UNIVERSIDADE DE LISBOA

FACULDADE DE CIÊNCIAS

DEPARTAMENTO DE INFORMÁTICA

Automatic Detection of Anomalous User Access Patterns to Sensitive Data

Mariana Galhardas Pina

Mestrado em Informática

Dissertação orientada por:
Professor Doutor Pedro M. Ferreira e Engenheiro José António dos Santos Alegria

2019

2019

Acknowledgements

I would like to thank DCY department at Altice for receiving me with open arms and for all the support given. Special thank you to Eng. José Alegria for the opportunity and guidance throughout the project. Another special thank you to Ricardo Ramalho for the guidance, help and motivation as a team leader in which I was integrated. Thank you to Professor Pedro Ferreira for accepting to be my advisor, for the time spent and all the guidance throughout this project.

Finally, thank you to my friends and family.

To my friends and family.

Resumo

Esta dissertação foi realizada por uma aluna da Faculdade de Ciências da Universidade de Lisboa, com licenciatura em Matemática Aplicada e atualmente a frequentar o mestrado em Informática. A proposta desta dissertação veio do departamento de Cibersegurança (DCY) da Altice Portugal (MEO) e a área de especialização é aprendizagem máquina (*machine learning*).

Nos últimos anos, especialmente em grandes organizações, o roubo de informação confidencial tem vindo a ser uma problemática cada vez maior. Este tipo de ataque tem, normalmente, duas origens distintas: colaboradores maliciosos ou malware instalado, possivelmente proveniente de um ataque de *phishing*. No entanto, atividade anónima sem intenção maliciosa também pode ser relevante, pois pode ser um indicador de um uso incorreto de recursos da rede ou de uma violação de política.

Este trabalho aborda este problema de segurança através da aplicação de técnicas de aprendizagem máquina com o objetivo de detetar anomalias, correspondentes a atividades ilícitas, no registo de acessos a dados de informações de clientes e/ou meta-dados feitos por utilizadores de *backoffice,*. Um dos objetivos é a distinção dessas anomalias, mais concretamente, a classificação dessas situações de roubo de informação confidencial em diferentes tipos, para que as pessoas responsáveis pela parte posterior da investigação interna saibam o que devem procurar. Para além disso, procuramos reduzir ao máximo o número de falsos positivos, mantendo um grau de deteção elevado.

Anteriormente, a empresa realizou um projeto com o mesmo objetivo final, no entanto, com uma metodologia completamente distinta. Nesse projeto foram aplicados métodos de estatística descritiva e heurísticas simples para a deteção de anomalias, tendo sido intitulou de *Cuscos*. O projeto *Cuscos* detetou um número bastante elevado de anomalias (1800), contudo identificou-se um número muito alto de possíveis falsos positivos, tendo sido uma problemática. Adicionalmente, a impossibilidade de distinguir os diferentes tipos de atividade ilícita, constituiu um obstáculo, tendo, assim, cada anomalia que ser estudada individualmente para que se descobrisse a sua causa. Como se pode ver pelos objetivos acima descritos, este projeto procura solucionar estas dificuldades.

Primeiro, fez-se uma caracterização estatística dos dados, onde se decidiu que características (*features)* dos dados originais deviam ser criadas ou extraídas para a construção de conjuntos de dados (*datasets*) e, posteriormente, para a aplicação dos algoritmos de aprendizagem máquina (*machine learning*) escolhidos. Nesta, foram consideradas duas abordagens: uma em ordem aos utilizadores e outra direcionada aos números de telefone. Como tal, foram criados dois conjuntos de dados, um para cada abordagem. De seguida, executaram-se os procedimentos necessários de pré-processamento e normalização dos dados. Finalmente, foram aplicados algoritmos de agrupamento de dados e deteção de anomalias aos conjuntos de dados criados. Os algoritmos de agrupamento de dados considerados foram: *k-means*, *DBSCAN* e *affinity propagation*; e os algoritmos de deteção de anomalias escolhidos foram: *elliptic envelope* e *isolation forest*. Para determinar os parâmetros adequados de cada um desses algoritmos, foram definidos intervalos de parâmetros e criadas tabelas de pontuação com os resultados obtidos a partir da aplicação desses algoritmos com as diferentes combinações de parâmetros. Para obter resultados específicos para diferentes perspetivas analíticas, além de serem aplicados em todo o conjunto de dados construído, os algoritmos também foram aplicados a diferentes combinações de algumas de suas características.

Tendo em conta que as anomalias finais são referentes a utilizadores, os resultados da abordagem dos números de telefone tiveram de ser convertidos, isto é, os utilizadores que acederam aos números de telefone considerados como anomalias pela aplicação dos algoritmos substituíram os números de telefone, sendo, assim, os utilizadores as anomalias consideradas. Em cada abordagem, foi escolhido um método de *ensemble* para decidir quais dos utilizadores detetados seriam considerados anomalias finais. Finalmente, obtiveram-se os resultados finais através de um *ensemble* por união do conjunto de resultados de ambas as abordagens e, posteriormente, criaram-se regras de decisão para classificar as anomalias em diferentes categorias.

Os resultados finais cumpriram todos os objetivos: detetaram-se anomalias relevantes de situações correspondentes a acessos ilícitos, reduziu-se o número de falsos positivos e cada anomalia detetada está classificada consoante o tipo de comportamento que representa.

Palavras-chave: aprendizagem automática, cibersegurança, roubo de informação, deteção de anomalias

Abstract

In recent years, especially in large organizations, the theft of valuable information has increasingly become a major problem.

This project focuses on users access to information related to customer telephone numbers inside a telecom company. The objective is to, through machine learning techniques, detect illicit accesses to this information, focusing on those likely to match information theft actions.

First, we made a statistical characterization of the data. Decided which features should be created or extracted from data to build the necessary datasets (two different approaches) to apply the algorithms, and then the required pre-processing and normalization procedures were executed. Finally, we applied clustering and anomaly detection algorithms to detect anomalies in the datasets. The algorithms considered were: k-means, DBSCAN, affinity propagation clustering methods, elliptic envelope and isolation forest anomaly detection methods. To determine optimal parameters for the algorithms on this data, parameter ranges were defined and score tables were created with the results obtained from different combinations of parameters. To obtain specific results for different analytic perspectives, besides being applied on the entire datasets built, the algorithms were also applied to different combinations of some of their features. Finally, after the algorithms application, ensemble methods were chosen and decision rules were created to classify the anomalies in different categories.

The final results met all objectives. Relevant anomalies were detected in situations corresponding to illicit accesses, the number of false positives was reduced and each detected anomaly is classified according to the type of behavior it represents.

Keywords: machine learning, cybersecurity, information theft, anomaly detection

Table of Contents

Chapter 1 Introduction 1

1.1 Motivation 1

1.2 Objectives 2

1.3 Document Structure 2

Chapter 2 Related Work 3

2.1 Previous Studies 3

2.2 Fundaments and Tools 5

2.2.1 Jupyter 5

2.2.2 Python 5

2.2.3 Machine Learning 5

2.2.4 Clustering Algorithms 6

2.2.5 Anomaly detection algorithms 7

2.2.6 Evaluation Metrics 7

2.2.7 HIDRA 9

2.2.8 Docker 10

Chapter 3 Methodology 11

3.1 Data Study 13

3.1.1 Characterization of APPS-GPD dataset 15

3.2 Pre-processing 23

3.2.1 User Approach 24

3.2.2 Telephones approach 26

3.2.3 Normalization 28

3.3 Anomalies 28

3.4 Algorithms 30

3.4.1 K-means 30

3.4.2 DBSCAN 31

3.4.3 Affinity Propagation 32

3.4.4 Elliptic Envelope 32
3.4.5 Isolation Forest 33
3.5 Parameterization 34
3.6 Ensemble 39
3.7 Classification 40
3.8 Process Automation 41
Chapter 4 Results 43
4.1 User Approach 43
4.2 Telephone Approach 53
4.3 Final Results 60
Chapter 5 Conclusions and Future Work 67
References 69
Appendix A 72
Appendix B – Users approach: Score Tables for perspective Ratios 73
Appendix C – Users approach: Score Tables for perspective AccessDays 77
Appendix D – Users approach: Score Tables for perspective AccessTel 81
Appendix E – Users approach: Score Tables for perspective TelDays 85
Appendix F – Users approach: Score Tables for perspective All 89
Appendix G – Telephones approach: Score Tables for perspective Ratios 93
Appendix H – Telephones approach: Score Tables for perspective AccessDay 95
Appendix I – Telephones approach: Score Tables for perspective AccessActor 97
Appendix J – Telephones approach: Score Tables for perspective DayActors 99
Appendix K – Telephones approach: Score Tables for perspective All 101
Appendix L – Final Results 103

List of Figures

Figure 1- Project methodology 12

Figure 2- Distribution of the number of times each actor made an access 15

Figure 3- Distribution of the number of times each actor made an access without extreme cases 16

Figure 4 - Distribution of the number of times each actor, which made less than 1000 accesses, made an access 16

Figure 5- Distribution of the number of different days each actor made an access 17

Figure 6- Distribution of the number of times each user, that made more than 10000 accesses, made an access 18

Figure 7- Distribution of the number of times each user, which made between 1000 and 10000 accesses, made an access 18

Figure 8-Distribution of the number of times each user, which made less than 1000 accesses, made an access 19

Figure 9-Distribution of the number of different days each user made an access .. 19

Figure 10-Distribution of the number of times each telephone number, which was accessed more than 1000 times, was accessed 20

Figure 11-Distribution of the number of times each telephone number, which was accessed between 10 and 1000 times, was accessed 21

Figure 12-Distribution of the number of times each telephone number, which was accessed less than 1000 times, was accessed 21

Figure 13-Distribution of the number of different days each telephone number was accessed 22

Figure 14-Distribution of the number of times each application was used to access telephone numbers 22

Figure 15-Distribution of the number of different days each application was used to access telephone numbers 23

Figure 16- Explanation of score tables created for the parameterization optimization 35

Figure 17- Automation process architecture 41

Figure 18-Anomalies detected by Elliptic Envelope 44

Figure 19-Anomalies detected by Affinity propagation 44

Figure 20-Anomalies detected by K-means 44
Figure 21- Anomalies detected by DBSCAN 44
Figure 22-Anomalies detected by Isolation Forest 44
Figure 23-Anomalies detected by Elliptic Envelope 45
Figure 24-Anomalies detected by DBSCAN 45
Figure 25-Anomalies detected by Affinity Propagation 46
Figure 26-Anomalies detected by Isolation Forest 46
Figure 27-Anomalies detected by K-means 46
Figure 28-Anomalies detected by Isolation Forest 47
Figure 29-Anomalies detected by K-means 47
Figure 30-Anomalies detected by DBSCAN 47
Figure 31-Anomalies detected by Elliptic Envelope 47
Figure 32-Anomalies detected by K-means 48
Figure 33-Anomalies detected by DBSCAN 48
Figure 34-Anomalies detected by Elliptic Envelope 48
Figure 35-Anomalies detected by Affinity Propagation 48
Figure 36-Anomalies detected by Isolation Forest 48
Figure 37-Anomalies detected by K-means 50
Figure 38-Anomalies detected by Affinity Propagation 49
Figure 39-Anomalies detected by DBSCAN 49
Figure 40-Anomalies detected by Elliptic Envelope 50
Figure 41-Anomalies detected by Isolation Forest 50
Figure 42-Anomalies detected by K-means 54
Figure 43-Anomalies detected by Elliptic Envelope 54
Figure 44-Anomalies detected by Isolation Forest 54
Figure 45-Anomalies detected by Isolation Forest 55
Figure 46-Anomalies detected by K-means 55
Figure 47-Anomalies detected by Elliptic Envelope 55
Figure 48-Anomalies detected by K-means 56

Figure 49-Anomalies detected by Isolation Forest 56
Figure 50-Anomalies detected by Elliptic Envelope 56
Figure 51-Anomalies detected by Elliptic Envelope 57
Figure 52-Anomalies detected by Isolation Forest 57
Figure 53-Anomalies detected by K-means..... 57
Figure 54-Anomalies detected by Elliptic Envelope 58
Figure 55-Anomalies detected by Isolation Forest 58
Figure 56-Anomalies detected by K-means..... 58
Figure 57-Final anomalies detected viewed in AccessTel perspective 62
Figure 58-Final anomalies detected viewed in AccessDay perspective 62
Figure 59-Final anomalies detected viewed in TelDays perspective..... 62
Figure 60-Final anomalies detected viewed in All perspective 62
Figure 61-Anomalies detected in Cuscos project viewed in AccessDays perspective 63
Figure 62-Anomalies detected in Cuscos project viewed in AccessTel perspective 63
Figure 63-Anomalies detected in Cuscos project viewed in All perspective 63
Figure 64-Anomalies detected in Cuscos project viewed in TelDays perspective. 63

List of Tables

Table 1- APPS-RGPD features explained 13

Table 2-Features of the dataset created for users approach 25

Table 3- Different perspectives to study in users approach 26

Table 4--Features of the dataset created for telephones approach 27

Table 5- Different perspectives to study in telephones approach 28

Table 6- Labels created to classify anomalies 29

Table 7- Parameterization range tested 37

Table 8- Rules followed to distinguish the anomaly labels created 40

Table 9-Users approach parameterization results 43

Table 10-Comparison of results of the different algorithms in AccessDay perspective 45

Table 11-Comparison of the results of the different algorithms in AccessTel perspective 46

Table 12-Comparison of the results of the different algorithms in TelDay perspective 47

Table 13-Comparison of the results of the different algorithms in Ratios perspective 49

Table 14-Comparison of the results of the different algorithms in All perspective 50

Table 15-Number of anomalies obtained for each algorithm applied to each perspective in users approach 51

Table 16-Comparison of results of the different ensemble methods in users approach 52

Table 17- Anomaly types detected in each ensemble method, in users approach .. 52

Table 18- Comparison of different algorithms detection of final anomalies in users approach 52

Table 19-Telephones approach parameterization results 53

Table 20-Comparison of the results of the different algorithms in AccessDay perspective 54

Table 21-Comparison of the results of the different algorithms in AccessActor perspective 55

Table 22-Comparison of the results of the different algorithms in DayActor perspective 56

Table 23-Comparison of the results of the different algorithms in Ratios perspective 57

Table 24-Comparison of the results of the different algorithms in All perspective 58

Table 25-Number of anomalies obtained for each algorithm applied to each perspective in telephones approach 59

Table 26-Comparison of results of the different ensemble methods in telephones approach..... 59

Table 27-Anomaly types detected in each ensemble method, in telephones approach 60

Table 28-Comparison of different algorithms detection of final anomalies in telephones approach 60

Table 29-Final anomaly types detected 61

Table 30-Total anomalies obtained..... 61

Table 31-Percentage coverage of confirmed anomalies 61

Table 32- Example of 4 detected identities in final results..... 64

Table 33- Anomaly description..... 64

Table 34- Anomaly description..... 64

Table 35- Anomaly description..... 65

Table 36- Anomaly description..... 65

Table 37- Legend for abbreviations used..... 72

Table 38- Score table for Isolation Forest..... 73

Table 39- Score table for K-means 74

Table 40- Score table for Elliptic Envelope..... 74

Table 41-Score Table for Affinity Propagation 75

Table 42- Score Table for DBSCAN 76

Table 43-Score Table for Isolation Forest 77

Table 44- Score Table for K-means..... 78

Table 45- Score Table for Elliptic Envelope 78

Table 46-Score Table for Affinity Propagation 79

Table 47-Score Table for DBSCAN 80

Table 48-Score Table for Isolation Forest 81

Table 49- Score Table for K-means 82

Table 50- Score Table for Elliptic Envelope 82

Table 51- Score Table for Affinity Propagation 83

Table 52-Score Table for DBSCAN 84

Table 53- Score Table for Isolation Forest 85

Table 54- Score Table for TelDays 86

Table 55- Score Table for Elliptic Envelope 86

Table 56- Score Table for Affinity Propagation 87

Table 57- Score Table for DBSCAN 88

Table 58- Score Table for Isolation Forest 89

Table 59- Score Table for K-means 90

Table 60- Score Table for Elliptic Envelope 90

Table 61- Score Table for Affinity Propagation 91

Table 62- Score Table for DBSCAN 92

Table 63- Table Score for Isolation Forest 93

Table 64-Score Table for K-means (silhouette score and VRC did not run 10 times when number of clusters was 2 and 3 because of situation explained before in which no anomalies were detected) 94

Table 65- Score Table for Elliptic Envelope 94

Table 66- Score Table for Isolation Forest 95

Table 67- Score Table for K-means 96

Table 68- Score Table for Elliptic Envelope 96

Table 69-Score Table for Isolation Forest 97

Table 70- Score Table for K-means (silhouette score and VRC did not run 10 times when number of clusters was 2, 3 and 4 because of situation explained before in which no anomalies were detected) 98

Table 71- Score Table for Elliptic Envelope 98

Table 72- Score Table for Isolation Forest 99
Table 73- Score Table for K-means 100
Table 74- Score Table for Elliptic Envelope 100
Table 75- Score Table for K-means 102
Table 76- Score Table for Elliptic Envelope 102
Table 77- Table with final Results 103
Table 78- Contribution of each algorithm for the detection of each anomaly 114

Chapter 1 Introduction

This dissertation was elaborated by Mariana Galhardas Pina, a student from Faculdade de Ciências of Universidade de Lisboa with a bachelor degree in Applied Mathematics and currently in a master's degree in Informatics. The proposal for this dissertation came from the cybersecurity (DCY) department of Altice Portugal (MEO), so the dissertation was carried out at the DCY department premises. The specialization area is machine learning and the problem being worked is towards anomaly detection.

1.1 Motivation

The theft of valuable information is a major problem in companies nowadays. This type of attack can usually have two distinct origins: the discovery and access to valuable information by malicious collaborators; or malware installed inside the network, originated from a phishing attack or hackers, for example. However, anomalous activities with no malicious intent may be relevant, because they may be an indicator of incorrect use of network resources or a security policy violation. It is in this kind of security problem that this project focus. The goal is to find anomalies in the log of queries made by users, to detect cases of the situations mentioned previously, and therefore contribute to the security of the company.

The use of machine learning in big data contexts is growing in most business areas, as it can help to find patterns, predicting results, reaching decisions, and find easier ways to get to these results. Nowadays, working with big data happens in most technology and online companies, and interpreting this data to reach conclusions is a very common problem. Machine learning provides algorithms that can facilitate this analysis: clustering and classification tasks, which along with anomaly detection techniques can contribute to effectively find relevant outliers, which are segments of data that do not follow the most common patterns. Anomalies in data can have different root causes, depending on the dataset. Regardless if they are good or bad, it can be useful to find those cases to eliminate, control, or even, augment them. In the specific application considered in this work, the goal is to uncover illicit data access activities and prevent harmful consequences for the company.

1.2 Objectives

The main objective is to employ machine learning techniques to implement a methodology to detect illicit data access activities that are likely to match information theft actions, detecting the less false positives possible. In addition to detecting the relevant anomalies, we want to be able to discriminate between different situations of illicit behavior. This will improve the company awareness on this kind of situations that might be occurring, as it will unveil the users and the type of illicit action they are undertaking. With this information, the company can then perform an internal investigation and pursue the necessary disciplinary or legal actions.

1.3 Document Structure

The document is organized by chapters, each one of them containing topics. Here it will be, briefly, explained each chapter.

Chapter 2: The second chapter is the related work, it will present the relevant theory in the area of this dissertation. It contains the subjects needed to be known to understand this project. It explains the origin of this project and which others methods have been applied to solve this kind of problem.

Chapter 3: The third chapter is the methodology and explains all the steps executed to reach the results.

Chapter 4: This chapter explains the results. In this chapter, all the results, based on the methodology used, are shown and the final decisions are discussed.

Chapter 5: The fifth chapter contains the conclusions and possible future work to investigate this kind of problem.

Chapter 6: The last chapter contains references to papers, links or books consulted to get the needed information.

Chapter 2 Related Work

2.1 Previous Studies

The company had a previous project, named *Cuscos*, using the same data with the same purpose. As the first approach to the same problem, *Cuscos* used simple descriptive statistical methods and simple heuristics to detect anomalies. In summary, from the original data, a dataset was created indexed by <user, telephone number> pairs. The features created characterized which users accessed the data of which telephone numbers. The median of each feature of the dataset was calculated, and with that median, for each <user, telephone number> pair features, the standard deviation was calculated. Then a threshold was chosen, and the pairs that had a standard deviation above the threshold were considered anomalies. This project captured a large number of anomalies (1800 anomalies) but had the problem of dealing with a potentially large number of possible false positives. With such amount of anomalies, it was never possible to investigate all of them. Another problem of the *Cuscos* methodology was the fact that it was not possible to discriminate different potential situations of illicit behavior, each anomaly had to be studied to discover why it was considered an anomaly. In this approach to the same problem instead of a descriptive statistical method, various machine learning algorithms will be used in various perspectives of the data to obtain more reliable anomalies. A process of classification of the anomalies will also be executed so that each anomaly obtained is already labeled as a specific type.

The paper "A survey of network anomaly detection techniques" [5] includes anomaly detection through classification, statistical methods, and clustering. The classification algorithms were used in an unsupervised environment. The algorithms used were support vector machine, bayesian networks, and neural networks. For the statistical methods, a distance measure based on the chi-square test statistic is developed and principal component analysis (PCA) is also used to detect anomalies. Finally, the clustering algorithms application seemed very efficient. Nearest neighbor, K-means and variations of k-means were used. The results after clustering are a division of the data based on similarity. Looking at the clusters dimension and distances it is possible to conclude that the ones with the biggest dimension are representative of the most common behavior, while the smallest clusters and farther away must be the samples that distance themselves from the pattern the most. This project will implement clustering algorithms

and it will also add specific anomaly detection algorithms to work together to detect anomalies.

The anomaly detection algorithms were brought to the table through the paper "Smart Audio Sensors in the Internet of Things Edge for Anomaly Detection" [12], it proposes a design framework for smart audio sensors able to record and pre-process raw audio streams, before wirelessly transmitting the computed audio features to a modular IoT gateway. Both Elliptic Envelope and Isolation Forest were deployed on an affordable IoT gateway to detect anomalous sound events happening in an office environment. Such good results in detecting anomalies to a different problem but with the same purpose (anomaly detection) brought up an interest in the way these algorithms would behave in this situation.

After deciding to use clustering algorithms it was necessary to determine the most appropriate ones. A promising algorithm, Affinity Propagation, is described in "Clustering by Passing Messages between Data Points" [10]. In most clustering algorithms the number of clusters has to be previously chosen, but affinity propagation can determine the best number of clusters alone, the way this algorithm works will be explained in a forthcoming subsection.

An interesting methodology was seen in the paper "An Application of Machine Learning to Anomaly Detection" [15]. To learn the characteristic patterns of actions, a temporal sequence was created (an ordered, fixed-length set of temporally adjacent actions) for each user, as the fundamental unit of comparison for a user profile. The basic action of the detection system was to compare incoming input sequences to the historical data and form an opinion as to whether or not they both represent the same user. The fundamental unit of comparison in the anomaly detection system is the command sequence. To classify sequences of new actions as consistent or inconsistent with sequence history, two fixed-length sequences can be compared using a similarity measure. The system computes a numerical similarity measure that returns a high value for pairs of sequences that it believes to have a close resemblance and a low value to pairs of sequences that it believes largely differ. This methodology seems very interesting and it is a future approach to be taken into account, using temporal series to define a pattern and obtain which behaviors do not follow it.

2.2 Fundaments and Tools

2.2.1 Jupyter

Jupyter is an open-source web application that allows users to create and share code documents, create code and view results together. Due to this flexibility of programming and visualization of the results at the moment, it was the chosen medium to the development of the dissertation.

2.2.2 Python

The chosen language was python because it is a general purpose programming language, very common to use in machine learning and it has very complete and efficient libraries for this purpose.

2.2.3 Machine Learning

Machine learning is based on the idea that we can learn from data, identify patterns and make predictions, with minimal human intervention, which allows us to make efficient decisions for a specific set of data.

Supervised vs unsupervised learning

Supervised learning is when we have input features (usually called matrix X, with one column per feature, one row per record) and output labels (usually called vector Y). The idea is to initially fit the algorithm to the data and to the label each record gets based on its features. Then after the algorithm was adjusted to the data, it is used to predict labels for the data that was not used yet. Since the data is supervised, after predicting the labels, we can use supervised evaluation metrics, like confusion matrix and accuracy.

In unsupervised learning we do not have previous knowledge, so we have just the input features (X). The model is fitted to the data anyway, but instead of adapting the features to a label it just adapts the model to the data behavior.

- Clustering: Clustering refers to unsupervised learning, which means it can be used when dealing with unlabeled data (i.e., data without defined categories). The goal of this type of algorithm is to group (cluster) the data based on features similarity.
- Classification: Classification refers to supervised learning, so it needs *ground truth* knowledge, it is the process of predicting the class of given data. The

algorithm must, first, fit the labeled data to get the knowledge of how, based on the characteristics, it should label the future data.

- Anomaly Detection: There is no specific definition of anomaly, it depends on the interpretation and situation. It is a record on the data that does not fit the standard that is considered "normal". Detection of anomalies is a technique where we identify these "abnormal" patterns, which are not what would be expected. It can be seen as detection of outliers or detection of novelties. The difference between these two terms is the fact that in outliers detection the data is “polluted”, which means it contains outliers, it contains observations that are far from the others. In novelties detection the data is not “polluted” with outliers, we are interested in detecting whether a new observation is an outlier. Clustering can help to detect anomalies, by considering anomalies the samples associated with much smaller or more distant clusters.

2.2.4 Clustering Algorithms

K-means

K-means is an algorithm that clusters the data by starting with a random division in clusters (number of clusters depends on what the user decides), and then recursively calculates the distance of the points to the cluster mean to decide in which cluster each point should belong to. As seen in [5], it can obtain very good results for anomaly detection.

DBSCAN

Density Based Spatial Clustering of Applications with Noise (DBSCAN), shown a good option for anomaly detection in [11], is an algorithm that clusters based on density. It has two important parameters, the minimal distance between points and the minimal number of neighbors a point is required to have to be considered valid. Given a set of points in a space, it clusters together those that are closer to each other (have the minimum number of neighbors, chosen by the user, in its neighborhood). The points in lower density regions (that do not satisfy the minimum number of neighbors) will be excluded from any cluster and be considered noise points.

Affinity Propagation

Affinity Propagation, introduced for anomaly detection in [10], is an interesting algorithm to use since it calculates the number of clusters by itself. In Affinity Propagation, the data points are seen as a network where all the data points send messages

to all other points. The subject of these messages relates to the willingness of the points to be “exemplars”. Exemplars are the points that best explain the other cluster data points and are the most significant within their cluster. Each cluster has only one exemplar. All the data points want to collectively determine which is their exemplar.

2.2.5 Anomaly detection algorithms

Elliptic Envelope

Another interesting approach was to use specific anomaly detection algorithms. This algorithm assumes that the regular data came from a Gaussian distribution. From this assumption, it tries to define the structure or pattern of the data and can detect outliers as observations which stand far enough from the base structure. It fits an ellipse around the central points and ignores the ones outside, fitting the data only on the points considered “normal”. It then predicts the points not considered “normal” as the outliers. According to [12] could be a good contribution to the final results.

Isolation Forest

Another algorithm to detect outliers is the Isolation Forest. Also introduced by [12], it is based on the logic of the decision trees (decision trees ensemble). It will "isolate" a record and randomly choose a feature, and then, randomly again, choose a split value between the maximum and minimum value of that feature. Since an outlier is an out-of-normal record, its features should have more distant values than the ones of a normal record, so the split should occur closer to the roots of the trees, that is, the path to isolate that record is shorter than the path to isolate a normal record, you need fewer splits.

2.2.6 Evaluation Metrics

Evaluation metrics should be applied after the application of the algorithms, and a high score in them is needed to have security the results. Below the evaluation metrics considered are explained.

- Silhouette Score is a clustering evaluation metric. For each sample, it calculates the mean intra-cluster distance (a) and the mean nearest-cluster distance (b). The Silhouette Coefficient for a sample is (b - a) / max (a, b). It calculates this coefficient for all samples and then returns the mean of all coefficients. It varies between -1 and 1, being 1 the best value and -1 the worst. A value near 0 indicates overlapping clusters. A negative value,

generally, indicates that a sample has been assigned to the wrong cluster, as a different cluster is more similar.

- The Calinski and Harabasz [16] Variance Ratio Criterion (VRC) measures the degree of separation between clusters and homogeneity within them. The higher the VRC value, the better are the clustering algorithm results expected to be.

$$VRC = \frac{trace\ B/(k-1)}{trace\ W/(n-k)}$$

Where *n* and *k* are the total number of samples and the number of clusters in the partition, respectively; the *B* and *W* terms are the between-cluster and the within-cluster sums of squares (covariance) matrices.

Normalization

Like in most machine learning situations, since we are dealing with features with very different intervals, the data needs to be normalized.

- Standardization: Standardizes features by removing the mean and scaling to unit variance. The standard score of a sample is calculated by:

$$Z= (x-u)/s$$

Where *u* is the mean of the training samples and *s* is the standard deviation of the training samples;

- Regular Normalization: Scales input vectors individually to the unit norm;
- Robust Scaler: Scales the features through statistics that are robust to outliers. It removes the median and scales the data according to the quantile range. It uses the Interquartile Range (IQR). The IQR is the range between the 1st quartile (25th quantile) and the 3rd quartile (75th quantile).

Comparison between the benefits of Data Standardization vs Regular Normalization vs Robust Scaler:

Advantages:

1. Standardization: scales features such that the distribution is centered around 0, with a standard deviation of 1;

2. Regular Normalization: shrinks the range such that the range is now between 0 and 1 (or -1 to 1 if there are negative values);
3. Robust Scaler: similar to normalization but instead it uses the interquartile range to be robust to outliers.

Disadvantages:

1. Standardization: is not good if the data is not normally distributed (i.e. no Gaussian Distribution);
2. Normalization: gets heavily influenced by outliers (i.e. extreme values);
3. Robust Scaler: does not take the average into account and only focuses on the parts where the bulk data is.

2.2.7 HIDRA

High Performance Infrastructure for Data Research and Analysis is a highly performant, available and scalable datastore used for security analytics, mostly based on machine learning, visualization, and forensics, to support fraud investigation and security incident response, developed by the Cyber Security and Privacy Direction (DCY) of Altice Portugal (MEO). This platform is fed with ETL (Process of data extraction, transformation and loading) processes from various security related sources (including users accesses to clients telephone numbers information). Hydra's operation is based on the integration of 3 base technologies: *Elasticsearch*, *Kibana*, and *Logstach*.

Elasticsearch

Elasticsearch[1] is a search engine that allows access to high volume data. It is available for many languages, such as Java, Python, and Ruby, and allows several types of queries (aggregations, intervals, specific values or dates, patterns…) in an easy way, which makes it so popular nowadays. It contains a Representational State Transfer Application Programming Interface (REST API) which facilitates development. *Elasticsearch* is used in Hydra to efficiently search and filter event data stored by indexing information.

[1] https://www.elastic.co/

Kibana

Kibana[2] is an open-source platform that allows to explore, visualize and analyze data stored with *Elasticsearch*. It allows to browse, visualize and interact with the data present in *Elasticsearch* indexes. It is a visualization plugin that has the necessary resources to study the data through the elaboration of dashboards.

Logstash

Logstash is an open-source ETL engine. Each process is organized as a pipeline which consists of three phases: collection (input), transformation and enrichment (filter) and data forwarding (output). A pipeline can be configured to collect data from multiple sources and forward the processed data to multiple destinations. Its architecture is highly extensible because each phase has multiple plugins and that allows *Logstash* to work with multiple technologies. It is horizontally scalable by allowing multiple workers to distribute the processing load. The internal queue mechanism provides reliability and resilience in the occurrence of performance shortages or failures. In the context of this project, *Logstash* is used in HIDRA to collect telephone access events efficiently and reliably and store them in *Elasticsearch* in a clean and organized format for fast search and retrieval.

2.2.8 Docker

Docker is a software which facilitates the creation, deployment and execution of software using containers. The main advantages of its use are portability between different environments and ease of update because all software and its dependencies are packaged inside the container, and security because it runs in an isolated environment inside the operating system. In this project *Jupyter* is used inside a Docker container with the image "jupyter/scipy-notebook" [19] to easily reproduce the same results in another machine if necessary and to protect other processes in the same machine from the impact of high resources usage, typical of machine learning processes.

[2] https://www.elastic.co/products/kibana

Chapter 3 Methodology

The project methodology is organized in six stages, as illustrated in Figure 1. The first stage, with the pre-processing of the data, starts with the logs from *elasticsearch* that are transferred into a jupyter notebook and then all the necessary transformations are performed to create the datasets for both approaches. The second stage, with the prepared datasets, studies the different perspectives to look at the dataset that can contribute to better results. The third stage englobes the improvement and application of the algorithms. It receives the datasets in the various perspectives and returns the identities considered anomalies in each algorithm application. Finally, we ensemble the anomalies obtained. The sixth stage receives the anomalies obtained from the chosen ensemble method and classifies them into the different distinctions. Each of these steps will be explained in detail in the sub-sections below.

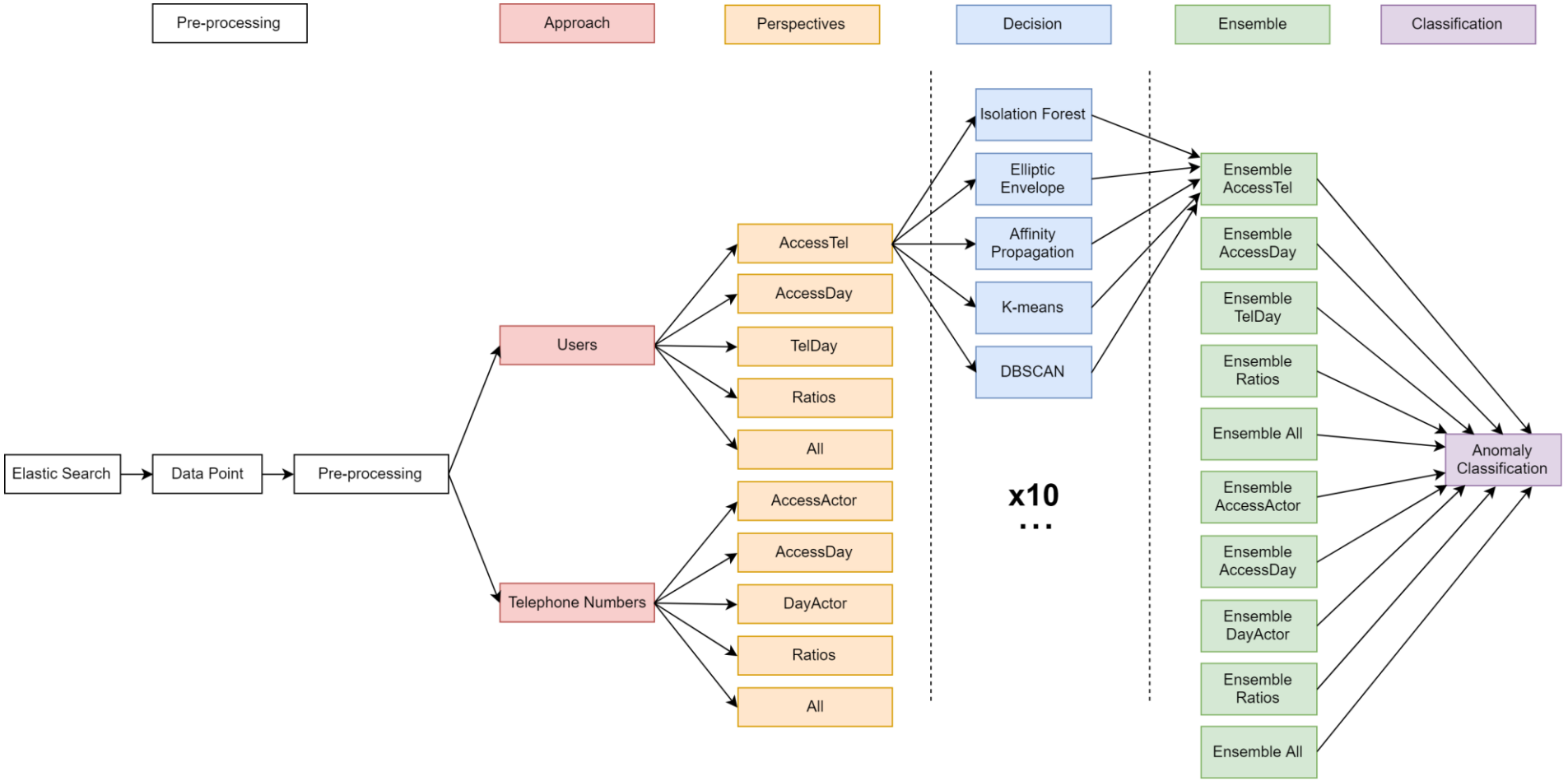

Figure 1- Project methodology

3.1 Data Study

The data provided is a big set of data logged during the normal working process of the company, the logs obtained in the interval period of 6 months create the APPS-RGPD dataset. The logs were exported from Elasticsearch to a jupyter notebook where the data was studied and treated. The APPS-RGPD dataset (application logs), contains 43 features, as shown in Table *1*. A log that forms this dataset means that a user accessed an application to consult a phone number. Each log characterizes an access, even though we are interested in behavior and not user characteristics, all features should be acknowledged and studied for a better decision on which records are relevant or not for the objective.

The features contained in the APPS-RGPD dataset are:

Table 1- APPS-RGPD features explained

action	Action performed by the actor, it will mostly be reads
action_details	Details about the action
action_query_invoice	Invoice consulted
action_query_ncc	NCC consulted
action_query_nif	NIF consulted
action_query_other	Another field used for the query (not well defined)
action_query_period_end	When the action terminated
action_query_period_start	When the action started
action_query_phone	Telephone number that was accessed
action_result	If the action ended successfully or not
actor	Who performed the access
actor_account_risk	Risk classification of the person who made the access
actor_account_state	If account is active or inactive
actor_company	Company of who made the access
actor_department	Department of who made the access
actor_details	Details about who accessed
actor_device_mac	MAC address of device used to make the access
actor_device_name	Device used to make the access

actor_domain	Active directory (AD) of who made the access
actor_identity_id	Anonymized identification of who made the access
actor_identity_risk	Risk classification (maximum) of the ILA identity of the person making the access
actor_ip	IP from where the access was made
actor_ip_ad_acc	AD account associated with this IP
actor_ip_as_country	Origin country of IP
actor_ip_as_name	AS name
actor_ip_as_number	AS number
actor_ip_hostname	Name of the machine from where the access was made
actor_ip_identity_id	ID of the AD account associated with this IP
actor_ip_range_type	If client ip is static or dynamic
actor_location	Details of location from where access was made
actor_location_geo	Detailed location from where access was made
actor_location_id	Location from where access was made
actor_network_range	Range of network from which the access was made
actor_network_type	Type of network from which the access was made
actor_type	If it is properly possible to associate the actor to a person
count	Events number
decorated	Details of the action performed
hint_actor_ad	Account indicated by the SFA portal
object	Application accessed
object_group	Group of the application accessed
query_features	Which ids were in the query
source	Event source
ts	Timestamp

Source and action will be ignored because the part of APPS-RGPD dataset used does not vary in these features. The action is always actors from source arm-audit accessing an application to consult telephone numbers.

3.1.1 Characterization of APPS-GPD dataset

A study was performed to gain more knowledge about the data, to find what the pattern of the relevant feature's behavior is, and to find which is the behavior that could be indicative of an illicit action being undertaken. Since the actors name and the telephone numbers are confidential data some plots will have the x-axis legend as a notion of the number of actors, identities or telephone numbers being represented.

Actor

In the period being studied, there are 7778 actors. The Figure 2 shows the distribution of the number of times each actor made an access, the chart highlights two things: most actors made accesses only a few times; and a very small fraction of actors made a large number of accesses.

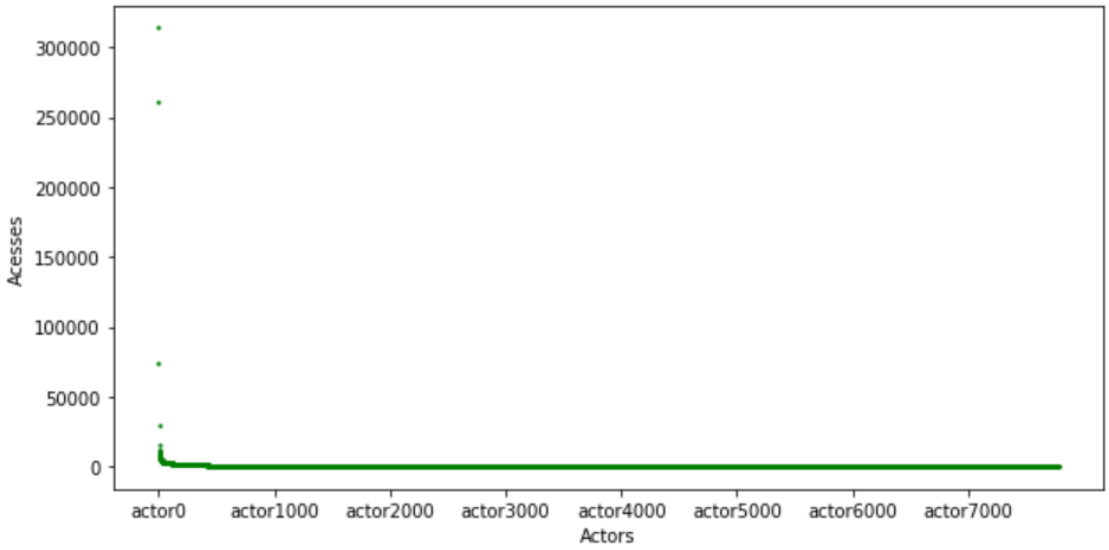

Figure 2- Distribution of the number of times each actor made an access

This distribution does not allow us to draw big conclusions because of the contrast in the values. As most actors appear only once or a few times and two actors appear 314102 and 261719 times. If the actors with many accesses are considered outliers and excluded from the study, it is possible to see the distribution curve with more detail, as presented in Figure 3.

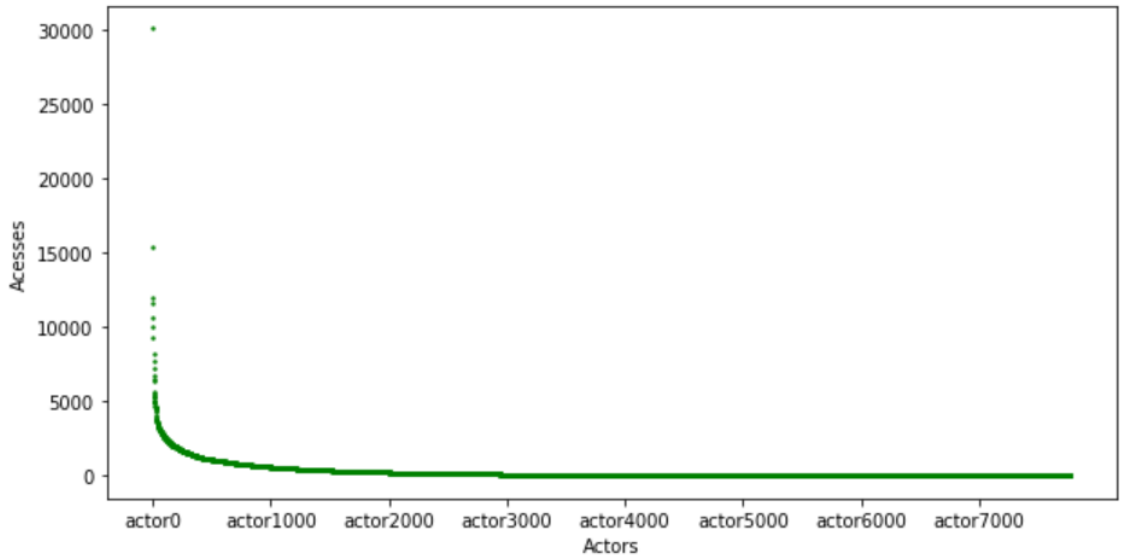

Figure 3- Distribution of the number of times each actor made an access without extreme cases

It is possible to see the distribution curve more subtly but the knowledge taken is the same. Most actors made accesses very few times compared to the few that made a lot of accesses.

Since in Figure 3 the distribution of the actors that appear less than 1000 times is not very perceptive, due to the large range of values, a graphic with the distribution of only the actors that appear less than 1000 times is presented in Figure 4.

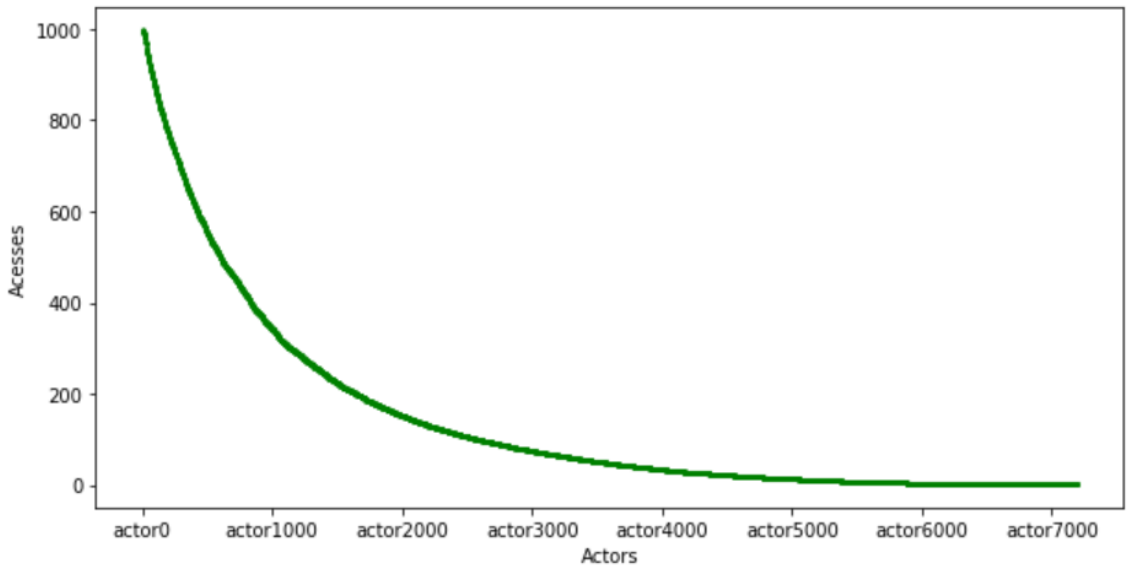

Figure 4 - Distribution of the number of times each actor, which made less than 1000 accesses, made an access

Another interesting analysis is to find how many different days each actor appears, as shown in Figure 5.

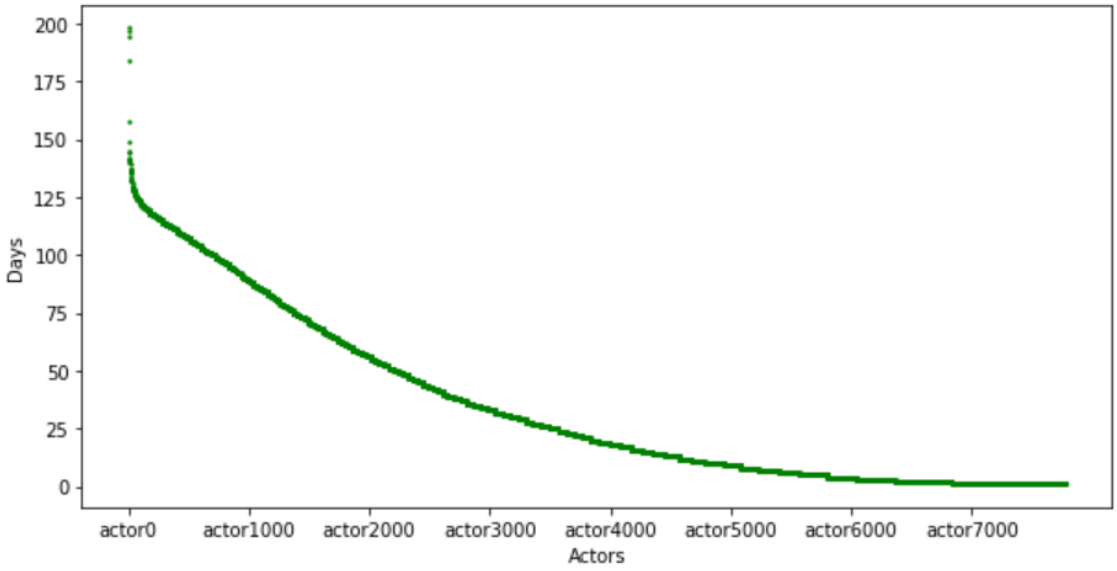

Figure 5- Distribution of the number of different days each actor made an access

No actor appears more than 184 different days and, as expected, since the majority of users only makes one access, mostly appear in just one day. The curve of this distribution is much more readable since there is not so much contrast in the number of days as there is in the number of times each actor appears, which means that many of the accesses are done in the same day.

Actor_identity_id

An actor is associated with an identity id, but an identity (person) can have many accounts (actor), so the distribution of this field should be similar to the actor field.

Two actor identities appear much more than the others, -1 (means that the identity is unknown), and test user (some users are just test accounts created for monitoring reasons), which appear 326921 and 224064 times respectively. They will be left out of our distribution for better visualization of this field. Also for the same reason, to study this field we will divide it into 3 different graphics.

First, in Figure 6, we will look at the identities that made the most accesses (above 10000 times).

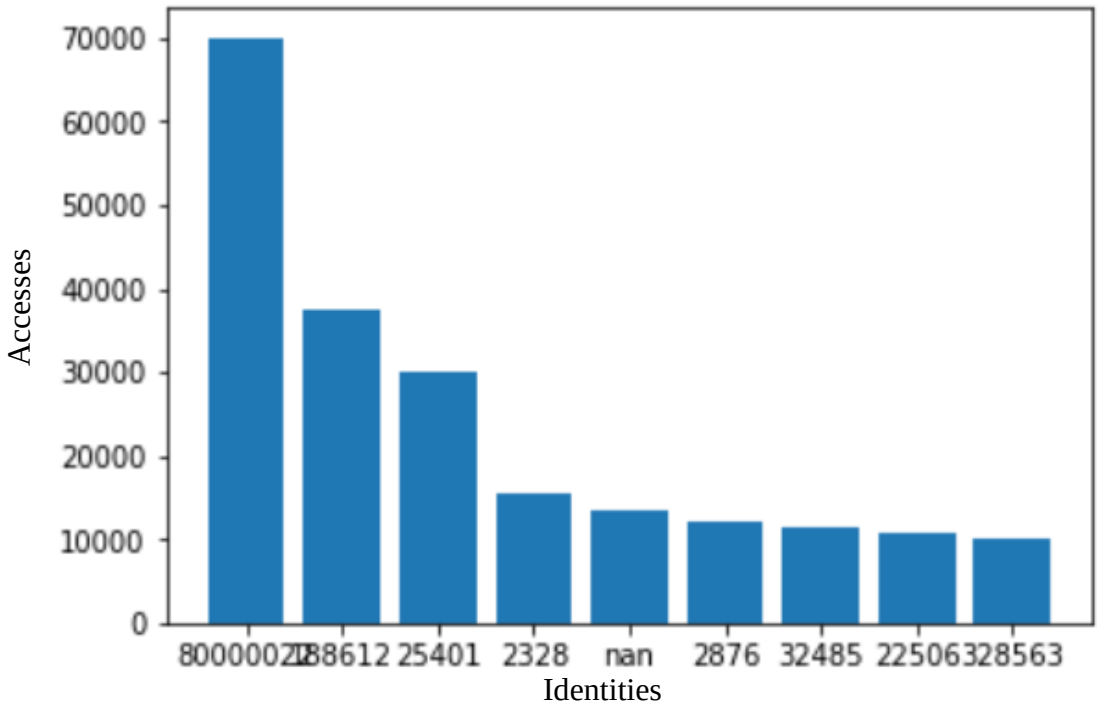

Figure 6- Distribution of the number of times each user, that made more than 10000 accesses, made an access

This field is empty in 13380 rows. Adding to that situations where the actor identity is not known, results in a total of 340301 times a number was accessed and there is no knowledge about the user identity id.

Figure 7 presents the identities that appeared between 10000 and 1000 times.

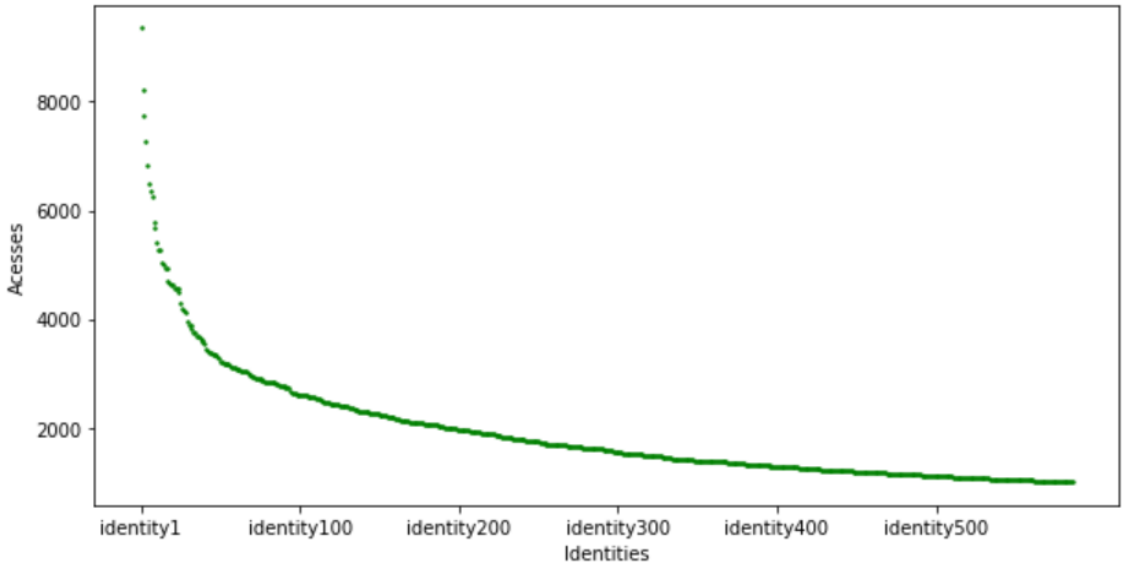

Figure 7- Distribution of the number of times each user, which made between 1000 and 10000 accesses, made an access

The distribution of the identities that appear more than a 1000 and less than 10000 times, is similar to the actors' field. Few user identities were responsible for more than 1000 accesses, and very few are responsible for more than 4000.

In Figure 8 we show the distribution of the identities that appear less than 1000 times.

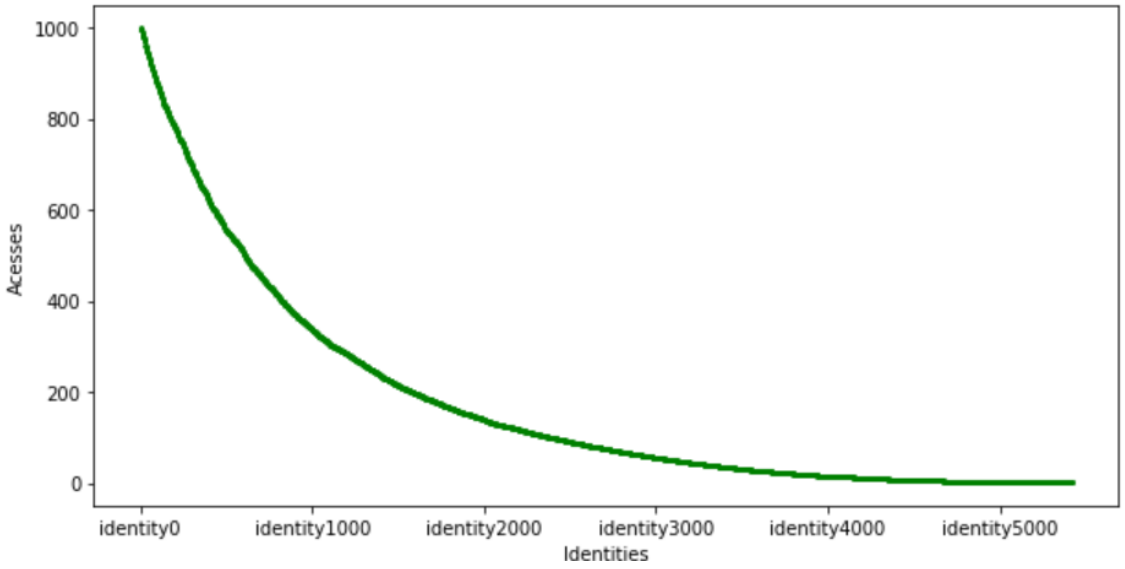

Figure 8-Distribution of the number of times each user, which made less than 1000 accesses, made an access

Most identities access telephone numbers less than 1000 times (5422 different identities), and the distribution is, as expected, very similar to the actor's distribution.

Figure 9 shows how many days each identity made accesses.

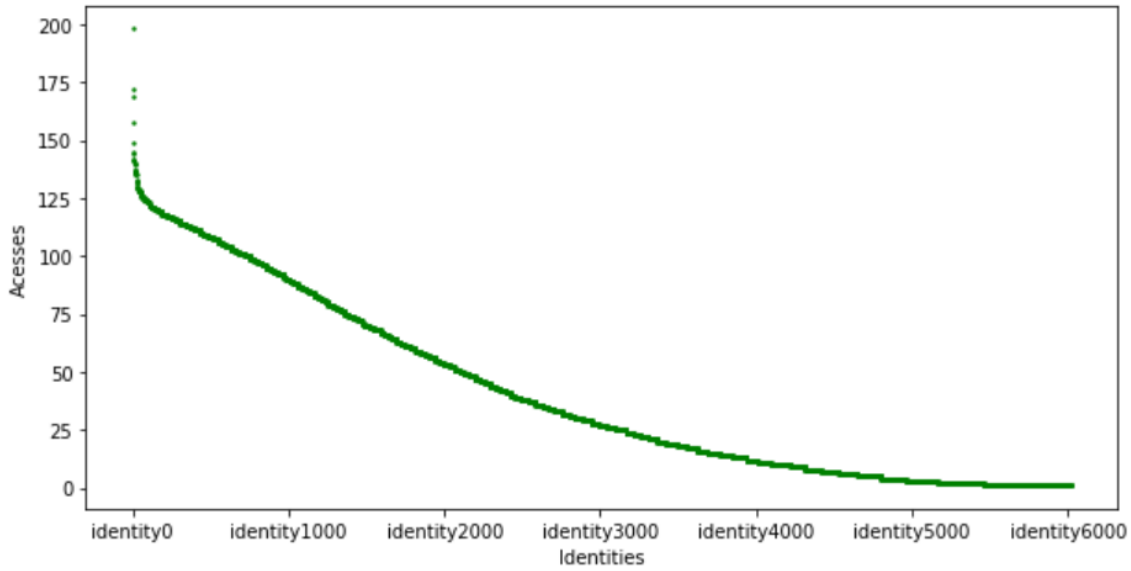

Figure 9-Distribution of the number of different days each user made an access

There is just one id that makes accesses every day. It will probably be a test account, but it is expected to be one of the obvious anomalies to be detected.

Action_query_phone

Action_query_phone represents the telephone number accessed by a user. It is the feature with the most variance, 641292 different telephone numbers are accessed during the period in study, therefore, once again, the distribution representation will be separated in 3 figures.

First, in Figure 10 we will be looking at the distribution of the phone numbers that are most commonly accessed (are accessed more than 1000 times).

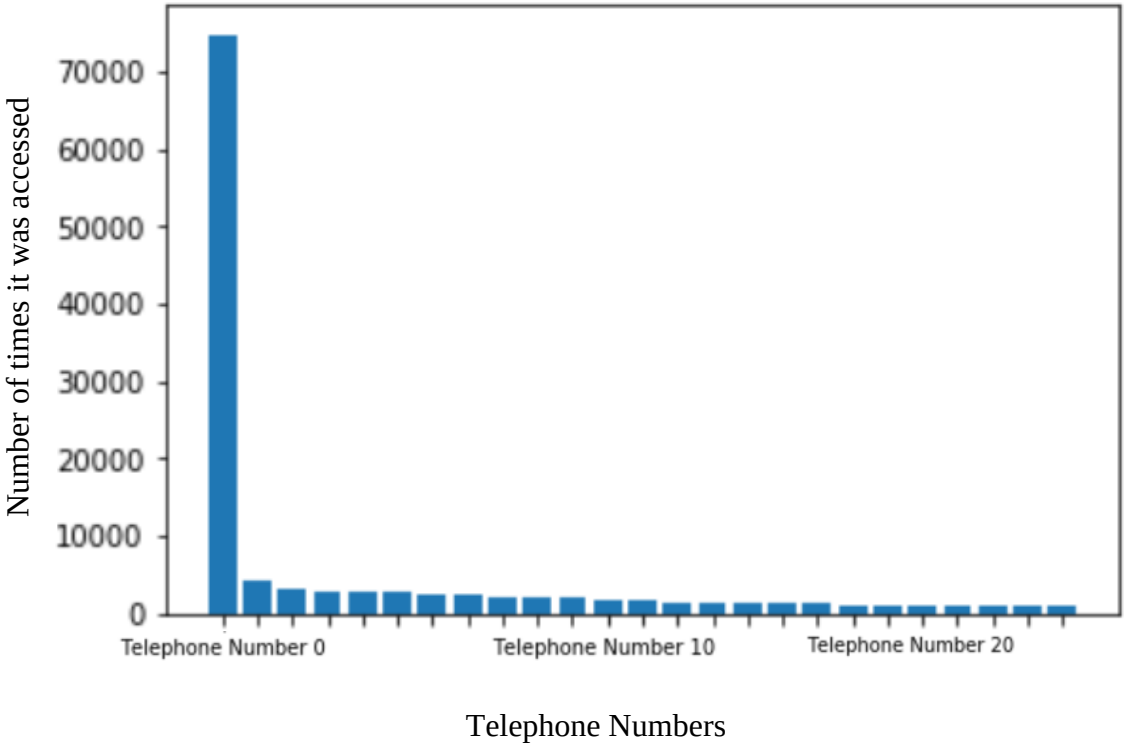

Figure 10-Distribution of the number of times each telephone number, which was accessed more than 1000 times, was accessed

There are 22 phone numbers that, in the period of 199 days, were accessed more than a 1000 times, the most observed number had 74766 accesses (this number is not represented graphically due to the fact that its magnitude would make the distribution much less detailed, it is a telephone number created for testing, it was considered an outlier and was excluded from the study).

Figure 11 observes the telephone numbers that were accessed between 1000 times and 10 times.

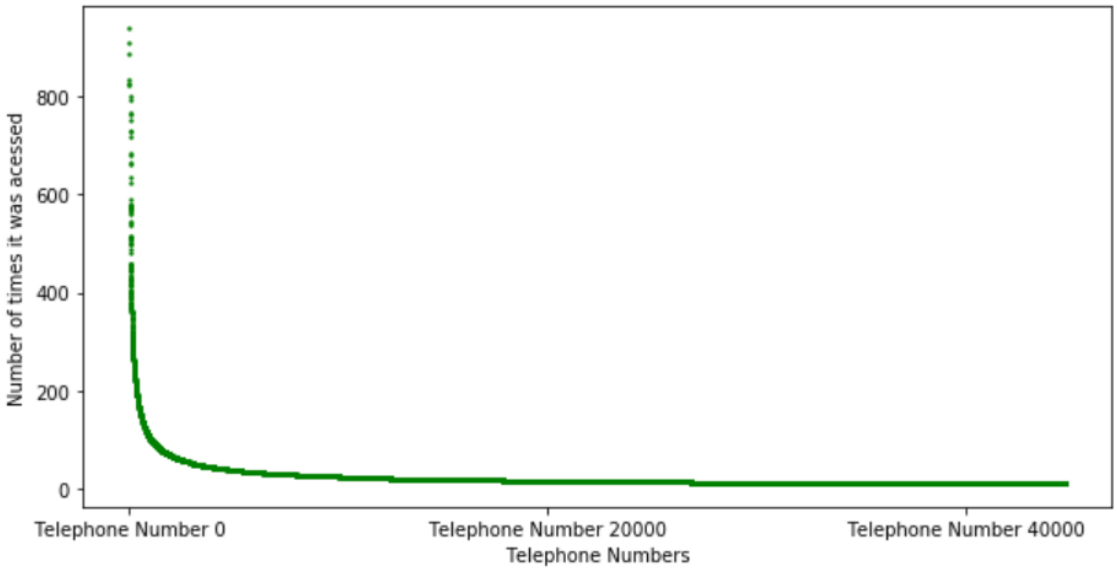

Figure 11-Distribution of the number of times each telephone number, which was accessed between 10 and 1000 times, was accessed

This distribution shows the most accentuated curve of the 3 plots since it involves a large amount of different telephone numbers and a big variance in quantities (the first has the biggest variance in quantities but very few telephone numbers, and the third has a lot of telephone numbers but a small variance in terms of quantities).

Figure 12 shows the telephone numbers accessed less than 10 times in the entire period.

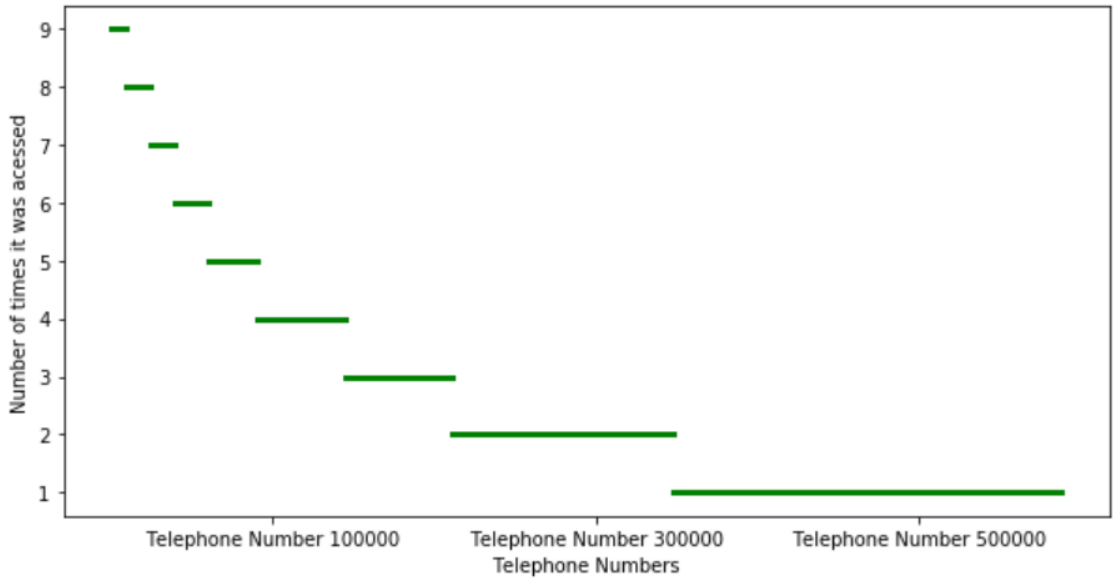

Figure 12-Distribution of the number of times each telephone number, which was accessed less than 1000 times, was accessed

As for the other fields, Figure 13 shows how many different days each telephone number is accessed, this is useful to discover what the pattern is for the number of days a

telephone number is accessed, if it is normal to be constantly accessed or, for example, if the pattern is for a telephone number to be consulted just one time in the entire period.

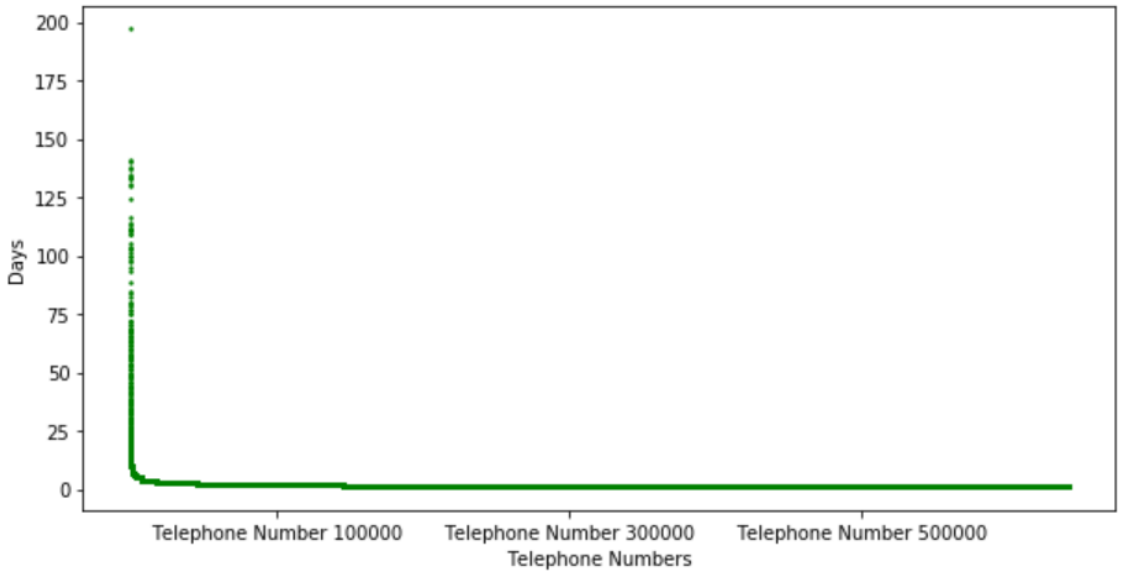

Figure 13-Distribution of the number of different days each telephone number was accessed

The maximum amount of days a telephone was accessed was in 199 distinct days.

Object

Actors access an application to read a telephone number. In Figure 14 we see which are the most common applications used. Even though this feature will not affect patterns as much as the ones presented before, it is still useful to learn its behavior.

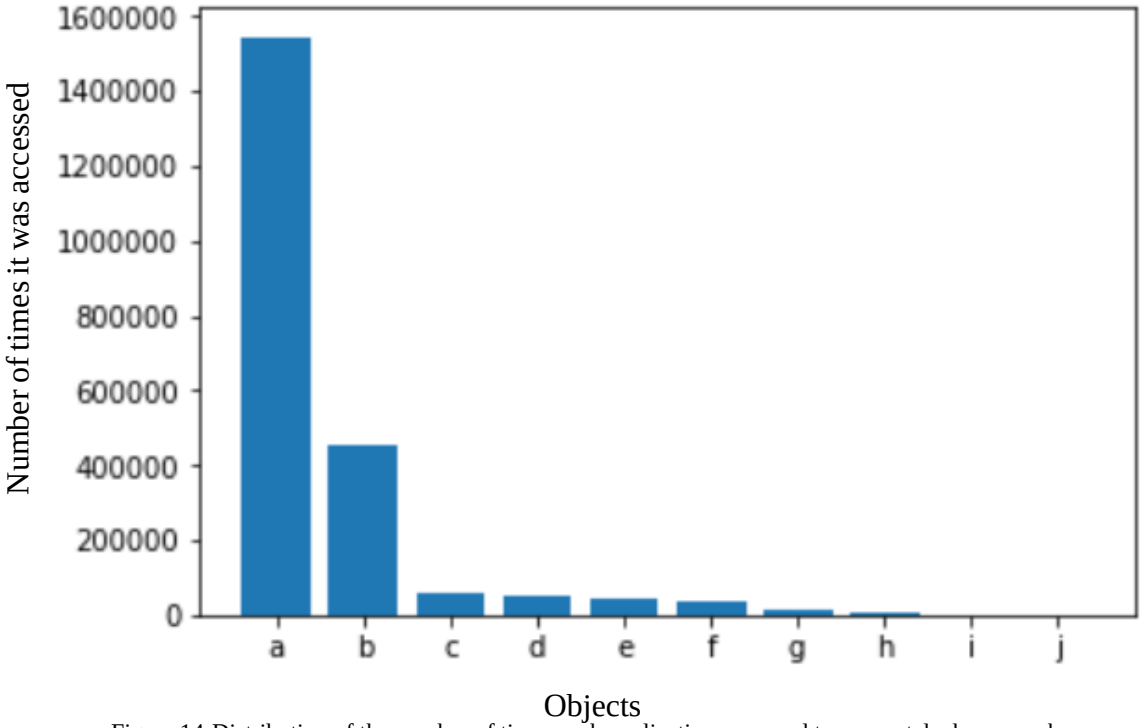

Figure 14-Distribution of the number of times each application was used to access telephone numbers

The most used applications are *a* and *b*, the first appearing in 1544429 access logs. Also interesting to see is how many different days each application is used, as shown in Figure 15.

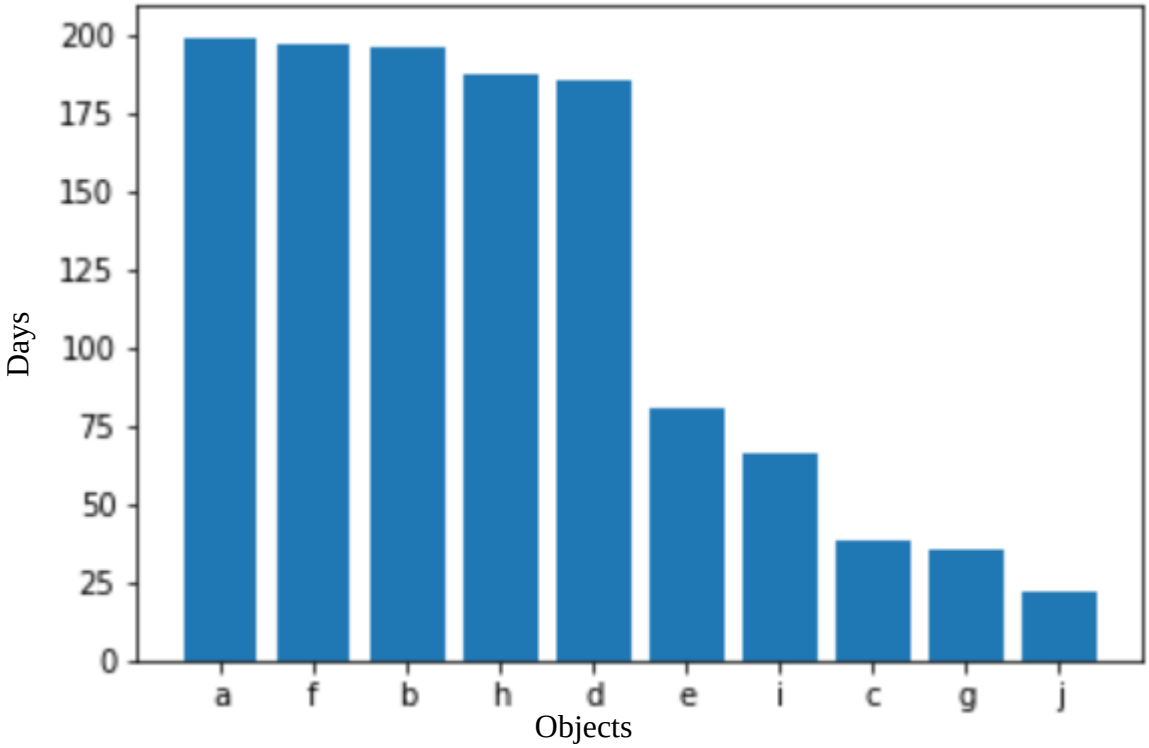

Figure 15-Distribution of the number of different days each application was used to access telephone numbers

The application that is accessed the most is *a*, being accessed every single day except for one.

Studying the distribution of these fields was helpful to learn which are the patterns and what is likely to be an anomaly. It was possible to see that the common behavior is for a user to do few accesses and, a user that makes several accesses may be indicative of illicit behavior.

3.2 Pre-processing

For the machine learning techniques to be effective, it was necessary to pre-process the data. After inspecting the data, it was decided which features of the original dataset are useful and should be used, which should be calculated, which should be transformed or treated, and which ones are not relevant.

As mentioned before, each log obtained from Elastic Search characterizes one access made by a user to a telephone number. Since the objective is to find anomalous behaviors, these characteristics alone will not be relevant. The relevant part is the behavior formed with the junction of accesses by user/telephone number. To obtain this, two approaches were performed.

3.2.1 User Approach

The users approach will be focused on users. The dataset created is indexed by user and has the relevant characteristics that define each user behavior. Our focus was the number of times a user made accesses, to how many telephones and in how many days. This way it is possible to detect the normal pattern of behavior and, for example, find which amounts of times or days is too much or too few, or which relations between amount of accesses and telephone numbers are more suspicious.

From the logs obtained from Elastic Search, which create the APPS-RGPD dataset, we retrieved the fields: *actor_identity_id*, *actor*, *action_query_phone*, *ts*, and *object*.

With these fields, a data frame was created (a type of table from the *pandas*[3] library, in python, very useful for machine learning) with every actor identity id. In some logs there is not identity id associated with the user. In those situations, the identity id was replaced with the actor. The field of the data frame with all the identity ids was also called *Actor_identity_id*. Then for each identity id, we went to the logs obtained and counted how many telephone numbers they had accessed. That field in the data frame is called *Action_query_phones*. Another interesting field was the number of days a user made accesses, for that the *ts* APPS-RGPD dataset field was used. For each access a user made, the date was checked, and it was counted in how many different days he had accessed a telephone number. The field was named *Days*. An obvious field to consider was the total amount of times a user made an access (field named *Count* in the data frame), it was simply to count how many times the id (or actor) appeared in the APPS-RGPD dataset. One more field was the number of different applications each user used. For each user we counted how many different applications that identity had in the logs *object* field. Finally, since this data frame was created for future applications of machine learning, the relations between these fields were also relevant. The ratio between the most significant behavior characteristics gave rise to three different fields: *Ratio_Access/Tel/Days*, *Ratio_Access/Tel* and *Ratio_Access/Days*. Table 2 describes each field of the dataset created for this approach.

[3] https://pandas.pydata.org/

Table 2-Features of the dataset created for users approach

Actor_identity_id	A unique, random, anonymized user identifier. Each user is identified by his identification number, in the cases this number does not exist, the user name will be used (actor).
Counts	Total number of accesses made by the user.
Action_query_phones	Number of telephone numbers the user accessed.
Objects	Number of applications the user used to access the telephone numbers.
Days	Number of different days in which the user made an access.
Ratio_Access/Tel/Days	Ratio between the total number of accesses of the user, the number of telephone numbers accessed by the user and the number of days in which the user made accesses.
Ratio_Access/Tel	Ratio between the total number of accesses of the user and the number of telephone numbers accessed by the user.
Ratio_Access/Days	Ratio between the total number of accesses of the user and the number of days in which the user made accesses.

Perspectives

To find different and specific kinds of anomalies, the algorithms were applied in different combinations of the created dataset features. This allows behaviors that could pass unnoticed when looking at all the features together. For example, a combination between the number of accesses made by the user and the number of telephone numbers the user accessed, will detect anomalies always focusing on these features, but when using

the entire dataset, if a user has these features somewhat out of the pattern but all the others are "normal" there is a higher chance that the anomaly will pass unnoticed. Table 3 describes the different perspectives analyzed.

Table 3- Different perspectives to study in users approach

AccessTel	Total number of accesses made by the user and the number of telephone numbers the user accessed.
AccessDay	Total number of accesses made by the user and the number of different days in which the user made an access.
TelDay	The number of telephone numbers the user accessed and the number of different days in which the user made an access.
Ratios	The three ratios created based on the other features.
All	All the features are considered.

3.2.2 Telephones approach

The telephones approach is focused on the telephone numbers. The objective of this approach is to detect the telephone numbers that have abnormal behavior. Then select the users that have accessed those telephone numbers and verify if these have abnormal nature. Once again, from the logs obtained from Elastic Search, which create the APPS-RGPD dataset, we retrieved the same fields: *actor_identity_id*, *actor*, *action_query_phone*, *ts*, and *object*. The difference came in the way these fields were used to create the new dataset.

A new dataframe was created, this one with every telephone number. Then it was counted, from the logs of accesses, how many different users had accessed each telephone number. This field was called *Actors*. The field *Days* was created again, but this time, for each time a telephone number was accessed, the date was checked and it was counted how many different days each telephone number was accessed. Another field that was also created was *Count*, the total number of times a telephone number was accessed. It was as simple as counting the number of times a telephone number appeared in the access logs. The field *Objects* in the dataframe means the number of different applications each telephone number was accessed by. For each telephone number, it was counted how many different applications had been used by users in the logs *object* field. Finally, just like in the users approach, the relations between these fields were also relevant. The ratio

between the most significant behavior characteristics gave rise to the three different fields: *Ratio_Access/Actor/Days*, *Ratio_Access/Actor* and *Ratio_Access/Days*. Table 4 describes each field of the dataset created for this approach.

Table 4--Features of the dataset created for telephones approach

Action_query_phone	The telephone number.
Count	Number of times that telephone number was accessed.
Actors	Number of users that accessed that telephone number.
Object	Number of applications users used to access that telephone number.
Days	Number of different days in which the telephone number was accessed.
Ratio_Access/Actor/Day	Ratio between the total number of times the telephone number was accessed, the number of users that accessed the telephone number and the number of days in which the telephone number was accessed.
Ratio_Access/Actor	Ratio between the total number of accesses of the user and the number of telephone numbers accessed by the user.
Ratio_Access/Days	Ratio between the total number of times the telephone number was accessed and the number of days in which the telephone number was accessed.

Perspectives

Just like for the users approach, we need to look at different perspectives of the dataset to detect the different types of anomalies. These are presented in Table 5.

Table 5- Different perspectives to study in telephones approach

AccessActor	Total number of times a telephone number was accessed and the number of users that accessed it.
AccessDay	Total number of times a telephone number was accessed and the number of different days in which it was accessed.
DayActor	The number of different days in which the telephone number was accessed and the number of users that accessed it.
Ratios	The three ratios created based on the other features.
All	All the features are considered.

3.2.3 Normalization

. Like in most machine learning situations, since we are dealing with features with very different intervals, the data needs to be normalized. Since, this data has “noise” (the data contains anomalies that are not supposed to be considered in the normal interval of each feature), instead of using the maximum and minimum number of a feature, we decided to use the 3rd quartile (75th quantile) and 1st quartile (25th quantile), respectively, therefore instead of the median we scale the data according to the quantile range. This means that Robust Scaler was the normalization method chosen.

3.3 Anomalies

Both datasets will be explored in various ways, to detect specific types of anomalies with the different algorithms. The types of anomalies considered are described in Table 6.

Table 6- Labels created to classify anomalies

1st type of anomaly	Many accesses to many telephone numbers, in many days
2nd type of anomaly	Many accesses to some telephone numbers, in many days
3rd type of anomaly	Many accesses to one or few telephone numbers, in many days
4th type of anomaly	Many accesses to many telephone numbers, in one or few days
5th type of anomaly	Many accesses to some telephone numbers, in one or few days
6th type of anomaly	Many accesses to one or a few telephone numbers, in one or few days
7th type of anomaly	Many accesses to a telephone number, just one day
8th type of anomaly	Some accesses to a telephone number, just one day
9th type of anomaly	Some accesses to one or a few telephone numbers, in one or few days
10th type of anomaly	Extreme cases (obvious anomalies)

All of these types could be relevant depending on the specific situation looked for. The 3rd, 4th and 7th type of anomaly are the ones that represent the most pertinent situations looked for. The 3rd type of anomaly could be a case of stalking or tracking. Someone is following the activity of a certain telephone for a long period. The 4th type represents a massive extraction of information of various telephone numbers in a short period. This could mean that someone is performing a fast search to raise clients for another telecommunications company. The 7th type, a massive extraction of information of a telephone number in a short period. This could mean, for example, someone is searching for some specific information about a telephone number. All these situations are examples of very relevant cases this project intends to detect. Even though the other ones are not as much of a priority, they could also represent relevant cases. The 8th type of anomaly, for example, could be a situation of someone searching for a telephone number because a friend from the outside of the company asked for. The same goes for the 9th anomaly type, but in this

case, instead of one, the person would be searching in two or three telephone numbers. The 10th type is very likely a test account due to its behavior numbers being so high. Succinctly, all the types could represent a relevant situation of illicit accesses to clients telephone numbers information, but a few types have priority, due to its specificity towards what we are looking for.

3.4 Algorithms

When it came to choosing the algorithms that were going to be used we tried to diversify. Both clustering and anomaly detection algorithms were taken into consideration. Anomaly detection algorithms to serve their purpose and clustering algorithms to divide the data in a way that it is possible to interpret the separation of the anomalies from the users that follow the pattern, meaning that bigger clusters represent identities that follow the pattern, and very small and more distant clusters represent identities having an "abnormal" behavior. In the case of DBSCAN, not only it returns clusters but also the identities that do not fit any cluster. These identities will also be interpreted as anomalies. K-means and DBSCAN were considered because they are two different general purpose clustering methods, affinity propagation was used because it has the ability to determine the number of clusters without specific parameters.

3.4.1 K-means

One of the algorithms used was K-means[4]. Since it is the most known and tested clustering algorithm it was relevant to take it into account. Since in the beginning the important is to start understanding the data and try different things, we started by it. Even though it is the most known and very simple clustering algorithm it can obtain very good results as seen in "A survey of network anomaly detection techniques" [5].

The most important parameter in this algorithm is the number of clusters chosen. This was the parameter that had to be optimized.

Parameters used:

- **N_clusters**: The number of clusters to form (To be decided in the parameterization step);
- **Init**: Method for initialization ('k-means++', selects initial cluster centers for k-means clustering in a smart way to speed up convergence);

[4] https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html

- **N_init:** Number of time the k-means algorithm will be run with different centroid seeds. The final results will be the best output of *n_init* consecutive runs in terms of inertia (10);
- **Max_iter:** Maximum number of iterations of the k-means algorithm for a single run. 7000 would be the number chosen for more rigorous results, but after trying it, the results remained equally good and the execution time increased a lot, so it did not compensate (3000);
- **Tol:** Relative tolerance with regards to inertia to declare convergence(1e-4);
- **N_jobs:** The number of parallel jobs to run (-1, means using all processors).

3.4.2 DBSCAN

In DBSCAN[5], we do not have to select a number of clusters directly. It has two other important parameters, the minimal distance between points and the minimal number of neighbors a point has to have to not be considered noise. As mentioned in "The Anomaly Detection by Using DBSCAN Clustering with Multiple Parameters" [11], DBSCAN performance can be very good compared to other algorithms, with the right parameterization applied.

Parameters used:

- **Eps:** The maximum distance between two samples for one to be considered as in the neighborhood of the other (to be decided in the parameterization step);
- **Min_samples:** The number of samples in a neighborhood for a point to be considered as a core point (to be decided in the parameterization step);
- **Algorithm:** The algorithm to be used by the NearestNeighbors module to compute pointwise distances and find nearest neighbors ('auto', the algorithm attempts to determine the best approach from the training data);
- **N_jobs:** The number of parallel jobs to run (-1, means using all processors).

[5] https://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html

3.4.3 Affinity Propagation

Affinity Propagation[6] is an interesting algorithm to use since it calculates the number of clusters alone. There were still a few parameters that needed to be refined, the number of iterations to converge, the maximum number of iterations and the preference for each point. It acquired great clustering results in "Smart Audio Sensors in the Internet of Things Edge for Anomaly Detection" [10], especially with its ability to determine the number of clusters the data should be divided in.

Parameters used:

- **Damping:** Damping factor is the extent to which the current value is maintained relative to incoming values (0.5);
- **Max_iter:** Maximum number of iterations (7000);
- **Convergence_iter:** Number of iterations with no change in the number of estimated clusters that stops the convergence (to be decided in parameterization step);
- **Preference:** Preferences for each point. Points with larger values of preferences are more likely to be chosen as exemplars (the negative value of the maximum distance between points, this was chosen to reduce the number of points as exemplars, reducing the number of clusters created);
- **Affinity:** Which affinity to use ('euclidean', uses the negative squared euclidean distance between points).

3.4.4 Elliptic Envelope

Another type of algorithms to try were anomaly detection algorithms. Unlike clustering algorithms, the results obtained in these are the points considered anomalies, with no need for an interpretation of the results to take the smallest clusters as the anomalies. The main parameter in Elliptic Envelope[7] is the amount of contamination. According to "Smart Audio Sensors in the Internet of Things Edge for Anomaly Detection" [12] results and methodology, elliptic envelope algorithm was expected to deliver a good contribution to the final results.

Parameters used:

[6] https://scikit-learn.org/stable/modules/generated/sklearn.cluster.AffinityPropagation.html
[7] https://scikit-learn.org/stable/modules/generated/sklearn.covariance.EllipticEnvelope.html

- **Store_precision:** Specify if the estimated precision is stored (true);
- **Support_fraction:** The proportion of points to be included in the support of the raw MCD (Minimum covariance determinant) estimate;
- **Contamination:** The amount of contamination of the data set, which means the proportion of outliers in the data set (to be decided in the parameterization step);
- **Random_state:** The seed of the pseudo-random number generator to use when shuffling the data ('None', the random number generator is the RandomState instance used by np.random).

3.4.5 Isolation Forest

For the same reasons given for elliptic envelope, another anomaly detection algorithm considered was Isolation Forest[8]. This algorithm also has the contamination parameter as its most important influencer in the results. "Smart Audio Sensors in the Internet of Things Edge for Anomaly Detection" [12] also relates isolation forest as a relevant contribution to the final results.

Parameters used:

- **Max_samples:** The number of samples to draw from X to train each base estimator (10000 for users approach and 100000 for telephones approach);
- **Contamination:** The amount of contamination of the data set, which means the proportion of outliers in the data set (to be decided in the parameterization step);
- **Max_features:** The number of features to draw from X to train each base estimator (the number of features of the dataset in which the algorithm is applied);
- **Random_state:** The seed of the pseudo-random number generator to use when shuffling the data ('None', the random number generator is the RandomState instance used by np.random);

[8] https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.IsolationForest.html

- **N_jobs:** The number of parallel jobs to run (-1, means using all processors);

3.5 Parameterization

As mentioned before, in each approach, the five algorithms were applied to each feature combination. Score tables were created for each case according to the methodology shown in Figure 16, which depicts the columns in each table. The results were then used to select suitable parameters for each method and perspective.

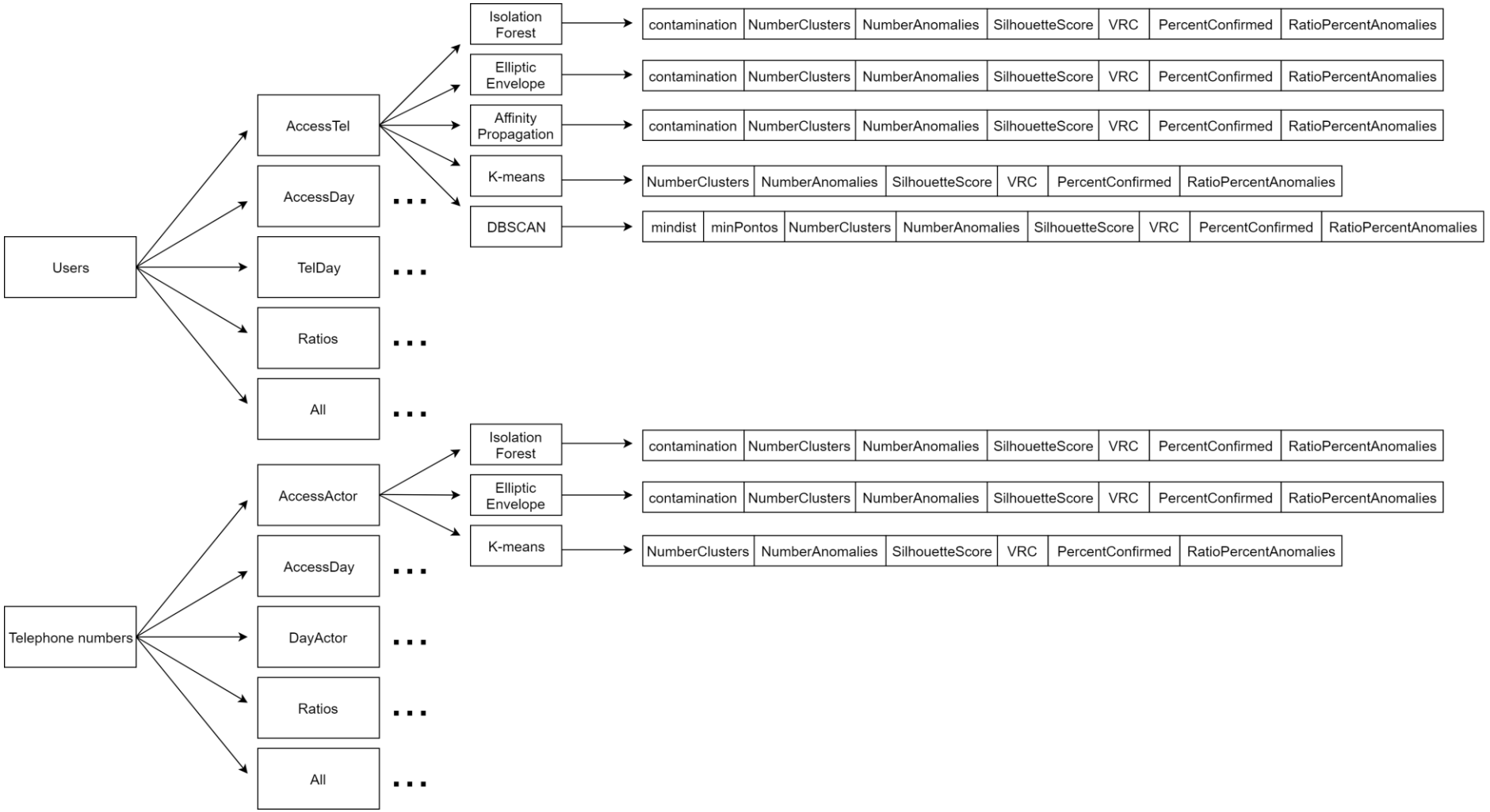

Figure 16- Explanation of score tables created for the parameterization optimization

The first column, or two first columns in the case of DBSCAN, in a score table are related to the method parameters being optimized. Then we have the number of clusters formed with that parameter (in the case of K-means the parameter itself is the number of clusters), the number of anomalies detected and the evaluation metrics, silhouette score and VRC. Then, there is the percentage that the detected anomalies cover from a list with 103 identities, already analyzed and confirmed as having suspicious behavior (this list contains known abnormal cases that have been approved as having suspicious behavior but does not contain, by far, all the anomalies existent), and, finally, the ratio between that percentage and the number of anomalies.

These score tables are referenced to users, therefore in the process of choosing the best parameterization for the telephones approach (in order to telephone numbers), the results obtained in each algorithm with the respective parameters are converted to the user approach. The conversion process is, as explained before, the telephones approach detects the telephone numbers that have abnormal behavior, and then the users that have accessed those telephone numbers are selected and considered the anomalies.

This project has the intention to help the company detect users and the type of illicit actions they are undertaking, so the process should be implemented periodically. Even though it is not established if it would be once a week, once a month, or even once every 2 or 3 months, the process should be minimally fast. The aggregated dataset for the telephone approach is much larger than the user dataset. For that reason, for the users approach 5 algorithms were applied, but for the telephones approach, the affinity propagation and DBSCAN algorithms weren't considered, due to their high time and space complexity.

In summary, the parameters being optimized and the values considered for each algorithm, are presented in Table 7.

Each parameter varied between a certain range of values:

Table 7- Parameterization range tested

Algorithm	Parameter	Interval
Isolation Forest	contamination	[0.001, 0.005, 0.01, 0.05, 0.08, 0.10, 0.15, 0.20, 0.25, 0.30, 0.35, 0.40]
K-means	number of clusters	[2,3,4,5,6,7,8,9,10,11,12,15,16,18,20,25,30,35,40,45, 50,60,70,100,150]
DBSCAN	mindist	[0.1,0.5,1,2,3,4,5,6,7,8,9,10,11,12,15,16,18,20,25,30, 35,40,45,50,60,70,100,150]
DBSCAN	minPoints	[1,2,3,4,5,6,7,8,9,10,11,12,15,16,18,20,25,30,35,40,4 5,50,60,70,100,150]
Affinity Propagation	convergence	[1,5,10,50,80,100,150,200,250,300,350,400,450,500, 550,600,700,750,800,850,900,950,1000]
Elliptic Envelope	contamination	[0.001, 0.005, 0.01, 0.05, 0.08, 0.10, 0.15, 0.20, 0.25, 0.30, 0.35, 0.40]

For a better evaluation and consistency of the results with the different parameters, the algorithms with random initialization were executed 10 times for each parameterization and the values on the scores are the mean of the results of the 10 executions. An algorithm with random initialization means that depending on the initialization that occurs, results may vary. In terms of K-means, for example, it means that initially all samples are assigned with a cluster label. Elliptic Envelope and Isolation Forest are the other two algorithms with random initialization.

Both clustering and anomaly detection algorithms were used, therefore a way to evaluate them similarly, to be possible to compare, was necessary. Silhouette score and VRC are clustering evaluation metrics, they are applied to a vector with the cluster label for each sample. The anomaly detection algorithms used, return a vector that can be interpreted as a clustering result with only two clusters. The anomalies are labeled as "-1" and the rest of the data labeled as "1". So, these evaluations metrics were used for both the clustering and anomaly detection algorithms.

As explained before, silhouette score calculates the silhouette coefficient for all samples and then returns the mean of these coefficients. To calculate that coefficient it has to calculate distances between all points. It was expected that the execution time applied to the data of the users approach and the data of the telephones approach would

be very different. With the telephones data, the time complexity was very high. After researching for ways to solve this problem, the one considered best was to use a sample of the data for the evaluation. Different sample sizes were tested to see the best way to find a time and accuracy balance, in other words, a way to have enough data to give a realistic idea but that also takes a reasonable amount of time to execute. The sample size used was 10000, which means that only 10000 random samples were used to evaluate an algorithm. Had this happened in the users approach, this solution would not have been as reliable, but since the three algorithms applied to the telephones approach all have random initialization, the silhouette score was calculated 10 times in all of them. Thus, the silhouette score that appears in each table score is the mean of 10 times the silhouette score was applied, which means, that 10000 random samples were chosen 10 times for the evaluation. This gives a possibility of up to 100000 different samples to be used.

One more problem encountered was a situation in which an algorithm applied to a determined perspective did not find any anomalies, it considered all samples similar to each other. This was a problem because for both silhouette score and VRC to work they need at least two clusters. To solve this problem, since it was only encountered in a situation in which the algorithm had to run 10 times, and out of 10 times, the problem only happened at most two times, the silhouette score and VRC did not run for these cases. This means that out of the 10 times the algorithm was applied, if no anomalies were found one time, the silhouette score and VRC in score table would be the mean of nine times they run. The score tables in the appendices that had this problem are marked.

To make the best parameterization decision for the users approach we had in mind the following factors, considering also previous knowledge on the anomalies:

- Silhouette Score must be higher than 0,70 (to consider only the cases in which the points firmly belong to the clusters);
- VRC has to be larger than 100 (the VRC must be high to have knowledge that we are working with an optimal number of clusters, after looking to the variances of its value on the different score tables, this was the value considered to be the minimum acceptable);
- Number of anomalies has to be larger than 10 (if less than 10 we would always obtain the ten most obvious anomalies which would not contribute to the objective of discovering new anomalies);
- Number of anomalies has to be smaller than 300 (if there are more, there is a big probability of considering higher proportions of false positives as anomalies);

- The *PercentConfirmed* should be the highest possible with the highest *RatioPercentAnomalies* possible.

Since the size of the dataset for the telephone numbers is much higher than the number of users the same rules could not be applied.

To make the best parameterization decision for the telephones approach we took into account the following factors, considering also previous knowledge on the anomalies:

- Silhouette Score must be higher than 0,70 (to consider only the cases in which the points firmly belong to the clusters);
- VRC has to be larger than 100 (the VRC must be high to have knowledge that we are working with an optimal number of clusters, after looking to the variances of its value on the different score tables, this was the value considered to be the minimum acceptable);
- Number of anomalies has to be smaller than 2000 (With this amount there is possibly already a high number of false positives, but this could be accepted since there were no results with options for the users approach limits imposed, and since this factor was taken into account in the posterior steps);
- The *RatioPercentAnomalies* should be the highest possible with the highest *PercentConfirmed possible*. (In this situation the *PercentConfirmed* value was not as relevant as in the users approach since the variance in the number of anomalies is so high that the little (in comparison) amount of *PercentConfirmed* coverage does not bring so much information.

3.6 Ensemble

The ensemble step is where the results of all algorithms, in all perspectives, are combined. The objective of this step is to find the most trustworthy way to combine all the different algorithm results and extract the best decisions from it.

To decide which was the best ensemble method, three approaches were taken into account.

- **UnionUnion**: Union of the results obtained in each algorithm for a feature combination, with the posterior union of the results obtained in all the perspectives;
- **IntersectUnion**: Intersection of the results obtained in each algorithm for a feature combination (an anomaly is only considered if it was obtained in all

the algorithms when applied to a certain feature combination), with the posterior union of the results obtained in all the perspectives;

- **VotingUnion**: Vote method, an anomaly is only considered if it was obtained in the majority (3 in the users approach and 2 in the telephones approach) of the algorithms applied to a certain feature combination, with the posterior union of the results obtained in all the perspectives.

3.7 Classification

At the end of the process, when we have the anomalies obtained, it is necessary to label each of them in one of the anomaly types previously defined. An anomaly can correspond to more than a type, but all of them need to belong to at least one. This was done through decision rules, where each anomaly was compared to each type conditions, and labelled to the types it belonged. After studying the data and a team process to define specific illicit situations to consider, rules and limits were created for each of the situations characterization.

Table 8 shows which were the conditions that made an anomaly belong to each type.

Table 8- Rules followed to distinguish the anomaly labels created

	Number of accesses	Number of days it made accesses	Number of telephone numbers it accessed
Type 1	>100	> 30	> 50
Type 2	>100	> 30	<= 50
Type 3	>100	> 30	<= 10
Type 4	>100	<= 30	> 50
Type 5	>100	<= 30	<= 50
Type 6	>40	<= 30	<= 10
Type 7	>= 20	== 1	== 1
Type 8	< 20	== 1	== 1
Type 9	< 40	<= 30	<= 10
Type 10	>10000	> 150	> 1000

3.8 Process Automation

All the different methodology steps were done in different jupyter notebooks. The final step would be to aggregate all of them in a single python process. But, with the different steps in different notebooks, with jupyter web interface, error solving becomes a much easier task, because each step is in a different jupyter notebook facilitating errors search. Therefore the final idea was to leave all the steps isolated but write the process in a shell script that would run all the notebooks in the right order. The process architecture is shown in Figure 17.

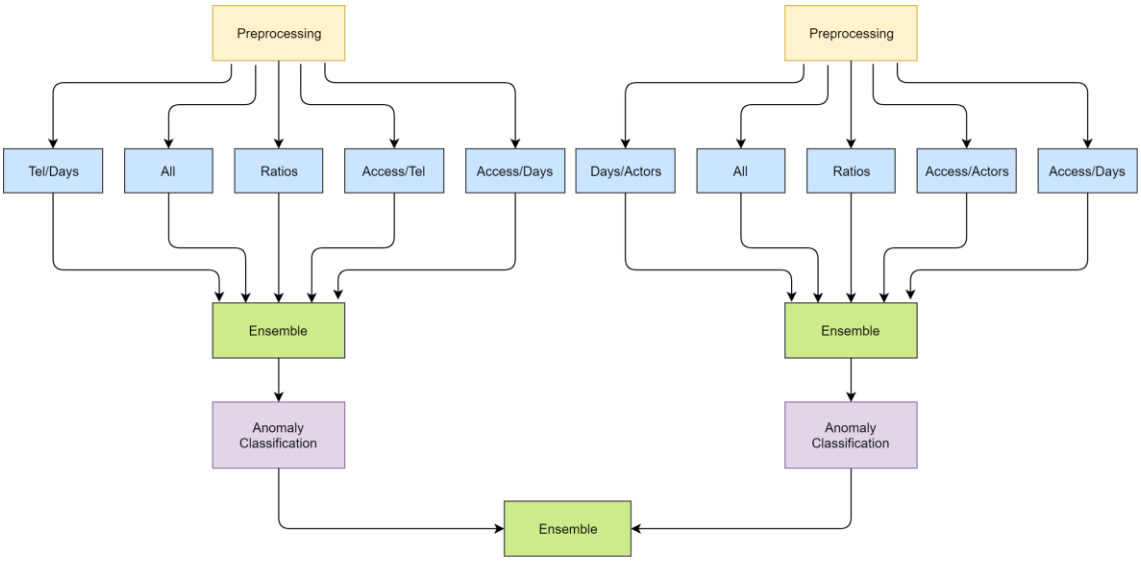

Figure 17- Automation process architecture

The pre-processing is the first part to run to prepare the data. Both approaches run in parallel. The perspective and decision steps wait until pre-processing is finished because they need the results to continue. Then, all the different perspectives run in parallel. After all the decisions in the different perspectives are done, their results are used in the ensemble. Finally, the notebook responsible for anomaly classification runs and is responsible for the final results, ending the process.

Chapter 4 Results

All the parameterization results related to the execution of the methodology presented in section 3.5 (Figure 16), are presented in appendices B to F (users approach) and G to K (telephones approach). Appendix A shows abbreviations used in the results tables.

4.1 User Approach

Following the rules created to choose the best parameterization, we obtained the best parameters for each <algorithm, feature perspective> pair, presented in Table 9.

Table 9-Users approach parameterization results

	AccessTel	AccessDay	TelDay	Ratios	All
Isolation Forest (percentage of contamination)	0.01	0.01	0.01	0.01	0.01
K-means (number of clusters)	6	7	4	8	9
DBSCAN {mindist, minPoints}	{1,35}	{1,150}	{0.5,35}	{3,45}	{3,4}
Affinity Propagation (convergence)	5	10	---	10	10
Elliptic Envelope (percentage of contamination)	0.01	0.01	0.01	0.01	0.01

With the optimal parameterization, the algorithms were applied to each of the five different feature perspectives. For the results of every application, an overall human analysis was performed to see if we were dealing with cases of possible anomalies or if there was a significant percentage of false positives. The outcome was positive since the anomalies found, that were studied in detail, had a suspicious behavior that fit at least one of the anomalies types defined. Since it is a human analysis, it has to be taken into account that it does not give us total certainty.

The affinity propagation applied to the TelDay perspective returned thousands of clusters for all the different parameterization tried, so this combination was not used for the final results.

To better understand the users detected by each algorithm, plots based on each perspective features were created. All plots had, the already known, test users removed for better visualization and interpretation. The red points are all the users and the blue points are the anomalies detected.

Perspective AccessDay

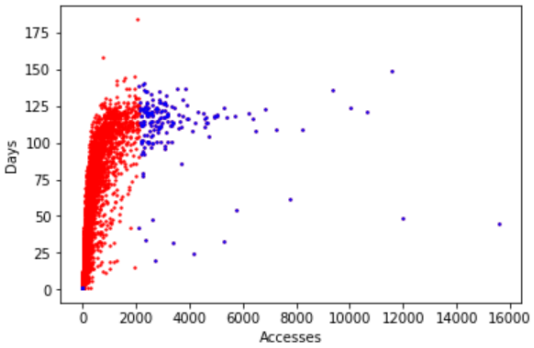

Figure 18- Anomalies detected by DBSCAN

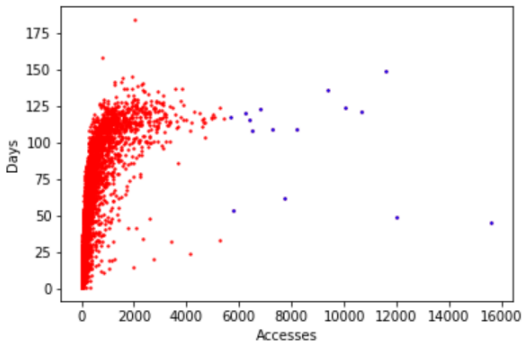

Figure 19 -Anomalies detected by K-means

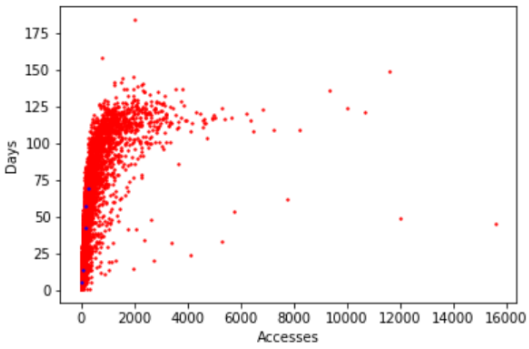

Figure 20-Anomalies detected by Affinity propagation

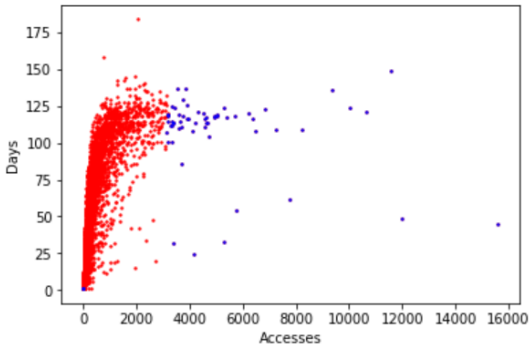

Figure 21-Anomalies detected by Elliptic Envelope

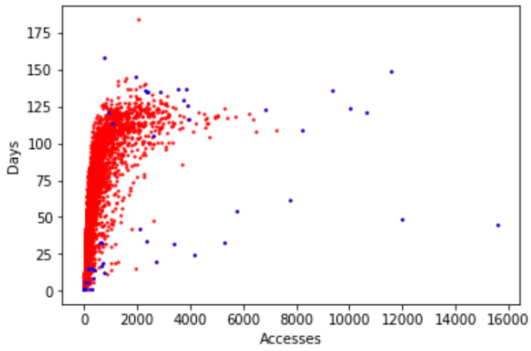

Figure 22-Anomalies detected by Isolation Forest

Table 10-Comparison of results of the different algorithms in AccessDay perspective

Algorithm	PercentConfirmed	RatioPercentAnomalies
Isolation Forest	2.912	0.044
K-means	0.971	0.065
DBSCAN	12.621	0.070
Affinity Propagation	0	---
Elliptic Envelope	5.825	0.088

Looking at the different plots of figures 18 to 22 it is possible to predict that DBSCAN detects the biggest amount of users out of the pattern (probably correct anomalies). Affinity propagation it is probably the one that not only detects the least, but also the least important ones. When observing Table 10 it is possible to see that DBSCAN has the most coverage over the list with 100 identities and the highest ratio between the number of anomalies and the percentage coverage, this means that there is a good balance between the number of anomalies detected and the confirmed anomalies coverage, with attention to the fact that the list containing 100 identities is just used as a guide since the 100 identities are not all the anomalies existent, neither the most important, it is just a list in which there is knowledge of identities confirmed as anomaly situations.

Perspective AccessTel

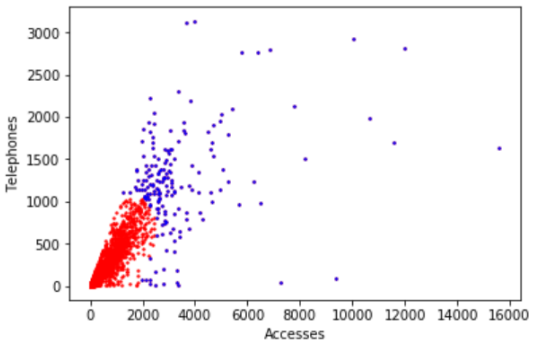

Figure 23-Anomalies detected by DBSCAN

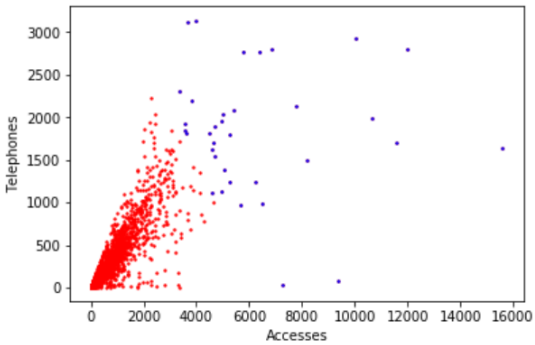

Figure 24-Anomalies detected by Elliptic Envelope

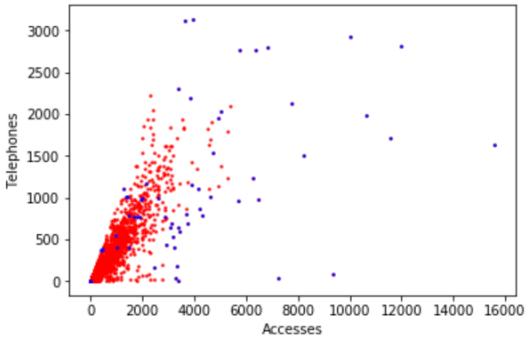

Figure 25-Anomalies detected by Isolation Forest

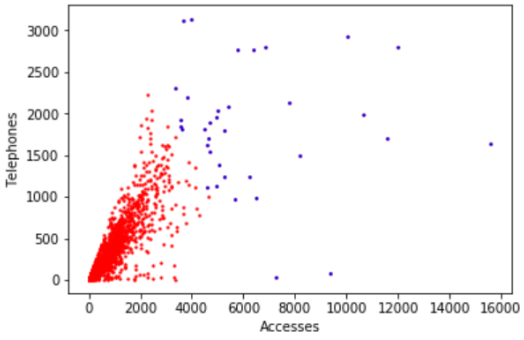

Figure 26-Anomalies detected by Affinity Propagation

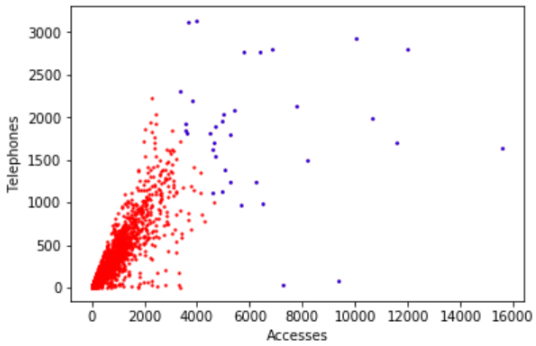

Figure 27-Anomalies detected by K-means

Table 11-Comparison of the results of the different algorithms in AccessTel perspective

Algorithm	PercentConfirmed	RatioPercentAnomalies
Isolation Forest	4.854	0.076
K-means	1.941	0.050
DBSCAN	12.621	0.070
Affinity Propagation	2.912	0.073
Elliptic Envelope	8.738	0.132

In this perspective, as it can be seen in figures 23 to 27, all algorithms seem to do a good job of detecting the most important anomalies. Isolation forest is probably the one that is the least suitable for this perspective, detecting not only the least but also some that maybe are not reliable. Table 11 shows that DBSCAN is the algorithm with the highest coverage over the confirmed anomalies, it does not have the highest ratio between the number of anomalies and the percentage coverage, but that is due to the fact that DBSCAN has the highest number of anomalies detected. Even though the number of anomalies being higher helps it to have a higher confirmed anomalies coverage, if we look at the balance between the two, and at the plots, it is possible to conclude DBSCAN is the best algorithm in this perspective.

Perspective TelDay

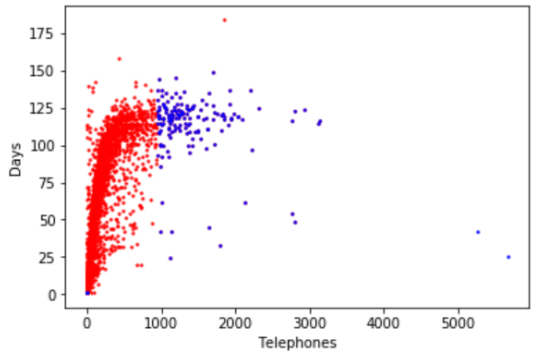

Figure 28-Anomalies detected by K-means

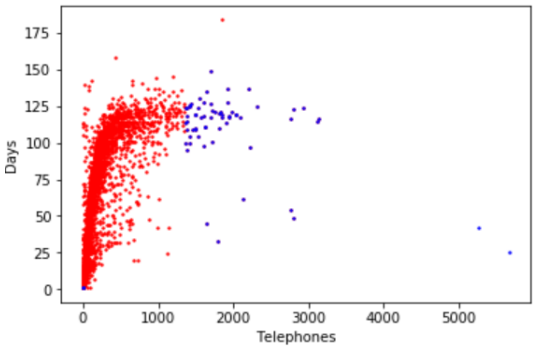

Figure 29-Anomalies detected by Elliptic Envelope

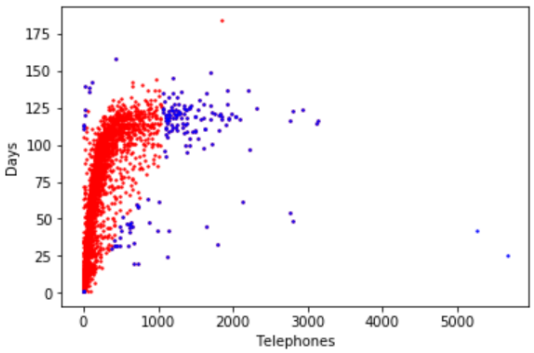

Figure 30-Anomalies detected by DBSCAN

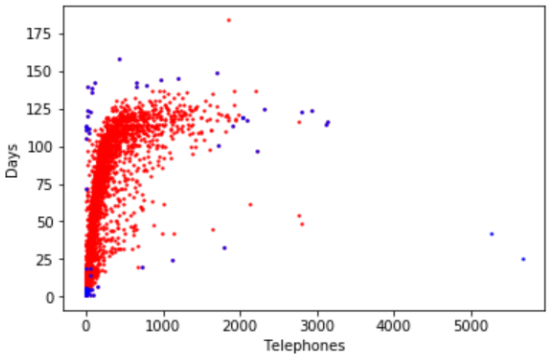

Figure 31-Anomalies detected by Isolation Forest

Table 12-Comparison of the results of the different algorithms in TelDay perspective

Algorithm	PercentConfirmed	RatioPercentAnomalies
Isolation Forest	5.825	0.090
K-means	5.825	0.037
DBSCAN	10.679	0.065
Elliptic Envelope	1.942	0.029

In the TelDay perspective all algorithms seem to behave very well, the only difference comes in quantities of anomalies detected, being DBSCAN the one that detects the most and Elliptic Envelope the one that detects the least (see figures 28 to 31). As shown in

Table *12*, DBSCAN, once again, has the highest coverage over the confirmed anomalies but Isolation Forest has a much higher ratio between the number of anomalies and the percentage coverage, this means that even though DBSCAN detects more anomalies, Isolation Forest anomalies could be more pertinent.

Perspective Ratios

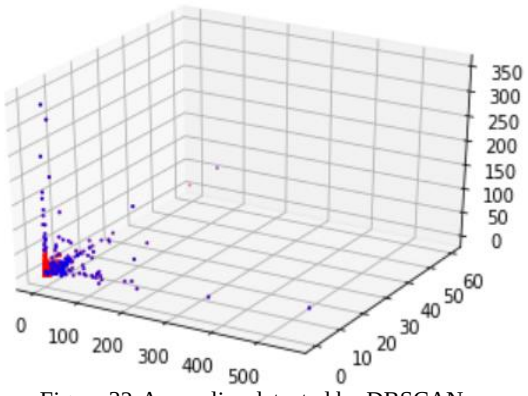

Figure 32-Anomalies detected by DBSCAN

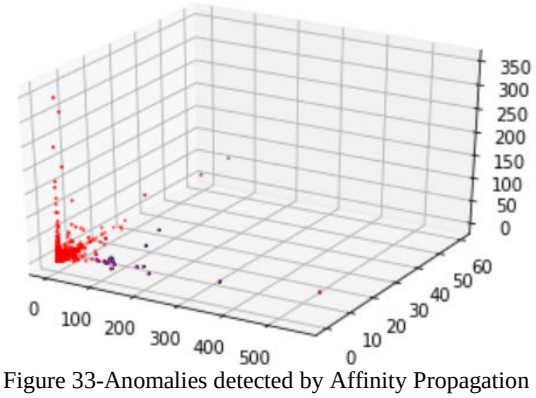

Figure 33-Anomalies detected by Affinity Propagation

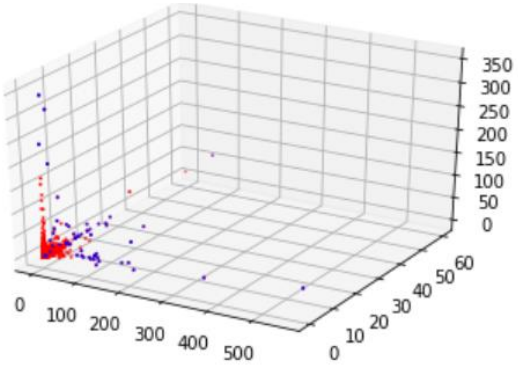

Figure 34-Anomalies detected by Elliptic Envelope

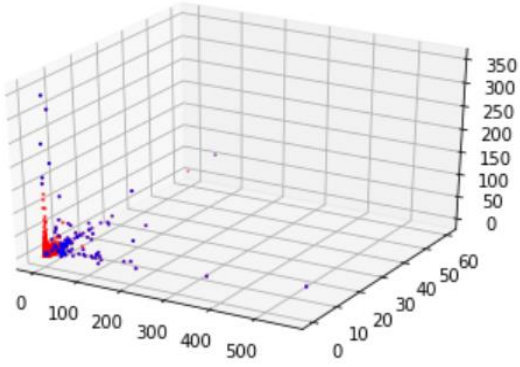

Figure 35-Anomalies detected by K-means

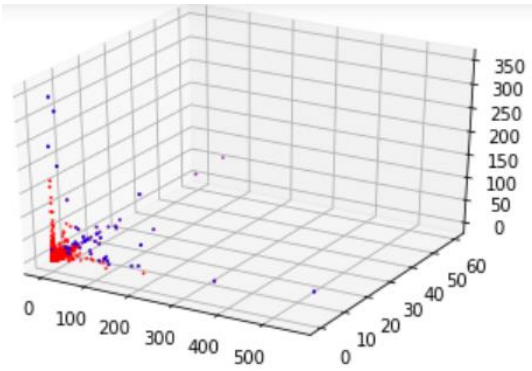

Figure 36-Anomalies detected by Isolation Forest

X-Axis: Access/Tel	Y-Axis: Access/Tel/Days	Z-Axis: Access/Days

Table 13-Comparison of the results of the different algorithms in Ratios perspective

Algorithm	PercentConfirmed	RatioPercentAnomalies
Isolation Forest	12.621	0.200
K-means	20.388	0.143
DBSCAN	24.272	0.145
Affinity Propagation	13.592	0.468
Elliptic Envelope	12.621	0.178

K-means, Elliptic Envelope, and Isolation Forest seem the most appropriate algorithms for this perspective (figures 32 to 36). Affinity propagation seems to detect right but very few anomalies. DBSCAN detects too many, it detects the important ones but probably it also detects some that are not that reliable. DBSCAN has the highest confirmed anomalies coverage but it has a low ratio between coverage percentage and number of anomalies, this is due to the fact that it detects so many anomalies. Affinity propagation has a very high ratio due to the opposite, it detects very few anomalies. Isolation Forest seems to have the best balance.

Perspective All

Since it is not possible to visualize all seven features, PCA was applied, so the results are harder to interpret. PCA is a dimensionality reduction algorithm that uses linear algebra to transform the dataset into a "compressed" form. A property of PCA is that we may choose the number of dimensions or principal components in the transformed result. This allows to simplify work when dealing with high dimensionality data. To apply PCA the data had to be normalized before.

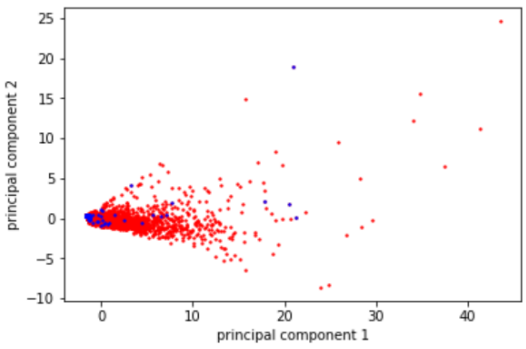

Figure 37-Anomalies detected by DBSCAN

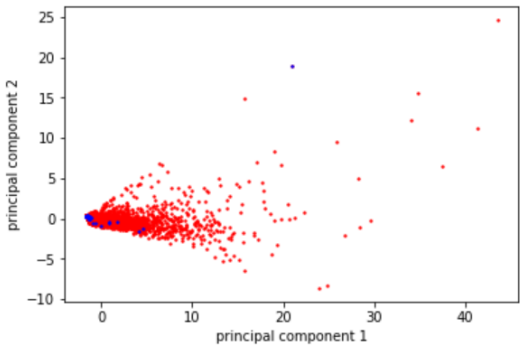

Figure 38-Anomalies detected by Affinity Propagation

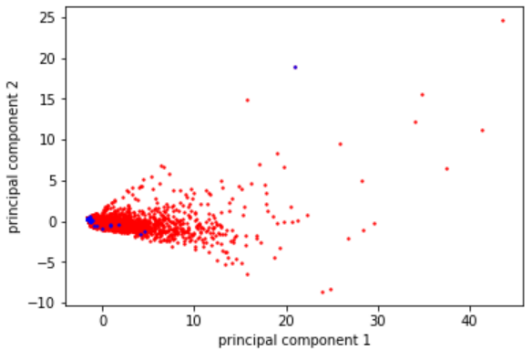

Figure 39-Anomalies detected by K-means

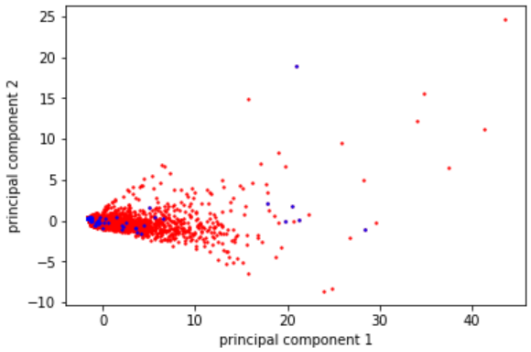

Figure 40-Anomalies detected by Elliptic Envelope

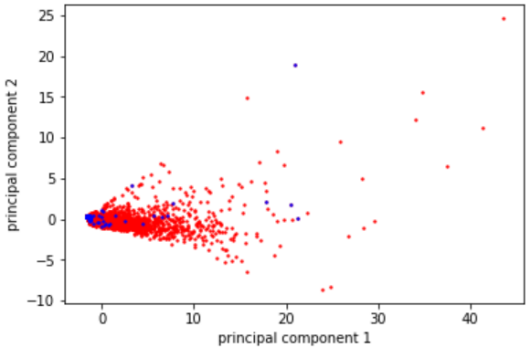

Figure 41-Anomalies detected by Isolation Forest

Table 14-Comparison of the results of the different algorithms in All perspective

Algorithm	PercentConfirmed	RatioPercentAnomalies
Isolation Forest	11.650	0.177
K-means	13.592	0.453
DBSCAN	27.184	0.165
Affinity Propagation	13.592	0.082
Elliptic Envelope	7.767	0.118

Looking at the plots in figures 37 to 41, it seems that Elliptic Envelope may be the most reliable algorithm for this perspective, but certainty will only be acquired when looking at the final results. When looking at Table 14, K-means seems to have the most reliable results, it has a high confirmed anomalies coverage even though it detects fewer anomalies. Once again, DBSCAN has the highest confirmed anomalies coverage.

Overall Results

Table 15-Number of anomalies obtained for each algorithm applied to each perspective in users approach

Number of anomalies	AccessTel	AccessDay	TelDay	Ratios	All
Isolation Forest	63	56	66	64	65
K-means	134	15	159	138	30
DBSCAN	198	179	164	167	165
Affinity Propagation	310	5	---	29	28
Elliptic Envelope	66	66	66	66	66

The final results differ a lot depending on the ensemble approach chosen. As demonstrated by the resuls in Table 16 and Table 17, wich are a summary of part of the results presented in Appendix L – Final Results. With a UnionUnion ensemble execution, a very high number of anomalies was found, that covered many of the anomalies already confirmed as illicit, and there is a high number of anomalies in each of the classification types. Even though this could seem like good results, the high percentage cover of the confirmed anomalies and the abundance in the different anomaly types happens only because of the high amount of anomalies obtained. If the UnionUnion approach was chosen there would be a high probability of false positives.

With an IntersectionUnion approach, we deal with the opposite, since to be considered an anomaly, the user would have to be detected by all algorithms, we are leaving behind a lot of possible illicit situations. A very low number of anomalies is obtained and only the most obvious classification types are being detected. This situation completely avoids false positives but in exchange, it does not detect all the relevant anomalies neither has a wide variety of anomaly types. If the IntersectUnion approach was chosen there would be a high probability of false negatives.

Finally, with the VotingUnion, we have a reasonable number of anomalies that covers, a good percentage of the confirmed anomalies and all the distinct classifications. Since each anomaly went to a process of voting, in which only the ones that were detected by the majority of algorithms were considered, the probability of existence of false positives is being strongly reduced.

Table 16-Comparison of results of the different ensemble methods in users approach

	UnionUnion	IntersectUnion	VotingUnion
Number of Anomalies obtained	556	59	219
Percentage coverage of confirmed anomalies	69%	33%	53%

Table 17- Anomaly types detected in each ensemble method, in users approach

Number of Anomalies	T. 1	T. 2	T. 3	T. 4	T. 5	T. 6	T. 7	T. 8	T. 9	T. 10
UnionUnion	357	32	12	18	30	33	4	39	88	150
IntersectUnion	38	9	4	2	3	6	0	0	0	35
VotingUnion	142	16	8	6	11	17	2	21	32	114

After choosing the Voting Union as the best ensemble method and observing its final results (in Appendix L), it was interesting to see which algorithms contributed the most. Table 18 shows us how many of the final anomalies we detected by each algorithm.

Table 18- Comparison of different algorithms detection of final anomalies in users approach

Algorithms	Final Anomalies detected
Isolation Forest	123
K-means	207
DBSCAN	219
Affinity Propagation	179
Elliptic Envelope	156

4.2 Telephone Approach

Once again, the rules created to choose the best parameterization were followed and the best parameters for each <algorithm, feature perspective> pair were obtained:

Table 19-Telephones approach parameterization results

	AccessActor	AccessDay	DayActor	Ratios	All
Isolation Forest (percentage of contamination)	0.001	0.001	0.001	0.01	0.001
K-means (number of clusters)	5	5	2	5	6
Elliptic Envelope (percentage of contamination)	0.001	0.001	0.001	0.05	0.001

Since we're dealing with a much larger dataset and the rules for the telephones approach had to allow more anomalies, as expected, the amount of anomalies each algorithm detects for each perspective is higher.

Just like the score tables, when an algorithm runs in the telephone approach the results are converted to the user approach, this means that the results presented refer to the users considered anomalies due to being responsible for the telephones that were considered anomalies by the application of the algorithms on the telephones approach dataset.

Like in the users approach, to better understand the users detected by each algorithm, plots based on each perspective features were created. Once again, all the plots had the already known test users removed for better visualization and interpretation. Following the same idea, the red points correspond to all the users existent and the blue points to the anomalies detected.

Perspective AccessDay

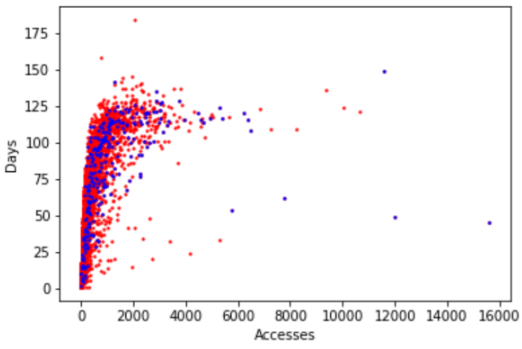

Figure 42-Anomalies detected by Elliptic Envelope

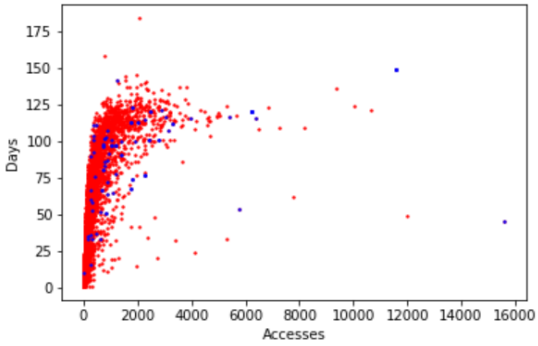

Figure 43-Anomalies detected by K-means

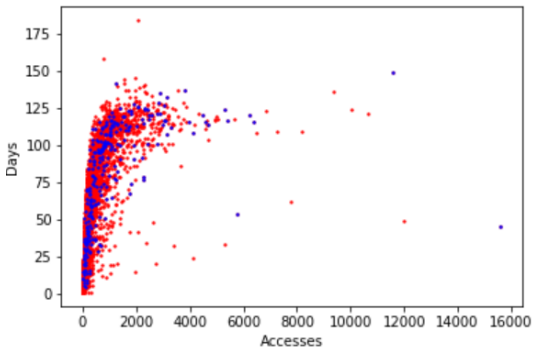

Figure 44-Anomalies detected by Isolation Forest

Table 20-Comparison of the results of the different algorithms in AccessDay perspective

Algorithm	PercentConfirmed	RatioPercentAnomalies
Isolation Forest	26.213	0.067
K-means	31.067	0.194
Elliptic Envelope	26.213	0.067

Perspective AccessActor

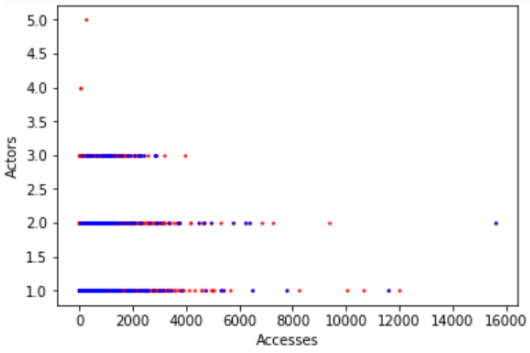

Figure 45-Anomalies detected by Isolation Fores

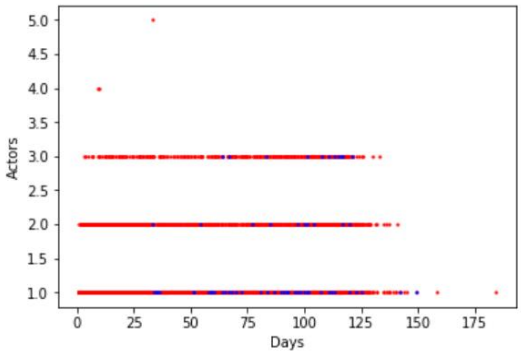

Figure 46-Anomalies detected by K-means

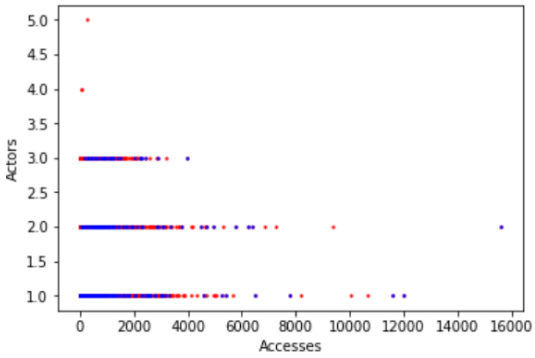

Figure 47-Anomalies detected by Elliptic Envelope

Table 21-Comparison of the results of the different algorithms in AccessActor perspective

Algorithm	PercentConfirmed	RatioPercentAnomalies
Isolation Forest	26.213	0.046
K-means	20.388	0.184
Elliptic Envelope	26.213	0.066

Perspective DayActor

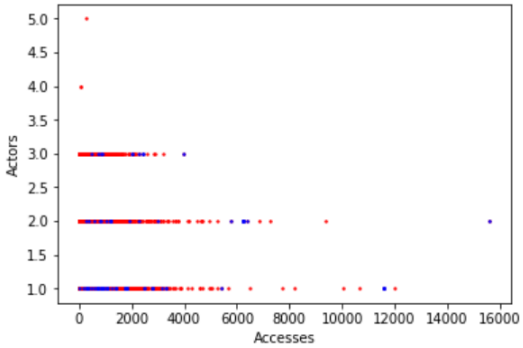

Figure 48-Anomalies detected by K-means

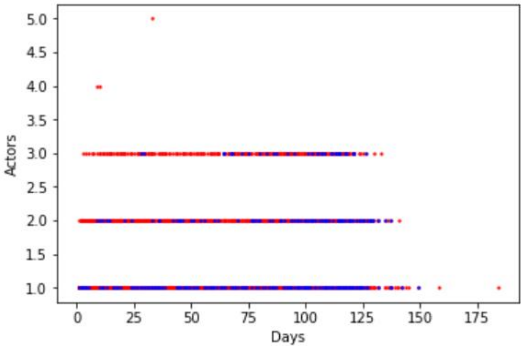

Figure 49-Anomalies detected by Isolation Forest

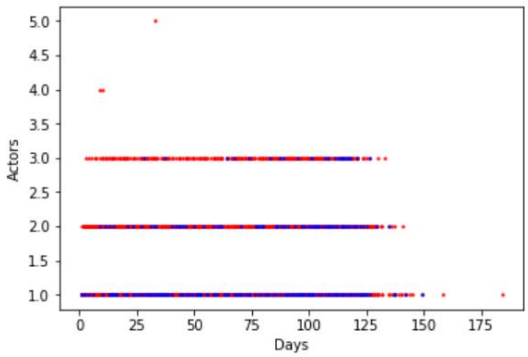

Figure 50-Anomalies detected by Elliptic Envelope

Table 22-Comparison of the results of the different algorithms in DayActor perspective

Algorithm	PercentConfirmed	RatioPercentAnomalies
Isolation Forest	27.184	0.065
K-means	18.446	0.200
Elliptic Envelope	29.126	0.077

Perspective Ratios

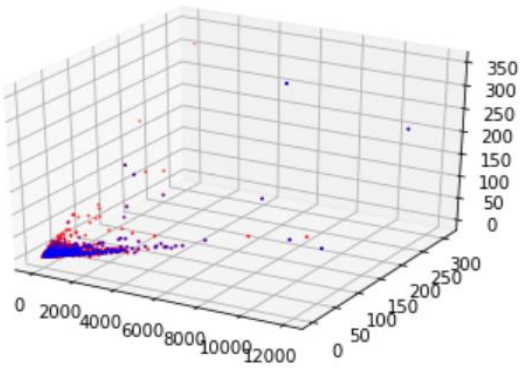

Figure 51-Anomalies detected by Elliptic Envelope

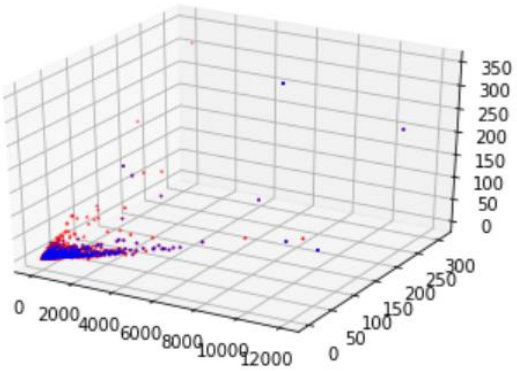

Figure 52-Anomalies detected by Isolation Forest

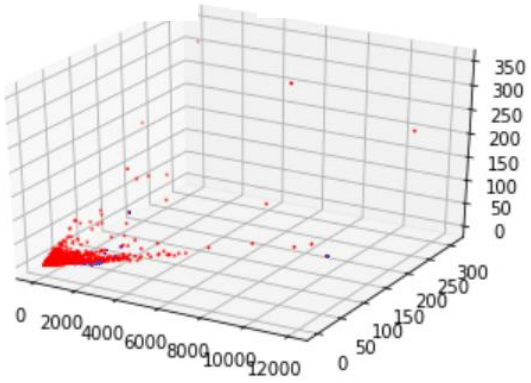

Figure 53-Anomalies detected by K-means

X-Axis: Access/Actor	Y-Axis: Access/Actor/Days	Z-Axis: Access/Days

Table 23-Comparison of the results of the different algorithms in Ratios perspective

Algorithm	PercentConfirmed	RatioPercentAnomalies
Isolation Forest	37.864	0.019
K-means	17.476	0.196
Elliptic Envelope	37.864	0.029

Perspective All

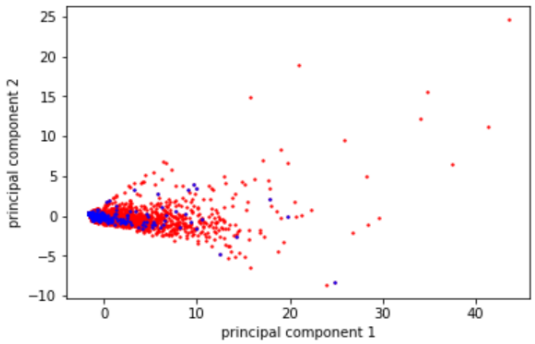

Figure 54-Anomalies detected by Elliptic Envelope

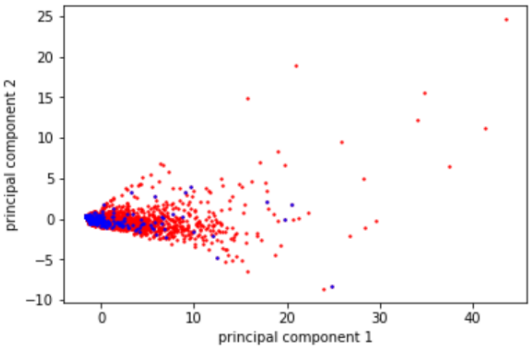

Figure 55-Anomalies detected by Isolation Forest

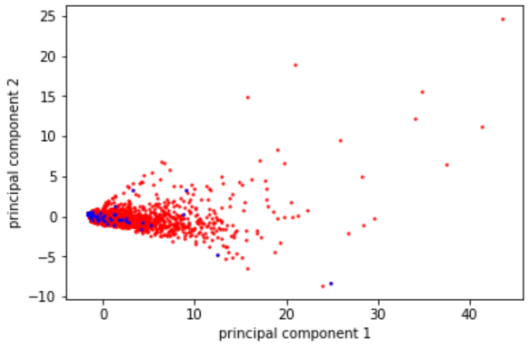

Figure 56-Anomalies detected by K-means

Table 24-Comparison of the results of the different algorithms in All perspective

Algorithm	PercentConfirmed	RatioPercentAnomalies
Isolation Forest	35.922	0.096
K-means	29.126	0.310
Elliptic Envelope	56.311	0.157

The results are visibly less reliable than in the users approach. The decisions to ensemble the results have to be much stricter to avoid false positives. It is necessary to have in consideration that the idea with the telephones approach was to detect the anomalies not detected by the users approach, so it was expected that the anomalies detected would not be in the same obvious area. The confirmed anomalies percentage is higher than in the users approach but that is due to the number of anomalies detected in the telephones approach being much higher.

Overall Results

Table 25-Number of anomalies obtained for each algorithm applied to each perspective in telephones approach

Number of anomalies	AccessActor	AccessDay	DayActor	Ratios	All
Isolation Forest	432	373	431	1272	346
K-means	142	160	92	89	94
Elliptic Envelope	393	393	377	1218	339

When choosing the best ensemble method for this approach we had to take into account that we're dealing with a higher number of anomalies, and since false positives remains a concern the choice must be stricter.

The UnionUnion ensemble method has 1547 anomalies, too many to have control over how many false positives may be in the final results, and, even though there are so many anomalies it only covers 43% of the confirmed anomalies, as shown in Table *26*.

The VotingUnion method also produces many anomalies and the percentage coverage of the confirmed anomalies does not compensate. This percentage could be ignored if we had confidence in the results, but in the users approach an anomaly being accepted meant it had appeared in three of the algorithms. Since in this approach only three algorithms were applied, the majority of the algorithms is just two, so for an anomaly to be accepted it only had to be detected by two of the algorithms. This does no

t give enough security in terms of confidence in the results.

The IntersectUnion method has much fewer anomalies and just like in the users approach all accepted anomalies have to be detected by all the algorithms applied.

Table 26-Comparison of results of the different ensemble methods in telephones approach

	UnionUnion	IntersectUnion	VotingUnion
Number of Anomalies obtained	1547	87	1087
Percentage coverage of confirmed anomalies	43%	22%	37%

Table 27-Anomaly types detected in each ensemble method, in telephones approach

Number of Anomalies	T. 1	T. 2	T. 3	T. 4	T. 5	T. 6	T. 7	T. 8	T. 9	T. 10
UnionUnion	1153	67	8	49	84	18	0	6	47	87
IntersectUnion	57	26	8	0	2	1	0	0	0	9
VotingUnion	866	49	8	33	54	14	0	2	15	75

In Table 27 it is possible to see that the UnionUnion detects the biggest number of anomalies in each type, it detects 1153 anomalies for type 1, which can not be considered reliable. VotingUnion also encounters the same problem, detecting 866 anomalies for the first type. Since it was established that in telephones approach we had to be stricter with the ensemble, IntersectUnion shows the most reliable results.

Since the IntersectUnion was considered the best ensemble method all algorithms contributed the same, since an anomaly had to be detected by all of them. Table 28 confirms that.

Table 28-Comparison of different algorithms detection of final anomalies in telephones approach

Algorithms	Final Anomalies detected
Isolation Forest	87
K-means	87
Elliptic Envelope	87

4.3 Final Results

After executing both approaches the results were aggregated by union, meaning that if an identity was considered an anomaly at the end of an approach, it was also considered an anomaly in the final results.

Tables 29, 30 and 31, show the number of anomalies from each anomaly type obtained, the number of anomalies obtained in both approaches and the final percentage coverage of confirmed anomalies.

All types of anomalies are detected, being type 1 and type 10 the most common ones. Users approach detected more anomalies, which was expected since it was the main approach. The telephones approach was executed to complement the first one. Nineteen

anomalies were detected by both approaches and the other 268 by only one of them (219 by the users approach and 87 by the second).

Table 29-Final anomaly types detected

	T. 1	T. 2	T. 3	T. 4	T. 5	T. 6	T. 7	T. 8	T. 9	T. 10
Number of Anomalies	187	35	11	6	13	18	2	21	32	114

Table 30-Total anomalies obtained

	Final number of anomalies	Anomalies from users approach	Anomalies from telephones approach	Anomalies detected in both approaches
Number of Anomalies	287	219	87	19

The final results cover 64% of the list with confirmed anomalies, this means that the final results detected 65 of the 103 confirmed anomalies. The results from the *Cuscos* project covered 58%, which means that the project had detected 59 of the confirmed anomalies. Since 287 anomalies were detected, there are 222 to confirm. In the Cuscos project situation, because 1800 anomalies were detected, there would be 1741 anomalies to confirm.

Table 31-Percentage coverage of confirmed anomalies

Final results percentage coverage of confirmed anomalies	*Cuscos* project results percentage coverage of confirmed anomalies
64%	58%

Finally, to illustrate the results, plots with the final anomalies in all the users perspectives were created, Figures 60 to 63. As referred before, the red points are all the users and the blue points are the anomalies detected.

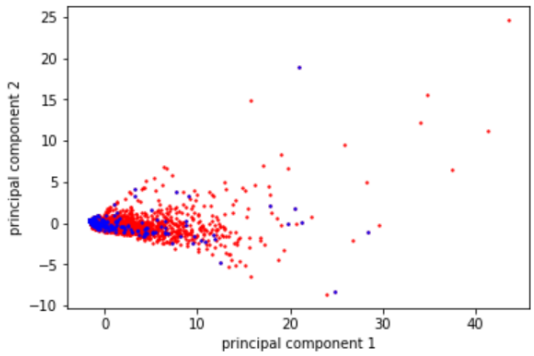

Figure 57-Final anomalies detected viewed in All perspective

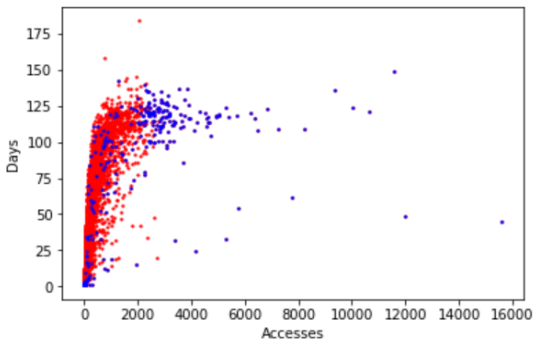

Figure 58-Final anomalies detected viewed in AccessDay perspective

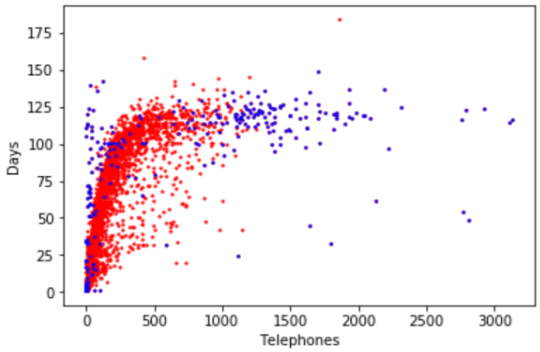

Figure 59-Final anomalies detected viewed in TelDays perspective

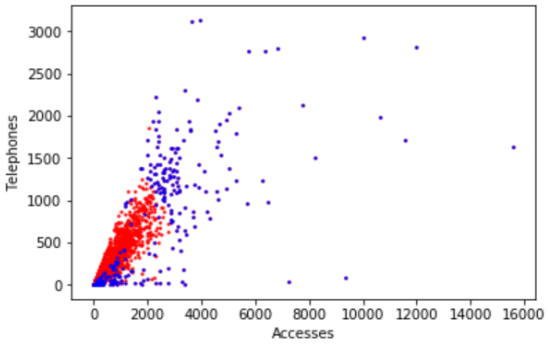

Figure 60-Final anomalies detected viewed in AccessTel perspective

As it is possible to observe almost all the anomalies detected are in the less dense parts of the plots, which is great news since it means that they are the users that are not in the pattern. A few blue points appear in the dense parts, this could mean the existence of a false positive or that in a determined perspective that point follows the pattern because the feature that made it an anomaly is not present.

Also to compare the results with the previous *Cuscos* project, the same plots were created but with the results of that project, as it is possible to see in Figures 64 to 67.

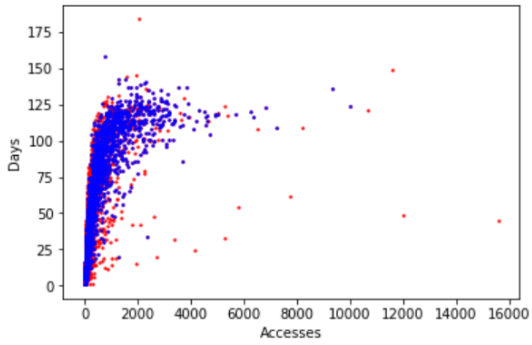

Figure 61-Anomalies detected in Cuscos project viewed in AccessDays perspective

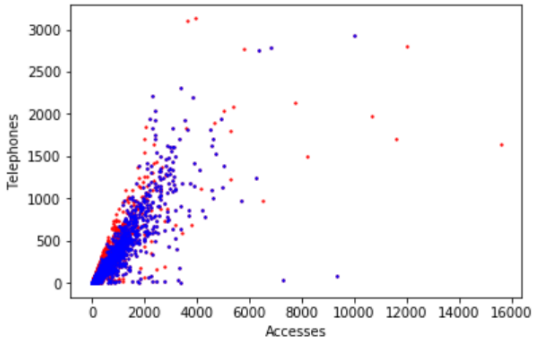

Figure 62-Anomalies detected in Cuscos project viewed in AccessTel perspective

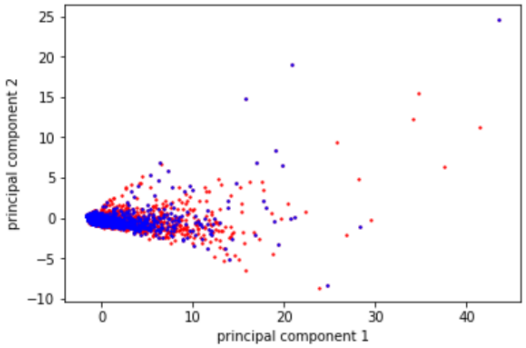

Figure 63-Anomalies detected in Cuscos project viewed in All perspective

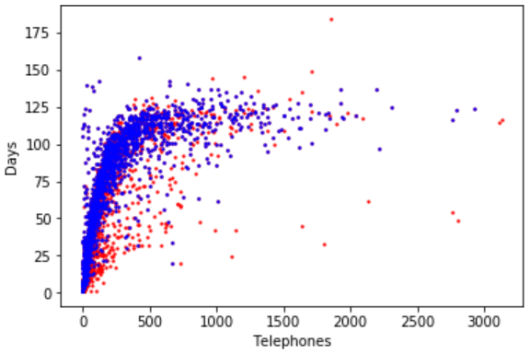

Figure 64-Anomalies detected in Cuscos project viewed in TelDays perspective

This allows to see that the *Cuscos* project results must have too many false positives and that this problem was reduced in this project. It is also possible to see that *Cuscos* did not detect many of the users that are visibly off the pattern in the plots.

Example

To have a better idea of how this project results will be useful, an example will be shown below. Table 32 shows 4 anomalies from the final results.

Table 32- Example of 4 detected identities in final results

1238	type4;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
188612	type10; type4;	All
3018	type2;	All
32614	type1;	All

These four anomalies would be sent to an investigation process so that the necessary disciplinary or legal actions can be pursued. As mentioned before, the type 4 anomalies would be the priority because it represents a more specific situation that is considered very relevant. There is an anomaly from type 4 and 10, this means that even though it is representing the same situation, its behavior numbers are so high that is very likely to be a test account. Both anomalies from type 1 and 2 would be investigated after. Tables 33 to 36 present the information related to each of the anomalies listed in Table 32.

Table 33- Anomaly description

Id	Days	Phones	Objects	Accesses
1238	1	102	1	309

As expected, the behavior in Table 33 seems to describe a situation of massive extraction of information of various telephones number in a short period. This type of anomaly is very important as it could mean that someone is performing a fast search to raise clients for another telecommunications company.

Table 34- Anomaly description

Id	Days	Phones	Objects	Accesses
188612	25	5662	1	37530

The behavior described in Table 34 confirms the possibility of this being a test account, and after investigating the situation it was possible to conclude that this is, in fact, a known test account. This is still a relevant result, as it would be a bad sign if these accounts went unnoticed.

Table 35- Anomaly description

Id	Days	Phones	Objects	Accesses
3018	112	38	1	3285

Table 36- Anomaly description

Id	Days	Phones	Objects	Accesses
32614	108	980	3	6492

Both the identities ‘3018’ and ‘32614’ behavior, described in Tables 35 and 36, also fits the type of anomaly they are classified as. Both could be justified as being someone from the financial department or some other acceptable reason, but could also be an illicit situation not being acknowledged, and for that reason, without priority, if there was not an acceptable justification, both would go through the investigation process.

This example used just 4 of the 287 final anomalies just to give an idea of the results being put in practice and what the final goal was.

Chapter 5 Conclusions and Future Work

The goal of this project was to try a different approach from the *Cuscos* project to discover anomalies, find a way to label them in types, reduce the number of false positives and find new anomalies. We were able to identify many of the anomalies detected in the *Cuscos* project, some entirely new, differentiate the different types and greatly reduce the number of false positives. We believe that it was successful.

In the users approach the best ensemble method, was the VotingUnion, since it covers more than half of the confirmed anomalies but also obtains others that were not caught before. It detects anomalies from the 10 different types but it does not have an unrealistic number of anomalies. Each anomaly has to be detected by the majority of the algorithms to be accepted, which decreases the probability of false positives. In terms of algorithms, from the final anomalies detected, DBSCAN is the one that detected the majority of them, detecting all 219 of them, in second place comes Kmeans, detecting 207. Finally, Isolation Forest was the one that detected the least amount from the anomalies considered relevant after the VotingUnion Ensemble, with 123.

In the telephones approach the best ensemble method was the IntersectUnion, since the numbers of anomalies were much higher and fewer algorithms were applied. To reduce false positives an anomaly had to be detected by all algorithms (just three were applied, so it's the same amount of algorithms as in the ensemble method chosen in the users approach). It covers only 22% of the confirmed anomalies but since each anomaly is supported by three algorithms it is acceptable. Since each anomaly had to be detected by the three algorithms to be accepted, all algorithms contributed the same for the final results of the telephones approach.

Results differing from the Cuscos project is something already expected since the Cuscos project was towards <user, telephone number> pairs and this project was towards the users and telephone numbers individually. Cuscos project detected about 1800 anomalies and it covered about 58% of the list of confirmed anomalies. This project obtained 287 anomalies and it covered about 64% of the confirmed anomalies. It was possible to label each anomaly to a certain behavior, new anomalies were detected and the probability of false positives is much lower than in the previous work. This shows the objectives were achieved.

For future work, it would be interesting to use time series to follow a user behavior [15]. Applying machine learning in time series could bring new relevant results. For example, the amount of times a user makes accesses could be normal compared with the number of days and amount of telephone numbers, but with time series, it would be

possible to detect if he had not ever made accesses and then, suddenly, made accesses fifteen days in a row. This example is not possible to detect with the actual methodology since, even though we have the number of distinct days, when a user accesses a telephone number in many different days we do not know if these days are close to each other or far apart.

Another obvious future work will be to use DBSCAN for the telephone numbers approach. The algorithm had a great contribution for the users approach so in the future it would be worth to use it for better results. The affinity propagation, on the other hand, does not seem as helpful, so in this situation, the algorithm results do not compensate its complexity.

References

1. Derel Lin. “Anomalous User Activity Detection in Enterprise Multi-Source Logs”. Conference paper, 2017.
2. J. R. Quinlan. “Induction of Decision Trees”, 1986 Kluwer Academic Publishers, Boston;
3. A. D. Kent, L. M. Liebrock, and J. C. Neil, “Authentication graphs: Analyzing user behavior within an enterprise network,” Computers & Security, vol. 48, 2015.
4. P. Thompson, “Weak models for insider threat detection,” in Proc. of SPIE Vol, vol. 5403, 2004.
5. M. Ahmed et al. / Journal of Network and Computer Applications, “A survey of network anomaly detection techniques”, 2015.
6. Pavel Berkhin, “Survey of Clustering Data Mining Techniques”, Accrue Software, Inc.
7. Monowar H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, “Network Anomaly Detection: Methods, Systems and Tools”, IEEE COMMUNICATIONS SURVEYS & TUTORIALS, VOL. 16, NO. 1, FIRST QUARTER 2014.
8. Henrik Brink, Joseph W. Richards, Mark Fetherolf. “REAL WORLD Machine Learning”, MANNING.
9. Jason Brownlee. “Machine Learning Mastery with Python – Understand Your Data, Create Accurate Models and Work Projects End-To-End”, 2016.
10. Brendan J. Frey, et al. “Clustering by Passing Messages Between Data Points”, Science 315, 972 (2007);
11. Tran Manh Thang and Juntae Kim, “The Anomaly Detection by Using DBSCAN Clustering with Multiple Parameters”;
12. Mattia Antonini and Massimo Vecchio, “Smart Audio Sensors in the Internet of Things Edge for Anomaly Detection”, 2018.
13. M. C. Cowgill, R. J. Harvey and L. T. Watson, “A Genetic Algorithm Approach to Cluster Analysis”, 1998
14. R.B. Calinski and J. Harabasz, A dendrite method for cluster analysis, Communications in Statistics 3, 1-27, (1974)

15. Terran Lane and Carla E. Brodley, “An Application of Machine Learning to Anomaly Detection”, 1997
16. https://www.researchgate.net/figure/A-The-VRC-and-Silhouette-Width-values-of-five-clustering-methods-for-real-gene-expression. Viewed May 5, 2019
17. https://www.docker.com/products/docker-enterprise. Viewed August 19, 2019
18. https://opensource.com/resources/what-docker. Viewed August 19, 2019
19. https://hub.docker.com/r/jupyter/scipy-notebook/. Viewed August 19, 2019
20. https://www.elastic.co/products/logstash. Viewed August 19, 2019
21. https://medium.com/@harshityadav95/getting-started-with-logstash-96f5f1000cb6. Viewed August 19, 2019

Appendix A

To facilitate the visualization of the tables in appendix some abbreviations had to be used, Table 37 explains the meaning of each abbreviation.

Table 37- Legend for abbreviations used

Legend	
contamination	Cont
convergence	Conv
mindist	MD
minPontos	MP
NumberClusters	NC
NumberAnomalies	NA
SilhouetteScore	SS
VRC	VRC
PercentConfirmed	PC
RatioPercentAnomalies	RPA
Elliptic Envelope	EE
Isolation Forest	IF
Affinity Propagation	AffProp

Appendix B – Users approach: Score Tables for perspective Ratios

Table 38- Score table for Isolation Forest

Cont	NA	SS	VRC	PC	RPA
0.001	7	0,997574615	1281,68927	6,699029126	0,957004161
0.005	32,6	0,988623399	252,2545191	25,24271845	0,775228008
0.01	64,2	0,977934037	126,2927037	34,75728155	0,541365193
0.05	323,5	0,894407194	25,45591003	55,04854369	0,170170826
0.08	510,9	0,836592675	15,96084164	66,31067961	0,130381388
0.1	653,4	0,790671737	12,00248914	65,33980583	0,100047826
0.15	938	0,703370178	8,156368233	70,29126214	0,075469982
0.2	1264,8	0,60555541	5,669708715	70,38834951	0,055743358
0.25	1411,6	0,564734149	5,163502915	74,27184466	0,054369201
0.3	1857,3	0,433729638	3,450894162	76,01941748	0,041160652
0.35	2100,6	0,36314238	2,959352242	78,05825243	0,037598594
0.4	2412,1	0,279728771	2,50182521	79,51456311	0,03418095

Table 39- Score table for K-means

NC	NA	SS	VRC	PC	RPA
2	1	0,999741294	1401474	0,970873786	0,970873786
3	2	0,998376826	8768785	1,941747573	0,970873786
4	6	0,985976315	9166676	5,825242718	0,970873786
5	29	0,968859585	11720905	28,15533981	0,970873786
6	31	0,962453623	13733381	30,09708738	0,970873786
7	30	0,959377864	15505791	29,12621359	0,970873786
8	145	0,872299129	16662533	50,48543689	0,348175427
9	140	0,866437913	18192786	48,54368932	0,346740638
10	140	0,86656121	18156785	48,54368932	0,346740638
11	177	0,844434846	18692557	49,51456311	0,279743294
12	221	0,825941625	19532039	49,51456311	0,224047797
15	175	0,488163751	24584792	57,2815534	0,327323162
16	225	0,463755386	26883833	53,39805825	0,237324703
18	265	0,504574157	30627562	60,19417476	0,227147829
20	360	0,496647292	33127428	67,96116505	0,188781014
25	284	0,409405165	40598867	66,01941748	0,232462738
30	395	0,406316096	48333966	68,93203883	0,174511491
35	426	0,433647078	52040299	71,84466019	0,168649437
40	608	0,38153502	55559485	76,69902913	0,126149719
45	589	0,389661748	57654817	77,66990291	0,131867407
50	620	0,401519927	62326120	74,75728155	0,120576261
60	881	0,385893245	70024084	76,69902913	0,087059057
70	1118	0,345004674	76164258	78,6407767	0,070340587
100	2009	0,370630262	93312387	81,55339806	0,040594026
150	3174	0,373122315	1,19E+08	85,4368932	0,026917736

Table 40- Score table for Elliptic Envelope

Cont	NA	SS	VRC	PC	RPA
0.001	7	0,997593637	1300,074248	6,796116505	0,970873786
0.005	30,8	0,989426631	267,7259665	21,3592233	0,695221778
0.01	66	0,977873663	126,1001423	40,77669903	0,617828773
0.05	320	0,897250104	25,82350509	58,25242718	0,182038835
0.08	528	0,83258087	15,15753089	58,25242718	0,110326567
0.1	660	0,792833292	11,96672822	62,13592233	0,094145337
0.15	834	0,74072632	9,236224263	65,04854369	0,077995856
0.2	1319	0,596911043	5,261211904	65,63106796	0,049758202
0.25	1649	0,504745552	3,997661898	72,81553398	0,044157389
0.3	1971,8	0,415310951	3,133966056	73,78640777	0,037432537
0.35	2308	0,322944265	2,501142339	78,6407767	0,034073127
0.4	2638	0,232623921	2,031091116	80,58252427	0,030546825

Table 41-Score Table for Affinity Propagation

Conv	NC	NA	SS	VRC	PC	RPA
1	4984	6110	0,000843338	136,9274	89,32038835	0,014618721
5	4977	6110	0,000620022	203,6966	89,32038835	0,014618721
10	6	29	0,967615559	11679513	28,15533981	0,970873786
50	6	29	0,967615559	11679513	28,15533981	0,970873786
80	6	29	0,967615559	11679513	28,15533981	0,970873786
100	6	29	0,967615559	11679513	28,15533981	0,970873786
150	6	29	0,967615559	11679513	28,15533981	0,970873786
200	6	29	0,967615559	11679513	28,15533981	0,970873786
250	6	29	0,967615559	11679513	28,15533981	0,970873786
300	6	29	0,967615559	11679513	28,15533981	0,970873786
350	6	29	0,967615559	11679513	28,15533981	0,970873786
400	6	29	0,967615559	11679513	28,15533981	0,970873786
450	6	29	0,967615559	11679513	28,15533981	0,970873786
500	6	29	0,967615559	11679513	28,15533981	0,970873786
550	6	29	0,967615559	11679513	28,15533981	0,970873786
600	6	29	0,967615559	11679513	28,15533981	0,970873786
700	6	29	0,967615559	11679513	28,15533981	0,970873786
750	6	29	0,967615559	11679513	28,15533981	0,970873786
800	6	29	0,967615559	11679513	28,15533981	0,970873786
850	6	29	0,967615559	11679513	28,15533981	0,970873786
900	6	29	0,967615559	11679513	28,15533981	0,970873786
950	6	29	0,967615559	11679513	28,15533981	0,970873786
1000	6	29	0,967615559	11679513	28,15533981	0,970873786

For the DBSCAN score table, since there are 725 combinations of parameters, only an excerpt with the relevant zone that has the best parameterization is shown.

Table 42- Score Table for DBSCAN

MD	MP	NC	NA	SS	VRC	PC	RPA
3	45	2	167	0,945615826	150,6231817	62,13592233	0,372071391
3	50	2	170	0,944664115	49,73349892	62,13592233	0,365505425
3	60	2	185	0,939951726	45,68667238	62,13592233	0,33586985
3	70	2	189	0,938692345	44,71603417	62,13592233	0,328761494
3	100	2	194	0,937109202	43,56493449	62,13592233	0,320288259
3	40	2	160	0,947831964	52,85004159	61,16504854	0,382281553
3	35	2	153	0,950029135	55,28344079	60,19417476	0,393425979
3	15	2	129	0,957673043	65,53274808	59,22330097	0,459095356
3	16	2	129	0,957673043	65,53274808	59,22330097	0,459095356
3	18	2	133	0,956386169	63,57735932	59,22330097	0,445287977
3	20	2	135	0,955747878	62,63359155	59,22330097	0,438691118
3	25	2	145	0,952585355	58,29811923	59,22330097	0,408436558
3	30	2	148	0,951626337	57,14134551	59,22330097	0,400157439
3	5	5	104	0,820849141	25,35254035	58,25242718	0,560119492
3	6	4	108	0,843479501	30,48199743	58,25242718	0,539374326
3	7	3	108	0,86099733	43,04156996	58,25242718	0,539374326
3	8	3	110	0,858277634	41,78007497	58,25242718	0,52956752
3	9	3	114	0,858200903	40,19887792	58,25242718	0,510986203
3	10	2	114	0,96248908	74,04898907	58,25242718	0,510986203
3	11	2	114	0,96248908	74,04898907	58,25242718	0,510986203
3	12	2	118	0,961203926	71,58301079	58,25242718	0,493664637
3	1	64	100	0,788099662	6044603,383	54,36893204	0,54368932
3	2	16	100	0,781916247	11,35322426	54,36893204	0,54368932
3	3	9	100	0,783754195	16,60893615	54,36893204	0,54368932
3	4	5	100	0,847052637	27,23485223	54,36893204	0,54368932
3	150	2	0	0,927423429	37,47635622	0	0

Appendix C – Users approach: Score Tables for perspective AccessDays

Table 43-Score Table for Isolation Forest

Cont	NA	SS	VRC	PC	RPA
0.001	6,2	0,993426016	5627,006	8,333333333	1,353174603
0.005	30,2	0,971874804	998,3187	13,75	0,45668052
0.01	63,7	0,944305806	491,97	20,41666667	0,319598453
0.05	325,1	0,788400191	136,5083	34,375	0,105626642
0.08	523,6	0,719090453	112,7538	43,75	0,083409138
0.1	657,7	0,676877727	102,7031	51,875	0,078869296
0.15	986,8	0,582712874	79,95788	56,04166667	0,056785221
0.2	1315,3	0,484376266	60,50006	67,08333333	0,051001524
0.25	1642,8	0,45906079	66,21446	70,41666667	0,042857695
0.3	1959,9	0,376070801	52,55676	72,70833333	0,03711409
0.35	2296,3	0,291275261	41,10482	69,79166667	0,030396585
0.4	2617,4	0,251877313	39,16419	77,91666667	0,029759021

Table 44- Score Table for K-means

NC	NA	SS	VRC	PC	RPA
2	2	0,997612677	64204,72	4,166666667	2,083333333
3	5	0,98694852	57734,98	8,333333333	1,666666667
4	5	0,986772766	74774,03	8,333333333	1,666666667
5	5	0,806309306	100235,2	8,333333333	1,666666667
6	5	0,80688946	122253	8,333333333	1,666666667
7	42	0,758349143	176422,9	18,75	0,446428571
8	18	0,704820017	255773,9	12,5	0,694444444
9	120	0,671093614	317392	27,08333333	0,225694444
10	40	0,647522565	399519,6	18,75	0,46875
11	146	0,6303512	456242,1	29,16666667	0,199771689
12	313	0,599827345	501775,6	43,75	0,139776358
15	314	0,564961006	636993,2	43,75	0,13933121
16	266	0,574041501	717945,6	43,75	0,164473684
18	347	0,540237979	836078,3	45,83333333	0,132084534
20	347	0,525563179	902227	45,83333333	0,132084534
25	589	0,514612647	1003416	60,41666667	0,102574986
30	796	0,495507702	1121649	66,66666667	0,083752094
35	1417	0,476575549	1187747	68,75	0,048517996
40	1708	0,469245864	1243482	75	0,043911007
45	1778	0,464780826	1273070	79,16666667	0,044525684
50	2084	0,459832206	1314545	79,16666667	0,037987844
60	2676	0,459759592	1382015	87,5	0,032698057
70	2725	0,441861558	1446496	89,58333333	0,032874618
100	3660	0,426372693	1643181	100	0,027322404
150	4101	0,418900422	1867915	100	0,024384297

Table 45- Score Table for Elliptic Envelope

Cont	NA	SS	VRC	PC	RPA
0.001	7	0,992900513	5194,038	8,333333333	1,19047619
0.005	33	0,971967211	1072,295	16,66666667	0,505050505
0.01	66	0,950783731	642,4933	25	0,378787879
0.05	330	0,85641949	274,0775	45,83333333	0,138888889
0.08	528	0,811600218	220,9407	58,33333333	0,110479798
0.1	660	0,785756526	198,7682	62,5	0,09469697
0.15	989	0,727626689	161,2772	70,83333333	0,071621166
0.2	1319	0,668847838	134,4018	72,91666667	0,055281779
0.25	1649	0,608097635	113,5769	75	0,04548211
0.3	1978	0,547924893	97,05747	79,16666667	0,040023593
0.35	2307,1	0,48744698	83,35425	85,41666667	0,037023392
0.4	2638	0,423379529	71,51479	87,5	0,033169067

Table 46-Score Table for Affinity Propagation

Conv	NC	NA	SS	VRC	PC	RPA
1	4163	5900	-2,4163E-05	0,433095949	100	0,016949153
5	4138	5900	0,001876707	13,52655719	100	0,016949153
10	6	20	0,929654326	78660,00393	12,5	0,625
50	6	20	0,929654326	78660,00393	12,5	0,625
80	6	20	0,929654326	78660,00393	12,5	0,625
100	6	20	0,929654326	78660,00393	12,5	0,625
150	6	20	0,929654326	78660,00393	12,5	0,625
200	6	20	0,929654326	78660,00393	12,5	0,625
250	6	20	0,929654326	78660,00393	12,5	0,625
300	6	20	0,929654326	78660,00393	12,5	0,625
350	6	20	0,929654326	78660,00393	12,5	0,625
400	6	20	0,929654326	78660,00393	12,5	0,625
450	6	20	0,929654326	78660,00393	12,5	0,625
500	6	20	0,929654326	78660,00393	12,5	0,625
550	6	20	0,929654326	78660,00393	12,5	0,625
600	6	20	0,929654326	78660,00393	12,5	0,625
700	6	20	0,929654326	78660,00393	12,5	0,625
750	6	20	0,929654326	78660,00393	12,5	0,625
800	6	20	0,929654326	78660,00393	12,5	0,625
850	6	20	0,929654326	78660,00393	12,5	0,625
900	6	20	0,929654326	78660,00393	12,5	0,625
950	6	20	0,929654326	78660,00393	12,5	0,625
1000	6	20	0,929654326	78660,00393	12,5	0,625

Table 47-Score Table for DBSCAN

MD	MP	NC	NA	SS	VRC	PC	RPA
1	1	22	23	0,795503135	20081,51	14,58333333	0,634057971
1	2	3	23	0,897004352	809,2305	14,58333333	0,634057971
1	3	3	23	0,897004352	809,2305	14,58333333	0,634057971
1	4	2	24	0,978485171	1383,991	14,58333333	0,607638889
1	5	2	25	0,977727461	1337,246	14,58333333	0,583333333
1	6	2	25	0,977727461	1337,246	14,58333333	0,583333333
1	7	3	38	0,859788022	654,0062	16,66666667	0,438596491
1	8	3	38	0,859788022	654,0062	16,66666667	0,438596491
1	9	3	38	0,858468246	633,2118	16,66666667	0,438596491
1	10	3	38	0,859491512	612,999	16,66666667	0,438596491
1	11	2	39	0,967707644	936,6201	16,66666667	0,427350427
1	12	2	39	0,967707644	936,6201	16,66666667	0,427350427
1	15	2	41	0,966367211	902,2831	16,66666667	0,406504065
1	16	2	42	0,965697823	886,0818	18,75	0,446428571
1	18	2	42	0,965697823	886,0818	18,75	0,446428571
1	20	2	47	0,962314327	811,6377	18,75	0,39893617
1	25	2	49	0,961029047	787,5716	18,75	0,382653061
1	30	2	54	0,957891356	735,2164	18,75	0,347222222
1	35	2	54	0,957891356	735,2164	13,125	0,243055556
1	40	2	54	0,957891356	735,2164	18,75	0,347222222
1	45	2	55	0,957270449	725,7435	18,75	0,340909091
1	50	2	62	0,953023296	667,8822	22,91666667	0,369623656
1	60	2	80	0,942879688	565,395	25	0,3125
1	70	2	100	0,932768775	494,6059	25	0,25
1	100	2	125	0,921369888	436,3876	27,08333333	0,216666667
1	150	2	179	0,900412695	363,1143	39,58333333	0,22113594

Appendix D – Users approach: Score Tables for perspective AccessTel

Table 48-Score Table for Isolation Forest

Cont	NA	SS	VRC	PC	RPA
0.001	6,9	0,991692339	4778,472	8,958333333	1,299603175
0.005	32,4	0,968174301	1024,292	18,125	0,559100328
0.01	63,7	0,943649013	590,2883	28,33333333	0,447311331
0.05	327	0,824964929	242,5474	48,125	0,147096727
0.08	525,3	0,770668001	202,4031	61,04166667	0,116214841
0.1	657,3	0,733370445	179,4333	60,20833333	0,091592037
0.15	979	0,692765793	174,4867	66,875	0,068295288
0.2	1264,1	0,612040804	136,7357	68,33333333	0,05410751
0.25	1627,6	0,5572962	122,5266	76,875	0,047218138
0.3	1963,3	0,440916457	88,48226	79,58333333	0,040533374
0.35	2208,2	0,414389964	83,84156	80,41666667	0,036425154
0.4	2495,5	0,392192445	88,37041	83,33333333	0,03358674

Table 49- Score Table for K-means

NC	NA	SS	VRC	PC	RPA
2	2	0,997213994	59019,11	4,166666667	2,083333333
3	5	0,984124215	47546,51	8,333333333	1,666666667
4	5	0,983944598	52269,62	8,333333333	1,666666667
5	5	0,826275567	80032,34	8,333333333	1,666666667
6	154	0,766460021	89194,62	20,83333333	0,135281385
7	134	0,771308865	120506,2	16,66666667	0,124378109
8	35	0,741628348	149773,2	14,58333333	0,416666667
9	135	0,702864793	172311,7	16,66666667	0,12345679
10	216	0,688182596	184493	20,83333333	0,096450617
11	143	0,658028911	195618,3	18,75	0,131118881
12	149	0,6572588	202812,1	20,83333333	0,139821029
15	253	0,598702555	220334	39,58333333	0,156455863
16	270	0,597389925	224408,1	39,58333333	0,146604938
18	273	0,600187227	229199,1	45,83333333	0,167887668
20	314	0,59624137	240137,1	43,75	0,13933121
25	493	0,573435614	271738,3	45,83333333	0,092968222
30	715	0,555569253	294382,7	64,58333333	0,09032634
35	721	0,535623319	321452,2	66,66666667	0,09246417
40	951	0,544852808	341884,5	68,75	0,072292324
45	1048	0,51186691	353268,8	68,75	0,065601145
50	981	0,500154889	365003,8	68,75	0,070081549
60	1649	0,490566603	402931,9	72,91666667	0,044218718
70	1733	0,485040775	427643,9	83,33333333	0,04808617
100	2110	0,455433187	495015,8	81,25	0,038507109
150	3028	0,442394536	607366,4	95,83333333	0,031649053

Table 50- Score Table for Elliptic Envelope

Cont	NA	SS	VRC	PC	RPA
0.001	7	0,991707628	4841,907	10,41666667	1,488095238
0.005	33	0,96759328	1017,79	25	0,757575758
0.01	66	0,944581719	627,2996	35,41666667	0,536616162
0.05	330	0,857128903	325,9364	54,16666667	0,164141414
0.08	528	0,817411104	275,8846	62,5	0,118371212
0.1	660	0,794660183	253,7028	66,66666667	0,101010101
0.15	989	0,742560344	212,9662	68,75	0,069514661
0.2	1318,5	0,684948576	179,2845	75	0,05688283
0.25	1649	0,622642859	151,7892	81,25	0,049272286
0.3	1978	0,55739309	129,3682	89,58333333	0,045289855
0.35	2308	0,488572885	110,4037	91,66666667	0,039716927
0.4	2638	0,416703254	94,23445	93,75	0,035538287

Table 51- Score Table for Affinity Propagation

Conv	NC	NA	SS	VRC	PC	RPA
1	4265	6110	-5,76082E-05	0,446282077	100	0,016366612
5	7	40	0,818317469	74933,92094	14,58333333	0,364583333
10	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
50	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
80	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
100	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
150	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
200	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
250	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
300	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
350	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
400	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
450	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
500	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
550	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
600	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
700	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
750	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
800	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
850	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
900	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
950	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416
1000	3123	5742	0,024153467	30,24007349	89,58333333	0,015601416

Table 52-Score Table for DBSCAN

MD	MP	NC	NA	SS	VRC	PC	RPA
1	1	47	73	0,658621172	6957,726	39,58333333	0,542237443
1	2	12	73	0,656648318	92,00939	39,58333333	0,542237443
1	3	7	73	0,658941057	139,7839	39,58333333	0,542237443
1	4	3	78	0,659640945	337,6312	39,58333333	0,507478632
1	5	4	89	0,648205148	212,1366	41,66666667	0,468164794
1	6	3	93	0,653739088	293,9172	43,75	0,470430108
1	7	3	94	0,637473928	280,336	43,75	0,465425532
1	8	3	98	0,638295954	273,4176	43,75	0,446428571
1	9	2	103	0,925217846	499,4031	43,75	0,424757282
1	10	2	110	0,922210647	487,0109	43,75	0,397727273
1	11	2	112	0,921312415	483,0045	43,75	0,390625
1	12	2	113	0,920843057	480,6405	43,75	0,387168142
1	15	3	146	0,805462477	234,2944	43,75	0,299657534
1	16	3	152	0,800479024	229,4438	43,75	0,287828947
1	18	2	158	0,905333965	430,9667	43,75	0,276898734
1	20	2	169	0,902219576	423,442	43,75	0,25887574
1	25	2	181	0,898225628	412,8265	43,75	0,241712707
1	30	2	189	0,895946037	406,8691	43,75	0,231481481
1	35	2	198	0,893181984	399,4597	45,625	0,230429293
1	40	2	0	0,889882503	392,2773	0	0
1	45	2	0	0,885264007	382,7018	0	0
1	50	2	0	0,881111265	373,7791	0	0
1	60	2	0	0,872301152	357,3791	0	0
1	70	2	0	0,869778789	352,5585	0	0
1	100	2	0	0,861573143	337,8646	0	0
1	150	2	0	0,847999072	316,8198	0	0

Appendix E – Users approach: Score Tables for perspective TelDays

Table 53- Score Table for Isolation Forest

Cont	NA	SS	VRC	PC	RPA
0.001	6,2	0,983106861	3961,906964	7,5	1,220238095
0.005	30,5	0,93841581	1035,50056	13,95833333	0,459820944
0.01	62,2	0,897887633	725,6472687	16,875	0,274152371
0.05	326,1	0,724827482	409,4058815	36,66666667	0,112417836
0.08	525,5	0,654312576	333,9051219	44,58333333	0,084851979
0.1	656,8	0,622121508	315,0472103	48,33333333	0,07361132
0.15	980,8	0,540577591	312,1055518	55	0,056065044
0.2	1314,8	0,45896944	250,2433391	63,125	0,048012306
0.25	1571,8	0,403808487	212,6125395	63,54166667	0,040025835
0.3	1969	0,353976839	221,0274491	72,08333333	0,036604413
0.35	2298,9	0,342776303	236,3386495	79,58333333	0,034619247
0.4	2629,8	0,255551275	182,0835687	78,54166667	0,029866385

Table 54- Score Table for TelDays

NC	NA	SS	VRC	PC	RPA
2	2	0,993948744	31417,3	4,166666667	2,083333333
3	2	0,807793261	37276,09	4,166666667	2,083333333
4	159	0,72943149	38438,17	12,5	0,078616352
5	159	0,729220393	42850,3	12,5	0,078616352
6	52	0,687508357	53041,81	8,333333333	0,16025641
7	125	0,671659377	66366,73	10,41666667	0,083333333
8	236	0,656767919	81902,54	12,5	0,052966102
9	136	0,642794409	95915,8	10,41666667	0,076593137
10	202	0,633408491	111471,9	12,5	0,061881188
11	174	0,608100333	129291	12,5	0,07183908
12	255	0,597489375	145600,4	12,5	0,049019608
15	539	0,566982288	172743	14,58333333	0,027056277
16	522	0,54898123	180637,2	14,58333333	0,02793742
18	505	0,522988729	186531,4	14,58333333	0,028877888
20	670	0,530531369	194046,6	14,58333333	0,021766169
25	981	0,511255655	210526,2	50	0,0509684
30	1003	0,508574435	223091,5	50	0,049850449
35	1277	0,496810795	230879,2	52,08333333	0,040785696
40	1576	0,486589444	238223,4	58,33333333	0,037013536
45	1662	0,461241535	237657,5	62,5	0,037605295
50	1995	0,468421472	243559,6	68,75	0,034461153
60	2236	0,460150926	247584,9	72,91666667	0,032610316
70	2641	0,470250297	254568,3	77,08333333	0,029187177
100	3132	0,44486671	278845,3	93,75	0,02993295
150	3770	0,421792421	304670,6	95,83333333	0,025419982

Table 55- Score Table for Elliptic Envelope

Cont	NA	SS	VRC	PC	RPA
0.001	7	0,982060942	3875,142365	8,333333333	1,19047619
0.005	33	0,944815538	1472,270775	8,333333333	0,252525253
0.01	66	0,91894271	1231,18083	8,333333333	0,126262626
0.05	330	0,841836619	1031,431276	12,5	0,037878788
0.08	528	0,809404537	948,9910139	14,58333333	0,027619949
0.1	660	0,789664478	896,2193269	18,75	0,028409091
0.15	989	0,738668445	766,5189696	37,5	0,037917088
0.2	1319	0,683207856	653,9468502	54,16666667	0,041066464
0.25	1649	0,628119533	564,2863165	62,5	0,037901759
0.3	1978	0,570489667	487,8821081	77,08333333	0,03897034
0.35	2308	0,51003453	421,4335907	81,25	0,03520364
0.4	2638	0,450878221	366,2570969	85,41666667	0,032379328

Table 56- Score Table for Affinity Propagation

Conv	NC	NA	SS	VRC	PC	RPA
1	3356	5830	0,000869323	25,15233809	97,91666667	0,016795312
5	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
10	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
50	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
80	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
100	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
150	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
200	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
250	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
300	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
350	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
400	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
450	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
500	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
550	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
600	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
700	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
750	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
800	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
850	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
900	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
950	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312
1000	3267	5830	0,004691219	26,45787772	97,91666667	0,016795312

Table 57- Score Table for DBSCAN

MD	MP	NC	NA	SS	VRC	PC	RPA
0.5	1	21	43	0,330916699	2938,071174	10,41666667	0,242248062
0.5	2	7	43	0,754241051	333,4875929	10,41666667	0,242248062
0.5	3	3	43	0,85574937	825,3281688	10,41666667	0,242248062
0.5	4	3	43	0,85574937	825,3281688	10,41666667	0,242248062
0.5	5	3	45	0,854282384	760,8195061	10,41666667	0,231481481
0.5	6	3	45	0,854282384	760,8195061	10,41666667	0,231481481
0.5	7	4	46	0,855569245	429,376321	12,5	0,27173913
0.5	8	2	47	0,92880472	1194,049892	12,5	0,265957447
0.5	9	3	53	0,826802331	614,2336281	12,5	0,235849057
0.5	10	2	56	0,922357039	1169,440454	12,5	0,223214286
0.5	11	2	57	0,921673106	1166,82318	12,5	0,219298246
0.5	12	2	57	0,921673106	1166,82318	12,5	0,219298246
0.5	15	2	63	0,916842129	1130,719541	12,5	0,198412698
0.5	16	2	65	0,914702986	1101,746763	14,58333333	0,224358974
0.5	18	2	67	0,912387435	1068,48192	16,66666667	0,248756219
0.5	20	2	83	0,90236756	1036,731677	16,66666667	0,200803213
0.5	25	2	106	0,887319999	959,2677544	20,83333333	0,196540881
0.5	30	2	153	0,872307978	989,5421694	20,83333333	0,136165577
0.5	35	2	164	0,864600461	933,8374523	27,08333333	0,165142276
0.5	40	2	183	0,857692915	922,0121242	27,08333333	0,147996357
0.5	45	2	194	0,854134142	918,3228998	29,16666667	0,150343643
0.5	50	2	0	0,849677015	936,4150695	0	0
0.5	60	2	0	0,838561061	947,9047479	0	0
0.5	70	2	0	0,832938345	945,2039124	0	0
0.5	100	2	0	0,822067732	936,1542527	0	0
0.5	1	21	43	0,330916699	2938,071174	10,41666667	0,242248062

Appendix F – Users approach: Score Tables for perspective All

Table 58- Score Table for Isolation Forest

Cont	NA	SS	VRC	PC	RPA
0.001	6,8	0,992772836	1136,904	9,375	1,374007937
0.005	32,6	0,987344693	231,5424	18,75	0,575284091
0.01	64,7	0,898036097	85,13888	19,79167	0,305668481
0.05	325,9	0,884184891	24,25781	46,45833	0,142920357
0.08	520	0,818980396	15,08234	54,375	0,104725424
0.1	655	0,772548877	11,21517	48,125	0,073463543
0.15	971,3	0,583921115	6,613347	52,5	0,054068923
0.2	1303,1	0,575145139	5,18344	66,66667	0,051160628
0.25	1626,8	0,478573633	3,99977	69,16667	0,04255675
0.3	1948,6	0,393791695	3,157682	72,70833	0,037314323
0.35	2295	0,299124888	2,501258	77,5	0,033767959
0.4	2597,3	0,210925139	2,041027	79,58333	0,030586449

Table 59- Score Table for K-means

NC	NA	SS	VRC	PC	RPA
2	2	0,993948744	31417,3	0,055555556	2,083333333
3	2	0,807793261	37276,09	0,055555556	2,083333333
4	159	0,72943149	38438,17	5,888888889	0,078616352
5	159	0,729220393	42850,3	5,888888889	0,078616352
6	52	0,687508357	53041,81	1,722222222	0,16025641
7	125	0,671659377	66366,73	4,611111111	0,083333333
8	236	0,656767919	81902,54	9,222222222	0,052966102
9	136	0,642794409	95915,8	5,055555556	0,076593137
10	202	0,633408491	111471,9	7,777777778	0,061881188
11	174	0,608100333	129291	6,555555556	0,07183908
12	255	0,597489375	145600,4	10,05555556	0,049019608
15	539	0,566982288	172743	20	0,027056277
16	522	0,54898123	180637,2	19,27777778	0,02793742
18	505	0,522988729	186531,4	18,83333333	0,028877888
20	670	0,530531369	194046,6	21,88888889	0,021766169
25	981	0,511255655	210526,2	33,55555556	0,0509684
30	1003	0,508574435	223091,5	33,72222222	0,049850449
35	1277	0,496810795	230879,2	42,44444444	0,040785696
40	1576	0,486589444	238223,4	50,72222222	0,037013536
45	1662	0,461241535	237657,5	52,72222222	0,037605295
50	1995	0,468421472	243559,6	61,83333333	0,034461153
60	2236	0,460150926	247584,9	67,11111111	0,032610316
70	2641	0,470250297	254568,3	74,61111111	0,029187177
100	3132	0,44486671	278845,3	83	0,02993295
150	3770	0,421792421	304670,6	92,22222222	0,025419982

Table 60- Score Table for Elliptic Envelope

Cont	NA	SS	VRC	PC	RPA
0.001	7	0,982060942	3875,142365	8,333333333	1,19047619
0.005	33	0,944815538	1472,270775	8,333333333	0,252525253
0.01	66	0,91894271	1231,18083	8,333333333	0,126262626
0.05	330	0,841836619	1031,431276	12,5	0,037878788
0.08	528	0,809404537	948,9910139	14,58333333	0,027619949
0.1	660	0,789664478	896,2193269	18,75	0,028409091
0.15	989	0,738668445	766,5189696	37,5	0,037917088
0.2	1319	0,683207856	653,9468502	54,16666667	0,041066464
0.25	1649	0,628119533	564,2863165	62,5	0,037901759
0.3	1978	0,570489667	487,8821081	77,08333333	0,03897034
0.35	2308	0,51003453	421,4335907	81,25	0,03520364
0.4	2638	0,450878221	366,2570969	85,41666667	0,032379328

Table 61- Score Table for Affinity Propagation

Conv	NC	NA	SS	VRC	PC	RPA
1	5159	6110	0,00065	84,63491	100	0,016366612
5	8	16	0,957964	6583795	18,75	1,171875
10	9	32	0,94048	8489630	33,33333333	1,041666667
50	9	32	0,94048	8489630	33,33333333	1,041666667
80	9	32	0,94048	8489630	33,33333333	1,041666667
100	9	32	0,94048	8489630	33,33333333	1,041666667
150	9	32	0,94048	8489630	33,33333333	1,041666667
200	9	32	0,94048	8489630	33,33333333	1,041666667
250	9	32	0,94048	8489630	33,33333333	1,041666667
300	9	32	0,94048	8489630	33,33333333	1,041666667
350	9	32	0,94048	8489630	33,33333333	1,041666667
400	9	32	0,94048	8489630	33,33333333	1,041666667
450	9	32	0,94048	8489630	33,33333333	1,041666667
500	9	32	0,94048	8489630	33,33333333	1,041666667
550	9	32	0,94048	8489630	33,33333333	1,041666667
600	9	32	0,94048	8489630	33,33333333	1,041666667
700	9	32	0,94048	8489630	33,33333333	1,041666667
750	9	32	0,94048	8489630	33,33333333	1,041666667
800	9	32	0,94048	8489630	33,33333333	1,041666667
850	9	32	0,94048	8489630	33,33333333	1,041666667
900	9	32	0,94048	8489630	33,33333333	1,041666667
950	9	32	0,94048	8489630	33,33333333	1,041666667
1000	9	32	0,94048	8489630	33,33333333	1,041666667

For the DBSCAN score table, since there are 725 combinations of parameters, only an excerpt with the relevant zone that has the best parameterization is shown.

Table 62- Score Table for DBSCAN

MD	MP	NC	NA	SS	VRC	PC	RPA
3	1	87	119	0,636188571	2049030,58	41,66666667	0,350140056
3	2	17	119	0,627834173	73,48719155	41,66666667	0,350140056
3	3	10	119	0,644993668	109,3776573	41,66666667	0,350140056
3	4	4	120	0,783224416	264,408829	41,66666667	0,347222222
3	5	3	120	0,798750397	383,0384558	41,66666667	0,347222222
3	6	3	123	0,79884861	372,9795299	41,66666667	0,338753388
3	7	3	127	0,798551558	360,1155519	41,66666667	0,32808399
3	10	2	149	0,947997701	565,3297296	45,83333333	0,307606264
3	8	3	136	0,794100891	331,4099235	41,66666667	0,306372549
3	9	3	144	0,79363273	311,732838	43,75	0,303819444
3	11	2	151	0,947328901	557,707419	45,83333333	0,303532009
3	15	2	167	0,941927996	504,579847	50	0,299401198
3	16	2	168	0,941591518	501,6931238	50	0,297619048
3	12	2	155	0,945982561	543,3783745	45,83333333	0,295698925
3	18	2	173	0,939898763	487,3769035	50	0,289017341
3	20	2	174	0,939559357	484,6017346	50	0,287356322
3	25	2	185	0,935910305	455,8260253	50	0,27027027
3	30	2	196	0,932278688	429,7724019	50	0,255102041
3	35	2	0	0,930271875	417,1190039	0	0
3	40	2	0	0,925025775	386,3231089	0	0
3	45	2	0	0,923708824	379,2960715	0	0
3	50	2	0	0,921054912	366,1260229	0	0
3	60	2	0	0,914866467	337,6946247	0	0
3	70	2	0	0,910982731	321,8074527	0	0
3	100	2	0	0,900755892	285,7958985	0	0
3	150	2	0	0,883986353	240,6963985	0	0

Appendix G – Telephones approach: Score Tables for perspective Ratios

Table 63- Table Score for Isolation Forest

Cont	NA	SS	VRC	PC	RPA
0.001	306,9	0,171935839	2348,671116	14,75728155	0,048731332
0.005	1222,3	0,651724741	1092,260817	31,55339806	0,026011128
0.01	1821,4	0,816497297	570,6009541	38,44660194	0,021144508
0.05	3181,2	0,540378954	187,1033077	43,49514563	0,013675396
0.08	3427,2	0,67721979	114,4898091	44,95145631	0,013140453
0.1	3633,9	0,435544668	80,62809802	45,4368932	0,012517595
0.15	3847,2	0,674710427	66,74886705	46,11650485	0,012001966
0.2	3999,1	0,520744918	47,02463664	46,01941748	0,011513405
0.25	4150,9	0,39797568	31,33238595	46,40776699	0,011182864
0.3	4205,8	0,406533014	32,24236694	46,40776699	0,011039528
0.35	4253,4	0,27513385	21,76541435	46,11650485	0,010851538
0.4	4338,2	0,211691851	17,19424783	46,60194175	0,010745463

Table 64-Score Table for K-means (silhouette score and VRC did not run 10 times when number of clusters was 2 and 3 because of situation explained before in which no anomalies were detected)

NC	NA	SS	VRC	PC	RPA
2	24	0,79765457	333378,3165	5,825242718	0,242718
3	24	0,78263856	583343,3668	5,825242718	0,242718
4	35	0,77293866	743378,3664	9,708737864	0,277393
5	89	0,767774269	782883,2955	17,47572816	0,196356
6	2278	0,670359216	880930,8775	42,7184466	0,018753
7	1009	0,658473098	881317,4806	32,03883495	0,031753
8	1514	0,63280485	905657,7671	35,9223301	0,023727
9	1153	0,657879099	943762,3047	33,98058252	0,029471
10	2038	0,638400287	971398,6044	38,83495146	0,019055
11	2563	0,635205549	1036998,488	42,7184466	0,016667
12	2563	0,646015056	1082380,927	42,7184466	0,016667
15	1539	0,658761099	1111047,604	37,86407767	0,024603
16	3061	0,673498355	1146796,909	43,68932039	0,014273
18	2881	0,717004858	1144504,089	42,7184466	0,014828
20	2820	0,717833219	1252095,828	41,74757282	0,014804
25	3054	0,71761229	1305109,827	44,66019417	0,014624
30	3115	0,731033874	1389557,704	43,68932039	0,014025
35	3365	0,743036388	1460112,484	43,68932039	0,012983
40	3457	0,783247432	1497044,126	46,60194175	0,01348
45	3545	0,792909381	1560882,705	46,60194175	0,013146
50	3645	0,78871582	1597395,484	46,60194175	0,012785
60	3593	0,817526317	1719676,075	46,60194175	0,01297
70	3991	0,842264347	1837152,854	46,60194175	0,011677
100	4208	0,877031506	2198242,083	46,60194175	0,011075
150	4210	0,903878319	2759702,478	46,60194175	0,011069

Table 65- Score Table for Elliptic Envelope

Cont	NA	SS	VRC	PC	RPA
0.001	90,2	0,998281553	1164,611	1,941747573	0,021527556
0.005	948,4	0,992474683	247,3747	32,7184466	0,034450151
0.01	1505,1	0,982189095	123,037	32,23300971	0,021216849
0.05	2857,2	0,901118726	23,60634	42,13592233	0,014974046
0.08	3314,1	0,842648539	14,28698	44,75728155	0,013602172
0.1	3415,9	0,799920622	11,17994	44,95145631	0,013183969
0.15	3705,5	0,703486843	7,038815	45,24271845	0,012234868
0.2	3992,5	0,597156296	4,968469	45,33980583	0,011364306
0.25	4080,4	0,503037641	3,726357	46,31067961	0,011356227
0.3	4222,1	0,404150379	2,898192	46,21359223	0,010950499
0.35	4326,8	0,30235826	2,306707	46,60194175	0,010771964
0.4	4376,3	0,204260416	1,863042	46,60194175	0,010649119

Appendix H – Telephones approach: Score Tables for perspective AccessDay

Table 66- Score Table for Isolation Forest

Cont	NA	SS	VRC	PC	RPA
0.001	382,9	0,974007065	129964,3959	27,37864078	0,07152534
0.005	1391,4	0,925816948	67779,39795	38,54368932	0,027708635
0.01	1972	0,908313973	60990,69953	41,65048544	0,021124813
0.05	3268,1	0,794822138	49434,77816	45,4368932	0,013908923
0.08	3544,3	0,749920511	43876,2308	45,63106796	0,012885999
0.1	3705,1	0,743406426	48582,56263	45,63106796	0,012316542
0.15	3919,9	0,668359873	39832,20399	45,82524272	0,011692101
0.2	4051,7	0,64148847	37830,79926	46,31067961	0,011432891
0.25	4190	0,578137599	32328,61816	46,60194175	0,011123798
0.3	4262,5	0,535800947	29734,95193	46,40776699	0,010887975
0.35	4316,6	0,549836062	31254,22088	46,60194175	0,010802306
0.4	4294,5	0,530245605	31799,86729	46,50485437	0,010838393

Table 67- Score Table for K-means

NC	NA	SS	VRC	PC	RPA
2	19	0,953455	3249780,703	3,883495146	0,204394481
3	42	0,953455	3249780,703	9,708737864	0,231160425
4	41	0,830364	814462,1937	9,708737864	0,236798484
5	102	0,812718	888271,9039	20,38834951	0,19988578
6	1180	0,748804	914833,1644	38,83495146	0,032910976
7	2248	0,693341	950012,1624	43,68932039	0,019434751
8	1901	0,724413	1004193,928	41,74757282	0,021960848
9	1094	0,645468	1033347,074	35,9223301	0,032835768
10	1118	0,659611	1100836,701	35,9223301	0,032130886
11	2248	0,627769	1127465,483	43,68932039	0,019434751
12	2430	0,625421	1145162,405	44,66019417	0,018378681
15	2567	0,65986	1203241,313	44,66019417	0,017397816
16	2705	0,659895	1225607,888	44,66019417	0,016510238
18	2753	0,66214	1245353,991	44,66019417	0,016222373
20	2904	0,664072	1266244,524	44,66019417	0,015378855
25	3251	0,74185	1343596,35	45,63106796	0,01403601
30	3403	0,792088	1383665,632	45,63106796	0,013409071
35	3423	0,759988	1451855,564	45,63106796	0,013330724
40	3447	0,757151	1488842,243	45,63106796	0,013237908
45	3872	0,765899	1548110,735	45,63106796	0,011784883
50	3946	0,829275	1651400,367	45,63106796	0,011563879
60	3983	0,854064	1747289,157	46,60194175	0,011700211
70	3968	0,882333	1892622,399	45,63106796	0,011499765
100	4121	0,925871	2353280,056	46,60194175	0,011308406
150	4121	0,953064	3249780,703	46,60194175	0,011308406

Table 68- Score Table for Elliptic Envelope

Cont	NA	SS	VRC	PC	RPA
0.001	393	0,984766205	150055,1	26,21359223	0,066701253
0.005	1292	0,939723048	87399,55	36,89320388	0,028555111
0.01	1840	0,920718689	76727,68	42,7184466	0,023216547
0.05	3000	0,828047311	63000,92	44,66019417	0,014886731
0.08	3408	0,783137563	57470,86	45,63106796	0,013389398
0.1	3557	0,762775301	54654,06	45,63106796	0,012828526
0.15	3978	0,661279444	42795,78	46,60194175	0,011714917
0.2	3978	0,663374746	42795,78	46,60194175	0,011714917
0.25	3978	0,661904005	42795,78	46,60194175	0,011714917
0.3	4167	0,588154277	35760,52	46,60194175	0,011183571
0.35	4167	0,58459395	35760,52	46,60194175	0,011183571
0.4	4368	0,473502289	27504,87	46,60194175	0,010668943

Appendix I – Telephones approach: Score Tables for perspective AccessActor

Table 69-Score Table for Isolation Forest

Cont	NA	SS	VRC	PC	RPA
0.001	419,1	0,971850433	108986,7	22,52427184	0,053947878
0.005	1349,6	0,919510273	66504,18	36,40776699	0,026980878
0.01	1912,8	0,895412815	61824,65	40,19417476	0,021012993
0.05	3247,4	0,791445043	53773,06	44,5631068	0,013726401
0.08	3503	0,76039211	51064,26	44,75728155	0,01279779
0.1	3666	0,724654559	44947,45	45,24271845	0,012348379
0.15	3903,9	0,648292314	39213,11	46,01941748	0,011788686
0.2	4039	0,644585403	42670,47	46,40776699	0,011492038
0.25	4165,5	0,58752025	36366,83	46,60194175	0,011190698
0.3	4219,6	0,551515881	32656,16	46,40776699	0,010999022
0.35	4229,8	0,542012133	31055,69	46,50485437	0,011022783
0.4	4364,2	0,482290305	29507,84	46,60194175	0,010678553

Table 70- Score Table for K-means (silhouette score and VRC did not run 10 times when number of clusters was 2, 3 and 4 because of situation explained before in which no anomalies were detected)

NC	NA	SS	VRC	PC	RPA
2	20	0,88837312	4834895	3,883495146	0,194174757
3	43	0,88837312	4834895	10,67961165	0,248363062
4	37	0,817990307	843361,1	8,737864078	0,236158489
5	92	0,792522931	905121,3	18,44660194	0,200506543
6	1797	0,704611898	939778,2	41,74757282	0,023231816
7	1668	0,719899235	984970,3	40,77669903	0,024446462
8	2429	0,680913349	1034358	43,68932039	0,017986546
9	2626	0,676539321	1069796	44,66019417	0,017006928
10	1221	0,634492661	1142791	35,9223301	0,029420418
11	2283	0,625206474	1191351	43,68932039	0,019136803
12	2556	0,62370157	1201208	44,66019417	0,017472689
15	2820	0,663363478	1298887	44,66019417	0,015836948
16	2950	0,661507103	1333213	44,66019417	0,015139049
18	3235	0,653892251	1399407	44,66019417	0,013805315
20	2794	0,668738163	1452325	43,68932039	0,015636836
25	3125	0,74045084	1565766	44,66019417	0,014291262
30	3305	0,676621629	1687810	44,66019417	0,013512918
35	3422	0,809207192	1788727	44,66019417	0,013050904
40	3640	0,827832936	1954768	44,66019417	0,012269284
45	4014	0,822653367	2067536	46,60194175	0,011609851
50	3926	0,82274727	2291244	45,63106796	0,011622789
60	3891	0,867372918	2607552	45,63106796	0,011727337
70	3998	0,89943198	2925980	46,60194175	0,011656314
100	4136	0,935985149	3890233	46,60194175	0,011267394
150	4136	0,965996698	5370329	46,60194175	0,011267394

Table 71- Score Table for Elliptic Envelope

Cont	NA	SS	VRC	PC	RPA
0.001	393	0,972628693	138042,5	26,21359223	0,066701253
0.005	1292	0,939104495	83176,97	36,89320388	0,028555111
0.01	1920	0,897016138	73832,57	43,68932039	0,022754854
0.05	3000	0,824320927	64562,49	44,66019417	0,014886731
0.08	3408	0,776474738	59691,27	45,63106796	0,013389398
0.1	3557	0,755852689	57135,9	45,63106796	0,012828526
0.15	3827	0,696821197	50328,69	45,63106796	0,011923456
0.2	3978	0,655305112	45471,45	46,60194175	0,011714917
0.25	3978	0,653833928	45471,45	46,60194175	0,011714917
0.3	4368	0,479966418	29667,47	46,60194175	0,010668943
0.35	4368	0,467223567	29667,47	46,60194175	0,010668943
0.4	4565	0,206821277	15992,77	46,60194175	0,010208531

Appendix J – Telephones approach: Score Tables for perspective DayActors

Table 72- Score Table for Isolation Forest

Cont	NA	SS	VRC	PC	RPA
0.001	410,6	0,952388729	150550,5	23,00970874	0,056474654
0.005	1375,1	0,924727799	144991,7	38,6407767	0,028104363
0.01	2002	0,887600907	147291,9	42,81553398	0,021388695
0.05	3440,1	0,781906197	160482,3	44,66019417	0,012987376
0.08	3662	0,754215232	152829,9	44,85436893	0,0122607
0.1	3788,3	0,742304051	164235	45,04854369	0,011896104
0.15	4051,7	0,71690537	185668,9	45,63106796	0,011262635
0.2	4034,3	0,717565998	181188,3	45,63106796	0,01131186
0.25	4060	0,723785465	191597,4	45,63106796	0,011239179
0.3	4233,6	0,756306302	220191,8	46,31067961	0,010944028
0.35	4274,9	0,765072971	222433,2	46,50485437	0,010883156
0.4	4208,8	0,754398467	216106,9	46,21359223	0,010986193

Table 73- Score Table for K-means

NC	NA	SS	VRC	PC	RPA
2	92	0,973737938	5077840	18,44660194	0,200506543
3	86	0,973737938	5077840	15,53398058	0,180627681
4	2761	0,841791995	623819,5	44,66019417	0,016175369
5	2761	0,843503852	649257	44,66019417	0,016175369
6	1358	0,859290731	694051	37,86407767	0,027882237
7	1530	0,849215218	720063,8	37,86407767	0,024747763
8	2079	0,851311786	755032,9	38,83495146	0,01867963
9	2350	0,913258896	758433	41,74757282	0,017764925
10	2350	0,90402468	800855	41,74757282	0,017764925
11	2803	0,917998719	839179,5	44,66019417	0,015932998
12	2761	0,904925963	877787,1	44,66019417	0,016175369
15	3248	0,929261829	959429,6	44,66019417	0,01375006
16	3303	0,932171093	993478	44,66019417	0,0135211
18	3605	0,957272261	1071916	44,66019417	0,012388403
20	3649	0,958509068	1105498	44,66019417	0,012239023
25	3718	0,968310247	1316004	44,66019417	0,012011887
30	3734	0,978623159	1454256	44,66019417	0,011960416
35	3734	0,982701296	1554090	44,66019417	0,011960416
40	3734	0,986764549	1738819	44,66019417	0,011960416
45	3734	0,986679964	1923231	44,66019417	0,011960416
50	3734	0,987761128	2036439	44,66019417	0,011960416
60	3734	0,991954961	2357282	44,66019417	0,011960416
70	3734	0,994775153	2631242	44,66019417	0,011960416
100	3734	0,995257359	3543302	44,66019417	0,011960416
150	3734	0,996740164	5616217	44,66019417	0,011960416

Table 74- Score Table for Elliptic Envelope

Cont	NA	SS	VRC	PC	RPA
0.001	377	0,965494318	206151	29,12621359	0,077257861
0.005	1374	0,921328611	168774,6	38,83495146	0,028264157
0.01	1997	0,895610877	193192,7	42,7184466	0,02139131
0.05	3595	0,792218129	231958,8	44,66019417	0,012422863
0.08	3595	0,795296048	231958,8	44,66019417	0,012422863
0.1	3773	0,811020224	295654,5	44,66019417	0,011836786
0.15	4308	0,770700546	232446,6	46,60194175	0,010817535
0.2	4308	0,775881803	232446,6	46,60194175	0,010817535
0.25	4308	0,774006982	232446,6	46,60194175	0,010817535
0.3	4816	0,774006982	232446,6	46,60194175	0,009676483
0.35	4816	0,774006982	232446,6	46,60194175	0,009676483
0.4	4816	0,774006982	232446,6	46,60194175	0,009676483

Appendix K – Telephones approach: Score

Tables for perspective All

Cont	NA	SS	VRC	PC	RPA
0.001	390,5	0,954169618	74249,03	21,84466019	0,056452221
0.005	1327,2	0,86501779	40257,77	36,40776699	0,027408996
0.01	1924,3	0,840375255	37503,13	40,97087379	0,021301815
0.05	3271,9	0,723115038	36305,46	44,95145631	0,013744416
0.08	3540,9	0,683553387	37937,42	44,95145631	0,012699879
0.1	3687,8	0,640015049	35891,98	45,24271845	0,012270525
0.15	3951,1	0,579175353	32611,41	45,63106796	0,011550796
0.2	4053,5	0,543577091	32668,29	46,31067961	0,011430915
0.25	4180	0,507257573	31540,77	46,50485437	0,011126436
0.3	4236,2	0,483635516	32211,74	46,60194175	0,01100158
0.35	4295,7	0,435778041	27437,46	46,11650485	0,010739432
0.4	4374,5	0,42611963	30578,33	46,60194175	0,010653384

Table 75- Score Table for K-means

NC	NA	SS	VRC	PC	RPA
2	26	0,882772553	575198,2	5,825242718	0,224047797
3	49	0,882772553	575198,2	10,67961165	0,217951258
4	46	0,838557805	528851,4	10,67961165	0,232165471
5	98	0,796891793	514085	19,41747573	0,198137507
6	94	0,789260494	478763,3	19,41747573	0,206568891
7	98	0,659619604	456383,8	19,41747573	0,198137507
8	1639	0,639835086	472299	37,86407767	0,023101939
9	2294	0,58400587	464968,4	41,74757282	0,018198593
10	2091	0,589436037	473718,6	39,80582524	0,019036741
11	2724	0,585836586	469443,2	44,66019417	0,016395079
12	2698	0,585787766	483545,7	44,66019417	0,016553074
15	2702	0,568579573	484065,6	42,7184466	0,015809936
16	1692	0,571792271	484446,3	40,77669903	0,024099704
18	2751	0,61531729	503318,4	44,66019417	0,016234167
20	2365	0,647511499	504153	43,68932039	0,018473286
25	3285	0,696979842	522830,7	45,63106796	0,013890736
30	3155	0,706069286	542403,7	44,66019417	0,014155371
35	3290	0,711818375	526648,3	44,66019417	0,013574527
40	3350	0,746568274	538825,8	44,66019417	0,013331401
45	3633	0,749701065	538091,1	44,66019417	0,012292924
50	3389	0,771126768	530916,9	45,63106796	0,013464464
60	3942	0,788824185	550539,8	46,60194175	0,011821903
70	3868	0,804079522	554833,3	46,60194175	0,012048072
100	4206	0,832533783	588059,3	46,60194175	0,011079872
150	4307	0,870929224	635481,6	46,60194175	0,010820047

Table 76- Score Table for Elliptic Envelope

Cont	NA	SS	VRC	PC	RPA
0.001	362,5	0,966012418	109221,6	26,40776699	0,073477073
0.005	1288,4	0,8863761	56827,68	38,25242718	0,029696132
0.01	1781,2	0,869294158	50315,34	40,19417476	0,022625519
0.05	2964,1	0,771615802	52174,36	44,75728155	0,015137053
0.08	3408,2	0,710476609	48559,94	45,72815534	0,013451196
0.1	3540,1	0,689417313	50649,91	45,72815534	0,012951669
0.15	3816,1	0,633562946	46612,84	46,11650485	0,012109166
0.2	3795,9	0,610002504	47698,55	46,60194175	0,012294396
0.25	4190,7	0,480640498	36621,36	46,60194175	0,011134957
0.3	4073,6	0,544124324	40964,43	46,60194175	0,011452898
0.35	4231,2	0,448316576	33323,45	46,60194175	0,011032122
0.4	4238,9	0,459124718	34484,65	46,60194175	0,011014477

Appendix L – Final Results

The two following tables show the final results and the algorithms responsible for the detection of each anomaly.

Table 77- Table with final Results

Id	Anomaly_Type	Detected_by_Algorithms
1093	type1;	All
1215	type10; type1;	All
1855	type10; type1;	All
188612	type10; type4;	All
18905	type6; type5;	All
20137	type3; type2;	All
2133	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
22506	type10; type1;	All
2328	type10; type1;	All
23308	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
23496	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
25401	type10; type1;	All
2697	type10; type1;	All
2852	type10; type1;	All
2876	type10; type1;	All
291554	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
311082	type2;	All
31754	type10; type1;	All
318909	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
320532	type10; type1;	All
32076	type10; type1;	All

32135	type10; type1;	All
321994	type10; type1;	All
32203	type10; type1;	All
32485	type10; type1;	All
32614	type1;	All
328563	type10; type1;	All
329234	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
336232	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
35839	type10; type1;	All
37234	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
38253	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
40499	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
41257	type10; type1;	All
41272	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
41522	type10; type1;	All
41762	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
4182	type10; type1;	All
4298	type10; type1;	All
43509	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
43522	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
43532	type10; type1;	All
43534	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
43553	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
43564	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
43565	type10; type1;	All
43566	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
43571	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
43843	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;

4428	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
45751	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
49936	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
50064	type10; type1;	All
50455	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
53239	type10; type1;	All
5334	type10; type1;	All
5884	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
57	type9;	All
58	type9; type8;	All
59	type9; type8;	All
60	type9; type8;	All
2936	type1;	All
3018	type2;	All
32323	type10; type1;	All
336499	type10; type4;	All
36205	type10; type1;	All
50396	type1;	Affinity Propagation; Isolation Forest; Elliptic Envelope; DBSCAN;
53293	type1;	All
53320	type1;	All
23303	type2;	All
19531	type1;	Affinity Propagation; Elliptic Envelope; DBSCAN;
23467	type1;	All
2930	type1;	All
31850	type1;	Affinity Propagation; Elliptic Envelope; DBSCAN;
32194	type1;	Affinity Propagation; Elliptic Envelope; DBSCAN;
46268	type10; type1;	All

5366	type10; type1;	All
7657	type10; type1;	All
2191	type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
23203	type10; type1;	All
23513	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
2589	type1;	All
2990	type1;	All
31792	type1;	Affinity Propagation; Isolation Forest; Elliptic Envelope; DBSCAN;
32291	type1;	All
3293	type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
341877	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
46680	type1;	Affinity Propagation; Elliptic Envelope; DBSCAN;
7552	type10; type1;	All
7561	type10; type1;	All
227543	type10; type1;	Affinity Propagation; Kmeans; Isolation Forest; DBSCAN;
291549	type10; type1;	Affinity Propagation; Kmeans; Isolation Forest; DBSCAN;
328790	type10; type1;	Affinity Propagation; Kmeans; Isolation Forest; DBSCAN;
35778	type10; type1;	Affinity Propagation; Kmeans; Isolation Forest; DBSCAN;
108559	type1;	All
23297	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
23307	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
23348	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
23358	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
23380	type1;	Affinity Propagation; Isolation Forest; DBSCAN;
23443	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
23454	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
23460	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;

23477	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
23481	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
23523	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
23553	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
23559	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
2706	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
289171	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
3027	type1;	Affinity Propagation; Isolation Forest; DBSCAN;
318526	type1;	Affinity Propagation; Kmeans; Isolation Forest; DBSCAN;
321881	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
3236	type1;	Affinity Propagation; Isolation Forest; DBSCAN;
329232	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
3300	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
37480	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
40077	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
40548	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
40570	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
40735	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
41012	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
41703	type1;	Affinity Propagation; Kmeans; Isolation Forest; DBSCAN;
4228	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
4329	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
4452	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
4464	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
45178	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
45410	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
45663	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
45668	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;

45710	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
46362	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
46541	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
49838	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
49859	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
50051	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
50060	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
53128	type1;	Affinity Propagation; Isolation Forest; DBSCAN;
5390	type1;	Affinity Propagation; Kmeans; DBSCAN;
6485	type1;	Affinity Propagation; Kmeans; DBSCAN;
7473	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
7575	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
329235	type10; type1;	Affinity Propagation; Kmeans; Isolation Forest; DBSCAN;
41592	type1;	Affinity Propagation; Isolation Forest; DBSCAN;
4324	type1;	Affinity Propagation; Kmeans; Isolation Forest; DBSCAN;
4618	type1;	Affinity Propagation; Kmeans; Isolation Forest; DBSCAN;
23558	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
330543	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
353430	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
41798	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
46669	type10; type1;	Affinity Propagation; Kmeans; DBSCAN;
5885	type10; type1;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
1238	type4;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
2960	type3; type2;	All
4234	type2;	All
48357	type3; type2;	All
544	type3; type2;	All
1998	type4;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;

43923	type6; type5;	All
53083	type2;	All
542574	type6;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
162	type5;	All
163	type4;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
164	type9; type8;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
165	type7; type6;	All
166	type4;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
19271	type3; type2;	All
31772	type3; type2;	All
31949	type6;	All
320967	type2;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
32109	type6;	All
36096	type3; type2;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
39855	type2;	All
39956	type6; type5;	All
46276	type3; type2;	All
52904	type2;	Affinity Propagation; Kmeans; Elliptic Envelope; DBSCAN;
177	type5;	Kmeans; Elliptic Envelope; DBSCAN;
111841	type9; type8;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
28529	type9; type8;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
31660	type9;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
31790	type9;	Kmeans; Elliptic Envelope; DBSCAN;
31882	type6;	Kmeans; Isolation Forest; DBSCAN;
31890	type9; type8;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
31967	type9;	Kmeans; Isolation Forest; DBSCAN;
32020	type9;	Kmeans; Elliptic Envelope; DBSCAN;
32041	type9; type7;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;

32083	type9;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
32340	type9; type8;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
32356	type9;	Kmeans; Isolation Forest; DBSCAN;
32425	type9; type8;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
32538	type9; type8;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
343908	type6;	Kmeans; Isolation Forest; DBSCAN;
431726	type9;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
43897	type9; type8;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
46184	type9; type8;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
547814	type9; type8;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
552568	type9; type8;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
552575	type9; type8;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
199	type5;	Kmeans; Elliptic Envelope; DBSCAN;
200	type6;	Kmeans; Isolation Forest; DBSCAN;
201	type6;	Kmeans; Isolation Forest; DBSCAN;
202	type5;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
203	type6; type5;	Kmeans; Isolation Forest; DBSCAN;
204	type6; type5;	Kmeans; Isolation Forest; DBSCAN;
205	type6;	Kmeans; Isolation Forest; DBSCAN;
206	type6;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
207	type6;	Kmeans; Isolation Forest; DBSCAN;
208	type5;	Kmeans; Isolation Forest; Elliptic Envelope; DBSCAN;
209	type9;	Kmeans; Elliptic Envelope; DBSCAN;
210	type6; type5;	Kmeans; Isolation Forest; DBSCAN;
211	type9;	Kmeans; Isolation Forest; DBSCAN;
212	type9; type8;	Kmeans; Elliptic Envelope; DBSCAN;
19318	type9; type8;	Kmeans; Elliptic Envelope; DBSCAN;
316726	type9; type8;	Kmeans; Elliptic Envelope; DBSCAN;

31969	type9; type8;	Kmeans; Elliptic Envelope; DBSCAN;
32506	type9; type8;	Kmeans; Elliptic Envelope; DBSCAN;
46988	type9; type8;	Kmeans; Elliptic Envelope; DBSCAN;
892	Does not belong to any type.	Kmeans; Elliptic Envelope; DBSCAN;
44764	type2;	Kmeans; Elliptic Envelope; Isolation Forest
2828	type1;	Kmeans; Elliptic Envelope; Isolation Forest
2474	type1;	Kmeans; Elliptic Envelope; Isolation Forest
4218	type1;	Kmeans; Elliptic Envelope; Isolation Forest
24651	type1;	Kmeans; Elliptic Envelope; Isolation Forest
53187	type1;	Kmeans; Elliptic Envelope; Isolation Forest
32763	type1;	Kmeans; Elliptic Envelope; Isolation Forest
53366	type1;	Kmeans; Elliptic Envelope; Isolation Forest
49841	type1;	Kmeans; Elliptic Envelope; Isolation Forest
22747	type1;	Kmeans; Elliptic Envelope; Isolation Forest
31887	type1;	Kmeans; Elliptic Envelope; Isolation Forest
49895	type1;	Kmeans; Elliptic Envelope; Isolation Forest
7525	type1;	Kmeans; Elliptic Envelope; Isolation Forest
49937	type1;	Kmeans; Elliptic Envelope; Isolation Forest
32188	type2;	Kmeans; Elliptic Envelope; Isolation Forest
289166	type1;	Kmeans; Elliptic Envelope; Isolation Forest
18598	type2;	Kmeans; Elliptic Envelope; Isolation Forest
2484	type2;	Kmeans; Elliptic Envelope; Isolation Forest
18766	type1;	Kmeans; Elliptic Envelope; Isolation Forest
44479	type6; type5;	Kmeans; Elliptic Envelope; Isolation Forest
2538	type1;	Kmeans; Elliptic Envelope; Isolation Forest
46492	type1;	Kmeans; Elliptic Envelope; Isolation Forest
42178	type1;	Kmeans; Elliptic Envelope; Isolation Forest

28393	type2;	Kmeans; Elliptic Envelope; Isolation Forest
49832	type1;	Kmeans; Elliptic Envelope; Isolation Forest
3376	Does not belong to any type.	Kmeans; Elliptic Envelope; Isolation Forest
2914	type1;	Kmeans; Elliptic Envelope; Isolation Forest
4215	type1;	Kmeans; Elliptic Envelope; Isolation Forest
7434	type1;	Kmeans; Elliptic Envelope; Isolation Forest
46675	type2;	Kmeans; Elliptic Envelope; Isolation Forest
345745	type1;	Kmeans; Elliptic Envelope; Isolation Forest
49767	type1;	Kmeans; Elliptic Envelope; Isolation Forest
23329	type1;	Kmeans; Elliptic Envelope; Isolation Forest
53364	type1;	Kmeans; Elliptic Envelope; Isolation Forest
23177	type3; type2;	Kmeans; Elliptic Envelope; Isolation Forest
33715	type1;	Kmeans; Elliptic Envelope; Isolation Forest
2688	type1;	Kmeans; Elliptic Envelope; Isolation Forest
32233	type3; type2;	Kmeans; Elliptic Envelope; Isolation Forest
324239	type1;	Kmeans; Elliptic Envelope; Isolation Forest
33496	type2;	Kmeans; Elliptic Envelope; Isolation Forest
311174	type2;	Kmeans; Elliptic Envelope; Isolation Forest
46943	type1;	Kmeans; Elliptic Envelope; Isolation Forest
47415	type1;	Kmeans; Elliptic Envelope; Isolation Forest
289169	type1;	Kmeans; Elliptic Envelope; Isolation Forest
18796	type2;	Kmeans; Elliptic Envelope; Isolation Forest
45431	type1;	Kmeans; Elliptic Envelope; Isolation Forest
4214	type1;	Kmeans; Elliptic Envelope; Isolation Forest
887	type2;	Kmeans; Elliptic Envelope; Isolation Forest
31823	Does not belong to any type.	Kmeans; Elliptic Envelope; Isolation Forest
28997	type1;	Kmeans; Elliptic Envelope; Isolation Forest

44999	type1;	Kmeans; Elliptic Envelope; Isolation Forest
39892	type2;	Kmeans; Elliptic Envelope; Isolation Forest
289156	type1;	Kmeans; Elliptic Envelope; Isolation Forest
47243	type1;	Kmeans; Elliptic Envelope; Isolation Forest
38258	type2;	Kmeans; Elliptic Envelope; Isolation Forest
32667	type1;	Kmeans; Elliptic Envelope; Isolation Forest
28679	type5;	Kmeans; Elliptic Envelope; Isolation Forest
487	type2;	Kmeans; Elliptic Envelope; Isolation Forest
98982	type1;	Kmeans; Elliptic Envelope; Isolation Forest
7598	type1;	Kmeans; Elliptic Envelope; Isolation Forest
1599	type2;	Kmeans; Elliptic Envelope; Isolation Forest
42137	type2;	Kmeans; Elliptic Envelope; Isolation Forest
7676	type1;	Kmeans; Elliptic Envelope; Isolation Forest
4221	type1;	Kmeans; Elliptic Envelope; Isolation Forest
32156	type1;	Kmeans; Elliptic Envelope; Isolation Forest
7381	type2;	Kmeans; Elliptic Envelope; Isolation Forest
1241	type3; type2;	Kmeans; Elliptic Envelope; Isolation Forest
2414	type1;	Kmeans; Elliptic Envelope; Isolation Forest

Table 78- Contribution of each algorithm for the detection of each anomaly

Id	Anomaly_Type	DBSCAN	EE	AffProp	IF	Kmeans
1093	type1;	Sim	Sim	Sim	Sim	Sim
1215	type10; type1;	Sim	Sim	Sim	Sim	Sim
1855	type10; type1;	Sim	Sim	Sim	Sim	Sim
188612	type10; type4;	Sim	Sim	Sim	Sim	Sim
18905	type6; type5;	Sim	Sim	Sim	Sim	Sim
20137	type3; type2;	Sim	Sim	Sim	Sim	Sim
2133	type10; type1;	Sim	Sim	Sim		Sim
22506	type10; type1;	Sim	Sim	Sim	Sim	Sim
2328	type10; type1;	Sim	Sim	Sim	Sim	Sim
23308	type10; type1;	Sim	Sim	Sim		Sim
23496	type10; type1;	Sim	Sim	Sim		Sim
25401	type10; type1;	Sim	Sim	Sim	Sim	Sim
2697	type10; type1;	Sim	Sim	Sim	Sim	Sim
2852	type10; type1;	Sim	Sim	Sim	Sim	Sim
2876	type10; type1;	Sim	Sim	Sim	Sim	Sim
291554	type10; type1;	Sim	Sim	Sim		Sim
311082	type2;	Sim	Sim	Sim	Sim	Sim
31754	type10; type1;	Sim	Sim	Sim	Sim	Sim
318909	type10; type1;	Sim	Sim	Sim		Sim
320532	type10; type1;	Sim	Sim	Sim	Sim	Sim
32076	type10; type1;	Sim	Sim	Sim	Sim	Sim
32135	type10; type1;	Sim	Sim	Sim	Sim	Sim
321994	type10; type1;	Sim	Sim	Sim	Sim	Sim
32203	type10; type1;	Sim	Sim	Sim	Sim	Sim
32485	type10; type1;	Sim	Sim	Sim	Sim	Sim
32614	type1;	Sim	Sim	Sim	Sim	Sim
328563	type10; type1;	Sim	Sim	Sim	Sim	Sim
329234	type10; type1;	Sim	Sim	Sim		Sim
336232	type10; type1;	Sim	Sim	Sim		Sim
35839	type10; type1;	Sim	Sim	Sim	Sim	Sim
37234	type10; type1;	Sim	Sim	Sim		Sim
38253	type10; type1;	Sim	Sim	Sim		Sim
40499	type10; type1;	Sim	Sim	Sim		Sim
41257	type10; type1;	Sim	Sim	Sim	Sim	Sim
41272	type10; type1;	Sim	Sim	Sim		Sim
41522	type10; type1;	Sim	Sim	Sim	Sim	Sim
41762	type10; type1;	Sim	Sim	Sim		Sim
4182	type10; type1;	Sim	Sim	Sim	Sim	Sim
4298	type10; type1;	Sim	Sim	Sim	Sim	Sim
43509	type10; type1;	Sim	Sim	Sim		Sim
43522	type10; type1;	Sim	Sim	Sim		Sim

43532	type10; type1;	Sim	Sim	Sim	Sim	Sim
43534	type10; type1;	Sim	Sim	Sim		Sim
43553	type10; type1;	Sim	Sim	Sim		Sim
43564	type10; type1;	Sim	Sim	Sim		Sim
43565	type10; type1;	Sim	Sim	Sim	Sim	Sim
43566	type10; type1;	Sim	Sim	Sim		Sim
43571	type10; type1;	Sim	Sim	Sim		Sim
43843	type10; type1;	Sim	Sim	Sim		Sim
4428	type10; type1;	Sim	Sim	Sim		Sim
45751	type10; type1;	Sim	Sim	Sim		Sim
49936	type10; type1;	Sim	Sim	Sim		Sim
50064	type10; type1;	Sim	Sim	Sim	Sim	Sim
50455	type10; type1;	Sim	Sim	Sim		Sim
53239	type10; type1;	Sim	Sim	Sim	Sim	Sim
5334	type10; type1;	Sim	Sim	Sim	Sim	Sim
5884	type10; type1;	Sim	Sim	Sim		Sim
57	type9;	Sim	Sim	Sim	Sim	Sim
58	type9; type8;	Sim	Sim	Sim	Sim	Sim
59	type9; type8;	Sim	Sim	Sim	Sim	Sim
60	type9; type8;	Sim	Sim	Sim	Sim	Sim
2936	type1;	Sim	Sim	Sim	Sim	Sim
3018	type2;	Sim	Sim	Sim	Sim	Sim
32323	type10; type1;	Sim	Sim	Sim	Sim	Sim
336499	type10; type4;	Sim	Sim	Sim	Sim	Sim
36205	type10; type1;	Sim	Sim	Sim	Sim	Sim
50396	type1;	Sim	Sim	Sim	Sim	
53293	type1;	Sim	Sim	Sim	Sim	Sim
53320	type1;	Sim	Sim	Sim	Sim	Sim
23303	type2;	Sim	Sim	Sim	Sim	Sim
19531	type1;	Sim	Sim	Sim		
23467	type1;	Sim	Sim	Sim	Sim	Sim
2930	type1;	Sim	Sim	Sim	Sim	Sim
31850	type1;	Sim	Sim	Sim		
32194	type1;	Sim	Sim	Sim		
46268	type10; type1;	Sim	Sim	Sim	Sim	Sim
5366	type10; type1;	Sim	Sim	Sim	Sim	Sim
7657	type10; type1;	Sim	Sim	Sim	Sim	Sim
2191	type1;	Sim	Sim	Sim		Sim
23203	type10; type1;	Sim	Sim	Sim	Sim	Sim
23513	type10; type1;	Sim	Sim	Sim		Sim
2589	type1;	Sim	Sim	Sim	Sim	Sim
2990	type1;	Sim	Sim	Sim	Sim	Sim
31792	type1;	Sim	Sim	Sim	Sim	
32291	type1;	Sim	Sim	Sim	Sim	Sim
3293	type1;	Sim	Sim	Sim		Sim

341877	type10; type1;	Sim	Sim	Sim		Sim
46680	type1;	Sim	Sim	Sim		
7552	type10; type1;	Sim	Sim	Sim	Sim	Sim
7561	type10; type1;	Sim	Sim	Sim	Sim	Sim
227543	type10; type1;	Sim		Sim	Sim	Sim
291549	type10; type1;	Sim		Sim	Sim	Sim
328790	type10; type1;	Sim		Sim	Sim	Sim
35778	type10; type1;	Sim		Sim	Sim	Sim
108559	type1;	Sim	Sim	Sim	Sim	
23297	type10; type1;	Sim		Sim		Sim
23307	type10; type1;	Sim		Sim		Sim
23348	type10; type1;	Sim	Sim	Sim		Sim
23358	type10; type1;	Sim		Sim		Sim
23380	type1;	Sim		Sim	Sim	
23443	type10; type1;	Sim	Sim	Sim		Sim
23454	type10; type1;	Sim		Sim		Sim
23460	type10; type1;	Sim		Sim		Sim
23477	type10; type1;	Sim		Sim		Sim
23481	type10; type1;	Sim		Sim		Sim
23523	type10; type1;	Sim		Sim		Sim
23553	type10; type1;	Sim		Sim		Sim
23559	type10; type1;	Sim		Sim		Sim
2706	type10; type1;	Sim	Sim	Sim		Sim
289171	type10; type1;	Sim		Sim		Sim
3027	type1;	Sim		Sim	Sim	
318526	type1;	Sim		Sim	Sim	Sim
321881	type10; type1;	Sim	Sim	Sim		Sim
3236	type1;	Sim		Sim	Sim	
329232	type10; type1;	Sim		Sim		Sim
3300	type10; type1;	Sim		Sim		Sim
37480	type10; type1;	Sim		Sim		Sim
40077	type10; type1;	Sim		Sim		Sim
40548	type10; type1;	Sim		Sim		Sim
40570	type10; type1;	Sim		Sim		Sim
40735	type10; type1;	Sim		Sim		Sim
41012	type10; type1;	Sim		Sim		Sim
41703	type1;	Sim		Sim	Sim	Sim
4228	type10; type1;	Sim		Sim		Sim
4329	type10; type1;	Sim		Sim		Sim
4452	type10; type1;	Sim		Sim		Sim
4464	type10; type1;	Sim		Sim		Sim
45178	type10; type1;	Sim	Sim	Sim		Sim
45410	type10; type1;	Sim		Sim		Sim
45663	type10; type1;	Sim	Sim	Sim		Sim
45668	type10; type1;	Sim		Sim		Sim

45710	type10; type1;	Sim		Sim		Sim
46362	type10; type1;	Sim	Sim	Sim		Sim
46541	type10; type1;	Sim		Sim		Sim
49838	type10; type1;	Sim		Sim		Sim
49859	type10; type1;	Sim	Sim	Sim		Sim
50051	type10; type1;	Sim		Sim		Sim
50060	type10; type1;	Sim		Sim		Sim
53128	type1;	Sim		Sim	Sim	
5390	type1;	Sim		Sim		Sim
6485	type1;	Sim		Sim		Sim
7473	type10; type1;	Sim		Sim		Sim
7575	type10; type1;	Sim		Sim		Sim
329235	type10; type1;	Sim		Sim	Sim	Sim
41592	type1;	Sim		Sim	Sim	
4324	type1;	Sim		Sim	Sim	Sim
4618	type1;	Sim		Sim	Sim	Sim
23558	type10; type1;	Sim		Sim		Sim
330543	type10; type1;	Sim	Sim	Sim		Sim
353430	type10; type1;	Sim	Sim	Sim		Sim
41798	type10; type1;	Sim		Sim		Sim
46669	type10; type1;	Sim		Sim		Sim
5885	type10; type1;	Sim	Sim	Sim		Sim
1238	type4;	Sim	Sim		Sim	Sim
2960	type3; type2;	Sim	Sim	Sim	Sim	Sim
4234	type2;	Sim	Sim	Sim	Sim	Sim
48357	type3; type2;	Sim	Sim	Sim	Sim	Sim
544	type3; type2;	Sim	Sim	Sim	Sim	Sim
1998	type4;	Sim	Sim		Sim	Sim
43923	type6; type5;	Sim	Sim	Sim	Sim	Sim
53083	type2;	Sim	Sim	Sim	Sim	Sim
542574	type6;	Sim	Sim		Sim	Sim
162	type5;	Sim	Sim	Sim	Sim	Sim
163	type4;	Sim	Sim		Sim	Sim
164	type9; type8;	Sim	Sim		Sim	Sim
165	type7; type6;	Sim	Sim	Sim	Sim	Sim
166	type4;	Sim	Sim		Sim	Sim
19271	type3; type2;	Sim	Sim	Sim	Sim	Sim
31772	type3; type2;	Sim	Sim	Sim	Sim	Sim
31949	type6;	Sim	Sim	Sim	Sim	Sim
320967	type2;	Sim	Sim	Sim		Sim
32109	type6;	Sim	Sim	Sim	Sim	Sim
36096	type3; type2;	Sim	Sim	Sim		Sim
39855	type2;	Sim	Sim	Sim	Sim	Sim
39956	type6; type5;	Sim	Sim	Sim	Sim	Sim
46276	type3; type2;	Sim	Sim	Sim	Sim	Sim

52904	type2;	Sim	Sim	Sim		Sim
177	type5;	Sim	Sim			Sim
111841	type9; type8;	Sim	Sim		Sim	Sim
28529	type9; type8;	Sim	Sim		Sim	Sim
31660	type9;	Sim	Sim		Sim	Sim
31790	type9;	Sim	Sim			Sim
31882	type6;	Sim			Sim	Sim
31890	type9; type8;	Sim	Sim		Sim	Sim
31967	type9;	Sim			Sim	Sim
32020	type9;	Sim	Sim			Sim
32041	type9; type7;	Sim	Sim		Sim	Sim
32083	type9;	Sim	Sim		Sim	Sim
32340	type9; type8;	Sim	Sim		Sim	Sim
32356	type9;	Sim			Sim	Sim
32425	type9; type8;	Sim	Sim		Sim	Sim
32538	type9; type8;	Sim	Sim		Sim	Sim
343908	type6;	Sim			Sim	Sim
431726	type9;	Sim	Sim		Sim	Sim
43897	type9; type8;	Sim	Sim		Sim	Sim
46184	type9; type8;	Sim	Sim		Sim	Sim
547814	type9; type8;	Sim	Sim		Sim	Sim
552568	type9; type8;	Sim	Sim		Sim	Sim
552575	type9; type8;	Sim	Sim		Sim	Sim
199	type5;	Sim	Sim			Sim
200	type6;	Sim			Sim	Sim
201	type6;	Sim			Sim	Sim
202	type5;	Sim	Sim		Sim	Sim
203	type6; type5;	Sim			Sim	Sim
204	type6; type5;	Sim			Sim	Sim
205	type6;	Sim			Sim	Sim
206	type6;	Sim	Sim		Sim	Sim
207	type6;	Sim			Sim	Sim
208	type5;	Sim	Sim		Sim	Sim
209	type9;	Sim	Sim			Sim
210	type6; type5;	Sim			Sim	Sim
211	type9;	Sim			Sim	Sim
212	type9; type8;	Sim	Sim			Sim
19318	type9; type8;	Sim	Sim			Sim
316726	type9; type8;	Sim	Sim			Sim
31969	type9; type8;	Sim	Sim			Sim
32506	type9; type8;	Sim	Sim			Sim
46988	type9; type8;	Sim	Sim			Sim
892	Does not belong to any type;	Sim	Sim			Sim
44764	type2;		Sim		Sim	Sim

2828	type1;		Sim		Sim	Sim
2474	type1;		Sim		Sim	Sim
4218	type1;		Sim		Sim	Sim
24651	type1;		Sim		Sim	Sim
53187	type1;		Sim		Sim	Sim
32763	type1;		Sim		Sim	Sim
53366	type1;		Sim		Sim	Sim
49841	type1;		Sim		Sim	Sim
22747	type1;		Sim		Sim	Sim
31887	type1;		Sim		Sim	Sim
49895	type1;		Sim		Sim	Sim
7525	type1;		Sim		Sim	Sim
49937	type1;		Sim		Sim	Sim
32188	type2;		Sim		Sim	Sim
289166	type1;		Sim		Sim	Sim
18598	type2;		Sim		Sim	Sim
2484	type2;		Sim		Sim	Sim
18766	type1;		Sim		Sim	Sim
44479	type6; type5;		Sim		Sim	Sim
2538	type1;		Sim		Sim	Sim
46492	type1;		Sim		Sim	Sim
42178	type1;		Sim		Sim	Sim
28393	type2;		Sim		Sim	Sim
49832	type1;		Sim		Sim	Sim
3376	Does not belong to any type;		Sim		Sim	Sim
2914	type1;		Sim		Sim	Sim
4215	type1;		Sim		Sim	Sim
7434	type1;		Sim		Sim	Sim
46675	type2;		Sim		Sim	Sim
345745	type1;		Sim		Sim	Sim
49767	type1;		Sim		Sim	Sim
23329	type1;		Sim		Sim	Sim
53364	type1;		Sim		Sim	Sim
23177	type3; type2;		Sim		Sim	Sim
33715	type1;		Sim		Sim	Sim
2688	type1;		Sim		Sim	Sim
32233	type3; type2;		Sim		Sim	Sim
324239	type1;		Sim		Sim	Sim
33496	type2;		Sim		Sim	Sim
311174	type2;		Sim		Sim	Sim
46943	type1;		Sim		Sim	Sim
47415	type1;		Sim		Sim	Sim
289169	type1;		Sim		Sim	Sim
18796	type2;		Sim		Sim	Sim

45431	type1;		Sim		Sim	Sim
4214	type1;		Sim		Sim	Sim
887	type2;		Sim		Sim	Sim
31823	Does not belong to any type;		Sim		Sim	Sim
28997	type1;		Sim		Sim	Sim
44999	type1;		Sim		Sim	Sim
39892	type2;		Sim		Sim	Sim
289156	type1;		Sim		Sim	Sim
47243	type1;		Sim		Sim	Sim
38258	type2;		Sim		Sim	Sim
32667	type1;		Sim		Sim	Sim
28679	type5;		Sim		Sim	Sim
487	type2;		Sim		Sim	Sim
98982	type1;		Sim		Sim	Sim
7598	type1;		Sim		Sim	Sim
1599	type2;		Sim		Sim	Sim
42137	type2;		Sim		Sim	Sim
7676	type1;		Sim		Sim	Sim
4221	type1;		Sim		Sim	Sim
32156	type1;		Sim		Sim	Sim
7381	type2;		Sim		Sim	Sim
1241	type3; type2;		Sim		Sim	Sim
2414	type1;		Sim		Sim	Sim