Lisbon School of Economics & Management
Universidade de Lisboa

MESTRADO

MÉTODOS QUANTITATIVOS PARA DECISÃO ECONÓMICA E EMPRESARIAL

TRABALHO FINAL DE MESTRADO

PROJETO

PREVISÃO DAS SAÍDAS POR INICIATIVA DO COLABORADOR

PROJETO APLICADO A UMA EMPRESA DE SEGUROS

TIAGO FILIPE PINTO REBELO

ORIENTAÇÃO: PROF. DR. PAULO PARENTE

DOCUMENTO ESPECIALMENTE ELABORADO PARA OBTENÇÃO DO GRAU DE MESTRE

OUTUBRO DE 2024

Agradecimentos

Gostaria de começar por expressar o meu agradecimento ao meu orientador, professor Paulo Parente, pela disponibilidade, apoio e orientação prestados ao longo da realização deste trabalho final de mestrado.

Expresso a minha profunda gratidão à Responsável de Compensação e Benefícios na empresa X, pela sugestão do tema deste projeto, pela disponibilização dos dados e por toda a disponibilidade demonstrada. Agradeço igualmente a toda a Direção de Recursos Humanos pela contínua colaboração com a Responsável de Compensação e Benefícios na partilha desses dados.

De seguida, agradeço à minha família, cujo apoio incondicional tornou tudo isto possível. Sou imensamente grato por me terem incentivado sempre a perseguir os meus sonhos e por me proporcionarem os recursos necessários para alcançar os meus objetivos.

Por fim, um agradecimento muito especial à minha namorada, cujo carinho e motivação constantes foram fundamentais para a conclusão deste projeto dentro do prazo.

Resumo

O *turnover* voluntário é um dos grandes desafios enfrentados pelos recursos humanos e pelas respetivas organizações, principalmente devido aos custos associados, como recrutamento, formação e perda de conhecimento organizacional. Nesse sentido, este projeto tem como objetivo desenvolver um modelo de *machine learning*, para uma empresa do setor segurador, que identifique quais os colaboradores com maior probabilidade de saírem voluntariamente, assim como as variáveis que mais influenciam essa decisão, fornecendo à empresa uma ferramenta importante na prevenção do *turnover* e de implementação de estratégias de retenção.

A metodologia adotada foi o CRISP-DM que é composta por seis fases: compreensão do negócio, compreensão dos dados, preparação dos dados, modelação, avaliação e implementação. O conjunto de dados era composto por 1121 observações das quais 92 eram saídas. Para lidar com o desbalanceamento das classes, aplicaram-se técnicas de *oversampling* e *undersampling*. Para encontrar os melhores hiperparâmetros de cada algoritmo foi utilizado o método RandomizedSearchCV.

O conjunto de dados foi dividido em 80% para treino e 20% para teste, utilizando validação cruzada *k-fold* com 5 *folds* (subconjuntos) no conjunto de treino. Foram avaliados cinco modelos de classificação: Regressão Logística, Árvore de Decisão, Floresta Aleatória, XGBoost e AdaBoost. A Floresta Aleatória destacou-se como o melhor modelo, com precisão e sensibilidade de 82%, e especificidade de 98%, no conjunto de teste. As variáveis mais relevantes para a previsão de *turnover* foram, por ordem de importância, idade, promoções, antiguidade e vma (remuneração fixa anual).

O modelo foi utilizado para criar diferentes níveis de risco de saída, permitindo à empresa aplicar estratégias proativas de retenção. Este trabalho contribui para a utilização de ferramentas preditivas na gestão de recursos humanos, oferecendo uma abordagem prática para reduzir o turnover e os custos a ele associados.

Palavras-Chave: *Turnover* voluntário, *Machine Learning*, Floresta Aleatória, Recursos Humanos, CRISP-DM

Abstract

Voluntary turnover is one of the major challenges faced by human resources and their organisations, mainly due to the associated costs, such as recruitment, training and loss of organisational knowledge. With this in mind, this project aims to develop a machine learning model for a company in the insurance sector that identifies which employees are most likely to leave voluntarily, as well as the variables that most influence this decision, providing the company with an important tool for preventing turnover and implementing retention strategies.

The methodology adopted was CRISP-DM, which consists of six phases: business understanding, data understanding, data preparation, modelling, evaluation and deployment. The data set consisted of 1121 observations of which 92 were voluntary leaves. To deal with imbalance data, oversampling and undersampling techniques were applied. The RandomisedSearchCV method was used to find the best hyperparameters for each algorithm.

The data set was divided into 80% for training and 20% for testing, using k-fold cross-validation with 5 folds in the training set. Five classification models were evaluated: Logistic Regression, Decision Tree, Random Forest, XGBoost and AdaBoost. Random Forest stood out as the best model with 82% of precision and sensitivity in the test sample. The most relevant variables for predicting turnover were, in order of importance, age, promotions, seniority and annual fixed remuneration.

The model was used to create different levels of risk of leaving, allowing the company to apply proactive retention strategies. This work contributes to the use of predictive tools in human resources management, offering a practical approach to reducing turnover and its associated costs.

Keywords: Voluntary Turnover, Machine Learning, Random Forest, Human Resources, CRISP-DM

Índice

Índice de Figuras

Índice de Tabelas

Lista de Abreviaturas

ADASYN: *Adaptive Synthetic Sampling*

AUC: *Area Under the Curve* (Área abaixo da curva ROC)

CRISP-DM: *Cross-Industry Standard Process for Data Mining*

DT: *Decison Tree* (Árvore de Decisão)

FN: Falsos Negativos

FP: Falsos Positivos

ID: Identificação

KNN: *K-Nearest Neighbors*

LR: *Logistic Regression* (Regressão Logística)

ML: *Machine Learning*

PCCC: Percentagem de Casos Classificados Corretamente

RF: *Random Forest* (Floresta Aleatória)

ROC: *Receiver Operating Characteristic*

SMOTE: *Synthetic Minority Over-sampling Technique*

VN: Verdadeiros Negativos

VP: Verdadeiros Positivos

XGB: *XGBoost*

1. Introdução

Nos últimos anos, a retenção de talentos e a redução do *turnover* voluntário tornaram-se questões cruciais para as organizações que atuam num mercado de trabalho cada vez mais competitivo, como é o caso do mercado segurador. O *turnover* voluntário, além de afetar o funcionamento das empresas, representa também um custo significativo em termos de recrutamento, formação de novos colaboradores e perda de conhecimento organizacional (Cascio, 2006).

Nesse sentido, o uso de técnicas de *machine learning* para a previsão de saídas voluntárias, tem-se mostrado uma abordagem bastante útil, para antecipar essa intenção de saída, de modo a criar estratégias de retenção. O principal objetivo deste projeto é desenvolver um modelo preditivo, capaz de identificar colaboradores com intenção de pedir demissão e compreender os principais fatores que influenciam essa decisão, no contexto real de uma empresa do setor de seguros, que por razões de segurança solicitou confidencialidade, sendo referida ao longo deste trabalho como "empresa X".

A estrutura do projeto divide-se em oito capítulos principais, incluindo esta introdução que corresponde ao primeiro capítulo. O segundo capítulo apresenta uma revisão da literatura sobre o *turnover* voluntário e as suas implicações nos custos e funcionamento de uma empresa, seguido, no capítulo 3, de uma análise teórica dos modelos de *machine learning* utilizados na análise dos dados. As ferramentas utilizadas são abordadas no capítulo 4, seguindo-se da descrição e aplicação da Metodologia CRISP-DM, no capítulo 5, que foi a metodologia adotada. Este capítulo dá especial ênfase à preparação dos dados, incluindo a exploração e o tratamento de variáveis. Esta preparação é fundamental para a modelação e avaliação dos algoritmos selecionados, Regressão Logística, Árvores de Decisão, Floresta Aleatória, XGBoost e AdaBoost.

No fim, os resultados do projeto são analisados no sexto capítulo, e as conclusões e recomendações são apresentadas no capítulo 7, destacando as limitações do trabalho e possíveis direções para continuação do projeto aqui elaborado. Espera-se que este trabalho contribua de forma significativa para reduzir o *turnover* da empresa, fornecendo ferramentas que ajudem a antecipar e identificar possíveis saídas, permitindo a criação antecipada de estratégias de retenção. Além disso, procura oferecer uma compreensão dos fatores que influenciam o *turnover* dos colaboradores, contribuindo de forma relevante

para a gestão de recursos humanos da empresa.

2. Revisão de Literatura

O *turnover* voluntário é definido por (Shaw et al., 2005), como a saída de um colaborador da organização pela rescisão do seu contrato.

O *turnover* voluntário de colaboradores é uma das maiores perdas de valor para uma empresa, resultando em elevados custos tanto diretos como indiretos. Esses custos englobam todas as despesas associadas à saída de um colaborador, como tempo e recursos gastos em procedimentos administrativos para encerrar o contrato de trabalho, gastos com novos recrutamentos para encontrar substitutos, investimentos em formação e integração de novos colaboradores, além de perda de produtividade devido à saída de colaboradores experientes e à perda de conhecimento dentro da organização (Cascio, 2006).

A capacidade de prever a saída de colaboradores pode fornecer contributos importantes para a gestão de recursos humanos, permitindo a implementação de estratégias de retenção mais eficazes. Em setores como o de seguros, onde o capital humano é um ativo essencial, compreender as variáveis que influenciam o *turnover* voluntário é crucial para desenvolver estratégias de retenção eficazes (Hom et al., 2017).

A literatura indica que o *turnover* pode ser influenciado por uma série de fatores individuais, organizacionais e de mercado (Holtom et al., 2008).

Apesar das entrevistas de saída serem uma prática comum, utilizada pelas empresas, muitas vezes não conseguem identificar as verdadeiras causas da rotatividade, tornando difícil a implementação de estratégias eficazes de retenção (Hom et al., 2017). Essa dificuldade reflete a complexidade do fenómeno, que é multifatorial e não é facilmente captado por métodos tradicionais de avaliação de satisfação (Marchington & Wilkinson, 2020).

Fatores individuais, como a satisfação no trabalho, o comprometimento organizacional e as perceções de justiça, são frequentemente identificados como determinantes importantes do *turnover* voluntário. Segundo a teoria das expectativas de Vroom (1964), a motivação dos colaboradores é afetada pela perceção de que o seu esforço resultará em recompensas. Quando existem discrepâncias entre as expectativas e a realidade, pode surgir insatisfação e um aumento da intenção de *turnover*, especialmente

quando os colaboradores percebem que o seu esforço não está a ser devidamente recompensado (Isaac et al., 2021).

Zhao et al. (2019) demonstraram que fatores relacionados com a remuneração têm uma correlação significativa com o *turnover*, reforçando a importância da sua inclusão nos modelos preditivos.

A antiguidade e a idade também podem exercer papéis significativos no *turnover*. Colaboradores com alguns anos de serviço, se não encontrem oportunidades de progressão na carreira, podem procurar alternativas externas (Griffeth et al., 2000). De forma semelhante, a idade pode influenciar o *turnover*. Colaboradores mais jovens costumam procurar maior mobilidade e crescimento profissional, enquanto os colaboradores mais velhos costumam priorizar a estabilidade (Henneberger & Sousa-Poza, 2002).

Do ponto de vista organizacional, a cultura corporativa, as práticas de recursos humanos e as oportunidades de crescimento e desenvolvimento, desempenham papéis cruciais na decisão dos colaboradores de permanecerem ou saírem da empresa. Arthur (1994) defende que práticas de gestão de recursos humanos, que promovem o desenvolvimento de carreiras e a equidade salarial podem reduzir o *turnover*, resultando num ambiente de trabalho mais satisfatório.

No contexto das seguradoras, fatores organizacionais como a pressão para atingir metas, a complexidade regulatória e o impacto emocional de lidar com sinistros podem contribuir para o *turnover*. Um estudo de Allen et al. (2010) destacou que as seguradoras enfrentam desafios únicos ligados ao stress no trabalho e à necessidade de equilibrar metas corporativas com o bem-estar dos colaboradores.

Fatores externos, como as condições do mercado de trabalho e da economia, também têm um papel importante na decisão de um colaborador permanecer ou abandonar a organização. Durante períodos de baixa taxa de desemprego, como indicado por Holtom et al. (2008), o *turnover* voluntário tende a aumentar, uma vez que os colaboradores têm mais oportunidades para procurar empregos alternativos. Em setores como o de serviços financeiros e seguros, onde a procura por profissionais qualificados é elevada, as ofertas de emprego externas tornam-se particularmente atrativas.

A distância até ao local de trabalho também tem sido identificada como um fator que pode aumentar a insatisfação dos colaboradores e, consequentemente, a intenção de *turnover*. Eidt (2023) destacou que longas distâncias são um fator significativo que contribui para a insatisfação dos colaboradores e que o teletrabalho pode ajudar a atenuar esses efeitos negativos. Eidt (2023) observa que a maioria dos estudos incluídos na sua revisão, concluíram que o teletrabalho tende a ter efeitos positivos na motivação intrínseca e a diminuir a intenção de *turnover*, melhorando assim a satisfação no trabalho. A flexibilidade oferecida pelo trabalho remoto pode assim, reduzir significativamente a intenção de *turnover*, pois proporciona um melhor equilíbrio entre a vida pessoal e profissional.

Embora o setor dos seguros seja tradicionalmente baseado em interações interpessoais, a adoção de tecnologias emergentes, como o *machine learning*, tem sido considerada como uma solução promissora para a retenção de talentos. Segundo Wang et al. (2020), a aplicação de algoritmos como Floresta Aleatória e Gradient Boosting para prever o *turnover* tem-se mostrado bem-sucedida em empresas de serviços, permitindo que os gestores tomem medidas proativas para reduzir a saída de colaboradores.

Zhao et al. (2019) focaram-se na análise e avaliação da capacidade de alguns métodos de *machine learning* para prever o *turnover* dos colaboradores, incluindo uma árvore de decisão (DT), uma floresta aleatória (RF), regressão logística (LR) e extreme gradient boosting (XGB).

Estes modelos utilizam variáveis como idade, antiguidade, salário e satisfação no trabalho, entre outros fatores, para calcular a probabilidade de um colaborador abandonar a organização. Isto permite que os gestores implementem medidas preventivas, como ajustes salariais ou melhorias nas condições de trabalho, antes que a saída se concretize.

O *turnover* voluntário de colaboradores é assim um desafio significativo para muitas empresas, resultando em elevados custos diretos e indiretos para as organizações. A compreensão dos fatores que influenciam a decisão dos colaboradores permanecerem ou saírem da empresa, assim como a previsão de potenciais colaboradores em risco de saída, são fundamentais para a elaboração de estratégias eficazes de retenção.

3. Contexto teórico dos modelos de *machine learning*

3.1. Regressão Logística

A regressão logística é um método de classificação tradicional que se baseia em discriminantes lineares, originalmente proposto em 1958 por Cox (Zhao et al., 2019).

"Com base no valor da probabilidade, o modelo cria uma fronteira linear que separa o espaço de entrada em duas regiões. A regressão logística é fácil de implementar e funciona bem em classes linearmente separáveis, o que o torna um dos classificadores mais utilizados" (Raschka (2015) citado por Zhao et al. (2019)).

Segundo Raschka (2015), a Regressão Logística utiliza a função sigmoide, que transforma a saída linear do modelo numa probabilidade que varia entre 0 e 1, permitindo a tomada de uma decisão binária com base no limiar de decisão (*threshold*), normalmente de 0,5.

O método baseia-se na estimativa por máxima verosimilhança dos coeficientes do modelo (Belyadi e Haghighat, 2021).

A Regressão Logística oferece, assim, uma forma eficaz de prever a probabilidade de um evento binário, como a saída de um colaborador. A função de probabilidade da regressão logística que define este modelo é representada pela seguinte equação:

$$p(X) = \frac{e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}{1+e^{\beta_0+\beta_1 X_1+\cdots+\beta_p X_p}}.$$

Onde:

- $p(X)$ é a probabilidade de saída do colaborador;
- $X_1, X_2, \ldots, X_p$ representam os preditores (por exemplo, idade, antiguidade, etc.);
- β_0 é o termo constante (intercepto);
- $\beta_1, \beta_2, \ldots, \beta_p$ são os coeficientes associados a cada preditor.

3.2. Árvores de Decisão

As Árvores de Decisão são frequentemente utilizadas em problemas de classificação e regressão, destacando-se pela sua simplicidade e pela facilidade de interpretação através de representações gráficas intuitivas. Este método de *machine learning*, que foi introduzido pela primeira vez por Morgan e Sonquist em 1963, utiliza uma estrutura semelhante a uma árvore para construir modelos de classificação (Zhao et al., 2019).

A estrutura da árvore consiste em nós de decisão e folhas, onde cada nó representa uma pergunta sobre uma característica do conjunto de dados, e cada folha representa a classe ou o valor de saída final (Breiman et al., 1984). A construção de uma árvore consiste em dividir repetidamente o conjunto de dados em subconjuntos menores, utilizando critérios específicos como o índice de Gini ou a entropia. O objetivo destes critérios é diminuir a impureza dos dados em cada divisão, de forma a reduzir a impureza dos dados ao longo das divisões (Hastie et al., 2009). Impureza, neste caso, é o grau de mistura das diferentes classes em um nó da árvore, o objetivo é que os nós apresentem maior pureza para garantir uma maior precisão.

No entanto, as Árvores de Decisão enfrentam limitações, como a falta de robustez em situações de alta variância, onde pequenas alterações nos dados de entrada podem afetar significativamente a estrutura da árvore e, consequentemente, a capacidade preditiva do modelo (Zhao et al., 2019). Apesar de serem fáceis de interpretar e implementar, as suas previsões podem não ser tão competitivas em comparação com modelos mais complexos.

3.3. Floresta Aleatória

A Floresta Aleatória, proposta por Breiman (2001), é um método de ensemble, ou seja, um método que combina previsões de vários outros modelos, neste caso, que utiliza uma combinação de várias árvores de decisão. Este modelo aplica a técnica de bagging (*bootstrap aggregating*). Na técnica de bagging, cada modelo aprende com o erro gerado pelo modelo anterior, utilizando um subconjunto ligeiramente diferente do conjunto de dados de treino, o que diminui a variância e minimiza o *overfitting* (Alhamid, 2021). *Overfitting* ocorre quando um modelo se ajusta demasiado aos dados de treino e não

consegue generalizar para novos dados, captando ruídos ou padrões específicos. Assim, a Floresta Aleatória ao aplicar a técnica de bagging, lida bem com o *overfitting* e aumenta a precisão das previsões.

O modelo faz uma agregação por votação para problemas de classificação, aumentando assim a robustez das previsões. Esta abordagem, reduz assim o risco de *overfitting*, que é uma das limitações das árvores de decisão.

Processo de Construção da Floresta Aleatória, representado na Figura 1:

1. Amostragem com Substituição (*Bootstrap*): a Floresta Aleatória utiliza o processo de amostragem com substituição, conhecido como *bootstrap*, para criar subconjuntos de dados de treino em cada iteração do modelo. Este método, descrito por Liaw e Wiener (2002), permite a criação de várias árvores de decisão a partir de diferentes amostras do conjunto de dados original.

2. Seleção Aleatória de Atributos: ao invés de considerar todos os atributos em cada nó, a Floresta Aleatória seleciona aleatoriamente um subconjunto de atributos. Isso aumenta a diversidade entre as árvores e diminui a correlação entre elas, resultando num modelo mais robusto (Hastie et al., 2009).

3. Agregação por Votação (para classificação): as previsões das árvores são agregadas, através da utilização da votação maioritária (para classificação) (Hastie et al., 2009). Em situações de classificação binária, a classe que receber mais votos é atribuída como a previsão final.

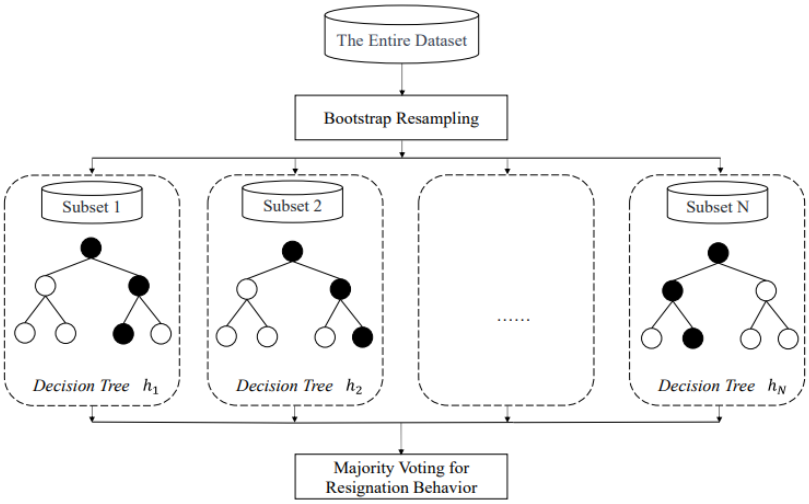


Figura 1 – Floresta Aleatória Ilustração (Liu et al., 2024)

3.4. XGBoost

O XGBoost, ou Extreme Gradient Boosting, introduzido por Chen (2014), é um algoritmo de *machine learning* muito utilizado que se tem vindo a destacar pela sua eficiência e performance.

Baseado na técnica de boosting, o XGBoost constrói modelos de forma sequencial, onde a cada iteração, o modelo seguinte corrige os erros do modelo anterior (Chen & Guestrin, 2016). A técnica de boosting consiste em combinar vários classificadores fracos em série, com o objetivo de formar um classificador forte, mais robusto e preciso.

Foi desenvolvido para ser altamente eficiente em termos de recursos computacionais (Chen & Guestrin, 2016). O XGBoost utiliza um termo de regularização para penalizar a complexidade do modelo e reduzir o efeito de *overfitting*, o que resulta numa previsão melhor e com tempos de execução muito mais rápidos (Zhao et al. 2019). Isso permite que o modelo se ajuste de forma flexível aos dados, mantendo um bom equilíbrio entre viés e variância.

É particularmente eficaz em problemas de classificação e regressão. O XGBoost utiliza um método de ensemble baseado em árvores, onde a cada iteração, uma nova árvore é adicionada ao modelo. Dessa forma, combina o erro de previsão e uma penalização pela complexidade do modelo (Chen & Guestrin, 2016).

3.5. AdaBoost

O AdaBoost, ou Adaptive Boosting, desenvolvido por Freund & Schapire (1997), é um algoritmo de ensemble baseado na técnica de boosting.

Este algoritmo é bastante utilizado devido ao facto de ser simples e preciso em termos de classificação, bem como pela sua capacidade de generalização (Schapire, 2003).

O funcionamento do AdaBoost consiste num ciclo no qual, a cada iteração, um novo classificador é treinado no conjunto de dados. Os erros de previsão do classificador anterior influenciam a distribuição de pesos das amostras, de modo que as amostras que foram mal classificadas recebam mais peso, enquanto as amostras bem classificadas

ficam com um peso reduzido (Freund & Schapire, 1997). Esse mecanismo de ajuste contínuo permite que o AdaBoost se concentre em casos difíceis, de forma a aumentar a precisão global do modelo.

A combinação de vários classificadores permite que o AdaBoost construa um modelo robusto. O processo contínuo de adição de classificadores fracos faz com que a precisão aumente até um certo ponto, embora exista o risco de *overfitting* caso o número de iterações seja excessivo (Freund & Schapire, 1997).

Resumidamente, o AdaBoost é uma solução eficaz para problemas de classificação, pois melhora continuamente a performance ao longo das iterações, aumentando a precisão do modelo à medida que treina diversos classificadores fracos, mas é menos ajustado para problemas que lidam com *overfitting*.

Todos os métodos aqui apresentados são facilmente implementados em Python, utilizando bibliotecas como Scikit-learn.

4. Ferramentas

Para o desenvolvimento do projeto, a base de dados foi criada utilizando o Microsoft Excel. Com a criação da base de dados, ainda em Excel, foram efetuadas diversas transformações nos dados, demonstradas na Tabela 1, para assegurar a qualidade e a consistência das informações.

Todas as análises, transformações dos dados pós Excel e a aplicação dos modelos de *machine learning*, foram realizadas na plataforma Anaconda Navigator, utilizando o ambiente *open-source* Jupyter Notebooks e respetivas bibliotecas.

5. Metodologia

A metodologia utilizada foi o CRISP-DM (Cross Industry Standard Process for Data Mining), que é um dos métodos mais utilizados para projetos de *data mining* e análise de dados. O ciclo que esta metodologia apresenta, fornece uma estrutura clara, que descreve os passos e fluxos de trabalho necessários na elaboração de um projeto (Sarkar, 2018).

Orientada por objetivos, é uma metodologia amplamente aceite em projetos de *data mining* que recorrem à aplicação de algoritmos de *machine learning* (Ayele, 2020).

A metodologia é composta por seis fases interativas que não seguem uma sequência linear. É comum avançar e retroceder entre as diferentes etapas, pois o resultado de cada fase pode influenciar a escolha da próxima etapa a ser desenvolvida (Chapman et al., 2000).

As diferentes fases e a forma como interagem entre si encontram-se representadas na Figura 2.

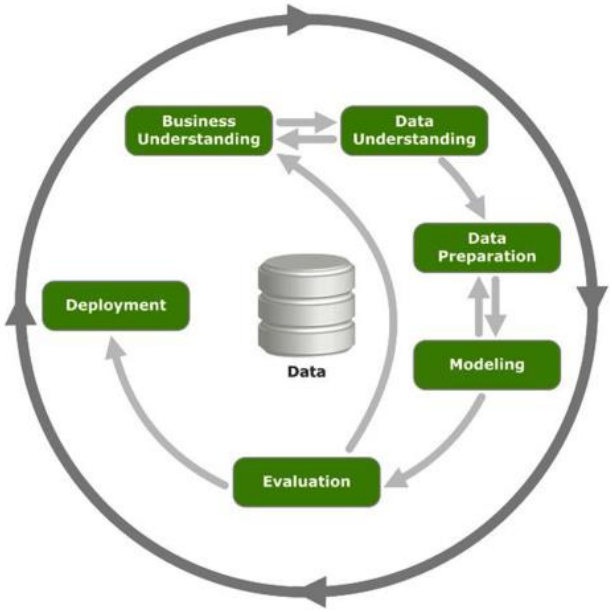


Figura 2 – Fases da metodologia CRISP-DM (retirado de Sarkar et al. (2018))

5.1. Compreensão do negócio

Na fase de compreensão do negócio, o analista deve compreender os objetivos do

negócio, definir e avaliar o problema, e estabelecer objetivos para o projeto.

O principal objetivo da empresa consiste em identificar os fatores determinantes da saída voluntária dos colaboradores e prever quais os colaboradores que podem vir a pedir demissão, atribuindo probabilidades de saída a cada colaborador. Assim, o departamento de recursos humanos pode antecipar potenciais saídas, identificar rapidamente as causas e desenvolver estratégias eficazes para reter esses colaboradores, reduzindo assim o *turnover* voluntário da empresa e os custos associados a esse *turnover*.

5.2. Compreensão dos dados

A segunda fase da metodologia diz respeito à recolha, descrição e identificação de problemas relacionados com os dados.

Este projeto utiliza uma base de dados composta por informações qualitativas e quantitativas, tanto pessoais como profissionais, dos colaboradores presentes na empresa à data de 30 de abril de 2024, assim como dos colaboradores que saíram entre janeiro de 2022 e outubro de 2024. No total a base de dados contém 1121 observações.

A empresa X utiliza a ferramenta Meta4, que é atualizada mensalmente com informações, à data, dos colaboradores ativos. Esta ferramenta permite fazer a extração desses dados para Excel e todos os ficheiros desde janeiro de 2022 foram disponibilizados para análise.

Os ficheiros contêm uma coluna com a data de saída, a qual só contém valor no mês em que a saída ocorre, e uma coluna que indica o motivo da saída. Assim, para compilar as informações relevantes, todos os ficheiros mensais foram agrupados por ano e filtrados pela data saída e pelo motivo “Iniciativa do Colaborador”. Esses dados foram então copiados para um único ficheiro Excel, resultando numa base de dados que inclui informações dos colaboradores no mês da sua saída.

Nesse mesmo ficheiro, foram adicionadas também as informações do Excel mensal referente ao mês de abril de 2024, que continha todos os colaboradores ativos nesse mês, data em que se iniciou o projeto. Ficou-se assim com a consolidação dos dados de colaboradores num único ficheiro Excel, contendo informações dos colaboradores ativos e de todos aqueles que saíram por iniciativa própria, à data da sua saída.

Foi adicionada uma nova coluna, com o nome “saída”, na qual o valor é 1 se o colaborador saiu e 0 se o colaborador permaneceu ativo. A partir de um outro ficheiro disponibilizado pela empresa, foi possível identificar quais os colaboradores que já haviam pedido demissão até outubro de 2024, assim como as respetivas datas de saída previstas. Foi então adicionado ao Excel da base de dados a data de saída desses colaboradores e o valor 1 na coluna “saída”.

Durante o processo de compilação dos dados, foram identificadas diversas colunas que não eram relevantes para o estudo, como email, número de telefone, NIF, entre outras, as quais foram eliminadas. Uma vez que se estava a criar a base de dados através de diversos ficheiros, foram logo detetadas algumas transformações a serem efetuadas após a recolha dos dados. Assim, após a recolha de dados foram realizadas algumas dessas transformações ainda em Excel, como mostra a Tabela 1, de forma a extrair informações mais relevantes para o estudo e facilitar o trabalho e análise posterior, no Python.

Ficou-se então com as seguintes variáveis:

Variável	Descrição / transformação inicial	Tipo	Nº Categorias	Valores Ausentes
ID	Identificação única do colaborador.	QN	1121	0
Saída	Variável dependente: 1 se o colaborador saiu por iniciativa própria, 0 caso contrário.	QN	2	0
ano_saída	Ano em que o colaborador saiu. Criada a partir da data de saída.	QN	3	1029
mês_saída	Mês em que o colaborador saiu. Criada a partir da data de saída.	QN	12	1029
Tipo Contrato	0 se "Contratado a termo certo", 1 se "Efetivo sem termo"	QN	2	0
empresa_origem	Ao longo dos anos a empresa teve algumas fusões e aquisições de outras empresas, pelo que esta variável indica em que empresa do grupo o colaborador tem origem.	QN	6	0
genero	0 se "Feminino", 1 se "Masculino".	QN	2	0
local_trab	Local onde o colaborador exerce as suas funções. Originalmente 49 locais diferentes, de forma a não tornar o modelo tão complexo, reduziu-se para 4 categorias: Porto, Lisboa, Açores e Outros	QN	4	0
ISENCAO_DE_HORARIO	1 se possui isenção de horário, 0 caso contrário.	QN	2	0
Categoria AE	Indica a fase da carreira do colaborador.	QN	21	0
JOB_GRADE	Indica o nível de importância e responsabilidade da função, com valores inteiros entre 10 e 24.	QO	15	0
NACIONALIDADE	Tinha 12 nacionalidades diferentes, transformou-se em 1 caso tenha nacionalidade portuguesa ou do país do grupo, 0 caso contrário	QN	2	0
Tem DEPENDENTES_IRS	0 se não tem dependentes para efeitos de IRS, 1 se tem.	QN	2	0
ESTADO_CIVIL	0 se solteiro, separado, divorciado ou viúvo; 1 se casado ou em união de facto.	QN	2	0
Job Family Cluster	Famílias de funções, clusters que agrupam as funções.	QN	11	0
Sales	1 se o colaborador está afeto à área de vendas, 0 caso contrário.	QN	2	0
VMA	Retribuição fixa, correspondente ao vencimento bruto anual do colaborador.	QC	-	0
idade	Criada a partir da data de nascimento do colaborador.	QC	-	0
antiguidade	Tempo de serviço na empresa. Criada a partir da data de entrada na empresa.	QC	-	0
work_dist	Distância, em km e por estrada, entre a residência e o local de trabalho, calculada através dos códigos postais.	QC	-	1
vme_caract	Caracterização do salário comparativamente aos pares internos: “baixo”, “normal” ou “alto”.	QO	3	0
nível_educ	Nível de educação, originalmente com 56 categorias, reduzido a 5 níveis de acordo com normas da empresa.	QO	5	0
promoções	Número de promoções. Criado através de um ficheiro de evolução salarial, é considerada promoção cada vez que o colaborador obteve um aumento salarial superior a 50€ de vencimento fixo mensal.	QD	-	0

Tabela 1 – Descrição das variáveis

Dicionário: QD – Quantitativa Discreta; QC – Quantitativa Contínua; QN – Qualitativa Nominal; QO – Qualitativa Ordinal

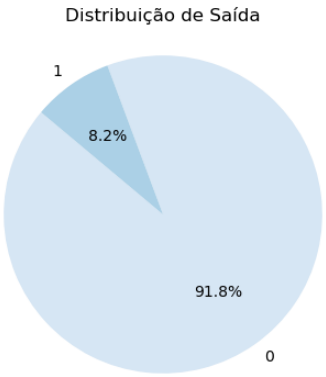


Figura 3 – Pie Chart / Diagrama de frequências relativas da variável dependente saída

Através do diagrama apresentado na Figura 3, é possível observar a distribuição da variável dependente Saída. Verifica-se que os dados estão desbalanceados, com apenas 8,2% de saídas, correspondendo a 92 observações, enquanto 91,8% dos colaboradores não saíram, representados em 1029 observações.

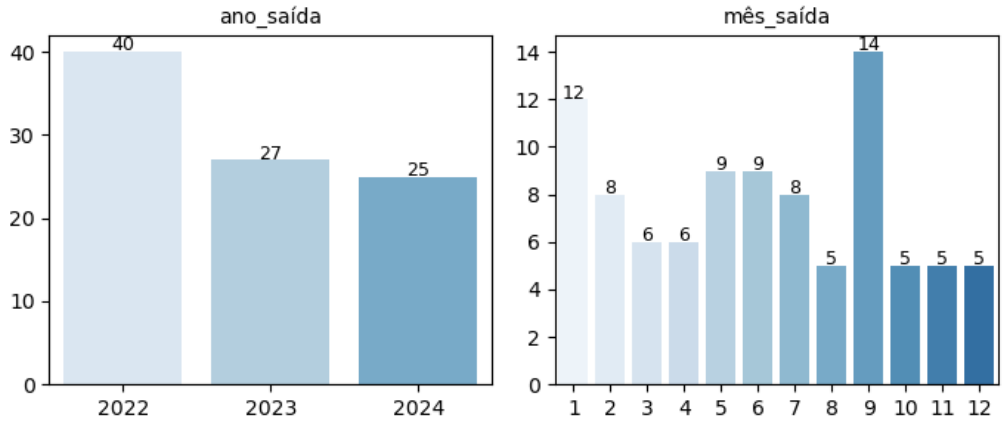


Figura 4 – Histograma de frequências absolutas de ano_saída e mês_saída

De 2022 para 2023, houve uma clara diminuição nas saídas por iniciativa do colaborador, no entanto, até outubro de 2024 já saíram 25 colaboradores, pelo que este número pode igualar ou até superar o registado em 2023. Através do histograma do mês_saída verifica-se que os meses que registam mais saídas são janeiro e setembro.

5.3. Preparação dos Dados

Esta fase inclui a seleção, limpeza e tratamento dos dados de forma a prepará-los para a fase de modelação.

5.3.1. Tratamento de Valores Ausentes

Lidar com os valores ausentes é um passo fundamental no pré-processamento dos dados. A escolha do método para lidar com estes valores é importante, uma vez que pode influenciar significativamente o desempenho dos modelos. Para tratar o problema de valores em falta, podem ser utilizadas diferentes abordagens:

- Eliminar linhas ou colunas;
- Substituir os valores em falta pela média, moda, mediana ou um outro valor específico;
- Utilizar um algoritmo baseado em pontos de distância.

Os valores em falta existentes, tal como mostra a Tabela 1 eram: ano_saída com 1024 observações, mês_saída com 1024 observações e work_dist com 1 observação.

Para as variáveis ano_saída e mês_saída, foi decidido eliminar as colunas, uma vez que apenas tinham valor nos colaboradores que saíram e, por isso, não eram interessantes para a fase de modelação. Essas variáveis foram consideradas apenas numa fase inicial de compreensão dos dados.

Para a variável work_dist, que é numérica e potencialmente importante para o modelo, foi escolhido utilizar o algoritmo K-Nearest Neighbors (KNN). O KNN Imputer funciona da seguinte forma:

- Distância Euclidiana: Utiliza a matriz de distância euclidiana para identificar os vizinhos mais próximos de cada ponto com valores ausentes.
- Imputação pela Média: Preenche os valores ausentes com a média dos valores dos vizinhos mais próximos encontrados.

Este algoritmo é eficaz, uma vez que leva em consideração a similaridade entre os pontos de dados, resultando numa imputação de valores ausentes que é mais precisa e relevante para o modelo.

5.3.2. Exploração de Variáveis Categóricas

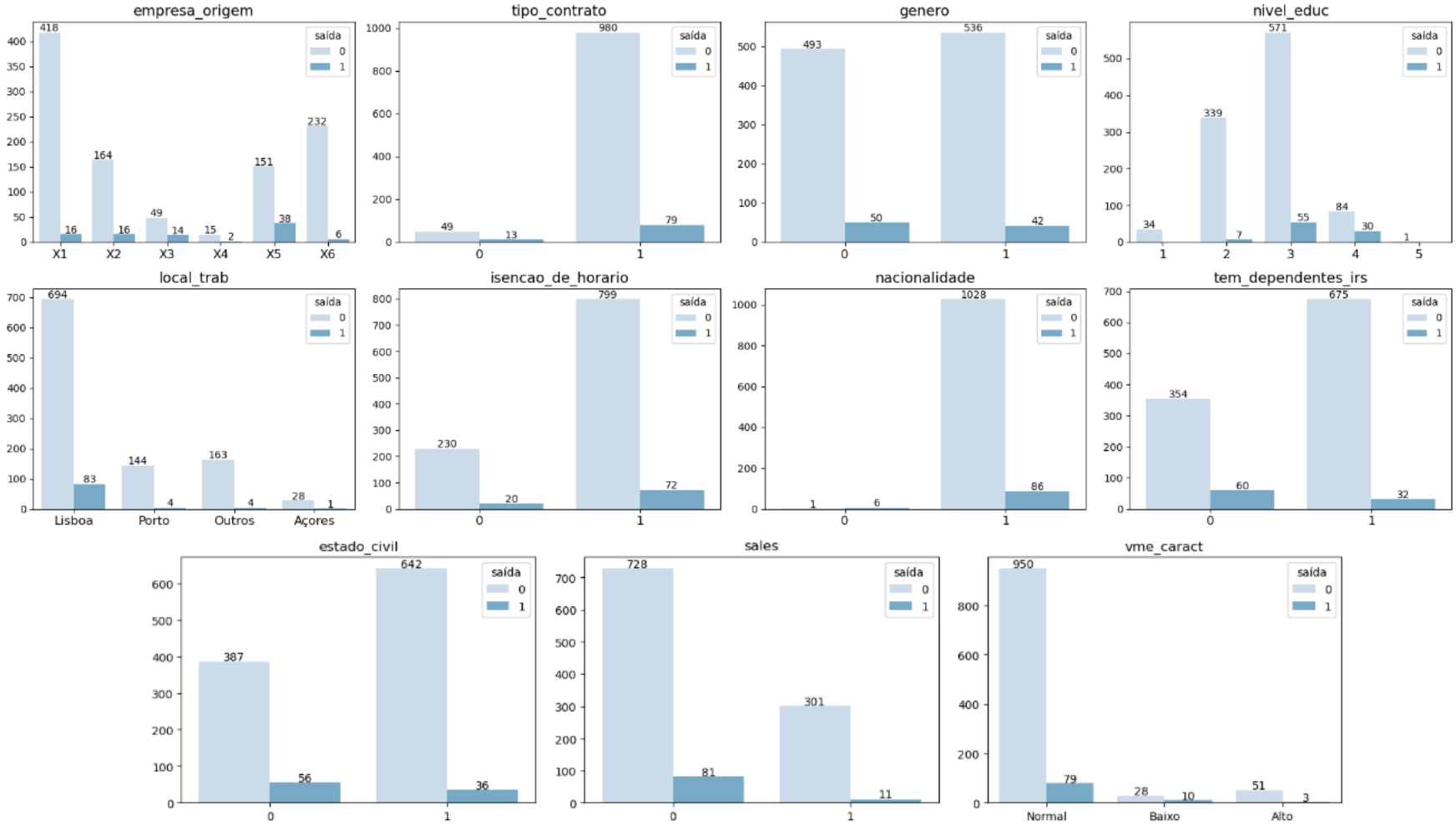


Figura 5 – 1º Histograma de frequências absolutas das variáveis categóricas pelo *target* (saída)

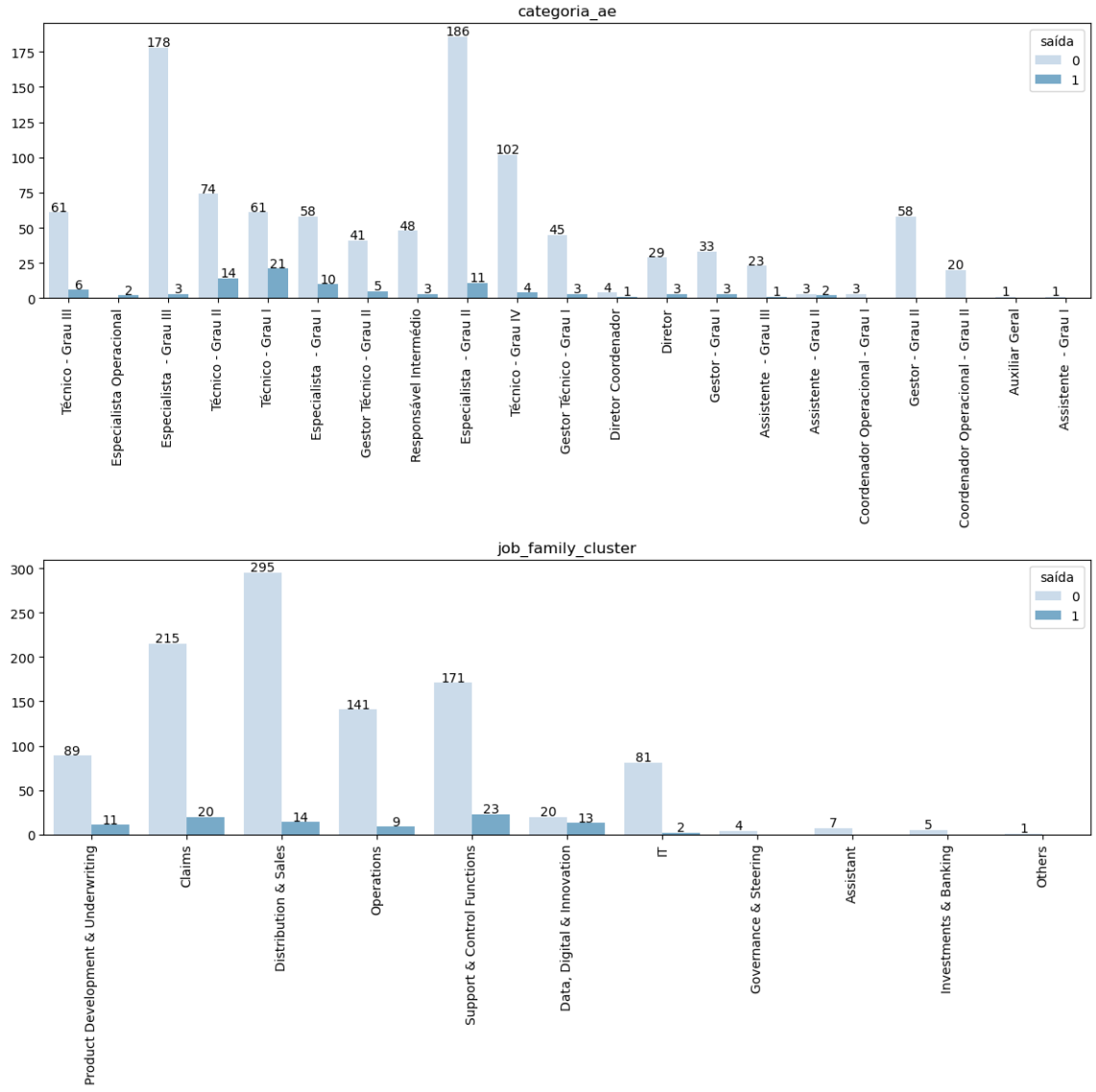


Figura 6 – 2º Histograma de frequências absolutas das variáveis categóricas pelo *target* (saída)

A maior parte dos colaboradores tem origem na antiga empresa do grupo X1. Na grande maioria, os colaboradores trabalham em Lisboa, onde está sediada a empresa.

Pode existir uma relação entre as saídas e a categoria Baixo da variável vme_caract, uma vez que nesta categoria há 28 colaboradores ativos e 10 saídas.

É possível ver que o tipo_contrato 0 (Contratado a Termo Certo) tem uma relação de saídas significativamente superior aos do tipo 1 (Efetivo sem termo). Mais de metade dos colaboradores tem o nível_educ 3 (Licenciatura ou Bacharelato).

Os colaboradores de nacionalidade 0 (estrangeira) mostram uma forte tendência para a saída, tendo saído 6 em 7, resultando em apenas 1 estrangeiro ativo na empresa atualmente. O género parece não ter grande relevância uma vez que ambos os géneros

possuem uma relação semelhante com a variável dependente.

Em relação à categoria_ae, o especialista grau III e grau II são os que têm mais colaboradores ativos, e os que mais saem são técnico grau II e técnico grau I.

No job_family_cluster, o grupo com mais colaboradores ativos e observações é Distribution & Sales. O grupo que apresenta mais saídas, em valor absoluto, é Support & Control Functions, com 23 saídas, mas este grupo possuí um elevado número de colabores ativos (171). Por outro lado, o grupo Data Digital & Innovation tem 13 saídas e 20 colaboradores ativos, pelo que apresenta uma taxa de saída relativamente alta.

5.3.2.1. Teste Qui-Quadrado

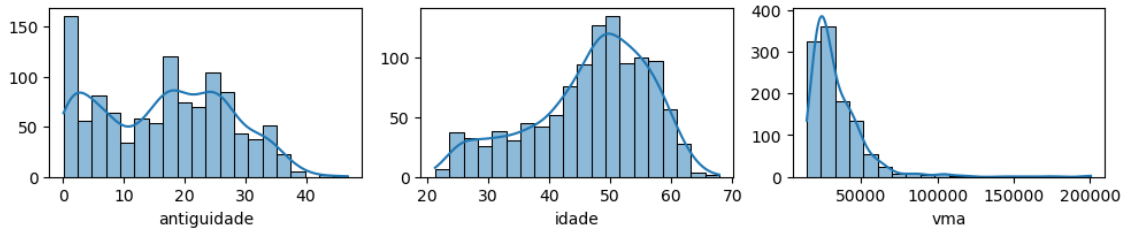
Foi realizado um Teste de Independência, utilizando o teste qui-quadrado de Pearson, com saída como variável dependente e um nível de significância (α) de 5%. As variáveis genero e isencao_de_horario, apresentaram um *p-value* superior a α, o que significa que não são consideradas estatisticamente significativas, e por isso foram eliminadas.

5.3.3. Engenharia de Recursos

Como mencionado anteriormente, durante a criação da base de dados, em Excel, já foram aplicadas técnicas de engenharia de recursos à maioria das variáveis. Nesta fase, foi necessário realizar a engenharia de recursos apenas na variável vme_caract., que foi transformada de categórica para ordinal. Neste novo formato, Baixo é representado por 0, Normal por 1 e Alto por 2.

5.3.4. Exploração de Variáveis Numéricas

Para exploração das variáveis numéricas recorreu-se inicialmente às Figuras 6 e 7, que apresentam, um histograma de frequências absolutas das variáveis numéricas e um pair-plot para relação entre pares de variáveis numéricas por saída, respetivamente.



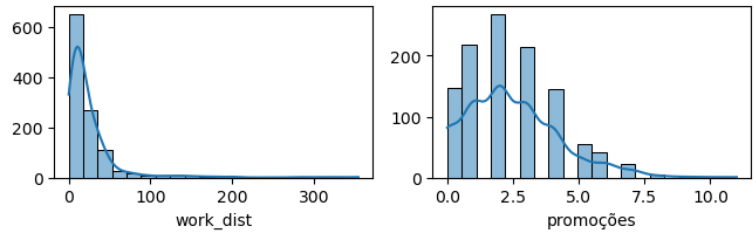


Figura 7 – Histograma de frequências absolutas das variáveis numéricas

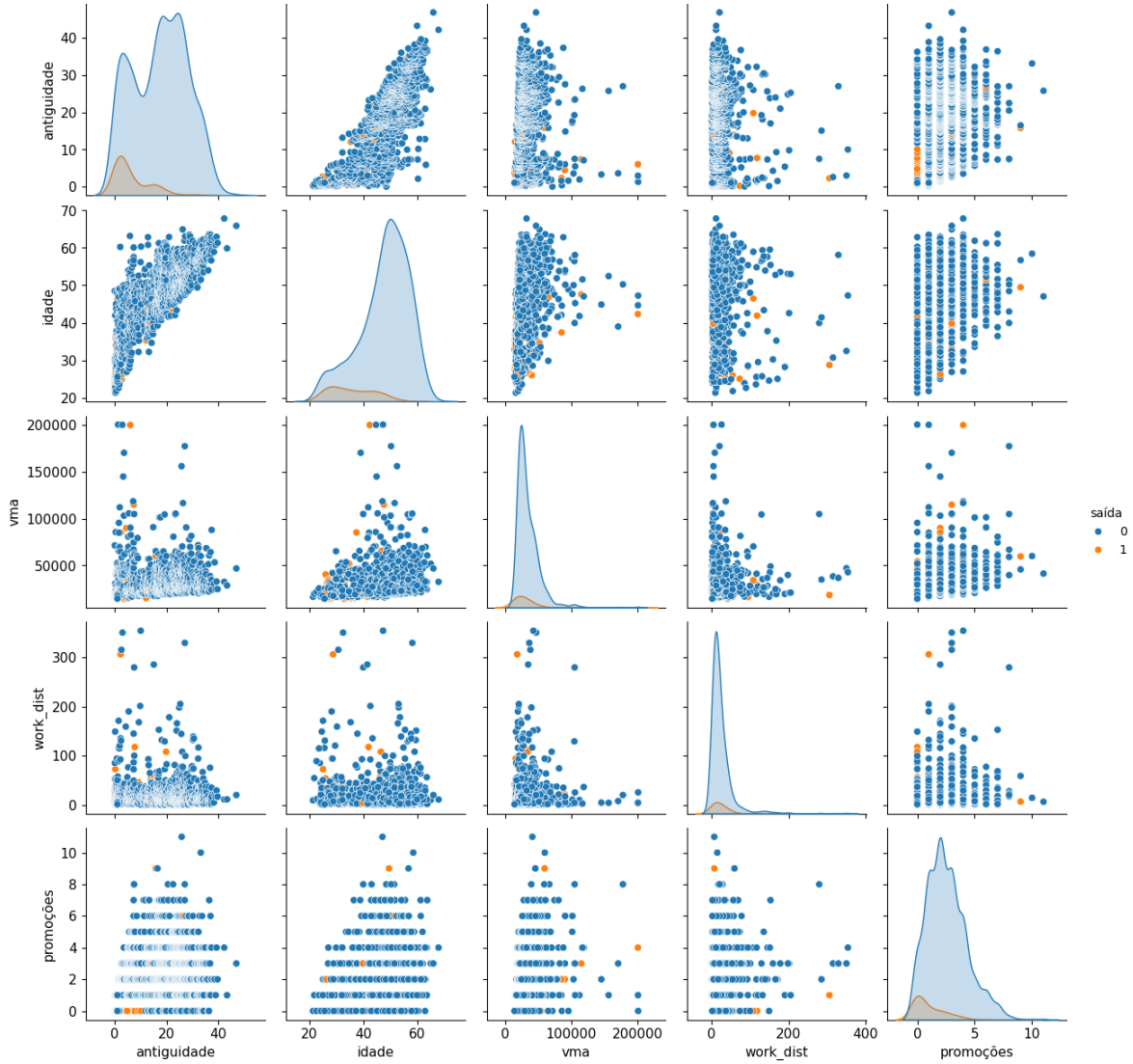


Figura 8 – Relação entre pares de variáveis numéricas por saída

Ao analisar a Figura 8, fica evidente, tal como já foi referido anteriormente, que o número de colaboradores ativos é bastante superior ao de saídas.

Nenhuma das variáveis apresenta uma distribuição normal perfeita, sendo que as variáveis idade e promoções são as únicas que apresentam uma distribuição mais próxima

da normal.

A distribuição da work_dist parece ser mais uniforme, mas ainda assim com uma leve assimetria à direita, indicando que a maioria dos colaboradores reside relativamente perto do trabalho. É necessário ter em atenção a região da cauda, pois pode comportar-se como um outlier, o que afeta o desempenho do modelo. Isso pode ser causado por erros nos dados, como a falta de atualização dos códigos postais ou incongruências em relação ao local de trabalho.

A distribuição do vma também mostra uma assimetria à direita, sugerindo que a maioria dos trabalhadores recebem vencimentos fixos anuais até 50.000€ anuais. O gráfico de dispersão entre idade e vma revela uma tendência positiva, onde colaboradores mais velhos tendem a receber remunerações mais elevadas. É possível verificar que no final da cauda da distribuição do vma, existe a presença de uma saída.

Onde se nota uma relação mais evidente com a saída, é nos colaboradores com menor idade, antiguidade na empresa e nos que receberam menos promoções.

Todas as variáveis numéricas apresentam distribuições assimétricas, algumas mais pronunciadas que outras, o que pode indicar a presença de outliers. Esses outliers devem ser analisados e, se necessário, removidos para não afetar o desempenho dos algoritmos.

Além disso, a estandardização dessas variáveis pode ser necessária para garantir que todas tenham a mesma escala, um aspeto crucial para muitos algoritmos de *machine learning*.

5.3.4.1. Análise de correlações

Após a eliminação das variáveis categóricas irrelevantes, a transformação da variável vme_caract numa variável ordinal e análise das variáveis numéricas, foi realizada uma análise das correlações entre as diferentes variáveis numéricas, ordinais e binárias, com o objetivo de compreender melhor as relações entre elas e identificar possíveis colinearidades.

Para isso, foi utilizada uma matriz de correlações de Spearman, representada através de um mapa de calor (Figura 9). A matriz de Spearman é adequada para medir a

correlação entre variáveis de diferentes tipos, sendo particularmente indicada quando as variáveis não seguem uma distribuição normal, além disso, é mais robusta a outliers.

Esta metodologia contribui para a simplificação do modelo, permitindo focar nas variáveis mais significativas e evitando a inclusão de variáveis correlacionadas, que poderiam introduzir ruído ou redundância na análise.

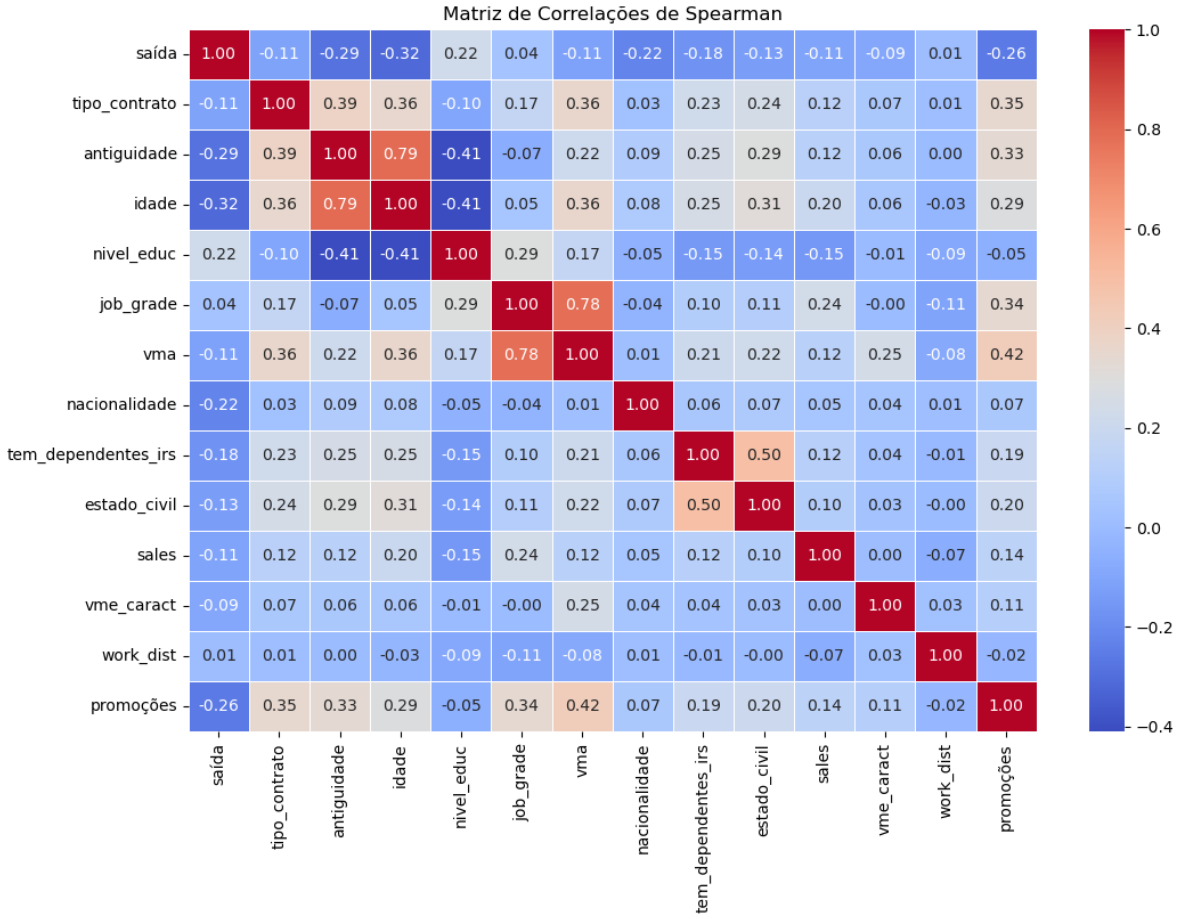


Figura 9 – Mapa de calor da matriz de correlações de Spearman

A partir do mapa de calor, foi possível verificar uma correlação elevada entre as variáveis idade e antiguidade, assim como entre job_grade e vma. No caso de idade e antiguidade, decidiu-se manter ambas as variáveis, sem proceder a transformações, pois ambas apresentaram boa correlação com a variável dependente. Além de que ambas são variáveis numéricas, considerando que o conjunto de dados já possui poucas variáveis numéricas, decidiu-se que na fase de modelação, será o próprio modelo a determinar se alguma destas variáveis deverá ser eliminada.

Quanto ao job_grade e ao vma, optou-se por manter a variável vma e eliminar o

job_grade para evitar problemas de multicolinearidade. Esta decisão baseou-se no facto de job_grade apresentar uma correlação menor com a variável dependente e ser uma variável ordinal, enquanto vma é uma variável numérica, o que pode ser mais vantajoso para a análise subsequente.

5.3.4.2. Remover Outliers

Para a remoção de outliers, foram utilizados boxplots para visualizar a distribuição dos dados, como está representado na Figura 10, e as abordagens IQR (Intervalo Interquartil) e Remoção Manual, para tratar os outliers.

O IQR calcula a amplitude interquartil: $IQR = Q3 - Q1$

De seguida considera os limites:

$$Limite\ inferior = Q1 - 1{,}5 \times IQR$$

$$Limite\ superior = Q3 + 1{,}5 \times IQR$$

Os valores fora destes limites são considerados outliers e consequentemente são removidos.

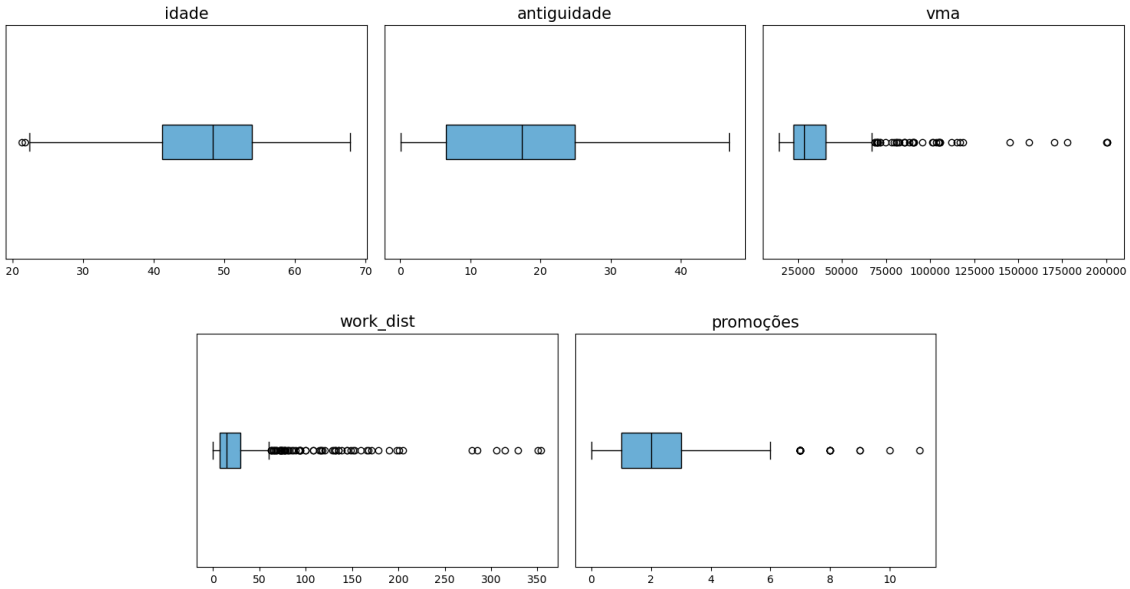


Figura 10 – Boxplot das variáveis numéricas, análise de outliers

A variável antiguidade não apresenta outliers. As variáveis idade e promoções contêm poucos outliers, nenhum dos quais corresponde a uma saída, nesse sentido, para estas variáveis foi aplicado o método IQR, e assim todos os outliers destas variáveis, representados na Figura 10, foram eliminados.

A variável vma apresenta 34 outliers, pelo que não foi considerado que se estavam a eliminar demasiados dados e também nesta variável foi aplicado o método IQR.

Já na variável work_dist, estabeleceu-se um limiar para remoção de outliers, uma vez que o IQR estava a eliminar uma quantidade excessiva de dados.

No total 7,7% dos dados foram eliminados ficando o *dataframe* com 959 colaboradores ativos e 87 saídas.

5.3.5. Remover saídas pré-reforma

Após conversa com a empresa, foi possível identificar quais os colaboradores que já tinham acordo pré-reforma para 2024 e 2025 e que, por isso, não fazia sentido incluir no modelo de previsão de saída por iniciativa voluntária. Nesse sentido, antes da modelação, foram eliminadas essas observações, o que diminuiu um pouco o desbalanceamento dos dados, que também é positivo para o modelo.

Após essa limpeza, ficou-se com 934 colaboradores ativos e 87 saídas por

iniciativa do colaborador.

5.3.6. Codificação e estandardização

Antes da modelação do algoritmo, pode ser necessário codificar e estandardizar as variáveis, de forma a melhorar o desempenho dos modelos. A codificação OneHotEncoding é utilizada para variáveis categóricas sem ordem, como local_trab, e a codificação OrdinalEncoding em variáveis ordinais como vme_caract.

OneHotEncoding, cria colunas binárias para cada categoria da variável. Esta técnica é ideal para transformar variáveis categóricas que não possuem uma ordem natural. Nesse sentido, esta técnica foi aplicada às variáveis empresa_origem, local_trab, categoria_ae e job_family_cluster.

OrdinalEncoding, é utilizado para variáveis categóricas que possuem uma ordem natural, atribuindo um valor inteiro a cada categoria. Esta abordagem permite que o modelo utilize a relação ordinal entre as categorias, reduzindo assim a complexidade. Nesta fase, não foi necessário aplicar esta técnica a nenhuma das variáveis, uma vez que vma_caract e nível_educ já tinham sido previamente transformadas em ordinais.

As variáveis numéricas foram estandardizadas através do método MinMaxScaler, de forma a garantir que tivessem a mesma escala, uma vez que apresentavam ordens de grandeza bastante diferentes. Enquanto o vma podia ter uma ordem de grandeza de 10^5, idade e antiguidade tinham uma ordem de grandeza de no máximo 10^2. O método MinMaxScaler é particularmente útil quando as variáveis a estandardizar não seguem uma distribuição normal.

Seja $X_1, \ldots, X_n$ uma amostra de dimensão n:

$$Z_i = \frac{X_i - X_{min}}{X_{max} - X_{min}}$$

Os dados estandardizados têm um valor mínimo de 0 e um valor máximo de 1

Em python foi utilizada a ferramenta MinMaxScaler da biblioteca sklearn, para a estandardização das variáveis: idade, antiguidade, vma, work_dist e promoções.

5.4. Modelação

Na fase de modelação, foi inicialmente aplicado o tradicional método de divisão dos dados em treino e teste. Uma vez que o conjunto de dados era consideravelmente pequeno, foi decidido dividir-se aleatoriamente os dados em 80% para treino e 20% para teste.

Esta divisão segue aquilo que são consideradas boas práticas de *machine learning*. Permite treinar um modelo utilizando uma porção substancial dos dados, de modo a ter dados suficientes para aprender e ao mesmo tempo, assegurar uma parte dos dados para avaliação num conjunto de dados independentes.

Para validação dos modelos e avaliação da sua capacidade de generalização, foi efetuada a validação cruzada *k-fold*, neste caso com 5 subconjuntos (*folds)*. Esta técnica consiste na divisão do conjunto de dados, em 5 subconjuntos de tamanho aproximadamente igual, treinando o modelo em 4 desses subconjuntos e testando no subconjunto restante. O processo é repetido 5 vezes, de modo que cada subconjunto seja utilizado como conjunto de teste uma vez. A validação cruzada permite reduzir a variabilidade nos resultados da avaliação, permitindo uma análise mais robusta do desempenho do modelo.

Para lidar com o desequilíbrio das classes foram aplicadas técnicas de *oversampling* e de *undersampling*.

O *oversampling* é uma técnica utilizada para aumentar artificialmente a quantidade de dados da classe minoritária, criando novas instâncias sintéticas, em vez de replicar dados existentes. O objetivo desta abordagem é equilibrar a distribuição das classes no conjunto de dados, garantindo que o modelo de *machine learning* não seja enviesado para a classe maioritária e, assim, melhorar o desempenho preditivo do modelo em relação à classe menos representada (He & Garcia, 2009).

Undersampling envolve a remoção de instâncias da classe maioritária para criar um conjunto de dados mais equilibrado. Isto é feito para garantir que o algoritmo de aprendizagem não se torne tendencioso para a classe maioritária devido à sua presença esmagadora no conjunto de dados (He & Garcia, 2009).

Para *oversampling*, as técnicas utilizadas e testadas foram o SMOTE *(Synthetic Minority Over-Sampling Technique*) e o ADASYN *(Adaptive Synthetic Sampling)*. O SMOTE gera amostras sintéticas baseadas nas semelhanças de características entre exemplos existentes da classe minoritária, para cada exemplo da classe minoritária, o algorítmo considera os K-Vizinhos mais próximos e cria amostras novas (He & Garcia, 2009). O ADASYN é um método que cria, de forma adaptativa, amostras de dados sintéticos com base na distribuição da classe minoritária, gera quantidades diferentes de dados sintéticos para cada exemplo minoritário, dependendo da sua distribuição de densidade. Esta abordagem ajuda a resolver o desequilíbrio concentrando-se em áreas onde a classe minoritária está sub-representada (He & Garcia, 2009). Foca-se assim em gerar amostras para exemplos mais difíceis de classificar.

As técnicas utilizadas e testadas para *undersampling* foram Tomeklinks e RandomUnderSampler. Os Tomeklinks são definidos como pares de vizinhos mais próximos minimamente distantes, pertencentes a classes opostas, isto significa que se duas instâncias formarem uma Tomeklink, pelo menos uma destas instâncias provavelmente será ruído (He & Garcia, 2009). Quando esse par é identificado, o exemplo da classe maioritária é removido, os Tomeklinks são usados para remover exemplos redundantes da classe maioritária. O RandomUnderSampler é uma técnica simples de *undersampling* onde são removidas aleatoriamente observações da classe maioritária até que as classes fiquem equilibradas. Este método pode ser eficaz na redução de viés em modelos de *machine learning* para balancear o número de observações de cada classe (He & Garcia, 2009). É uma abordagem direta, sendo particularmente útil quando se trabalha com dados muito desbalanceados.

Estas técnicas são essenciais para lidar com o desequilíbrio de classes e garantir que os modelos não se tornem enviesados em relação à classe mais predominante no conjunto de dados, neste caso a classe 0 (colaboradores ativos).

Inicialmente, foi testada uma Regressão Logística, para avaliar o desempenho do conjunto de dados num modelo mais simples, de modo a servir como referência para comparar com o desempenho de algoritmos de *machine learning* mais avançados. De seguida foram testados os modelos: Árvore de Decisão, Floresta Aleatória, XGBoost e AdaBoost.

Para otimizar e identificar os melhores hiperparâmetros nos modelos, foram considerados os métodos de pesquisa *RandomizedSearchCV* (aleatória) e *GridSearchCV* (em grelha).

O *GridSearchCV* é um dos métodos mais utilizados na pesquisa pelos melhores hiperparâmetros. Realiza uma pesquisa exaustiva pelas melhores combinações possíveis, de forma a testar todas as combinações possíveis numa grelha específica, para encontrar a solução ideal. No entanto, quando existem muitos parâmetros, a velocidade de pesquisa é muito lenta e são desperdiçados excessivos recursos computacionais (Liu et al., 2024).

O *RandomizedSearchCv* realiza uma amostragem aleatória dos hiperparâmetros, dentro de um espaço predefinido. Permite uma exploração de um espaço maior de dados, de forma mais eficiente em termos de tempo de computação. Como o espaço de parâmetros no *RandomizedSearchCV* pode ser mais vasto, sem afetar a eficiência computacional em excesso, este método pode captar uma parte significativa do espaço de hiperparâmetros e encontrar uma combinação melhor do que o *GridSearchCV*, que por testar todas as combinações possíveis, está limitado no espaço de hiperparâmetros, devido à eficiência.

Uma vez que esta é uma fase que envolve diversas simulações na procura pelos melhores hiperparâmetros, optou-se por utilizar o método *RandomizedSearchCV*, pois o *GridSearchCV* exige muito tempo de execução, além de consumir recursos computacionais elevados, sem garantir uma performance superior à do *RandomizedSearchCV*.

Além disso, para cada modelo foi utilizada a função *predict_proba*, que retorna as probabilidades das classes ao invés de uma classificação binária direta. Saito & Rehmsmeier (2015) discutem que, em cenários de desequilíbrio de classes, o ajuste do limiar de decisão (*threshold*) é fundamental para otimizar o modelo em relação a métricas como precisão ou recall, ao invés de confiar no limiar de decisão padrão de 0,5, principalmente quando a classe minoritária é a mais importante.

Em cenários de desequilíbrio de classes, ajustar o limiar de decisão é uma estratégia comum para maximizar a eficácia da previsão de classes minoritárias, o que é relevante no contexto da previsão de *turnover* de colaboradores (Wang et al., 2020).

Ao ajustar o limiar de decisão utilizando probabilidades previstas (predict_proba), é possível melhorar a sensibilidade do modelo em relação à classe minoritária, permitindo identificar corretamente os colaboradores com maior probabilidade de sair.

Esta fase envolveu diversas simulações na procura pelos melhores hiperparâmetros. Os modelos na fase de modelação foram otimizados com alteração do limiar de decisão. Os parâmetros que resultaram na melhor pesquisa de hiperpârametros de cada modelo, e os respetivos melhores hiperpârametros encontrados, estão presentes no anexo A1.

Relativamente às técnicas de *oversampling* e *undersampling*, as técnicas de *oversampling* que resultaram nos melhores modelos foram: SMOTE para Árvore de Decisão, XGBoost e AdaBoost, e foi ADASYN para Regressão Logística e Floresta Aleatória. Nas técnicas de *undersampling* a que resultou nas melhores performances foi o RandomUnderSampler para todos os modelos.

5.5. Avaliação

Para avaliar com exatidão a performance e efeitos de classificação dos modelos, são utilizados os seguintes indicadores de avaliação:

- Matriz de Confusão: permite uma análise mais detalhada e visual relativamente aos erros e acertos do modelo, onde VP são os verdadeiros positivos, VN os verdadeiros negativos, FP os falsos positivos e FN os falsos negativos.

		Turnover previsto	
		Classe 0 - ativo (negativo)	Classe 1 - saída (positivo)
Turnover Real	Classe 0 - ativo (negativo)	Verdadeiros Negativos (VN)	Falsos Positivos (FP)
	Classe 1 - saída (positivo)	Falsos Negativos (FN)	Verdadeiros Positivos (VP)

Tabela 2 – Matriz de Confusão

- PCCC ou *Accuracy*: “refere-se à proporção de amostras corretamente previstas em relação ao número total de amostras” (Liu et al., 2024).

$$PCCC = \frac{VP + VN}{VP + VN + FP + FN}$$

- Precisão: "a precisão do *turnover* é o rácio entre as saídas efetivas e o total de amostras de saídas na previsão" (Liu et al., 2024).

$$Precisão = \frac{VP}{VP + FP}$$

- Sensibilidade (*Recall* da classe 1): "é a proporção das amostras que são corretamente previstas como saídas, em relação às amostras de saídas reais" (Liu et al., 2024). É a capacidade de o modelo identificar corretamente os que saem.

$$Sensibilidade = \frac{VP}{VP + FN}$$

- Especificidade (*Recall* da classe 0): É a capacidade de o modelo identificar corretamente os que não saem.

$$Especificidade = \frac{TN}{TN + FP}$$

- *F1-Score* ou valor F1: "é a média harmónica da precisão e do recall" (Liu et al., 2024).

$$F1 - Score = 2 \times \frac{Precisão \times Recall}{Precisão + Recall}$$

- Suporte: é nº de observações de cada classe.

- AUC (Area Under the ROC Curve): o valor AUC é a área abaixo da curva ROC, que mostra a performance de um modelo de classificação em diferentes limiares de decisão e que pode refletir de forma abrangente o desempenho de classificação do modelo de *machine learning*. Na prática, é a relação entre sensibilidade e especificidade. O valor AUC situa-se entre 0 e 1 e quanto maior for o valor AUC, melhor será o efeito de classificação do modelo. É uma métrica importante para medir a capacidade de um modelo em discriminar entre as classes ao variar o limiar de decisão (Fawcett, 2006). É útil quando se pretende alterar o limiar de decisão para otimizar o modelo (Saito & Rehmsmeier, 2015).

5.6. Implementação

A última etapa da metodologia CRISP-DM é designada implementação e consiste na aplicação prática do projeto. Nesta fase são efetuadas a Discussão de Resultados, a Conclusão e a explicação de Limitações e Recomendações, que serão detalhados nos capítulos seguintes. A implementação resulta assim na fase final da metodologia e consequentemente do projeto, neste caso com apresentação dos resultados, tanto para a empresa como para a comunidade académica.

O modelo final foi guardado através da biblioteca joblib e ficará à disposição da empresa, caso esta decida proceder à sua implementação. Para facilitar a tomada de decisões e identificar quais colaboradores atualmente ativos têm maior probabilidade de sair voluntariamente, o modelo poderá ser aplicado ao conjunto de colaboradores atual, de forma a gerar a previsão da probabilidade de saída voluntária de cada colaborador. Assim, os colaboradores poderão ser agrupados em diferentes níveis de risco de saída, o que permitirá à empresa a identificação e intervenção, através de medidas preventivas eficientes, tanto nos grupos de risco em geral, como em cada colaborador individualmente.

6. Discussão dos Resultados

Inicialmente, foi realizada uma comparação relativamente à performance dos modelos no conjunto de treino, utilizando validação cruzada. As métricas PCCC, Precisão, Sensibilidade, Especificidade e AUC foram utilizadas para avaliar os modelos, com base nas médias obtidas durante a validação cruzada.

Para avaliação do conjunto de teste foram utilizadas as métricas PCCC, Precisão, Sensibilidade, Especificidade, AUC e *F1-Score*, utilizando o mesmo limiar de decisão que otimizou os modelos durante a fase de treino.

A Tabela 3 apresenta os resultados dos melhores modelos para cada algoritmo. Os modelos apresentaram um desempenho algo distinto no conjunto de treino e teste, para a classe 1, o que indica a presença de *overfitting* nesta classe. No entanto, os resultados no conjunto de teste foram bastante positivos. Tal como indicado pela literatura, os melhores desempenhos no conjunto de teste foram aqueles que melhor lidam com *overfitting*, a Floresta Aleatória e o XGBoost. O modelo com pior desempenho, que teve mais problemas em lidar com o *overfitting* foi a Regressão Logística, seguido da Árvore de Decisão, como já era de esperar, uma vez que são modelos mais simples e de menor complexidade.

No conjunto de treino, os modelos de ensemble demonstraram resultados bastante similares, AdaBoost apresentou PCCC de 96,45% e AUC de 99.43%, seguido do XGBoost que teve PCCC de 95,38% e AUC de 99,18%, e da Floresta Aleatória que teve um PCCC de 94,77% e AUC de 98,96%. Como se sabe pela literatura, o AdaBoost pode ter alguns problemas em lidar com *overfitting* e, apesar de apresentar uma boa performance no conjunto de teste (com uma precisão e sensibilidade de 76,47%), não se conseguiu destacar, sendo apenas o terceiro melhor, ficando atrás do XGBoost, que foi o segundo melhor, apresentou igual sensibilidade, mas uma precisão superior (81,25%).

O maior destaque foi mesmo para a Floresta Aleatória, que se revelou o melhor dos modelos, com uma qualidade significativamente superior aos restantes modelos, lidando melhor com o *overfitting*. Os seus resultados foram bastante positivos, tendo em conta as condições e objetivos do projeto (PCCC=97,07%, Precisão=82,35%, Sensibilidade=82,35, Especificidade=98,40%, AUC=96,59 e *F1-Score*=82,35%).

A técnica de Bagging, usada pela Floresta Aleatória, é geralmente preferida quando o objetivo é reduzir a variância e quando se trabalha com modelos propensos ao *overfitting*. A técnica de Boosting, como usada no XGBoost e AdaBoost, é mais adequada quando se pretende reduzir o viés e melhorar a precisão, especialmente em conjuntos de dados complexos.

Nesse sentido, os resultados obtidos foram os esperados, com a Floresta Aleatória a apresentar-se como o melhor modelo para este projeto.

		Regressão Logistica	Árvore de Decisão	Floresta Aleatória	XGBoost	AdaBoost
Treino 80% (Média Validação Cruzada)	PCCC	92,09%	92,69%	94,77%	95,38%	96,45%
	Precisão (classe 1)	94,10%	91,10%	96,39%	96,05%	96,39%
	Sensibilidade	89,81%	94,64%	93,03%	94,64%	96,51%
	Especificidade	94,37%	90,75%	96,51%	96,11%	96,38%
	AUC	97,78%	94,60%	98,96%	99,18%	99,43%
Teste (20%)	PCCC	91,22%	91,71%	97,07%	96,59%	96,10%
	Precisão (classe 1)	47,62%	50,00%	82,35%	81,25%	76,47%
	Sensibilidade	52,63%	88,24%	82,35%	76,47%	76,47%
	Especificidade	94,15%	92,02%	98,40%	98,40%	97,87%
	AUC	89,55%	89,94%	96,59%	96,87%	94,77%
	F1-Score (classe 1)	52,63%	63,83%	82,35%	78,79%	76,47%
Limiar de Decisão		58%	50%	61%	60%	55%

Tabela 3 – Avaliação dos melhores modelos por algoritmo

No anexo A2, encontra-se a importância relativa dos preditores para a saída por iniciativa do colaborador, de todos os modelos, à exceção da Regressão Logística. De forma a evitar um anexo demasiado extenso, este anexo mostra apenas os preditores com valores de importância relativa acima de 0,02. Os 4 preditores que mais influenciam as previsões de cada um dos modelos, que apresentam todos um valor de importância relativa superior a 0,05, são:

- Árvore de Decisão: (idade, promoções, vma, empresa_origem_X2);
- Floresta Aleatória: (idade, promoções, antiguidade, vma);

- XGBoost: (idade, promoções, local_trab_Porto, empresa_origem_X2);
- AdaBoost: (promoções, idade, vma, antiguidade).

Verifica-se que os modelos consideram praticamente as mesmas variáveis com maior importância, idade, promoções e vma, todas variáveis numéricas. A Floresta Aleatória e AdaBoost têm ainda em comum antiguidade que também é numérica, enquanto Árvore de Decisão e XGBoost identificam empresa_origem_X2 como o quarto preditor mais importante. O XGBoost é o único que cosidera local_trab_Porto uma das suas variáveis mais importantes.

Este padrão mostra a consistência e similaridade entre os modelos na seleção de variáveis, sendo que mais uma vez a Floresta Aleatória se destaca, uma vez que equilibra bem a identificação das variáveis de maior importância, idade é o mais importante 3 vezes, promoções o segundo mais importantes 3 vezes, vma aparece 2 vezes como terceiro mais importante, sendo que a Floresta Aleatória o identifica como quarto mais importante e identifica antiguidade como o terceiro mais importante enquanto AdaBoost considera essa variável a quarta mais importante.

Portanto, a Floresta Aleatória não só apresenta um desempenho superior nas métricas de avaliação, como também evidencia uma análise mais abrangente das variáveis preditivas. Isso torna-a a escolha mais apropriada para este tipo de problema, onde é fundamental equilibrar a análise de diversas variáveis importantes, minimizando o risco de *overfitting* e maximizando a generalização para novos dados.

As 5 variáveis com maior importância no modelo da Floresta Aleatória são, por ordem de importância, idade (20%), promoções (16%), antiguidade (14%), vma (8%) e work_dist (3%), que são todas as variáveis numéricas da base de dados e juntas, representam 51% de importância no modelo.

Para uma análise mais aprofundada das características dos que tendem a sair, foram calculadas as medianas dos colaboradores da classe 0 e da classe 1, das variáveis com maior importância. Essas medianas estão representadas na Figura 11.

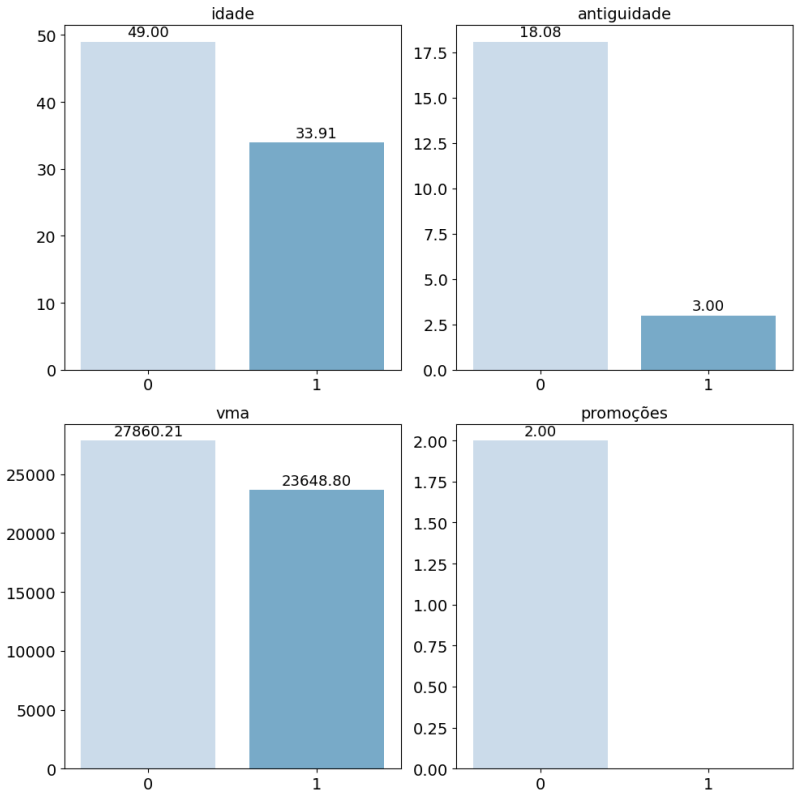


Figura 11 – Mediana por classes dos preditores com maior influência no modelo

A idade é a variável que mais influência o *turnover*, pela Figura 11 observa-se que colaboradores mais velhos, tendem a ficar mais na empresa com a mediana destes a situar-se nos 49 anos, já os colaboradores mais jovens apresentaram maior probabilidade de sair por iniciativa voluntária, principalmente aqueles que estão mais próximos dos 34 anos. Olhando novamente para a Figura 8, é possível observar que a densidade dos dados que saem é maior no lado esquerdo do gráfico, indicando uma concentração maior de saídas entre os colaboradores mais jovens, enquanto a densidade relativa aos que ficam é maior no lado direto do gráfico, indicando que a maior parte dos que ficam são colaboradores com idade mais avançada. Além disso, podemos ver que nas idades mais baixas a densidade dos que saem é praticamente metade da dos que ficam, uma proporção elevada e que só se encontra nesta variável.

Promoções é a segunda característica que mais influência o modelo de *turnover*, apresentou uma mediana de 0 entre os colaboradores que saíram. Isto indica que maior parte dos colaboradores que saem não têm promoções, pelo que este tipo de colaboradores é mais propenso a sair por iniciativa própria.

A Antiguidade, é o terceiro preditor que mais influência o modelo de *turnover*.

Fica evidente pela Figura 11, que colaboradores mais recentes na empresa, têm uma maior tendência para a saída. A mediana indica que os colaboradores com uma antiguidade próxima dos 3 anos de serviço, apresentaram uma probabilidade maior de sair por iniciativa voluntária, este valor torna-se ainda mais impactante quando se verifica que a mediana de antiguidade dos que permanecem na empresa é de aproximadamente 18 anos.

A quarta variável que mais influência o *turnover* é o vma, colaboradores com salários relativamente inferiores têm uma tendência maior para a saída, com a mediana destes colaboradores a situar-se nos 23.648,80€ anuais.

Com a divisão de 80% dos dados para treino e 20% para teste, o suporte para a classe 1 (saídas) contou com 17 observações e para a classe 0 (colaboradores ativos) com 188 observações, totalizando 205 observações na amostra de teste.

A variável ID foi convertida em índice no início do projeto, de forma a analisar individualmente cada trabalhador, e dessa forma, ter uma visão mais detalhada e mais individualizada das previsões realizadas pelo modelo.

Com o objetivo de criar níveis de risco de saída, foram definidos inicialmente 4 níveis de probabilidade de risco de saída, com base no limiar de decisão do conjunto de treino.

Esses níveis foram calibrados para fornecer uma avaliação robusta da probabilidade de saída, o primeiro nível, que inclui os casos abaixo do primeiro limiar de decisão, foi definido como aquele em que não saem colaboradores. O segundo nível, entre o primeiro e o segundo limiar de decisão, tem como objetivo que o modelo identifique corretamente todos os colaboradores que efetivamente saíram, sem deixar nenhum dos que saiu de fora, ou seja, sem nenhum falso negativo, o objetivo é que a sensibilidade seja 1. O terceiro nível é aquele em que o modelo é ótimo, que foi aquele que foi avaliado anteriormente e comparado com os outros algoritmos de *machine learning*, este nível fica entre o segundo e o terceiro limiar de decisão. O último nível tem como objetivo que todas as observações consideradas como saídas, sejam saídas reais (Precisão=1), este nível fica acima do terceiro limiar de decisão.

Estes limiares de decisão foram testados no conjunto de teste, de forma a testar a capacidade do modelo em criar estes níveis:

- Modelo com limiar de decisão de 0,21

Matriz de Confusão	
131	57
0	17

	Precisão	Recall	Valor F1	Support	Accuracy
0	1,00	0,70	0,82	188	0,72
1	0,23	1,00	0,63	17	

Tabela 4 – Avaliação Floresta Aleatória no conjunto teste com limiar de decisão de 0,21

Considera como saídas aqueles que apresentam probabilidade acima de 21%, este modelo identifica corretamente todos os 17 colaboradores que saíram, mas considera que 57 saíram, que na realidade não saíram;

- Modelo ótimo, com limiar de decisão de 0,61

Matriz de Confusão	
185	3
3	14

	Precisão	Recall	Valor F1	Support	Accuracy
0	0,98	0,98	0,98	188	0,97
1	0,82	0,82	0,82	17	

Tabela 5 – Avaliação Floresta Aleatória no conjunto teste com limiar de decisão de 0,61

Considera como saídas aquele que apresentam probabilidade acima de 61%. Acerta em 14 das 17 saídas e considera de forma errada que 3 colaboradores saíram;

- Modelo com 0,79 de limiar de decisão

Matriz de Confusão	
188	0
12	5

	Precisão	Recall	Valor F1	Support	Accuracy
0	0,94	1,00	0,97	188	0,94
1	1,00	0,29	0,45	17	

Tabela 6 – Avaliação Floresta Aleatória no conjunto teste com limiar de decisão de 0,79

Considera como saídas aquele que apresentam probabilidade acima de 79%, este modelo acerta corretamente em 12 colaboradores que saíram, mas deixa de fora 5 saídas reais.

Uma vez que o modelo captou muito bem os níveis de saída, foi criado um nível extra entre o 1 e o 2, designado de risco de saída médio, de forma a encontrar um limiar de decisão que, entre estes níveis, captasse o máximo de verdadeiros negativos possível, mas com o menor número de falsos positivos possível. O limiar encontrado foi de 0,51 e a avaliação desse nível resultou em 15 verdadeiros positivos e 11 falsos positivos, com apenas 2 falsos negativos.

Matriz de Confusão	
177	11
2	15

	Precisão	Recall	Valor F1	Support	Accuracy
0	0,99	0,94	0,96	188	0,94
1	0,58	0,88	0,70	17	

Tabela 7 – Avaliação Floresta Aleatória no conjunto teste com limiar de decisão de 0,51

Assim, na implementação do modelo, tal como se observa pela Tabela 8 são considerados 5 níveis.

Aqueles que tiverem uma probabilidade de saída abaixo de 21%, não estão em risco de saída, os que tiverem probabilidade entre 21% e 51% são considerados de risco mínimo, nesta banda é preciso ter alguma atenção, mas a maior parte dos colaboradores dentro deste intervalo não saem.

Dos 51% aos 61% são considerados de risco médio, neste nível pode-se começar a ter alguma atenção, mas a proporção de saídas reais ainda é um pouco baixa, de 1 acerto para 7 erros.

Entre os 61% e os 79% são de risco elevado, nestes colaboradores é preciso uma especial atenção e começar a criar algumas estratégias de retenção, apesar de alguns destes colaboradores não terem saído, esse número é bastante reduzido e a maior parte dos colaboradores dentro desta banda sai, 9 acertos para 3 erros.

Os que apresentarem uma probabilidade de saída acima dos 79% estão em risco

máximo de saída e para estes, é imperativo criar uma estratégia de retenção, caso seja do interesse da empresa reter esses colaboradores.

$p(X)$	Descrição do Risco	
[0 ; 0,21 [	Sem risco de saída	
[0,21 ; 0,51 [	Risco de saída baixo	
[0,51 ; 0,61 [	Risco de saída médio	
[0,61 ; 0,79 [	Risco de saída elevado	
[0,79 ; 1,00]	Risco de saída máximo	

Tabela 8 – Níveis de risco de saída

7. Conclusão

A retenção de talentos e a redução do *turnover* voluntário passaram a ser uma das questões centrais para muitas organizações, devido ao impacto negativo que este fenómeno pode causar nos custos operacionais, na produtividade e no ambiente saudável das empresas.

Sendo o turnover voluntário uma decisão tomada pelos próprios colaboradores, constitui um desafio ainda maior para as organizações, que não têm controlo direto sobre essa saída. Isso reforça a necessidade de uma abordagem proativa para antecipar e reduzir os riscos envolvidos.

Nesse contexto, o principal objetivo deste estudo foi desenvolver um modelo preditivo que ajudasse a empresa X, do setor segurador, a identificar os colaboradores com maior probabilidade de saída voluntária, bem como entender as principais variáveis que levam a essa saída, possibilitando a criação de estratégias de retenção mais eficazes. A metodologia CRISP-DM, amplamente reconhecida na área de *data mining*, foi fundamental para orientação metodológica de todas as fases do projeto sem saltar etapas, desde a compreensão do negócio e dos dados, até à modelação, avaliação e implementação do modelo final.

Entre os modelos testados, a Floresta Aleatória destacou-se como a solução mais robusta. Apresentou métricas de avaliação no conjunto de treino similares ao XGBoost e AdaBoost, mas no conjunto de teste, demonstrou-se o melhor modelo a lidar com o *overfitting*. A Floresta Aleatória obteve uma PCCC de 97,07%, precisão de 82,35%, sensibilidade de 82,35%, especificidade de 98,40% AUC de 96,59% e *F1-Score* de 82,35%, valores que superaram de forma inequívoca os restantes modelos. A capacidade deste modelo de identificar corretamente os colaboradores com maior risco de saída foi um ponto crucial, tornando-o uma ferramenta valiosa para a estratégia de retenção de talentos no departamento de Recursos Humanos da empresa X.

Em relação aos principais fatores que influenciam o turnover voluntário, a análise apontou que variáveis como idade, promoções, antiguidade, salário exercem influência significativa nas decisões de saída, por esta ordem de importância. Colaboradores mais jovens, com poucas ou nenhumas promoções e menor antiguidade demonstraram uma maior propensão a deixar a organização.

A implementação do modelo permitiu a criação de diferentes níveis de risco de saída, divididos em cinco categorias, com base na probabilidade de saída de cada colaborador: sem risco, risco baixo, risco médio, risco elevado e risco máximo. Através destes grupos e perfis associados, a empresa pode desenvolver ações específicas para cada grupo, como revisões salariais e planos de progressão de carreira, de modo a reter os colaboradores mais valiosos e reduzir o *turnover* voluntário.

Os objetivos propostos pelo estudo foram alcançados com sucesso. O modelo preditivo desenvolvido permite prever com boa precisão a probabilidade e risco de *turnover* e oferece também uma visão dos fatores que influenciam essa decisão, possibilitando à empresa, adotar uma abordagem mais estratégica e personalizada na gestão de talentos. Este estudo demonstra que a aplicação de técnicas de *machine learning* pode proporcionar uma vantagem competitiva significativa, ao capacitar as empresas para antecipar a intenção de saída voluntária e reduzir o impacto negativo dessas saídas.

No geral, este trabalho confirma que a utilização de algoritmos de *machine learning*, em particular da Floresta Aleatória, é uma solução eficaz para a previsão de *turnover* voluntário, sendo possível identificar com precisão os colaboradores em risco e os principais fatores que influenciam essa decisão. Este projeto, proporciona assim um contributo relevante para a gestão de recursos humanos, oferecendo ferramentas práticas para a retenção de talento que resulta numa redução dos custos associados ao *turnover*.

Além disso, este projeto permitiu-me colocar em prática alguns dos conhecimentos adquiridos ao longo do mestrado em Métodos Quantitativos para Decisão Económica e Empresarial, principalmente relacionados com a unidade curricular de Fundamentos de Ciência de Dados, e foi essencial para aprofundar conhecimentos nessa área, devido à vertente prática e de análise de dados em situação empresarial real.

8. Limitações e Recomendações

As principais limitações deveram-se ao reduzido número de observações da base de dados e ao desbalanceamento existente entre colaboradores ativos e saídas. A base de dados era muito pequena, para o habitual neste tipo de problemas, e com menos de 10% saídas foi uma grande limitação do projeto.

Para um futuro trabalho, ou continuação do projeto aqui elaborado, sugere-se a realização de um benchmark externo. Este benchmark permitirá, através de uma banda salarial mediana, aferir se o colaborador está acima, abaixo, ou dentro desta banda salarial, o que pode ser um indicador muito importante na decisão da saída voluntária. O colaborador dentro da empresa pode estar considerado como um vencimento normal para a função, mas comparativamente ao mercado, estar abaixo das espectativas. Isto pode ocorrer pelo facto de certas funções, serem embrionárias na empresa, ou não desempenharem funções com um nível de dificuldade elevado, o que faz com que aquela função na empresa não seja tão bem remunerada, como a mesma função num mercado concorrente que tenha uma importância e dificuldade técnica acrescida.

Sugere-se também a realização de questionários anuais de satisfação, que não sejam anónimos, de forma a comparar os índices de resposta dos colaboradores ativos, com os que saem voluntariamente. Esta sugestão pode ser mais difícil de implementar, uma vez que as pessoas tendem a responder de forma mais honesta em questionários anónimos, mas destes questionários anónimos nada podemos concluir em termos individuais.

Por fim, sugere-se um aprofundamento em técnicas para lidar com *overfitting*, uma vez que os dados sofreram com esse problema.

Referências bibliográficas

Allen, D.G., Bryant, P.C., & Vardaman, J.M. (2010). Retaining Talent: Replacing Misconceptions with Evidence-Based Strategies. *Academy of Management Perspectives*, *24*(2), 48-64. https://doi.org/10.5465/amp.24.2.48

Alhamid, M., (2021). Ensemble Models. Medium. https://towardsdatascience.com/ensemble-models-5a62d4f4cb0c

Arthur, J. B. (1994). Effects of Human Resource Systems on Manufacturing Performance and Turnover. *Academy of Management Journal*, *37*(3), 670-687.

Ayele, W. Y. (2020). Adapting CRISP-DM for idea mining: A data mining process for generating ideas using a textual dataset. *International Journal of Advanced Computer Science and Applications*, *11*(6), 20–32.

Belyadi, H., & Haghighat, A. (2021). Supervised learning. In *Machine learning guide for oil and gas using Python*. https://dokumen.pub/qdownload/machine-learning-guide-for-oil-and-gas-using-python-a-step-by-step-breakdown-with-data-algorithms-codes-and-applications-0128219297-9780128219294.html

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5-32. https://doi.org/10.1023/A:1010933404324

Breiman, L., Friedman, J., Olshen, R. and Stone, C. (1984). *Classification and regression trees* (1st ed.). Chapman and Hall/CRC. https://doi.org/10.1201/9781315139470

Cascio, W. F. (2006). *Managing Human Resources: Productivity, Quality of Work Life, Profits* (7th ed.). McGraw-Hill.

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). *CRISP-DM 1.0: Step-by-step data mining guide*. USA: SPSS Inc.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Association for Computing Machinery*, 785-794. https://dl.acm.org/doi/pdf/10.1145/2939672.2939785

Eidt, L. (2023). *The Impact of Telework on Employee Motivation and Turnover Intention* [Tese de Mestrado, University of Twente]. Repositório da Universidade de Twente. https://essay.utwente.nl/94856/

Fawcett, T. (2006). An introduction to ROC analysis. *Pattern Recognition Letters*, *27*(8), 861-874. https://doi.org/10.1016/j.patrec.2005.10.010

Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, *55*(1), 119–139. https://doi.org/10.1006/jcss.1997.1504

Griffeth, R. W., Hom, P. W., & Gaertner, S. (2000). A meta-analysis of antecedents and correlates of employee turnover: Update, moderator tests, and research implications for the next millennium. *Journal of Management*, *26*(3), 463-488. https://journals.sagepub.com/doi/epdf/10.1177/014920630002600305

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). Springer. https://doi.org/10.1007/978-0-387-84858-7

He, H., & Garcia, E. A. (2009). Learning from Imbalanced Data. *IEEE Transactions on Knowledge and Data Engineering*, *21*(9), 1263-1284. https://ieeexplore.ieee.org/document/5128907

Henneberger, F., & Sousa-Poza, A. (2002). Demographic and economic determinants of turnover: Evidence from Switzerland. *Schweizerische Zeitschrift für Volkswirtschaft und Statistik*, *138*(4), 563-579. https://www.sjes.ch/papers/1990-IV-1.pdf

Holtom, B. C., Mitchell, T. R., Lee, T. W., & Eberly, M. B. (2008). Turnover and Retention Research: A Glance at the Past, a Closer Review of the Present, and a Venture into the Future. *Academy of Management Annals*, *2*(1), 231-274. https://doi.org/10.5465/19416520802211552

Hom, P. W., Lee, T. W., Shaw, J. D., & Hausknecht, J. P. (2017). One Hundred Years of Employee Turnover Theory and Research. *Journal of Applied Psychology*, *102*(3), 530-545. https://doi.org/10.1037/apl0000103

Isaac, R. G., Zerbe, W. J., & Pitt, D. C. (2001). Leadership And Motivation: The Effective Application Of Expectancy Theory. *Journal of Managerial Issues, 13*(2), 212–226. http://www.jstor.org/stable/40604345

James, G., Witten, D., Hastie, T., Tibshirani, R., & Taylor, J. (2023). *An Introduction to Statistical Learning with Applications in Python*. Springer.

Liaw, A., & Wiener, M. (2002). Classification and Regression by randomForest. *R News, 2*(3), 18-22. https://journal.r-project.org/articles/RN-2002-022/RN-2002-022.pdf

Liu, M., Yang, B., & Song, Y. (2024). Research on Predicting the Turnover of Graduates Using an Enhanced Random Forest Model. *Behavioral Sciences, 14*(7), 562. https://doi.org/10.3390/bs14070562

Marchington, M., & Wilkinson, A. (2020). *Human Resource Management at Work: People Management and Development*. Kogan Page.

Raschka, S. (2015). *Python Machine Learning*. Packt Publishing Ltd.

Saito, T., & Rehmsmeier, M. (2015). The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PloS One, 10*(3), 1-21. https://doi.org/10.1371/journal.pone.0118432

Sarkar, D., Panwar, N., Bali, R., & Ghosh, T. (2018). *Hands-On Transfer Learning with Python*. Packt Publishing. https://subscription.packtpub.com/book/data/9781788831307/1/ch01lvl1sec04/crisp-dm

Schapire, R. E. (2003). The boosting approach to machine learning: An overview. In D. D. Denison, M. H. Hansen, C. Holmes, B. Mallick, & B. Yu (Eds.), *Nonlinear estimation and classification* (pp. 149-171). Springer. https://doi.org/10.1007/978-0-387-21579-2_9

Shaw, D., Duffy, K., Johnson, L., & Lockhart, E. (2005). Turnover, social capital losses, and performance. *Academy of Management Journal*, *48*(4), 594–606. https://doi.org/10.5465/amj.2005.17843940

Vroom, V. H. (1964). *Work and Motivation*. New York: Wiley.

Wang, Y., Zhang, X., & Zhang, W. (2020). Using Machine Learning to Predict Employee Turnover in the Service Industry. *Expert Systems with Applications*, 142, 113019. https://doi.org/10.1016/j.eswa.2019.113019

Zhao, Y., Hryniewicki, M. K., Cheng, F., Fu, B., & Zhu, X. (2019). Employee turnover prediction with machine learning: A reliable approach. In K. Arai, S. Kapoor, & R. Bhatia (Eds.), *Intelligent Systems and Applications* (Vol. 869, pp. 737–758). Springer. https://doi.org/10.1007/978-3-030-01057-7_56

Anexos

Anexo A1 – Pesquisa de Hiperpârametros e respetivos melhores hiperparâmetros dos vários modelos

	Pesquisa Hiperparâmetros	Melhores Hiperparâmetros
Regressão Logística	'C': uniform(0.01, 10)	'C': 3.7446396818500567
	'penalty': ['l1', 'l2']	'penalty': 'l1'
	'class_weight': ['balanced', {0: 0.35, 1: 0.65}]	'class_weight': 'balanced'
Árvore de Decisão	'criterion': ['gini', 'entropy']	'criterion': 'gini'
	'splitter': ['best', 'random']	'splitter': 'best'
	'max_depth': randint(5, 25)	'max_depth': 20
	'min_samples_split': randint(2, 10)	'min_samples_split': 3
	'min_samples_leaf': randint(1, 10)	'min_samples_leaf': 3
	'min_impurity_decrease': [0.0, 0.001, 0.01, 0.1]	'min_impurity_decrease': 0.0
	'class_weight': [{0: 0.35, 1: 0.65}, {0: 0.4, 1: 0.6}, 'balanced']	'class_weight': {0: 0.35, 1: 0.65}
Floresta Aleatória	'n_estimators': randint(100, 600)	'n_estimators': 360
	'max_depth': randint(5, 20)	'max_depth': 15
	'min_samples_split': randint(2, 15)	'min_samples_split': 13
	'min_samples_leaf': randint(1, 15)	min_samples_leaf': 1
	'criterion': ['gini', 'entropy']	'criterion': 'entropy'
	'class_weight': [{0: 0.35, 1: 0.65}, 'balanced']	'class_weight': 'balanced'
	'min_impurity_decrease': [0.0, 0.001, 0.01, 0.1]	'min_impurity_decrease': 0.001
XGBoost	'n_estimators': randint(500, 1000)	'n_estimators': randint(500, 1000)
	'max_depth': randint(5, 25)	'max_depth': 6
	'learning_rate': uniform(0.001, 0.03)	'learning_rate': 0.011894640243622707
	'subsample': uniform(0.5, 0.5)	'subsample': 0.7956641857754387
	'colsample_bytree': uniform(0.5, 0.5)	'colsample_bytree': 0.7262818486236282
AdaBoost	'n_estimators': randint(500, 1000)	'n_estimators': 816
	'learning_rate': uniform(0.001, 0.1)	'learning_rate': 0.08824262944092157
	'estimator__max_depth': randint(5, 20)	'estimator__max_depth': 11
	'estimator__criterion': ["entropy", "gini"]	'estimator__criterion': 'entropy'

Anexo A2 – Importância Relativas dos Preditores dos diversos modelos