID[a]**
1	10^3 10^3 10	*E. faecium* *S. haemolyticus* *C. glabrata*	*E. faecium* *S. haemolyticus* —	*E. faecium* (14.4%) *S. haemolyticus* (55.8%) —	*E. faecium* (25.0%) *S. haemolyticus* (20.1%) —	*E. faecium* (65.1%) *S. haemolyticys* (30.4%) —	*E. faecium* (22.9%) *S. haemolyticus* (20.1%) Not identified*	*E. faecium* (50.3%) *S. haemolyticus* (22.1%) *C. glabrata* (88.6%)
2	10^3 1 Not determined	*E. avium* *E. coli* Anaerobes	—[#] —[#] —[#]	Not identified* Not identified* Several species (94.0%)	Not identified* Not identified* Several species (27.0%)	Not identified* Not identified* Several species (54.2%)	Not identified* Not identified* Several species (14.2%)	Not identified* Not identified* Several species (100%)
3	1	*S. epidermidis*	—[#]	*S. aureus* (100%)	*S. aureus* (0.1%)	Not identified*	*S. pseudintermedius* (3.4%)	Not identified*
4	10^3	*S. aureus*	*S. aureus*	*S. aureus* (100%)	*S. aureus* (0.3%)	*S. aureus* (100%)	*S. aureus* (8.3%)	*S. aureus* (100%)
5	$\geq 10^5 \geq 10^5$ 10^3 10^3 Not determined 10	*E. coli* *K. oxytoca* *S. anginosus* *E. faecalis* Anaerobes *C. albicans*	*E. coli* *K. oxytoca* —[#] *E. faecalis* —[#] —[#]	*E. coli* (0.4%) Not identified* *S. anginosus* (0.03%) *E. faecalis* (0.8%) Several species (45.0%) —	*E. coli* (10.2%) *K. oxytoca* (0.5%) *S. anginosus* (0.4%) *E. faecalis* (0.3%) Several species (8.0%) —	*E. coli* (7.0%) *K. pneumoniae* (0.01%) *S. anginosus* (0.3%) *E. faecalis* (0.7%) Several species (89.1%) —	*E. coli* (3.6%) *K. michiganensis* (0.1%) *S. anginosus* (0.1%) *E. faecalis* (0.1%) Several species (60.3%) —	*E. coli* (7.6%) *K. oxytoca* (1.7%) *S. anginosus* (0.09%) *E. faecalis* (3.7%) Several species (86.2%) Not identified*
6	10^3	*E. faecium*	*E. faecium*	*E. faecium* (4.2%)	*E. faecium* (14.8%)	*E. faecium* (5.5%)	*E. faecium* (1.4%)	*E. faecium* (4.1%)
7	10^2	*S. aureus*	—[#]	*S. aureus* (100%)	*S. aureus* (93.8%)	*S. aureus* (100%)	*S. aureus* (14.2%)	*S. aureus* (100%)
8	10^3	*O. intermedium*	*O. intermedium*	*O. intermedium* (100%)	*O. nthropic* (88.9%)	*O. intermedium* (99.8%)	*O. intermedium* (13.1%)	*O. intermedium* (49.5%)
9	10^3	*S. aureus*	*S. aureus*	*S. aureus* (100%)	*S. aureus* (99.5%)	*S. aureus* (100%)	*S. aureus* (12.7%)	*S. aureus* (100%)
10	10^3	*S. marcescens*	—[#]	*S. marcescens* (32.5%)	*S. marcescens* (94.8%)	*Serratia* spp. (100%)	*S. marcescens* (1.4%)	*S. marscescens* (38.4%)

Table 4. Microorganisms identified by conventional methods, WGS and using shotgun metagenomics and the taxonomic classification methods in webpages (BaseSpace, Taxonomer and CosmosID). [a]The number of colonies of a given species was estimated from the number of colonies with the same morphology on the same plate; [b]The relative abundance is calculated using total number of reads as denominator; [c]The relative abundance is calculated with the total number of classified reads as denominator; [d]miniKraken database was used; [e]Full Analysis mode was used; [#]Although there was a laboratory identification, no isolates were available for WGS; *No reads matched that specific pathogen, not even at the genus level.

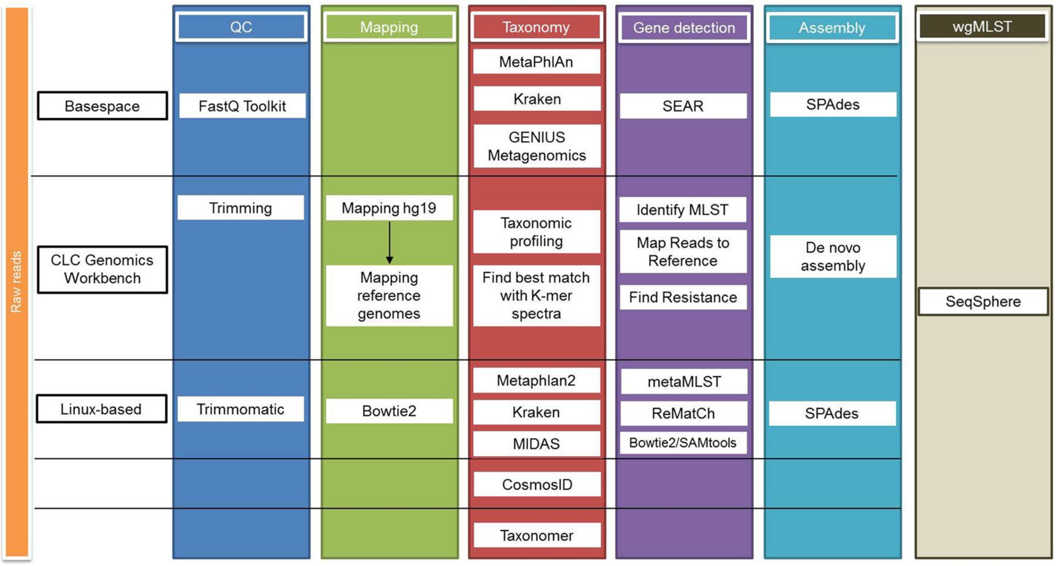


Figure 1. Scheme of the bioinformatic analysis of the metagenomics samples.

should be used in the context of the CLC Genomics Workbench, since it had a higher sensitivity and PPV compared to the Taxonomic Profiling tool.

In a clinical setting, a combination of high sensitivity and high PPV of any new method is key. Popular software designed for bacterial identification can predict dozens to hundreds of species in *in vitro* generated bacterial communities of known composition[8]. We observed the same when using Kraken and Taxonomer when comparing to

Method	Total number of bacteria identified[a]	True positives[a]	False positives	False negatives	Sensitivity (%)	PPV (%)
Culture/MALDI-TOF	9	9	0	0	100%	100%
MetaPhlAn (BaseSpace)	16	7	9	2	78%	44%
Genius (BaseSpace)	35	8	27	1	89%	23%
Kraken (BaseSpace)	959	7	952	2	78%	1%
Taxonomer (Full Analysis)	4649	8	4641	1	89%	0%
CosmosID	35	8	27	1	89%	23%
Taxonomic Profiling (CLC Genomics Workbench v10.0.1)	17	6	11	3	67%	35%
Best match K-mer spectra (CLC Genomics Workbench v10.0.1)	12	8	4	1	89%	67%
Kraken (Unix)	198	7	191	2	78%	4%
MetaPhlAn2 (Unix)	15	7	6	4	75%	75%
MIDAS (Unix)	34	7	26	2	88%	50%

Table 5. Performance of the different taxonomic classification methods for each sample. Sensitivity and positive predictive value were calculated using culture/MALDI-TOF as standards. [a]Excluding the samples with non-identified anaerobic bacteria (Samples 2 and 5). Abbreviations: PPV, positive predictive value.

culture-based methods. For both Kraken and Taxonomer, relative abundance cut-off values may be required to limit the number of species identified. However, which cut-off values should be used are a matter of debate, since in some cases, even if applying a cut-off value as low as 1.0% (comparable to what was found in the negative control) would have resulted in decreased sensitivity (e.g. the *Streptococcus anginosus* identified by culture in Sample 5 would have been disregarded). The methods that employ several parameters to infer microbial identification are superior, because they not only rely on the relative abundance of bacterial species, but also on the genome coverage and on the proportion of the genome that was covered. On the other hand, in some cases SMg may be more sensitive than culture in identifying pathogens, reflecting the higher sensitivity or the capacity to detect bacterial species which are non-culturable in the conditions used or that are no longer culturable, such as due to prior antimicrobial therapy. In such cases, other methods like 16 S rRNA gene sequencing or the recently described 16S-23S rRNA-coding region sequencing method[9] may be used for discrepancy analyses. However, here we decided to use culture-based methods as the gold standard, since this is still the method of choice in clinical microbiology.

One limitation of this study was the exclusion of culture-negative samples and thus their inclusion would have affected the calculation of the specificity values. However, as mentioned above, culture-negative samples do not necessarily mean that the samples are pathogen-free, but it might only reflect the low sensitivity or capacity of culture-based methods to detect non-culturable bacterial species. As with other (molecular) methods, several controls should be included to validate the obtained results, including a negative control. In our negative control, we detected an *O. intermedium* strain, although with only 1.0% of the reads mapping to the reference genome and covering only 1.4% of the reference genome (accession number NZ_ACQA01000002). These results may be due to contamination during library preparation (e.g. sample-to-sample contamination prior to indexing), the result of sequencing artefacts (e.g. demultiplexing errors), or to incorrect classification during data analysis (e.g. highly similar regions)[3]. Our samples and sequencing libraries were handled in laminar flow cabinets; however, we cannot also exclude the possibility of contamination. Furthermore, the reagents used may also be or become contaminated with DNA leading the detection of these contaminating species, something that has been described previously[7]. This poses a challenge for interpretation, because some positive samples also had very low numbers of reads for some pathogens (<1%). When approaching this limit of detection, small numbers of pathogen reads will be difficult to interpret, as they can represent true-positives with low abundance in the sample, or artifacts such as contamination during library preparation[3].

In terms of antimicrobial resistance gene detection, ReMatCh (Unix) and the CLC Genomics Workbench Find Resistance tool gave comparable results. Since ReMatCh (Unix) performs the analysis at the read level, while CLC Genomics Workbench performs it at the contig level, we suggest that both strategies should be employed in parallel when looking for antimicrobial resistance genes. It is also important to emphasize that the contig-level approach employed by CLC Genomics Workbench may give negative results if the sequence coverage is set to a high percentage (e.g. above 80%). This is due to the assembly method, which may split the antimicrobial resistance genes into different contigs, when the number of reads is too low. This phenomenon was observed in Sample 1, for the *aac(6')-aph(2")* gene, which was split into three different contigs, each part corresponding to less than 40% of the gene. Only when applying a cut-off value of ≥20% for sequence coverage could we identify all three parts of the gene, which in total corresponded to 89% of the entire sequence. Finally, it is important to point out that the ResFinder database (used here), and other databases, focus on acquired genes, not including chromosomal point mutations resulting in antimicrobial resistance. However, a recently developed tool, PointFinder, was added to ResFinder for the detection of chromosomal point mutations associated with antimicrobial resistance[10] and an updated database will be available soon.

Another challenge is to infer where these antimicrobial resistance genes are located (chromosome or plasmid). The study of mobile genetic elements, including plasmids, carrying antimicrobial resistance genes present in clinical samples is important to predict possible treatment failures and the spread of resistance within and across bacterial species. When performing bacterial isolation followed by WGS, information on polymicrobial infections

Sample number	Conventional identification (MALDI-TOF)	Conventional susceptibility testing (VITEK 2)[b]	WGS CLC Genomics Workbench	Shotgun metagenomics	
				ReMatCh (Unix)	CLC Genomics Workbench[a]
1	*E. faecium* *S. haemolyticus*	LEV, ERY, CLI OXA, GEN, CIP, FOS, ERY, CLI	*erm(B), msr(C), ant(6')-Ia, aph(3')-III, dfrG* *blaZ, mecA, ant(6')-Ia, aph(3')-III, aac(6')-aph(2"), erm(C), mph(C), msr(A), dfrG*	*erm(B), msr(C), ant(6')-Ia, aph(3')-III, aac(6')-aph(2"), blaZ, mecA, erm(C), mph(C), msr(A), dfrG*	*erm(B), msr(C), ant(6')-Ia, aph(3')-III, aac(6')-aph(2"), blaZ, mecA, erm(C), mph(C), msr(A), dfrG*
2	*E. avium* *E. coli* Anaerobes	DOX, CLI susceptible —	—[#] —[#] —[#]	Not detected Not detected *catS, lnu(D), lsa(C), cepA-44, tet(Q)*	Not detected Not detected *catS, lnu(D), lsa(C), cepA-44, tet(Q), fusA*
3	*S. epidermidis*	OXA, GEN, TEC, FUS, CIP, ERY, CLI	—[#]	Not detected	Not detected
4	*S. aureus*	PEN, ERY	*blaZ, spc, erm(A)*	Not detected	Not detected
5	*E. coli* *K. oxytoca* *S. anginosus* *E. faecalis* Anaerobes	susceptible AMX susceptible DOX, CLI —	—[#] *blaOXY-1–3* —[#] *tet(M), lsa(A)* —[#]	— Not detected — *tet(M)* *cfxA4, tet(Q)*	— Not detected — *tet(O)* *cfxA4, tet(Q)*
6	*E. faecium*	PEN, AMX, CFX, IMP, GENhl, STRhl, LEV, ERY, CLI, AMP/SUL	*erm(B), msr(C), ant(6')-Ia, aph(3')-III, aac(6')-aph(2"), dfrG*	Not detected	Not detected
7	*S. aureus*	PEN	*blaZ*	*blaZ, norA*	*blaZ*
8	*O. intermedium*	AMX, PIP/TAZ, CFX, CFT, CTZ, IMP, FOX, TOB, FOS, NIT, TMP	*bla*OCH-2	*bla*OCH-5	*bla*OCH-2
9	*S. aureus*	PEN	—[#]	*blaZ*	*blaZ*
10	*S. marcescens*	AMX, AMC, CFX, FOX, NIT, POL	—[#]	*blaSST-1, tet(41), oqxB, aac(6')-Ic*	*tet(41), oqxB, aac(6')-Ic*

Table 6. Antimicrobial resistance phenotypes and antimicrobial resistance genes detected using different approaches. [a]The analysis aborted when the script tried to connect to NCBI. [b]Only non-susceptibility is indicated. Abbreviations: AMP/SUL, ampicillin/sulbactam; AMX, amoxicillin; AMC, amoxicillin/clavulanate; CFX, cefuroxime; FOS, fosfomycin; FOX, cefoxitin; CIP, ciprofloxacin; CLI, clindamycin; DOX, doxycycline; ERY, erythromycin; FUS, fusidic acid; GEN, gentamicin; GENhl, gentamicin high-level; LEV, levofloxacin; NIT, nitrofurantoin; PEN, penicillin; POL, polymyxin B; STRhl, streptomycin high-level; TEC, teicoplanin.

may be lost. This is mainly driven by a bottleneck in culture, where some bacterial species are not isolated with standard work up protocols (frequently anaerobes and slow-growing organisms). The presence of antimicrobial resistance genes in plasmids of bacteria other than those isolated through culture poses a risk since they are not identified by conventional methods but could potentially be horizontally transmitted to pathogenic bacteria under the antimicrobial selective pressure of treatment. Antimicrobial administration may also select minority populations where these resistance determinants are found. Furthermore, the understanding of how plasmids are shared by different bacteria in a bacterial community (e.g. within an infection site or in the gut) can improve our understanding of how these elements disseminate across species and from patient to patient[11]. The SMg approach is clearly more efficient than culture in identifying the "cloud" of plasmids present in a given sample (Fig. 4) and which can be potentially transferred to more pathogenic species generating problems of resistance, as was the case with the emerge of vancomycin resistance *S. aureus*[12].

Whole-genome sequencing has been used extensively for several purposes[13] and is considered to have the potential of playing an important role in clinical microbiology[14]. It is the ongoing goal of medical molecular microbiology to develop faster typing methods that can be used for outbreak surveillance. For this purpose, we assembled the metagenomics data and compared it with the assemblies given by WGS. Surprisingly, the assemblies provided by SPAdes in BaseSpace were closer to the assemblies provided by WGS. When comparing the genomes obtained through WGS and SMg, we could see that in 4 out of 8 bacterial isolates the number of different alleles was ≤ 7. This showed the potential of SMg to draw phylogenetic relationships from uncultured bacterial genomes, although more potentially limited than those obtained using WGS data from axenic cultures. As for the detection of resistance genes, a key limiting factor may be the number of bacterial reads, reflected in a lower genome coverage (e.g. samples 4 and 6). In these cases, we would have to either improve the human-DNA depletion step, improve the microbial enrichment or perform sequencing at a higher sequencing depth to have enough microbial reads to be able to get a more appropriate genome coverage. Yet, this last step will severely raise the sequencing costs, which might render the methodology unfeasible for routine application.

In this study, we evaluated the results of metagenomics pipelines using three different methods. CLC Genomics Workbench has advantages over the other methods. It does not require previous knowledge of Unix-based tools, it is arguably the most user-friendly and delivered reliable results for microbial identification and antimicrobial resistance gene detection. The downside was the assembly approaches, which provided lower wgMLST allele detection, when compared to the assemblies using SPAdes (BaseSpace and Unix). BaseSpace, the other commercial solution, on the other hand, provided only a few tools that can be used for metagenomics data. Furthermore, since Illumina did not develop the apps themselves, they offered no direct support. Contacting the developers (via email and posting on their forum) does not guarantee a solution to the issues in a time frame compatible with a routine clinical microbiology laboratory work. The dependence and no direct control over a third party to resolve software bugs and provide a stable platform illustrates a disadvantage of a cloud-based system like

		WGS	Shotgun metagenomics	
Sample number	**Conventional identification (MALDI-TOF)**	**CLC Genomics Workbench v10.1.1**	**CLC Genomics Workbench v10.1.1**	**metaMLST (Unix-based)**
1	*E. faecium* *S. haemolyticus*	ST117 ST25	Not detected (6 alleles identified correctly) Not detected (3 alleles identified correctly)	ST117 Not detected
2	*E. avium* *E. coli* Anaerobes	—[#] —[#] —[#]	— Not detected —	— Not detected —
3	*S. epidermidis*	—[#]	Not detected	Not detected
4	*S. aureus*	ST30	Not detected	Not detected
5	*E. coli* *K. oxytoca* *S. anginosus* *E. faecalis* Anaerobes	ST141 ST40 —[#] ST179 —[#]	ST141 Not detected — Not detected —	ST4508 Not detected — Not detected —[#]
6	*E. faecium*	ST117	Not detected	Not detected
7	*S. aureus*	ST30	ST30	ST667
8	*O. intermedium*	—	—	—
9	*S. aureus*	—[#]	Not detected	Not detected
10	*S. marcescens*	—[#]	—	—

Table 7. Results of MLST using by whole genome sequencing and shotgun metagenomics. Abbreviations: ST, sequence type.

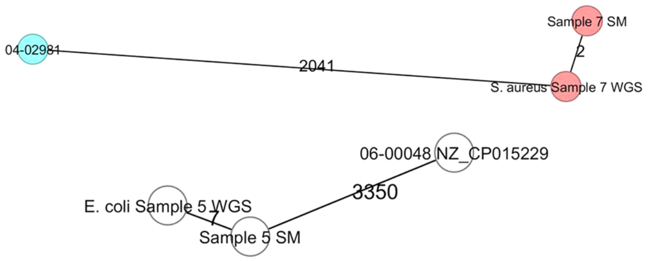


Figure 2. Minimum-spanning tree based on wgMLST allelic profiles of two *S. aureus* genomes and two *E. coli* genomes obtained through SM and WGS in comparison to reference strains 04-02981 (GenBank accession number NC_017340) and 06-00048 (NZ_CP015229), respectively. Each circle represents an allelic profile based on sequence analysis. The numbers on the connecting lines illustrate the numbers of target genes with differing alleles.

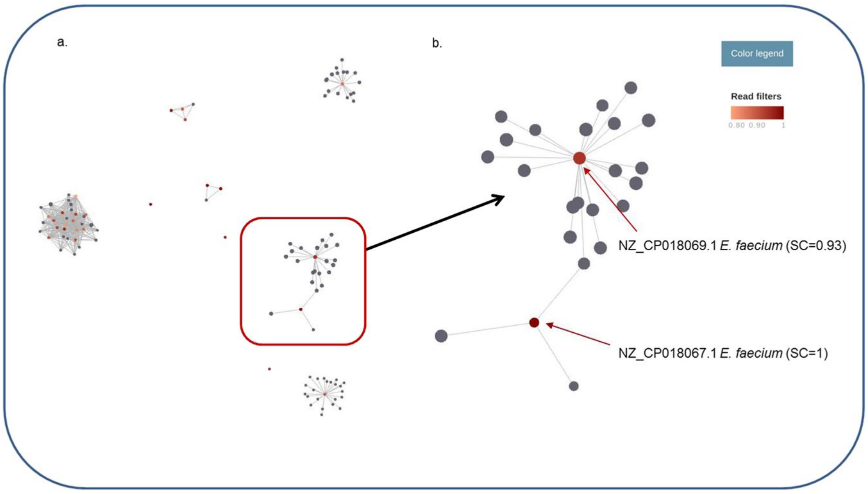


Figure 3. (**a**) Overview of the nodes (representing plasmid sequences) and links between plasmids (connecting similar plasmids) found in Sample 1 (SMg) using the pATLAS tool. (**b**) A closer look at one of the cloud of plasmids. The color gradient in each cloud of plasmids represents the plasmid sequence coverage (SC), varying between 0–0.79 (grey) and 0.80–1 (red gradient).

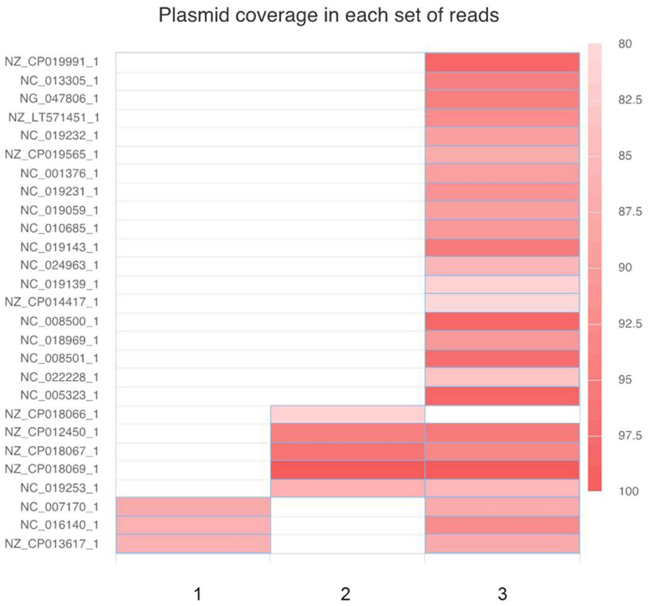


Figure 4. A heatmap comparing the identified plasmids using bowtie2 in *S. haemolyticus* WGS (1), *E. faecium* WGS (2) and in the SMg dataset (3) isolated from sample 1.

BaseSpace. Finally, the Unix-based pipeline complemented the data on antimicrobial resistance genes but did not offer better results in terms of microbial identification and MLST typing. However, many more freely available tools for this last purpose could have been used, potentially improving on the results obtained. Reference-guided assembly approaches, taking advantage of the species information derived in the first steps of our analysis pipelines, will deserve further study in the future since these may provide higher quality assemblies from metagenomics data. The main advantage of an open-source approach is its flexibility since it allows the user to choose the most adequate method for each desired outcome.

There were several limitations to this study. First, the number of samples included was low and some of the bacterial isolates were not available for further WGS analysis. However, the extended data analyses performed in each sample limited the number of samples to be included. It is our intention to move forward with the most adequate pipelines for each purpose and apply them to additional patients' samples. Second, the samples differed greatly from each other. However, in our point of view, this was beneficial to the study, since it did not bias the analyses as it could have happened if only one type of sample had been used. Finally, we used three different extraction methods that could have influenced the final results. Yet, as can be seen in Table 1, the number of human reads differed between samples, even when using the same extraction kit. This suggests none of the kits is clearly superior to the others and that the ratio between host and microbial DNA or other individual sample characteristics will be the major determinants of the proportion of microbial reads.

In conclusion, this study showed the potential but also highlighted the problems of implementing shotgun metagenomics for the identification and typing of pathogens directly from clinical samples. Based on the results obtained here we can conclude that the tools and databases used for taxonomic classification and antimicrobial resistance will have a key impact on the results, cautioning about the comparison between studies using different methods and suggesting that efforts need to be directed towards standardization of the analysis methods if SMg is to be used routinely in clinical microbiology.

Methods

Sample collection. Nine body fluid samples and one tissue sample entering the Medical Microbiology laboratory were selected for metagenomics sequencing. These included one sample from peritoneal fluid, five from pus (3 abscesses and 2 empyema), two from synovial fluid of knees with prosthesis, one from sputum and one from a bone biopsy (Table 1). All samples were stored at 4 °C for a variable period (2–10 days). The samples used for the present analyses were collected during routine diagnostics and infection prevention and control investigations. All procedures were carried out according to guidelines and regulations of UMCG concerning the use of patient materials for the validation of clinical methods, which are in compliance with the guidelines of the Federation of Dutch Medical Scientific Societies (FDMSS). Every patient entering the UMCG is informed that samples taken may be used for research and publication purposes, unless they indicate that they do not agree to it. This procedure has been approved by the Medical Ethical Committee of the UMCG. Informed consent was obtained from all individuals or their guardians prior to study participation. All samples were used after performing and completing a conventional microbiological diagnosis and were coded to protect patients' confidentiality. All experiments were performed in accordance with the guidelines of the Declaration of Helsinki and the institutional regulations.

Classic culturing and susceptibility testing. The samples were cultured following methods routinely used in our institution. Briefly, samples were streaked onto five plates (Mediaproducts BV, Groningen, The Netherlands) - blood agar (aerobic), chocolate agar (aerobic), McConkey agar (aerobic), Brucella agar (anaerobic) and Sabouraud Dextrose + AV (aerobic) - and incubated overnight under aerobic and anaerobic atmosphere at 37 °C. The two pus samples were also plated onto Phenylethyl alcohol sheep blood agar (PEA), Kanamycin vancomycin laked blood (KVLB) agar and Bacteroides bile esculin (BBE) agar and incubated under anaerobic conditions overnight. The isolates recovered were subjected to susceptibility testing by Vitek 2 using either the AST-P559 (Gram-positive bacteria) or the AST-N344 (Gram-negative bacteria) card (bioMérieux, Marcy-l'Étoile, France) and identified by MALDI-TOF MS (Bruker Daltonik, Gmbh, Germany) using standard protocols.

DNA extraction, library preparation and sequencing. The DNA for metagenomic sequencing was isolated using the Ultra-Deep Microbiome Prep (Molzym Life Science, Bremen, Germany), Micro-Dx™ kit (Molzym Life Science) or QIAamp DNA Microbiome Kit (Qiagen, Hilden, Germany) directly from the clinical samples and a negative control consisting of a mock sample of DNA and RNA free water (Table 1). These kits include human DNA depletion steps. The QIAamp DNA Microbiome Kit was used according to the manufacturer's protocol with an additional 5 min air-dry step before elution. For microbial lysis, a Precellys 24 homogenizer (Bertin, Montigny-le-Bretonneux, France) set to 3 times 30 seconds at 5000 rpm separated by 30 seconds was used. After extraction, DNA was quantified with the Qubit 2.0 (Life Technologies, ThermoFisher Scientific, Waltham, Massachusetts, EUA) and NanoDrop 2000 (ThermoFisher Scientific). The DNA quality was assessed using the Genomic DNA ScreenTape and Agilent 2200 TapeStation System (Agilent Technologies, California, United States of America). Isolated DNA was purified using Agencourt AMPure XP beads (Beckman Coulter, California, United States of America) according to the manufacturer's instructions, to eliminate small DNA fragments and chemical contaminants (e.g. benzonase). The DNA was then diluted to 0.2 ng/μl and 1 ng was used for the library preparation, using the Nextera XT Library Preparation kit (Illumina, California, United States of America), according to the manufacturer's protocol. Cluster generation and sequencing were performed with the MiSeq Reagent Kit v2 500-cycles Paired-End in a MiSeq instrument (Illumina). Samples were sequenced in batches of 5 samples on a single flow cell.

For the DNA extraction of bacterial isolates (when an isolate was recovered from culture), we used the UltraClean Microbial DNA Isolation Kit (Mo Bio), with some modifications. We started with solid cultures and resuspended a 10 μl-loopfull of culture directly into the tube with the microbeads and microbead solution. The library preparation, cluster generation and sequencing was performed as described above. Strains were sequenced in batches of 12 to 16 on a single flow cell.

Bioinformatics analyses. In order to evaluate and compare the accuracy and reliability of the bioinformatics analyses in providing the closest results to culture and WGS of any cultured isolates, three different pipelines (two commercially and one freely available) were used (Fig. 1). Different tools to perform raw read quality control, filtering and trimming were used and reads were mapped against the human genome (hg19) before performing taxonomic classification. Reads mapping to hg19 where removed from the analysis to increase the efficiency of the bioinformatics tools. Typing (MLST), phylogenetic analysis, plasmid analysis, detection of antimicrobial resistance and virulence genes was performed. To determine the appropriateness of SMg as predictor of the WGS (chromosome and plasmids), SMg results obtained were compared with the results of WGS of any bacterial isolates obtained from culturing the sample.

All the parameters used in each approach are available in Supplementary Table 1.

Unix-based approach. For the metagenomics data, read quality control and cleaning was performed using FastQC v0.11.5 and Trimmomatic v0.36, respectively, through the INNUca v2.6 pipeline (https://github.com/B-UMMI/INNUca), excluding assembly and polishing. Using a reference mapping approach against the human genome (UCSC hg19), human reads were discarded using Bowtie 2 v2.3.2[15] and SAMtools v1.3.1[16]. Those paired reads that did not map against the human genome were used in subsequent analyses. The bacterial species were identified through Kraken v0.10.5-beta[17] using the miniKraken database (pre-built 4 GB database constructed from complete bacterial, archaeal and viral genomes in RefSeq, as of Dec. 8, 2014), MIDAS[18] using the midas_db_v1.2 database (>30,000 bacterial reference genomes, as of May 9, 2018) and MetaPhlAn2 v2.0[19] using the database provided by the tool (~13,500 bacterial and archaeal, ~3,500 viral, and ~110 eukaryotic reference genomes, as of May 9, 2018). The sequence type (ST) was obtained through metaMLST v1.1[20] based on the metamlstDB_2017. Antimicrobial resistance genes were detected using ReMatCh v3.2 (https://github.com/B-UMMI/ReMatCh), a read mapping tool that uses Bowtie 2 v2.3.2[15] and the following rules for gene presence/absence: genes were considered present when ≥80% of the reference sequence was covered and the sample sequence was ≥70% identical to the one used as reference. For that, ResFinder database (2231 genes, downloaded on 29-06-2017) was used as reference and, due to the low coverage of microbial metagenomics samples, a minimal coverage depth of 1 read was set to consider a reference sequence position as covered (and therefore present in the sample data), as well as to perform base call (used for sequence identity determination). Finally, the assembly was accomplished through SPAdes v3.10.1[21].

Plasmid detection was achieved by running the script PlasmidCoverage (https://github.com/tiagofilipe12/PlasmidCoverage), using the plasmid sequences downloaded from NCBI RefSeq (ftp://ftp.ncbi.nlm.nih.gov/genomes/refseq/plasmid/, as of May 11, 2017). The script uses Bowtie 2 v2.2.9[15], to map the pre-processed input reads against the plasmid database (bowtie2 index for all plasmid sequences). For bowtie2 we used the '-k' option, allowing each read to map to as many plasmid sequences as present in the NCBI RefSeq plasmid database (since plasmid sequences are modular)[22,23]. Then, this pipeline used SAMtools v1.3.1[16] to estimate the coverage for each position, and reported the length of plasmid sequence covered (in percentage) and average depth (mean number of reads mapped against a given position in each plasmid). Plasmids with less than 80% of its length covered were excluded from the final results in line with what has described elsewhere[11]. The pATLAS tool (http://www.patlas.site) was used to visualize which plasmids were present.

For the WGS reads of the bacterial isolates, the whole INNUca v2.6 pipeline was run, including SPAdes assembly and polishing. Plasmids were detected as mentioned previously.

Commercial-based approach. The fastq files containing the reads were uploaded into CLC Genomics Workbench v10.1.1, using the following options: Illumina import, paired-reads, paired-end (forward-reverse) and minimum distance of 1 and a maximum distance of 1000 (default). The trimming was performed using the default settings, except the quality trimming score limit was set to 0.01 and we added a Trim adapter list containing Illumina adapters. The mapping was performed with the Map Reads to Reference tool, using the hg19 genome as reference. The default settings were used with the addition of the collect un-mapped reads option. The de novo assembly tool was used for the assembly (even for the metagenomics reads) and, apart from the word size, which was changed to 29, all the settings were default. Two tools were used for the microbial identification, Taxonomic Profiling and Find Best Matches using K-mer Spectra (Microbial Genomics Module). In both, the bacterial and fungal databases were downloaded from NCBI RefSeq (with the Only Complete Genomes option turned off; minimum length 500,000 nucleotides) on 08-07-2017 (bacterial, 70,868 sequences) and 25-05-2017 (fungal, 377 sequences). The antimicrobial resistance genes were detected, based on the assembled contigs, using the Find Resistance tool (Microbial Genomics Module) and were initially only considered present when they were $\geq$70% identical to the reference and $\geq$80% of the sequence was covered. The analysis was also repeated using $\geq$40% and $\geq$20% of sequence coverage for comparison purposes. The database containing the antimicrobial resistance genes was downloaded directly to the software from ResFinder (https://cge.cbs.dtu.dk/services/data.php, downloaded on 05-07-2017, 2156 sequences). The MLST was determined through the Identify MLST tool (Microbial Genomics Module), using all MLST schemes available at PubMLST (04-03-2017). The same database used for plasmid detection in Unix, was used for mapping the reads in CLC Genomics Workbench. Again, plasmids with less than 80% of its length covered were excluded from the final results. For WGS reads we used the Trim Sequences tool and the assembly, antimicrobial resistance genes detection, and MLST determination were performed as before.

Web-based approaches. The fastq files containing the reads were uploaded into the BaseSpace website. First, the raw forward and reverse fastq reads were subjected to FASTQ Toolkit for adapter/quality trimming and length filtering with standard settings and length filtering adjusted to a minimum of 100 and a maximum of 500. The trimmed reads were then used as input for all the following processes. The available microorganism identification apps Kraken v1.0.0, MetaPhlAn v1.0.0 and GENIUS v.1.1.0 were used with the standard settings/parameters. SEAR was used to detect antimicrobial resistance genes, maintaining the standard settings except for the clustering stringency which was set to 0.98 and the annotation stringency was set to 40. The SPAdes Genome Assembler v3.9.0 app was run with the standard parameters for multi cell data type. For metagenomic datatype settings, the running mode was set to only assembly and careful mode was disabled.

The reads were uploaded into CosmosID (https://app.cosmosid.com/login) and Taxonomer[24] (https://www.taxonomer.com/) directly without any quality trimming. We used the Full Analysis mode in Taxonomer.

wgMLST analyses. Typing was done by MLST and wgMLST analyses obtained using Ridom SeqSphere + v4.0.1. The genomic data (assembled contigs) obtained from SMg was compared to the data obtained through WGS. Since no cg/wgMLST scheme was available for *Escherichia coli*, *Enterococcus faecalis*, *Ochrobactrum intermedium* and *Staphylococcus haemolyticus*, cgMLST and accessory genome schemes were constructed, using Ridom SeqSphere+ cgMLST Target Definer with the following parameters: a minimum length filter that removes all genes smaller than 50 bp; a start codon filter that discards all genes that contain no start codon at the beginning of the gene; a stop codon filter that discards all genes that contain no stop codon or more than one stop codon or that do not have the stop codon at the end of the gene; a homologous gene filter that discards all genes with fragments that occur in multiple copies within a genome (with identity of 90% and >100 bp overlap); and a gene overlap filter that discards the shorter gene from the cgMLST scheme if the two genes affected overlap >4 bp. The remaining genes were then used in a pairwise comparison using BLAST version 2.2.12 (parameters used were word size 11, mismatch penalty −1, match reward 1, gap open costs 5, and gap extension costs 2). All genes of the reference genome that were common in all query genomes with a sequence identity of $\geq$90% and 100% overlap and, with the default parameter stop codon percentage filter turned on, formed the final cgMLST scheme. The combination of all alleles in each strain formed an allelic profile that was used to generate minimum spanning trees using the parameter "pairwise ignore missing values" during distance calculation[25].

Statistical analysis. The sensitivity and positive predictive value of each taxonomic classification method were determined. Classical culture and MALDI-TOF identifications were considered as the gold standard. The true positives were considered when the same bacterial species were identified by culture/MALDI-TOF and the taxonomic classification method. The false positives were detected when bacterial species different from those identified by culture/MALDI-TOF, were identified by the taxonomic classification method. The false negatives were determined when the bacterial species identified by culture/MALDI-TOF were not identified by the taxonomic classification method.

Accession codes. The paired-trimmed-un-mapped reads (hg19) generated for each sample have been submitted to SRA under project number SRP126380. The cgMLST schemes are deposited in figshare under the DOI:10.6084/m9.figshare.5679376.

References

1. Hasman, H. *et al*. Rapid whole-genome sequencing for detection and characterization of microorganisms directly from clinical samples. *J. Clin. Microbiol.* **52**, 139–146 (2014).
2. Willmann, M. *et al*. Antibiotic selection pressure determination through sequence-based metagenomics. *Antimicrob. Agents Chemoth.* **59**, 7335–7345 (2015).
3. Graf, E. H. *et al*. Unbiased detection of respiratory viruses by use of a RNA sequencing-based metagenomics: a systematic comparison to a commercial PCR panel. *J. Clin. Microbiol.* **54**, 1000–1007 (2016).
4. Gyarmati, P. *et al*. Metagenomic analysis of bloodstream infections in patients with acute leukemia and therapy-induced neutropenia. *Sci. Rep.* **6**, 23532 (2016).
5. Nasheri, N., Petronella, N., Ronholm, J., Bidawid, S. & Corneau, N. Characterization of the genomic diversity of norovirus in linked patients using a metagenomic deep sequencing approach. *Front. Microbiol.* **8**, 73 (2017).
6. Schlaberg, R., Chiu, C. Y., Miller, S., Procop, G. W. & Weinstock, G. Professional Practice Committee and Committee on Laboratory Practices of the American Society for Microbiology & Microbiology Resource Committee of the College of American Pathologists. Validation of Metagenomic Next-Generation Sequencing Tests for Universal Pathogen Detection. *Arch. Pathol. Lab. Med.* **141**, 776–786 (2017).
7. Street, T. L. *et al*. Molecular diagnosis of orthopedic-device-related infection directly from sonication fluid by metagenomic sequencing. *J. Clin. Microbiol.* **55**, 2334–2347 (2017).
8. Peabody, M. A., Van Rossum, T., Lo, R. & Brinkman, F. S. L. Evaluation of shotgun metagenomics sequence classification methods using *in silico* and *in vitro* simulated communities. *BMC Bioinformatics* **16**, 363 (2015).
9. Sabat, A. J. *et al*. Targeted next-generation sequencing of the 16S-23S rRNA region for culture-independent bacterial identification - increased discrimination of closely related species. *Sci. Rep.* **7**, 3434 (2017).
10. Zankari, E. *et al*. PointFinder: a novel web tool for WGS-based detection of antimicrobial resistance associated with chromosomal point mutations in bacterial pathogens. *J. Antimicrob. Chemother.* **72**, 2764–2768 (2017).
11. Jitwasinkul, T. *et al*. Plasmid metagenomics reveals multiple antibiotic resistance gene classes among the gut microbiomes of hospitalised patients. *J. Glob. Antimicrob. Resist.* **6**, 57–66 (2016).
12. Melo-Cristino, J., Resina, C., Manuel, V., Lito, L. & Ramirez, M. First case of infection with vancomycin-resistant *Staphylococcus aureus* in Europe. *Lancet* **382**, 205 (2013).
13. Deurenberg, R. H. *et al*. Application of next generation sequencing in clinical microbiology and infection prevention. *J. Biotechnol.* **243**, 16–24 (2017).
14. Rossen, J. W. A., Friedrich, A. W. & Moran-Gilad, J. & ESCMID Study Group for Genomic and Molecular Diagnostics (ESGMD). Practical issues in implementing whole-genome-sequencing in routine diagnostic microbiology. *Clin. Microbiol. Infect.* **24**, 355–360 (2018).
15. Langmead, B. & Salzberg, S. Fast gapped-read alignment with Bowtie 2. *Nat. Meth.* **9**, 357–359 (2012).
16. Li, H. *et al*. The Sequence alignment/map (SAM) format and SAMtools. *Bioinformatics* **25**, 2078–2079 (2009).
17. Wood, D. E. & Salzberg, S. L. Kraken: ultrafast metagenomic sequence classification using exact alignments. *Gen. Biol.* **15**, R46 (2014).
18. Nayfach, S., Rodriguez-Mueller, B., Garud, N. & Pollard, K. S. An integrated metagenomics pipeline for strain profiling reveals novel patterns of bacterial transmission and biogeography. *Genome Res.* **26**, 1–14 (2016).
19. Segata, N. *et al*. Metagenomic microbial community profiling using unique clade-specific marker genes. *Nat. Methods* **8**, 811–814 (2012).
20. Zolfo, M., Tett, A., Jousson, O., Donati, C. & Segata, N. MetaMLST: multi-locus strain-level bacterial typing from metagenomic samples. *Nucleic Acids Res.* **45**, e7 (2017).
21. Bankevich, A. *et al*. SPAdes: A new genome assembly algorithm and its applications to single-cell sequencing. *J. Comp. Biol.* **19**, 455–477 (2012).
22. Smillie, C., Garcillán-Barcia, M. P., Francia, M. V., Rocha, E. & de la Cruz, F. Mobility of plasmids. *Microbiol. Mol. Biol. Rev.* **74**, 434–452 (2010).
23. Garcillán-Barcia, M. P., Alvarado, A. & de la Cruz, F. Identification of bacterial plasmids based on mobility and plasmid population biology. *FEMS Microbiol. Rev.* **35**, 936–956 (2011).
24. Flygare, S. *et al*. Taxonomer: an interactive metagenomics analysis portal for universal pathogen detection and host mRNA expression profiling. *Gen. Biol.* **17**, 111 (2016).
25. Ruppitsch, W. *et al*. Defining and evaluating a core genome multilocus sequence typing scheme for whole-genome sequence-based typing of *Listeria monocytogenes*. *J. Clin. Microbiol.* **53**, 2869–2876 (2015).

Acknowledgements

We thank Peter Posma, Yvette Bisselink and Brigitte Dijkhuizen for excellent technical assistance. We thank Dr. Michael Lustig and colleagues from Molzym Life Science for helping with extraction protocols. This project has received funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement 713660. This work was partly supported by the INTERREG VA (202085) funded project EurHealth-1Health, part of a Dutch-German cross-border network supported by the European Commission, the Dutch Ministry of Health, Welfare and Sport (VWS), the Ministry of Economy, Innovation, Digitalisation and Energy of the German Federal State of North Rhine-Westphalia and the German Federal State of Lower Saxony.

Author Contributions

N.C., J.A.C., M.R., S.P., I.A., A.W.F. and J.W.A. conceived the experiment(s), N.C., L.S. and E.C.R. conducted the experiment(s), N.C., L.S., M.M., C.I.M., T.F.J., S.R., M.C., J.A.C. and M.R. analyzed the results, N.C. and L.S. wrote the manuscript. All authors reviewed the manuscript.

Additional Information

Supplementary information accompanies this paper at https://doi.org/10.1038/s41598-018-31873-w.

Competing Interests: The authors declare no competing interests.

Journal of
Antimicrobial
Chemotherapy

J Antimicrob Chemother 2019; **74**: 3626–3628
doi:10.1093/jac/dkz363
Advance Access publication 23 August 2019

Detection of a novel *mcr-5.4* gene variant in hospital tap water by shotgun metagenomic sequencing

Giuseppe Fleres[1], Natacha Couto[1], Leonard Schuele[1], Monika A. Chlebowicz[1], Catarina I. Mendes [1], Luc W. M. van der Sluis[2], John W. A. Rossen[1], Alex W. Friedrich[1] and Silvia García-Cobos[1]*

[1]University of Groningen, University Medical Center Groningen, Department of Medical Microbiology, Groningen, The Netherlands; [2]Center of Dentistry and Oral Hygiene, University Medical Center Groningen, 9712 CP Groningen, The Netherlands

*Corresponding author. E-mail: s.garcia.cobos@umcg.nl

Sir,

Colistin is considered a last-resort antibiotic for treating serious infections caused by MDR Gram-negative bacteria. The efficacy of this antibiotic is challenged by the emergence and global spread of mobile colistin resistance (*mcr*) determinants, which threaten human, animal and environmental health. The first mobile colistin resistance gene (*mcr-1*) was reported in 2015 and since then up to eight different variants have been described.[1] In 2017, Borowiak *et al.*[2] described a new transposon-associated phosphoethanolamine transferase mediating colistin resistance, named *mcr-5*, in d-tartrate-fermenting *Salmonella enterica* subsp. *enterica* serovar Paratyphi B isolated from poultry. The *mcr-5.3* variant has been recently reported in *Stenotrophomonas* spp. from sewage water.[3] Here we report for the first time (to the best of our knowledge) the detection of an *mcr-5* gene in a hospital water environment using short-read metagenomic sequencing (SRMseq) and subsequent characterization using long-read metagenomic sequencing (LRMseq) to reveal its genetic environment.

In June 2017, eight tap-water samples (900 mL) were collected at the University Medical Center Groningen. Water samples were filtered (0.2 μm) and after DNA extraction (PowerWater DNA Extraction Kit, QIAGEN), SRMseq was performed on a MiSeq instrument (500 cycles) (Illumina). Antibiotic resistance genes were identified in the metagenome assemblies (CLC Genomics Workbench v10.1.1, QIAGEN) using ABRicate-0.7 (https://github.com/tseemann/abricate) and applying the following thresholds: $>70\%$ identity and $>80\%$ coverage. One sample contained an *mcr*-type gene (5× sequencing depth), with the nucleotide change 313C>T (amino acid change F105L) with respect to the original *mcr-5.1* gene, which was designated *mcr-5.4* by NCBI (accession no. MK965519). This sample was selected for LRMseq; the DNA libraries were prepared using the Rapid PCR Barcoding Kit (SQK-RPB004) from Oxford Nanopore Technologies (ONT) and loaded into a FLO-MIN106 R9.4 flow cell. The run was performed on a MinION device (ONT) and it proceeded for 24 h. The data were basecalled using Albacore (https://github.com/rrwick/Basecalling-comparison) and further processed with Poretools[4] and Porechop (https://github.com/rrwick/Porechop). Trimmed reads from SRMseq and LRMseq were used for hybrid-assembly analysis by metaSPAdes-3.13.0.[5] After a BLAST search using the hybrid contig containing the *mcr-5.4* gene, the plasmid pSE13-SA01718 (accession no. KY807921.1) was listed as one of the hits with the highest identity and we used it as a reference for genome comparison with the Artemis Comparison Tool (ACT) v1.0.[6] The *mcr-5.4*-carrying contig from the hybrid assembly was annotated using PATRIC v3.5.27.[7] Trimmed reads from SRMseq were used to investigate the bacterial composition by OneCodex.[8] Finally, in order to predict the bacterial host of the *mcr-5.4* gene, a contig-binning analysis of the hybrid-assembled metagenome was performed using MaxBin2 v2.2.4 (https://sourceforge.net/projects/maxbin2/), probability threshold 0.9 and minimum contig length 1000 bp. The resulting bin containing the *mcr-5.4* gene was selected for taxonomy classification using Kraken2 (https://github.com/DerrickWood/kraken2) (minikraken2 DB v1).

SRMseq showed the *mcr-5.4* gene detected in a contig of 2113 bp flanked by two truncated protein-coding sequences (CDSs), encoding the ChrB domain protein (involved in chromate resistance) and the Major Facilitator Superfamily (MFS) transporter. The hybrid-assembly analysis resulted in a contig of 8456 bp consisting of nine CDSs and four truncated CDSs (Figure 1). Comparative analysis of the genetic environment of the *mcr-5* gene, between the annotated hybrid metagenome contig and the reference plasmid pSE13-SA01718, showed a region of 4670 bp with 98% identity, corresponding to the backbone of the Tn*6452* transposon (Figure 1). We observed three truncated CDSs for the MFS-type transporter in our contig instead of two as previously described in the reference sequence pSE13-SA01718. These differences did not appear to be due to sequencing errors when we checked the sequence MK965519, (i) using pilon (https://github.com/broadinstitute/pilon) to correct for errors in short-read sequencing data and (ii) using CLC Genomic Workbench to update the hybrid contig by mapping both long and short reads against the hybrid contig. We also observed a region of 3786 bp, with no identity either with the reference plasmid pSE13-SA01718 (Figure 1) or with any other sequence in the GenBank database.

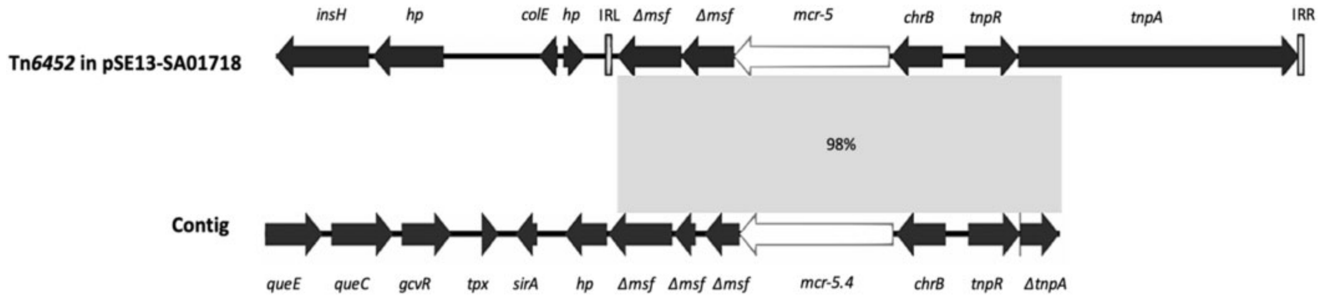


Figure 1. Comparative analysis of the genetic environment of *mcr-5* between the reference plasmid pSE13-SA01718 (accession no. KY807921.1) and the annotated hybrid metagenome contig (accession no. MK965519). The contig carrying the *mcr-5.4* gene consists of the following putative gene products: 7-carboxy-7-deazaguanine synthase (*queE*), 7-cyano-7-deazaguanine synthase (*queC*), glycine cleavage system transcriptional antiactivator GcvR (*gcvR*), thiol peroxidase (*tpx*), sulphurtransferase TusA family protein (*sirA*), hypothetical protein (*hp*), truncated MFS-type transporter (Δ*msf*), lipid A phosphoethanolamine transferase (*mcr-5.4*), ChrB domain protein (*chrB*), transposon resolvase (*tnpR*) and truncated transposon transposase (Δ*tnpA*). Areas with 98% identity between sequences are represented in light grey. Arrows indicate the position and direction of the genes. The transposon Tn*6452* sequence in the reference plasmid pSE13-SA01718 is bounded by inverted repeats: IRL and IRR.

Species previously described to harbour an *mcr-5* gene are *Escherichia coli*, *Pseudomonas aeruginosa*, *Salmonella enterica*, *Aeromonas hydrophila* and *Cupriavidus gilardii.* The bacterial composition analysis of the water sample using SRMseq showed the presence of *Pseudomonas* spp. (relative abundance: 0.004%), *Cupriavidus* spp. (relative abundance: 0.001%) and *Aeromonas* spp. (relative abundance: 0.0003%). The binning analysis produced a bin positive for the *mcr-5.4* gene consisting of 1336 contigs (genome size: 5 175 285 bp; genome completeness: 68.2%). This bin was taxonomically classified as bacteria (70.73%) and proteobacteria (64.90%), and from this the most abundant class was Gammaproteobacteria (37.20%) (order Pseudomonadales, 15.57%), followed by Betaproteobacteria (14.90%) (order Burkholderiales, 10.63%).

Colistin resistance determinants (*mcr*) have been rarely reported in water environments; *mcr-1* has been detected in both hospital sewage and in environmental water streams and *mcr-3* in environmental water.[9,10] To the best of our knowledge, this is the first-time description of an *mcr-5* gene in an indoor and healthcare water environment. Despite the fact that the comparative analysis showed the hybrid contig covering a large region of Tn6452, neither the left inverted repeat (IRL) nor the right inverted repeat (IRR) have been found. In addition, the lack of the right transposon region does not allow us to search for other possible inverted repeats. Thus, it is not possible to conclude whether the described *mcr-5.4* gene is transferable or not. Taxonomic analysis suggested the order of Pseudomonadales as the most probable host of the *mcr-5.4* gene in the water sample. Further studies are needed to determine the frequency of this gene in hospital water and other water environments and to evaluate the potential risks for patients and healthcare workers.

Acknowledgements

We would like to thank Erwin C. Raangs for technical assistance.

Funding

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement 713660 (MSCA-COFUND-2015-DP 'Pronkjewail'), which includes in-kind contributions by commercial partners. None of the commercial partners had any influence on interpretation of reviewed data and conclusions drawn, or on drafting of the manuscript. This work was partly supported by the INTERREG VA (202085)-funded project EurHealth-1Health, part of a Dutch–German cross-border network supported by the European Commission, the Dutch Ministry of Health, Welfare and Sport (VWS), the Ministry of Economy, Innovation, Digitalization and Energy of the German Federal State of North Rhine-Westphalia and the German Federal State of Lower Saxony.

Transparency declarations

None to declare.

References

1 Wang X, Wang Y, Zhou Y *et al*. Emergence of a novel mobile colistin resistance gene, *mcr-8*, in NDM-producing *Klebsiella pneumoniae*. *Emerg Microbes Infect* 2018; **7**: 122.

2 Borowiak M, Fischer J, Hammerl J *et al*. Identification of a novel transposon-associated phosphoethanolamine transferase gene, *mcr-5*, conferring colistin resistance in d-tartrate fermenting *Salmonella enterica* subsp. *enterica* serovar Paratyphi B. *J Antimicrob Chemother* 2017; **72**: 3317–24.

3 Li J, Liu S, Fu J *et al*. Co-occurrence of colistin and meropenem resistance determinants in a *Stenotrophomonas* strain isolated from sewage water. *Microb Drug Resist* 2019; **25**: 317–25.

4 Loman N, Quinlan A. Poretools: a toolkit for analyzing nanopore sequence data. *Bioinformatics* 2014; **30**: 3399–401.

5 Nurk S, Meleshko D, Korobeynikov A *et al*. metaSPAdes: a new versatile metagenomic assembler. *Genome Res* 2017; **27**: 824–34.

6 Carver T, Rutherford K, Berriman M *et al*. ACT: the Artemis Comparison Tool. *Bioinformatics* 2005; **21**: 3422–3.

7 Wattam A, Davis J, Assaf R *et al*. Improvements to PATRIC, the all-bacterial Bioinformatics Database and Analysis Resource Center. *Nucleic Acids Res* 2016; **45**: D535–42.

8 Minot SS, Krumm N, Greenfield NB. One Codex: a sensitive and accurate data platform for genomic microbial identification. *bioRxiv* 2015; doi: 10.1101/027607.

9 Zhao F, Feng Y, Lü X *et al*. An IncP plasmid carrying the colistin resistance gene *mcr-1* in *Klebsiella pneumoniae* from hospital sewage. *Antimicrob Agents Chemother* 2016; **61**: e02229–16.

10 Tuo H, Yang Y, Tao X *et al*. The prevalence of colistin resistant strains and antibiotic resistance gene profiles in Funan river, China. *Front Microbiol* 2018; **9**: 3094.

J Antimicrob Chemother 2019; **74**: 3628–3630
doi:10.1093/jac/dkz387
Advance Access publication 11 September 2019

Detection of chromosome-mediated *tet*(X4)-carrying *Aeromonas caviae* in a sewage sample from a chicken farm

Chong Chen[1], Liang Chen[2], Yan Zhang[1], Chao-Yue Cui[1], Xiao-Ting Wu[1], Qian He[1], Xiao-Ping Liao[1], Ya-Hong Liu[1] and Jian Sun[1]*

[1]*National Risk Assessment Laboratory for Antimicrobial Resistance of Animal Original Bacteria, College of Veterinary Medicine, South China Agricultural University, Guangzhou, China;* [2]*Hackensack-Meridian Health Center for Discovery and Innovation, Nutley, NJ, USA*

*Corresponding author. E-mail: jiansun@scau.edu.cn

Sir,

Recently, novel plasmid-mediated tigecycline resistance mechanisms, Tet(X3) and Tet(X4), have been described in Enterobacteriaceae and *Acinetobacter* isolates from animals and humans.[1] It raises a global antimicrobial resistance concern because these tetracycline-inactivating enzymes are able to inactivate the entire family of tetracycline antibiotics, including the newly FDA-approved eravacycline and omadacycline.[1] The plasmid-borne *tet*(X3) and *tet*(X4) genes have been identified in >15 different Gram-negative species, with *Escherichia coli* being the most common.[1,2] Here we report, to the best of our knowledge, the first identification of the *tet*(X4) gene on the chromosome of an *Aeromonas caviae* strain from sewage in China. *Aeromonas* species, including *A. caviae*, are important zoonotic pathogens of poikilotherms, but are now emerging as important human pathogens,[3] and have been frequently found to carry antimicrobial resistance genes (e.g. $bla_{\text{NDM-1}}$ and *mcr-3* variants) on the chromosome.[4–6]

During a routine antimicrobial resistance surveillance study, a *tet*(X4)-positive strain, WCW1-2, was isolated on an LB agar plate containing tigecycline (2 mg/L) from a sewage sample from a chicken farm in 2018 in Guangdong, China. The 16S rRNA sequencing analysis further suggested that it belonged to *A. caviae*, which shared >99.9% nucleotide identity with the isolates from patients in USA and Brazil (accession numbers CP026055 and CP024198). Antimicrobial susceptibility testing was conducted by broth microdilution, with *E. coli* ATCC 25922 as the quality control strain, and interpreted according to the CLSI guideline.[7] The *tet*(X4)-positive *A. caviae* WCW1-2 was resistant to tetracycline, amoxicillin/clavulanic acid, ciprofloxacin and trimethoprim/sulfamethoxazole, but remained susceptible to amikacin, cefotaxime, colistin, gentamicin and meropenem (Table S1, available as Supplementary data at *JAC* Online). In addition, WCW1-2 exhibited high MICs of tigecycline (16 mg/L), eravacycline (4 mg/L) and omadacycline (8 mg/L).

The tigecycline resistance in *A. caviae* WCW1-2 failed to transfer to sodium azide-resistant *E. coli* J53 by filter mating, but further gene cloning of *tet*(X4) and its putative promoter into a pUC18 vector (primers in Table S2) confirmed the *tet*(X4)-mediated tigecycline resistance. Susceptibility testing results showed that the *tet*(X4) construct had 64- to 512-fold increases in MICs of tetracycline (128 mg/L), chlortetracycline (256 mg/L), oxytetracycline (128 mg/L), doxycycline (32 mg/L), minocycline (16 mg/L), tigecycline (8 mg/L), eravacycline (2 mg/L) and omadacycline (8 mg/L), which were consistent with the results for the parental strain WCW1-2 (Table S1).

Genomic DNA of *A. caviae* WCW1-2 was then completely sequenced using a combination of the Nanopore GridION and Illumina HiSeq platforms (Nextomics, Wuhan, China), followed by assembling with Unicycler.[8] The results of WGS revealed that WCW1-2 belonged to a novel ST, ST645, and harboured one chromosome of 4684096 bp (CP039832), but without plasmids. The *tet*(X4) gene was found to be on the chromosome of WCW1-2, which explained the failure of transfer of tigecycline resistance, and shared a homology region (namely upstream of the Δ*merR* gene and downstream of the *ucpA* gene) with the chromosome of another *A. caviae* strain (Figure 1a). Moreover, WCW1-2 harboured an additional 15 antimicrobial resistance genes encoding resistance to β-lactams ($bla_{\text{MOX-5}}$ and $bla_{\text{OXA-10}}$), aminoglycosides [*aadA1*, *aph(3')-Ia*, *aph(3")-Ib* and *aph(6)-Id*], fluoroquinolones [*qnrVC4* and *aac(6')-Ib-cr*], phenicols (*cmlA1*, *catB3* and *floR*), trimethoprim/sulfamethoxazole (*sul1*, *dfrA14* and *dfrB4*) and tetracyclines [*tet*(A)].

A further BLASTn search for the *tet*(X4) gene against the NCBI database identified a series of Enterobacteriaceae carrying the same subtype from humans (e.g. NZ_NQAI01000053) and pigs (e.g. NZ_NQBP01000050), including the first described *tet*(X4)-harbouring plasmid, p47EC (MK134376) (Figure 1b). Analysis of their genetic environments revealed that the *tet*(X4) gene was usually

MICROBIAL GENOMICS

METHOD

Mendes *et al.*, *Microbial Genomics* 2020;6

DOI 10.1099/mgen.0.000328

DEN-IM: dengue virus genotyping from amplicon and shotgun metagenomic sequencing

Catarina I. Mendes[1,2,*,†], Erley Lizarazo[2]†, Miguel P. Machado[1], Diogo N. Silva[1], Adriana Tami[2], Mário Ramirez[1], Natacha Couto[2], John W. A. Rossen[2] and João A. Carriço[1]

Abstract

Dengue virus (DENV) represents a public health threat and economic burden in affected countries. The availability of genomic data is key to understanding viral evolution and dynamics, supporting improved control strategies. Currently, the use of high-throughput sequencing (HTS) technologies, which can be applied both directly to patient samples (shotgun metagenomics) and to PCR-amplified viral sequences (amplicon sequencing), is potentially the most informative approach to monitor viral dissemination and genetic diversity by providing, in a single methodological step, identification and characterization of the whole viral genome at the nucleotide level. Despite many advantages, these technologies require bioinformatics expertise and appropriate infrastructure for the analysis and interpretation of the resulting data. In addition, the many software solutions available can hamper the reproducibility and comparison of results. Here we present DEN-IM, a one-stop, user-friendly, containerized and reproducible workflow for the analysis of DENV short-read sequencing data from both amplicon and shotgun metagenomics approaches. It is able to infer the DENV coding sequence (CDS), identify the serotype and genotype, and generate a phylogenetic tree. It can easily be run on any UNIX-like system, from local machines to high-performance computing clusters, performing a comprehensive analysis without the requirement for extensive bioinformatics expertise. Using DEN-IM, we successfully analysed two types of DENV datasets. The first comprised 25 shotgun metagenomic sequencing samples from patients with variable serotypes and genotypes, including an *in vitro* spiked sample containing the four known serotypes. The second consisted of 106 paired-end and 76 single-end amplicon sequences of DENV 3 genotype III and DENV 1 genotype I, respectively, where DEN-IM allowed detection of the intra-genotype diversity. The DEN-IM workflow, parameters and execution configuration files, and documentation are freely available at https://github.com/B-UMMI/DEN-IM).

DATA SUMMARY

1. The Supplementary Material and tables are available at Figshare under https://doi.org/10.6084/m9.figshare.11316599.v1.

2. The 106 DENV-3 amplicon sequencing paired-end short-read datasets are available under BioProject PRJNA394021 and the 78 DENV-1 amplicon sequencing single-end short-read datasets are available under BioProject PRJNA321963. The 25 shotgun metagenomics dataset is available under BioProject PRJNA474413. The accession numbers for all the samples in the shotgun metagenomics dataset are available in the Supplementary Material (https://doi.org/10.6084/m9.figshare.11316599.v1).

3. The accession numbers for the 41 samples belonging to the Zika virus, Chikungunya virus and yellow fever virus

---

Received 21 August 2019; Accepted 23 December 2019; Published 05 March 2020

Author affiliations: [1]Instituto de Microbiologia, Instituto de Medicina Molecular, Faculdade de Medicina, Universidade de Lisboa, Lisboa, Portugal; [2]University of Groningen, University Medical Center Groningen, Department of Medical Microbiology and Infection Prevention, Groningen, The Netherlands.

***Correspondence:** Catarina I. Mendes, cimendes@medicina.ulisboa.pt

Keywords: dengue virus; surveillance; metagenomics; reproducibility; workflow; containerization; scalability.

Abbreviations: CDS, coding sequence; DENV, dengue virus; HPC, high-performance computing; HTS, high-throughput sequencing; NCR, non-coding region; QC, quality control; RT-PCT, reverse transcription polymerase chain reaction.

Metagenomic sequencing data available under BioProject PRJNA474413. DEN-IM reports for the analysed datasets are available in Figshare under https://doi.org/10.6084/m9.figshare.11316599.v1. Phylogeny inference trees for the dengue virus typing database available in Figshare at https://doi.org/10.6084/m9.figshare.11316599.v1. The supplemental material is available in Figshare at https://doi.org/10.6084/m9.figshare.11316599.v1. DEN-IM's source code and documentation available at https://github.com/B-UMMI/DEN-IM.

†These authors contributed equally to this work

Data statement: All supporting data, code and protocols have been provided within the article or through supplementary data files. Supplementary Material is available with the online version of this article.

amplicon and shotgun metagenomic datasets are available in the Supplementary Material (https://doi.org/10.6084/m9.figshare.11316599.v1).

4. DEN-IM reports for the analysed datasets are available at Figshare (https://doi.org/10.6084/m9.figshare.11316599.v1).

5. Phylogeny inference trees for the dengue virus typing database are available at Figshare (https://doi.org/10.6084/m9.figshare.11316599.v1).

6. Code for the DEN-IM workflow is available at https://github.com/B-UMMI/DEN-IM and documentation, including step-by-step tutorials, is available at https://github.com/B-UMMI/DEN-IM/wiki).

INTRODUCTION

The dengue virus (DENV), a single-stranded positive-sense RNA virus belonging to the genus *Flavivirus*, is one of the most prevalent arboviruses and is mainly concentrated in tropical and subtropical regions. Infection with DENV results in symptoms ranging from mild fever to haemorrhagic fever and shock syndrome [1]. Transmission to humans occurs through the bite of *Aedes* mosquitoes, namely *Aedes aegypti* and *Aedes albopictus* [2]. In 2010, it was predicted that the burden of dengue disease would reach 390 million cases per year worldwide [3]. Dengue has the greatest clinical significance of any arbovirus because of the high morbidity and mortality associated with it [4]. DENV is a significant public health challenge in countries where infection is endemic due to the high health and economic burden. Despite the emergence of novel therapies and ecological strategies to control the mosquito vector, there are still important knowledge gaps concerning the virus biology and its epidemiology [2].

The viral genome of ~11 000 nucleotides consists of a coding sequence (CDS) of approximately 10.2 Kb that is translated into a single polyprotein encoding three structural proteins (capsid, C; premembrane, prM; envelope, E) and seven non-structural proteins (NS1, NS2A, NS2B, NS3, NS4A, NS4B and NS5). Additionally, the genome contains two non-coding regions (NCRs) at its 5′ and 3′ ends [5].

DENV can be classified into four serotypes (1, 2, 3 and 4), differing from each other by 25–40 % at the amino acid level. They are further classified into genotypes that vary by up to 3 % at the amino acid level [2]. The DENV-1 serotype comprises five genotypes (I–V), DENV-2 comprises six (I–VI, also named American, Cosmopolitan, Asian-American, Asian II, Asian I and Sylvatic), DENV-3 comprises four (I–III and V) and DENV-4 also comprises four (I–IV).

Although real-time reverse transcription polymerase chain reaction (RT-PCR) will probably remain the front line tool in dengue aetiological diagnosis, the implementation of a surveillance system relying on high-throughput sequencing (HTS) technologies allows the simultaneous identification and characterization by serotyping and genotyping of DENV cases at the nucleotide level in a single methodological step. Due to the high sensitivity of these technologies, previous studies have shown that viral sequences can be obtained directly from patient sera using a shotgun metagenomics approach [6]. Alternatively, HTS can be used in an amplicon sequencing approach in which a PCR step is used to preamplify viral sequences before sequencing. In recent years, HTS has been used successfully as a tool for the identification of DENV directly from clinical samples with as few as ~2 reads in a total of 10^6 reads [6, 7]. This also allows the rapid identification of the serotype and genotype, which is important for disease management, as the genotype may be associated with disease outcome [8].

Several initiatives aim to facilitate the identification of the DENV serotype and genotype from HTS data. The Genome Detective project (https://www.genomedetective.com/) offers the online Dengue Typing Tool [9] (https://www.genomedetective.com/app/typingtool/dengue/), which relies on BLAST and phylogenetic methods in order to identify the closest serotype and genotype, but it requires assembled genomes in the FASTA format as input. The same project also offers the Genome Detective Typing Tool (https://www.genomedetective.com/app/typingtool/virus/) [10], which identifies which viruses are present in an HTS sample and provides their assembled genome. Additionally, several tools are available for viral read identification and assembly, such as VIP [11],

Impact Statement

The risk of exposure to DENV is increasing, not only because of travel to endemic regions, but also due to the broader dissemination of the mosquito, making the burden of dengue very significant. The decreasing costs and wider availability of high-throughput (HTS) sequencing make it an ideal technology to monitor dengue virus's (DENV's) transmission. Metagenomics approaches decrease the time required to obtain nearly complete DENV sequences without the need for time-consuming viral culture through the direct processing and sequencing of patient samples. A ready-to-use bioinformatics workflow, enabling the reproducible analysis of DENV, is therefore particularly relevant for the development of a straightforward HTS workflow. DEN-IM was designed to perform a comprehensive analysis in order to generate either assemblies or consensus of full DENV coding sequences and to identify their serotype and genotype. DEN-IM can also detect all four DENV serotypes and the respective genotypes present in a spiked sample, raising the possibility that DEN-IM can play a role in the identification of co-infection cases whose prevalence is increasingly perceived in highly endemic areas. Although it is ready to use, the DEN-IM workflow can be easily customized to the user's needs. DEN-IM enables reproducible and collaborative research, and is accessible to a wide group of researchers, regardless of their computational expertise and the resources available.

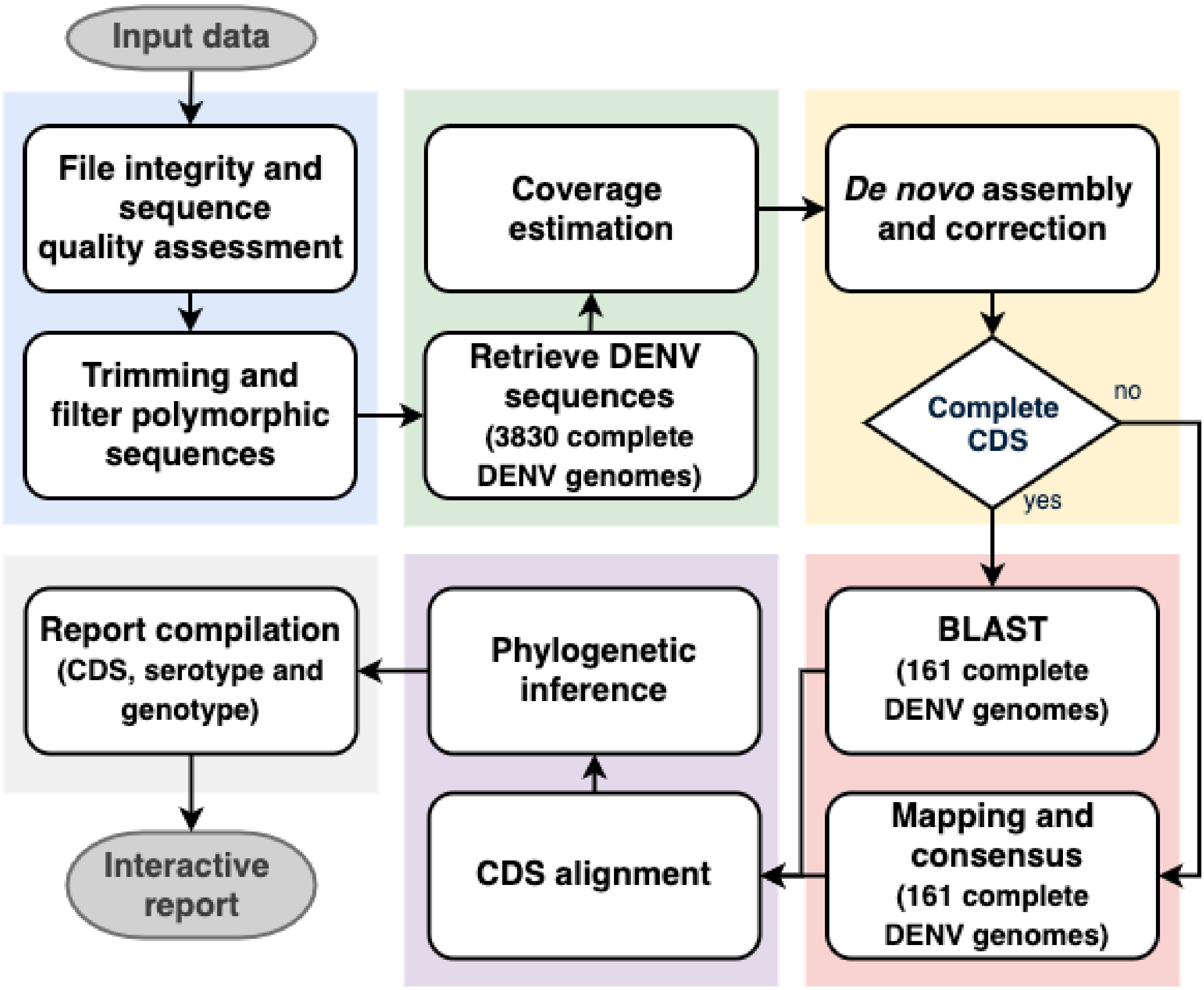


Fig. 1. The DEN-IM workflow separated into five different components. The raw sequencing reads are provided as input to the first block (in blue), responsible for quality control and elimination of low-quality reads and sequences. After successful preprocessing of the reads, these enter the second block (green) for retrieval of the DENV reads using the mapping database of 3858 complete DENV genomes as a reference. This block also provides an initial estimate of the sequencing depth. After the *de novo* assembly and assembly correction block (yellow), the CDSs are retrieved and then classified with the reduced-complexity DENV typing database containing 161 sequences representing the known diversity of DENV serotypes and genotypes (red). If a complete CDS fails to be assembled, the reads are mapped against the DENV typing database and a consensus sequence is obtained for classification and phylogenetic inference. All CDSs are aligned and compared in a phylogenetic analysis (purple). Lastly, a report is compiled (grey) with the results of all the blocks of the workflow.

virusTAP [12] and drVM [13], but none performs genotyping of the identified reads.

We developed DEN-IM as a ready-to-use, one-stop, reproducible bioinformatic analysis workflow for the processing and phylogenetic analysis of DENV using short-read HTS data. DEN-IM is implemented in Nextflow [14], workflow management software that uses Docker (https://www.docker.com) containers with preinstalled software for all the workflow tools. The DEN-IM workflow, as well as parameters and documentation, are available at https://github.com/B-UMMI/DEN-IM.

The DEN-IM workflow

DEN-IM is a user-friendly automated workflow enabling the analysis of amplicon and shotgun metagenomics data for the identification, serotyping, genotyping and phylogenetic analysis of DENV, as represented in Fig. 1, accepting raw short-read sequencing data (FASTQ files) as input, single-end or paired-end, and informing the user with an interactive and comprehensive HTML report (see Fig. S1 available in the online version of this article), as well as providing output files of the whole pipeline.

It is implemented in Nextflow, a workflow management system that allows the effortless deployment and execution of complex distributed computational workflows in any UNIX-based system, from local machines to high-performance computing clusters (HPCs) with a container engine installation, such as Docker (https://www.docker.com/), Shifter [15] or Singularity [16]. DEN-IM integrates Docker-containerized images, which are compatible with other container engines, for all the tools necessary for its execution, ensuring reproducibility and the tracking of both software code and version, regardless of the operating system used.

Users can customize the workflow execution either by using command line options or by modifying the simple plain-text configuration files. To make the execution of the workflow as simple as possible, a set of default parameters and directives is provided. An exhaustive description of each parameter is available in the Supplementary Material (see DEN-IM_Supplemental_material.pdf, Workflow parameters).

The local installation of the DEN-IM workflow, including the Docker containers with all the tools needed and the curated DENV database, requires 15 gigabytes (GB) of free disk space. The minimum requirements to execute the workflow are at least 5 GB of memory and four CPUs. The disk space required for execution depends greatly on the size of the input data, but for the datasets used in this article, DEN-IM generates approximately 5 GB of data per GB input data.

DEN-IM workflow can be divided into the following components.

Quality control and trimming

The quality control (QC) and trimming block starts with a process to verify the integrity of the input data. If the sequencing files are corrupted, the execution of the analysis of that sample is terminated. The sequences are then processed by FastQC https://www.bioinformatics.babraham.ac.uk/projects/fastqc/), version 0.11.7) to determine the quality of the individual base pairs of the raw data files. The low-quality bases and adapter sequences are trimmed by Trimmomatic (version 0.36) [16]. In addition, reads with a read length shorter than 55 nucleotides after trimming are removed from further analyses. Lastly, the low-complexity sequences, containing over 50 % of poly-A, poly-N or poly-T nucleotides, are filtered out of the raw data using PrinSeq [17] (version 0.10.4).

Retrieval of DENV sequences

In the second step, DENV sequences are selected from the sample using Bowtie 2 [18] (version 2.2.9) and SAMtools (version 1.4.1) [19]. As a reference, we provide the DENV mapping database, a curated DENV database composed of 3858 complete DENV genomes. An in-depth description of this database is available in the Supplementary Material (see DEN-IM_Supplemental_material.pdf, Dengue virus reference databases). In paired-end data, a permissive approach is followed by allowing for mates to be kept in the sample even when only one read maps to the database in order to keep as many DENV-derived reads as possible. The output of this block is a set of processed reads of putative DENV origin.

Assembly

DEN-IM applies a two-assembler approach to generate assemblies of the DENV CDSs. To obtain a high confidence assembly, the processed reads are first *de novo* assembled with SPAdes (version 3.12.0) [20]. If the full CDS fails to be assembled into a single contig, the data are reassembled with the MEGAHIT assembler (version 1.1.3) [21], a more permissive assembler developed to retrieve longer sequences from metagenomics data. The resulting assemblies are corrected with Pilon (version 1.22) [22] after mapping the processed reads to the assemblies with Bowtie 2.

If more than one complete CDS is present in a sample, each of the sequences will follow the rest of the DEN-IM workflow independently. If no full CDS is assembled with either SPAdes or MEGAHIT, the processed reads are passed on to the next module for consensus generation by mapping, effectively constituting DEN-IM′s two-pronged approach using both assemblers and mapping.

Typing

For each complete DENV CDS, the serotype and genotype is determined with the Seq_Typing tool (https://github.com/B-UMMI/seq_typing, version 2.0) [23] using BLAST [24] and the custom DENV typing database containing 161 complete sequences (see DEN-IM_Supplemental_material.pdf, Dengue virus reference databases). The tool determines which reference sequence is most closely related to the query based on the identity and length of the sequence covered, returning the serotype and genotype of the reference sequence.

If a complete CDS cannot be obtained through the assembly process, the processed reads are mapped against the same DENV typing database, with Bowtie 2, using the Seq_Typing tool, with similar criteria for coverage and identity to those used with the BLAST approach. If a type is determined, the consensus sequence obtained follows through to the next step in the workflow. Otherwise, the sample is classified as non-typable and processing is terminated.

Phylogeny

All complete DENV CDSs and consensus sequences analysed in a workflow execution are aligned with MAFFT (version 7.402) [25]. By default, or if the number of samples analysed is less than four, four representative sequences for each DENV serotype (1 to 4) from the National Center for Biotechnology Information (NCBI) are also included in the alignment. The NCBI references included are NC_001477.1 (DENV-1), NC_001474.2 (DENV-2), NC_001475.2 (DENV-3) and NC_002640.1 (DENV-4). The closest reference sequence to each analysed sample in the DENV typing database can also be retrieved and included in the alignment. With the resulting alignment, a maximum-likelihood tree is constructed with RaXML (version 8.2.11) [26].

Output and report

The output files of all tools in DEN-IM's workflow are stored in the 'results' folder in the directory of DEN-IM's execution, as well as the execution log file DEN-IM and for each component.

The HTML report (see Fig. S1), stored in the 'pipeline_results' directory, contains all results divided into four sections: report overview, tables, charts and phylogenetic tree. The report overview and all tables allow for the selection, filtering and highlighting of particular samples in the analysis. All tables have information on whether a sample failed or passed the quality control metrics, highlighted by green, yellow or red signs for pass, warning and fail messages, respectively.

The *in silico* typing table contains the serotype and genotype results for each CDS analysed, as well as the identity, coverage and GenBank ID of the closest reference in the DENV typing database. The quality control table shows information regarding the number of raw base pairs and number of reads in the raw input files and the percentage of trimmed reads. The mapping table includes the results for the mapping of the trimmed reads to the DENV mapping database, including the overall alignment rate, and an estimation of the sequence depth including only the DENV reads. For the assembly statistics table, the number of CDSs in each sample, the number of contigs and the number of assembled base pairs generated by either SPAdes or MEGAHIT assemblers is included. The number of contigs and assembled base pairs after correction with Pilon is also presented in the table. The assembled contig size distribution scatter plot is available in the chart section, showing the contig size distribution for the Pilon-corrected assembled CDSs.

Lastly, a phylogenetic tree is included, rooted at the midpoint for visualization purposes, and with each tip coloured according to the genotyping results. If the option to retrieve the closest typing reference is selected, these sequences are also included in the tree with respective typing metadata. The tree can be displayed in several conformations provided by the Phylocanvas JavaScript library (http://phylocanvas.net, version 2.8.1) and it is possible to zoom in on or collapse selected branches. The bootstrap support values of the branches can be displayed, and the tree can be exported as a Newick tree file or as a PNG image.

Software comparison

DEN-IM offers core assembly functionality, leveraging a *de novo* and consensus assembly approach to obtain a full CDS sequence to perform geno- and serotyping, followed by phylogenetic positioning of the samples analysed. This results in a phylogenetic tree showing the genotyping results, presented in an HTML file.

There are several alternative tools, both command line- and online-based, capable of identifying DENV reads and performing assembly (Table 1). VIP and drVM are both stand-alone pipelines, like DEN-IM, and several components overlap with DEN-IM's, but the retrieval of viral sequences is not targeted for DENV, and no serotyping and genotyping is performed. VIP is, overall, the most similar to DEN-IM by performing viral identification (although it is not specific for DENV), assembly and phylogenetic analysis against the reference database and producing an HTML report with the results obtained. It is not possible to customize VIP′s database to target only DENV sequences with genotyping information. VirusTAP is a web server for the identification of viral reads using the ViPR and IRD databases, or alternatively with the RefSeq Virus database. GenomeDetective is also a web service that provides two tools, one for the assembly of viral sequences from raw data (Virus tool) and another for serotyping and genotyping of DENV FASTA sequences (Dengue Typing tool). Both tools need to be run consecutively, with the

Table 1. Comparison of CDEN-IM with different tools for the identification and genotyping of DENV from sequencing data

Tool	Interface	Quality control	DENV read selection	Assembly	DENV sero- and genotyping	Phylogeny	Report
DEN-IM	CLI	✓	✓	✓	✓	✓	✓ (all samples)
VIP	CLI	✓	✓†	✓	✗	✓	✓
VirusTAP	Web	✓	✓†	✓	✗	✗	✓ (one per sample, downloadable)
drVM	CLI/GUI*	✓	✓†	✓	✗	✗	✗
GenomeDetective Virus tool	Web	✓	✗	✓	✗	✗	✓ (one per sample)
GenomeDetective Dengue Typing tool	Web	✗	✗	✗	✓‡	✓§	✓ (one per sample)

*GUI only available on a virtual machine.

†Targeted for viral sequences, but not specific for DENV.

‡Sequence file can be received from GenomeDetective Virus tool, as well as independently uploaded.

§Limited to the positioning of a sample in a tree of static representative isolates.

Virus Tool providing a link to redirect to the Dengue Typing tool when a DENV sequence is identified.

Of all the tools listed in Table 1, only Genome Detective offers a tool to determine the DENV sero- and genotypes from a FASTA sequence, but the need to run their virus identification tool beforehand to obtain a sequence from the raw sequencing data increases the time required to obtain a typing result, especially when a large number of sequences need to be analysed. DEN-IM provides the same information of the Genome Detective Virus Typing tool, with the addition of a phylogenetic tree with all samples analysed plus automatic selection of the closest genomes present in the database (optional) and NCBI DENV references (optional). Moreover, these tools are not open source, so we are unable to compare the methodology used with our own. Additionally, there might be privacy issues in submitting data to external services, such as VirusTAP and GenomeDetective, especially when handling metagenomics data that contain human sequences subjected to strict privacy laws in most countries. Therefore, a stand-alone tool is preferable for these analyses since these can be run in secure local environments. DEN-IM's main advantage when compared to web-based platforms is the ability to analyse batches of samples in a scalable manner, obtaining a report summarizing all the samples analysed and a phylogeny analysis of all DENV CDSs recovered.

RESULTS

To evaluate the DEN-IM workflow performance, we analysed three datasets, one containing shotgun metagenomics sequencing data from patient samples (see Table S1), a second with amplicon sequencing data, a set with 106 paired-end samples obtained from Parameswaran *et al.* [27] and another set with 78 single-end samples available under BioProject PRJNA321963, and a third dataset of publicly available sequences, both from amplicon and shotgun metagenomics, containing 45 Chikungunya virus (CHIKV) samples, 66 Zika virus (ZKV) samples and 21 yellow fever virus (YFV) samples (see Table S2). All analyses were executed with the default resources and parameters (available at https://github.com/B-UMMI/DEN-IM). In the shotgun metagenomics and the single-end amplicon sequencing datasets the closest typing reference in the final tree and the NCBI DENV references for each serotype were included in the phylogenetic analysis. The resulting reports for each dataset are available on Figshare at https://doi.org/10.6084/m9.figshare.11316599.v1.

Shotgun metagenomics dataset

We analysed a dataset containing 22 shotgun metagenomics paired-end short-read Illumina sequencing samples from positive dengue cases, 1 positive control (purified from a DENV culture), 1 negative control (blank) and an *in vitro* spiked sample containing the 4 DENV serotypes (see DEN-IM_Supplemental_material.pdf, Shotgun Metagenomics Sequencing Data). On average, each sample took 7 min to analyse. A total of 75 CPU hours were used to analyse the 25 samples, with a total size of 17 GB. This analysis resulted in 69 GB of data.

The negative control and the 92–1001 sample had no reads after trimming and filtering of low-complexity reads, and so they were removed from further analysis (see Table S3). When mapping to the DENV mapping database, the percentage of DENV reads in the 21 clinical samples, the positive control and the spiked sample passing QC ranged from 0.01 % (sample UCUG0186) to 85.38 % (positive control – PC – sample). After coverage depth estimation, the analysis of the samples 91–0115 and UCUG0186 was terminated due to a low proportion of DENV reads (0.05 and 0.01 %, respectively). They failed to meet the threshold criterion of having an estimated depth of coverage of ≥10× (estimated coverages of 3.17× and 5.65×, respectively). The sequence data from sample 91–0106 only contained 960 DENV reads (0.03 %), but these were successfully assembled into a CDS with an estimated depth of coverage of 14.71×.

In the assembly module, the remaining 19 samples, the spiked sample and the PC were assembled with DEN-IM's two assembler approach. Twenty-four full CDSs were assembled (see Fig. S2), even in samples originally having DENV read content of as low as 0.03 % of the total reads. Sixteen samples, including the spiked sample and the positive control, were assembled in the first step with the SPAdes assembler, and five were assembled in the second step with the MEGAHIT assembler. In the spiked sample, all four CDSs were successfully assembled and recovered.

Serotype and genotype were successfully determined for the 24 DENV CDSs by BLAST (see Fig. S2). The most common were serotype 2 genotype III (Asian-American) and serotype 4 genotype II, with eight samples each (33 %), followed by serotype 3 genotype III (n=5, 21 %), serotype 1 genotype V (n=2, 8 %) and serotype 2 genotype V (Asian I) (n=1, 4 %). All CDSs recovered and the respective closest reference genome in the typing database were aligned and a maximum-likelihood phylogenetic tree was obtained to visualize the relationship between the samples (Fig. 2). There was a perfect concordance between the serotyping and genotyping results and the major groups in the tree.

Four distinct CDSs were assembled for the spiked sample that resulted in different coverages of each serotype CDS (2032× times coverage for DENV-2, 229× coverage for DENV-1, 76× coverage for DENV-3 and 30× times coverage for DENV-4), in accordance with the ranking order of the real-time RT-PCR results (see DEN-IM_Supplemental_material.pdf, Shotgun Metagenomics Sequencing Data).

The amplicon sequencing dataset

To validate DEN-IM's performance in an amplicon sequencing approach, a dataset of 106 paired-end HTS samples of PCR products using primers targeting DENV-3 [27] were analysed (see DEN-IM_Supplemental_material.pdf, Amplicon Sequencing Data). On average, each sample took 5 min to analyse. The 106 samples, 51 GB in size, took 3622 CPU hours to analyse, resulting in 424 GB of data.

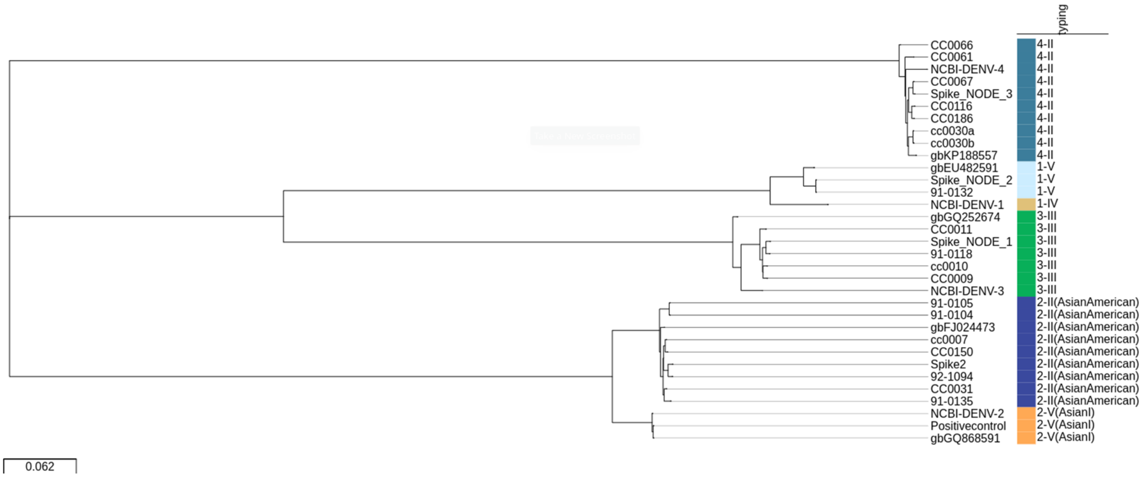


Fig. 2. Phylogenetic reconstruction of the shotgun metagenomic dataset. Maximum-likelihood tree in the DEN-IM report for the 24 complete CDSs (*n*=21 samples) obtained with the metagenomics dataset, the respective closest references in the typing database (identified by their GenBank ID) and the NCBI DENV references for each serotype (NCBI-DENV-1, NC_001477.1; NCBI-DENV-2, NC_001474.2; NCBI-DENV-3, NC_001475.2; NCBI-DENV-4, NC_002640.1). The tree is midpoint-rooted for visualization purposes and the scale represents the average substitutions per site. The colours depict the DENV genotyping results.

No samples failed the quality control block (see Table S4). The proportion of DENV reads ranged from 24.72 % (SRR5821236) to 99.81 % (SRR5821254) of the total processed reads. The samples with less than 70 % DENV DNA were profiled taxonomically with Kraken2 [28] and the minikraken2_v2 database (ftp://ftp.ccb.jhu.edu/pub/data/kraken2_dbs/) and the source of contamination was determined to have come largely from human DNA (see Table S5).

Of the 106 samples, 43 (41 %) managed to assemble a complete CDS sequence (see Table S4), whereas a mapping approach was used for the remaining 63 samples (60 %) and a consensus CDS was generated. For the assembled CDSs, all but one were assembled with MEGAHIT after not producing a full CDS with SPAdes. Moreover, pronounced variation in the size of the assembled contigs is evident in the contig size distribution plot (see Fig. S3).

All 106 CDSs recovered belonged to serotype 3 genotype III. Despite the same classification, the maximum-likelihood tree indicates that there is detectable genetic diversity within the dataset (486 SNPs in 10 237 nucleotides) (Fig. 3).

A second amplicon dataset, containing 78 DENV-1 single-end samples recovered from different *A. aegypti* isofemale hosts were analysed (see DEN-IM_Supplemental_material.pdf, Amplicon Sequencing Data). On average, each sample took 3 min to analyse. The 78 samples, 19 GB in size, took 278 CPU hours to be analysed, resulting in 203 GB of data.

No samples failed the quality control block and the proportion of DENV reads ranged from 59 (SRR3539343) to 96 % (SRR3539408) of the total processed reads (see Table S6). Of the 78 samples, 53 (68 %) assembled a complete CDS sequence and in the remaining 25 (32 %) the complete CDS was obtained through mapping. All CDSs recovered, the respective closest reference genome in the typing database and NCBI's references for each DENV serotype were aligned and a maximum-likelihood phylogenetic tree was obtained (Fig. 4). All 78 samples belonged to serotype 1 genotype I and, similarly to the previous dataset of 106 samples, there was detectable genetic diversity within the dataset (651 SNPs in 10 808 nucleotides, excluding reference sequences).

The non-DENV arbovirus dataset

In order to evaluate DEN-IM's specificity to DENV sequences, a third dataset of publicly available sequences of arbovirus other than DENV, from both amplicon and shotgun metagenomics, was analysed containing 45 CHIKV samples, 66 ZKV samples and 21 YFV samples (see Table S2). All 132 samples failed DEN-IM's workflow, 16 due to insufficient sequencing data remaining after quality trimming, and the remaining 116 due to very low estimated coverage of the DENV genome (less than 0.01×), as expected.

Conclusion

We have successfully analysed two DENV datasets, one comprising 25 shotgun metagenomics sequencing samples and another comprising 106 paired-end and 78 single-end targeted metagenomics samples.

In the first dataset, we recovered 24 CDSs from 19 clinical samples, including a spiked sample and a positive control that were correctly serotyped and genotyped. Besides the negative control, three samples did not return typing information due to failing quality checks.

The proportion of DENV reads in the metagenomics samples was highly variable. This may reflect the viral load in patients in which DENV was detected by real-time RT-PCR. In the spiked

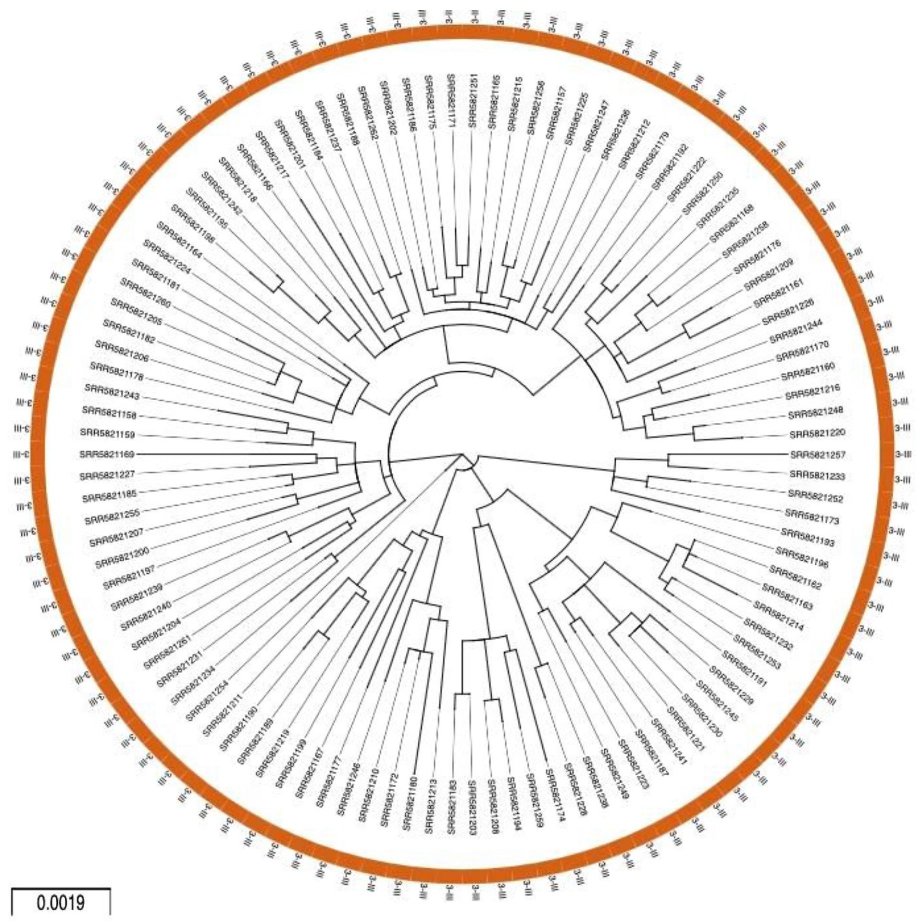


Fig. 3. Phylogenetic reconstruction of the paired-end targeted metagenomic dataset. Maximum-likelihood circular tree in the DEN-IM report for the 106 complete CDSs obtained with the targeted metagenomics dataset (*n*=106). All samples belong to serotype 3 genotype III. The scale represents the average substitutions per site.

sample, containing four distinct DENV serotypes, all four were correctly detected despite not being present in equal concentrations, highlighting the potential of the DEN-IM workflow to accurately detect and recover multiple DENV genomes from samples with DENV co-infection, even if the serotypes are present in low abundance. Indeed, recent studies from areas of high endemicity suggest that co-infection with multiple DENV serotypes may occur frequently [29, 30] and the co-circulation of different DENV strains of the same serotype, but distinct genotypes, in these areas [29] raises the possibility of simultaneous infection with more than one genotype.

When analysing the 106 paired-end targeted metagenomics dataset, only 43 CDS samples were *de novo* assembled. For the remaining 63 samples, consensus sequences were obtained through mapping. In all samples DENV 3-III was correctly identified. Similar results were obtained for the 78 single-end samples where 53 CDS were *de novo* assembled, and 25 consensus sequences were obtained through mapping. All samples were identified as DENV-1 I. These two datasets demonstrate the success of DEN-IM's two-pronged approach of combining assembler and mapping. DEN-IM's specificity was shown when it found no false-positive results when analysing a dataset containing arboviruses other than DENV.

DEN-IM is built with modularity and containerization as keystones, leveraging the parallelization of processes and guaranteeing reproducible analyses across platforms. The modular design allows for new modules to be easily added and tools that become outdated to be easily updated, ensuring DEN-IM's sustainability. The software versions are also described in the Nextflow script and configuration files, and in the Docker files for each container, allowing the traceability of each step of data processing.

Having been developed in Nextflow, DEN-IM runs on any UNIX-like system and provides out-of-the-box support for several job schedulers (e.g. PBS, SGE, SLURM) and integration with containerized software such as Docker or Singularity. While it has been developed to be ready to use by non-experts, not requiring any software installation or parameter tuning, it can still be easily customized through the configuration files.

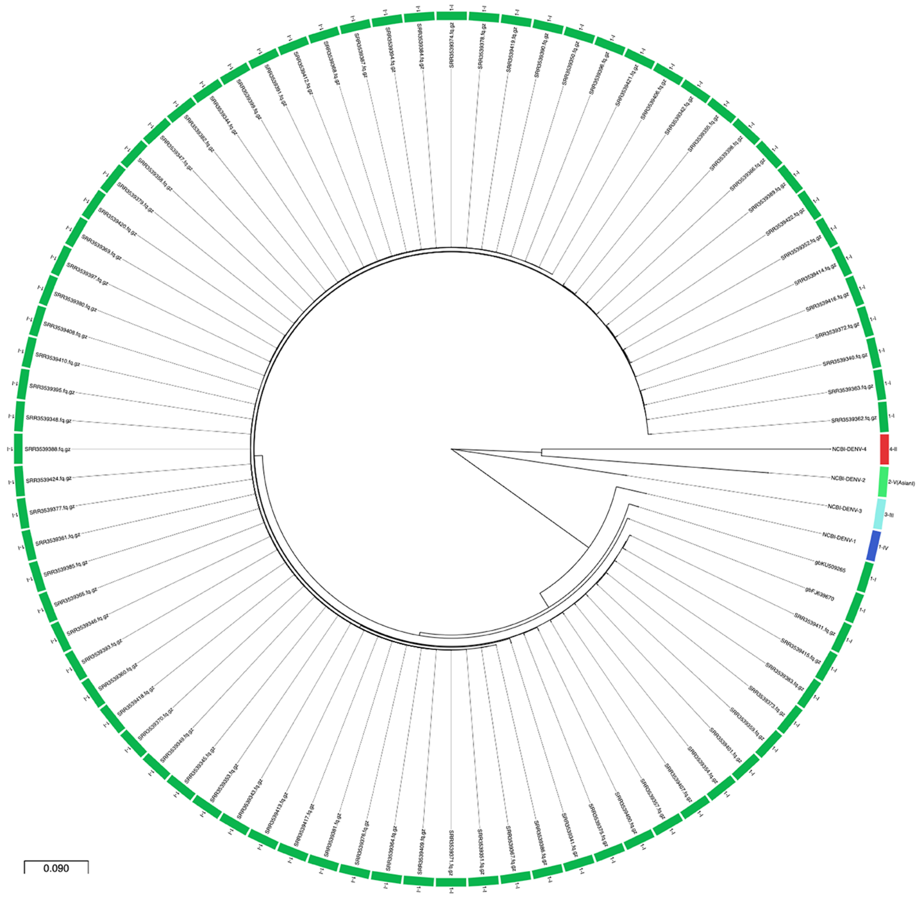


Fig. 4. Phylogenetic reconstruction of the single-end targeted metagenomic dataset. Maximum-likelihood circular tree in the DEN-IM report for the 78 complete CDSs obtained with the targeted metagenomics dataset (n=78) and the NCBI DENV references for each serotype (NCBI-DENV-1, NC_001477.1; NCBI-DENV-2, NC_001474.2; NCBI-DENV-3, NC_001475.2; NCBI-DENV-4, NC_002640.1). All samples belong to serotype 1 genotype I. The scale represents the average substitutions per site.

The interactive HTML reports (see Fig. S1) provide an intuitive platform for data exploration, allowing the user to highlight specific samples, filter and reorder the data tables, and export the plots as needed.

Together with the workflow and software containers, a database containing 3858 complete DENV genomes for DENV sequence retrieval and a subset database with 161 curated DENV genomes for serotyping and genotyping are provided. While constructing these databases, the obstacles reported by Cuypers *et al.* [31] were apparent, namely the lack of formal definition of a DENV genotype and the lack of a standardized classification procedure that could assign sequences to a previously defined genotypic/sub-genotypic clade [31]. Discrepancies between the phylogenetic relationship and the genotype assignment were frequent and, throughout this study, the classification of some strains within the ViPR database [32] was updated. As suggested previously [31], further evaluation of DENV classification will benefit future research and investigation into the population dynamics of this virus. Our typing approach was designed to use the currently accepted DENV classification. However, DEN-IM can be easily modified if a new DENV classification system is to be established in the future.

DEN-IM provides a user-friendly workflow that makes it possible to analyse short-read raw sequencing data from shotgun or targeted metagenomics for the presence, typing and phylogenetic analysis of DENV. The use of containerized workflows, together with shareable reports, will allow an easier comparison of results globally, promoting collaborations that can benefit the populations where DENV is endemic. The DEN-IM source code is freely available in the DEN-IM GitHub repository (https://github.com/B-UMMI/DEN-IM), which includes a wiki with full documentation and easy-to-follow instructions.

Funding information
C. I. M. was supported by the Fundação para a Ciência e Tecnologia (grant SFRH/BD/129483/2017). E. L. received a Abel Tasman Talent Program grant from the UMCG, University of Groningen, Groningen, The Netherlands. This work was partly supported by the ONEIDA project (LISBOA-01–0145-FEDER-016417) co-funded by FEEI–Fundos Europeus Estruturais e de Investimento from Programa Operacional

Regional Lisboa 2020 and by national funds from FCT–Fundação para a Ciência e a Tecnologia and by UID/BIM/50005/2019, project funded by Fundação para a Ciência e a Tecnologia (FCT)/ Ministério da Ciência, Tecnologia e Ensino Superior (MCTES) through Fundos do Orçamento de Estado.

Acknowledgements

The authors would like to thank Tiago F. Jesus and Bruno Ribeiro-Gonçalves for their invaluable help with the Nextflow implementation. We would also like to thank Erwin C. Raangs from the UMCG for his assistance with the sequencing of the shotgun metagenomics dataset. Additionally, the authors thank Lize Cuypers, Krystof Theys, Pieter Libin and Gilberto Santiago for their discussions on DENV nomenclature and classification. This work was performed in collaboration with the ESCMID Study Group on Molecular and Genomic Diagnostics (ESGMD), Basel, Switzerland.

Author contributions

C. I. M., E. L., N. C., M. R., J. A. C. and J. W. A. R. designed the workflow. C.I.M implemented and optimized the workflow, created the Docker containers and wrote the manuscript. M. P. M. implemented the DENV genotyping module in the workflow and D. N. S. contributed to the development of DEN-IM's HTML report. E. L., A. T. and N. C. provided the shotgun metagenomics data used to test and validate the workflow and wrote the manuscript. A. T., N. C., M. R., J. A. C. and J. W. A. R. critically revised the article. All authors read, commented on and approved the final manuscript.

Conflicts of interest

The authors declare that there are no confilcts of interest.

Ethical statement

This study followed international standards for the ethical conduct of research involving human subjects. Data and sample collection were carried out within the DENVEN and IDAMS (International Research Consortium on Dengue Risk Assessment, Management and Surveillance) projects. The study was approved by the Ethics Review Committee of the Biomedical Research Institute, Carabobo University (Aval Bioetico #CBIIB(UC)–014 and CBIIB-(UC)–2013–1), Maracay, Venezuela; the Ethics, Bioethics and Biodiversity Committee (CEBioBio) of the National Foundation for Science, Technology and Innovation (FONACIT) of the Ministry of Science, Technology and Innovation, Caracas, Venezuela; the regional health authorities of Aragua state (CORPOSALUD Aragua) and Carabobo state (INSALUD); and by the Ethics Committee of the Medical Faculty of Heidelberg University and the Oxford University Tropical Research Ethics Committee.

Data bibliography

1. Catarina Inês Mendes. DEN-IM supplemental material and tables are deposited at Figshare with DOI; https://doi.org/10.6084/m9.figshare.11316599.v1.

2. Catarina Inês Mendes. DEN-IM reports for the analysed datasets tables are deposited at Figshare with DOI; https://doi.org/10.6084/m9.figshare.11316599.v1.

3. Catarina Inês Mendes. Phylogeny inference trees for the dengue virus typing database are deposited at Figshare with DOI; https://doi.org/10.6084/m9.figshare.11316599.v1.

4. Catarina Inês Mendes. Code for the DEN-IM workflow (https://github.com/B-UMMI/DEN-IM).

References

1. World Health Organization. Dengue: guidelines for diagnosis, treatment, prevention, and control. *Spec Program Res Train Trop Dis* 2009;x:147.
2. Diamond MS, Pierson TC. Molecular insight into dengue virus pathogenesis and its implications for disease control. *Cell* 2015;162:488–492.
3. Bhatt S, Gething PW, Brady OJ, Messina JP, Farlow AW *et al*. The global distribution and burden of dengue. *Nature* 2013;496:504–507.
4. Lourenço J, Tennant W, Faria NR, Walker A, Gupta S *et al*. Challenges in dengue research: a computational perspective. *Evol Appl* 2018;11:516–533.
5. Leitmeyer KC, Vaughn DW, Watts DM, Salas R, Villalobos I *et al*. Dengue virus structural differences that correlate with pathogenesis. *J Virol* 1999;73:4738–4747.
6. Yozwiak NL, Skewes-Cox P, Stenglein MD, Balmaseda A, Harris E *et al*. Virus identification in unknown tropical febrile illness cases using deep sequencing. *PLoS Negl Trop Dis* 2012;6:e1485.
7. Lee CK, Chua CW, Chiu L, Koay ES-C. Clinical use of targeted high-throughput whole-genome sequencing for a dengue virus variant. *Clin Chem Lab Med* 2017;55:e209.
8. Fatima Z, Idrees M, Bajwa MA, Tahir Z, Ullah O *et al*. Serotype and genotype analysis of dengue virus by sequencing followed by phylogenetic analysis using samples from three mini outbreaks-2007-2009 in Pakistan. *BMC Microbiol* 2011;11:200.
9. Fonseca V, Libin PJK, Theys K, Faria NR, Nunes MRT *et al*. A computational method for the identification of dengue, Zika and Chikungunya virus species and genotypes. *PLoS Negl Trop Dis* 2019;13:e0007231.
10. Vilsker M, Moosa Y, Nooij S, Fonseca V, Ghysens Y *et al*. Genome detective: an automated system for virus identification from high-throughput sequencing data. *Bioinformatics* 2019;35:871–873.
11. Li Y, Wang H, Nie K, Zhang C, Zhang Y *et al*. Vip: an integrated pipeline for metagenomics of virus identification and discovery. *Sci Rep* 2016;6:1–10.
12. Yamashita A, Sekizuka T, Kuroda M. VirusTAP: viral Genome-Targeted assembly pipeline. *Front Microbiol* 2016;7:1–5.
13. Lin H-H, Liao Y-C. drVM: a new tool for efficient genome assembly of known eukaryotic viruses from metagenomes. *Gigascience* 2017;6:1–10.
14. Di Tommaso P, Chatzou M, Floden EW, Barja PP, Palumbo E *et al*. Nextflow enables reproducible computational workflows. *Nat Biotechnol* 2017;35:316–319.
15. Gerhardt L, Bhimji W, Canon S, Fasel M, Jacobsen D *et al*. Shifter: containers for HPC. *J Phys Conf Ser* 2017;898:082021.
16. Kurtzer GM, Sochat V, Bauer MW. Singularity: scientific containers for mobility of compute. *PLoS One* 2017;12:e0177459–20.
17. Schmieder R, Edwards R. Quality control and preprocessing of metagenomic datasets. *Bioinformatics* 2011;27:863–864.
18. Langmead B, Salzberg SL. Fast gapped-read alignment with Bowtie 2. *Nat Methods* 2012;9:357–359.
19. Li H. A statistical framework for SNP calling, mutation discovery, association mapping and population genetical parameter estimation from sequencing data. *Bioinformatics* 2011;27:2987–2993.
20. Bankevich A, Nurk S, Antipov D, Gurevich AA, Dvorkin M *et al*. SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing. *J Comput Biol* 2012;19:455–477.
21. Li D, Liu C-M, Luo R, Sadakane K, Lam T-W. MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph. *Bioinformatics* 2015;31:1674–1676.
22. Walker BJ, Abeel T, Shea T, Priest M, Abouelliel A *et al*. Pilon: an integrated tool for comprehensive microbial variant detection and genome assembly improvement. *PLoS One* 2014;9:e112963.
23. Machado MP, Ribeiro-Gonçalves B, Silva M, Ramirez M, Carriço JA. Epidemiological surveillance and typing methods to track antibiotic resistant strains using high throughput sequencing. *Methods Mol Biol* 2017;1520:331–356.
24. Altschul S*et al*. Gapped blast and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res* 1997;25:3389–3402.
25. Nakamura T, Yamada KD, Tomii K, Katoh K. Parallelization of MAFFT for large-scale multiple sequence alignments. *Bioinformatics* 2018;34:2490–2492.
26. Stamatakis A. RAxML version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics* 2014;30:1312–1313.

27. **Parameswaran P**, **Wang C**, **Trivedi SB**, **Eswarappa M**, **Montoya M** *et al*. Intrahost selection pressures drive rapid dengue virus microevolution in acute human infections. *Cell Host Microbe* 2017;22:400–410.
28. **Wood DE**, **Salzberg SL**. Kraken: ultrafast metagenomic sequence classification using exact alignments. *Genome Biol* 2014;15:R46.
29. **Marinho PES**, **Bretas de Oliveira D**, **Candiani TMS**, **Crispim APC**, **Alvarenga PPM** *et al*. Meningitis associated with simultaneous infection by multiple dengue virus serotypes in children, Brazil. *Emerg Infect Dis* 2017;23:115–118.
30. **Reddy MN**, **Dungdung R**, **Valliyott L**, **Pilankatta R**. Occurrence of concurrent infections with multiple serotypes of dengue viruses during 2013–2015 in northern Kerala, India. *PeerJ* 2017;5:e2970.
31. **Cuypers L**, **Libin P**, **Simmonds P**, **Nowé A**, **Muñoz-Jordán J** *et al*. Time to harmonize dengue Nomenclature and classification. *Viruses* 2018;10:569.
32. **Pickett BE**, **Greer DS**, **Zhang Y**, **Stewart L**, **Zhou L** *et al*. Virus pathogen database and analysis resource (ViPR): a comprehensive bioinformatics database and analysis resource for the coronavirus research community. *Viruses* 2012;4:3209–3226.

GigaScience, 2022, **0**, 1–16
DOI: 10.1093/gigascience/giac122
Technical Note

LMAS: evaluating metagenomic short *de novo* assembly methods through defined communities

Catarina Inês Mendes [1,*], Pedro Vila-Cerqueira [1], Yair Motro [2], Jacob Moran-Gilad [2], João André Carriço [1] and Mário Ramirez [1]

[1]Instituto de Microbiologia, Instituto de Medicina Molecular, Faculdade de Medicina, Universidade de Lisboa, 1649-028 Lisboa, Portugal
[2]Faculty of Health Sciences, Ben-Gurion University of the Negev, 8410501 Beer-Sheva, Israel
*Correspondence address. Catarina I. Mendes, Instituto de Microbiologia, Instituto de Medicina Molecular, Faculdade de Medicina, Universidade de Lisboa, 1649-028 Lisboa, Portuga. E-mail: cimendes@medicina.ulisboa.pt

Abstract

Background: The *de novo* assembly of raw sequence data is key in metagenomic analysis. It allows recovering draft genomes from a pool of mixed raw reads, yielding longer sequences that offer contextual information and provide a more complete picture of the microbial community.

Findings: To better compare *de novo* assemblers for metagenomic analysis, LMAS (Last Metagenomic Assembler Standing) was developed as a flexible platform allowing users to evaluate assembler performance given known standard communities. Overall, in our test datasets, *k*-mer De Bruijn graph assemblers outperformed the alternative approaches but came with a greater computational cost. Furthermore, assemblers branded as metagenomic specific did not consistently outperform other genomic assemblers in metagenomic samples. Some assemblers still in use, such as ABySS, MetaHipmer2, minia, and VelvetOptimiser, perform relatively poorly and should be used with caution when assembling complex samples. Meaningful strain resolution at the single-nucleotide polymorphism level was not achieved, even by the best assemblers tested.

Conclusions: The choice of a *de novo* assembler depends on the computational resources available, the replicon of interest, and the major goals of the analysis. No single assembler appeared an ideal choice for short-read metagenomic prokaryote replicon assembly, each showing specific strengths. The choice of metagenomic assembler should be guided by user requirements and characteristics of the sample of interest, and LMAS provides an interactive evaluation platform for this purpose. LMAS is open source, and the workflow and its documentation are available at https://github.com/B-UMMI/LMAS and https://lmas.readthedocs.io/, respectively.

Keywords: shotgun metagenomics, *de novo* assembly, benchmark, draft genome quality, simulation

Background

Short-read shotgun metagenomics has the potential to offer comprehensive microbial detection and characterization of complex clinical or environmental samples. Despite becoming an increasingly used approach, it comes at the cost of producing massive amounts of data that require expert handling and processing, as well as adequate computational resources. The *de novo* assembly process is key when analyzing metagenomic data since it allows recovering contigs representing the replicons present in the sample, be it prokaryotic chromosomes, plasmids, or viruses, from a pool of mixed raw reads. These contigs are longer sequences that offer better contextual information than reads alone and provide a more complete picture of the microbial community than the species composition. Despite efforts for the development, standardization, and assessment of software for metagenomic analysis, both commercial and open source [1–5], the *de novo* assembly process still represents a critical point in these analyses.

The assembly of draft genomes has become a central step when analyzing pure bacterial cultures, for instance, allowing genomic comparisons through single-nucleotide polymorphisms (SNPs) or gene-by-gene methods, such as core-genome multilocus sequence typing (cgMLST). de Bruijn graph (dBg) algorithms are currently the most widely used approaches in modern assembly software. dBg handles unresolvable repeats by essentially fragmenting the sequence, that is, forming multiple contigs for each of the possibly contiguous sequences present in the sample. Additionally, the inherent heterogeneity of complex samples, potentially containing a multitude of replicons, could make traditional genome assemblers, implementing optimizations based on the assumption of having a single genome in the sample, not suitable for metagenomics [6].

Several dedicated metagenomic assembly tools for short-read data are available [6]. These tools are generally assumed to perform better when dealing with complex samples having a combination of intragenomic and intergenomic repeats and uneven and low-coverage sequencing depths of some of the replicons [7]. Not using dedicated metagenomic assemblers was suggested to come with the cost of generating artificial variation and chimeric contigs, especially in samples that contain closely related species [8]. However, no formal comparison has looked at increased accuracy or gains in contiguity of assemblies obtained with metagenomic assemblers versus traditional assemblers.

With an ever-increasing range of both traditional and metagenomic assemblers becoming available, choosing the best-performing tool can be an arduous and time-consuming task since the choice may vary depending on the purpose of the

Received: May 4, 2022. **Revised:** September 26, 2022. **Accepted:** November 16, 2022

analysis, organism of interest, complexity of the sample, and computational infrastructure available. Additionally, the evaluation of the resulting contigs is not straightforward since one metric is not sufficient to classify an assembly, particularly with complex samples [7, 9]. Despite several *de novo* assembly validation methods relying on features of the created contigs themselves, such as QUAST [10], being useful in identifying inconsistencies indicative of potential assembly errors, the use of reference-based validation methods offers the possibility of a more complete evaluation of accuracy and is particularly important to benchmark attempts to reconstruct communities. MetaQUAST [11], a modification of QUAST, extends the original software by performing assembly evaluation based on aligning contigs to a reference, which can be provided or inferred by the software, and reports, in addition to the standard metrics for single genomes reported by QUAST, the number of interspecies translocations and the number of possibly misassembled contigs.

The use of mock communities, with known composition, abundance, and genomic information, provides a ground truth against which the success of the assembly of a complex sample can be evaluated. Furthermore, this can be done in circumstances in which the errors introduced by sample processing and the sequencing and associated methods used approximate, as much as possible, their effects in real samples. Such mock communities facilitate the identification of misassemblies, such as chimeric sequences generated from the improper combination of 2 distinct replicons, indels, or single-nucleotide variants improperly created by the assembler. On the other hand, the comparison of the performance of 2 assemblers is only possible if the input data are the same and if the same evaluation metrics are applied [3].

To tackle these challenges, we developed LMAS (Last Metagenomic Assembler Standing), an automated workflow to enable the benchmarking of traditional and metagenomic prokaryotic *de novo* assembly software using defined mock communities. The results of LMAS are presented in an interactive HTML report where selected global and reference replicon-specific performance metrics can be explored. The mock communities can be provided by the user to better reflect the samples of interest. New assemblers can be added with minimal changes to the pipeline so that LMAS can be expanded to include novel algorithms as they are developed. The portability and ease of use of LMAS are intended to provide users with a continuous benchmarking platform to easily evaluate the performance of assemblers, in mock communities mimicking as closely as possible their samples of interest.

The LMAS Workflow

Workflow overview

LMAS is a user-friendly automated workflow enabling the benchmarking of traditional and metagenomic prokaryotic *de novo* assembly software using defined mock communities (Fig. 1). LMAS was implemented in Nextflow [12] to provide flexibility and ensure the transparency and reproducibility of the results. LMAS relies on the use of Docker [13] containers for each assembler, allowing versions to be tracked and changed easily.

Installation and usage

LMAS can be installed through Bioconda [14] or GitHub [15], with detailed instructions available in the documentation [16]. LMAS requires as inputs the complete reference replicons (genomes, plasmids, or any other replicons present) and short-read paired-end raw data. All complete references (linear replicons) should be provided in a single file. These raw data can be obtained *in silico* by creating simulated reads from the reference replicons or sequencing mock communities of known composition. Optionally, information on the input samples in a markdown file can be provided to be presented in the report.

A step-by-step execution tutorial is available at [17]. Users can customize the workflow execution either by using command-line options or by modifying the simple plain-text configuration files. To make the execution of the workflow as simple as possible, a set of default parameters and directives is provided. A complete description of each parameter is available in Supplementary Material (see Supplementary Material, Workflow parameters), as well as in the documentation [18]. The results are presented in an interactive HTML report, stored in the "*report*" folder in the directory of LMAS's execution. The output files of all assemblers and quality assessment processing scripts in the workflow are stored in the "*results*" folder, in the same location.

Supported assemblers and selection criteria

A collection of *de novo* assembly tools was compiled, including OLC and dBg assembly algorithms, the latter including both single *k*-mer and multiple *k*-mer value approaches, and hybrid assemblers implementing both algorithms, including both genomic and metagenomic assemblers (Supplementary Table S1). Of these, 11 assemblers were selected based on the date of last update (at least 2015) and are implemented in LMAS: ABySS [19] (version 2.3.1), GATB Minia Pipeline [20] (commit hash 9d56f42), IDBA-UD [21] (version 1.1.3), MEGAHIT [22] (version 1.2.9), MetaHipMer2 [23] (version 2.0.0.65-gaad446d-dirty-AddGtest), metaSPAdes [24] (version 3.15.3), minia [25] (version 3.2.6), SKESA [26] (version 2.5.0), SPAdes [27] (version 3.15.3), Unicycler [28] (version 0.4.9), and VelvetOptimiser [29] (commit hash 092bdee) (Table 1). The execution commands for each assembler are available as Supplementary Material (see Supplementary Material, Short-read de novo assemblers) and in the documentation [30].

New assemblers can be added with minimal changes to the pipeline so that LMAS can be expanded as novel algorithms are developed. A template is available to facilitate their integration, and a step-by-step guide is included in the documentation [31]. The only 2 requirements for the addition of a new assembler are the execution command for the assembler for paired-end short-read data and a Nextflow-compatible container with the assembler and any dependencies.

Assembly quality metrics

The success of an assembly is evaluated in 2 steps: globally (see Global metrics) and relative to each of the replicons present in the sample (see Per reference metrics). In both, the tabular presentation in the reports allows the comparison of exact values between assemblers, and the interactive plots allow a more intuitive overview and easy exploration of results. In addition to the assembly success metrics, computational resource statistics are registered for each assembler (see Supplementary Material, LMAS metrics, Computational performance metrics).

Global metrics

The computation of the global metrics is performed through statistics inherent to the complete set of contigs assembled per sample, independent of the species/sample of origin. The metrics are presented, in tabular form, for the complete set of contigs and those filtered for a minimum length, and also graphically for the contigs filtered for a minimum length. The statistics include in-

Table 1: Prokaryotic *de novo* assemblers integrated into LMAS

Assembler	Type	Algorithm
GATBMiniaPipeline	Metagenomic	Multiple k-mer de Bruijn graph
IDBA-UD	Metagenomic	Multiple k-mer de Bruijn graph
MEGAHIT	Metagenomic	Multiple k-mer de Bruijn graph
MetaHipMer2	Metagenomic	Multiple k-mer de Bruijn graph
metaSPAdes	Metagenomic	Multiple k-mer de Bruijn graph
ABySS	Genomic	Single k-mer de Bruijn graph
MINIA	Genomic	Single k-mer de Bruijn graph
SKESA	Genomic	Multiple k-mer de Bruijn graph
SPAdes	Genomic	Multiple k-mer de Bruijn graph
Unicycler	Genomic	Multiple k-mer de Bruijn graph
VelvetOptimizer	Genomic	Single k-mer de Bruijn graph

formation on contig number, size, and ambiguous bases and the proportion of reads mapping to the created contigs. Two statistics are a consolidation of per reference metrics: misassemblies (i.e., contigs that do not reflect the structural organization in the reference replicons) and the overall size of gaps in all reference replicons not covered by any contig. A more detailed description of all global metrics is available in Supplementary Material (see Supplementary Material, LMAS metrics, Global metrics).

Per reference metrics

For the computation of the reference-based metrics, only the filtered set (FS) contigs are considered, for each reference replicon in the sample. These contigs are the ones exceeding the user-defined minimum sequence length, filtered using BBTools (version 38.44). After this initial step, the contigs are mapped to the reference replicons with minimap2 [32] (version 2.22). The metrics are computed through custom Python code (see Supplementary Material, Assembly filtering and mapping) for each replicon in the file provided as input. A detailed description of all reference-based metrics is available in Supplementary Material (see Supplementary Material, LMAS metrics, Per reference metrics).

In addition to the statistics shared with the global metrics, LMAS also calculates the number of mismatches relative to each reference, the COMPASS [9] metrics, and 2 new metrics we propose: LSA and Pls.

LSA represents the fraction of the longest single alignment between a contig and the reference, relative to the reference length. The Pls, or Phred-like score, is a scoring function based on the identity of each aligned contig to the reference replicon. Similarly to the Phred quality score [33], a measure of the quality of the identification of the bases by sequencing, the Pls measures the quality of the assembly of a contig. The formula of Pls is similar to the Phred score formula but uses as the error function the identity of the base in the contig to that of the reference replicon. The formula to obtain the Pls metric per contig is Equation 1.

$$Phred\ (E) = \begin{cases} -\log(E) \times 10 & \text{if } 0 < E \leq 1 \\ 60 & \text{if } E = 0 \end{cases} \tag{1}$$

where $E = 1 - Identity$.

The LMAS report

The LMAS results are presented in an interactive HTML (Fig. 2). The LMAS report is composed of 2 main panels: a top summary panel with information on input samples (provided by the user) and the resources used during LMAS's execution and a bottom panel where selected global and reference-specific assembly metrics can be explored for each sample. LMAS constructs the HTML file after workflow completion, storing it in the “reports” folder. The report data can be easily shared between users and requires only a browser for visualization.

Summary panel

The top panel of the report contains information on the input samples and overall performance of the assemblers in LMAS, divided into 3 tabs: Overview, Performance, and About us. On the top-right corner of the report, direct links to LMAS's source repository and documentation are provided.

- *Overview*: This tab contains information on the input data, including the name and number of reads of the raw sequencing data, as well as the name of the reference file. Additional information provided by the user about the community used as input is also presented here.
- *Performance*: This tab contains a table with information on the version, the containers used, and computational performance metrics for each assembler in LMAS.
- *About us*: This tab contains information on the LMAS GitHub repositories and the LMAS development team.

Metrics panel

The bottom portion of the report contains the explorable global and reference-specific performance metrics per input raw sequencing data sample. Each sample has its own tab, and the global or reference replicon-specific metrics can be accessed in the drop-down menu.

Global metrics

A table displays the global assembly metrics computed for the complete and FS contigs. If an assembler fails to produce an assembly, it is marked on the table with a red warning sign. The global metric plots are interactive, allow zooming in on particular areas, and provide extra information as hover text boxes. The plots can be saved as PNG in whatever view the user selects.

Per reference metrics

Similarly to the global assembly metrics, a table displays the computed set of reference-restricted metrics for the FS contigs. If an assembler fails to produce sequences that align to the reference, these are marked in the table with a red warning sign. Information on the expected reference replicon length and the GC content is calculated from the input files and reported above the table. The per reference metric plots are also interactive, allowing the same type of operations as the global metric plots.

Comparison with other assembly evaluation software programs

The assessment and evaluation of genome assemblies has been a relevant field ever since the emergence of the assembly process itself, and therefore many solutions have been proposed [3, 7, 9–11, 34–36]. The Critical Assessment of Metagenome Interpretation (CAMI) proposed a set of recommendations and best practices for benchmarking in microbiome research [37]. These recommendations include the reporting of computational performance, which may condition the choice of software by the users, such as runtime, disk space, and memory consumption, also reported by LMAS (see Supplementary Material, LMAS metrics). As also suggested by CAMI, LMAS tracks the exact program version and command-line calls through its implementation in Nextflow. Moreover, using containerized assemblers and being easily installable through Bioconda, LMAS facilitates deployment in diverse user machines. Unlike the CAMI tutorial, in which users are asked to download and install the necessary tools, in LMAS, everything is provided in a one-stop reproducible workflow that effortlessly handles all preprocessing, assembly, postprocessing, traceability, and report production steps, freeing users to focus on providing relevant samples for analysis and interpreting the results in view of the intended applications.

Concerning software for assembly quality assessment currently available, the most widely adopted is QUAST [10] or, when dealing with metagenomic data, its extension metaQUAST [11], which was also adopted by the CAMI challenges [3, 5] and suggested in the CAMI Tutorial [37]. Although several features of these tools overlap with LMAS's quality assessment components, these differ from LMAS in the sense that they are not a single-step workflow allowing a traceable and reproducible assembly of mock communities. Unlike QUAST and metaQUAST, whose purpose is to evaluate assemblies, the purpose of LMAS is to allow users to evaluate assembler performance for a given sample of interest. Supplementary Table S2 shows the comparison of the output and computed assembly quality metrics generated by LMAS, QUAST, and metaQUAST.

Results and Discussion

To illustrate the use of LMAS and evaluate the performance of the chosen assemblers, we initially used the 8 bacterial genomes and 4 plasmids of the ZymoBIOMICS Microbial Community Standards as reference. As input, we used the raw sequence reads of mock communities with an even and logarithmic distribution of species, from real sequencing runs [38] and simulated read datasets, with and without error, matching the distribution of species in each sample [39]. Our dataset is composed of samples ENN (*in silico* generated evenly distributed without error), EMS (*in silico* generated evenly distributed with Illumina MiSeq error model), ERR2984773 (evenly distributed real Illumina MiSeq sample), LNN (*in silico* generated logarithmically distributed without error), LHS (*in silico* generated logarithmically distributed with Illumina HiSeq error model), and ERR2935805 (logarithmically distributed real Illumina HiSeq sample) (see Supplementary Table S3). Detailed information about the generation of the input samples is available as Supplementary Material (see Supplementary Materials, ZymoBIOMICS Microbial Community Standards, Supplementary Table S4). To evaluate the reproducibility of an assembler performance, the LMAS workflow was run 3 times for all samples using default parameters, and the resulting data were processed for each sample (see Supplementary Materials, Assessment of assembly success). Supplementary Tables S5 to S10 present an overview of the average global performance per assembler for each sample in LMAS.

To test assembler performance with an even more complex dataset, we used the 12 strain BMock community standard (accession SRX4901583, real Illumina HiSeq 2500 sample) [40]. This sample includes a noneven distribution of species, with the most abundant replicon having 3,093× coverage (*Muricauda* sp. ES.050) and the lowest only 0.1× coverage (*Micromonospora coxensis*) (Supplementary Table S24). For the sake of a less resource-intense evaluation, we downsampled to have 20% of the reads available in the original sample and further processed only these. The main challenges of this dataset are possibly to assemble the genomes of the 3 *Micromonospora* spp. and the 2 *Halomonas* spp. strains, which have an Average Nucleotide Identity through BLAST (ANIb) >0.84 and 0.98, respectively (Supplementary Table S25). The 2 Marinobacter spp. have an ANIb of 0.78.

To represent more realistic samples of a human microbiome study, the Gut-Mix-RR and Gut-Mix-HiLo standards, including 20 species known to be present in the human gut, were used as reference (accessions SRR11487941 and SRR11487935, respectively, both real Illumina MiSeq samples) [41]. For the Gut-Mix-RR, the abundance of each bacterial genome is relatively even (maximum of 66×, an average of 22.32×, and a minimum of 5.99×; Supplementary Table S28). The Gut-Mix-HiLo has an uneven abundance of species (maximum of 115×, an average of 20.45×, and a minimum of 0.34×; Supplementary Table S28). The genomes in these mock communities are fairly diverse (average ANIb = 0.67, Supplementary Table S29), with the 2 subspecies of *Bifidobacterium longum* (ANIb = 0.95) possibly being the most challenging. It is worth noting that only draft genomes are available for 8 of the strains, including one of the *B. longum* subspecies (Supplementary Table S28). The *Roseburia hominis* and *Roseburia intestinalis* are the closest related closed replicons (ANIb >0.77) in this sample.

Some assemblers perform poorly

Of the 12 *de novo* prokaryotic assemblers included in LMAS, 4 stand out as having an overall poor performance in the ZymoBIOMICS Microbial Community Standards dataset: ABySS, MetaHipmer2, minia, and VelvetOptimiser. Both ABySS and MetaHipmer2 performed inconsistently with differing resource requirements for the same sample in different runs, namely, runtime and memory allocation (see Supplementary Materials, Resource requirements differ greatly, Supplementary Fig. S2). Moreover, ABySS failed to produce an assembly for sample ERR2984773 for 1 of the runs (see Supplementary Table S7) and for sample LHS in any of the 3 runs in the time limit of 3 days (see Supplementary Table S9), and MetaHipmer2 failed to produce an assembly for samples LNN and LHS in all 3 runs (see Supplementary Tables S8–S9). VelvetOptimiser generated the highest number of inconsistent contigs across the 3 LMAS runs (Fig. 3, Supplementary Table S11), with 1.69% of the total contigs created present in only 1 or 2 runs. Although not as extreme as VelvetOptimiser, ABySS (0.52%), minia (0.14%), GATBMiniaPipeline (0.32%), MetaHipMer2 (0.11%), and IDBA-UD (0.08%) also showed inconsistencies in contig size.

Regarding the quality assessment of the assemblies produced (Fig. 4, Supplementary Table S12), ABySS and minia are the only single *k*-mer dBg assemblers in the collection and were found to mostly underperform relative to their multiple *k*-mer dBg counterparts as reported previously [3, 37, 42, 43], generally resulting in more fragmented assemblies, although there were significant differences in performance across samples. Among multi-

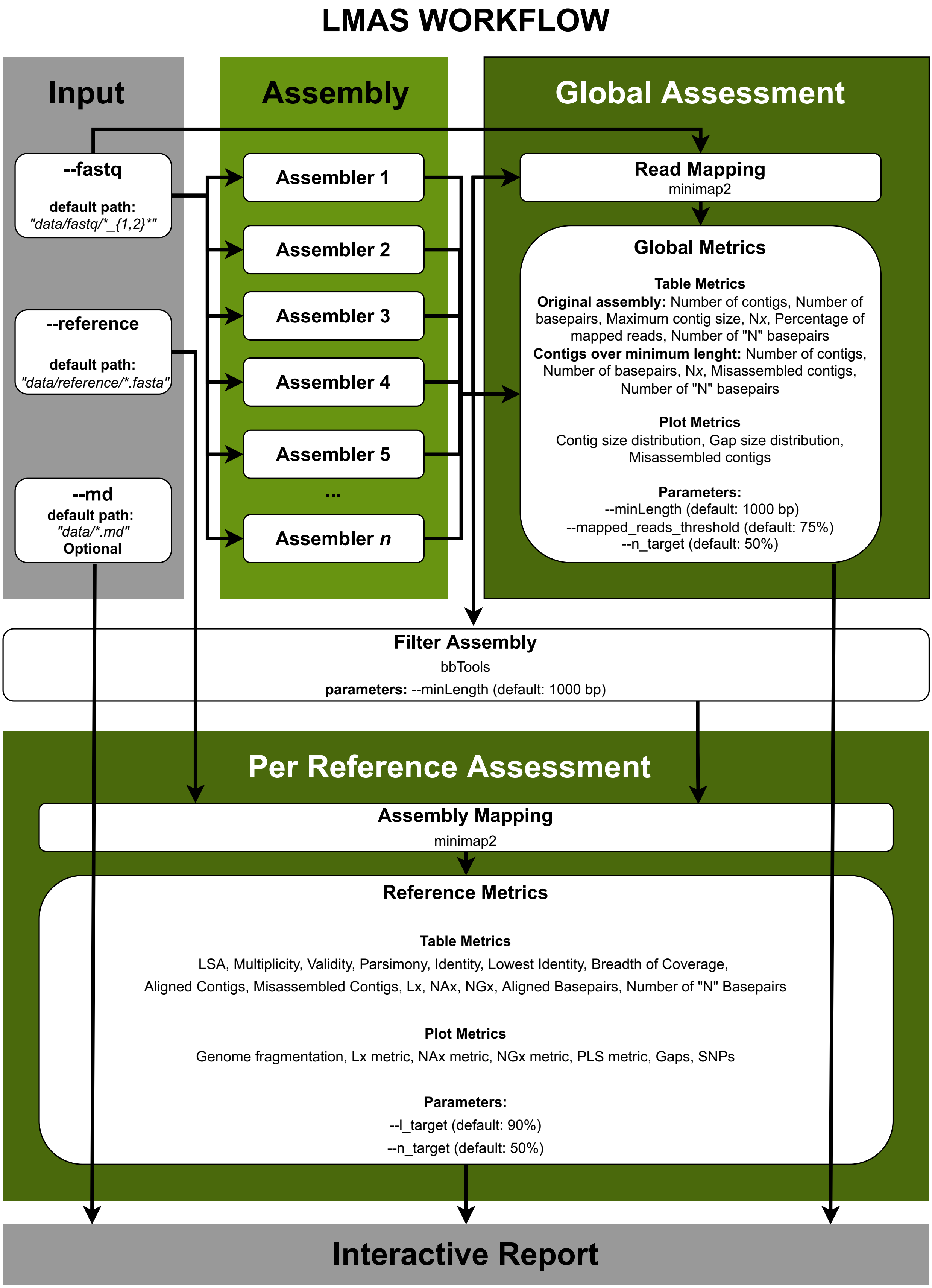
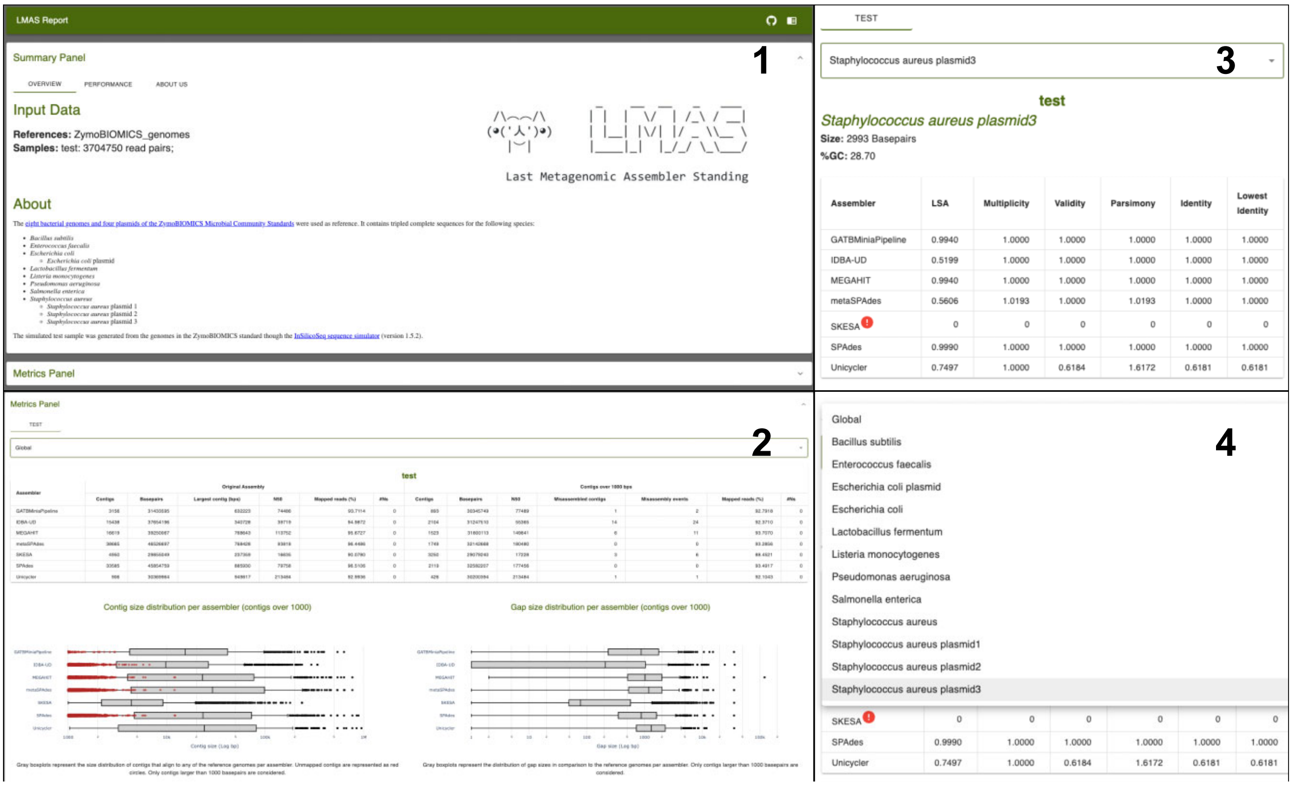


Figure 1: The LMAS workflow. The input sequencing data are assembled in parallel, resources permitting, by the set of assemblers included in LMAS. The resulting contigs are processed and the global quality assessment is performed. After filtering for the user-defined minimum contig size, the remaining sequences are mapped against the provided reference and the resulting information is processed to evaluate assembly quality by replicon in the reference file. All results, as well as optional text information describing the samples, are grouped in the LMAS report.



Input Data

References: ZymoBIOMICS_genomes
Samples: test: 3704750 read pairs;

About

The eight bacterial genomes and four plasmids of the ZymoBIOMICS Microbial Community Standards were used as reference. It contains tripled complete sequences for the following species:

- *Bacillus subtilis*
- *Enterococcus faecalis*
- *Escherichia coli*
  - *Escherichia coli* plasmid
- *Lactobacillus fermentum*
- *Listeria monocytogenes*
- *Pseudomonas aeruginosa*
- *Salmonella enterica*
- *Staphylococcus aureus*
  - *Staphylococcus aureus* plasmid 1
  - *Staphylococcus aureus* plasmid 2
  - *Staphylococcus aureus* plasmid 3

The simulated test sample was generated from the genomes in the ZymoBIOMICS standard though the InSilicoSeq sequence simulator (version 1.5.2).

test

Contig size distribution per assembler (contigs over 1000)

Gray boxplots represent the size distribution of contigs that align to any of the reference genomes per assembler. Unmapped contigs are represented as red circles. Only contigs larger than 1000 basepairs are considered.

Gap size distribution per assembler (contigs over 1000)

Gray boxplots represent the distribution of gap sizes in comparison to the reference genomes per assembler. Only contigs larger than 1000 basepairs are considered.

Staphylococcus aureus plasmid3

test

Staphylococcus aureus plasmid3

Size: 2993 Basepairs

%GC: 28.70

Assembler	LSA	Multiplicity	Validity	Parsimony	Identity	Lowest Identity
GATBMiniaPipeline	0.9940	1.0000	1.0000	1.0000	1.0000	1.0000
IDBA-UD	0.5199	1.0000	1.0000	1.0000	1.0000	1.0000
MEGAHIT	0.9940	1.0000	1.0000	1.0000	1.0000	1.0000
metaSPAdes	0.5606	1.0193	1.0000	1.0193	1.0000	1.0000
SKESA ❗	0	0	0	0	0	0
SPAdes	0.9990	1.0000	1.0000	1.0000	1.0000	1.0000
Unicycler	0.7497	1.0000	0.6184	1.6172	0.6181	0.6181

Global
Bacillus subtilis
Enterococcus faecalis
Escherichia coli plasmid
Escherichia coli
Lactobacillus fermentum
Listeria monocytogenes
Pseudomonas aeruginosa
Salmonella enterica
Staphylococcus aureus
Staphylococcus aureus plasmid1
Staphylococcus aureus plasmid2
Staphylococcus aureus plasmid3

SKESA ❗	0	0	0	0	0	0
SPAdes	0.9990	1.0000	1.0000	1.0000	1.0000	1.0000
Unicycler	0.7497	1.0000	0.6184	1.6172	0.6181	0.6181

Figure 2: The LMAS report. All results, as well as optional text information describing the samples, are grouped in the LMAS report, an interactive and responsive HTML file, for exploration in any browser. Links for LMAS source code and documentation are available in the top-right corner of the report. (1) The summary panel of the LMAS report contains information on the input reference sequences and raw sequencing data samples (provided by the user) and the overall computational performance of the assemblers in LMAS. (2) The LMAS metric panel contains the explorable global and reference-specific performance metrics per input raw sequencing data sample. The tabular presentation allows direct comparison of exact values between assemblies, and the interactive plots allow for an intuitive overview and easy exploration of results. (3) If an assembler fails to produce an assembly or fails to assemble sequences that map to the reference replicon, it is marked in the table with a red warning sign. (4) The global or reference replicon-specific metrics can be accessed for each sample in the drop-down menu.

ple *k*-mer assemblers, VelvetOptimiser frequently produced a very high number of contigs of very small size (over 99% of the contigs not surpassing the minimum length of 1,000 bp) and therefore a low N50 (an average of 29,768 bp versus a global average of 84,114 bp) (Supplementary Tables S5–S10). Additionally, ABySS and VelvetOptimizer produced contigs with a very large number of Ns, with an average of 1,019 and 3,035 uncalled bases per assembly, respectively. MetaHipMer2, although having overall average metrics in the 2 evenly distributed mock samples (ENN and EMS, Supplementary Tables S5–S6) where it was able to run successfully, severely underperformed in the real samples (ERR2984773 and ERR2935805, Supplementary Tables S7 and S10). Generally, the performance scores of the assemblers decreased considerably for the real samples in comparison with the simulated ones, either with or without error, underscoring the importance of using mock samples instead of simulated reads to evaluate assembler performance. High utilization of the reads in the dataset is observed for most assemblers, with on average at least 90% of the reads mapping back to the assembly, except for ABySS, MetaHipMer2, and VelvetOptimiser, whose values are in the range of 46% to 79%. Despite an overall good performance, SPAdes produced the highest number of misassembled contigs in the logarithmically distributed sample, with an average of 98 and a maximum of 572 (sample ERR2935805, Supplementary Table S10), in comparison to the global average of 11 misassembled contigs for all assemblers across all samples. However, this behavior was not consistent across samples, with the evenly distributed sample showing similar misassembled contigs between SPAdes and other assemblers, similarly to the other mock samples tested (see below).

Due to their poor performance discussed above, the following assemblers have not been included in subsequent analyses: ABySS, MetaHipmer2, minia, and VelvetOptimiser.

Metagenomic dedicated assemblers do not outperform genomic assemblers

After excluding the poorly performing assemblers, LMAS includes 3 genomic (SKESA, SPAdes, and Unicycler) and 4 labeled as metagenomic-specific (GATBMiniaPipeline, IDBA-UD, MEGAHIT, and metaSPAdes) *de novo* prokaryotic assemblers, all implementing multiple *k*-mer dBg algorithms. As observed in Fig. 5, Supplementary Table S13, and Supplementary Fig. S3, there were very significant differences between the best- and the worst-performing assemblers of each type for the ZymoBIOMICS Microbial Community Standards dataset, with this difference being more pronounced for metagenomic assemblers. The best-performing assemblers of each type behaved frequently quite similarly, and the differences between them tended to be attenuated after filtering for contigs <1 kbp. Still, for the linearly distributed samples (ENN, EMS, and ERR2984773), the overall worst performers tended to be metagenomic assemblers. In contrast, for the logarithmically distributed samples (LNN, LHS, and ERR2935805), the opposite was observed, with genomic assemblers tending to be the worst performing (Fig. 5). For the logarithmically distributed samples, the number of base pairs recov-

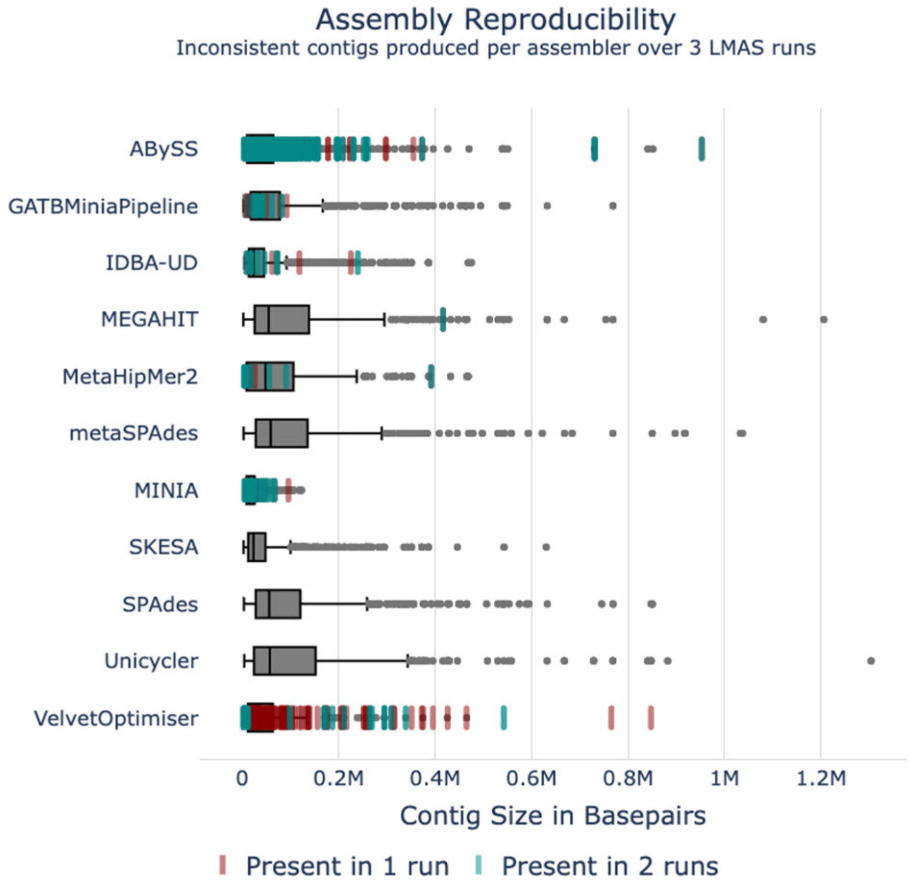


Figure 3: Assembly reproducibility. Inconsistent contigs produced per assembler over 3 LMAS runs. The distribution of contig sizes, in base pairs, consistently present in all 3 LMAS runs are indicated in the gray boxplots for each assembler. If an assembler produced a contig only present in 2 of the runs (as determined by its size), its size is indicated in teal. If a contig is present in a single run, it is represented in red.

ered is significantly lower than expected from their composition for both genomic and metagenomic assemblers, particularly after filtering (Supplementary Table S13), as contigs representing the less abundant species are not recovered by either type of assemblers (see assembler performance is influenced by replicon abundance in the sample). For this dataset, the fact that an assembler is branded as genomic or metagenomic does not translate into better or worse performance in dealing with these complex samples, but rather characteristics of the individual assemblers themselves determine their performance.

In the BMock standard, similarly to the ZymoBIOMICS standard, no significant difference was observed between the genomic and metagenomic assemblers, particularly after filtering for contigs <1 kbp (Supplementary Table S26). Both *Marinobacter* replicons (2,615,840,697 and 2,616,644,829; 448× and 135× coverage, respectively) were successfully recovered by all genomic and metagenomic assemblers with >0.87 breadth of coverage (Supplementary Table S27). The 2 most abundant *Micromonospora* replicons (2,623,620,557 and 2,623,620,567; 15× and 18× coverage, respectively) were also recovered to a breadth of coverage >0.95 by most assemblers, except the genomic assemblers SKESA and Unicycler (Supplementary Table S27). All assemblers, with the exception of SKESA and GATBMiniaPipeline, recovered both *Halomonas* replicons (2,623,620,617 and 2,623,620,618) with a breadth of coverage of >0.84 (Supplementary Table S27). When considering this set of closely related replicons, metagenomic assemblers also did not perform consistently better than genomic assemblers in the number of misassembled contigs or SNPs relative to the reference genome. Among the metagenomic assemblers, GATBMiniaPipeline and IDBA-UD performed particularly well but with values close to those of the 2 best genomic assemblers (SPAdes and Unicycler). IDBA-UB performed significantly worse with the *Micromonospora* replicons, possibly because of their lower coverage. As could have been expected, the number of SNPs in the lower-coverage *Halomonas* replicon was consistently higher than in the one with higher coverage (1.6-fold to 85.0-fold depending on the assembler), despite their relatively modest (12.6%) difference in estimated coverage in the sample and very significant depth of coverage (>500×) (Supplementary Tables S24 and S27). However, when comparing the number of SNPs in the *Marinobacter* replicons, this relationship is reversed, with the replicon with higher coverage (448×) having more SNPs relative to the reference than the one with lower coverage (135×) (Supplementary Table S27). This indicates that other factors, such as characteristics of the replicon, the actual representation in the sample, and the closeness to other replicons in the sample, may influence the performance of assemblers. It is also interesting to see that the contigs generated by themselves do not allow the distinction of closely related strains, since the number of SNPs relative to the reference genomes (if the assembler is able to cover >0.79 of the genome) is in the ranges of 315 to 9,915 and 9,773 to 70,196 for the higher and lower depth of coverage *Halomonas* replicons, respectively.

For the Gut-Mix-RR and Gut-HiLo-RR mock communities, the same pattern was observed as with the other mock communities, with the differences between the metagenomic and genomic assemblers being attenuated after filtering for contigs <1 kbp (Sup-

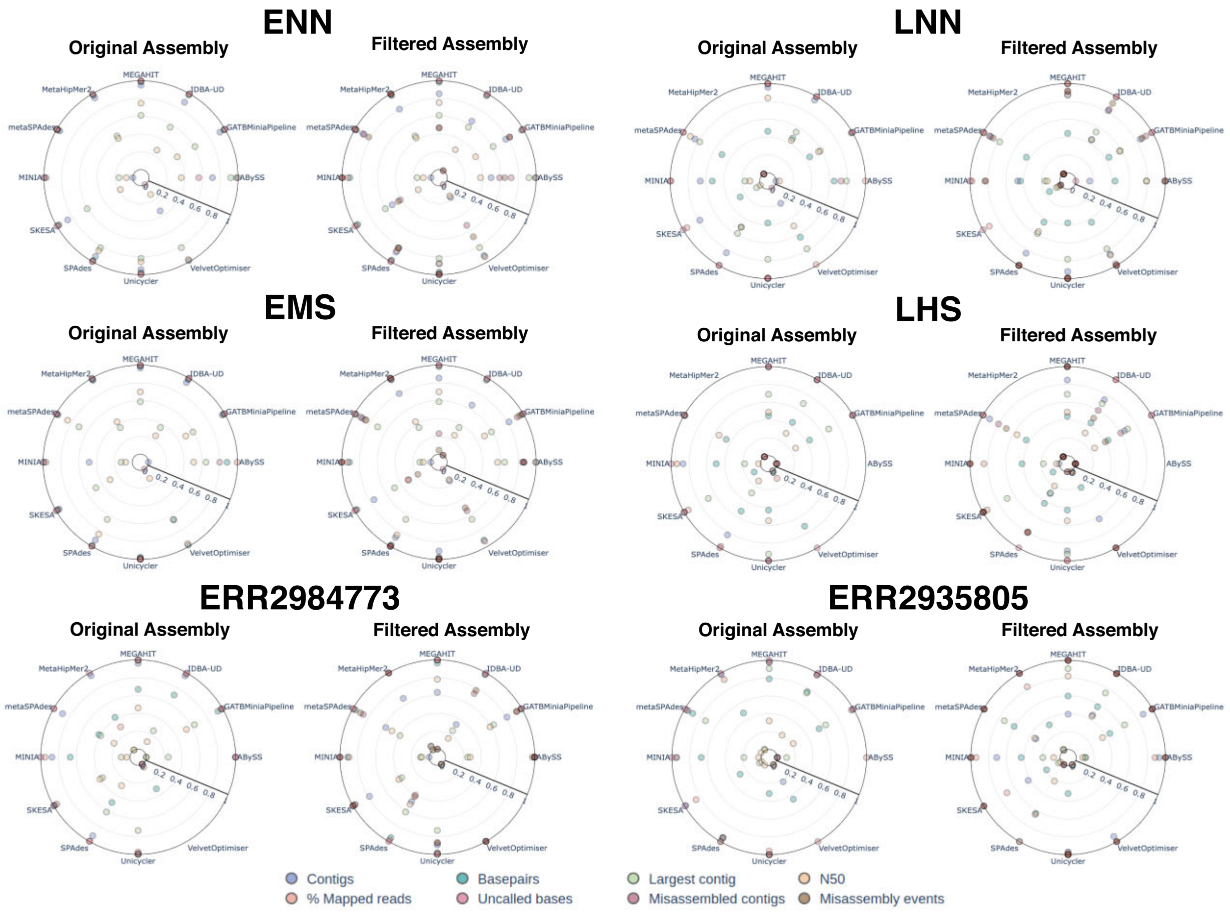


Figure 4: Assembler performance for the ZymoBIOMICS Microbial Community Standards dataset. For each sample in the dataset, the best score of each assembler in the 3 LMAS runs was selected. The results for each global assembly metric were normalized, with 1 representing the best result and 0 the worst. For the original assembly, the following metrics are presented: number of contigs produced (in blue), number of base pairs produced (in teal), the size of the largest contig assembled (in green), N50 (in yellow), percentage of mapped reads to the assembly (in orange), and uncalled bases (in red). For the filtered assembly, the additional metrics are presented: number of misassembled contigs (in purple) and number of misassembly events (in brown).

plementary Table S30 and S31). Particularly for the evenly distributed Gut-Mix-RR sample, when considering the subset of *Roseburia* spp., the replicon with the lowest coverage (12× for *R. intestinalis* versus 18× for *R. hominis*) had a consistently higher number of SNPs, with the exception of the SKESA assemblies, where the opposite was observed. This is similar to what was observed for the *Halomonas* replicons in the BMock12 community standard.

Success is not straightforward

Several factors contribute to suboptimal performance of the assembly process, including DNA isolation and library preparation protocol; sequencing technology, depth, and read length; and possible contamination and inherent characteristics of the sample composition.

Assembler performance is influenced by species

For the 8 bacterial genomes present in the ZymoBIOMICS Microbial Community Standards dataset samples, even in those with an even distribution of the genomes (ENN, EMS, and ERR2984773), variations in the assembly metrics were observed (Fig. 6, Supplementary Figures S4–S6, Supplementary Tables S14–S16). For all samples in the dataset, the genomes are recovered almost completely, with all replicons being >90% represented in the resulting assemblies. *Lactobacillus fermentum* is the least represented genome (92.2%–94.9%). Most replicon sequences are recovered in <100 contigs, except for *Pseudomonas aeruginosa*, *Escherichia coli*, and *Salmonella enterica*, and not considering IDBA-UB, which frequently produces a larger number of contigs when compared to other assemblers. However, in other mock samples, this worse performance of IDBA-UD in terms of number of contigs is not so clear. The absolute values of other metrics of assembly quality, such as LSA, misassembly events, or uncalled bases, are also different between bacterial genomes (Supplementary Tables S14–S16). The fact that *S. enterica* is a closely related species to *E. coli*, with high level of genetic similarity (ANIb >0.8, Supplementary Table S23), could have created difficulties for resolving the assemblies in a mixed sample and led to the lower coverage observed, the higher number of contigs, and the increased number of misassembled contigs identified in these species in some samples. However, in the case of the larger number of contigs of *P. aeruginosa*, no related species are present in the sample, and these possibly reflect intrinsic properties of the replicon such as a high number of prophages integrated in the bacterial genome [44]. Similarly, replicon characteristics could be behind the lower breadth of coverage consistently observed in *L. fermentum* assemblies.

In the BMock12 and the Gut-Mix samples, which have pairs of much more similar replicons, it is true that the closely related replicons do have higher numbers of contigs. However, it is pos-

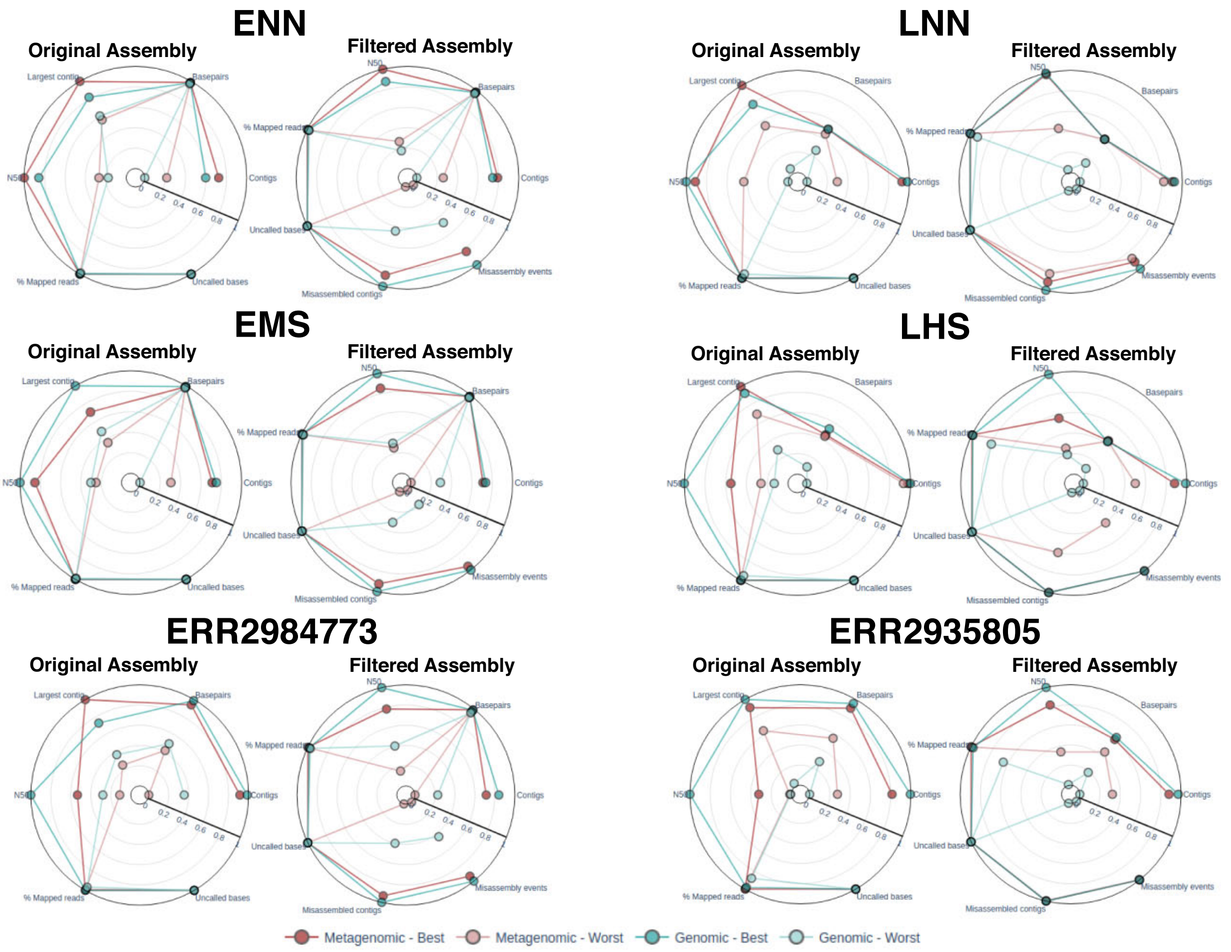


Figure 5: Performance of genomic and metagenomic assemblers for the ZymoBIOMICS Microbial Community Standards dataset. For each sample in the dataset and for the 3 runs, the best and worst scores for each assembler category were selected: genomic (in blue) and metagenomic (in red). The results for each global assembly metric were normalized, with 1 representing the best result and 0 the worst. For the original assembly, the following metrics are presented: number of contigs produced, number of base pairs produced, the size of the largest contig assembled, N50, percentage of mapped reads to the assembly, and uncalled bases. For the filtered assembly, the additional metrics are presented: number of misassembled contigs and number of misassembly events.

sible that the properties of the individual replicons also have an impact on the number of contigs generated by the assemblers (Supplementary Fig. S8). Another potential confounder that was not explored is the length of the reads, with Miseq samples having 300-bp reads and HiSeq samples having 150-bp reads.

Longer contigs have higher confidence

The Pls metric, which measures the error rate of a contig relative to the reference, shows that for every replicon, longer contigs have higher Pls (Fig. 7). This could justify the option of filtering an assembly by length, even beyond the 1,000-bp minimum contig size implemented by default in LMAS. Not only are we eliminating shorter, less informative contigs in terms of genetic context, but these are also the ones most likely to contain errors relative to the reference sequence.

Certain genomic regions are problematic for all assemblers

Some genomic regions in several replicons are consistently a challenge for all assemblers. As observed in Fig. 8, all genomes of the ZymoBIOMICS Microbial Community Standards dataset present certain regions that fail to assemble for all tools in all runs, even those generating high-quality draft assemblies. Of all 7 assemblers considered, only GATBMiniaPipeline, MEGAHIT, and IDBA-UD showed inconsistency in the gaps produced over the 3 LMAS runs (Supplementary Table S17), as expected from producing variable sets of contigs. The regions consistently missing for all assemblers in all runs are rich in repetitive elements, such as ribosomal RNA and transfer RNA coding sequences and mobile genetic elements (Supplementary Table S18), with larger gaps corresponding to tandem sets of these elements. This reflects an intrinsic limitation of short-read sequencing since the length of a read pair is not enough to bridge across the repetitive element, preventing the generation of contigs representing these regions. This is something that could be addressed by the use of long-read sequencing technologies. Despite this, some assemblers are able to produce contigs that represent some of these large tandem regions, such as MEGAHIT and SKESA for *E. faecalis* and IDBA-UD, MEGAHIT, and metaSPADES for *L. monocytogenes*, but such performance is not consistent for all reference replicons. For instance, SKESA fails to assemble 2 large regions of the *S. enterica* genome that all other assemblers successfully cover.

For the BMock12 community standard, the same pattern of consistency of difficult regions across replicons can be observed for all replicons, with the exception of the lowest abundance repli-

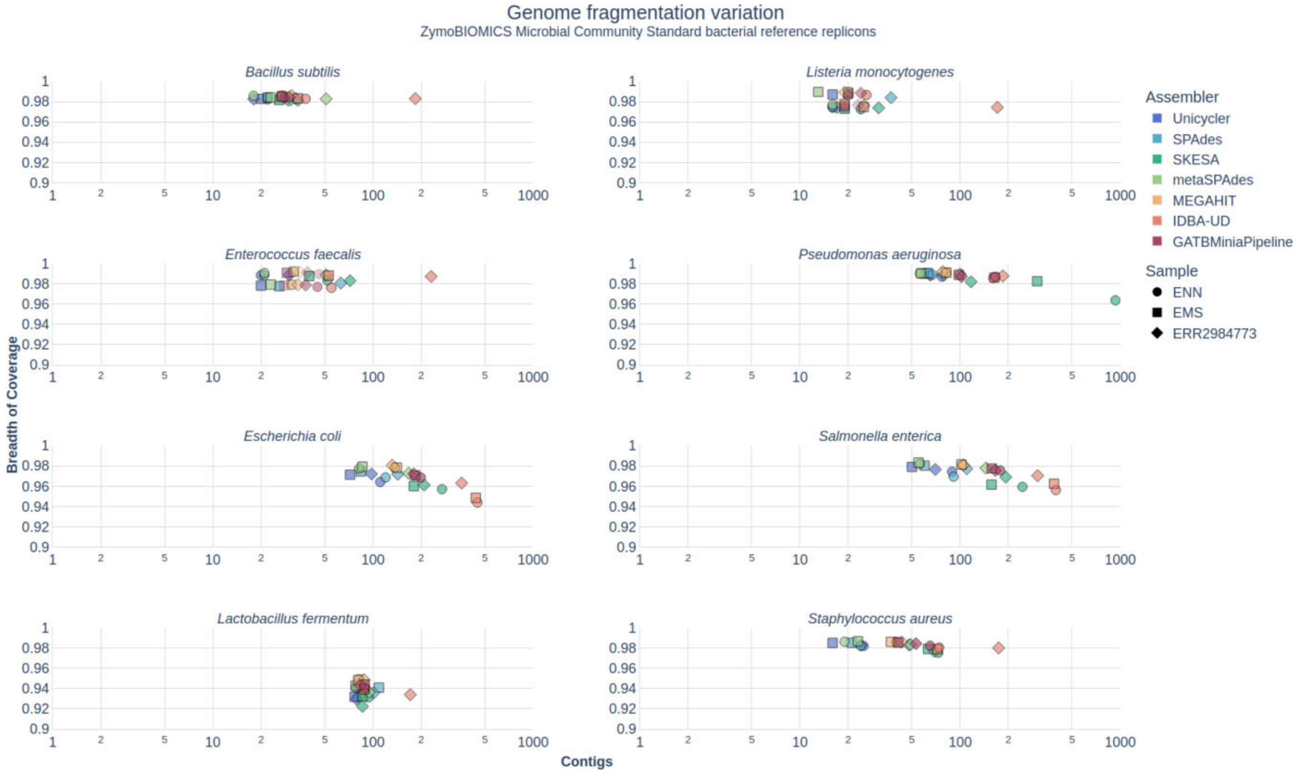


Figure 6: Genome fragmentation for each reference replicon of the ZimoBIOMICS community standards dataset for the evenly distributed samples. Genome fragmentation for the 3 LMAS runs is represented by the number of contigs and breadth of coverage of the reference per assembler for the evenly distributed samples: ENN (evenly distributed without error model, identified by a circle), EMS (evenly distributed with Illumina MiSeq error model, identified by a square), and ERR2984773 (real Illumina MiSeq sample, identified by a diamond). Each assembler is identified with the following color scheme—dark blue, Unicycler; light blue, SPAdes; dark green, SKESA; light green, metaSPAdes; yellow, MEGAHIT; orange, IDBA-UD; red, GATBMiniaPipeline.

cons (see Supplementary Fig. S9). Interestingly, the 2 closely related *Halomonas* replicons present a very dissimilar gap pattern, with a high number of gaps ($n = 2{,}789$ and $n = 2{,}702$) distributed throughout the replicon sequence, which possibly reflects the difficulty of assembling closely related replicons in the same sample.

Assembler performance is influenced by replicon abundance in the sample

The logarithmically distributed samples (LNN, LHS, and ERR2935805) of the ZymoBIOMICS community standard dataset showed greater variation in the assembly success metrics than the evenly distributed samples (Supplementary Tables S8–S10), reflecting the difficulty of recovering sequences of the lowest abundant replicons. For the 3 replicons with an estimated depth of coverage >15×, a similar pattern is observed in logarithmically distributed samples as in evenly distributed samples, albeit with greater dispersion in the number of contigs generated and with a markedly decreased breath of coverage for some assemblers and samples in the logarithmically distributed samples (Fig. 6 and Supplementary Fig. S7). Almost no contigs >1,000 bp were retrieved for replicons with an estimated depth of coverage of <2×, resulting in a very low breadth of coverage (<1%) (Supplementary Table S4, Supplementary Table S22). This leads to a severe underrepresentation of the diversity of the community in the generated contigs, particularly of plasmid sequences due to their smaller length and abundance as was described previously [3, 45, 46]. This happens despite the greater sequencing depth of these samples versus those with an even distribution (>5-fold difference in the number of reads).

For the BMock12 community standard, the very low replicon abundance (*Micromonospora coxensis*, 2,623,620,609, 0.02× coverage) fails to assemble in all tools (Supplementary Fig. S9, Supplementary Table S24). The *Micromonospora echinaurantiaca* (2,623,620,557, 14.9× coverage) and *Micromonospora echinofusca* (2,623,620,567, 18.2× coverage) fail to assemble with SKESA and Unicycler, and the *Propionibacteriaceae* replicon (2,615,840,646, 31.9× coverage) fails to be assembled with SKESA. In the Gut-Mix-RR standards, there are several replicons with <1× to 20× depth of coverage. Significant breath of coverage (>0.7) was obtained with the contigs created by most assemblers, including successful assemblies of some of the replicons by SKESA and Unicycler that had failed with higher-coverage replicons in the BMock12 standard. Taking together these data and that of the ZymoBIOMICS standard, these suggest that it is hard to establish a universal breakpoint at which each of the assemblers is able to generate high breadth of coverage contigs, with the actual genome of interest and the composition of the sample possibly playing a role. Nevertheless, coverages >15× result in high breadth of coverage contigs, albeit in the lower range with many contigs and a significant number of SNPs.

Conclusions

The purpose of LMAS is to empower users to test assembler performance in meaningful conditions for their experimental setup and objectives. Suitable mock communities, reproducing the users' samples of interest, can be used as a gold standard to evaluate assembler performance. To illustrate LMAS's functionalities,

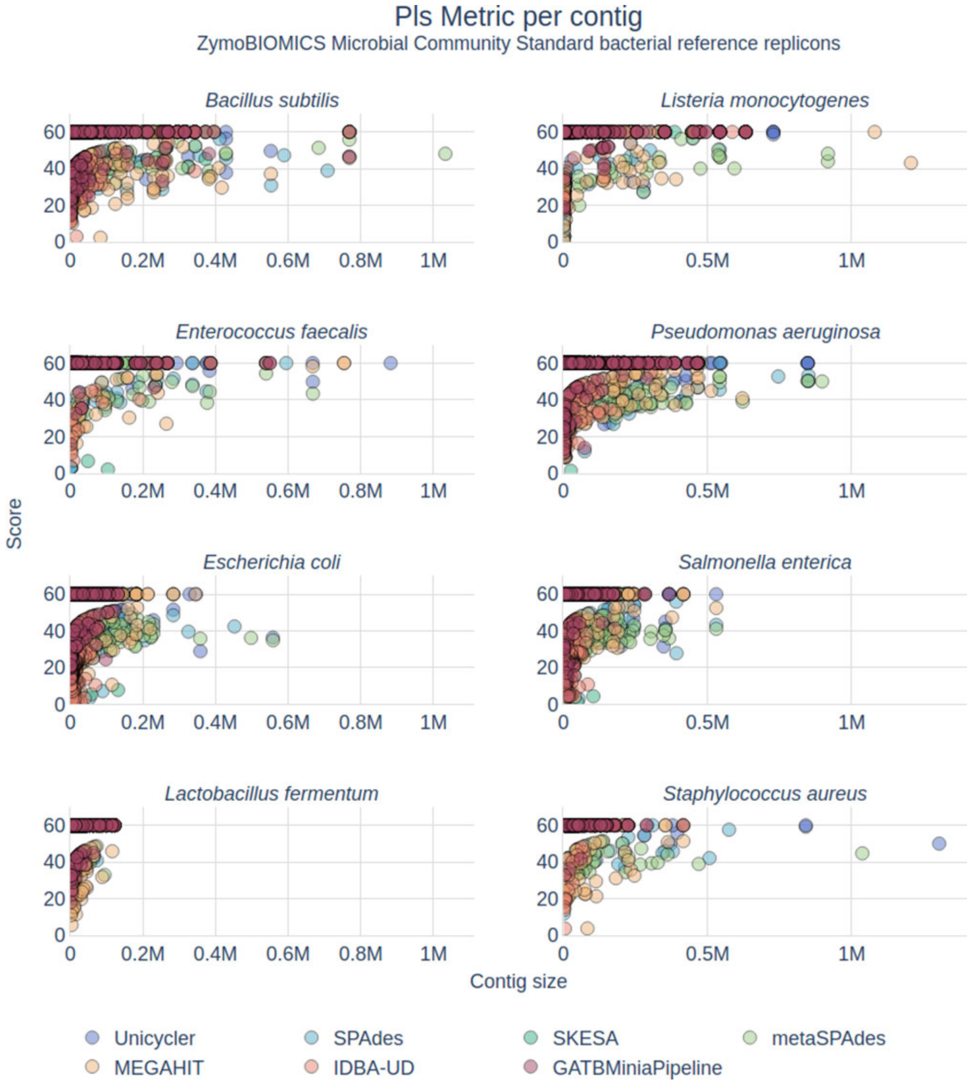


Figure 7: Phred-like score (Pls) per contig for each reference replicon of the ZimoBIOMICS community standards datasets. The Pls score was calculated for each unique contig produced by each assembler in 3 LMAS runs and is represented in relation to its contig size. Each contig is colored according to the assembler with the following color scheme—dark blue, Unicycler; light blue, SPAdes; dark green, SKESA; light green, metaSPAdes; yellow, MEGAHIT; orange, IDBA-UD; red, GATBMiniaPipeline.

we analyzed 3 well-known samples used in several studies. Although the 8-species ZymoBIOMICS Microbial Community Standards might not be representative of the metagenomic complexity of the samples of interest of most researchers, we hoped that its relative simplicity meant that the results shown would represent a best-case scenario, since as sample complexity increases, so do the challenges to assembler performance. However, the results of the BMock12 and Gut-Mix community standards suggest that the actual genome of interest and community composition play an important part in the results of individual assemblers.

Our results showed significant differences in both global and reference-dependent assembly quality metrics generated by each *de novo* assembler. The performance of each assembler varied depending on the species of interest and its abundance in the sample, with less abundant species presenting a significant challenge for all assemblers. The fact that an assembler is branded as specific for metagenomics does not guarantee a better performance in metagenomic samples, with assemblers used for genomic assembly outperforming the worst metagenomic assembler tested. Even when considering communities with very similar replicons, the overall performance of metagenomic assemblers was not consistently better than that of genomic assemblers. The results also indicate that the recovery of assemblies allowing strain-level discrimination at the SNP level is highly unlikely based solely on the assembler-generated contigs. The following assemblers showed significant performance problems, and their usability may be limited, at least with the default parameters we used: ABySS, MetaHipmer2, minia, and VelvetOptimiser.

The choice of *de novo* assembler depends greatly on the computational resources available, the species of interest, its representation in the sample, and, possibly, the composition of the community in the sample. In our testing with any of the mock communities, no assembler stood out as an undisputed all-purpose choice for short-read metagenomic prokaryotic genome assembly, with different assemblers showing specific strengths. Users would thus benefit from analyzing the results of sequencing mock communities (ideally) or of artificially generated reads simulating their samples of interest to guide their choice of assembler. LMAS was developed to be an easy-to-use and flexible tool for this purpose. From the results that we obtained with various mock com-

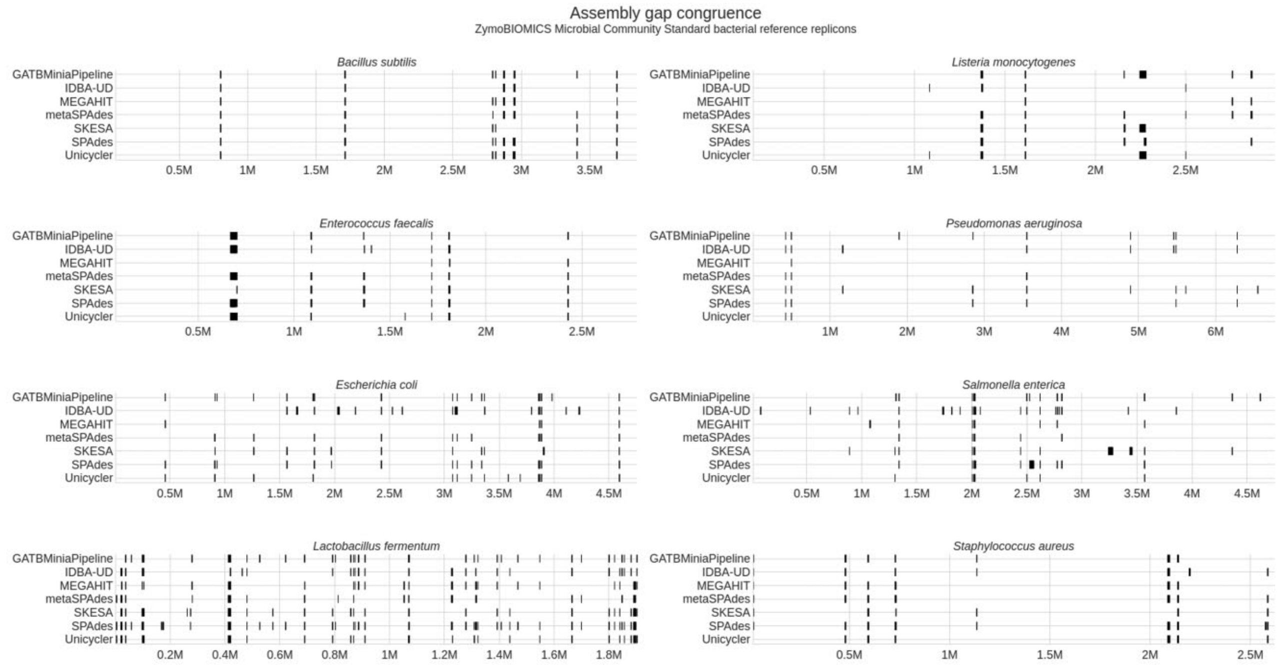


Figure 8: Location of gaps in comparison to the reference sequence, per assembler, for each reference replicon of the ZimoBIOMICS community standards datasets. The resulting plot contains the consistent gaps obtained from a 3-LMAS run for the evenly distributed dataset (ENN, EMS, and ERR2984773) for GATBMiniaPipeline, IDBA-UD, MEGAHIT, metaSPAdes, SKESA, SPAdes, and Unicycler assemblers.

munities, the following assemblers performed consistently well (presented in alphabetical order): MEGAHIT, metaSPAdes, SKESA, SPAdes, and Unicycler. From our assessment, we conclude that these assemblers are the most likely candidates to perform well in other complex samples.

LMAS was built with modularity and containerization as keystones, leveraging the parallelization of processes and guaranteeing reproducibility across platforms. The modular design allows for new assemblers to be easily added and existing assemblers to be easily updated, allowing LMAS to function as a continuous benchmarking platform and ensuring its future relevance as improvements in assembly software are proposed. LMAS will also support evaluating the gains of any cumulative improvements to existing assemblers using the same benchmark set adapted to a specific project or goal. Such reproducibility, capacity to easily add assemblers of interest not included in the current version, and flexibility for future extensions are important principles in computational method benchmarking. Moreover, in addition, by lowering the barriers to perform comparisons between assemblers, LMAS will encourage users to compare software performance against mock communities of special interest, depending on their operational focus.

The interactive report provides an intuitive platform for data exploration, allowing the user to easily sift through global and reference-specific performance metrics for each sample, as well as providing information on the assemblers executed to allow traceability of the results. Producing an extensive, metric-rich report allows users interested in different aspects of assembler performance to make informed decisions, particularly when choosing among the top-performing assemblers, which show only minor differences.

LMAS applies several well-known assembly metrics and proposes 2 more: LSA, which represents the fraction of the longest single alignment between a contig and the reference, and Pls, a scoring function based on the identity of each aligned contig to the reference replicon. The entire set of assembly quality metrics used in LMAS allows not only the assessment of quality based on statistics inherent to a set of assembled contigs but also a comparison to a ground truth provided using samples of known composition and reference sequences. The LMAS report provides an interactive and intuitive platform for the exploration of these results, allowing users to easily test assemblers in mock samples with species composition and distribution relevant for their own studies.

Although computationally intensive due to the complex nature of the *de novo* assembly process, LMAS is the only software-integrating assembly and its evaluation into a single pipeline, guaranteeing the same conditions are met for all tools. With LMAS, it is now possible to continuously evaluate which *de novo* assembler produces the most relevant results for a given community of interest. The LMAS workflow is open source, and its code and documentation are available at https://github.com/B-UMMI/LMAS and https://lmas.readthedocs.io/, respectively.

Availability of Supporting Source Code and Requirements

Project name: LMAS
Project home page: https://github.com/B-UMMI/LMAS
Operating system(s): UNIX-like systems.
Programming languages: Nextflow, Python, Bash, Javascript
Other requirements: Java version 8 or highest. Docker/Singularity/Shifter
License: GNU GPL v3
RRID: SCR_022251

Additional Files

Supplementary Fig. S1. LMAS misassembly classification. Misassembled contigs are classified into 6 main categories: chimera, insertion, deletion, inversion, rearrangement, translocation, and duplication, according to the mapping orientation, the distance between blocks in the contig, and the mapping coordinates in the reference replicon. If a contig is classified as being chimeric, no further classification is performed. The other categories are classified independently of each other, with combinations being possible, to better reflect the differences in comparison to the reference. If a contig is broken into multiple sequence blocks but fails to be classified in any of the previous categories, it is reported as being inconsistent.

Supplementary Fig. S2. Computational resources used by each assembler for the evenly and logarithmically distributed samples. Each plot describes the distribution of resource consumption for 3 LMAS runs for the ZymoBIOMICS Microbial Community Standards dataset for the following metrics: (A) CPU/hour, (B) maximum memory in GB, (C) data written to disk in GB, (D) data read from dist in GB, and (E) runtime in hours. The mean for all samples and all assemblers is indicated in red. The samples are indicated as follows: ENN: dark blue, EMS: teal, ERR2984773: green, LNN: light green, LHS: yellow, ERR2935805: light orange.

Supplementary Fig. S3. Performance per reference of genomic and metagenomic assemblers for the evenly distributed samples in the ZymoBIOMICS Microbial Community Standards dataset. For each sample in the dataset and for the 3 runs, the best and worst scores for each assembler category were selected: genomic (in blue) and metagenomic (in red). The results for each global assembly metric were normalized, with 1 representing the best result and 0 the worst.

Supplementary Fig. S4. Assembler performance per reference for the ZymoBIOMICS Microbial Community Standards dataset for sample ENN. The best score for each assembler was selected for 3 LMAS runs. The results for each global assembly metric were normalized, with 1 representing the best result and 0 the worst. The following assemblers are represented: GATBMiniaPipeline: dark blue, IDBA-UD: light blue, MEGAHIT: dark green, metaSPAdes: light green, SKESA: yellow, SPAdes: orange, Unicycler: red.

Supplementary Fig. S5. Assembler performance per reference for the ZymoBIOMICS Microbial Community Standards dataset for sample EMS. The best score for each assembler was selected for 3 LMAS runs. The results for each global assembly metric were normalized, with 1 representing the best result and 0 the worst. The following assemblers are represented: GATBMiniaPipeline: dark blue, IDBA-UD: light blue, MEGAHIT: dark green, metaSPAdes: light green, SKESA: yellow, SPAdes: orange, Unicycler: red.

Supplementary Fig. S6. Assembler performance per reference for the ZymoBIOMICS Microbial Community Standards dataset for sample ERR2984773. The best score for each assembler was selected for 3 LMAS runs. The results for each global assembly metric were normalized, with 1 representing the best result and 0 the worst. The following assemblers are represented: GATBMiniaPipeline: dark blue, IDBA-UD: light blue, MEGAHIT: dark green, metaSPAdes: light green, SKESA: yellow, SPAdes: orange, Unicycler: red.

Supplementary Fig. S7. Genome fragmentation for each reference replicon of the ZimoBIOMICS community standards dataset for the logarithmically distributed samples. Genome fragmentation for the 3 LMAS runs is represented by the number of contigs and breadth of coverage of the reference per assembler for the logarithmically distributed samples: LNN (logarithmically distributed without error model, identified by a circle), LHS (logarithmically distributed with Illumina HiSeq error model, identified by a square), and ERR2935805 (real Illumina HiSeq sample, identified by a diamond). Each assembler is identified with the following color scheme—dark blue: Unicycler, light blue: SPAdes, dark green: SKESA, light green: metaSPAdes, yellow: MEGAHIT, orange: IDBA-UD, red: GATBMiniaPipeline.

Supplementary Fig. S8. Genome fragmentation for each reference replicon of the BMock12 community standards dataset sample. Genome fragmentation is represented by the number of contigs and breadth of coverage of the reference per assembler. Each assembler is identified with the following color scheme—dark blue: Unicycler, light blue: SPAdes, dark green: SKESA, light green: metaSPAdes, yellow: MEGAHIT, orange: IDBA-UD, red: GATBMiniaPipeline. Each reference replicon is identified by its IMG Taxon ID: 2615840527, *Muricauda* sp.; 2615840533, *Thioclava* sp.; 2615840601, *Cohaesibacter* sp.; 2615840646, *Propionibacteriaceae* bacterium; 2615840697, *Marinobacter* sp. LV10R510-8; 2616644829, *Marinobacter.* sp LV10MA510-1; 2617270709, *Psychrobacter* sp.; 2623620557, *Micromonospora echinaurantiaca*; 2623620567, *Micromonospora echinofusca*; 2623620609, *Micromonospora coxensis*; 2623620617, *Halomonas* sp. HL-4; and 2623620618, *Halomonas* sp. HL-93.

Supplementary Fig. S9. Location of gaps in comparison to the reference sequence, per assembler, for each reference replicon of the BMock12 community standards datasets. The resulting plot contains the gaps obtained for GATBMiniaPipeline, IDBA-UD, MEGAHIT, metaSPAdes, SKESA, SPAdes, and Unicycler assemblers.

Supplementary Table S1. Tools available for the *de novo* assembly of prokaryotic genomes. For each tool, its publication is indicated, if available, as well as the assembly algorithm implemented if it was developed explicitly to handle metagenomic datasets. The tools are ordered by the date of the last update, with the source code indicated when available. The tools incorporated in LMAS are indicated as such.

Supplementary Table S2. Comparison of metrics and features of LMAS with QUAST and MetaQUAST.

Supplementary Table S3. The ZymoBIOMICS Microbial Community Standards datasets. Set of raw sequence reads used as input in LMAS of mock communities with an even and logarithmic distribution of species, from real sequencing runs and simulated read datasets, with and without error, matching the intended distribution of species in ZymoBIOMICS Microbial Community Standards.

Supplementary Table S4. Microbial composition of the ZymoBIOMICS Microbial Community Standards dataset with even and logarithmic distribution of species. Theoretical microbial composition of the standards and the corresponding number of reads generated for each replicon.

Supplementary Table S5. Global quality metrics variation in 3 LMAS runs for sample ENN per assembler. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, is presented for each metric for each assembler.

Supplementary Table S6. Global quality metrics variation in 3 LMAS runs for sample EMS per assembler. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, is presented for each metric for each assembler.

Supplementary Table S7. Global quality metrics variation in 3 LMAS runs for sample ERR2984773 per assembler. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, is presented for each metric for each assembler.

Supplementary Table S8. Global quality metrics variation in 3 LMAS runs for sample LNN per assembler. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, is presented for each metric for each assembler.

Supplementary Table S9. Global quality metrics variation in 3 LMAS runs for sample LHS per assembler. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, is presented for each metric for each assembler.

Supplementary Table S10. Global quality metrics variation in 3 LMAS runs for sample ERR2935805 per assembler. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, is presented for each metric for each assembler.

Supplementary Table S11. Inconsistent contigs produced by the assemblers in 3 LMAS runs. For each assembler, the total number of contigs produced over the 3 runs of the LMAS workflow is indicated, as well as the contigs present in only 2 runs and a single run.

Supplementary Table S12. Global assembly metrics for single and multiple *k*-mer dBg assemblers. The median and the minimum and maximum values obtained are presented for each metric for all samples in 3 runs of LMAS. Single *k*-mer bBg assemblers: ABySS and minia. Multiple *k*-mer bBg assembler: GATBMiniaPipeline, IDBA-UD, MEGAHIT, MetaHipMer2, metaSPAdes, SKESA, SPAdes, Unicycler, and VelverOptimiser.

Supplementary Table S13. Global assembly metrics for genomic and metagenomic multiple *k*-mer dBg assemblers. The median and the minimum and maximum values obtained are presented for each metric for all samples in 3 runs of LMAS. Genomic assemblers: SKESA, SPAdes, and Unicycler. Metagenomic assemblers: GATBMiniaPipeline, IDBA-UD, MEGAHIT, and metaSPAdes.

Supplementary Table S14. Per reference quality metrics variation in 3 LMAS runs for sample ENN per assembler of the ZymoBIOMICS Microbial Community Standards dataset. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, is presented for each metric for each assembler.

Supplementary Table S15. Per reference quality metrics variation in 3 LMAS runs for sample EMS per assembler of the ZymoBIOMICS Microbial Community Standards dataset. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, is presented for each metric for each assembler.

Supplementary Table S16. Per reference quality metrics variation in 3 LMAS runs for sample ERR2984773 per assembler of the ZymoBIOMICS Microbial Community Standards dataset. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, is presented for each metric for each assembler.

Supplementary Table S17. Inconsistent gaps produced by the assemblers in 3 LMAS runs. For each assembler, the total number of gaps consistently produced in relation to the reference replicons over the 3 runs of the LMAS workflow is indicated, as well as gaps present in only 2 runs and a single run.

Supplementary Table S18. Annotation of consistent gaps produced by the assemblers in 3 LMAS runs.

Supplementary Table S19. Number of transfer RNA and ribosomal RNA coding sequencing, as well as mobile elements in ZymoBIOMICS Microbial Community Standards reference replicons. The average calculated for all samples in the dataset for the 3 independent LMAS runs, followed by the minimum and maximum values obtained, is presented for each metric for each assembler.

Supplementary Table S20. Taxonomic classification of the ZymoBIOMICS Microbial Community Standards dataset. The classification was performed with Kraken2, using the Standard Database. The results are presented as the percentage of classified reads for the 8 bacterial species in the community, as well as unclassified reads and the group of reads that are classified as species not contained in the community standard.

Supplementary Table S21. Global assembly metrics for dBg assemblers with single and multiple *k*-mer algorithms.

Supplementary Table S22. Reference-based quality metrics in 3 LMAS runs for the ZimoBIOMICS community standards dataset.

Supplementary Table S23. Pairwise comparisons of the ZymoBIOMICS Microbial Community Standards reference replicons. All pairwise comparisons among the set of genomes were conducted using Average Nucleotide Identity through BLAST as a proxy for DNA–DNA hybridization.

Supplementary Table S24. Microbial composition of the BMock12 microbial community standard dataset.

Supplementary Table S25. Pairwise comparisons of the BMock12 microbial community standard reference replicons.

Supplementary Table S26. Global quality metrics for the BMock12 sample SRX4901583 per assembler.

Supplementary Table S27. Per reference quality metrics for the BMock12 sample SRX4901583 per assembler.

Supplementary Table S28. Microbial composition of the Gut-Mix-RR microbial community standard dataset.

Abbreviations

bp: base pair; cgMLST: core-genome multilocus sequence typing; dBg: de Bruijn graphs; FS: filtered set; GB: gigabytes; HPC: high-performance computing cluster; LMAS: Last Metagenomic Assembler Standing; LSA: longest single alignment; OLC: overlap-layout-consensus; Pls: Phred-like score; SNP: single-nucleotide polymorphism.

Data Availability

The datasets analyzed during the current study are available in the Zenodo repository, under [48]. All supplemental material is available in the Zenodo repository, under [49]. Likewise, all figures in the current manuscript are available in their original format in the Zenodo repository, under [50]. Real sequencing data of the ZymoBIOMICS Microbial Community Standards are available under accessions ERR2984773 and ERR2935805 [38]. All data generated or analyzed during this study are included in this published article, its supplementary information files, and the data analysis repository located at [47]. Additionally, the reports for the ZymoBIOMICS Microbial Community Standards, BMock12 Community Standard, and NIBSC Gut DNA Reference are available at [51], [52], and [53], respectively.

An archival copy of the GitHub repository (https://github.com/B-UMMI/LMAS) is also available via the *GigaScience* database, GigaDB [54].

Competing Interests

M.R. received honoraria for serving on the speakers' bureau of Pfizer and Merck Sharp and Dohme and for participating in expert panels of GlaxoSmithKline and Merck Sharp and Dohme. The other authors declare that they have no competing interests.

Funding

Catarina Inês Mendes was supported by the Fundação para a Ciência e Tecnologia (grant SFRH/BD/129483/2017).

Authors' Contributions

C.I.M. and M.R. designed the workflow. C.I.M. implemented and optimized the workflow, created the Docker containers, generated mock shotgun metagenomics data used to test and validate the workflow, contributed to the development of the HTML report, and analyzed the data. C.I.M. and M.R. wrote the manuscript. P.V.C. contributed to the development of the HTML report. M.R., J.A.C., Y.M., and J.M.G critically revised the manuscript. All authors read, commented on, and approved the final manuscript.

Acknowledgments

The authors thank Rafael Mamede for his contribution to the implementation and commentary on the several interactive plots implemented throughout the LMAS report. The authors also thank Nabil Fareed-Alikan for his insightful commentary on the interpretation of the results reported in this manuscript and Anthony Underwood and Robert A. Petit III for their assistance in building the LMAS Nextflow workflow. The authors also thank Samuel Nicholls, Joshua Quick, Shuiquan Tang, and Nicholas Loman for publicly providing the sequencing data for the ZymoBIOMICS Microbial Community Standards. Likewise, the authors thank Volkan Sevim, Juna Lee, Robert Egan, Alicia Clum, Hope Hundley, Janey Lee, R. Craig Everroad, Angela M. Detweiler, Brad M. Bebout, Jennifer Pett-Ridge, Markus Göker, Alison E. Murray, Stephen R. Lindemann, Hans-Peter Klenk, Ronan O'Malley, Matthew Zane, Jan-Fang Cheng, Alex Copeland, Christopher Daum, Esther Singer, and Tanja Woyke for providing the sequencing data for the BMock12 Commynity Standard, and to Gregory C. A. Amos, Alastair Logan, Saba Anwar, Martin Fritzsche, Ryan Mate, Thomas Bleazard, and Sjoerd Rijpkema for providing the sequencing data of the NIBSC Gut DNA Reference Gut-Mix-RR and Gut-Mix-HiLo Community Standard.

References

1. Angers-Loustau, A, Petrillo, M, Bengtsson-Palme, J, *et al.* The challenges of designing a benchmark strategy for bioinformatics pipelines in the identification of antimicrobial resistance determinants using next generation sequencing technologies. *F1000Research* 2018;**7**:459.
2. Gruening, B, Sallou, O, Moreno, P, *et al.* Recommendations for the packaging and containerizing of bioinformatics software. *F1000Research* 2019;**7**:742.
3. Sczyrba, A, Hofmann, P, Belmann, P, *et al.* Critical assessment of metagenome interpretation—a benchmark of metagenomics software. *Nat Methods* 2017;**14**(11):1063–71.
4. Couto, N, Schuele, L, Raangs, EC, *et al.* Critical steps in clinical shotgun metagenomics for the concomitant detection and typing of microbial pathogens. *Sci Rep* 2018;**8**(1): 13767.
5. Meyer, F, Fritz, A, Deng, Z-L, *et al.* Critical assessment of metagenome interpretation—the second round of challenges. *Bioinformatics* 2021;**19**:429–40.
6. Ayling, M, Clark, MD, Leggett, RM. New approaches for metagenome assembly with short reads. *Briefings Bioinf* 2020;**21**(2):584–94.
7. Olson, ND, Treangen, TJ, Hill, CM, *et al.* Metagenomic assembly through the lens of validation: recent advances in assessing and improving the quality of genomes assembled from metagenomes. *Briefings Bioinf* 2019;**20**(4):1140–50.
8. Teeling, H, Glockner, FO. Current opportunities and challenges in microbial metagenome analysis—a bioinformatic perspective. *Briefings Bioinf* 2012;**13**(6):728–42.
9. Bradnam, KR, Fass, JN, Alexandrov, A *et al.* Assemblathon 2: evaluating de novo methods of genome assembly in three vertebrate species. *GigaScience* 2013;**2**(1). https://doi.org/10.1186/2047-217X-2-10
10. Gurevich, A, Saveliev, V, Vyahhi, N, *et al.* QUAST: quality assessment tool for genome assemblies. *Bioinformatics* 2013;**29**(8):1072–5.
11. Mikheenko, A, Saveliev, V, Gurevich, A. MetaQUAST: evaluation of metagenome assemblies. *Bioinformatics* 2016;**32**(7):1088–90.
12. Di Tommaso, P, Chatzou, M, Floden, EW, *et al.* Nextflow enables reproducible computational workflows. *Nat Biotechnol* 2017;**35**(4):316–9.
13. Merkel, D. Docker: lightweight Linux containers for consistent development and deployment. *Linux J* 2014;**2014** (239)
14. Lmas: Anaconda.org. https://anaconda.org/bioconda/lmas. Accessed 2022 Apr 6.
15. Mendes, I, Vila-Cerqueira, P, Ramirez, M. LMAS: Last (Meta)genomic Assembler Standing. 2021. https://github.com/B-UMMI/LMAS. Accessed 2022 Apr 4.
16. LMAS's—LMAS 0.1 documentation. https://lmas.readthedocs.io/en/latest/index.html. Accessed 2022 Apr 4.
17. Basic Usage—LMAS 0.1 documentation. https://lmas.readthedocs.io/en/latest/user/basic_usage.html. Accessed 2022 Apr 4.
18. Parameters—LMAS 0.1 documentation. https://lmas.readthedocs.io/en/latest/user/parameters.html. Accessed 2022 Apr 4.
19. Jackman, SD, Vandervalk, BP, Mohamadi, H, *et al.* ABySS 2.0: resource-efficient assembly of large genomes using a Bloom filter. *Genome Res* 2017;**27**(5):768–77.
20. GATB/gatb-minia-pipeline. 2022. https://github.com/GATB/gatb-minia-pipeline. Accessed 2022 Apr 4.
21. Peng, Y, Leung, HCM, Yiu, SM, *et al.* IDBA-UD: a de novo assembler for single-cell and metagenomic sequencing data with highly uneven depth. *Bioinformatics* 2012;**28**(11):1420–8.
22. Li, D, Liu, C-M, Luo, R, *et al.* MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph. *Bioinformatics* 2015;**31** (10):1674–1676.
23. Georganas, E, Egan, R, Hofmeyr, S, *et al.* Extreme scale de novo metagenome assembly. In: *SC18: International Conference for High Performance Computing, Networking, Storage and Analysis*. Dallas, TX: IEEE.
24. Nurk, S, Meleshko, D, Korobeynikov, A, *et al.* metaSPAdes: a new versatile metagenomic assembler. *Genome Res* 2017;**27**(5):824–34.
25. Chikhi, R, Rizk, G. Space-efficient and exact de Bruijn graph representation based on a Bloom filter. *Algorithms Mol Biol* 2013;**8**(1):22.
26. Souvorov, A, Agarwala, R, Lipman, DJ. SKESA: strategic k-mer extension for scrupulous assemblies. *Genome Biol* 2018;**19**(1):153.
27. Bankevich, A, Nurk, S, Antipov, D, *et al.* SPAdes: a new genome assembly algorithm and its applications to single-cell sequencing. *J Comput Biol* 2012;**19**(5):455–77.
28. Wick, RR, Judd, LM, Gorrie, CL, *et al.* Unicycler: resolving bacterial genome assemblies from short and long sequencing reads. *PLoS Comput Biol* 2017;**13**(6):e1005595.
29. Seemann, T. VelvetOptimiser: automate your Velvet assemblies. 2021. https://github.com/tseemann/VelvetOptimiser. Accessed 2022 Apr 4.

30. Short-Read (Meta)Genomic Assemblers—LMAS 0.1 documentation. https://lmas.readthedocs.io/en/latest/user/assemblers.html. Accessed 2022 Apr 4.
31. Add Assembler Process—LMAS 0.1 documentation. https://lmas.readthedocs.io/en/latest/dev/add_process.html. Accessed 2022 Apr 4.
32. Li, H. Minimap2: pairwise alignment for nucleotide sequences. *Bioinformatics* 2018;**34**(18):3094–100.
33. Ewing, B, Green, P. Base-calling of automated sequencer traces using Phred. II. Error probabilities. *Genome Res* 1998;**8** (3):186–94.
34. Manchanda, N, Portwood, JL, Woodhouse, MR, *et al.* GenomeQC: a quality assessment tool for genome assemblies and gene structure annotations. *BMC Genomics* 2020;**21**(1):193.
35. Meader, S, Hillier, LW, Locke, D, *et al.* Genome assembly quality: assessment and improvement using the neutral indel model. *Genome Res* 2010;**20**(5):675–84.
36. Challis, R, Richards, E, Rajan, J, *et al.* BlobToolKit—interactive quality assessment of genome assemblies. *G3 (Bethesda)* 2020;**10**(4):1361–74.
37. Meyer, F, Lesker, T-R, Koslicki, D, *et al.* Tutorial: assessing metagenomics software with the CAMI benchmarking toolkit. *Nat Protoc* 2021;**16**(4):1785–801.
38. Nicholls, SM, Quick, JC, Tang, S, *et al.* Ultra-deep, long-read Nanopore sequencing of mock microbial community standards. *GigaScience* 2019;**8**(5):giz043.
39. Gourlé, H, Karlsson-Lindsjö, O, Hayer, J, *et al.* Simulating Illumina metagenomic data with InSilicoSeq. *Bioinformatics* 2019;**35**(3):521–2.
40. Sevim, V, Lee, J, Egan, R, *et al.* Shotgun metagenome data of a defined mock community using Oxford Nanopore, PacBio and Illumina technologies. *Scientific Data* 2019;**6**(1):285.
41. Amos, GCA, Logan, A, Anwar, S, *et al.* Developing standards for the microbiome field. *Microbiome* 2020;**8**(1):98.
42. Xavier, BB, Sabirova, J, Pieter, M, *et al.* Employing whole genome mapping for optimal de novo assembly of bacterial genomes. *BMC Res Notes* 2014;**7**(1):484.
43. Mahadik, K, Wright, C, Kulkarni, M, *et al.* Scalable genome assembly through parallel de Bruijn graph construction for multiple k-mers. *Sci Rep* 2019;**9**(1):14882.
44. Johnson, G, Stark, N, Wolfe, AJ, *et al.* Complete genome sequence of a Pseudomonas aeruginosa isolate from a kidney stone. *Microbiol Resour Announc* 2019;**8**(38):e01073–19.
45. Meyer, F, Fritz, A, Deng, Z-L, *et al.* Critical assessment of metagenome interpretation: the second round of challenges. *Nat Methods* 2022;**19**(4):429–40.
46. Fritz, A, Hofmann, P, Majda, S, *et al.* CAMISIM: simulating metagenomes and microbial communities. *Microbiome* 2019;**7**(1):17.
47. LMAS Manuscript Analysis. 2022. https://github.com/B-UMMI/LMAS_Manuscript_Analysis. Accessed 2022 Apr 4.
48. Mendes, CI. 2021 LMAS Test Dataset *Zenodo Repository* [Accessed April 4 2022] https://doi.org/10.5281/zenodo.4588969
49. Mendes, CI. 2022 LMAS Supplemental Material [Data set] *Zenodo* https://doi.org/10.5281/zenodo.7129554
50. Mendes, CI. 2022 LMAS Figures [Data set] *Zenodo* https://doi.org/10.5281/zenodo.7007317
51. LMAS Test Dataset. ZymoBIOMICS Microbial Community Standards. *Zenodo repository*. https://doi.org/10.5281/zenodo.7088960
52. LMAS Test Dataset. BMock12 Community Standards. *Zenodo repository*. https://doi.org/10.5281/zenodo.7092431
53. LMAS Test Dataset. NIBSC Gut DNA Reference. *Zenodo repository*. https://doi.org/10.5281/zenodo.7092693
54. Mendes, CI, Vila-Cerqueira, P, Motro, Y, *et al.* Supporting data for “LMAS: Evaluating metagenomic short de novo assembly methods through defined communities”. *GigaScience Database* 2022. http://dx.doi.org/10.5524/102335

GigaScience, 2022, **11**, 1–13
DOI: 10.1093/gigascience/giac003
TECH NOTE

Future-proofing and maximizing the utility of metadata: The PHA4GE SARS-CoV-2 contextual data specification package

Emma J. Griffiths[1,*], Ruth E. Timme[2], Catarina Inês Mendes[3], Andrew J. Page[4], Nabil-Fareed Alikhan[4], Dan Fornika[5], Finlay Maguire[6], Josefina Campos[7], Daniel Park[8], Idowu B. Olawoye[9,10], Paul E. Oluniyi[9,10], Dominique Anderson[11], Alan Christoffels[11], Anders Gonçalves da Silva[12], Rhiannon Cameron[1], Damion Dooley[1], Lee S. Katz[13,29], Allison Black[14], Ilene Karsch-Mizrachi[15], Tanya Barrett[15], Anjanette Johnston[15], Thomas R. Connor[16,17], Samuel M. Nicholls[18], Adam A. Witney[19], Gregory H. Tyson[20], Simon H. Tausch[21], Amogelang R. Raphenya[22], Brian Alcock[22], David M. Aanensen[23,24], Emma Hodcroft[25,26], William W. L. Hsiao[1,5,27], Ana Tereza R. Vasconcelos[28], Duncan R. MacCannell[29] and on behalf of the Public Health Alliance for Genomic Epidemiology (PHA4GE) consortium

[1]Faculty of Health Sciences, Simon Fraser University, Burnaby V5A 1S6, BC, Canada
[2]Center for Food Safety and Applied Nutrition, U.S. Food and Drug Administration, College Park, MD 20740, USA
[3]Instituto de Microbiologia, Instituto de Medicina Molecular, Faculdade de Medicina, Universidade de Lisboa, Lisboa 1649-028, Portugal
[4]Microbes in the Food Chain, Quadram Institute Bioscience, Norwich, Norfolk NR4 7UQ, UK
[5]BC Centre for Disease Control Public Health Laboratory, Vancouver, BC V5Z 4R4, Canada
[6]Faculty of Computer Science, Dalhousie University, Halifax, NS B3H 1W5, Canada
[7]INEI-ANLIS “Dr Carlos G. Malbrán,” Buenos Aires C1282AFF, Argentina
[8]Infectious Disease and Microbiome Program, The Broad Institute of MIT and Harvard, Cambridge, MA 02142, USA
[9]African Center of Excellence for Genomics of Infectious Diseases (ACEGID), Redeemer’s University, Ede, Osun State 232103, Nigeria
[10]Department of Biological Sciences, College of Natural Sciences, Redeemer’s University, Ede, Osun State 232103, Nigeria
[11]South African Medical Research Council Bioinformatics Unit, South African National Bioinformatics Institute, University of the Western Cape, Bellville 7530, South Africa
[12]Microbiological Diagnostic Unit Public Health Laboratory, The Peter Doherty Institute for Infection and Immunity, The University of Melbourne, Melbourne, VIC 3000, Australia
[13]Center for Food Safety, University of Georgia, Atlanta, GA 30333, USA
[14]Department of Epidemiology, University of Washington, WA 98109, USA
[15]National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health, Bethesda, MD 20894, USA
[16]Organisms and Environment Division, School of Biosciences, Cardiff University, Cardiff CF10 3AX, UK
[17]Public Health Wales, University Hospital of Wales, Cardiff CF14 4XW, UK
[18]University of Birmingham, Birmingham B17 2TT, UK
[19]Institute for Infection and Immunity, St George’s, University of London, London SW17 0RE, UK
[20]Center for Veterinary Medicine, U.S. Food and Drug Administration, Laurel, MD 20708, USA
[21]Department of Biological Safety, German Federal Institute for Risk Assessment, Berlin 12277, Germany
[22]Department of Biochemistry and Biomedical Sciences and the Michael G. DeGroote Institute for Infectious Disease Research, McMaster University, Hamilton, ON L8S 4L8, Canada
[23]Centre for Genomic Pathogen Surveillance, Wellcome Genome Campus, Cambridge CB10 1SA, UK
[24]The Big Data Institute, Li Ka Shing Centre for Health Information and Discovery, Nuffield Department of Medicine, University of Oxford, Oxford OX3 7LF, UK
[25]Biozentrum, University of Basel, Basel 3012, Switzerland
[26]Swiss Institute of Bioinformatics, Lausanne, Switzerland
[27]Department of Pathology and Laboratory Medicine, University of British Columbia, Vancouver, BC V6T 1Z7 V6T 1Z7, Canada
[28]Bioinformatics Laboratory National Laboratory of Scientific Computation LNCC/MCTI, Petrópolis 25651-075, Brazil
[29]Office of Advanced Molecular Detection, National Center for Emerging and Zoonotic Infectious Diseases, Centers for Disease Control and Prevention, GA 30333, USA
*Correspondence address. E-mail: emma_griffiths@sfu.ca

Abstract

Background: The Public Health Alliance for Genomic Epidemiology (PHA4GE) (https://pha4ge.org) is a global coalition that is actively working to establish consensus standards, document and share best practices, improve the availability of critical bioinformatics tools and resources, and advocate for greater openness, interoperability, accessibility, and reproducibility in public health microbial bioinformatics. In the face of the current pandemic, PHA4GE has identified a need for a fit-for-purpose, open-source SARS-CoV-2 contextual data standard.

Results: As such, we have developed a SARS-CoV-2 contextual data specification package based on harmonizable, publicly available community standards. The specification can be implemented via a collection template, as well as an array of protocols and tools to support both the harmonization and submission of sequence data and contextual information to public biorepositories.

Received: August 12, 2021. **Revised:** December 15, 2021. **Accepted:** January 7, 2022

Conclusions: Well-structured, rich contextual data add value, promote reuse, and enable aggregation and integration of disparate datasets. Adoption of the proposed standard and practices will better enable interoperability between datasets and systems, improve the consistency and utility of generated data, and ultimately facilitate novel insights and discoveries in SARS-CoV-2 and COVID-19. The package is now supported by the NCBI's BioSample database.

Keywords: genomics, metadata, SARS-CoV-2, bioinformatics, data standards

Findings

The importance of contextual data for interpreting SARS-CoV-2 sequences

First identified in late 2019 in Wuhan, China, the SARS-CoV-2 virus has now spread to virtually every country and territory in the world, resulting in millions of confirmed cases, and deaths, globally [1, 2]. Understanding, monitoring, and preventing transmission, as well as the development of vaccines and effective therapeutic options, have been primary goals of the public health response to SARS-CoV-2.

Tracking the spread and evolution of the virus at global, national, and local scales has been aided by the analysis of viral genome sequence data alongside SARS-CoV-2 epidemiology. Large-scale sequencing efforts are often formalized as consortia across the world, including the COG-UK in the UK [3], SPHERES in the USA [4], CanCOGeN in Canada [5], the Latin American Genomics SARS-CoV-2 Network [6, 7], 2019nCoVR in China [8], the South Africa NGS Genomic Surveillance Network [9], AusTrakka in Australia and New Zealand [10], and INSACOG in India [11]. In addition to these initiatives, many agencies, universities, and hospital laboratories around the world are also sequencing and sharing sequence data at an unprecedented pace. Deposition of these sequences into public repositories such as the Global Initiative on Sharing All Influenza Data (GISAID) and the International Nucleotide Sequence Database Collaboration (INSDC) has enabled rapid global sharing of data [12, 13]. At the time of writing, 174 countries had undertaken open sequencing initiatives (GISAID accessed 2021–06-23) depositing 2,057,675 sequences, which are being reused and analysed on a massive scale. The open data sharing paradigm has had tremendous success in the genomic epidemiology of foodborne pathogens [14, 15] and has the potential to reveal a deeper understanding of SARS-CoV-2 origin, pathogenicity, and basic biological characteristics when submissions from environmental samples and wild hosts are included alongside human clinical samples [16].

SARS-CoV-2 sequencing, analysis, and open sharing have played a crucial role in a number of developments during the pandemic, such as dispelling misinformation about the origins of the virus [17], the identification and surveillance of variants of concern [18, 19], the improvement of diagnostic performance and rapid testing [20–22], and the development of vaccines, which are currently being distributed in the largest global vaccination program the world has ever seen [23]. Viral genomic sequences are also being used to understand transmission and reinfection events [24], as well to monitor the prevalence and diversity of lineages during different exposure events and in different settings, e.g., animal reservoirs [25], long-term care facilities [26–28], healthcare and other work sites [29–33], and conferences and other public gatherings [34], as well as before and after public health responses (e.g., border controls and travel restrictions, lockdowns and quarantines, vaccination), through successive waves of infections [35–46]. However, it is critical to note that public health sequence data are of limited value without accompanying contextual metadata.

Contextual data consist of sample metadata (e.g., collection date, sample type, geographical location of sample collection), as well as laboratory (e.g., date and location testing, cycle threshold [CT] values), clinical outcomes (e.g., hospitalization, death, recovery), epidemiological (e.g., age, sex, exposures, vaccination status), and methods (e.g., sampling, sequencing, bioinformatics) data that enable the interpretation of sequence data. High-quality contextual data are also crucial for quality control. For example, detecting systematic batch effect errors related to certain sequencing centres and methods can help evaluate which variants represent real, circulating viruses, as opposed to artefacts of sample handling or sequencing that may arise owing to different aspects of experimental design, laboratory procedures, bioinformatics processing, and applied quality control thresholds [47–49].

Good data stewardship practices are critical not only for auditability and reproducibility but for posterity—documenting critical information about samples, methods, risk factors and outcomes, and so forth can help future-proof information used to build a roadmap for dealing with future public health crises. Contextual data, however, are often collected on a project-specific basis according to local needs and reporting requirements, which results in the collection of different data types at different levels of granularity, with different meanings and implicit bias of variables and attributes. Furthermore, the information is often collected as free text or, if structured, according to organization or initiative-specific data dictionaries, using different fields, terms, formats, abbreviations, and jargon.

The variability in the way information is encoded in private databases tends to propagate to public repositories, which makes the information more difficult to interpret and to use. There are different existing standards that can be used to structure contextual data, like minimum information checklists (MIxS [50], MIGS [51], the NIAID/BRC Project, and Sample Application Standard [52]) and various interoperable ontologies (OBO Foundry [53]), which make information easier to aggregate and reuse for different types of analyses. However, these attribute packages and metadata standards developed by different organizations are usually scoped to cover as many use cases and pathogens as possible and, as such, can include fields of information not applicable to SARS-CoV-2, or that may be subject to privacy concerns, or exclude fields commonly used in public health surveillance and investigations. Because different types of contextual data are subject to different ethical, practical, and privacy concerns, not all components of existing standards are immediately or widely collectable and shareable. As a result, the range of generic metadata standards being applied to SARS-CoV-2 data presents challenges for data harmonization [54] and analysis critical for fighting the disease and ending the pandemic.

In light of these challenges, PHA4GE has identified a need for a fit-for-purpose, open-source SARS-CoV-2 contextual data specification that can be used to consistently structure information as part of good data management practices and for data sharing with trusted partners and/or public repositories. The specification was developed by consensus among domain experts, and incorporates existing community standards with an emphasis on SARS-CoV-2 public health needs and ensuring privacy while maximizing information content and interoperability across datasets and databases to better enable analyses to fight COVID-19. The specification package also contains a number of accompanying materials such as standard operating procedures, tools, a reference guide, and repository submission protocols (protocols.io) to help put the standard into practice.

SARS-CoV-2 Contextual Data Specification: The Framework

The purpose of the PHA4GE SARS-CoV-2 specification is to provide a mechanism for consistent structure, collection, and formatting of fields and values containing SARS-CoV-2 contextual data pertaining to clinical, animal, and environmental samples. We emphasize that the purpose of this specification is not to force data sharing but rather to provide a framework to structure data consistently across disparate laboratory and epidemiological databases so that they can be harmonized for different uses (Fig. 1). Data sharing is just one use case and can involve sharing between divisions within a single agency, sharing between partners based on memorandums of understanding, or submission to public repositories.

The PHA4GE SARS-CoV-2 contextual data specification was created through broad consultation with representatives from public health laboratories, research institutes, and universities in 11 countries (Argentina, Australia, Brazil, Canada, Germany, Nigeria, Portugal, South Africa, Switzerland, the United Kingdom, the United States of America) who are involved with SARS-CoV-2 genome sequencing and analysis efforts at various scales. Based on this consultation and consensus, the specification contains different fields covering a wide array of data types described in Box 1 (Fig. 1). The specification attempts to harmonize different data standards (e.g., INSDC, GISAID, MIxS, MIGS, Sample Application Standard) by reusing fields or mapping to fields, as much as possible. Because PHA4GE embraces FAIR data stewardship principles (Findability, Accessibility, Interoperability, and Reuse of digital assets), we strived to implement FAIR principles in the design and implementation of the specification for data management and data sharing. At their core, these principles emphasize machine-actionability and consistency of data and are critical for dealing with the volume and complexity of genomic sequence and contextual data. Principles of FAIR data stewardship that have been implemented include improving machine-actionability of data by using a formal, accessible, shared, and broadly applicable language for knowledge representation, reusing existing standards and ontology-based vocabulary to increase interoperability, providing a data use license, capturing data provenance, and making all resources open, free, and widely accessible.

The versioned specification is available as a contextual data collection template (.xlsx) and in machine-amenable JSON format from GitHub (version 3.0.0) [55]. The collection template also offers standardized terms for a number of fields in the form of pick lists. The fields are colour-coded to indicate required (yellow), strongly recommended (purple), or optional status (white). Fields useful for surveillance were prioritized as "required". Formats for data elements like dates are also prescribed according to international standards (e.g., dates should be formatted according to ISO 8601).

The template is also supported by several materials such as term and field-level Reference Guides (available as tabs in the collection template Excel workbook), which provide definitions, data entry guidance, and examples of usage [55]. The field-level Reference Guide also provides mapping of PHA4GE fields to existing contextual data standards, highlighting public health and SARS-CoV-2–specific fields that were missing, as well as fields in those other standards that were considered out of scope.

The Open Biological and Biomedical Ontology (OBO) Foundry is a community of researchers who use a prescribed set of principles and practices to develop a wide range of interoperable ontologies focused on the life sciences [56]. Fields and terms in the specification have been mapped to existing OBO Foundry ontology terms, and where required, new ontology terms have been developed and are being made available in different application and domain-specific ontologies within The Foundry (see Table 1 for a list of source ontologies). As of version 3.0.0 and beyond, terms in pick lists provided in the collection template are presented with corresponding ontology identifiers in the format "Label [ontology ID]", e.g., Blood [UBERON:0 000 178]. Axioms and additional cross references to ontologies and existing standards are actively being developed in collaboration with community developers. We anticipate that our contributions to these freely available, open-source resources will be of use to the COVID-19 research community.

Protocols have also been created and are openly available on protocols.io [57], including a curation Standard Operating Procedure (SOP) containing instructions for using the collection template, as well as guidance for a number of privacy and practical concerns. A series of versioned SARS-CoV-2 sequence and contextual data submission protocols and accompanying instructional videos for how to prepare submissions and navigate through the various submission portals for GISAID, NCBI, and EMBL-EBI are also provided.

A mapping file indicating which PHA4GE fields correspond to contextual data elements recommended by the World Health Organization has been provided to help data providers comply with international guidance [58]. This mapping file also includes tabs indicating which PHA4GE fields correspond to those found in different repository submission forms to facilitate data transformations for submissions. Such transformations can be automated using a contextual data harmonization application called the DataHarmonizer [59]. PHA4GE has worked with the developers of the DataHarmonizer to offer the PHA4GE standard as a template in the tool (I. Gill et al., in preparation). Users can standardize and validate entered data and export it as GISAID and NCBI-ready submission forms (BioSample, SRA, GenBank, and GenBank source modifier forms). It should be noted that other excellent contextual data transformation tools have been developed by the community, such as METAGENOTE, multiSub, and a GISAID-to-ENA conversion script [60–62].

The different specification package materials are outlined in Table 2.

Getting Started—How To Use the Standard

In designing the specification we first considered the goals of data collection and harmonization. Consulted stakeholders believed that the primary priority of standardizing data should be improved support for SARS-CoV-2 genomic surveillance activities and the submission of sequence data and minimal metadata to public repositories. The two most important attributes for tracking transmission from pathogen genomic data are temporal information describing when a sample was collected and spatial information describing where a virus was sampled.

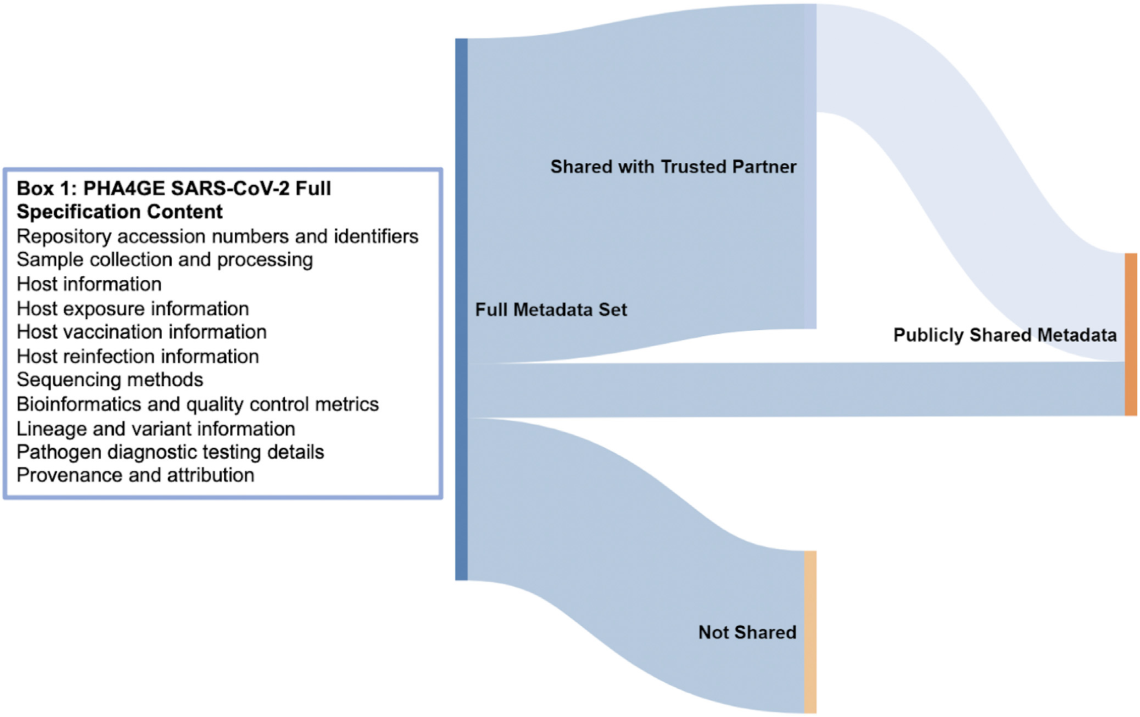


Figure 1: Contextual data flow. Contextual data can be captured and structured using the PHA4GE specification so that they can be more easily harmonized across different data sources and providers. Different subsets of the harmonized data can be (i) shared with public repositories, e.g., GISAID and INSDC; (ii) shared with trusted partners, e.g., national sequencing consortia, public health partners; and (iii) kept private and retained locally with the potential for sharing in the future for particular surveillance or research activities. While fields have been colour-coded in the template to indicate whether they are considered "required," "strongly recommended," or "optional," how the specification is implemented and whether any of the data are shared is ultimately at the discretion of the user. Box 1 describes the information types covered in the full specification.

Table 1: Ontologies implemented in the PHA4GE SARS-CoV-2 specification

Ontology[1]	**Link**
BRENDA Tissue Ontology (BTO)	https://obofoundry.org/ontology/bto.html
Cell Line Ontology (CLO)	https://obofoundry.org/ontology/clo.html
Environmental conditions, treatments and exposures ontology (ECTO)	https://obofoundry.org/ontology/ecto.html
Environment Ontology (ENVO)	https://obofoundry.org/ontology/envo.html
Food Ontology (FoodOn)	https://obofoundry.org/ontology/foodon.html
Gazetteer Ontology (GAZ)	https://obofoundry.org/ontology/gaz.html
Gender, Sex, and Sexual Orientation Ontology (GSSO)	https://obofoundry.org/ontology/gsso.html
Genomic Epidemiology Ontology (GenEpiO)	https://obofoundry.org/ontology/genepio.html
Genomics Cohorts Knowledge Ontology (GECKO)	https://obofoundry.org/ontology/gecko.html
Human Disease Ontology (DOID)	https://obofoundry.org/ontology/doid.html
Human Phenotype Ontology (HP)	https://obofoundry.org/ontology/hp.html
Mammalian Phenotype Ontology (MP)	https://obofoundry.org/ontology/mp.html
Measurement Method Ontology (MMO)	https://obofoundry.org/ontology/mmo.html
Mondo Disease Ontology (MONDO)	https://obofoundry.org/ontology/mondo.html
Mouse Pathology Ontology (MPATH)	https://obofoundry.org/ontology/mpath.html
National Cancer Institute Thesaurus (NCIT)	https://obofoundry.org/ontology/ncit.html
NCBI Taxonomy Ontology (NCBITaxon)	https://obofoundry.org/ontology/ncbitaxon.html
Neuro Behaviour Ontology (NBO)	https://obofoundry.org/ontology/nbo.html
Ontology for Biomedical Investigations (OBI)	https://obofoundry.org/ontology/obi.html
Ontology of Medically Related Social Entities (OMRSE)	https://obofoundry.org/ontology/omrse.html
Population and Community Ontology (PCO)	https://obofoundry.org/ontology/pco.html
UBERON Multi-species Anatomy Ontology (UBERON)	https://obofoundry.org/ontology/uberon.html
Unit Ontology (UO)	https://obofoundry.org/ontology/uo.html
Vaccine Ontology (VO)	https://obofoundry.org/ontology/vo.html

[1] Vocabulary for fields and terms in the specification have been sourced or mapped to OBO Foundry domain and application ontologies, which are highlighted in this list. New fields and terms for which there were no existing equivalents have been developed and submitted to these ontologies, expanding these community resources.

Table 2: Resources that form the PHA4GE SARS-CoV-2 contextual data specification package [55]

Resource[1]	Description	Link
Collection template and controlled vocabulary pick lists	Spreadsheet-based collection form containing different fields (identifiers and accessions, sample collection and processing, host information, host exposure, vaccination and reinfection information, lineage and variant information, sequencing, bioinformatics and quality control metrics, diagnostic testing information, author acknowledgements). Fields are colour-coded to indicate required, recommended, or optional status. Many fields offer pick lists of controlled vocabulary. Vocabulary lists are also available in a separate tab	https://github.com/pha4ge/SARS-CoV-2-Contextual-Data-Specification/raw/master/PHA4GE%20SARS-CoV-2%20Contextual%20Data%20Template.xls
Reference guides	Field and term definitions, guidance, and examples are provided as separate tabs in the collection template .xlsx file (see Term Reference Guide and Field Reference Guide)	https://github.com/pha4ge/SARS-CoV-2-Contextual-Data-Specification/raw/master/PHA4GE%20SARS-CoV-2%20Contextual%20Data%20Template.xlsx
Curation protocol on protocols.io	Step-by-step instructions for using the collection template are provided in an SOP. Ethical, practical, and privacy considerations are also discussed. Examples and instructions for structuring sample descriptions as well as sourcing additional standardized terms (outside those provided in pick lists) are also discussed	dx.doi.org/10.17504/protocols.io.btpznmp6
Mapping file of PHA4GE fields to metadata standards	PHA4GE fields are mapped to existing metadata standards such as the Sample Application Standard, MIxS 5.0, and the MIGS Virus Host-associated attribute package. Mappings are available in the Reference guide tab. Mappings highlight which fields of these standards are considered useful for SARS-CoV-2 public health surveillance and investigations, and which fields are considered out of scope	https://github.com/pha4ge/SARS-CoV-2-Contextual-Data-Specification/raw/master/PHA4GE%20SARS-CoV-2%20Contextual%20Data%20Template.xlsx
Mapping of PHA4GE fields to WHO metadata recommendations	PHA4GE fields are mapped to corresponding contextual data elements recommended by the World Health Organization	https://github.com/pha4ge/SARS-CoV-2-Contextual-Data-Specification/blob/master/PHA4GE%20to%20WHO%20and%20Sequence%20Repository%20Field%20Mappings.xlsx
Mapping file of PHA4GE fields to EMBL-EBI, NCBI, and GISAID submission requirements	Many PHA4GE fields have been sourced from public repository submission requirements. The different repositories have different requirements and field names. Repository submission fields have been mapped to PHA4GE fields to demonstrate equivalencies and divergences.	https://github.com/pha4ge/SARS-CoV-2-Contextual-Data-Specification/blob/master/PHA4GE%20to%20WHO%20and%20Sequence%20Repository%20Field%20Mappings.xlsx
Data submission protocol (NCBI) on protocols.io	The SARS-CoV-2 submission protocol for NCBI provides step-by-step instructions and recommendations aimed at improving interoperability and consistency of submitted data	dx.doi.org/10.17504/protocols.io.bui7nuhn
Data submission protocol (EMBL-EBI) on protocols.io	The SARS-CoV-2 submission protocol for ENA provides step-by-step instructions and recommendations aimed at improving interoperability and consistency of submitted data	dx.doi.org/10.17504/protocols.io.buqnnvve
Data submission protocol (GISAID) on protocols.io	The SARS-CoV-2 submission protocol for GISAID provides step-by-step instructions and recommendations aimed at improving interoperability and consistency of submitted data	dx.doi.org/10.17504/protocols.io.bumknu4w
JSON structure of PHA4GE specification	A JSON structure of the PHA4GE specification has been provided for easier integration into software applications	https://raw.githubusercontent.com/pha4ge/SARS-CoV-2-Contextual-Data-Specification/master/PHA4GE_SARS-CoV-2_Contextual_Data_Schema.json
PHA4GE template in the DataHarmonizer	Javascript application enabling standardized data entry, validation, and export of contextual data as submission-ready forms for GISAID and NCBI. The SOP for using the software can be found at https://github.com/Public-Health-Bioinformatics/DataHarmonizer/wiki/PHA4GE-SARS-CoV-2-Template	https://github.com/Public-Health-Bioinformatics/DataHarmonizer/releases

[1] There are a number of resources that form the PHA4GE SARS-CoV-2 contextual data specification package that are described in the table. The package has been compiled to support user implementation and data sharing, with integration into workflows and new software applications in mind. SOP: standard operating procedure.

Comparisons of minimal contextual data requirements across different national sequencing efforts, as well as submission requirements for INSDC and GISAID databases, yielded a minimal set of 14 fields that have been annotated as "required" in the specification (colour-coded yellow in the collection template). The required fields, corresponding definitions, and guidance notes are described in Table 3. A number of other fields have been annotated as "strongly recommended" (colour-coded purple in the collection template) for capturing sample collection and processing methods, critical epidemiological information about the host, and acknowledging scientific contributions. Fields colour-coded white are considered optional.

Because many contextual data fields are stored in different locations and databases (e.g., LIMS, epidemiology case report forms and databases), a benefit of implementing the PHA4GE collection template is that it enables the capture of these different pieces of information in one place. The collection template also offers pick lists for a variety of fields, e.g., a curated INSDC country list for "geo_loc name (country)," the standardized name of the virus under the "organism" field (i.e., severe acute respiratory coronavirus 2), and a multitude of standardized terms for sample types (anatomical materials and sites, environmental materials and sites, collection devices and methods). The "purpose of sequencing" field provides standardized tags that can be used to highlight sampling strategy criteria (e.g., baseline surveillance [random sampling] or targeted sequencing [non-random sampling]), which are very important for understanding bias when interpreting patterns in sequence data. The pick lists provided are neither exhaustive nor comprehensive but have been curated from current literature representing active sampling and surveillance activities.

If a pick list is missing standardized terms of interest, the reference guide also provides links to different ontology look-up services, enabling users to identify additional standardized terms. The reference guide provides definitions for the fields, additional guidance regarding the structure of the values in the field, and any suggestions for addressing issues pertaining to privacy and identifiability. The curation SOP provides users with step-by-step instructions for populating the template, looking up standardized terms, and how best to structure sample descriptions. The SOP also highlights a number of ethical, practical, and privacy considerations for data sharing.

Implementation of the PHA4GE specification around the world

The amount of and manner in which the specification is implemented is ultimately at the discretion of the user. To date, versions of the specification are being implemented in the CanCOGeN (Canada) and SPHERES (USA) SARS-CoV-2 sequencing initiatives, the AusTrakka (Australia and New Zealand) data sharing platform [1–3], and by the Global Emerging Pathogens Treatment Consortium (Africa) [63], the African Centre of Excellence for Genomics of Infectious Diseases (ACEGID) in Nigeria [64], the Baobab LIMS [65] at the South African National Bioinformatics Institute (SANBI) [66], and the Latin American Genomics Network [67].

Canada is implementing a version of the PHA4GE specification to harmonize contextual data across all data providers for national SARS-CoV-2 surveillance [5]. Harmonized contextual information is provided by different jurisdictions and stored in the national genomics surveillance database at the Public Health Agency of Canada's National Microbiology Laboratory. A hypothetical worked example is provided to demonstrate how free text information can be structured according to the specification and how subsets of the contextual data can be shared according to jurisdictional policies (Fig. 2).

While the primary use case of the specification is for public health sequencing, the sample collection fields have been developed to enable capture of information for a wide range of sample types, including environmental samples (e.g., swabs of hospital equipment and patient rooms, wastewater samples) and non-human hosts (e.g., wildlife, agricultural animal samples).

Submitting Data to Public Sequence Repositories

Many existing SARS-CoV-2 sequences have only been deposited in GISAID, with a proportion of submitters also depositing matching raw read data in the INSDC (i.e., NCBI, European Molecular Biology Laboratory–European Bioinformatics Institute [EMBL-EBI], and DNA Data Bank of Japan [DDBJ]). While consensus genomes are widely deposited and used for public surveillance purposes, raw read data are critical for comparing methods and assessing reproducibility, as well as identifying minor variants. Linkage of contextual data to consensus sequences as well as raw data in public repositories is vital.

Within the INSDC, the contextual data are stored as accessioned BioSamples [68] with a consistent set of attribute names and standardized values. BioSamples add value, promote reuse, and enable interoperability of data submitted from laboratories that may only be connected by following the same metadata standard. The INSDC databases have until recently provided a generic pathogen metadata template for the BioSample that is heavily utilized for bacterial genomic surveillance [69]. GISAID uses a different format and data structure for associating metadata primarily for influenza surveillance and now extended to include SARS-CoV-2. The ENA provides a virus metadata checklist (ENA virus pathogen reporting standard checklist) developed as part of the COMPARE project [70], which is very similar to the GISAID submission requirements.

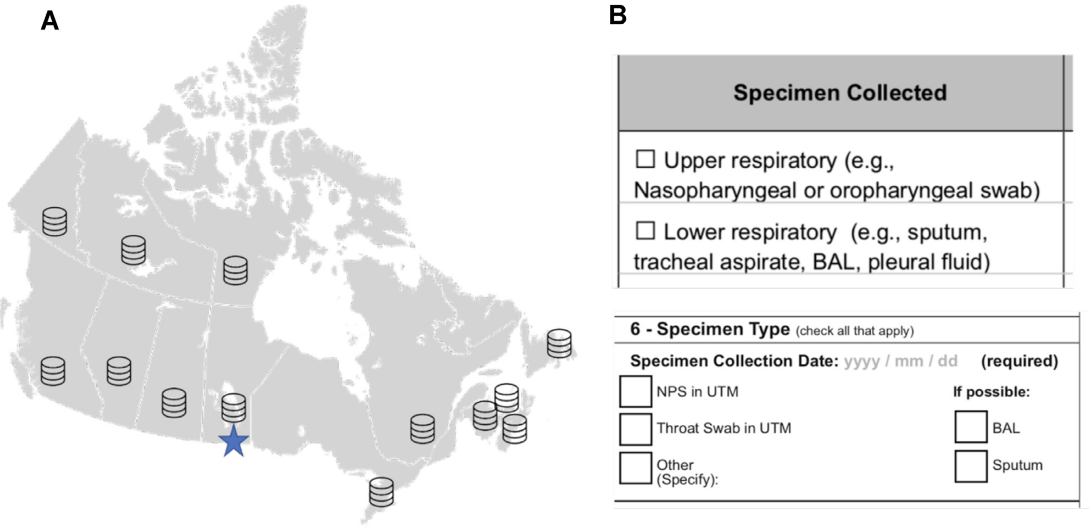
Building on these existing standards, a metadata specification for SARS-CoV-2 genomic surveillance was developed that is broad enough for internal laboratory use while providing mechanisms for mapping/transforming standardized contextual data for public release to INSDC and GISAID. Recently, PHA4GE worked with NCBI to develop a dedicated SARS-CoV-2 BioSample submission package in the NCBI Submission Portal, which incorporates many fields from the PHA4GE standard [71]. The Genomics Standards Consortium will also align its forthcoming "MIxS for SARS-CoV-2" package with this specification. EMBL-EBI will also offer the PHA4GE standard to submitters as one of its validated checklists. Taken together, the PHA4GE specification has already had widespread impact on contextual information data structures around the world.

The detailed mapping of PHA4GE fields to public repository submission requirements, as well as guidance and advice, are available as supporting documents (see Table 1). We have also provided detailed protocols for data submission to the three participating repositories, GenBank/SRA (NCBI), ENA (EMBL-EBI), and GISAID. An overview of how the PHA4GE specification is integrated into public repository submissions is presented in Fig. 3. PHA4GE recommendations for FAIR SARS-CoV-2 data submissions are as follows:

Table 3: Minimal (required) contextual data fields

Field name[1]	Definition	Guidance
specimen collector sample ID	The user-defined name for the sample	Every Sample ID from a single submitter must be unique. It can have any format, but we suggest that you make it concise, unique, and consistent within your laboratory, and as informative as possible
sample collected by	The name of the agency that collected the original sample	The name of the agency should be written out in full (with minor exceptions) and consistent across multiple submissions
sequence submitted by	The name of the agency that generated the sequence	The name of the agency should be written out in full (with minor exceptions) and be consistent across multiple submissions
sample collection date	The date on which the sample was collected	Record the collection date accurately in the template. Required granularity includes year, month, and day. Before sharing these data, ensure that this date is not considered identifiable information. If this date is considered identifiable, it is acceptable to add "jitter" to the collection date by adding or subtracting calendar days. Do not change the collection date in your original records. Alternatively, "received date" may be used as a substitute in the data you share. The date should be provided in ISO 8601 standard format "YYYY-MM-DD"
geo_loc name (country)	Country of origin of the sample	Provide the country name from the pick list in the template
geo_loc name (state/province/region)	State/province/region of origin of the sample	Provide the state/province/region name from the GAZ geography ontology. Search for geography terms at https://www.ebi.ac.uk/ols/ontologies/gaz
Organism	Taxonomic name of the organism	Use "Severe acute respiratory syndrome coronavirus 2"
Isolate	Identifier of the specific isolate	This identifier should be an unique, indexed, alphanumeric ID within your laboratory. If submitted to the INSDC, the "isolate" name is propagated throughout different databases. As such, structure the "isolate" name to be ICTV/INSDC compliant in the following format: "SARS-CoV-2/host/country/sampleID/date"
host (scientific name)	The taxonomic, or scientific name of the host	Common name or scientific name are required if there was a host. Scientific name example: *Homo sapiens*. Select a value from the pick list. If the sample was environmental, put "not applicable."
host disease	The name of the disease experienced by the host	This field is only required if there was a host. If the host was a human select COVID-19 from the pick list. If the host was asymptomatic, this can be recorded under "host health state details." "COVID-19" should still be provided if the patient is asymptomatic. If the host is not huma, and the disease state is not known or the host appears healthy, put "not applicable."
purpose of sequencing	The reason that the sample was sequenced	The reason why a sample was originally collected may differ from the reason why it was selected for sequencing. The reason a sample was sequenced may provide information about potential biases in sequencing strategy. Provide the purpose of sequencing from the pick list in the template. The reason for sample collection should be indicated in the "purpose of sampling" field
sequencing instrument	The model of the sequencing instrument used	Select a sequencing instrument from the pick list provided in the template
consensus sequence software name	The name of software used to generate the consensus sequence	Provide the name of the software used to generate the consensus sequence
consensus sequence software version	The version of the software used to generate the consensus sequence	Provide the version of the software used to generate the consensus sequence

[1] Through consultation and consensus, 14 fields were prioritized for SARS-CoV-2 surveillance, which are considered required in the specification. Field names, definitions, and guidance are presented.



Specimen Collected

☐ Upper respiratory (e.g., Nasopharyngeal or oropharyngeal swab)

☐ Lower respiratory (e.g., sputum, tracheal aspirate, BAL, pleural fluid)

6 - Specimen Type (check all that apply)

Specimen Collection Date: yyyy / mm / dd **(required)**

☐ NPS in UTM

☐ Throat Swab in UTM

☐ Other (Specify):

If possible:

☐ BAL

☐ Sputum

C

Anonymised Example Data:

Province A screens travelers as part of an international border testing program. A sample is selected by the lab team (Johnny Bloggs, Bhav Singh, Tina Lee) for sequencing as part of the **travel surveillance** program. The sample, collected **on November 13 2020**, is a **nasal swab** from an individual who **shares a household with a known case** that **recently traveled to country X**. The individual was **asymptomatic**. Diagnostic RT-qPCR testing based on the E gene yielded a **CT value of 23**. The individual was a **43 year old female**. The lab implements the **ARTIC protocol** to perform amplicon sequencing on an **Illumina MiSeq** using the **primer scheme described by Freed et al (2020), and ncov-tools** for bioinformatic processing and analysis.

Standardized Contextual Data:

specimen collector sample ID: provA_12345
sample collected by: Province A Public Health Lab
sample collector contact email: provlabA@lab.ca
sequence submitted by: Province A Public Health Lab
sequence submitter contact email: provlabA@lab.ca
geo_loc_name (country): Canada [GAZ:00002560]
geo_loc_name (state/province/region): Province A
sample collection date: 2020-11-13
anatomical site: Nasopharynx (NP) [UBERON:0001728]
collection device: Swab [GENEPIO:0100027]
purpose of sampling: Diagnostic testing [GENEPIO:0100002]
purpose of sequencing: International travel surveillance [GENEPIO:0100014]
host (scientific name): Severe acute respiratory syndrome coronavirus 2 [NCBITaxon:2697049]
host disease: COVID-19 [MONDO:0100096]
host age: 43
host age bin: 40 – 49 [GENEPIO:0100053]
host age unit: year [UO:0000036]
host gender: Female [NCIT:C46110]
host health state: Asymptomatic [NCIT:C3833]
exposure setting: Contact with known COVID-19 case [GENEPIO:0100184]
sequencing instrument: Illumina MiSeq [GENEPIO:0100125]
sequencing protocol name: ARTIC protocol
amplicon pcr primer scheme: Freed at al (2020)
amplicon size: 1200 bp
raw sequence data processing method: https://github.com/jts/ncov-tools
dehosting method: https://github.com/jts/ncov-tools
consensus sequence software name: Freebayes
consensus sequence software version: 1.3.2
gene name 1: E gene (orf4) [GENEPIO:0100151]
diagnostic pcr Ct value 1: 23
authors: Johnny Bloggs, Bhav Singh, Tina Lee

Figure 2: The PHA4GE specification is being implemented in CanCOGeN to harmonize contextual data across jurisdictions. (A) CanCOGeN is Canada's SARS-CoV-2 national genomic surveillance initiative. Canada has a decentralized health system, with one federal and 13 provincial/territorial public health jurisdictions. Provinces/Territories have authority over how data are collected, stored, and shared. Every Canadian public health jurisdiction uses different collection instruments (e.g., case report forms), different data management systems, and different pipelines and software to perform bioinformatic analyses. Provinces/Territories share sequencing data and accompanying contextual data with the National Microbiology Lab's national SARS-CoV-2 genomics database (starred) according to a version of the PHA4GE specification for national surveillance activities. (B) Excerpts from two different province-specific case collection forms. Sample type information is collected in data collection instruments using different fields, different terms, at different levels of granularity, using abbreviations and formats. BAL: bronchoalveolar lavage; NPS: nasopharyngeal swab; UTM: universal transport medium. (C) An anonymized example of how the standard consistently structures contextual information and how it is being used for data sharing. The contextual data specification provides a wide variety of fields and pick lists of terms. In the example, the full set of standardized information shown would be shared by the province with the national database. Standardized information in boldface would be shared with public repositories; however select data elements (underscored) would be withheld according to jurisdictional data sharing policies. The specification enables users to harmonize and integrate data provenance, sampling strategy criteria, epidemiological information, and methods.

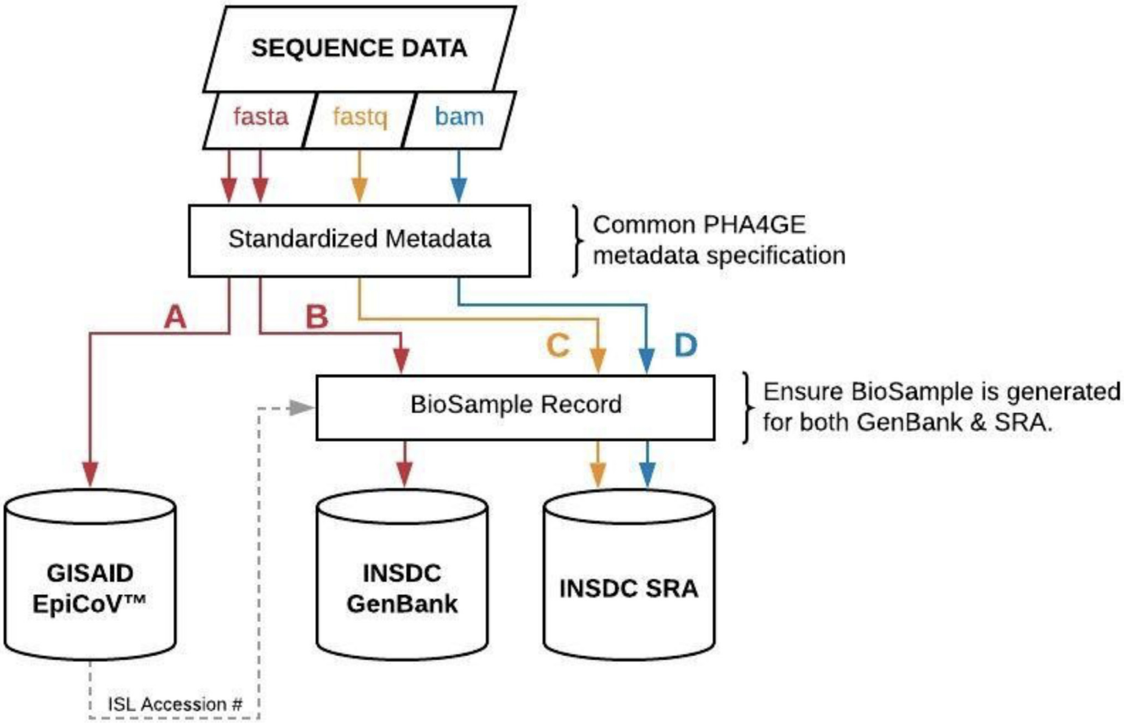


Figure 3: Overview of how the PHA4GE SARS-CoV-2 contextual data specification can be integrated into public repository submission. The PHA4GE collection template provides a one-stop shop for different data types that are important for global surveillance. The protocols provided as part of the specification package describe how PHA4GE fields can be mapped to different repository submission forms. Consensus sequences (FASTA), accompanied by a subset of PHA4GE fields, can be submitted to the GISAID EpiCoV database (A). Consensus sequences (FASTA) (B) as well as raw/processed data (FASTQ, BAM) (C, D) can be submitted to INSDC databases (e.g., GenBank, SRA) with different subsets of PHA4GE fields as part of a BioSample record. BioSamples are propagated throughout INSDC databases.

1. submit raw sequencing data and assembled/consensus genomes to INSDC and GISAID when permitted by jurisdictional data-sharing policies
2. create a BioSample record when submitting to the INSDC using the PHA4GE guidance, populating the mandatory and recommended fields where possible
3. curate public records (sequence data and contextual data), updating them when subsequent information becomes available or retracting if/when records become untrustworthy.

The specification has been used to submit standardized contextual data to different repositories by laboratories and sequencing initiatives globally. A selection of accession numbers for submissions to different repositories is provided in Table 4.

Conclusion

The collective response to the SARS-CoV-2 pandemic has resulted in an unprecedented deployment of genomic surveillance worldwide, bringing together public health agencies, academic research institutions, and industry partners. This unified action provides opportunities to more effectively understand and respond to the pandemic. Yet it also provides an enormous challenge because realizing the full potential of this opportunity will require standardization and harmonization of data collection across these partners. With our SARS-CoV-2 metadata specification we have endeavoured to create a mechanism for promoting consistent, standardized contextual data collection that can be applied broadly. We envision that given the increased uptake, this specification will improve the consistency of collected data, making information reusable by agencies as they continue working towards an increased understanding of SARS-CoV-2 epidemiological and biological characteristics, and harmonizing them such that community-based data-sharing efforts are not excessively burdened. We anticipate that the experience and lessons learned creating the specification package for SARS-CoV-2 will better enable the rapid development and deployment of pathogen-specific standards for public health pathogen genomic surveillance in the future.

Methods

The PHA4GE SARS-CoV-2 data specification was developed by first comparing existing metadata standards (e.g., MIxS/MIGS, the NIAID/BRC Sample Application Standard) and various sequence repository submission requirements (e.g., GISAID, INSDC), as well as national and international case report forms.

A gap analysis was performed to identify SARS-CoV-2 public health surveillance data elements that were missing in available standards. Fields in existing standards that were deemed to be out of scope were excluded from the specification. Terms for pick lists were sourced from public health documents, the literature, and, when available, various interoperable ontologies (OBO Foundry). The fields and terms from the gap analysis were structured in the collection template (.xlsx). Field definitions, guidance for use, examples, and mappings to various standards were developed as part of the Reference Guides provided in separate tabs in the template workbook. Vocabulary lists were also provided in a separate tab in the template workbook to enable validation and to enable users to add terms to pick lists as needed, according to instruc-

Table 4: A selection of accession numbers of harmonized contextual data records submitted to different public repositories

Data contributor	Repository	Accession No.
African Centre of Excellence for Genomics of Infectious Diseases (Nigeria)	GISAID	EPI_ISL_1 035 827
		EPI_ISL_1 035 826
		EPI_ISL_1 035 825
COVID-19 Genomic Surveillance Regional Network (Latin America)	GISAID	EPI_ISL_2 158 821
		EPI_ISL_2 158 802
		EPI_ISL_2 158 810
COVID-19 Genomic Surveillance Regional Network (Latin America)	EMBL-EBI	SAMEA8968916
Rhode Island Department of Health/Broad Institute (SPHERES)	NCBI	SAMN18306978
Massachusetts General Hospital/Broad Institute (SPHERES)	NCBI	SAMN18309294
Flow Health/Broad Institute (SPHERES)	NCBI	SAMN18308763
New Brunswick Diagnostic Virology Reference Center/Public Health Agency of Canada (CanCOGeN)	NCBI	SAMN16784832
Toronto Invasive Bacterial Diseases Network/McMaster University (CanCOGeN)	NCBI	SAMN17505317
Bat coronavirus phylogeography—Université de La Réunion, UMR Processus Infectieux en Milieu Insulaire Tropical (PIMIT) and Field Museum of Natural History	NCBI	SAMN20400589
		SAMN20400588

tions provided in the curation SOP. The specification was also encoded as a JSON file.

The specification was reviewed by public health, bioinformatics, and data standards experts from different public health agencies, research institutes, and sequencing consortia and adapted according to feedback. Upon request by community members, versioned protocols for public repository submission were created and deposited in protocols.io.

The first version of the specification was made publicly available in August 2020 with a CC-BY 4.0 International attribution license. Iterative improvements were made to a development branch of the specification over the next 10 months as the pandemic evolved, and in response to user feedback and requests. The second major release (2.0) was made publicly available in May 2021. A third major release (3.0) including ontology mappings and the term-level reference guide was made publicly available in December 2021. The PHA4GE template was incorporated into the contextual data harmonization, validation, and transformation tool called The DataHarmonizer through a collaborative effort with the Centre for Infectious Disease Genomics and One Health (Simon Fraser University). Details regarding DataHarmonizer development can be found elsewhere (e.g., [72] and manuscript in preparation (I. Gill et al., in preparation).).

Availability and Requirements

- Project name: SARS-CoV-2-Contextual-Data-Specification
- Project home page: https://github.com/pha4ge/SARS-CoV-2-Contextual-Data-Specification
- Operating system: Platform independent
- Programming language: Not applicable
- Other requirements: xlsx-compatible spreadsheet software
- License: CC-BY 4.0 International
- RRID:SCR_021378
- biotools:pha4ge_sars-cov-2_contextual_data_specification

Data Availability

Snapshots of the specification and DataHarmonizer are available in the *GigaScience* GigaDB repository [73].

Abbreviations

ACEGID: African Center of Excellence for Genomics of Infectious Diseases; CanCOGeN: Canadian COVID Genomics Network; COG-UK: COVID-19 Genomics UK Consortium; COVID-19: coronavirus disease of 2019; EBI: European Bioinformatics Institute; EFO: Experimental Phenotype Ontology; EMBL-EBI: European Molecular Biology Laboratory's European Bioinformatics Institute; ENA: European Nucleotide Archive; FAIR: Findable, Accessible, Interoperable, Reusable; GAZ: Gazetteer Ontology; GenEpiO: Genomic Epidemiology Ontology; GISAID: Global Initiative on Sharing All Influenza Data; HP: Human Phenotype Ontology; INSDC: International Nucleotide Sequence Database Collaboration; INSACOG: Indian SARS-CoV-2 Genomics Consortium; JSON: JavaScript Object Notation; LIMS: Laboratory Information Management System; MIGS: Minimum Information about a Genomic Sequence; MIxS: Minimum Information about any Sequence; MP: Mammalian Phenotype Ontology; NCBI: National Center for Biotechnology Information; NCBITaxon: NCBI Taxonomy Ontology; NCIT: National Cancer Institute Thesaurus; OBI: Ontology for Biological Investigations; OBO Foundry: Open Biological and Biomedical Ontology Foundry; PHA4GE: Public Health Alliance for Genomic Epidemiology; SANBI: South African National Bioinformatics Institute; SARS-CoV-2: severe acute respiratory syndrome coronavirus 2; SOP: standard operating procedure; SPHERES: SARS-CoV-2 Sequencing for Public Health Emergency Response, Epidemiology and Surveillance; SRA: Sequence Read Archive; UBERON: Uber-Anatomy Ontology; UO: Unit Ontology; WHO: World Health Organization.

Competing Interests

The authors declare that they have no competing interests.

Funding

The Bill & Melinda Gates Foundation supported the establishment and work of the PHA4GE consortium. A.J.P. and N.F.A. were supported by the Biotechnology and Biological Sciences Research Council (BBSRC), the Quadram Institute Bioscience BBSRC funded Core Capability Grant (project No. BB/CCG1860/1), and the BBSRC Institute Strategic Programme Microbes in the Food Chain BB/R012504/1 and its constituent project BBS/E/F/000PR10352. F.M. was supported by a Donald Hill Family Fellowship in Computer Science. C.I.M. was supported by the Fundação para a Ciência e Tecnologia (grant SFRH/BD/129483/2017). Work by E.J.G., R.C., D.D., and W.W.L.H. was funded by a Genome Canada Bioinformatics and Computational Biology 2017 Grant #286GET and a Genome Canada CanCOGeN grant E09CMA. The work of I.K.M. T.B., and A.J. was supported by the National Center for Biotechnology Information of the National Library of Medicine (NLM), National Institutes of Health.

Authors' Contributions

E.J.G.: Conceptualization, Methodology, Investigation, Software, Visualization, Writing—Original Draft Preparation, Validation, Supervision; R.E.T.: Methodology, Investigation, Software, Validation, Writing—Original Draft Preparation; C.I.M.: Methodology, Software, Writing—Review & Editing; A.J.P.: Methodology, Writing—Original Draft Preparation; N.F.A.: Methodology, Software, Validation, Writing—Original Draft Preparation; D.F.: Methodology, Software; F.M.: Writing-Review and Editing, J.C.: Validation, Writing—Review & Editing; D.P.: Validation, Writing—Review & Editing; I.B.O.: Validation, Writing—Review & Editing; D.A.: Software, Validation, Writing—Review & Editing; A.C.: Writing—Review & Editing; A.G.S.: Software, Validation, Writing—Review & Editing; R.C.: Software, Validation; D.D.: Software, Validation; L.S.K.: Validation, Writing—Review & Editing; A.B.: Methodology, Writing—Original Draft Preparation; I.K.M.: Software, Validation, Writing—Review & Editing; T.B.: Software, Validation, Writing—Review & Editing; A.J.: Software, Validation, Writing—Review & Editing; T.R.C.: Validation, Writing—Review & Editing; S.M.N.: Validation, Writing—Review & Editing; A.A.W.: Writing—Review & Editing; P.E.O.: Writing—Review & Editing; G.H.T.: Writing—Review & Editing; S.H.T.: Writing—Review & Editing; A.R.R.: Writing—Review & Editing; B.A.: Writing—Review & Editing; D.M.A.: Writing—Review & Editing; E.H.: Writing—Review & Editing; W.W.L.H.: Writing—Review & Editing; A.T.R.V.: Writing—Review & Editing; D.R.M.: Conceptualization, Methodology, Visualization, Writing—Review & Editing, Funding Acquisition

Acknowledgements

The authors thank the US Centers for Disease Control and Prevention's Technical Outreach and Assistance for States Team (TOAST) for their feedback, support, and assistance in disseminating the PHA4GE specification package among US public health networks.

References

1. World Health Organization. Coronavirus disease (COVID-19). https://www.who.int/emergencies/diseases/novel-coronavirus-2019. Accessed 21 June 2021.
2. Dong, E, Du, H, Gardner, L. An interactive web-based dashboard to track COVID-19 in real time. *Lancet Infect Dis* 2020;**20**(5):533–4.
3. The COVID-19 Genomics UK (COG-UK) consortium. An integrated national scale SARS-CoV-2 genomic surveillance network. *Lancet Microbe* 2020;**1**(3):e99–e100.
4. Centers for Disease Control and Prevention. SPHERES: SARS-CoV-2 Sequencing for Public Health Emergency Response, Epidemiology, and Surveillance. https://www.cdc.gov/coronavirus/2019-ncov/variants/spheres.html. Accessed 22 June 2021.
5. Genome Canada. Canadian COVID-19 Genomics Network (CanCOGeN). https://www.genomecanada.ca/en/cancogen. Accessed 22 June 2021.
6. Pan American Health Organization. Laboratory guidelines for the detection and diagnosis of COVID-19 virus infection. https://www.paho.org/en/documents/laboratory-guidelines-detection-and-diagnosis-covid-19-virus-infection. Accessed 22 June 2021.
7. Candido, DS, Claro, IM, De Jesus, JG, *et al.* Evolution and epidemic spread of SARS-CoV-2 in Brazil. *Science* 2020;**369**(6508):1255–60.
8. Zhao, WM, Song, S-H, Chen, M-L, *et al.* The 2019 novel coronavirus resource. *Yi Chuan Hered* 2020;**42**:212–21.
9. NGS-SA: Network for Genomic Surveillance South Africa. http://www.krisp.org.za/ngs-sa/ngs-sa_network_for_genomic_surveillance_south_africa/. Accessed 22 June 2021.
10. AusTrakka. https://www.cdgn.org.au/austrakka. Accessed 22 June 2021.
11. Indian SARS-CoV-2 Genomics Consortium (INSACOG). http://dbtindia.gov.in/insacog. Accessed 21 June 2021.
12. Shu, Y, Mccauley, J. GISAID: global initiative on sharing all influenza data – from vision to reality. *Eurosurveillance* 2017;**22**(13):30494.
13. Karsch-Mizrachi, I, Takagi, T, *et al.* The International Nucleotide Sequence Database Collaboration. *Nucleic Acids Res* 2018;**46**(D1):D48–51.
14. Allard, MW, Strain, E, Melka, D, *et al.* Practical value of food pathogen traceability through building a whole-genome sequencing network and database. *J Clin Microbiol* 2016;**54**(8):1975–83.
15. Kubota, KA, Wolfgang, WJ, Baker, DJ, *et al.* PulseNet and the changing paradigm of laboratory-based surveillance for foodborne diseases. *Public Health Rep* 2019;**134**(2_suppl):22S–8S.
16. Cook, JA, Arai, S, Armién, B, *et al.* Integrating biodiversity infrastructure into pathogen discovery and mitigation of emerging infectious diseases. *Bioscience* 2020:**70**(7):531–4.
17. Andersen, KG, Rambaut, A, Lipkin, WI, *et al.* The proximal origin of SARS-CoV-2. *Nat Med* 2020;**26**(4):450–2.
18. Gupta, RK. Will SARS-CoV-2 variants of concern affect the promise of vaccines?. *Nat Rev Immunol* 2021;**21**(6):340–1.
19. Public Health England. *Technical Briefing 16: SARS-CoV-2 variants of concern and variants under investigation.* https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/994839/Variants_of_Concern_VOC_Technical_Briefing_16.pdf. Accessed 22 June 2021.
20. Po-E, L, Guttierez, A, Davenport, K, *et al.*. A public website for the automated assessment and validation of SARS-CoV-2 diagnostic PCR assays. *Bioinformatics* 2021;1024–5.
21. Kuchinski, KS, Nguyen, J, Lee, TD, *et al.* Mutations in emerging variant of concern lineages disrupt genomic sequencing of SARS-CoV-2 clinical specimens. *Int J Infect Dis* 2022;**114**:51–4.
22. Ganguli, A, Mostafa, A, Berger, J, *et al.* Rapid isothermal amplification and portable detection system for SARS-CoV-2. *Proc Natl Acad Sci U S A* 2020;**117**(37):22727–35.
23. World Health Organization. COVID-19 vaccine tracker and landscape. https://www.who.int/publications/m/item/draft-landscape-of-covid-19-candidate-vaccines. Accessed 22 June 2021.
24. Tillett, RL, Sevinsky, JR, Hartley, PD, *et al.* Genomic evidence for reinfection with SARS-CoV-2: a case study. *Lancet Infect Dis* 2021;**21**(1):52–58.
25. Oude Munnink, BB, Sikkema, RS, Nieuwenhuijse, DF, *et al.* Transmission of SARS-CoV-2 on mink farms between humans and mink and back to humans. *Science* 2021;**371**(6525):172–7.
26. Lai, C-C, Wang, J-H, Ko, W-C, *et al.* COVID-19 in long-term care facilities: an upcoming threat that cannot be ignored. *J Microbiol Immunol Infect* 2020;**53**(3):444–6.

27. Aggarwal, D, Myers, R, Hamilton, WL, *et al.* The role of viral genomics in understanding COVID-19 outbreaks in long-term care facilities. *Lancet Microbe* 2021: doi:10.1016/S2666-5247(21)00208-1.
28. Murti, M, Goetz, M, Saunders, A, *et al.* Investigation of a severe SARS-CoV-2 outbreak in a long-term care home early in the pandemic. *Can Med Assoc J* 2021;**193**(19):E681–8.
29. Dyal, JW, Grant, MP, Broadwater, K, *et al.* COVID-19 among workers in meat and poultry processing facilities—19 States, April 2020. *MMWR Morb Mortal Wkly Rep* 2020;**69**(18): doi:10.15585/mmwr.mm6918e3.
30. Günther, T, Czech-Sioli, M, Indenbirken, D, *et al.* SARS-CoV-2 outbreak investigation in a German meat processing plant. *EMBO Mol Med* 2020;**12**(12):e13296.
31. Taylor, J, Carter, RJ, Lehnertz, N, *et al.* Serial testing for SARS-CoV-2 and virus whole genome sequencing inform infection risk at two skilled nursing facilities with COVID-19 Outbreaks - Minnesota, April-June 2020. *MMWR Morb Mortal Wkly Rep* 2020;**69**(37):1288–95.
32. Loconsole, D, Sallustio, A, Accogli, M, *et al.* Investigation of an outbreak of symptomatic SARS-CoV-2 VOC 202012/01-lineage B.1.1.7 infection in healthcare workers, Italy. *Clin Microbiol Infect* 2021;**27**(8):1174.e1–e4.
33. Frampton, D, Rampling, T, Cross, A, *et al.* Genomic characteristics and clinical effect of the emergent SARS-CoV-2 B.1.1.7 lineage in London, UK: a whole-genome sequencing and hospital-based cohort study. *Lancet Infect Dis* 2021;**21**(9):1246–56.
34. Da Silva Filipe, A, Shepherd, JG, Williams, T, *et al.* Genomic epidemiology reveals multiple introductions of SARS-CoV-2 from mainland Europe into Scotland. *Nat Microbiol* 2021;**6**(1):112–22.
35. Oude Munnink, BB, Nieuwenhuijse, DF, Stein, M, *et al.* Rapid SARS-CoV-2 whole-genome sequencing and analysis for informed public health decision-making in the Netherlands. *Nat Med* 2020;**26**(9):1405–10.
36. Du Plessis, L, Mccrone, JT, Zarebski, AE, *et al.* Establishment and lineage dynamics of the SARS-CoV-2 epidemic in the UK. *Science* 2021;**371**(6530):708–12.
37. Githinji, G, De Laurent, ZR, Mohammed, KS, *et al.* Tracking the introduction and spread of SARS-CoV-2 in coastal Kenya. *Nat Commun* 2021;**12**(1):4809.
38. Meredith, LW, Hamilton, WL, Warne, B, *et al.* Rapid implementation of SARS-CoV-2 sequencing to investigate cases of health-care associated COVID-19: a prospective genomic surveillance study. *Lancet Infect Dis* 2020;**20**(11):1263–71.
39. Zhang, W, Govindavari, JP, Davis, BD, *et al.* Analysis of genomic characteristics and transmission routes of patients with confirmed SARS-CoV-2 in Southern California during the early stage of the US COVID-19 pandemic. *JAMA Network Open* 2020;**3**(10):e2024191.
40. Long, S, Olsen, RJ, Christensen, PA, *et al.* Molecular architecture of early dissemination and massive second wave of the SARS-CoV-2 virus in a major metropolitan area. *mBio* 2020;**11**(6): e02707–20.
41. Geoghegan, JL, Ren, X, Storey, M, *et al.* Genomic epidemiology reveals transmission patterns and dynamics of SARS-CoV-2 in Aotearoa New Zealand. *Nat Commun* 2020;**11**(1):6351.
42. Seemann, T, Lane, CR, Sherry, NL, *et al.* Tracking the COVID-19 pandemic in Australia using genomics. *Nat Commun* 2020;**11**(1):4376.
43. McLaughlin, A, Montoya, V, Miller, RL, *et al.* Early and ongoing importations of SARS-CoV-2 in Canada. medRxiv 2021: doi:10.1101/2021.04.09.21255131.
44. Fauver, JR, Petrone, ME, Hodcroft, EB, *et al.* Coast-to-coast spread of SARS-CoV-2 during the early epidemic in the United States. *Cell* 2020;**181**(5):990–6.e5.
45. Knock, ES, Whittles, LK, Lees, JA, *et al.* Key epidemiological drivers and impact of interventions in the 2020 SARS-CoV-2 epidemic in England. *Sci Transl Med* 2021;**13**(602): doi:10.1126/scitranslmed.abg4262.
46. Lane, CR, Sherry, NL, Porter, AF, *et al.* Genomics-informed responses in the elimination of COVID-19 in Victoria, Australia: an observational, genomic epidemiological study. *Lancet Public Health* 2021;**6**(8):e547–56.
47. De Maio, N, Walker, C, Borges, R, *et al.*. Issues with SARS-CoV-2 sequencing data. *Virological* 2020. https://virological.org/t/issues-with-sars-cov-2-sequencing-data/473. Accessed 22 June 2021.
48. Rayko, M, Komissarov, A. Quality control of low-frequency variants in SARS-CoV-2 genomes. bioRxiv 2020: doi:10.1101/2020.04.26.062422.
49. Poon, LLM, Leung, CSW, Chan, KH, *et al.* Recurrent mutations associated with isolation and passage of SARS coronavirus in cells from non-human primates. *J Med Virol* 2005;**76**(4):435–40.
50. Yilmaz, P, Kottmann, R, Field, D, *et al.* Minimum information about a marker gene sequence (MIMARKS) and minimum information about any (x) sequence (MIxS) specifications. *Nat Biotechnol* 2011;**29**(5):415–20.
51. Field, D, Garrity, G, Gray, T, *et al.* The minimum information about a genome sequence (MIGS) specification. *Nat Biotechnol* 2008;**26**(5):541–7.
52. Dugan, VG, Emrich, SJ, Giraldo-Calderón, GI, *et al.* Standardized metadata for human pathogen/vector genomic sequences. *PLoS One* 2014;**9**(6):e99979.
53. Smith, B, Ashburner, M, Rosse, C, *et al.* The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nat Biotechnol* 2007;**25**(11):1251–5.
54. Schriml, LM, Chuvochina, M, Davies, N, *et al.* COVID-19 pandemic reveals the peril of ignoring metadata standards. *Sci Data* 2020;**7**(1):188.
55. The PHA4GE SARS-CoV-2 Contextual Data Specification. https://github.com/pha4ge/SARS-CoV-2-Contextual-Data-Specification. Accessed 22 June 2021.
56. The OBO Foundry. http://www.obofoundry.org. Accessed 22 June 2021.
57. PHA4GE - research group on protocols.io. Protocols.io. https://www.protocols.io/groups/pha4ge. Accessed 22 June 2021.
58. World Health Organization. Guidance for surveillance of SARS-CoV-2 variants: interim guidance. WHO/2019-nCoV/surveillance/variants2021.1. https://www.who.int/publications/i/item/WHO_2019-nCoV_surveillance_variants. Accessed: 10 August 2021.
59. Hsiao Public Health Bioinformatics Lab. The DataHarmonizer. https://github.com/Public-Health-Bioinformatics/DataHarmonizer. Accessed 22 June 2021.
60. METAGENOTE. https://metagenote.niaid.nih.gov. Accessed 13 December 2021.
61. multiSub. https://github.com/maximilianh/multiSub. Accessed 13 December 2021.
62. gisaid-to-ena script. https://github.com/enasequence/ena-content-dataflow/tree/master/scripts/gisaid_to_ena. Accessed 13 December 2021.
63. GET Africa – ONE AFRICA, ONE HEALTH, ONE DESTINY. https://www.getafrica.org/. Accessed 22 June 2021.
64. African Centre of Excellence in Genomics of Infectious Diseases (ACEGID). https://acegid.org/. Accessed 1 September 2021.

65. Baobab LIMS. https://baobablims.org/. Accessed 22 June 2021.
66. SANBI – South African National Bioinformatics Institute. https://www.sanbi.ac.za. Accessed 22 June 2021.
67. COVID-19 Genomic Surveillance Regional Network - PAHO/WHO. https://www.paho.org/en/topics/influenza/covid-19-genomic-surveillance-regional-network. Accessed 22 June 2021.
68. Barrett, T, Clark, K, Gevorgyan, R, *et al.* BioProject and BioSample databases at NCBI: facilitating capture and organization of metadata. *Nucleic Acids Res* 2012;**40**(D1): D57–63.
69. NCBI Pathogen Detection Portal. https://www.ncbi.nlm.nih.gov/pathogens/. Accessed 22 June 2021.
70. Compare Europe. https://www.compare-europe.eu/. Accessed 22 June 2021.
71. Dedicated SARS-CoV-2 BioSample submission package in the NCBI Submission Portal. https://ncbiinsights.ncbi.nlm.nih.gov/2021/05/11/sars-cov-2-biosample-submission-package/. Accessed 22 June 2021.
72. The DataHarmonizer: A contextual data tool for curation, validation and transformation. https://github.com/cidgoh/DataHarmonizer. Accessed 1 September 2021.
73. Griffiths E, Timme, RE, Mendes, CI, *et al.* Supporting data for "Future-proofing and maximizing the utility of metadata: the PHA4GE SARS-CoV-2 contextual data specification package." *GigaScience Database* 2022. http://dx.doi.org/10.5524/100977.

MICROBIAL GENOMICS

PERSONAL VIEW
van der Putten *et al*., *Microbial Genomics* 2022;8:000790
DOI 10.1099/mgen.0.000790

MICROBIOLOGY SOCIETY
OPEN DATA OPEN ACCESS

Software testing in microbial bioinformatics: a call to action

Boas C.L. van der Putten[1,2,*,†], C. I. Mendes[3]†, Brooke M. Talbot[4], Jolinda de Korne-Elenbaas[1,5], Rafael Mamede[3], Pedro Vila-Cerqueira[3], Luis Pedro Coelho[6,7], Christopher A. Gulvik[8], Lee S. Katz[9,10] and The ASM NGS 2020 Hackathon participants

Abstract

Computational algorithms have become an essential component of research, with great efforts by the scientific community to raise standards on development and distribution of code. Despite these efforts, sustainability and reproducibility are major issues since continued validation through software testing is still not a widely adopted practice. Here, we report seven recommendations that help researchers implement software testing in microbial bioinformatics. We have developed these recommendations based on our experience from a collaborative hackathon organised prior to the American Society for Microbiology Next Generation Sequencing (ASM NGS) 2020 conference. We also present a repository hosting examples and guidelines for testing, available from https://github.com/microbinfie-hackathon2020/CSIS.

BACKGROUND

Computational algorithms, software, and workflows have enhanced the breadth and depth of microbiological research and expanded the capacity of infectious disease surveillance in public health practice. Scientists now have a wealth of bioinformatic tools for addressing pertinent questions quickly and keeping pace with the availability of larger and more complex biological datasets. Despite these advances, we are finding ourselves in a crisis of computational reproducibility [1].

Modern software engineering advocates reliable software testing standards and best practices. Different approaches are employed: from unit testing to system testing [2], going from testing every individual component to testing a tool as a whole (Fig. 1). The extent of testing is a balance between the resources available and increasing sustainability and reproducibility. Continuous Integration (CI), where code changes are frequently integrated and assertion of the new code's correctness before integration is often automatedly performed through tests, provides a robust approach for ensuring the reproducibility of scientific results without requiring human interaction. Comprehensive testing of scientific software might prevent computational errors which subsequently lead to erroneous results and retractions [3, 4]. However, the role of testing extends beyond that, as it also provides a way to measure software coverage, and therefore its robustness, allowing for reported issues to be converted into testable actions (regression tests), and the expansion and refactoring of existing code without compromising its function.

Software testing among peers across fields aligns with previous efforts of hackathons to create a more unified and informed bioinformatics software community [5]. In this context, we hosted a cooperative hackathon prior to the ASM NGS conference in

---

Received 14 October 2021; Accepted 02 February 2022; Published 08 March 2022

Author affiliations: [1]Department of Medical Microbiology, Amsterdam UMC, University of Amsterdam, the Netherlands; [2]Department of Global Health, Amsterdam Institute for Global Health and Development, Amsterdam UMC, University of Amsterdam, the Netherlands; [3]Instituto de Microbiologia, Instituto de Medicina Molecular, Faculdade de Medicina, Universidade de Lisboa, Lisboa, Portugal; [4]Department of Biological and Biomedical Sciences, Emory University, Atlanta, GA, USA; [5]Department of Infectious Diseases, Public Health Laboratory, Public Health Service of Amsterdam, the Netherlands; [6]Institute of Science and Technology for Brain-Inspired Intelligence, Fudan University, PR China; [7]Key Laboratory of Computational Neuroscience and Brain-Inspired Intelligence, PR China; [8]Bacterial Special Pathogens Branch, Division of High-Consequence Pathogens and Pathology, Centers for Disease Control and Prevention, Atlanta, GA, USA; [9]Enteric Diseases Laboratory Branch, Division of Foodborne, Waterborne, and Environmental Diseases, Centers for Disease Control and Prevention, Atlanta, GA, USA; [10]Center for Food Safety, University of Georgia, Griffin, GA, USA.
***Correspondence:** Boas C.L. van der Putten, boas.vanderputten@amsterdamumc.nl

Keywords: software testing; continuous integration; computational biology.

Abbreviations: ASM NGS, American Society for Microbiology Next Generation Sequencing; CI, continous integration; CSIS, code safety inspection service.

†These authors contributed equally to this work

Data statement: All supporting data, code and protocols have been provided within the article or through supplementary data files. One supplementary table and one supplementary figure are available with the online version of this article.

000790

Impact Statement

In the field of microbial bioinformatics, good software engineering practises are not yet widely adopted. Many microbial bioinformaticians start out as (micro)biologists and subsequently learn how to code. Without abundant formal training, a lot of education about good software engineering practices comes down to an exchange of information within the microbial bioinformatics community. This paper serves as a resource that could help microbial bioinformaticians get started with software testing if they have not had formal training.

2020, demonstrating that the microbial bioinformatics community can contribute to software sustainability using a collaborative platform (Table S1, available in the online version of this article). From this experience, we would like to propose collaborative software testing as an opportunity to continuously engage software users, developers, and students to unify scientific work across domains. We have outlined the following recommendations for ensuring software sustainability through testing and offer a repository of automated test knowledge and examples at the Code Safety Inspection Service (CSIS) repository on GitHub (https://github.com/microbinfie-hackathon2020/CSIS).

RECOMMENDATIONS

Based on our experiences from the ASM NGS 2020 hackathon, we developed seven recommendations that can be followed during software development.

Establish software needs and testing goals

Manually testing the functionality of a tool is feasible in early development but can become laborious as the software matures. Developers may establish software needs and testing goals during the planning and designing stages to ensure an efficient testing structure. Table 1

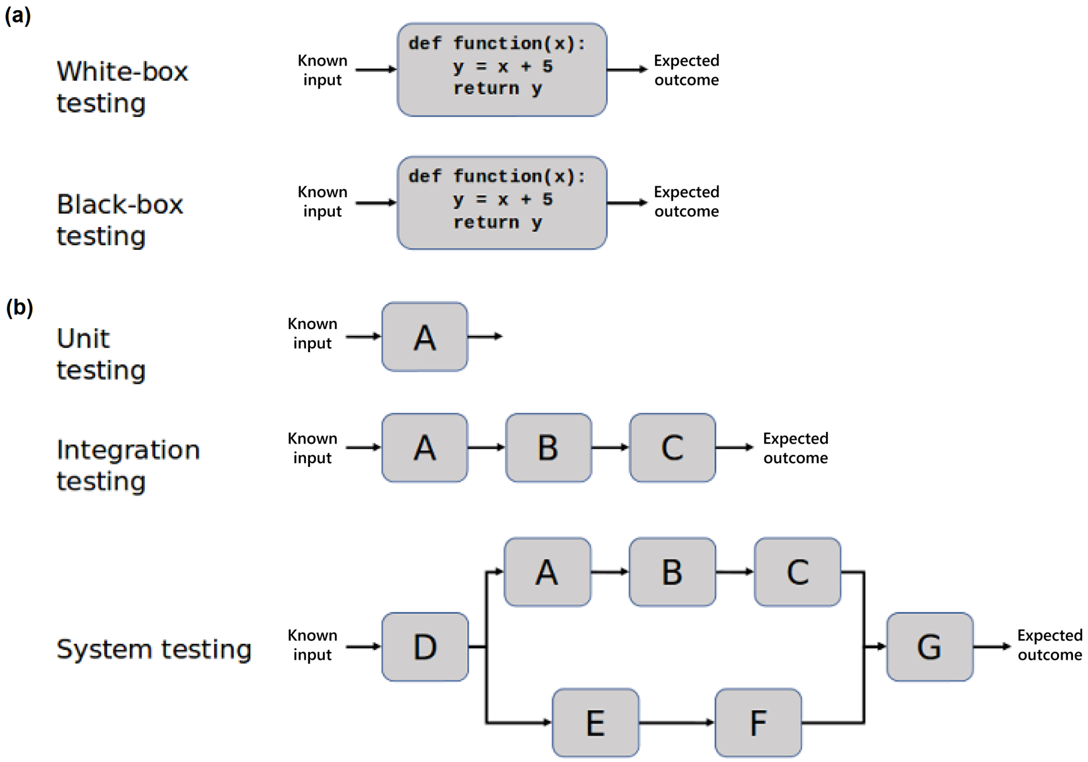


Fig. 1. Testing strategies. (a) White-box vs. black-box testing. In white-box testing, the tester knows the underlying code and structure of the software, where the tester does not know this in black-box testing. Note that this distinction is not strictly dichotomous and is considered less useful nowadays (b) Unit vs. integration vs. system testing. When software comprises several modules, it is possible to test each single module (unit testing), groups of related modules (integration testing) or all modules (system testing). Note that the terms white-box testing and unit testing are sometimes used interchangeably but relate to different concepts.

Table 1. Overview of testing approaches. Software testing can be separated into three types: installation, functionality and destructive. Each component is described, followed by an example on a real-life application on *Software X*, a hypothetical nucleotide sequence annotation tool

Name	Description	Example
	Installation testing: can the software be invoked on different setups?	
Installation testing	Can the software be installed on different platforms?	*Test whether Software X can be installed using apt-get, pip, conda and from source.*
Configuration testing	With which dependencies can the software be used?	*Test whether Software X can be used with different versions of* BLAST+.
Implementation testing	Do different implementations work similarly enough?	*Test whether Software X works the same between the standalone and webserver versions.*
Compatibility testing	Are newer versions compatible with previous input/output?	*Test whether Software X can be used with older versions of the UniProtKB database.*
Static testing	Is the source code syntactically correct?	*Check whether all opening braces have corresponding closing braces or whether code is indented correctly in Software X.*
	Standard functionality testing: does the software do what it should in daily use?	
Use case testing	Can the software do what it is supposed to do regularly?	*Test whether Software X can annotate different FASTA files: with spaces in the header, without a header, an empty file, with spaces in the sequence, with unknown characters in the sequences, et cetera.*
Workflow testing	Can the software successfully traverse each path in the analysis?	*Test whether Software X works in different modes (using fast mode or using one dependency over the other).*
Sanity testing	Can the software be invoked without errors?	*Test whether Software X works correctly without flags, or when checking dependencies or displaying help info.*
	Destructive testing: what makes the software fail?	
Mutation testing	How do the current tests handle harmful alterations to the software?	*Test whether changing a single addition to a subtraction within Software X causes the test suite to fail.*
Load testing	At what input size does the software fail?	*Test whether Software X can annotate a small plasmid (10 kbp), a medium-size genome (2 Mbp) or an unrealistically large genome for a prokaryote (1 Gbp).*
Fault injection	Does the software fail if faults are introduced and how is this handled?	*Test whether Software X fails if nonsense functions are introduced in the gene calling code.*

Gbp, Giga-base-pair; kbp, kilo-base-pair; Mbp, Mega-base-pair.

provides an overview of testing methodologies and can serve as a guide to developers that aim to implement testing practises. A minimal test set could address the validation of core components or the programme as a whole (system testing) and gradually progress toward verification of key functions which can accommodate code changes over time (unit testing, Fig. 1). Ideally, testing should be implemented from the early stages of software development (test-driven development). Defining the scope of testing is important before developing tests. For pipeline development, testing of each individual component can be laborious and can be expedited if those components already implement testing of their own. Testing of the pipeline itself should take priority.

Input test files: the good, the bad, and the ugly

When testing, it is important to include test files with known expected outcomes for a successful run. However, it is equally important to include files or other inputs on which the tool is expected to fail. For example, some tools should recognize and report an empty input file or a wrong input format. Therefore, the test dataset should be small enough to be easily deployed (see recommendation #4) but as large as necessary to cover all intended test cases. Data provenance should be disclosed, either if it's from real data or originated *in silico*. Typically, a small test data is packaged with the software. Examples of valid and invalid file formats are available through the BioJulia project (https://github.com/BioJulia/BioFmtSpecimens). The nf-core project (https://nf-co.re/) provides a repository with test data for a myriad of cases (https://github.com/nf-core/test-datasets).

Use an established framework to implement testing

Understanding the test workflow can not only ensure continued software development but also the integrity of the project for developers and users. Testing frameworks improve test development and efficiency. Examples include unittest (https://docs.python.org/3/library/unittest.html) or pytest (https://docs.pytest.org/en/stable/) for Python, and testthat (https://testthat.r-lib.org/) for R, testing interfaces such as TAP (http://testanything.org/), or built-in test attributes such as in Rust. Although many tests can be implemented using a combination of frameworks, personal preferences (e.g. amount of boilerplate code required) might drive your choice. Additionally, in Github Actions the formulas of each test block can be explicitly stated using the standardised and easy-to-follow YAML (https://yaml.org/, Fig. S1, available in the online version of this article), already adopted by most continuous integration platforms (recommendation #4). For containerised software, testing considerations differ slightly and have been covered previously by Gruening *et al.* (2019) [6].

Testing is good, automated testing is better

When designing tests, planning for automation saves development time. Whether your tests are small or comprehensive, automatic triggering of tests will help reduce your workload. Many platforms trigger tests automatically based on a set of user-defined conditions. Platforms such as GitHub Actions (https://github.com/features/actions) and GitLab CI (https://about.gitlab.com/stages-devops-lifecycle/continuous-integration) offer straightforward automated testing of code seamlessly upon deployment. A typical workflow, consisting of a minimal testing framework (see recommendation #1 and #3) and a small test dataset (see recommendation #2), can then be directly integrated within your project hosted on a version control system, such as GitHub (https://github.com/), and directly integrated with a continuous integration provider, such as GitHub Actions in GitHub. Testing considerations for containerised software has been covered previously by Gruening *et al.* (2019) [6].

Ensure portability by testing on several platforms

The result of an automated test in the context of one computational workspace does not ensure the same result will be obtained in a different setup. It is important to ensure your software can be installed and used across supported platforms. One way to ensure this is to test on different environments, with varying dependency versions (e.g. multiple Python versions, instead of only the most recent one). Developers can gain increased benefits of testing if tests are run on different setups automatically (see recommendation #4 and Fig. S1).

Showcase the tests

For prospective users, it is good to know whether you have tested your software and, if so, which tests you have included. This can be done by displaying a badge [7] (see https://github.com/microbinfie-hackathon2020/CSIS/blob/main/README.md#example-software-testing), or linking to your defined testing strategy e.g. a GitHub Actions YAML, (see recommendation #2, Fig. S1). Documenting the testing goal and process enables end-users to easily check tool functionality and the level of testing [8].

It may be helpful to contact the authors, directly or through issues in the code repository, whose software you have tested to share successful outcomes or if you encountered abnormal behaviour or component failures. An external perspective can be useful to find bugs that the authors are unaware of. A set of issue templates for various situations is available in the CSIS repository on GitHub (https://github.com/microbinfie-hackathon2020/CSIS/tree/main/templates).

Encourage others to test your software

Software testing can be crowdsourced, as showcased by the ASM NGS 2020 hackathon. Software suites such as Pangolin (https://github.com/cov-lineages/pangolin) [9] and chewBBACA (https://github.com/B-UMMI/chewBBACA) [10] have implemented automated testing developed during the hackathon. For developers, crowdsourcing offers the benefits of fresh eyes on your software. Feedback and contributions from users can expedite the implementation of software testing practices. It also contributes to software sustainability by creating community buy-in, which ultimately helps the software maintainers keep pace with dependency changes and identify current user needs.

CONCLUSIONS

Testing is a critical aspect of scientific software development, but automated software testing remains underused in scientific software. In this hackathon, we demonstrated the usefulness of testing and developed a set of recommendations that should improve the development of tests. We also demonstrated the feasibility of producing test suites for already-established microbial bioinformatics software (Table S1).

Funding information
C.I.M. was supported by the Fundação para a Ciência e Tecnologia (grant SFRH/BD/129483/2017). L.P.C. was partially supported by Shanghai Municipal Science and Technology Major Project (2018SHZDZX01) and ZJLab. R. M. was supported by the Fundação para a Ciência e Tecnologia (grant 2020.08493.BD).

Acknowledgements
The findings and conclusions in this report are those of the authors and do not necessarily represent the official position of the Centers for Disease Control and Prevention (CDC). The mention of company names or products does not constitute an endorsement by the CDC.

Author contributions
In addition to the authors, the following participants were responsible for automating tests for bioinformatic tools and contributing a community resource for identifying software that can pass unit tests, available at https://github.com/microbinfie-hackathon2020/CSIS. Participants are listed alphabetically: Áine O'Toole, Amit Yadav, Justin Payne, Mario Ramirez, Peter van Heusden, Robert A. Petit III, Verity Hill, Yvette Unoarumhi.

Conflicts of interest
The authors declare that there are no conflicts of interest.

References

1. Stodden V, Seiler J, Ma Z. An empirical analysis of journal policy effectiveness for computational reproducibility. *Proc Natl Acad Sci U S A* 2018;115:2584–2589.
2. Krafczyk M, Shi A, Bhaskar A, Marinov D, Stodden V. Scientific tests and continuous integration strategies to enhance reproducibility in the scientific software context. In: *Proceedings of the 2nd International Workshop on Practical Reproducible Evaluation of Computer Systems*. Phoenix, AZ, USA: Association for Computing Machinery, 2019. pp. 23–28.
3. Chang G, Roth CB, Reyes CL, Pornillos O, Chen Y-J, *et al*. Retraction. *Science* 2006;314:1875.
4. Hall BG, Salipante SJ. Retraction: Measures of clade confidence do not correlate with accuracy of phylogenetic trees. *PLoS Comput Biol* 2007;3:e158.
5. Busby B, Lesko M, August 2015 and January 2016 Hackathon participants, Federer L. Closing gaps between open software and public data in a hackathon setting: User-centered software prototyping. *F1000Res* 2016;5:672.
6. Gruening B, Sallou O, Moreno P, da Veiga Leprevost F, Ménager H, *et al*. Recommendations for the packaging and containerizing of bioinformatics software. *F1000Res* 2018;7:742.
7. Trockman A, Zhou S, Kästner C, Vasilescu B. Adding sparkle to social coding: an empirical study of repository badges in the npm ecosystem. In: *Proceedings of the 40th International Conference on Software Engineering*. Gothenburg, Sweden: Association for Computing Machinery, 2018. pp. 511–522.
8. Karimzadeh M, Hoffman MM. Top considerations for creating bioinformatics software documentation. *Brief Bioinform* 2018;19:693–699.
9. O'Toole Á, Scher E, Underwood A, Jackson B, Hill V, *et al*. Assignment of epidemiological lineages in an emerging pandemic using the pangolin tool. *Virus Evol* 2021;7:veab064.
10. Silva M, Machado MP, Silva DN, Rossi M, Moran-Gilad J, *et al*. chewBBACA: A complete suite for gene-by-gene schema creation and strain identification. *Microb Genom* 2018;4.