DM

# Inteligência Artificial Aplicada à Avaliação de Crédito Bancário

DISSERTAÇÃO DE MESTRADO

**João Artur Vieira Santos**
MESTRADO EM ENGENHARIA INFORMÁTICA

UNIVERSIDADE da MADEIRA
A Nossa Universidade
www.uma.pt

junho | 2022

# Inteligência Artificial Aplicada à Avaliação de Crédito Bancário

DISSERTAÇÃO DE MESTRADO

**João Artur Vieira Santos**
MESTRADO EM ENGENHARIA INFORMÁTICA

ORIENTAÇÃO
Karolina Baras

COORIENTAÇÃO
Pedro Camacho

FCEE

Mestrado em Engenharia Informática

# Inteligência Artificial aplicada à avaliação de crédito bancário

*João* Santos

supervisionado por

Prof. Dr. Karolina Baras and Eng. Pedro Camacho

20 de junho de 2022

# Resumo

A avaliação de crédito é uma ferramenta financeira, importante, para bancos e instituições financeiras determinarem se devem emitir o empréstimo para potenciais mutuários. A utilização de inteligência artificial levou a um melhor desempenho dos modelos de avaliação de crédito. Várias técnicas de *machine learning* baseadas em estatísticas foram empregues para esta tarefa, sendo que a regressão logística é o padrão da indústria na modelação de risco de crédito. Porém, estudos demonstram que algoritmos de *ensemble*, que principalmente podem ser divididos em *bagging ensembles* e *boosting ensembles*, têm se mostrado muito promissores.

Esta tese tem como objetivo comparar diversos tipos de modelos de *machine learning* de forma a determinar quais oferecem o melhor desempenho para a classificação de crédito bancário. Para tal, este estudo irá realizar comparações com diversos tipos de modelos de classificação, desde os modelos tradicionais, como *Logistic regression* (LR), *Linear discriminant analysis* (LDA) e *Artificial neural network* (ANN), a modelos mais recentes como *ensemble* homogéneos, tais como *AdaBoost*, *Gradient-Boosted Decision Trees* (GBDT), *eXtreme Gradient Boosting* (XGBoost), *Light Gradient Boosting Machine* (LightGBM) e *CatBoost*, até modelos mais experimentais como o caso de modelos *ensemble* heterogéneos.

A contribuição final desta tese será fornecer informação de que modelos de *machine learning* atualmente mais se adequam a avaliação de crédito bancário, com intuito de substituir os métodos tradicionais.

**Keywords:** Artificial intelligence · Machine Learning · Homogeneous ensembles · Heterogeneous ensembles · Supervised Learning · Credit Scoring

# Abstract

Credit assessment is an important financial tool for banks and financial institutions to determine whether to issue the loan to potential borrowers. The use of artificial intelligence led to a better performance of credit assessment models. Several statistical-based machine learning techniques were employed for this task, with logistic regression being the industry standard in credit risk modeling. However, studies show that ensemble algorithms, which mainly can be divided into bagging ensembles and boosting ensembles, have shown to be very promising.

This thesis aims to compare different types of machine learning models in order to determine which ones offer the best performance for bank credit rating. To this end, this study will carry out comparisons with different types of classification models, from traditional models like Logistic regression (LR), Linear discriminant analysis (LDA) and Artificial neural network (ANN) to more recent models such as homogeneous ensemble like AdaBoost, Gradient-Boosted Decision Trees (GBDT), eXtreme Gradient Boosting (XGBoost), Light Gradient Boosting Machine (LightGBM) and CatBoost to more experimental models such as the case of heterogeneous ensemble models.

The final contribution of this thesis will be to provide information on which machine learning models are currently best suited to bank credit assessment, in order to replace traditional methods.



**Keywords:** Artificial intelligence · Machine Learning · Homogeneous ensembles · Heterogeneous ensembles · Supervised Learning · Credit Scoring

# Agradecimentos

Antes de mais, gostaria de agradecer a toda equipa Nearsoft solutions pelas facilitações que me concederam e pelo apoio incondicional, ao longo destes meses. Sem eles, nada disto seria possível.

À orientadora desta dissertação, a Doutora Professora Karolina Baras, pela orientação prestada, pelo seu incentivo, disponibilidade e apoio que sempre demonstrou.

Ao coorientador Engenheiro Pedro Camacho, pelo seu incentivo, disponibilidade e apoio que sempre me demonstrou. Aqui lhe exprimo a minha graditão.

A todos os amigos e colegas que de uma forma directa ou indirecta, contribuíram, ou auxiliaram na elaboração do presente estudo, pela paciência, atenção e força que prestaram em momentos menos fáceis. Para não correr o risco de não enumerar algum não vou identificar ninguém, aqueles a quem este agradecimento se dirige sabê-lo-ão, desde já os meus agradecimentos.

A minha namorada por ter caminhado ao meu lado, pela sua paciência, compreensão e ajuda prestada durante a elaboração da presente dissertação. Agradeço-lhe por ser a fonte da minha determinação e alegria.

Um agradecimento especial ao Senhor Luís Manuel Rodrigues Alves, que sempre me tratou como um filho, aconselhando-me, guiado-me sempre para um melhor rumo. Sempre acreditando em mim, mesmo quando eu próprio não acreditava. Sua companhia faz muita falta. Obrigado do fundo do meu coração.

Por último, mas não menos importante, devo expressar a minha profunda gratidão aos meus pais e familiares que sempre me apoiaram e encorajaram ao longo dos meus anos de estudo. Esta dissertação certamente não seria possível sem eles. Obrigado a todos.

João Artur Vieira Santos

# Índice

# Lista de Figuras

# Lista de Tabelas

# Nomenclatura

*AdaBoost* Adaptive Boosting

*ANN* Artificial Neural Networks

*ANN* Artificial Neural Networks

*AUC* Area under ROC curve

*BS* Brier Score

*CSV* Comma-separated values

*DL* Deep Learning

*DT* Decision Tree

*EDA* Exploratory Data Analysis

*FN* False Negative

*FP* False Positive

*GBDT* Gradient-Boosted Decision Trees

*GMSC* Give Me Some Credit

*GPU* Graphics Processing Unit

*LDA* Linear Discriminant Analysis

*LR* Logistic Regression

*LSTM* Long Short-Term Memory

*OCHE* Overfitting-Cautious Heterogeneous Ensemble

*PCA* Principal Component Analysis

*RF* Random Forest

*ROC* Receiver Operating Characteristic

*SVM* Support Vector Machine

*TN* True Negative

*TP* True Positive

x

*XGBoost* eXtreme Gradient Boosting

# 1 Introdução

Ao longo dos anos tem havido um enorme progresso em *hardware* de computador, software e em tecnologias web o que levou a uma mudança irreversível na nossa sociedade. Hoje é difícil imaginar um agente económico sem computadores, Internet ou dispositivos móveis. A evolução das Tecnologias de informação oferecem uma grande oportunidade para as empresas expandirem a sua base de clientes, introduzindo novos produtos ou melhorar os já existentes e aumentando assim a sua eficiência. Entre os avanços Tecnológicos da Informática, o progresso feito no ramo da Inteligência Artificial é particularmente notável. O setor financeiro foi um dos primeiros a fazer experiências com tecnologias de IA, principalmente devido à sua provável contribuição para um aumento de lucro. Portanto, é essencial observar o papel que a IA tem na transformação da banca digital.

A inteligência artificial(IA) e as técnicas de *machine learning* estão em transformação e irão revolucionar a forma como é abordada a gestão de risco financeiro. Investimentos em tais tecnologias tornaram o uso de soluções de IA essenciais, para compreender e controlar riscos, tais como, decidir a quantia que um banco deve emprestar a um cliente, fornecer sinais de alerta aos operadores do mercado financeiro sobre o risco de posição, detetar tentativas de fraudes internas ou externas, melhoria na conformidade e redução no modelo de risco.

Tendo em consideração estes riscos, o principal objetivo desta tese é obter uma melhor compreensão de como utilizar inteligência artificial para a avaliação de risco de crédito, com o foco em *machine learning* e na investigação de quais as técnicas, ferramentas, métricas e modelos utilizados neste momento.

A avaliação do risco de crédito é um tema indispensável nas instituições financeiras. O risco de crédito é definido como a probabilidade, de um mutuário não pagar o valor emprestado. A decisão de conceder ou rejeitar um empréstimo é muito crítica e baseia-se nas informações pessoais do solicitante, histórico de crédito, status de vida e fidelidade [1]. Esta tese estuda em que medida é fiável prever o risco de classificação de crédito, utilizando *machine learning* e qual será atualmente o melhor modelo de *machine learning* a utilizar, para cumprir este objetivo. Será realizada uma investigação empírica, com o objetivo de comparar e analisar qual será a melhor abordagem de utilização de IA.

## 1.1 Motivação

Esta tese tem como motivação compreender o papel, que a inteligência artificial tem na área da classificação de crédito financeiro, nomeadamente o impacto que o desenvolvimento de novos mod-

elos ou abordagens têm nesta área, investigando quais as abordagens que já estão a ser produzidas e concluir quais destas apresentam melhores resultados.

### 1.1.1 Inteligência artificial

Inteligência artificial é a capacidade de um sistema de computadores imitar funções cognitivas humanas, como aprendizagem e resolução de problemas. Através da IA, um sistema de computadores utiliza matemática e lógica para simular raciocínio que os seres humanos utilizam para aprender novas informações e tomar decisões.

*Machine learning* é uma aplicação de IA. É o processo de usar modelos matemáticos de dados para ajudar um computador a aprender sem instrução direta. Isto permite que um sistema informático continue a aprender e a melhorar por si só, com base na experiência [2].

### 1.1.2 Visão geral dos problemas a serem explorados

A gestão do risco de crédito é essencial para as instituições financeiras, cuja atividade principal é a concessão de crédito. A avaliação do crédito do consumidor ou da empresa é de extrema importância,visto que, instituições financeiras podem incorrer em perdas significativas quando os tomadores de crédito entram em inadimplência.

Para controlar as suas perdas de contas incobráveis, as instituições financeiras, precisam avaliar adequadamente os riscos de crédito dos mutuários. Consequentemente, foi necessário reunir os dados dos mutuários, resultando no desenvolvimento de diversos métodos estatísticos para medir e analisar o risco de crédito de forma objetiva.

Devido à sua importância académica e prática, muitas investigações têm sido conduzidas nesta área. Como por exemplo, estudos comparativos como Xia et al. [3], Dastile et al. [4] e Liu et al. [5] comparam metodologias e classificadores com ideias modernas, de forma a conseguirem criar novas abordagens e soluções para o problema em questão.

## 1.2 Objetivos

- **[O1.] Elaboração de um planeamento de modelação.** - O objetivo é elaborar as fases necessários para a realização das experiências da tese. Processo fundamental para o desenvolvimento de algoritmos de *machine learning*, englobando todas áreas necessárias do processo de modelação desde a recolha de *datasets*, a análise e tratamento de dados, a investigação de algoritmos de *machine learning*, métricas de avaliação, bibliotecas, etc..
- **[O2.] Determinar quais os modelos de ML mais favoráveis para avaliação de crédito bancário.** - Tendo em conta os estudos abordados e quais as técnicas/modelos mais utilizados

atualmente em avaliação de crédito bancário, a tese tem como objetivo principal abordar possíveis novas técnicas de ML que possivelmente ponderam substituir as comummente utilizadas.

### 1.3 Estrutura do relatório

A tese está organizada da seguinte forma: (1) **Introdução** secção que contextualiza o tema abordado pela tese, apresenta a motivação, os problemas a serem explorados e os objetivos da tese; (2) **Estado atual do conhecimento** a secção onde é retratado a investigação realizada pela tese com o intuito de recolher conhecimento necessário para a realização da mesma. É abordado os estudos mais recentes sobre a avaliação de credito bancário, as bibliotecas e ferramentas utilizadas, os modelos de *machine learning* mais adequados e as métricas de avaliação comummente utilizadas para avaliar tais modelos; (3) **Metodologia** aborda as etapas necessárias para a realização da parte experimental da tese, tendo como ponto fulcral a elaboração de um planeamento de modelação que planifique todas etapas necessárias para a resolução das experiências desejadas; (4) **Analise e tratamento dos dados** nesta secção a tese ira abordar os dados recolhidos de forma a conseguir ter uma melhor perceção dos dados e prepara los para o processo de modelação, para tal será aplicado o processo de *Exploratory Data Analysis* (EDA) para de seguida ser realizado um pré-processamento de dados que deixará os conjuntos de dados aptos para o processo de modelação com algoritmos de *machine learning*; (5) **Resultados** secção onde será apresentado os resultados provenientes do processo do planeamento de modelação; (6) **Conclusão** irá abordar os objetivos da tese, limitações, recomendações para trabalhos futuros e as principais conclusões retiradas pela a investigação.

# 2 Estado atual do conhecimento

De forma a conseguir obter um contexto sobre o tema que será abordado ao longo desta tese, será fundamental a realização de um estudo do estado atual do conhecimento. Esta secção ira abordar as tecnologias e técnicas que estão a ser empregues neste momento, bem como uma visão geral dos trabalhos já realizados nesta área, relatados por investigadores que nos seus estudos utilizaram e compararam diversos tipos de classificadores, utilizando diversos tipos de métricas com vários *datasets* distintos.

Avaliação de crédito financeiro é um dos processos mais cruciais no setor da indústria financeira por ser capaz de avaliar a qualidade de crédito de indivíduos e empresas. Várias técnicas de *Machine learning* baseadas em estatísticas têm sido empregadas para essa tarefa. Segundo a pesquisa realizada por Jadhav [6] técnicas de ML, são usadas para construir modelos de avaliação de crédito, ajudando instituições financeiras a tomar decisões de aceitar ou rejeitar o crédito de certo cliente. Muito do sucesso do setor bancário depende fortemente da avaliação do risco de crédito dos potenciais devedores. A análise de risco de crédito é uma parte importante da gestão de risco financeiro. A classificação de crédito indica um nível relativo de risco de crédito e é uma abordagem analítica fundamental para avaliação de risco de crédito.

Com o objetivo de entender melhor como projetar um sistema de avaliação de crédito, as seguintes secções irão abordar quais as bibliotecas, ferramentas, tópicos, *datasets*, algoritmos e métricas que estão a ser utilizadas:

- A secção 1 aborda os estudos já realizados nesta área, analisando-os de forma a saber quais os algoritmos, *datatsets* e métricas mais utilizadas neste tipo de investigação.
- A secção 2.2 será uma visão geral de qual a linguagem de programação que ira ser utilizada e quais as suas bibliotecas fundamentais.
- Na secção 2.3 irá abordar o uso de *supervised leaning* e os modelos utilizados.
- A secção 2.4 irá abordar quais as métricas a utilizar para avaliar os modelos.
- Por fim a secção 2.5 fará um resumo das observações mais importantes tiradas do estado atual do conhecimento.

## 2.1 Revisão da literatura

Nesta secção, a tese irá abordar investigações/estudos já realizados nesta área com objetivo de averiguar quais os *datasets* que existem direcionados a classificação de crédito, quais os modelos utilizados nestas investigações e quais as métricas de avaliação que são mais utilizadas para avaliar e comparar tais modelos.

É de mencionar que todos os artigos nesta revisão de literatura foram selecionados por serem os mais atuais possíveis, utilizando tecnologias e conhecimento atualizado.

Na tabela 1 é possível visualizar todos os artigos presentes para esta revisão de literatura e quais os *datasets* , algoritmos e métricas utilizados pelos autores.

Tabela 1: Trabalhos Relacionados

| **Autor** | **Ano** | **Datasets** | **Algoritmos** | **Métricas** |
|---|---|---|---|---|
| Xia et al [3] | 2020 | Crédito Australiano, Crédito Alemão, Crédito Japonês, PPDai, GMSC[1] | OCHE, RF, GBDT, XGBoost, CatBoost, LightGBM, LR, DT, NB , ANN, SVM | Exactidão, AUC, H-measure, Brier Score |
| Liu et al [5] | 2020 | Crédito Australiano, Crédito Alemão, Crédito Japonês, crédito Tailandês Lending Club, WE | Augmented Gradient Boosting, Decision Tree | Exatidão, AUC, Brier Score, Sensibilidade Score, Precisão Score |
| Shen et al [7] | 2020 | Crédito Alemão, Crédito Tailandês | Adaptive Boosting, LSTM[2], NN | AUG, Kolmogorov–Smirnov Statistic |
| Dastile et al [4] | 2020 | Crédito Alemão, Crédito Australiano, | Boosting, ANN, Bagging, DT,KNN, LR, NB, RF , SVM, XGB, CNN, LDA | PCC, AUC, Type I, Type II, F-measure, G-mean, K-S, Sensibilidade |
| Mercep et al [8] | 2020 | Dados proprietários do Banco da Croácia | LR, SVM, RF, XGB, Deep feedfoward | ROC, H-measure, Brier Score |
| Pereira et al [9] | 2020 | CRSP[X] | LR, LDA, GNB, DT, RF, SVM e kNN | AUC, Exatidão |
| Hamori et al [10] | 2018 | Crédito Alemão, CréditoTailandês | NN, Bagging, Boosting, RF , DNN | ROC, AUC, Exatidão, F-score |
| Barboza et al [11] | 2017 | American Canadian (1985 - 2013) | SVM, Boosting, Bagging, RF, Neural Networks | Type I , Type II, AUC, ACC |

[1] GMSC - "Give Me Some Cash" [12]
[2] LSTM - "Long-Short-Term-Memory"
[X] CRSP - "Center for Research in Security Prices" [12]

A maioria dos artigos publicados utilizam um ou mais *datasets* disponíveis publicamente, como os *datasets* de crédito UCI [13] ou os dados que foram disponibilizados para um evento, *Kaggle Give Me Some Credit* (GMSC )[12]. No artigo de Mercep et al [8] foram explorados mais de

187 artigos e quarenta e cinco por cento desses artigos usaram os *datasets* de crédito da UCI australiano, japonês ou alemão. Existindo exceções, visto que, alguns investigadores desenvolvem modelos de classificação de crédito corporativo, e outros colaboram com instituições financeiras e assim obtendo acesso a dados proprietários.

O tamanho da amostra de desenvolvimento também varia significativamente, variando de menos de 1000 exemplares (por exemplo, conjuntos de dados UCI, dados corporativos), até 150.000 no caso de dados do GMSC.

Analisando os estudos representados na tabela 1 é possível verificar estas mesmas observações feitas por Mercep et al [8], em que os *datasets* mais utilizados para a investigação de modelos de classificação de crédito são os três *datasets* mais populares de crédito da UCI[13], nomeadamente, o australiano, japonês e o alemão. Estes *datasets* são comummente usados em trabalhos relacionados, o que viabiliza a realização de comparações detalhadas entre vários artigos. No entanto, estes são relativamente pequenos e, portanto, estão um pouco longe da realidade de avaliação de crédito.

É possível observar na figura 1, um gráfico de barras representando o número de utilizações de cada *dataset* nos artigos escolhidos nesta revisão de literatura.

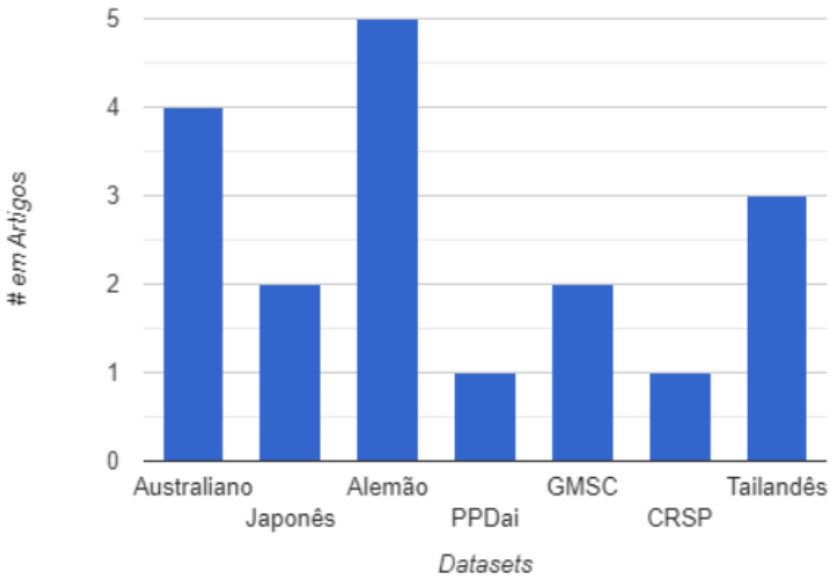



Fig. 1: Datasets utilizados nos artigos desta secção

Em relação aos algoritmos mais utilizados para elaboração de modelos de classificação binária nos artigos abordados é de salientar os resultados obtidos utilizando métodos *Ensemble*. Estudos como Barboza et al [11] observaram que métodos de *Ensemble* como Bagging e Boosting apresentam melhores resultados do que métodos tradicionais como LR, LDA e ANN, apresentando

resultados, em média, aproximadamente 10% mais precisos. Este estudo conclui que a utilização de classificadores homogéneos apresentam uma performance superior em relação a classificadores individuais.

A investigação de Hamori et al [10] obteve resultados semelhantes. Foram empregues os métodos de ensemble(bagging e boosting) e oito métodos de redes neuronais com diferentes funções de ativação. O desempenho de cada método foi comparado em termos da sua capacidade de prever o risco de inadimplência usando varias métricas de validação como a exatidão, ROC , AUC e F-score. Os resultados obtidos neste estudo afirmam que o desempenho dos métodos de *ensemble*, nomeadamente, da classificação por Boosting é superior a de outros métodos de *Machine Learning* incluindo de redes neuronais.

Xia et al [3] conclui que a utilização de métodos de ensemble oferecem um desempenho superior, o que tem atraído muita atenção de investigadores da área de avaliação de riscos de crédito. Isto consegue-se observar na figura 2 onde o numero de vezes que métodos de ensemble são usados em artigos de investigação é considerável.

Neste artigo é desenvolvido um modelo de avaliação de crédito tree-based overfitting-cautious heterogeneous ensemble (OCHE), que envolve cinco algoritmos eficientes baseados em árvore, nomeadamente, RF , Gradient-Boosted Decision Trees(GBDT), XGBoost, LightGBM e CatBoost. Uma estratégia de selecção de ensemble de overfitting cauteloso é desenvolvida como forma de atribuir pesos a modelos base dinamicamente. Em seguida, os resultados são combinados com métodos de média ponderada.

As comparações feitas com OCHE em diversos datasets demonstraram que este modelo proposto tem um desempenho significativamente superior ao da maioria dos modelos individuais e ensemble homogéneos. Afirmando assim, que classificadores heterogéneos conseguem ser superior do que a classificadores homogéneos e individuais. Em termos de custos computacionais, esta proposta pode poupar até 30% do tempo de execução sob aceleração da Graphics Processing Unit(GPU) em relação ao suporte de CPU apenas.[3]

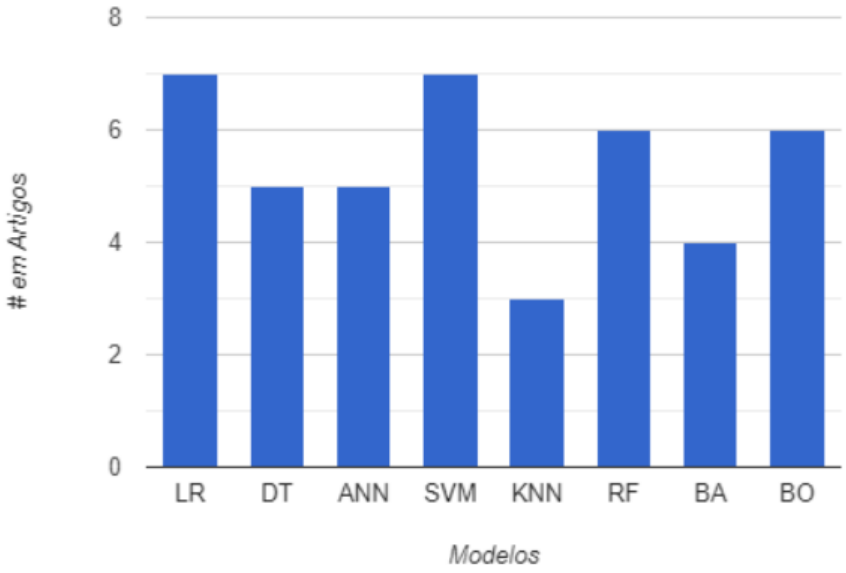



Fig. 2: Modelos utilizados nos artigos desta secção

O artigo Liu et al [5] propõe um modelo de *ensemble*, *multi-grained augmented gradient boosting decision trees (mg-GBDT)* para avaliação de crédito. Neste método proposto, um *scan* multi-granulado é introduzido para *feature augmentation*, o que enriquece os variáveis de entrada da GBDT. Resultados experimentais em 6 *datasets* de crédito mostram que mg-GBDT é superior aos algoritmos de *ensemble* avançados, como GBDT e XGBoost para avaliação de crédito, reduzindo o erro. Além disso, o benefício da estrutura baseada em árvore, a interoperabilidade global e interoperabilidade local do mg-GBDT é explorada, o que fornece resultados de decisão de modelo mais intuitivos e referências para gestores.

Na pesquisa realizada no artigo Dastile et al [4] foram realizados diversos testes de desempenho (no datset da UCI alemão e australiano) de modelos estatísticos, de ML clássico e de DL com o objetivo de realizar uma análise e comparação. Concluindo mais uma vez que classificadores de *ensemble* geralmente superam classificadores únicos. Este artigo recomenda que estudos futuros devem se concentrar mais no uso de classificadores de base heterogéneos em vez de homogéneos em métodos de *ensemble* para permitir uma diversidade de classificadores. Isto vai de acordo com conclusões já tiradas em outros artigos desta revisão de literatura.

Mercep et al [8] realizou um estudo onde utilizava *Deep Learning* para a classificação de crédito. Conclui que o modelo de DL ofereceu um ganho de desempenho esperado quando comparado a modelos lineares como LR, SVM-Linear. Além disso, todos os três modelos não lineares (DL, RF e XGBoost) treinados neste artigo conseguiram oferecer melhor desempenho do que os seus equivalentes lineares, com o modelo de DL e XGBoost obtendo os melhores resultados. Um obstáculo

com que se depararam foi avaliar qual destes dois modelos (DL e XGBoost ) seria o melhor, pois não é incomum que um modelo tenha uma pontuação melhor na métrica de avaliação AUC, enquanto que outro tenha uma H-measure mais alta. Dito isto, o fator de diferenciação será quais das mediadas de desempenho serão as mais relevantes para cada caso em especifico e escolher o melhor modelo de acordo.

Shen et al [7] aborda o problema de quando somos confrontados com *big data* para classificação de crédito onde a utilização de métodos tradicionais de estatística e de ML têm sido considerados difíceis de revelar a complexa relação entre as variáveis dos dados de créditos. No entanto, este artigo aplica uma solução baseada em DL, usando uma Long short-term memory (LSTM), que efetivamente irá aumentar a previsão prevista, apesar de esta tecnologia de DL não ser amplamente aplicada à avaliação de crédito. O artigo argumenta que seu desempenho poderá ser afetado pela escala e desequilíbrio dos dados. Um método SMOTE[1] melhorado foi proposto para processamento de dados de crédito desequilibrado.

O artigo propõe um modelo inovador juntando o sucesso obtido por investigações com modelos de *ensemble* e um método SMOTE melhorando para lidar com desequilíbrio dos dados com *Deep Learning*, isto resultará num modelo de *deep ensemble learning* construído com base na rede LSTM e no algoritmo AdaBoost utilizando SMOTE melhorando para avaliação de crédito. Para garantir uma comparação justa, todos os outros métodos de classificação também foram *ensembled* com Adaboost e o algoritmo de SMOTE melhorado. Em comparação com a abordagem de *ensemble* testada e os modelos individuais como LR, LDA , SVM, KNN e NB, o modelo de LSTM *ensemble* teve pontuações de AUC e KS significativamente melhores para os dois *datasets* de crédito (UCI alemão e tailandês).

Métricas de avaliação são ferramentas fundamentais para a avaliação dos desempenhos dos algoritmos e essencial para permitir a viabilidade de realizar uma comparação. É beneficiável a escolha de métricas comummente utilizadas em artigos e investigações para assim os resultados obtidos nesta tese serem fiáveis quando comparados com outros trabalhos relevantes da área. Na figura 3 é possível observar quais as métricas utilizadas nos artigos revistos nesta secção de revisão de literatura.

[1]Synthetic Minority Oversampling Technique(SMOTE) - Técnica utilizada para equilibrar *datasets*. [14]

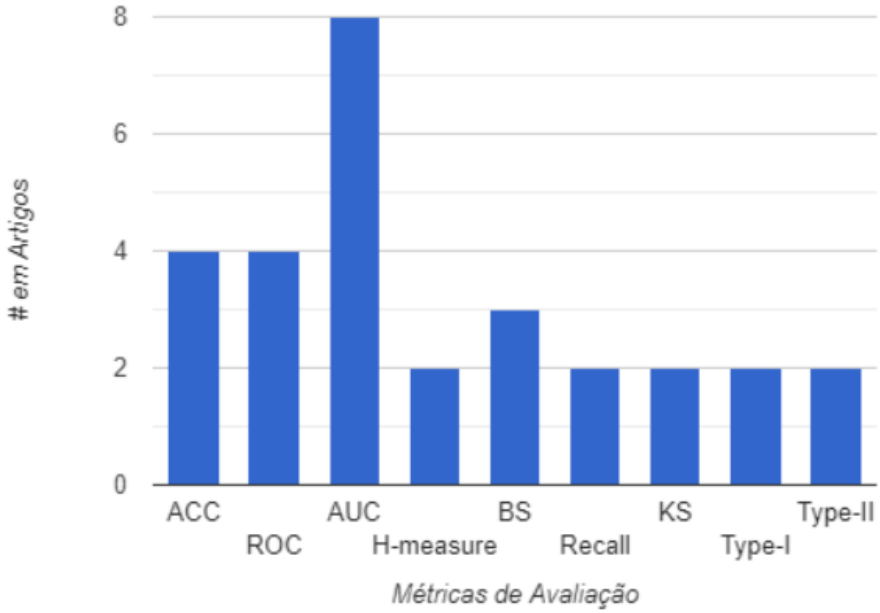



Fig. 3: Métricas utilizados nos artigos desta secção

## 2.2 Bibliotecas de Inteligência Artificial: Visão Geral

Em termos das plataformas utilizadas para Inteligência Artificial, existem muitos algoritmos e linguagens de programação(p.ex. LISP, R, Javascript e Java). No entanto, Python é uma das linguagens de programação mais dominantes e de mais rápido crescimento para *Machine learning* e *data science*. Python oferece uma grande flexibilidade, sendo fácil de aprender, possui um desenvolvimento rápido, têm uma grande comunidade ativa, e uma capacidade de trabalhar em complexas aplicações numéricas, científicas e de investigação.

Dada sua popularidade e presença em todos os artigos/livros pesquisados nesta tese, Python irá ser utilizado como a principal linguagem. Python oferece múltiplas bibliotecas para todas as fases do desenvolvimento de ML. Bibliotecas como NumPy e Pandas são de grande importância para representação de dados. Enquanto que bibliotecas como Sckit-learn, StatsModels, Keras, Theano e Tensorflow têm grande relevância em análise estatística e no processo de desenvolvimento de modelos de ML. Matplotlib e Seaborn facilitam a visualização de dados. Por último, de forma a conseguir gerir estas bibliotecas existe o pip e Conda. Na figura 4 podemos observar as bibliotecas de Python descritas.

Abaixo, é fornecida uma descrição das bibliotecas mencionadas na figura 4, cobrindo alguns dos seus benefícios e utilidades.

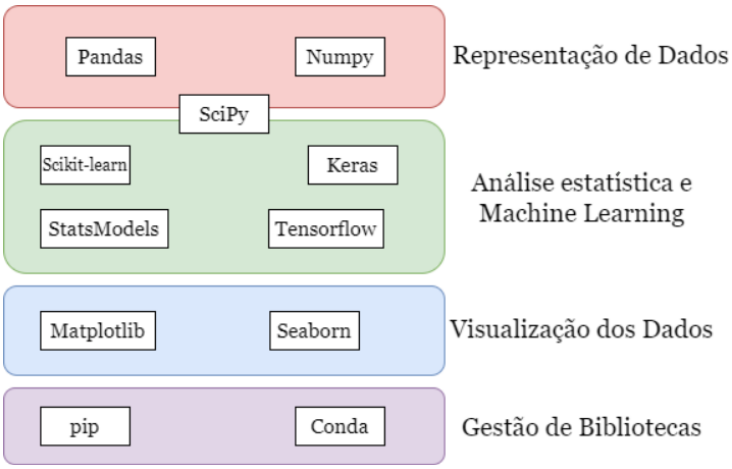



Fig. 4: Bibliotecas de Python

### 2.2.1 Conjunto SciPy

O pacote de bibliotecas SciPy não está diretamente relacionado com ML, mas muitas bibliotecas de ML confiam nos componentes SciPy no seu trabalho. Descrevamos brevemente os principais componentes incluídos neste conjunto.

- **NumPy** - é a biblioteca que implementa arrays e ferramentas de alto desempenho. O núcleo computacional é escrito em C (52% da base de código) com a parte de interface em Python (48% da base de código). As funções da Álgebra Linear dependem fortemente da biblioteca LAPACK. NumPy implementa uma variedade de métodos de álgebra linear para trabalhar com vetores, matrizes e tensores (neste caso, matrizes multidimensionais). Também suporta a computação paralela, utilizando as capacidades vetoriais das CPUs modernas.[15]

- **Scipy** - é a biblioteca que implementa muitos métodos matemáticos, tais como equações algébricas e soluções de equações diferenciais, interpolação polinomial, vários métodos de otimização, etc.[15]

- **Pandas** - é a biblioteca concebida para trabalhar com séries cronológicas e dados de tabelas (estrutura de dados DataFrame). É escrita quase inteiramente em arrays NumPy puros e é frequentemente utilizada em ML para organizar o treino e testar amostras.[15]

### 2.2.2 Keras

Keras é uma biblioteca Python que fornece ligações a outras ferramentas de Deep Learning (DL) tais como TensorFlow, CNTK, Theano, versão beta com MXNet e Deeplearning4j. Foi desenvolvida com foco em permitir uma experimentação rápida e é lançada sob a licença do MIT. Keras pode ser executada sem problemas em GPUs e CPUs, dadas as framewokrs subjacentes.

Os benefícios da utilização da Keras podem se resumir ao facto de ser código aberto, estar em rápida evolução e possuir ferramentas de backend que é utilizado por empresa industriais fortes como a Google e a Microsoft. Possui uma API popular para o desenvolvimento de DL com boa documentação. Utiliza uma forma limpa e conveniente de definir rapidamente modelos DL no topo dos backends (por exemplo, TensorFlow, Theano).Keras envolve bibliotecas de backend, abstraindo as suas capacidades e escondendo a sua complexidade.[16]

### 2.2.3 Scikit-Learn

Scikit-Learn é amplamente conhecido como uma ferramenta código aberto popular de Python que contém uma biblioteca abrangente de algoritmos de ML. O projecto Scikit-Learn começou como um projecto Google Summer of Code de David Cournapeau. Desde 2015, está em desenvolvimento activo patrocinado pelo INRIA, Telecom ParisTech e ocasionalmente pelo Google através do Google Summer of Code.[17]

Estende a funcionalidade dos pacotes NumPy e SciPy com numerosos algoritmos ML e fornece funções para realizar classificação, regressão, agrupamento, redução da dimensionalidade, selecção de modelos e pré-processamento. Utiliza também o pacote Matplotlib para traçar gráficos.[18]

As principais vantagens da utilização desta biblioteca é o facto de ser código aberto, comercialmente utilizável e muito popular, ser financiado pelo INRIA, Telecom Paristech, Google, entre outros. Possui um conjunto de algoritmos e implementações bem actualizados e abrangentes. Por fim, o seu rico ecossistema está intimamente ligado a pacotes estatísticos e científicos de Python.[16]

### 2.2.4 TensorFlow

TensorFlow é uma biblioteca de código aberto utilizada principalmente para DL. Foi originalmente desenvolvida pelas divisões do Google, mas em 2015 foi lançada como software de código aberto sob a licença Apache 2.0. A versão estável à data da escrita é a 2.5.0.[19]

O núcleo computacional é escrito em C++ (60% de todo o código) utilizando tecnologia CUDA, o que permite utilizar o potencial das placas gráficas nos cálculos. A parte da interface é implementada em Python (30% de toda a base de código). Existem também ligações não oficiais para outras linguagens, mas apenas as interfaces C++ e Python são oficialmente suportadas.[15]

A biblioteca baseia-se no princípio dos fluxos de dados (dataflow), segundo o qual o programa é organizado sob a forma de blocos computacionais associados uns aos outros sob a forma de um gráfico direccionado, que é chamado gráfico computacional. Os dados são processados através da passagem de um bloco para outro.[15]

Esta arquitetura facilita o uso de cálculos paralelos em CPUs de multi-núcleos e sistemas de cluster distribuídos. Além disso, é adequado para a construção de redes neuronais nas quais cada neurónio é apresentado por um componente independente.Além do gráfico computacional, o TensorFlow usa uma estrutura de dados chamada tensor. [15]

Tensor são matrizes multi-dimensionais com um tipo uniforme(chamado dtype). Podemos pensar em tensors como um tipo de array tal como os arrays em Numpy. Todos os tensores são imutáveis com os números e strings do Python, não é possível actualizar o conteúdo de um tensor, apenas cria um novo.[20]

O TensorFlow Lite é uma solução leve para dispositivos móveis e integrados. Ele permite a implementação de ML em dispositivos com baixa latência e um pequeno tamanho binário, mas tem cobertura para um conjunto limitado de operadores. Ele também oferece suporte à aceleração de hardware com a API Android Neural Networks.

Alguns dos benéficos que o TensorFlow oferece são: de longe é a ferramenta DL mais popular, código aberto, em rápida evolução, apoiada por uma empresa industrial forte (Google); biblioteca numérica para programação de fluxo de dados que fornece a base para a investigação e desenvolvimento DL; trabalha eficientemente com expressões matemáticas envolvendo matrizes multidimensionais; computação GPU/CPU, eficiente em configurações multi-GPU, computação móvel, alta escalabilidade de computação entre máquinas e possui enormes datasets. [16]

### 2.2.5 Matplotlib

Matplotlib é uma biblioteca de visualização de dados multiplataforma construída em arrays NumPy e projetada para funcionar com a pilha SciPy mais ampla. Foi concebido por John Hunter em 2002, originalmente como um patch para IPython para permitir realizar plotting interativo no estilo de MATLAB via gnuplot a partir da linha de comando IPython.[21]

Uma das características mais importantes da Matplotlib é a sua capacidade de funcionar bem com muitos sistemas operativos e *backends* gráficos. O Matplotlib suporta dezenas de *backends* e tipos de saída, o que significa que é confiável para funcionar independentemente do sistema operativo que estiver a utilizar ou do formato de saída que desejar.

Esta abordagem multiplataforma, tudo-em-um tem sido um dos grandes pontos fortes da Matplotlib. Levou a uma grande base de utilizadores, o que por sua vez levou a uma base de desenvolvedores ativa e as poderosas ferramentas e ubiquidade da Matplotlib dentro do mundo científico Python.[22]

## 2.3 Modelos de Supervised Learning : Visão Geral

A classificação de crédito é um problema especificamente de classificação binária, em que o objetivo é classificar bons e maus empréstimos. Nesta secção definimos primeiro o problema que o *supervised learning* tenta resolver para depois apresentar os métodos mais frequentemente utilizados na classificação de créditos.

Os problemas de modelação preditiva de classificação são diferentes dos problemas de modelação preditiva de regressão, pois a classificação é a tarefa de prever um *label* de uma classe e a regressão é a tarefa de prever uma quantidade contínua. No entanto, ambos partilham o mesmo conceito de utilizar variáveis conhecidas para fazer previsões, logo existe uma sobreposição significativa entre os dois modelos. Assim na figura 5 irão ser representados os modelos de classificação e de regressão de forma a distingui-los.[1]

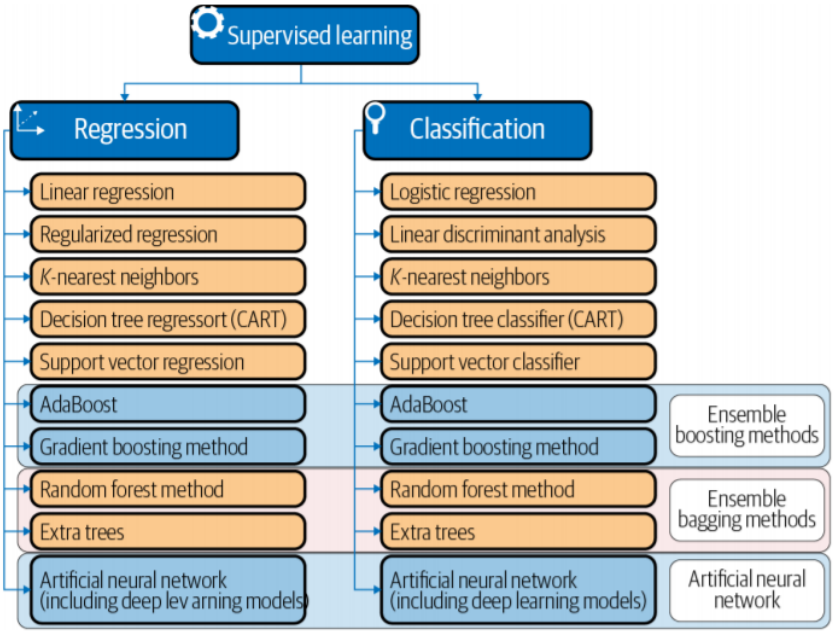



Fig. 5: Modelos de Regressão e Classificação - fonte:[1]

O problema de *supervised learning* encontra-se quando temos variáveis de entrada ($X$), também conhecidas como *features*, e variáveis de saída ($Y$), também conhecidas como *labels* ou *targets*, e queremos que um algoritmo aprenda a função de mapeamento desde a variável de entrada até à de saída.

$$Y = f(X) \tag{1}$$

O objetivo é conseguir aproximar tão bem esta função de mapeamento que quando se tem novos dados de entrada ($X$), seja possível prever as variáveis de saída ($Y$) para esses dados.

Chama-se *supervised learning* porque o processo de aprendizagem de um algoritmo a partir do conjunto de dados de treino pode ser pensado como um professor que supervisiona o processo de aprendizagem. Conhece-se as respostas corretas, o algoritmo faz iterativamente previsões sobre os dados de formação e é corrigido pelo professor. A aprendizagem para quando o algoritmo atinge um nível aceitável de desempenho[23].

Tendo em conta a tabela 1 da secção 2.1 a tese reuniu os modelos mais comummente utilizados pelos estudos abordados, agrupando-os por classificadores individuais e classificadores de *ensemble*.

### 2.3.1 Classificadores individuais

Os classificadores individuais aplicam um único modelo de classificação, representam o tipo de classificadores mais comuns na área de avaliação de crédito financeiro, sendo o uso de LR um padrão da industria de modelação de risco de crédito[8]. Devido a este tipo de classificação ser usado como padrão irá ser usado como um *benchmark* de comparação, tais comparações serão realizadas no âmbito de comprovar se realmente algoritmos de classificação individuais demonstram melhores resultados que todos os outros[24].

#### 2.3.1.1 Linear discriminant analysis

*Linear discriminant analysis* (LDA) é um modelo de *supervised learning* que encontra uma combinação linear de características que caracteriza ou separa duas ou mais classes de objetos ou eventos. Este método projeta um conjunto de dados num espaço de menor dimensão com uma boa separabilidade de classes, ao contrário de métodos como *Principal Component Analysis* (PCA), para evitar a sobreposição e por sua vez reduzir os custos computacionais. A combinação resultante pode ser utilizada como classificador linear ou, mais comummente, para a redução da dimensionalidade antes da classificação subsequente.

O objetivo de uma LDA em muitas situações é projetar um espaço de características (um *dataset* de n dimensões) num sub espaço mais pequeno, mantendo ao mesmo tempo a informação não discriminatória da classe. Em geral, a redução da dimensionalidade não só ajuda a reduzir os custos computacionais para uma dada tarefa de classificação, mas também pode ser útil para evitar o *overfitting*, minimizando o erro na estimativa dos parâmetros.[1]

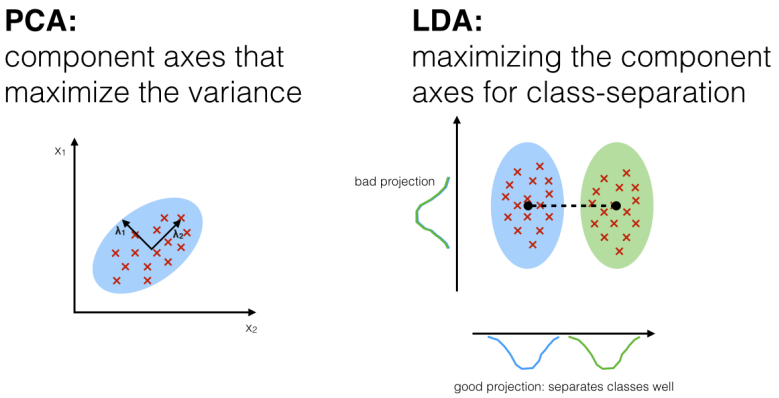



Fig. 6: PCA VS LDA fonte:[1]

#### 2.3.1.2 Logistic regression

O modelo de Logistic regression(LR) surge do desejo de modelar as probabilidades das classes de saída dada uma função que é linear em x, ao mesmo tempo que assegura que a saída permanece entre os valores zero e um, como seria de esperar das probabilidades.

Se treinarmos um modelo de regressão linear em vários exemplos em que Y = 0 ou 1, podemos acabar por prever algumas probabilidades que são inferiores a zero ou superiores a um, o que não faz sentido. Em vez disso, o modelo de *logistic regression*, que é uma modificação da *linear regression* assegura uma probabilidade entre zero e um, aplicando a função sigmóide.

$$y = \frac{exp(\beta_0 + \beta_{1x1} + ... + \beta_{ix1})}{1 + exp(\beta_0 + \beta_{1x1} + ... + \beta_{ix1})} \tag{2}$$

A equação 2 mostra um modelo de *Logistic regression*. Semelhante à regressão linear, os valores de entrada ($x$) são combinados linearmente usando pesos ou valores de coeficiente para prever um valor de saída ($y$). O output proveniente da equação 2 é uma probabilidade que se transforma num valor binário(0 ou 1) para obter a previsão do modelo. Onde $y$ é a saída prevista, $\beta_0$ é o bias e $\beta_1$ é o coeficiente para cada valor de entrada ($x$). Cada coluna nos dados de entrada tem coeficiente associado $\beta$ (um valor real constante) que deve ser aprendido usando os dados de treino.[1]

#### 2.3.1.3 k-Nearest neighbor

O *k-nearest neighbors* (KNN) é considerado um "lazy learner", uma vez que não há aprendizagem exigida no modelo. Para um novo ponto de dados, as previsões são feitas através de pesquisa através de todo o conjunto de treino para os k casos mais semelhantes (os vizinhos) e resumindo a variável de saída para aquelas instâncias K.

Para determinar quais as instâncias de k num conjunto de dados de treino que são mais semelhantes a uma nova entrada, é utilizada uma medida de distância. A medida de distância mais popular é a distancia euclidiana [2]. Quando é introduzida no modelo uma nova entrada é feito a sua distância em comparação com os seus vizinhos e é realizada uma votação para atribuir em que categoria se irá encaixar melhor.[1]

#### 2.3.1.4 Decision tree

*Decision tree* (DT) ou árvore de decisão é definida como um gráfico acíclico utilizado para tomar decisões. Em cada nodo do gráfico o exemplo vai sendo testado segundo uma feature especifica e esta será avaliada segundo um critério. Se o valor passar nesse critério, essa feature segue para um nodo seguinte. Se não passar, segue para outro nodo e assim sucessivamente, até termos a classe a que o exemplo está associado.[9]

A figura 7 mostra um exemplo de uma simples árvore de decisão para prever se uma pessoa é homem ou mulher com base em duas entradas de altura (em centímetros) e peso (em quilogramas).

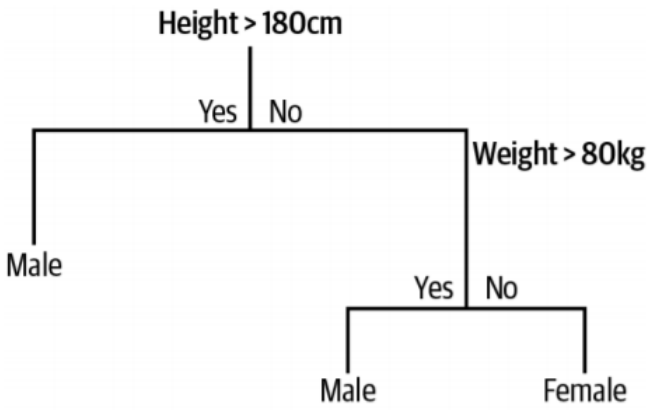



Fig. 7: Decision tree - fonte:[1]

O objetivo deste modelo é conseguir prever a que classe é que o nosso valor de entrada mais se identifica. Este é um modelo não-paramétrico, visto que ao longo da DT o valor de entrada vai sofrendo avaliações e vai sendo testado em cada nodo pelas condições definidas.

#### 2.3.1.5 Support vector machines

O objetivo do algoritmo *Support Vector Machine* (SVM) é maximizar a margem (mostrada como área sombreada na Figura 8), que é definida como a distância entre o hiperplano de separação (ou

[2] $\sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \cdots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2}$.

limite de decisão) e as amostras de treino que estão mais próximas deste hiperplano, os chamados vetores de apoio. A margem é calculada como a distância perpendicular desde a linha até apenas aos pontos mais próximos, como mostrado na Figura 8. Assim, a SVM calcula um limite de margem máxima que leva a uma partição homogénea de todos os *data points*.[1]

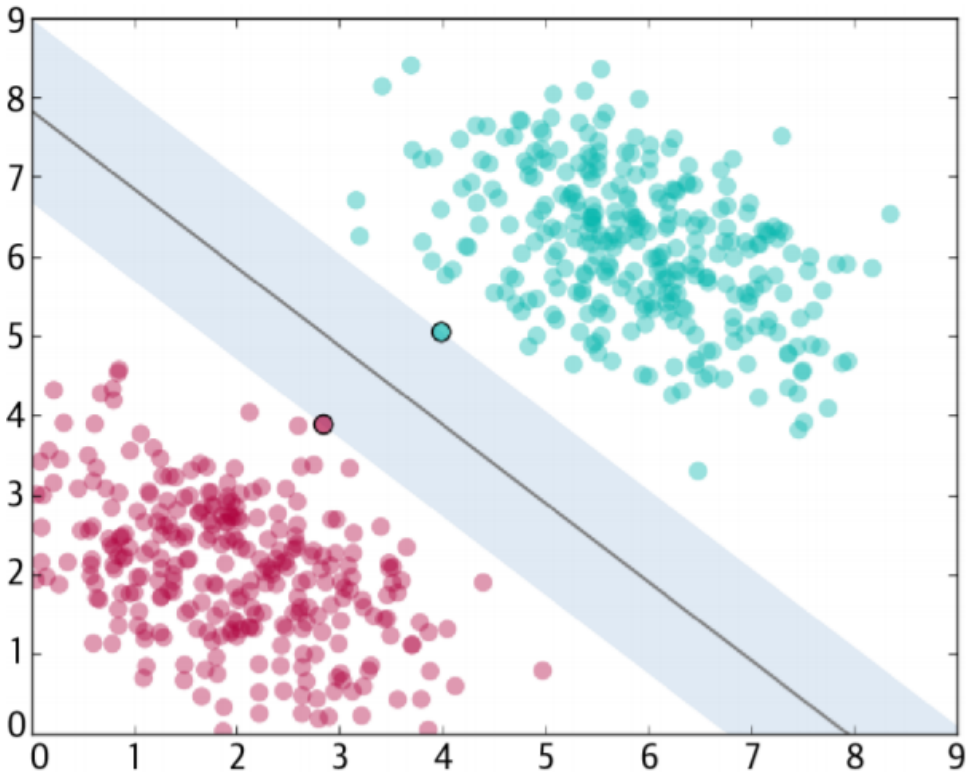



Fig. 8: Support Vector Machine - fonte: [1]

Nem sempre é possível encontrar um hiperplano ou um limite de decisão, sendo nestes casos utilizados *kernels*. Um *kernel* é apenas uma transformação dos dados de entrada que permite ao algoritmo SVM tratar/processar os dados mais facilmente. Usando *kernels*, os dados originais são projetados numa dimensão superior para melhorar a classificação dos dados.

#### 2.3.1.6 Artificial neural networks(ANNs)

Como já observamos, existem muitos tipos diferentes de modelos utilizados em ML. No entanto, uma classe de modelos de ML que se destaca são as *artificial neural networks*. Dado que são utilizadas em todos os tipos de ML, esta secção abrangerá os princípios básicos das ANNs.

As ANNs são sistemas de computação baseados numa coleção de unidades conectadas ou nós chamados de neurónios artificiais, que modelam vagamente os neurónios de um cérebro biológico.

Cada ligação, como as sinapses[3] num cérebro biológico, pode transmitir um sinal a partir de um neurónio artificial para outro.

*Artificial neural networks* são redutíveis a um modelo de classificação ou regressão com a função de ativação do nó na camada de saída. No caso de um problema de regressão, o nó de saída tem uma função de ativação linear(ou não tem função de ativação). Uma função linear produz uma saída continua que vai de menos infinito a mais infinito. Assim, a camada de saída será a função linear dos nós da camada anterior, e será um modelo baseado em regressão.

No caso de um problema de classificação, é recomendado que o nó de saída tenha uma função de ativação sigmóide ou softmax. Estas funções produzem um output que varia de zero a um, representando a probabilidade do valor alvo. A função softmax também pode ser utilizada para múltiplos grupos para classificação.

### 2.3.2 Classificadores Ensemble

O objetivo dos modelos de *ensemble* é combinar diferentes classificadores num meta classificador que tenha melhor desempenho de generalização do que cada classificador individual. Por exemplo, assumindo que recolhemos previsões de 10 modelos, os métodos de *ensemble* permitir-nos-iam combinar estrategicamente as suas previsões para chegar a uma previsão mais precisa e robusta do que as previsões individuais de um só modelo.

Os dois métodos de *ensemble* mais populares são *bagging* e *boosting*. *Bagging* (ou *boot-strap aggregation*) é uma técnica de treino que junta vários modelos individuais numa forma paralela. Cada modelo é treinado por um subconjunto aleatório dos dados. *Boosting*, por outro lado, é uma técnica de *ensemble* que treina vários modelos individuais numa forma sequencial. Isto é feito construindo um modelo a partir dos dados de treino e depois criando um segundo modelo que tenta corrigir os erros do primeiro. Os modelos são adicionados até que o conjunto de treino seja perfeitamente previsto ou até que um número máximo de modelos seja adicionado. Cada modelo individual aprende com os erros cometidos pelo modelo anterior.[1]

Os classificadores de *ensemble* integram a previsão de múltiplos modelos, estes classificadores podem ser considerados homogéneos ou heterogéneos. Distinguimos *ensemble* homogéneos, visto que criam modelos de base usando o mesmo algoritmo enquanto que *ensemble* heterogéneos empregam modelos bases diferentes.[24]

#### 2.3.2.1 Ensemble homogéneos

Os classificadores de *ensemble* homogéneos reúnem as previsões de múltiplos modelos base. Artigos abordados na secção 2.1 demonstram que a combinação de modelos aumenta a precisão exata (por

[3]Região de contacto entre dois neurónios.

exemplo, Xia et al. [3] & Dastile et al [4]). Modelos homogéneos *ensemble* criam os modelos base de forma independente ou dependente. Por exemplo, o algoritmo de *bagging* deriva de modelos de base independentes de amostras de *bootstrap* dos dados originais. Os algoritmos de *boosting*, por outro lado, é um *ensemble* dependente. Adicionam iterativamente modelos de base que são treinados para evitar os erros do *ensemble* atual. O denominador comum dos *ensemble* homogéneos é a utilização de modelos base como único algoritmo de classificação.[24]

Os classificadores de *ensemble* homogéneos que iram ser utilizados ao longo desta tese são extensões dos métodos *bagging* e de *boosting*.

– **Bootstrap Aggregation (Bagging)** é um algoritmo de *ensemble* de *machine learning* que combina as previsões de varias árvores de decisão. Especificamente, é um *ensemble* de modelos de árvores de decisão, embora a técnica de *bagging* também possa ser utilizada para combinar as previsões de outros tipos de modelos.

  No *bagging* é criado um *ensemble* fazendo várias amostras diferentes do mesmo conjunto de dados de treino e atribuindo uma árvore de decisão em cada uma delas. Dado que cada amostra do conjunto de dados do treino é diferente, cada árvore de decisão é diferente, fazendo por sua vez previsões e erros de previsão ligeiramente diferentes. As previsões para todas as árvores de decisão criadas são combinadas, resultando num erro inferior ao de uma só árvore.[25]

  *Bagging* fornece a base para todo um conjunto de algoritmos de *ensemble* utilizados nesta tese, tais como os algoritmos **BagDT**, **BagNN** e **RF**.

  - **Random forest**(RF) é uma versão ajustada de *decision trees* usando *bagging*. Este algoritmo é construído através de diversas árvores de decisão(DT) e o resultado que o modelo fornece é a classe que ocorre com maior frequência, ou seja, é a moda entre as classes. Um problema com árvores de decisão é o facto de serem gananciosas. É escolhido a variável a dividir usando um algoritmo ganancioso que minimiza o erro. Mesmo após o *bagging*, as árvores de decisão podem ter muitas semelhanças estruturais e resultar numa alta correlação nas previsões. Combinar previsões de vários modelos em conjuntos funciona melhor se as previsões dos submodelos não estiverem correlacionadas ou, na melhor das hipóteses, forem fracamente correlacionadas[1].

– **Boosting** é uma poderosa técnica de *ensemble* de *machine learning*. O *Boosting* pode ser entendido contrastando-o com o *bagging*. *Boosting* funciona de uma forma semelhante, onde múltiplas árvores são encaixadas em diferentes amostras de um conjunto de dados de treino e as previsões das árvores são combinadas utilizando uma votação simples para a classificação ou a média para a regressão para resultar numa melhor previsão que uma única árvore de decisão.[26]

Fornece a base para todo um conjunto de algoritmos de *ensemble* utilizados nesta tese, tais como os algoritmos **AdaBoost**, **GBDT**, **XGBoost**, **LightGBM** e **CatBoost**.

- **Adaptive Boosting ou AdaBoost** é uma técnica de *boosting* na qual a ideia básica é tentar prever sequencialmente, e cada modelo subsequente tenta corrigir os erros do seu predecessor. Em cada iteração, o algoritmo *AdaBoost* altera a distribuição da amostra modificando os pesos ligados a cada uma das instâncias. Este algoritmo aumenta os pesos das instâncias previstas erradamente e diminui os das instâncias previstas corretamente[1].

- **Gradient-boosted decision trees** (GBDT) são uma técnica de supervised learning para otimizar o valor preditivo de um modelo por meio de etapas sucessivas no processo de aprendizagem. Cada iteração da árvore de decisão envolve o ajuste dos valores dos coeficientes, pesos ou bias aplicados a cada uma das variáveis de entrada usadas para prever o valor alvo, com o objetivo de minimizar a *loss function*[27].

  As GBDT são um método popular para resolver problemas de previsão nos domínios de classificação e regressão. A abordagem melhora o processo de aprendizado simplificando o objetivo e reduzindo o número de iterações para chegar a uma solução suficientemente ótima. Modelos Gradient-boosted provaram-se repetidamente em várias competições classificando em precisão e eficiência, tornando-os um componente fundamental no kit de ferramentas de *data science*.

- **eXtreme Gradient Boosting** (XGBoost), é uma biblioteca de *machine learning* escalável e distribuída de *Gradient-boosted decision trees (GBDT)*. Fornece um impulso ao *parallel tree boosting* e é a principal biblioteca de *machine learning* para problemas de regressão e classificação. *XGBoost* ganhou um reconhecimento significativo nos últimos anos com os seus resultado em competições na plataforma Kaggle[28].

  O *XGBoost* foi integrado com uma grande variedade de ferramentas e bibliotecas, tais como scikit-learn para entusiastas de Python e caret para utilizadores de R. Além disso, o *XGBoost* está integrado com estruturas de processamento distribuído, como Apache Spark e Dask.

- **Light Gradient Boosting Machine** (LightGBM) é uma estrutura de *gradient boosting* que utiliza algoritmo de aprendizagem baseado em árvores. *LightGBM* cresce na vertical enquanto outros algoritmos crescem na horizontal, o que significa que o *LightGBM* cresce em função das folhas das árvores enquanto outros algoritmos cresce em função do nível[29].

  Foi concebido para ser distribuído e eficiente com as seguintes vantagens: Velocidade de treino mais rápida e maior eficiência; Menor utilização de memória; Melhor precisão; Apoio à aprendizagem paralela, distribuída e GPU; Capaz de tratar dados em grande escala[30].

- **CatBoost** é um algoritmo de *machine learning* que usa *gradient boosting* em árvores de decisão de código aberto, seu nome deriva de "*category*" e "*boosting*". O *CatBoost* constrói árvores simétricas (equilibradas), ao contrário do *XGBoost* e do *LightGBM*. Em cada passo, as folhas da árvore anterior são divididas usando a mesma condição. O par de características que contabiliza a menor perda é selecionado e utilizado para todos os nós do nível. Esta arquitetura de árvore equilibrada ajuda na implementação eficiente da CPU, diminui o tempo de previsão e controla o *overfitting* à medida que a estrutura serve de regularização[31].

#### 2.3.2.2 Ensemble heterogéneos

Os classificadores de *ensemble* heterogéneos também combinam múltiplos modelos de classificação, mas criam estes modelos usando diferentes algoritmos de classificação. Abrangendo assim classificadores individuais e classificadores *ensemble* homogéneos. A ideia é que diferentes algoritmos têm "visões" diferentes sobres os mesmo dados e podem complementar-se mutuamente. Recentemente, este tipo de classificação tem atraído muito interesse sendo motivo de investigação em múltiplos artigos, por essa razão a tese irá prestar especial atenção a este tipo de classificação.[24]

Para aplicar classificadores *ensemble* heterogéneos a tese selecionou os seguintes técnicas de *machine learning*:

– **Staking Ensembles (ou Stacked Generalization)** é um modelo *ensemble* de *machine learning*. Utiliza um algoritmo de *meta-learning* que aprende qual a melhor forma de combinar as previsões de dois ou mais algoritmos base.

  A vantagem do *stacking* é o facto de ser possível aproveitar as capacidades de uma gama de modelos com bom desempenho numa tarefa de classificação e fazer previsões com melhor desempenho do que qualquer modelo único no *ensemble*.

  A arquitetura de um modelo de *stacking* envolve que dois ou mais modelos base, frequentemente referidos como modelos de nível-0, e um meta-modelo que combina as previsões dos modelos base, referido como modelo de nível-1.

  - Modelos de nível-0 (Modelos Base): O conjunto de dados de treino são encaixados nos modelos cujas previsões são compiladas.
  - Modelos de nível-1 (Meta-modelo): Modelo que aprende a melhor forma de combinar as previsões dos modelos base.

  Os modelos de nível-0 são frequentemente complexos e diversificados. Como tal, é muitas vezes uma boa ideia utilizar uma gama de modelos que fazem pressupostos muito diferentes sobre como resolver a tarefa de modelação preditiva, tais como modelos lineares, árvores de

decisão, SVM, Redes neuronais, entre outros. Outros algoritmos de *ensemble* também podem ser utilizados como modelos base, dando o exemplo de RF.

O meta-modelo é frequentemente simples, fornecendo uma interpretação suave das previsões feitas pelos modelos base. Como tal, os modelos lineares são frequentemente utilizados como meta-modelo.

- **Voting Ensembles (ou "Majority voting ensembles")** é um modelo de *ensemble* que combina as previsões de vários outros modelos, seja modelos compostos por classificadores individuais ou modelos *ensemble* homogéneos. É uma técnica que pode ser utilizada para melhorar o desempenho de um modelo, idealmente conseguindo um melhor desempenho do que qualquer classificador individual[32].

  Este tipo de *ensemble* é considerado um meta-modelo, um modelo de modelos. Como tal, pode ser utilizado com qualquer coleção de modelos de *machine learning* já existentes e os modelos existentes não precisam de estar cientes de que estão a ser utilizados num conjunto[32].

  Há duas abordagens para *majority voting ensemble* para a classificação; *hard voting* e o *soft voting*. Para esta tese será empregue apenas o *hard voting* visto que é mais apropriado quando os modelos utilizados no *voting ensemble* preveem etiquetas de classe[32].

  Uma limitação da técnica do *voting ensemble* é que pressupõe que todos os modelos do conjunto são igualmente eficazes. Isto pode não ser o caso porque alguns modelos podem ser melhores do que outros, especialmente se forem utilizados diferentes algoritmos de ML para treinar cada modelo do *ensemble*[33].

- **Weighted average Ensembles** é uma abordagem que combina as previsões de múltiplos modelos, em que a contribuição de cada modelo é ponderada proporcionalmente a sua capacidade ou habilidade[33].

  Ao contrario do *voting ensemble* este modelo assume que os modelos pertencentes a um *ensemble* não são todos igualmente capazes e, em vez disso, alguns modelos são melhores do que outros e devem receber mais votos ou mais peso quando se faz uma previsão.

  Uma previsão feita pelo modelo *weighted average* envolve primeiro a atribuição de um coeficiente de peso fixo para cada membro/algoritmo do *ensemble*. Tais pesos encontram-se entre os valores de 0 e 1, representando uma percentagem do peso[33].

## 2.4 Métricas de Avaliação: Visão Geral

As métricas utilizadas para avaliar os algoritmos de ML são muito importantes, a escolha das métricas de avaliação a utilizar influencia a forma como o desempenho dos algoritmos de ML

é medido e comparado. A métrica influencia tanto a forma como se pondera a importância das diferentes características nos resultados como qual o algoritmo que melhor se adapta a dada situação[1].

Nesta secção serão abordadas as métricas de avaliação mais utilizadas para modelos de classificação. A importância das métricas foi baseada na sua utilização nos artigos investigados, que podem ser observados na tabela 1 na secção da revisão da literatura.

### 2.4.1 Matriz de Confusão

Existem parâmetros básicos que precisamos de considerar quando se trata do desempenho dos modelos de classificação. Estes parâmetros são melhor descritos e definidos através da Matriz de Confusão. Matriz de Confusão ou Matriz de Erro é um dos conceitos-chave quando falamos de problemas de classificação. Esta matriz é uma representação tabular das previsões do modelo vs valores reais[34].

A Matriz de Confusão é simplesmente uma matriz quadrada que relata as contagens do verdadeiro positivo (TP), verdadeiro negativo (TN), falso positivo (FP), e falso negativo (FN ) das previsões de um classificador, como mostrado na Figura 9.

As métricas(Exatidão, Precisão e Sensibilidade) são estimadas com base nos valores tirados da matriz de confusão(TP, TN, FP e FN). Onde, TP e TN denotam a classe que obteve a classificação correta e FP e FN representam a classe que obteve os resultados de classificação incorretos. Na figura 9 são ilustradas estas mesmas métricas.

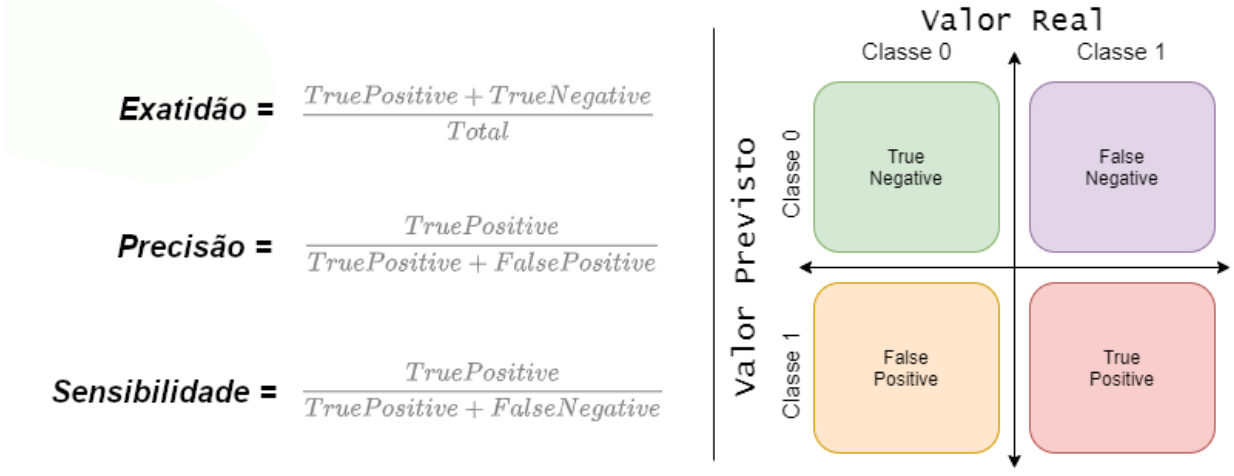



Fig. 9: Exatidão, Precisão, Sensibilidade, Matriz de Confusão

### 2.4.2 Exatidão

Conforme mostrado na Figura 9, a Exatidão é o número de previsões corretas feitas a dividir por todas as previsões feitas. Esta é a métrica de avaliação mais comum para problemas de classificação. É mais adequado quando há um número igual de observações em cada classe, o que raramente é o caso.

Dito isto, Exatidão é uma métrica complicada porque pode dar impressões erradas sobre o modelo. Especialmente em situações em que o *dataset* é desequilibrado. Por exemplo, se existir uma divisão de 99/1 entre duas classes, A e B, onde o raro evento B, é a classe positiva, poderíamos construir um modelo que fosse 99% exato, dizendo que tudo pertencia à classe A. Obviamente que a construção de um tal modelo que ignore a identificação da classe B não é fiável. Assim , é necessário métricas diferentes que desencorajem este tipo de comportamento[34].

### 2.4.3 Precisão

A Precisão é a percentagem de instâncias positivas em relação ao total de casos positivos previstos. Aqui, o denominador é composto por todos os verdadeiros positivos. A precisão é uma boa medida para determinar quando o custo de falsos positivos é alto.

A Precisão é uma métrica muito útil e que acarreta mais informação do que a Exatidão. Essencialmente, com Precisão, respondemos à pergunta: "Que proporção de identificações positivas foi correta?"[34].

### 2.4.4 Sensibilidade

Sensibilidade (ou *Recall*) é a percentagem de instâncias positivas do total de instâncias positivas reais. Portanto, o denominador é o numero real de instâncias positivas presentes no conjunto de dados. Sensibilidade é uma boa medida quando há um alto custo associado a falsos negativos[1].

Pode ser descrita como a capacidade do classificador de encontrar todas as amostras positivas. Com esta métrica, estamos a tentar responder à pergunta: "Que proporção de positivos reais foi identificada corretamente?"[34].

### 2.4.5 Brier Score

O Brier Score (BS) foi introduzida por Brier em 1950 para abordar a questão da verificação das previsões do tempo e, desde então, foi adotada fora do campo da meteorologia como uma métrica de avaliação simples para previsões de resultados binários[35]. O BS para uma amostra de n previsões binárias, onde temos um evento que poderá ou não acontecer, é representado na seguinte equação:

$$BS = \frac{1}{n}\sum_{t=1}^{n}(f_t - o_t)^2 \tag{3}$$

Onde $f_t$ é a probabilidade prevista de o evento ocorrer de acordo com a previsão $t^{th}$ e o $o_t$ é igual a 1 ou 0, dependendo se o evento ocorreu subsequentemente ou não. Conforme definido na equação, a pontuação é orientada negativamente, o que significa que valores menores da pontuação indicam melhores previsões.[36]

### 2.4.6 Curva ROC

Uma ferramenta útil para prever a probabilidade de um resultado binário é a curva *Receiver Operating Characteristic* (ROC). É um gráfico da taxa de falso positivo (eixo x) versus a taxa de verdadeiro positivo (eixo y) para vários valores de limiar candidatos diferentes entre 0,0 e 1,0. Dito de outra forma, traça a taxa de falsos alarmes em relação à taxa de sucesso. A forma da curva contém muitas informações, incluindo o que mais interessa para conseguir avaliar um problema, a taxa de falsos positivos esperada e a taxa de falsos negativos. Na figura 10 esta representado um exemplo de um gráfico de curva ROC.

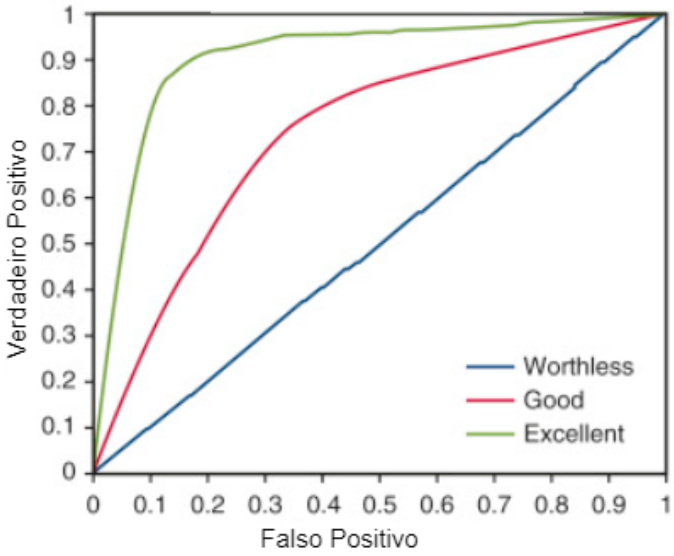



Fig. 10: Curva ROC

### 2.4.7 AUC

*Area under ROC curve* (AUC) representa o grau ou medida de separabilidade. Diz o quanto o modelo é capaz de distinguir entre as classes. Quanto maior a AUC, melhor será o modelo em prever

zeros como zeros e uns como uns. Um AUC de 0,5 significa que o modelo não tem capacidade de separação de classes.

A interpretação probabilística do resultado AUC é que se for escolhido aleatoriamente um caso positivo e um caso negativo, a probabilidade de que o caso positivo supere o caso negativo de acordo com o classificador é dada pelo AUC.

## 2.5 Resumo dos trabalhos relacionados

Após a análise realizada na secção de revisão de literatura é necessário esclarecer três pontos fulcrais que são: Quais os *datasets* que melhor se adequam a esta tese, tendo em conta a informação reunida por vários estudos; Os modelos que serão escolhidos para a fase experimental da tese, tendo como referência os modelos dos estudos abordados; As métricas que melhor monitorizam o desempenho de cada modelo.

Em relação aos *datasets* que serão utilizados, a tese baseando-se na figura 1 consegue afirmar a existência de uma grande utilização dos *datasets* da UCI [13], mais concretamente do *dataset* alemão e o australiano. Estes dois *datasets* de crédito irão ser utilizados como forma de treinar os modelos/algoritmos. Devido que estes *datasets* serem comummente usados torna-os fiáveis como meio de comparação com outros trabalhos relacionados, tornando também os resultados da tese comparáveis com outros resultados.

Tendo como base os estudos analisados na revisão de literatura, a tese dará especial importância a modelos *ensemble*, concretamente, a modelos de *ensemble* homogéneos e heterogéneos já que foram estes os que obtiveram, em geral, os melhores resultados nos artigos abordados.

A maioria dos artigos foram selecionados por terem sido publicados recentemente, sendo todos os resultados discutidos fiáveis e atuais. Dito isto, modelos *ensemble* de classificação homogéneos e heterogéneos serão o foco principal que a tese irá abordar, onde irão ser testados vários modelos de classificação de *ensemble* e comparados com os resultados dos modelos de classificação individuais como forma de *benchmark*. Sendo o objetivo determinar quais os modelos atuais mais propícios para avaliação de riscos em crédito bancário e se realmente segundo os artigos abordados se modelos de *ensemble* são os mais adequados.

Os modelos que serão utilizados como *benchmark* serão compostos por classificadores individuais (p. ex. LR, SVM, ANN, DT, KNN), devido a sua presença em *benchmark* de outros estudos abordados.

A escolha das métricas irá ser baseadas naquelas que são mais comummente utilizadas, assim os resultados obtidos pelos modelos / algoritmos desta tese serão mais fiáveis quando comparadas com outros trabalhos desta área. Observando a figura 3 é de salientar que a métrica AUC é a

de longe a mais usada para este tipo de investigação, o que fará com que seja uma da principais métricas desta tese.

Outras métricas que também serão utilizadas será a ACC (Exatidão) e o Brier Score para além de serem métricas com muita utilização também são métricas que dão um *feedback* importante acerca do desempenho do modelo.

# 3 Metodologia

Esta secção aborda as etapas necessárias para a realização da parte experimental da tese, tendo como ponto fulcral revelar quais os métodos/algoritmos mais atuais e precisos para a classificação de crédito bancário. Numa fase inicial será feito uma recolha dos *datasets* que irão ser utilizados para treinar tais modelos, onde as variáveis de cada *dataset* serão descritas. Numa fase posterior será feito um planeamento de como é que a tese ira realizar tais experiências.

## 3.1 Recolha dos Dados

Os projetos de modelação preditiva envolvem a aprendizagem a partir de dados. Os dados referem-se a exemplos ou casos do domínio que caracterizam o problema que se pretende resolver. Na aprendizagem supervisionada, os dados são compostos de exemplos em que cada exemplo tem um elemento de entrada que será fornecido a um modelo e um elemento de saída ou alvo que se espera que o modelo preveja. A classificação é um exemplo de um problema de aprendizagem supervisionada em que o alvo é um rótulo[37].

Os dados introduzidos podem ter muitas formas, tais como imagem, série cronológica, texto, vídeo, e assim por diante. Neste caso os dados obtidos são dados normalmente referido como dados tabulares ou dados estruturados. Trata-se de dados como os que podemos observar em folhas de cálculo, numa base de dados, ou num ficheiro de variáveis separadas por vírgulas (CSV)[37].

Em álgebra linear, tabelas grandes de dados são referidas como uma matriz. Para os efeitos desta tese as tabelas serão compostas por filas e colunas. Onde uma linha representa um exemplo do domínio do problema, e pode ser referida como um "exemplo" ou uma "instância". Uma coluna representa as propriedades observadas sobre o exemplo e pode ser referida como uma "variável", um "atributo" ou *feature*.

- **Fila** - Um único exemplo do domínio, muitas vezes chamado de instância ou exemplo em ML.
- **Coluna** - Uma única propriedade registada para cada exemplo, muitas vezes chamada de variável ou característica em ML.

As colunas utilizadas para a entrada do modelo são referidas como variáveis de entrada, e a coluna que contém o alvo a prever é referida como a variável de saída ou *target*. As linhas utilizadas para treinar um modelo são referidas como o conjunto de dados(*dataset*) treino e as linhas utilizadas para avaliar o modelo são referidas como o conjunto de dados(*dataset*) de teste.

- **Variáveis de entrada** - Colunas no *dataset* fornecido a um modelo para fazer uma previsão.
- **Variáveis de saída** - Coluna no *dataset* a ser previsto por um modelo.

No caso dos dados que serão utilizados encontram-se guardados em formato de Comma-separated values(CSV). Esta é uma representação padrão em ML que é portátil, bem compreendida, e pronta para o processo de modelação sem dependências externas.

Foram empregues 2 *datasets* de crédito de dados reais de forma a obter dados necessários para o treino de modelos. Estes são o *dataset* Australiano e o Alemão provenientes do *UCI Machine Learning repository* [13]. Como já visto na secção 2.1 estes *datasets* são comummente usados em literaturas relacionadas, o que garante a viabilidade de realização de comparações detalhadas com outros estudos.

Ambos os *datasets* apenas sendo recolhidos do domínio ainda se encontram no seu estado bruto, com possíveis valores nulos, indesejados ou com variáveis que não trazem qualquer beneficio para a aprendizagem do modelo. Será necessário a realização de uma analise e de um pré-processamento dos mesmo para os deixar aptos para o seu uso em *machine Learning*.

Uma visão geral destes *datasets* é apresentado na Tabela 2, onde é possível observar o numero de instâncias, o numero de *features* (variáveis), o rácio de bom ou mau empréstimo e o domínio de cada *dataset*.

Tabela 2: Resumo dos Datasets de crédito

| Nome | Abreviatura | Nº Instancias | Nº Features | Bom/Mau | Fonte |
|---|---|---|---|---|---|
| Dataset de credito aprovado Australiano | Dataset Australiano | 690 | 14 | 307/383 | [13] |
| Dataset de crédito Alemão | Dataset Alemão | 1000 | 20 | 700/300 | [13] |

### 3.1.1 Dataset Alemão

O *dataset* Alemão é constituído por 1000 instâncias, sendo 700 dessas instâncias etiquetadas como sendo um bom empréstimo e as restantes 300 etiquetadas como sendo um mau empréstimo. Cada instância deste *dataset* é constituído por 20 *features*, onde 13 dessas variáveis são valores categóricos e 7 valores numéricos.

Na tabela 3 são apresentados todas as variáveis do *dataset* alemão, incluindo o nome de cada variável, o seu tipo (categórico ou numérico) e os seus valores. Isto é possível graças a documentação deixada pelo Dr. Hans Hofmann. [13]

Tabela 3: Features do Dataset Alemão

| | Features | Tipo | Valores |
|---|---|---|---|
| 1 | Situação da conta corrente existente | Categórico | A11: ... < 0 DM<br>A12: 0 <= ... < 200 DM<br>A13: ... >= 200 DM<br>A14: sem conta corrente |
| 2 | Duração do empréstimo | Numérico | |
| 3 | Histórico de crédito | Categórico | A30: Nenhum crédito pedido / todos créditos pagos devidamente<br>A31: Todos os créditos neste banco pagos devidamente<br>A32: Crédito existente devolvido devidamente até agora<br>A33: Demora em pagar no passado<br>A34: Conta critica / outros créditos existentes (em outros bancos) |
| 4 | Propósito do empréstimo | Categórico | A40: Carro (novo)<br>A41: Carro (usado)<br>A42: Mobília / equipamento<br>A43: Radio / Televisão<br>A44: Aparelhos domésticos<br>A45: Reparações<br>A46: Educação<br>A47: Férias/Viagens<br>A48: Requalificação<br>A49: Negócios<br>A410: Outros |
| 5 | Quantidade do Crédito | Numérico | |
| 6 | Conta Poupança / Títulos | Categórico | A61: ... < 100 DM<br>A62: 100 <= ... < 500 DM<br>A63: 500 <= ... < 1000 DM<br>A64: ... >=1000 DM<br>A65: Desconhecido / sem conta poupança |
| 7 | Presente emprego deste | Categórico | A71: Desempregado<br>A72: ... < 1 Anos<br>A73: 1 <= ... < 4 Anos<br>A74: 4 <= ... < 7 Anos<br>A75: ... <= 7 Anos |
| 8 | Taxa de parcelamento em percentagem do rendimento disponível | Numérico | |

DM - Deutsche Mark - Moeda alemã utilizada na altura deste dataset.

| | Features | Tipo | Valores |
|---|---|---|---|
| 9 | Estatuto pessoal e sexo | Categórico | A91: Masculino : Divorciado/ Separado<br>A92: Feminino : Divorciada/ Separada/ Casada<br>A93: Masculino : Solteiro<br>A94: Masculino : Casado/ Viúvo<br>A95: Feminino : Solteira |
| 10 | Outros devedores / fiadores | Categórico | A101: Nenhum<br>A102: Co-requerente<br>A103: Fiador |
| 11 | Residência atual desde | Numérico | |
| 12 | Imóvel | Categórico | A121: Bens Imóveis<br>A122: Se não A121 : Contrato com cooperativa de crédito de habitação/ Seguro de vida<br>A123: Se não A121 e A122 : Carro ou outro, que não esteja no feature 6<br>A124: Desconhecido / Sem Imóvel |
| 13 | Idade | Numérico | |
| 14 | Outros planos de pagamentos | Categórico | A141: Banco<br>A142: Lojas<br>A143: Nenhum |
| 15 | Habitação | Categórico | A151: Aluguer<br>A152: Próprio<br>A153: De graça |
| 16 | Número de créditos existentes neste banco | Numérico | |
| 17 | Trabalho | Categórico | A171: Desempregado / não qualificado - sem contrato<br>A172: Não qualificado - com contrato<br>A173: Funcionário qualificado / efectivo<br>A174: Administração/por conta própria/ funcionário altamente qualificado |
| 18 | Número de pessoas responsáveis por fornecer manutenção para | Numérico | |
| 19 | Telemóvel | Categórico | A191: Nenhum<br>A192: Sim, registado no nome do cliente |
| 20 | Trabalhador estrangeiro | Categórico | A201: Sim<br>A202: Não |

### 3.1.2 Dataset Australiano

Comparando o Dataset Australiano ao Alemão é de notar que é ligeiramente mais pequeno, no entanto é mais equilibrado. Sendo que é formado por 690 instâncias com 307 dessas etiquetadas como sendo bons empréstimos e 383 como sendo maus empréstimos. Cada instância deste dataset possui 14 variáveis, sendo 8 dessas variáveis valores numéricos e os restantes 6 valores categóricos.

Na table 4 será apresentado todas as variáveis do dataset Australiano, incluindo o nome de cada variável, o seu tipo (categórico ou numérico) e os seus valores. Infelizmente uma particularidade desde dataset é o facto de todas as variáveis e valores foram alterados para símbolos sem sentido como forma de proteger a confidencialidade dos dados.

Tabela 4: Features do Dataset Australiano

| | Features | Tipo | Valores |
|---|---|---|---|
| 1 | Confidencial | Categórico | a,b |
| 2 | Confidencial | Numérico | 13.75-80.25 |
| 3 | Confidencial | Numérico | 0-28 |
| 4 | Confidencial | Categórico | g,p,gg |
| 5 | Confidencial | Categórico | ff,d,i,k,j,aa,m,c,w, e, q, r,cc, x |
| 6 | Confidencial | Categórico | v, h, bb, j, n, z, dd, ff, o |
| 7 | Confidencial | Numérico | 0-28.5 |
| 8 | Confidencial | Categórico | t,f |
| 9 | Confidencial | Categórico | t,f |
| 10 | Confidencial | Numérico | 0-67 |
| 11 | Confidencial | Categórico | tf |
| 12 | Confidencial | Categórico | g, p ,s |
| 13 | Confidencial | Numérico | 0-2000 |
| 14 | Confidencial | Numérico | 0-100000 |

## 3.2 Planeamento da modelação

Nesta secção a tese irá planificar as etapas necessárias para a resolução das experiências desejadas, sendo o objetivo final comparar todos os resultados obtidos pelos diversos tipos de classificadores e conseguir concluir quais os modelos mais vantajosos para a classificação de crédito bancário. Foi desenvolvido um fluxograma, baseado na investigação de Xia [3], que irá descrever todos os passos necessários para este processo.

No fluxograma, representado na figura 11, após a recolha dos *datasets* em bruto, é necessário analisa-los e trata-los deforma a conseguir filtrar quais as variáveis mais relevantes para o treino dos modelos, quais as possíveis relações que as variáveis do *dataset* têm entre si e entender de uma forma geral todos os dados dos *datasets* com o intuito de os utilizar da melhor forma possível, obtendo assim um melhor resultado dos modelos de classificação testados.

É uma boa pratica explorar os dados de um *dataset* primeiro, compreendendo quais as relações que existem e assim reunir o máximo de conhecimento possível.

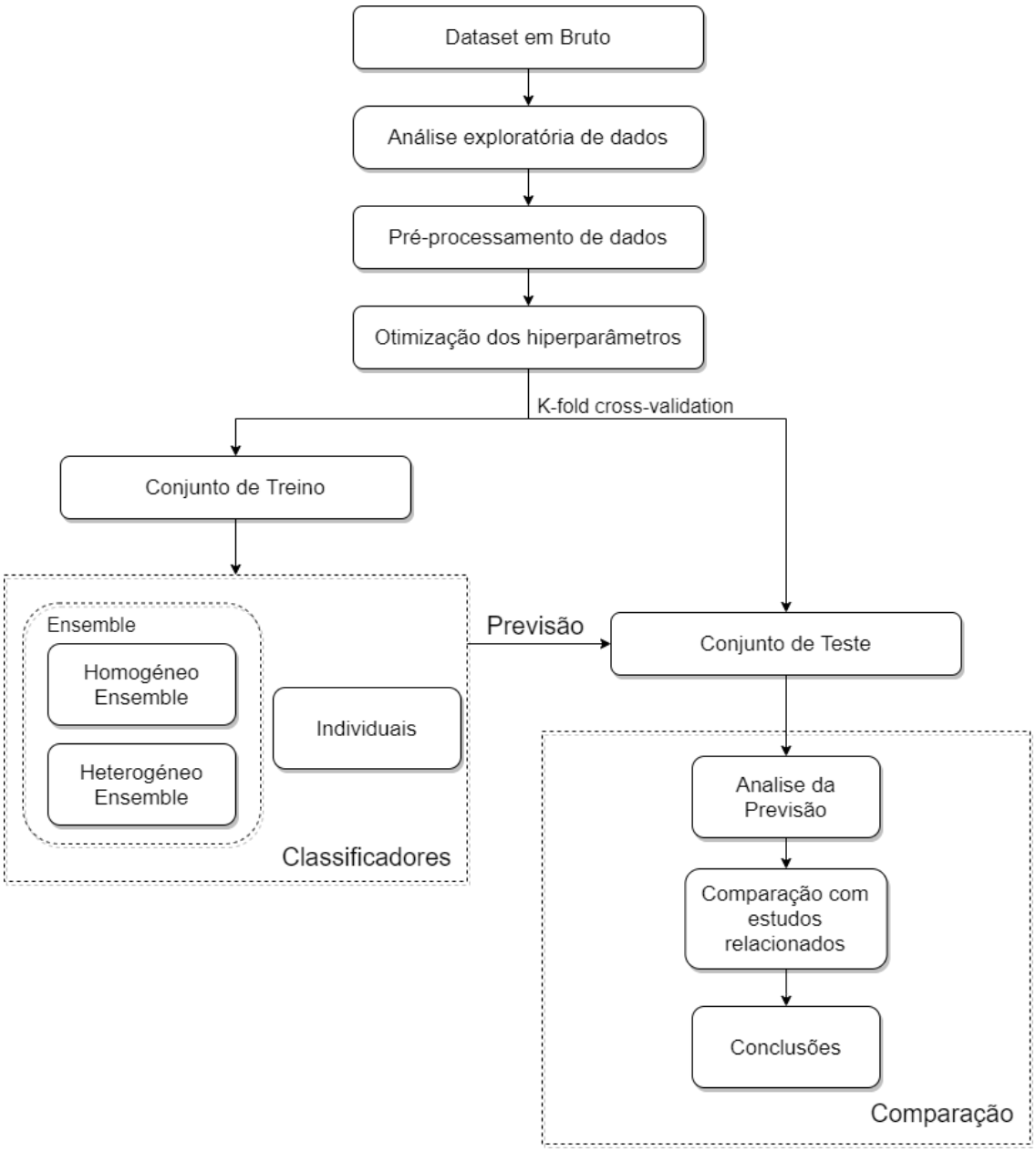



Fig. 11: Fluxograma das etapas do planeamento da modelação usado pela tese

### 3.2.1 Análise exploratória de dados

A análise exploratória de dados, mais conhecido por *Exploratory Data Analysis*(EDA) é utilizada por analistas de dados para analisar e investigar *datasets* e resumir as suas principais caracterísi-

cas, empregando frequentemente métodos de visualização de dados. Determina quais as melhores formas de manipular dados em bruto de forma a obter as respostas necessárias, facilitando aos *data scientists* a descoberta de padrões, detetar anomalias, testar uma hipótese e verificar suposições. O principal objetivo da EDA é ajudar a visualizar os dados antes de fazer quaisquer suposições. Ajuda a identificar erros óbvios, bem como a compreender melhor os padrões dentro dos dados, detetar aberrações ou eventos anómalos e encontrar relações interessantes entre as variáveis[38].

Analistas de dados utilizam a EDA para assegurar que os resultados que produzem são válidos e aplicáveis a quaisquer resultados e objetivos comerciais desejados. A EDA pode ajudar a responder a perguntas sobre desvios padrão, variáveis categóricas, e intervalos de confiança. Uma vez que a EDA esteja completa e os conhecimentos sejam desenhados, as suas características podem então ser utilizadas para uma análise de dados ou modelagem mais sofisticada, incluindo *machine learning*.[38]

Esta analise irá ser realizada apenas para o *dataset* alemão, visto que apresenta uma documentação solida acerca das suas variáveis enquanto que o *dataset* Australiano são confidenciais.

### 3.2.2 Pré-processamento dos dados

Após a Análise exploratória de dados será realizado um pré-processamento dos dados de forma a que estes fiquem de acordo e prontos para o processo de modelação com algoritmos de *machine learning*. No pré-processamento é necessário garantir que todas as variáveis do *dataset* que sejam valores categóricos sejam devidamente tratadas, sendo que algoritmos de *machine learning* apenas recebem valores numéricos como valor de entrada. Para tal, irá ser aplicado a técnica de *one-hot encoding*, que ira transformar todos os dados categóricos em dados numéricos, neste caso em valores zero ou um. Outro problema que é preciso resolver é o facto do *dataset* ser irregular na escala de valores numéricos o que é preciso transformar todos os valores numéricos do *dataset* numa única escala, como por exemplo, todos os valores tem que estar compreendidos entre os valores de zero a um.

### 3.2.3 Otimização de hiperparâmetros

A otimização de hiperparâmetros refere-se à realização de uma pesquisa com o objetivo de descobrir o conjunto de argumentos de configuração de modelo específicos que resultam no melhor desempenho do modelo em um conjunto de dados específico. Os modelos utilizados nesta tese, tanto os modelos de classificação individuais como os modelos de *ensemble* possuem hiperpârametros que necessitam ser afinados para garantir uma melhor performance do modelo.[39]

Existem varias formas de realizar a otimização de hiperparâmetros (p. ex. Grid Search, Random Search), mas a tese ira empregar o método de otimização Bayesiana, visto que é um dos

métodos mais modernos, rápidos e eficazes. Para a implementação desta ferramenta a tese ira usar a biblioteca Python de código aberto Scikit-Optimize que fornece uma implementação de otimização bayesiana que pode ser usada para ajustar os hiperparâmetros de modelos ML da biblioteca scikit-Learn Python.[40]

Otimização bayesiana fornece uma técnica baseada no teorema de Bayes para direcionar a busca de um problema de otimização que seja eficiente e eficaz. Este funciona construindo um modelo probabilístico da função objetivo, chamada de função substituto, que é então pesquisado de forma eficiente com uma função de aquisição antes que as amostras candidatas sejam escolhidas para avaliação da função objetivo real.[40]

É importante notar que esta biblioteca, Scikit-Optimize, fornece apoio para afinar os hiperparâmetros dos algoritmos de *machine learning* oferecidos pela biblioteca de sckit-learn que é a principal biblioteca utilizada para a implementação dos algoritmos ao longo desta tese.

A Tabela 5 fornece uma expressão detalhada dos hiperparâmetros de todos os classificadores escolhidos.

Após todos os algoritmos estarem com os seus hiperparâmetros devidamente otimizados a tese irá se focar no treino dos modelos selecionados, mais em concretamente na técnica de validação dos modelos, ou seja, a forma de como irá treinar e testar os modelos. Para tal, será empregue a técnica de *cross-validation*.

### 3.2.4 K-fold cross validation

Cross-validacion é um procedimento de reamostragem utilizado para avaliar modelos de Machine learning numa amostra de dados limitada.

O procedimento tem um único parâmetro chamado k que se refere ao numero de grupos em que uma determinada amostra de dados deve ser dividida. Como tal, o procedimento é muitas vezes chamado de K-fold cross validation.

No K-fold cross validation o *dataset* é dividido em K subconjuntos e o método de validação é repetido k vezes. Cada vez, que um dos k subconjuntos é usado como o conjunto de teste os outros k-1 subconjuntos são agrupados para formar um conjunto de treino. Quando se escolhe um valor especifico para k, este pode ser utilizado em vez de k na referencia ao modelo, onde se k = 4 torna-se em 4-fold cross validation[41].

Observando a figura 12 é possível visualizar um exemplo de 4-fold cross validation onde um *dataset* é subdividido em 4 subconjunto (Folds) com 4 iterações.

Tendo em conta o tamanho dos *datasets* a tese irá implementar 2-fold cross validation.

Tabela 5: Hiperparâmetros utilizando Otimização bayesiana

| Modelo | Hiperparâmetros | Função | Espaço de pesquisa |
|---|---|---|---|
| DT | criterion | Medir a qualidade de uma divisão. | ['entropy','gini'] |
| | max_depth | Profundidade máxima da árvore. | (1, 100) |
| | min_samples_split | Número mínimo de amostras necessárias para dividir um nó. | (2, 100) |
| | min_samples_leaf | Número mínimo de amostras necessárias para estar num nó de folha. | (1, 100) |
| | max_leaf_nodes | Número máximo de nós. | (2, 100) |
| SVM | C | Parâmetro de regularização. | (1e-6, 100.0) |
| | gamma | Coeficiente do kernel para 'rbf', 'poli' e 'sigmoid'. | (1e-6, 100.0) |
| BagDT & BagNN | n_estimators | O número de estimadores de base no ensemble. | (10, 500) |
| | max_features | O número de variáveis a retirar de X para treinar cada estimador de base. | (1, 58) |
| | max_samples | O número de instâncias a retirar de X para treinar cada estimador de base | (1, 58) |
| GBDT | criterion | A função de medir a qualidade de uma divisão. | ['friedman_mse', 'mse', 'mae'] |
| | loss | A função loss é otimizada.. | ['deviance', 'exponential'] |
| | max_depth | Profundidade máxima dos estimadores de regressão individuais. | (3, 20) |
| | subsample | Fração de amostras a utilizar para a adaptação dos base learners individuais. | (0.1, 1.0) |

| Modelo | Hiperparâmetros | Função | Espaço de pesquisa |
|---|---|---|---|
| RF | n_estimators | O número de árvores na floresta | (10, 500) |
| | max_depth | A profundidade máxima da árvore. | (1, 20) |
| | criterion | A função de medir a qualidade de uma divisão. | ['gini', 'entropy'] |
| XGBoost | n_estimators | O número de estimadores de base no ensemble. | (1, 500) |
| | gamma | Perdas necessárias para fazer uma nova partição num nó de folha da árvore. | (1e-6, 100.0) |
| | eta | Encolhimento utilizado na atualização para evitar o overfitting. | (0.0, 1.0) |
| | max_depth | Profundidade máxima da árvore. | (1, 100) |
| LightGBM | num_leaves | Número máximo de folhas numa árvore. | (10, 100) |
| | max_depth | A profundidade máxima da árvore. | (1, 100) |
| | min_data_in_leaf | número mínimo de dados numa folha. | (1, 100) |
| | gamma | Perdas necessárias para fazer uma nova partição num nó de folha da árvore. | (1e-6, 100.0) |
| CatBoost | learning_rate | Utilizado para reduzir o gradient step | (1e-3, 1) |
| | l2_leaf_reg | Coeficiente no termo L2 regularization da função de custo. | (0.0, 10.0) |
| | random_strength | Quantidade de aleatoriedade a utilizar para pontuar as divisões quando a estrutura em árvore é selecionada. | (0.0, 10.0) |

Tabela 6: Hiperparâmetros utilizando Otimização bayesiana - Continuação

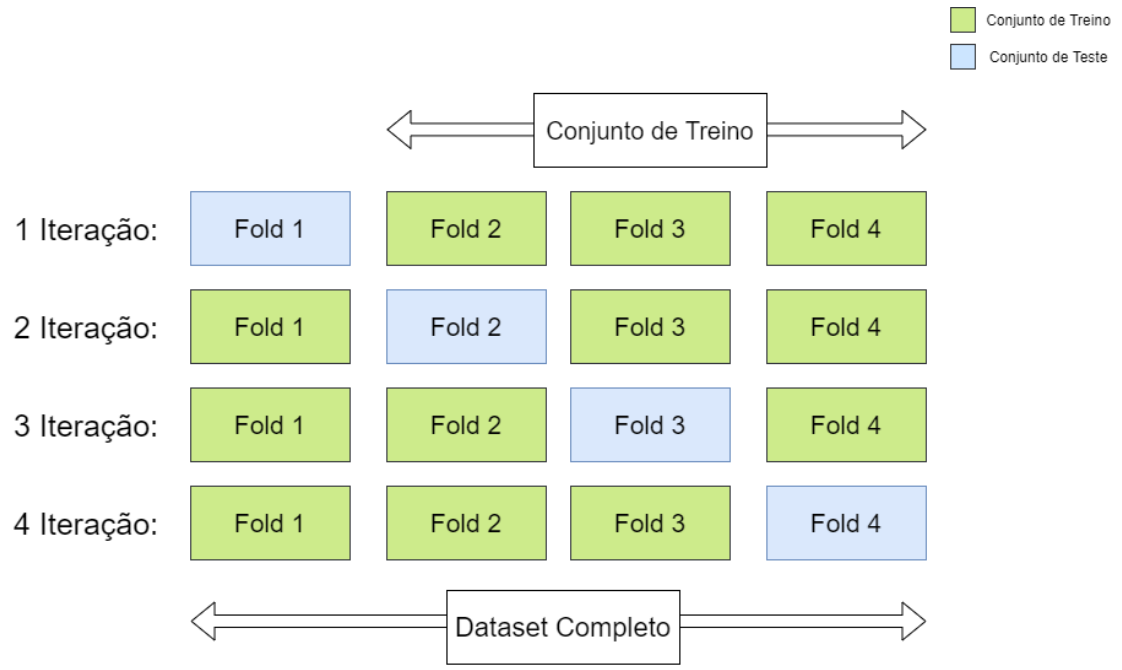



Fig. 12: Exemplo de k-fold cross validation
fonte:[42]

### 3.2.5 Algoritmos de classificação

A tese irá comparar 20 algoritmos de classificação diferentes. Esta seleção foi inspirada por estudos anteriores, referidos na secção 2.1 e abrange várias abordagens diferentes. Os algoritmos dividem-se em classificadores individuais e classificadores *ensemble*.

Dado o grande número de classificadores, não é exequível a utilização de todos os algoritmos. A seguinte tabela lista os métodos selecionados que serão utilizados para o desenvolvimento da tese. Os modelos de classificação individuais serão os *benchmarks*, ou seja, serão o padrão do que os resultados base deverão ser e essenciais para comparar os resultados desta tese com outros estudos relacionados.

Como já referido na sub secção 3.2.3 é usado o método de Otimização bayesiana para ajustar os hiperpârametros dos classificadores utilizados. Na tabela 5 esta representado todos os hiperpârametros que irão ser afinados, é representado os hiperpârametros escolhidos para ajustar cada modelo, a função de tais hiperpârametros e o espaço de pesquisa utilizado pela Otimização bayesiana. Os hiperparâmetros não mencionados na tabela seguem a configuração padrão.

Com um especial foco na implementação de classificadores do tipo ensemble heterogéneos foi desenvolvido pela tese vários classificadores heterogéneos utilizando três tipos de ensemble, Stacking, Majority Voting e Weighted Average. A incorporação dos modelos base nos classificadores de *ensemble* heterogéneos apresentados na tabela 7 estão diretamente relacionados por alguns

Tabela 7: Algoritmos de classificação utilizados

| Tipo de Classificador | Algoritmo de Classificação | Acrónimo | Tipo de Ensemble |
|---|---|---|---|
| Individuais | Regressão linear | LR | - |
| | Rede neuronal artificial | ANN | - |
| | Linear Discriminant Analysis | LDA | - |
| | Árvore de decisão | DT | - |
| | SVM radial base function | SVM-Rbf | - |
| Ensemble homogéneos | Bagged Decision tree | BagDT | Bagging |
| | Bagged MLP | BagMLP | Bagging |
| | Bagged NN | BagNN | Bagging |
| | Random forest | RF | Bagging |
| | Adaptive Boosting | AdaBoost | Boosting |
| | Gradient-Boosted Decision Trees | GBDT | Boosting |
| | XGBoost | XGBoost | Boosting |
| | LightGBM | LightGBM | Boosting |
| | CatBoost | CatBoost | Boosting |
| Ensemble heterogéneos | XGBoost + LightGNM + CatBoost | XLC-SA | Stacking |
| | AdaBoost + RF + LightGBM + CatBoost | ARLC-SA | Stacking |
| | XGBoost + LightGNM + CatBoost | XLC-MV | Majority Voting |
| | RF + LightGBM + CatBoost | RLC-MV | Majority Voting |
| | XGBoost + LightGNM + CatBoost | XLC-WA | Weighted Average |
| | RF + LightGBM + CatBoost | RLC-WA | Weighted Average |

estudos abordados na secção 2.1 e com resultados obtidos em experiências realizadas pela tese. Consequentemente, obtendo os classificadores heterogéneos XLC , RLC e ARLC.

# 4 Analise e tratamento dos dados

Nesta secção será realizado uma analise e tratamento dos *datasets*. Numa primeira fase será realizado uma *Exploratory Data Analysis* (EDA) com o objetivo de compreender os dados em bruto e conseguir detetar quais as *features* que realmente são relevantes para o processo de modelação. Em seguida, será realizado um tratamento de dados que irá preparar os dados de maneira que possam ser utilizados para a modelação com *machine learning*.

## 4.1 Exploratory Data Analysis (EDA)

### 4.1.1 Preparação de dados

É necessário garantir a integridade do *dataset*, verificar se este, possui todos os *Features* que era suposto e se nenhum desses *Features* possui campos vazios, verificar quantas variáveis numéricas e categóricas realmente têm, observar se o *dataset* é equilibrado, ou não, e analisar cada *feature* de forma a conseguir traçar correlações.

```
checkingAccount          0
Duration                 0
CreditHistory            0
Purpose                  0
CreditAmount             0
SavingsAccount           0
employment               0
InstallmentRate          0
PersonalStatusSex        0
debtorsGuarantors        0
residence                0
Property                 0
Age                      0
Otherinstallment         0
Housing                  0
existingCreditsatBank    0
Job                      0
peopleBeingLiable        0
Telephone                0
foreignWorker            0
Risk                     0
dtype: int64
```

(a) Soma dos valores nulos

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 21 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   checkingAccount       1000 non-null   object
 1   Duration              1000 non-null   int64
 2   CreditHistory         1000 non-null   object
 3   Purpose               1000 non-null   object
 4   CreditAmount          1000 non-null   int64
 5   SavingsAccount        1000 non-null   object
 6   employment            1000 non-null   object
 7   InstallmentRate       1000 non-null   int64
 8   PersonalStatusSex     1000 non-null   object
 9   debtorsGuarantors     1000 non-null   object
 10  residence             1000 non-null   int64
 11  Property              1000 non-null   object
 12  Age                   1000 non-null   int64
 13  Otherinstallment      1000 non-null   object
 14  Housing               1000 non-null   object
 15  existingCreditsatBank 1000 non-null   int64
 16  Job                   1000 non-null   object
 17  peopleBeingLiable     1000 non-null   int64
 18  Telephone             1000 non-null   object
 19  foreignWorker         1000 non-null   object
 20  Risk                  1000 non-null   int64
dtypes: int64(8), object(13)
memory usage: 164.2+ KB
```

(b) Informação geral do Dataset

Fig. 13: Informação do Dataset

Na figura 13a, é feita a soma de todos os campos com valores nulos do *dataset*, conclui-se que todos os valores resultantes destas somas devolveram zero, o que significa que este *dataset* não possui campos com valores nulos. Na figura adjacente 13b confirma-se que o número de *features* com valores categóricos e numéricos se encontra coerente, quando comparado com o que já tinha sido estabelecido previamente, na secção 3.1.1. É de realçar que nestas figuras é representada uma nova *Feature(Risk)* que irá ser usada como *Label/Target*.

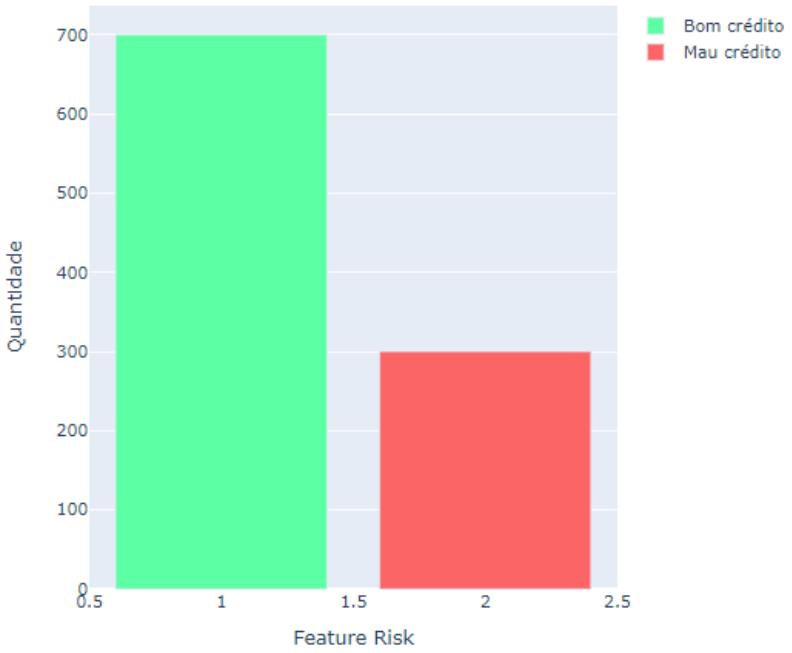
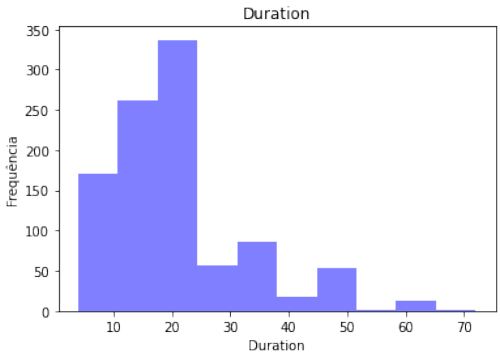
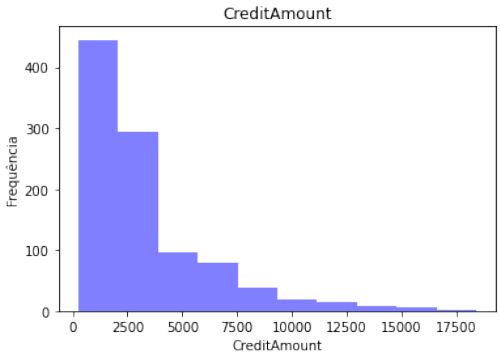



Fig. 14: Distribuição da feature Risk

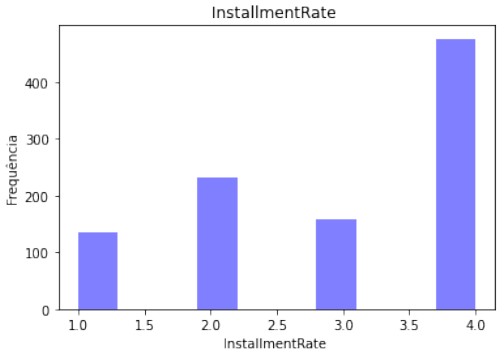
O *dataset* alemão da UCI é ligeiramente desequilibrado, por possuir mais dados etiquetados de uma certa classe do que de outra. A figura 14, valida esta afirmação, onde é possível observar uma discrepância entre estas duas classes, 700 dados foram etiquetados com o valor numérico de 1 que simboliza créditos atribuídos correctamente e os restantes 300 foram etiquetados com o valor numérico de 2, que se refere a créditos atribuídos incorrectamente. Esta variável em específico irá ser utilizada como *Label* ou *target* do modelo, que significa que irá ser utilizada como o *Output* do modelo. Ou seja, as previsões feitas pelo modelo enquanto o treino e validação do modelo, irão ser comparadas com os valores do *label* de modo a confirmar se tal previsão é realmente correcta ou não.



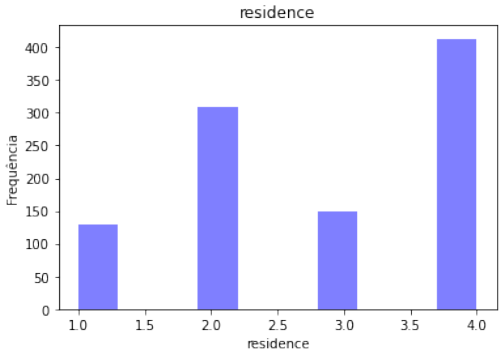

(a) Duração de empréstimo



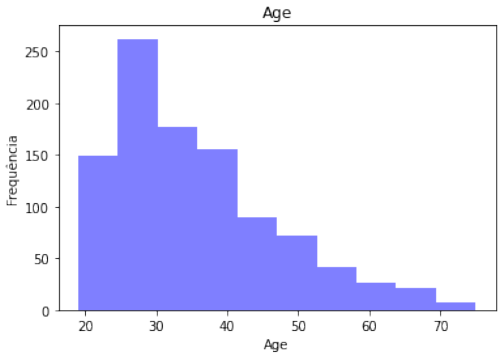

(b) Quantidade do empréstimo



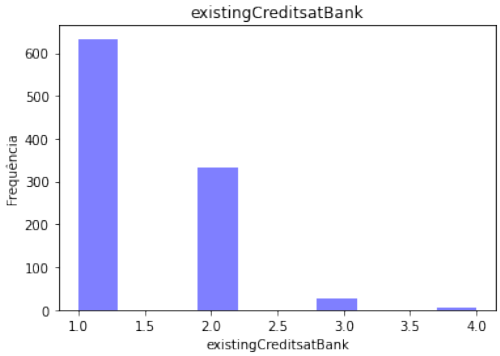

(c) Taxa de prestações



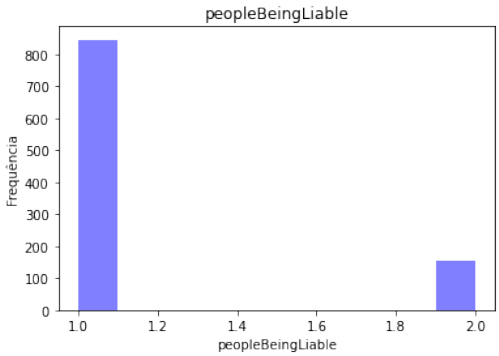

(d) Residência atual



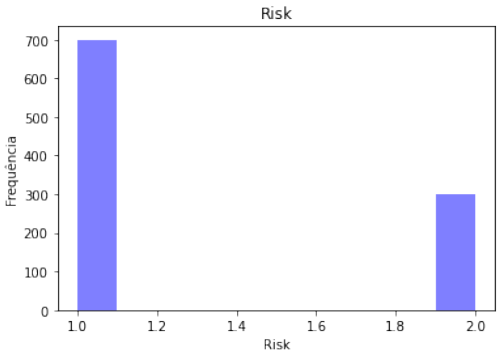

(e) Idade




(f) Número de créditos




(g) Número de pessoas responsáveis




(h) Riscos de empréstimos

Fig. 15: Distribuição das features numéricas

### 4.1.2 Análise de *Features* Numéricas

Tendo em conta, a distribuição de todas as *features* numéricas retratadas na figura 15, é possível realizar várias observações à cerca de como os dados se encontram distribuídos pelo *dataset*. Analisando, os gráficos de distribuição é possível afirmar que, a maioria dos empréstimos é feita por pessoas com idades compreendidas, entre os 20 a 40 anos, com quantias até 4 mil DM numa duração que pode ir de 10 a 25 meses.

Outras informações relevantes, que são possíveis retirar são: No gráfico 15d, a maior parte dos créditos foram dados a pessoas com residência atual de quatro anos; A maioria das pessoas que pediram crédito já tinham um crédito existente no mesmo banco segundo o gráfico 15f;

Algumas *features* são mais aliciantes do que outras, especialmente quando o objetivo é realizar uma análise mais profunda aos dados. Dito isto, a tese irá se focar prioritariamente na análise de certas *features* numéricas, nomeadamente , na duração do crédito, a quantia do crédito e a idade do cliente, contrastando-as sempre com o seu risco, sendo possível, averiguar se tal variável tem um certo tipo de ligação quando se trata de um bom ou mau empréstimo e assim traçar possíveis correlações.

Iniciando a análise da duração dos créditos, a tese afirma que a maioria dos créditos registados neste *dataset* têm uma duração de 6 , 12 , 18 e 24 meses, correspondendo nomeadamente a meio ano, um ano , um ano e meio e dois anos. Uma preferência pela duração de crédito é evidente, tanto seja por parte dos clientes ou pela do banco. É de salientar que créditos com 6, 12 e 24 meses aparentam ser os mais benéficos para o banco, tendo em consideração o primeiro gráfico da figura 16, onde as quantidades de bons créditos nestas durações são consideravelmente mais elevados do que os maus créditos, revelando uma provável correlação entre a duração de um crédito e o seu risco.

No segundo gráfico da figura 16, é realizado uma análise à quantidade concedida de cada crédito com a sua duração. Existindo uma correlação aparente entre estas duas *features*, uma vez que, quando maior for a duração de um crédito, maior a quantia do crédito, por consequência o gráfico possui uma tendência ascendente.

Observando o terceiro gráfico da figura 16, na duração entre os 0-20 meses os valores médios da frequência de bons empréstimos são superiores, à frequência média da dos maus, e durações com mais de 20 meses apresentam geralmente, frequências médias de maus empréstimos com valores superiores à média de bons empréstimos. Logo existe uma maior probabilidade de créditos com duração entre 0 - 20 meses de serem considerados bons créditos e créditos com a duração superior a 20 meses de serem maus créditos.

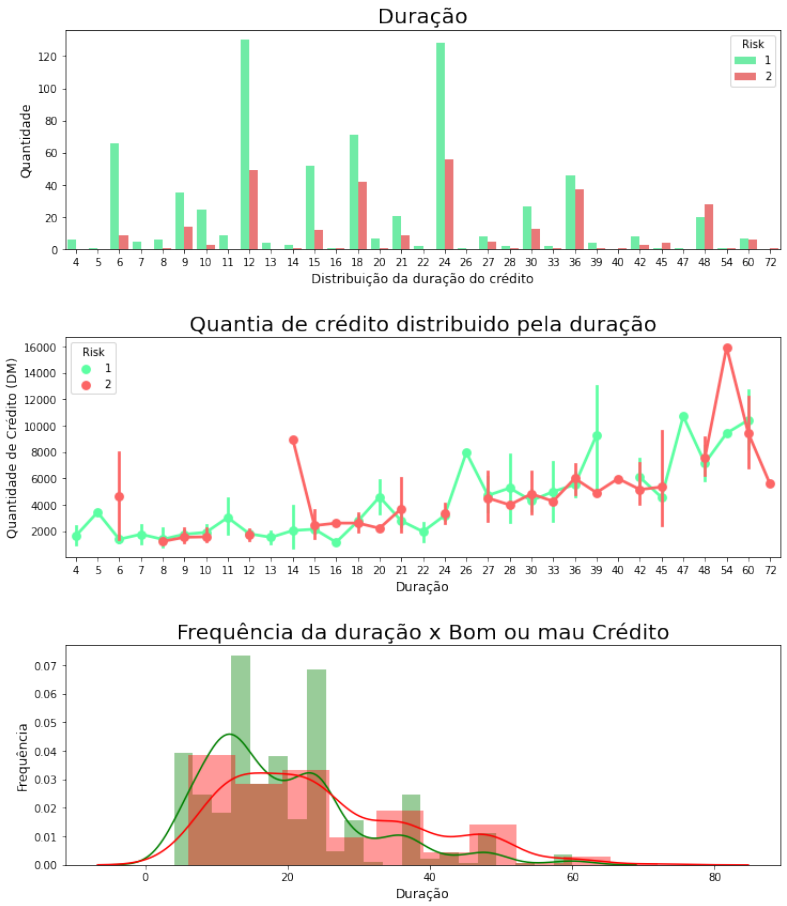



Fig. 16: Analise aprofundada a feature numérica duração do crédito

Analisando a distribuição de pessoas, responsáveis pela manutenção do crédito, foi possível averiguar que este recurso não irá ser utilizado para o treino de um modelo de ML, pela razão de não existir correlação entre o número de pessoas responsáveis e o risco. É possível observar esta afirmação na figura 17, onde ambos os rácios do risco de crédito são iguais, o que não mostra diferença entre uma ou duas pessoas.

De forma a facilitar a análise da idade, foi criado diferentes faixas etárias, com base nas idades dos clientes. Sendo mais simples de visualizar os valores de crédito emprestado aos clientes, pertencentes a cada faixa etária. Esta nova coluna é constituída por valores categóricos de forma a conseguir representar as idades dos clientes por faixa etária:

- **Jovem adulto:** Clientes com idades compreendidas entre os 19 e os 29 anos.
- **Adultos:** Clientes com idades compreendidas entre os 30 e os 40 anos.
- **Seniores:** Clientes com idades compreendidas entre os 41 e os 55 anos.
- **Idosos adulto:** Clientes com idades superior a 55 anos.

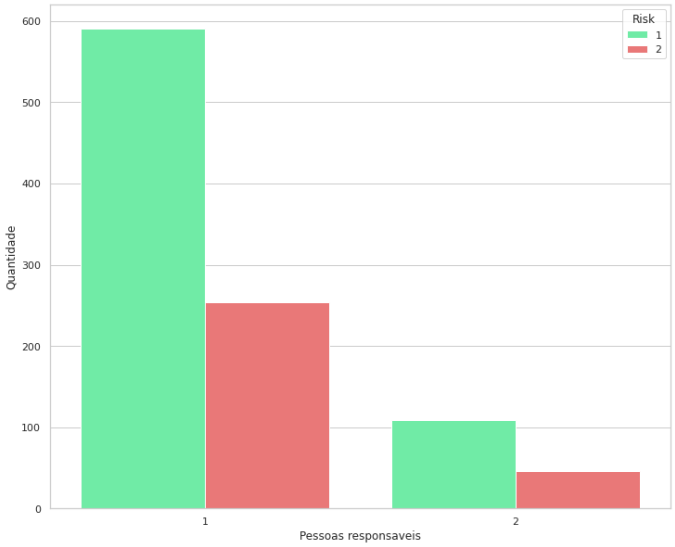



Fig. 17: Distribuição do número pessoas responsáveis pela manutenção do crédito

Analisando o primeiro gráfico da figura 18, é plausível afirmar que a maioria dos empréstimos feitos pelo banco, são feitos a jovens adultos e a adultos. Uma possível razão desta distribuição, poderá ser devido a que nestas faixas etárias é mais comum o investimento em habitações ou automóveis. Neste gráfico também é observável, o facto de que, clientes nas faixas etárias de adultos, seniores e idoso sejam um melhor investimento de crédito em relação a jovens adultos, devido a terem uma maior quantidade de bons créditos.

Semelhantemente, as conclusões anteriormente retiradas da análise da figura 16, no segundo gráfico da figura 18, confirmam que quanto maior for a duração de um crédito, mais propício será de ser um mau crédito. Empréstimos realizados a qualquer faixa etária, com uma duração até 20 meses têm uma probabilidade significativamente maior de serem um bom empréstimo, do que empréstimos com uma duração superior a 20 meses.

Resumindo, toda a análise anteriormente feita é de salientar as seguintes principais observações:

– **Clientes mais jovens tendem a ser um maior risco do que clientes mais velhos.** Tento em conta que, por norma clientes mais velhos, tenham mais estabilidade financeira do que clientes mais novos. Num ponto de vista bancário, talvez seja melhor apresentar melhores programas de crédito a clientes com maiores condições de pagamento.

– **Maiores quantias de crédito apresentam mais riscos que quantias menores.** Pela perspetiva de um cliente, que contraiu um empréstimo ou entrou num programa de crédito, é mais fácil pagar uma quantia baixa.

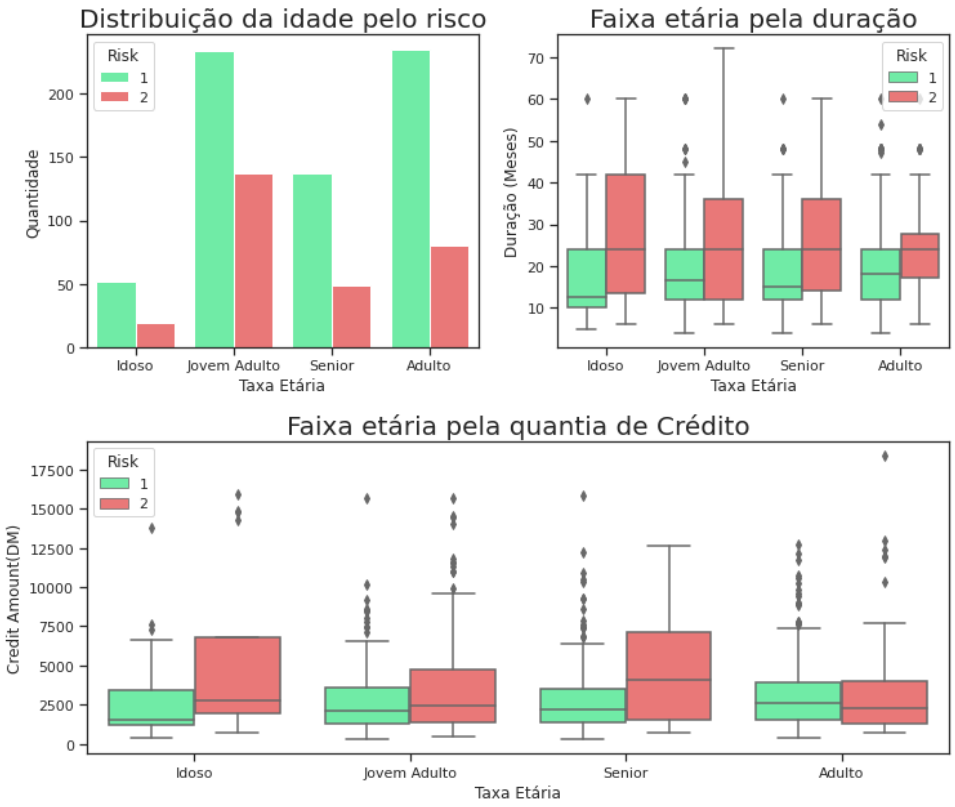



Fig. 18: Análise aprofundada a feature numérica idade

– **Quanto maior a duração maior o risco.** Programas de crédito comuns, são baseados em juros ou taxas que aumentam ao longo do tempo, clientes com grandes quantias de crédito durante muito tempo terão por norma de pagar muito mais até o fim.

### 4.1.3 Análise de Features Categóricas

Nesta secção serão abordados as *features* de valores categóricos.Numa primeira fase, serão realizadas análises a *features* categóricas que irão fornecer uma visão geral do *dataset* em termos de distribuição dos seus dados. Para cada coluna categórica, é agora possível saber quais são as maioridades. Entretanto, para dar mais poder à análise, é acrescentado a cada entrada categórica a aprovação do risco. As próximas figuras ilustram gráficos distributivos das *features* com valores categóricos.

A tese irá realizar, uma breve análise de distribuição de cada *feature*, sendo que algumas serão necessários análises mais específicas e profundas. Começando a análise pelo gráfico 19a é de realçar o facto de que a maioria dos clientes, com crédito do banco não possuem conta corrente no mesmo(A14), no entanto, em comparação a outros com contas correntes, estes clientes de longe são os que possuem uma maior quantidade de créditos positivos. Outro ponto importante a realçar

é que clientes com contas correntes com valores inferiores a 0 DM(A11) ou valores compreendidos entres 0 a 200 DM(A12), são os que apresentam quantidades de créditos negativos mais elevado, onde no caso da A11 chegando quase aos 50%.

Segundo o gráfico 19c, o propósito principal dos clientes, ao pedirem um crédito ao banco é adquirir carro, seja ele um carro novo(A40) ou um carro usado(A41), outro propósito com grande frequência é a compra de Televisão/Rádio(A43). Observando a distribuição dos dados comparando com o seu risco é de notar que propósitos como carro usado(A41), televisão/rádio(A43) e requalificações(A48) demonstram os melhores rácios, sendo que a probabilidade de créditos com este propósito, resultam maioritariamente em créditos positivos.

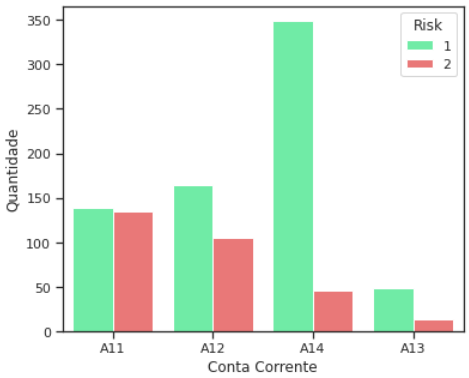
Analisando, a distribuição das contas poupanças é possível concluir que, a maioria dos clientes que procuram créditos possuem contas poupanças a baixo dos 100 DM(A61), não demonstrando um rácio favorável em relação ao risco, em comparação com os outros tipos de conta poupança. Uma correlação aparente deste gráfico é o facto de quanto maior for o valor da conta poupança mais provável é o risco do crédito ser considerado bom. Clientes com poucas poupanças são mais propícios a ter um mau crédito.

Clientes solteiros do sexo masculino representam, a maioria dos clientes. Um dos problemas que esta *feature* apresenta é o facto de que, o campo A92 representa clientes do sexo feminino que sejam divorciadas/Separadas e casadas o que torna este campo muito abrangente e um pouco contraditório, tornando-se impossível de avaliar mulheres casadas.

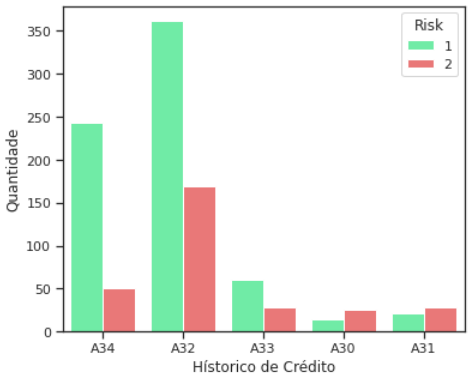
Um atributo importante a ter em atenção ao analisar, é o rendimento do cliente, visto que, um cliente sem rendimento algum não irá conseguir liquidar o seu crédito. Para este fim a tese irá analisar dois gráficos distributivos, nomeadamente, a distribuição do tipo de trabalho de cada cliente(figura 19f), e quantos anos cada cliente possui no trabalho atual(figura 19g). No primeiro gráfico, é possível determinar que a maioria dos créditos são atribuídos a clientes que são trabalhadores qualificados e efetivos(A173), realçando uma frequência de bons empréstimos superior a todos os outros. No segundo gráfico é de salientar que, clientes com o mesmo emprego entre 1 a 4 anos(A73) demonstram uma quantidade superior de empréstimos pedidos, no entanto trabalhadores com mais de 7 anos de trabalho(A75), mostram um rácio mais positivo no que toca ao risco do crédito. Logo é possível traçar duas correlações destes dois gráficos:

- Por norma, clientes que sejam qualificados são mais propícios a terem um empréstimo considerado positivo que clientes não qualificados.

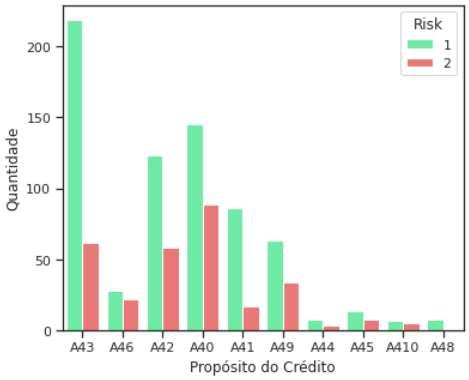
- Quanto mais anos um cliente tiver no seu emprego atual mais provável é de conseguir saudar o crédito.



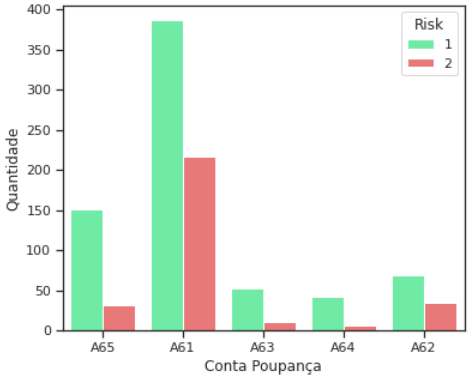

(a) Distribuição de contas correntes



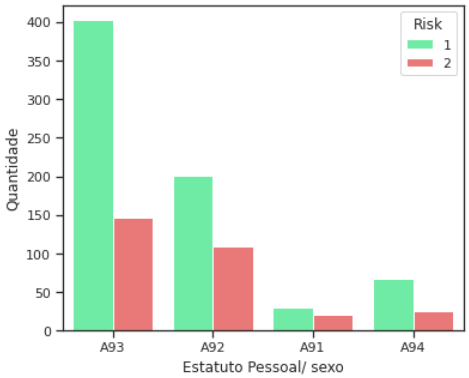

(b) Distribuição do Histórico de crédito



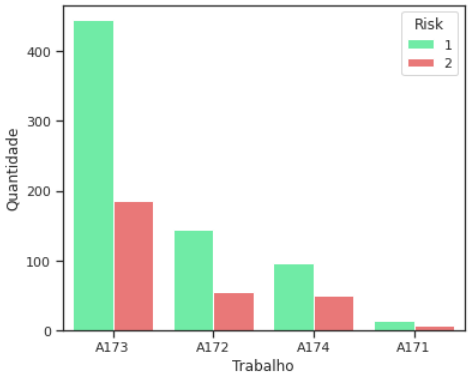

(c) Distribuição do propósito de crédito



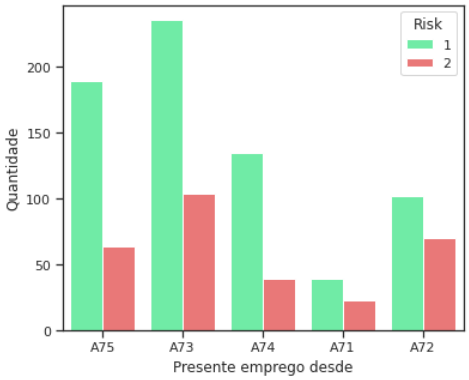

(d) Distribuição das contas poupanças



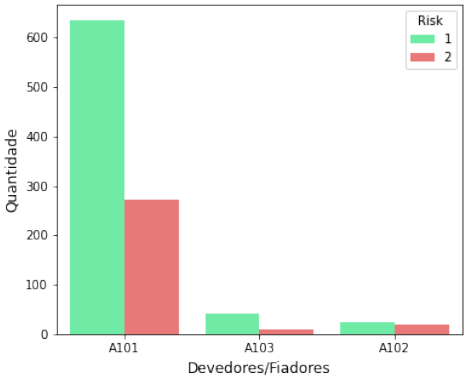

(e) Distribuição do estatuto/sexo




(f) Distribuição dos trabalhos

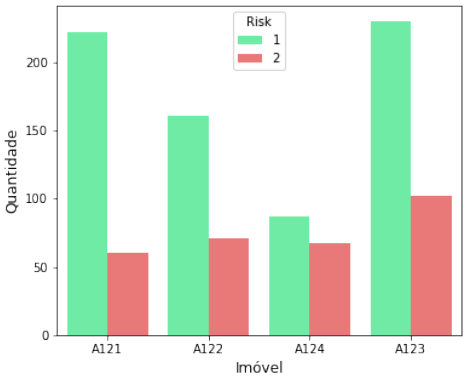



(g) Distribuição de presente emprego

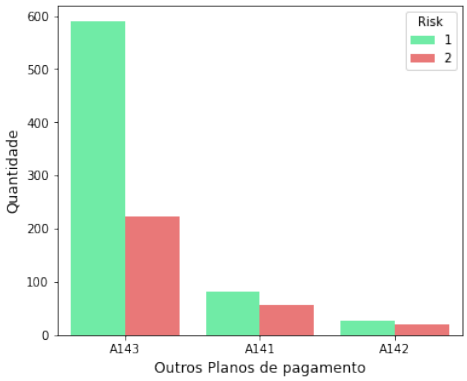



(h) Distribuição de devedores/fiadores

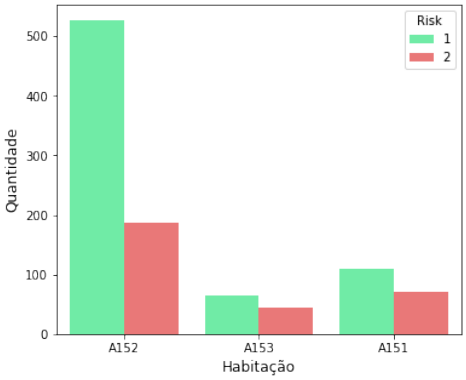
Fig. 19: Gráficos distributivos dos features categóricos



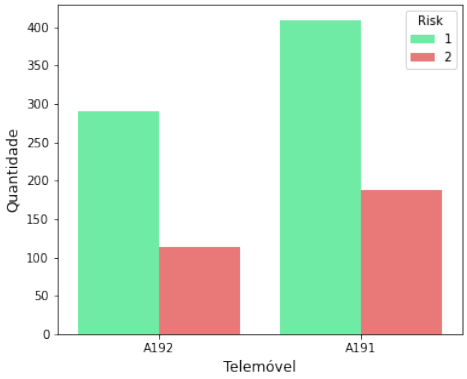

(a) Distribuição de imóveis



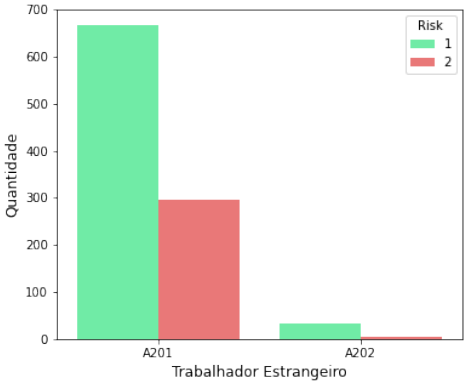

(b) Distribuição de planos de pagamento




(c) Distribuição de Habitações




(d) Distribuição de telemóveis




(e) Distribuição trabalhadores estrangeiros

Fig. 20: Gráficos distributivos dos features categóricos(Cont.)

Analisando, a distribuição dos dados de trabalhadores estrangeiros, revela que a grande maioria dos clientes deste *dataset* são trabalhadores estrangeiros(A201).

Ao visualizar, a distribuição dos clientes que têm ou não telemóvel associado, é de concluir que esta *feature* não possui qualquer tipo de relação com o risco do crédito, isto é possível observar no gráfico da figura 20d, onde apesar de haver uma quantidade diferente entre ter, ou não, o nome associado a um telemóvel o rácio entres eles é idêntico, o que indica não existir nenhuma correlação possível, tornando esta *feature* inútil e irá ser excluída do *dataset*.

Observando o gráfico 21, que representa a distribuição da habitação, verificamos que a maioria dos clientes que têm créditos possuem habitação própria(A152).

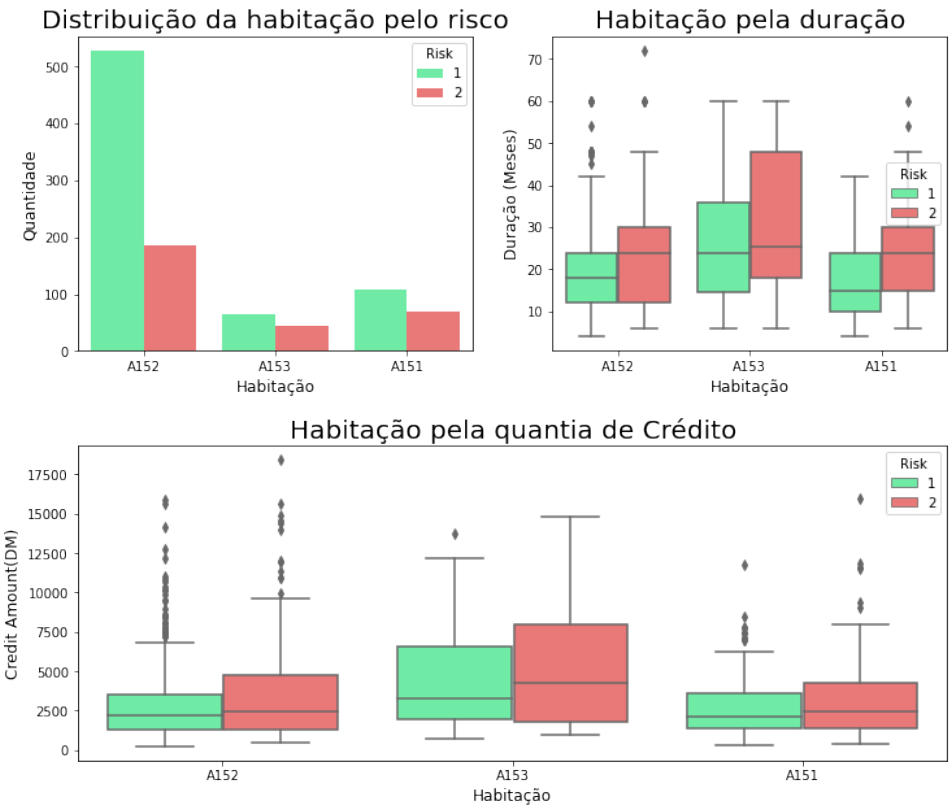



Fig. 21: Análise profunda à *feature* numérica idade

Ao contrastar, a habitação com a duração e a quantia de crédito representado na figura 21, é possível afirmar que, clientes com habitação própria(A142) e com habitação alugada(A151), apresentam uma duração de crédito e uma quantia de crédito menor do que clientes que possuem casa de graça(A143). Observando, a quantidade de bons e maus créditos clientes com casa própria apresentam valores mais favoráveis que todos os outros. Disto isto, clientes que possuem casa própria têm em média créditos com uma duração menor, com uma quantia de crédito menor e

com uma maior frequência de casos positivos de crédito do que clientes com casa alugada ou gratuita.

Resumindo toda a análise feita anteriormente, é de salientar as principais observações:

- **Os clientes com pouca conta poupança e conta corrente apresentam maior risco.**
- **Clientes que são trabalhadores qualificados e com um maior número de anos de trabalho apresentam menos risco.** Quanto melhor as condições dos rendimentos do cliente mais propicio será a saudar o crédito.
- **Quando os clientes pedem crédito para carros novos, é um sinal de que o risco pode ser mais elevado.**
- **Clientes com casa própria apresentam menos risco.**
- **Clientes solteiros do sexo masculino representam a maioria dos clientes.**

## 4.2 Pré-processamento de Dados

Os dados recolhidos, do seu domínio são referidos como dados em bruto e são recolhidos no contexto de um problema, que se pretende resolver. Dados em bruto podem ser definidos como sendo os dados na forma fornecida a partir de um domínio. Neste caso ambos *datasets* guardados num formato CSV ainda se encontram no seu estado bruto.

Em quase todos os casos, os dados brutos precisarão de ser alterados, antes que possam ser utilizados como base para a modelação com *Machine Learning*. Os casos sem preparação de dados, são tão raros ou triviais que é praticamente uma regra, preparar dados brutos em todos os projetos de *Machine Learning*.[37]

Nesta secção, a tese irá realizar uma preparação dos dados de forma a que, estes fiquem de acordo com o processo de modelação com algoritmos de ML. Para tal é preciso resolver vários problemas, que por norma dados em bruto trazem para um projeto.

### 4.2.1 One-hot encoding

A maioria dos algoritmos de *Machine Learning* não conseguem receber dados sem ser em valores numéricos. E como já foi referido na secção de Recolha de dados, ambos os *datasets* possuem varias variáveis que não são valores numéricos mas sim valores categóricos. Na figura 22, é representado um excerto do *dataset* alemão onde é possível observar algumas das tais variáveis com valores categóricos(p. ex. "checkingAccount" , "CreditHistory", etc..)

| | checkingAccount | Duration | CreditHistory | Purpose | CreditAmount | SavingsAccount | employment | InstallmentRate |
|---|---|---|---|---|---|---|---|---|
| 0 | A11 | 6 | A34 | A43 | 1169 | A65 | A75 | 4 |
| 1 | A12 | 48 | A32 | A43 | 5951 | A61 | A73 | 2 |
| 2 | A14 | 12 | A34 | A46 | 2096 | A61 | A74 | 2 |
| 3 | A11 | 42 | A32 | A42 | 7882 | A61 | A74 | 2 |
| 4 | A11 | 24 | A33 | A40 | 4870 | A61 | A73 | 3 |

Fig. 22: Dataset Alemão em bruto

A tese irá utilizar *One-Hot Encoding* como forma de resolver este problema, esta abordagem permite que a representação de dados categóricos seja mais expressiva. Muitos algoritmos de ML não conseguem trabalhar diretamente com dados categóricos, logo dados categóricos devem ser convertidos em números. Isto é necessário, tanto para variáveis de entrada como de saída.

*One-Hot Encoding* simplesmente pega numa coluna que tenha dados categóricos, que tenha sido codificada com etiquetas e divide a coluna em várias novas colunas, consoante o número de diferentes categorias que essa coluna possui[43]. Observando, a figura 22, a coluna *checkingAccount* tem valores categóricos(A11 , A12 , A13 e A14), aplicando o *one-hot encoding* cada um destes valores vai passar a ser uma coluna nova, e dependendo do valor de cada instância a nova coluna terá valores numéricos entre 1 ou 0. A figura 23, demonstra a coluna antes e depois da aplicação do *one-hot encoding*.

| checkingAccount |
|---|
| A11 |
| A12 |
| A14 |
| A11 |
| A11 |

| checkingAccount_A11 | checkingAccount_A12 | checkingAccount_A13 | checkingAccount_A14 |
|---|---|---|---|
| 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 |

Fig. 23: Aplicação do *One-Hot Encoding* no *dataset* alemão

### 4.2.2 Feature scaling

*Feature scaling* em *Machine Learning* é um dos passos mais críticos durante o pré-processamento de dados, antes de criar um modelo de ML. O dimensionamento pode ser a diferença entre um modelo fraco de ML e um modelo melhor. As técnicas mais comuns de *feature scaling* são a Normalização e a Padronização[44].

Como já foi referido anteriormente, algoritmos de ML apenas compreendem valores numéricos, no entanto se existir uma grande diferença no intervalo desses valores numéricos, onde certos

valores podem variar entre os milhares e outros podem variar entre as dezenas, isto faz com que exista uma suposição subjacente de que números mais elevados têm alguma superioridade. Assim sendo, estes números mais significativos começam a desempenhar um papel mais decisivo quando a altura do treino do modelo.[44]

O algoritmo de ML, funciona em números e não sabe o que esse número representa. Uma duração de 20 meses e um idade de 20 anos representam duas coisas completamente diferentes, mas como variável de um modelo são tratadas como iguais.

| Duration | CreditAmount | InstallmentRate | residence | Age | checkingAccount_A11 | checkingAccount_A12 |
|---|---|---|---|---|---|---|
| 6 | 1169 | 4 | 4 | 67 | 1 | 0 |
| 48 | 5951 | 2 | 2 | 22 | 0 | 1 |
| 12 | 2096 | 2 | 3 | 49 | 0 | 0 |
| 42 | 7882 | 2 | 4 | 45 | 1 | 0 |
| 24 | 4870 | 3 | 4 | 53 | 1 | 0 |

Fig. 24: Dataset Alemão antes de feature scaling

Dando como exemplo referenciando a figura 24 onde temos duas features de "Duration" e "CreditAmount". A duração não pode ter uma comparação significativa com a quantia de crédito. Assim o algoritmo de suposição faz com que, "Duration" $<$ "CreditAmount", portanto faz com que a feature de "CreditAmount", seja mais importante do que a "Duration".

*Feature* com números mais significativos, começam a desempenhar um papel mais decisivo quando modelo é treinado. Assim, *Feature scaling* é necessário para colocar todas as *features* em um pé de igualdade sem qualquer importância inicial. Outra das razões, pela qual será feito um *feature scaling* é que alguns algoritmos como de redes neuronais de *gradient descent* são convertidos muito mais rapidamente.

Apesar de existir várias abordagens para *feature scaling* a tese irá utilizar a Min Max Scaler. Que é representada pela seguinte equação:

$$x_{novo} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{4}$$

Transforma as *features* escalando-as para um determinado intervalo. Este, transforma cada *feature* individualmente de modo a que, esteja na gama dada no conjunto de treino. Para este caso o intervalo será definido entre zero e um.

| Duration | CreditAmount | InstallmentRate | residence | Age | checkingAccount_A11 | checkingAccount_A12 |
|---|---|---|---|---|---|---|
| 0.029412 | 0.050567 | 1.000000 | 1.000000 | 0.857143 | 1 | 0 |
| 0.647059 | 0.313690 | 0.333333 | 0.333333 | 0.053571 | 0 | 1 |
| 0.117647 | 0.101574 | 0.333333 | 0.666667 | 0.535714 | 0 | 0 |
| 0.558824 | 0.419941 | 0.333333 | 1.000000 | 0.464286 | 1 | 0 |
| 0.294118 | 0.254209 | 0.666667 | 1.000000 | 0.607143 | 1 | 0 |

Fig. 25: Dataset alemão pós aplicação do Min Max Scaler

Como é possível observar na figura 25, todos os valores do *dataset* estão compreendidos num intervalo entre [0,1].

# 5 Resultados

Esta secção irá abordar os resultados obtidos ao longo da tese, sendo possível observá-los na seguinte tabela 8.

Tabela 8: Resultados modelos

| Classificador | Modelo | Dataset | ACC | AUC | Recall | BS |
|---|---|---|---|---|---|---|
| Individuais | LR | Alemão | 0.740 | 0.647 | 0.416 | 0.260 |
| | | Australiano | 0.856 | 0.858 | 0.876 | 0.143 |
| | ANN | Alemão | 0.738 | 0.659 | 0.462 | 0.263 |
| | | Australiano | 0.859 | 0.860 | 0.869 | 0.140 |
| | LDA | Alemão | 0.746 | 0.664 | 0.462 | 0.254 |
| | | Australiano | 0.857 | 0.862 | **0.905** | 0.142 |
| | DT | Alemão | 0.731 | 0.633 | 0.392 | 0.269 |
| | | Australiano | 0.856 | 0.863 | 0.934 | 0.143 |
| | SVM | Alemão | **0.755** | **0.672** | **0.466** | **0.245** |
| | | Australiano | **0.866** | **0.867** | 0.884 | **0.133** |
| Homogéneos | BagDT | Alemão | 0.742 | 0.608 | 0.273 | 0.258 |
| | | Australiano | 0.865 | 0.865 | 0.868 | 0.134 |
| | BagMLP | Alemão | 0.723 | 0.609 | 0.326 | 0.277 |
| | | Australiano | 0.855 | 0.855 | 0.862 | 0.144 |
| | BagNN | Alemão | 0.751 | 0.678 | 0.497 | 0.249 |
| | | Australiano | 0.863 | 0.861 | 0.843 | 0.136 |
| | RF | Alemão | 0.758 | 0.639 | 0.343 | 0.253 |
| | | Australiano | 0.862 | 0.856 | 0.807 | 0.137 |
| | AdaBoost | Alemão | 0.746 | 0.676 | **0.502** | 0.254 |
| | | Australiano | 0.823 | 0.821 | 0.803 | 0.176 |
| | GBDT | Alemão | 0.760 | 0.673 | 0.456 | 0.243 |
| | | Australiano | 0.842 | 0.841 | 0.819 | 0.157 |
| | XGBoost | Alemão | 0.758 | 0.661 | 0.420 | 0.242 |
| | | Australiano | 0.871 | 0.872 | 0.882 | 0.128 |
| | LightGBM | Alemão | 0.765 | **0.680** | 0.469 | 0.235 |
| | | Australiano | 0.872 | 0.873 | 0.873 | 0.127 |
| | CatBoost | Alemão | **0.766** | 0.679 | 0.463 | **0.233** |
| | | Australiano | **0.873** | **0.875** | **0.889** | **0.126** |
| Heterogéneos | XLC-SA | Alemão | 0.761 | 0.660 | 0.410 | 0.239 |
| | | Australiano | 0.868 | 0.869 | 0.880 | 0.131 |
| | ARLC- SA | Alemão | 0.763 | 0.659 | 0.403 | 0.237 |
| | | Australiano | 0.865 | 0.864 | 0.853 | 0.134 |
| | XLC-MV | Alemão | 0.772 | 0.681 | 0.453 | 0.227 |
| | | Australiano | 0.868 | 0.868 | 0.870 | 0.131 |
| | RLC-MV | Alemão | **0.774** | 0.677 | 0.437 | **0.225** |
| | | Australiano | **0.875** | **0.875** | 0.873 | **0.124** |
| | XLC-WA | Alemão | 0.750 | 0.680 | **0.493** | 0.250 |
| | | Australiano | 0.870 | 0.870 | **0.908** | 0.130 |
| | RLC-WA | Alemão | 0.764 | **0.687** | 0.481 | 0.236 |
| | | Australiano | 0.855 | 0.854 | 0.865 | 0.144 |

Os modelos apresentam os resultados obtidos, das métricas selecionadas na revisão de literatura, onde cada resultado é apresentado em formato percentual. Para uma visualização mais simplificada os resultados com valores mais significativos de cada classificador encontram-se a negrito.

A análise destes resultados terá como principal objetivo, delinear quais dos modelos comummente propostos pelos estudos abordados, são ou não relevantes. Salientando que, os resultados obtidos utilizam o planeamento da modelação recomendado pela tese, representado na figura 11, e se a utilização de classificadores *ensemble* contribuem, ou não, para um melhor desempenho na classificação de crédito, em relação aos métodos mais tradicionais como LR, LDA e ANN [11].

Numa primeira análise da tabela é possível verificar que a maioria dos classificadores de *ensemble* oferecem um melhor desempenho do que os classificadores individuais, esta afirmação vai de acordo com outros estudos, como por exemplo de Barboza et al. [11] e Dastile et al. [4]. Outro ponto possível de retirar, é o facto que em geral os classificadores heterogéneos têm um desempenho ligeiramente melhor do que os classificadores homogéneos.

Observando os resultados obtidos pelos classificadores individuais, que são os *benchmarks*, é possível concluir que o modelo SVM, obteve melhor desempenho em quase todas as métricas. Tanto no *dataset* alemão como no australiano, é de realçar os resultados de 67% e 87% obtidos na métrica AUC, este corresponde à área sob o ROC. ROC é uma métrica abrangente, que reflete as variáveis contínuas de sensibilidade e especificidade. Revela a relação entre sensibilidade e especificidade dos algoritmos de classificação de crédito, logo quanto maior for o valor da AUC, melhor será o desempenho da classificação [5].

Nos classificadores homogéneos é evidente que os modelos que se destacam são os modelos de XGBoost, CatBoost e LightGBM. Visto que, estes modelos são de gradient boosting, poderá indicar que a utilização deste tipo de modelos seja mais adequado, para casos de classificação binária.

CatBoost foi o modelo que mais se destacou entre estes modelos, apresentando resultados de 77% e 87% de exatidão nos *datasets* alemão e australiano respetivamente, obteve 68% e 88% na métrica de AUC e 46% e 89% na sensibilidade, sendo que o valor de 46% de sensibilidade no *dataset* alemão revela, que o algoritmo está a ter dificuldades a identificar os casos verdadeiramente positivos, ou seja, este algoritmo não está a conseguir prever realmente, o que são casos de um mau empréstimo.

Estes valores podem querer insinuar que o *datasets* alemão seja desequilibrado, por possuir mais dados etiquetados de uma certa classe do que de outra. Algo já observado no gráfico da figura 14 na secção de análise e tratamento de dados.

Em relação à métrica BS obteve resultados de 0.23 e 0.13 nos *datasets* alemão e australianos respetivamente, o que demonstra o nível de confiança do modelo. BS indica a medida de quão longe as suas previsões estão dos verdadeiros valores, logo quanto menor for o valor de BS, melhor a previsão.

Observando os classificadores *ensemble* heterogéneos, é de realçar os resultados obtidos pelo modelo proposto pela tese RLC-MV, obtendo os melhores resultados de 77% e 87% na métrica exatidão, e 0.22 e 0.12 na métrica de BS nos *datasets* alemão e australiano, respetivamente. Porém, na métrica de sensibilidade do *dataset* alemão obteve 44%, voltando a afirmar que apesar de este modelo obter um valor de 77% na exatidão, apenas obteve 44% na sensibilidade, o que nos indica que este modelo não é capaz de detetar corretamente todos os casos que realmente são um mau empréstimo. Confirmando o facto de que um *dataset* com dados desequilibrados, pode ter uma grande influência para este tipo de classificação.

O modelo XLC-WA obteve 50% e 90% de sensibilidade, tornando-se o modelo com o melhor desempenho, nesta métrica. Sendo possível afirmar que, este modelo foi capaz de prever metade dos maus empréstimos do *dataset* alemão e 90% no *dataset* australiano. Apesar dos resultados obtidos nas outras métricas, não serem os melhores é preciso ter em conta o foco, para o qual estes modelos estão a ser treinados, no sentido de que, numa instituição bancária é preferivél não conseguir prever um bom empréstimo, do que não conseguir prever um mau empréstimo.

## 5.1 Comparação dos resultados

A utilização de datasets públicos e métricas em comum asseguram uma comparação viável com os estudos relacionados abordados na revisão de literatura. A seguinte tabela apresenta uma comparação utilizando os resultados obtidos pelos classificadores propostos pela tese com os resultados dos classificadores propostos pelos estudos abordados.

Esta comparação será realizada apenas para o dataset alemão, escolhendo apenas os estudos que tenham obtido os seus resultados com o mesmo dataset e com as mesmas métricas, estes estudos são: Xia et al. [3] e Liu et al. [5].

Observando a tabela de comparação é possível concluir que os classificadores propostos pelos estudos relacionados apresentam um desempenho superior. O que poderá indicar que os classificadores propostos nesta tese poderão ter algum problema na fase de treino ou na fase de otimização de hiperparâmetros, ou que os classificadores proposto pela tese simplesmente tem pior desempenho que os classificadores dos estudos relacionados.

Os estudos utilizados na comparação, Xia et al. [3] e Liu et al. [5] apresentam investigações de conceitos originais de modelos e de planeamentos de modelação que vão muito além do abrangido

Tabela 9: Comparação do desempenho no dataset alemão.

| Estudo | Classificador | AUC | BS |
|---|---|---|---|
| Xia et al. [3] | OCHE | 0.803 | 0.158 |
| Liu et al. [5] | mg-GBDT | 0.793 | 0.160 |
| Tese | RLC-MV | 0.774 | 0.225 |
| Tese | XLC - WA | 0.764 | 0.236 |

pela tese. Obtendo assim resultados superiores. No entanto, retirando observações semelhantes, como o facto de que classificadores de *ensemble* heterogéneos, são em geral superiores aos classificadores homogéneos e individuais, em nível de desempenho.

### 5.2 Resumo dos resultados

Os resultados obtidos, estão de acordo com a maioria dos estudos abordados, na secção de revisão de literatura, afirmando que a utilização de classificadores *ensemble*, são superiores as técnicas comummente utilizadas nesta área, para a avaliação de riscos de crédito.

Com os resultados obtidos nesta tese é possível validar esta afirmação. A tese apresenta resultados semelhantes, aos de estudos como Barboza et al. [11], Dastile et al. [4] e Hamori et al. [10], que afirmam que métodos de *ensemble* apresentam resultados superiores aos métodos tradicionais como LR, LDA e ANN.

É de realçar, que as possibilidades existentes para o desenvolvimento de modelos *ensemble* heterogéneos são imensas. No desenvolvimento de um modelo heterogéneo é necessário ter em conta múltiplos fatores, como por exemplo: o número de modelos base a utilizar; o tipo de modelo base; a otimização de hiperparametros; estratégias de seleção de *ensemble*; estratégias de fusão *ensemble*, etc. Dito isto, o processo de desenvolvimento dos *ensemble* heterogéneos nesta tese, teve como base alguns estudos analisados e os resultados obtidos por modelos homogéneos.

Resumindo a tese afirma que, pelos resultados obtidos, a utilização de classificadores *ensemble* são superiores aos métodos de classificação individuais, que são comummente utilizados por instituições financeiras, indicando assim uma melhor opção da utilização de *machine learning* para a avaliação de crédito bancário.

# 6 Conclusão

Esta secção fornecerá, uma revisão dos objetivos da tese (**Secção 6.1**), informações sobre as limitações encontradas ao longo do desenvolvimento da tese (**Secção 6.2**), trabalho futuros (**Secção 6.3**) e as conclusões (**Secção 6.4**).

## 6.1 Objetivos

Relativamente aos objetivos da investigação colocadas na secção 1.2, foram obtidas as seguintes conclusões ao analisar os resultados e a metodologia.

(i) **[O1.] - Elaboração de um planeamento de modelação.**

Na secção da metodologia 3.1 foi desenvolvido um processo de planeamento de modelação que foi utilizado para coordenar todas as fases e processos necessários para a realização da parte experimental da tese, resultando num fluxograma representado na figura 11. Um dos objetivos desta tese era o desenvolvimento de um planeamento de modelação em que fosse possível alcançar os resultados esperados pela tese. Este objetivo foi alcançado devido a todo o conhecimento adquirido durante a investigação e pesquisa realizada pela tese no estado atual do conhecimento, onde baseando-se em múltiplos estudos relacionados e pesquisas, foi possível elaborar um fluxograma que engloba todas as fases necessárias para a implementação de modelos de ML, produzindo resultados capazes de serem comparados com estudos relacionados.

(ii) **[O2.] - Determinar quais os modelos de ML mais favoráveis para avaliação de crédito bancário.** Tendo em conta os resultados obtidos tanto pela tese como pelos estudos abordados, é aceitável afirmar que modelos de classificação de *ensemble* são superiores aos modelos comummente utilizados pelas entidades financeiras. Dito isto, a tese afirma que os modelos de ML mais favoráveis para avaliação de crédito são os modelos de classificadores heterogéneos, com estes modelos é possível combinar o desempenho e a flexibilidade de outros como por exemplo, XGBoost, CatBoost e LightGBM, formando assim classificadores muito mais equilibrados, complexos e versáteis.

A tese demonstra que é possível implementar modelos de *machine learning* para a avaliação de credito bancário, mais eficientes do que os usados como padrão pela indústria, cumprindo assim este objetivo.

## 6.2 Limitações

A existência de *datasets* desequilibrados é algo comum em problemas de classificação, onde existem vários métodos disponíveis para tentar equilibrar um *dataset*. A tese não aplicou nenhum destes

métodos aos seus *datasets*, considerando que, a escolha de um destes encontra-se fora do âmbito da tese, o que não deixa de ser uma possível limitação.

Apesar de que, no planeamento de modelação proposto pela tese, exista uma fase de otimização de hiperparâmetros onde foi utilizada a técnica de otimização bayesiana, baseada no teorema de Bayes, existem inúmeras técnicas de otimização, onde o uso de uma outra técnica poderia corresponder a uma alteração dos resultados.

No desenvolvimento de *ensemble* heterogéneos existe um enorme número de possibilidades a ter em conta, como por exemplo, o número de modelos a utilizar, quais os tipos de modelos, estratégias de seleção de *ensemble*, estratégias de fusão *ensemble*, etc. Logo a existência de melhores modelos de *ensemble* heterogéneos é uma realidade. Com o tempo disponibilizado, não foi possível a tese cobrir todas estas possibilidades.

## 6.3 Trabalho futuro

A tese tem alguns inconvenientes que podem ser melhorados em trabalhos futuros. Em primeiro lugar, a escolha de modelos base para os classificadores heterogéneos, em trabalhos futuros, seria interessante a investigação de outros modelos, pois a tese acredita que poderá levar a um aumento de desempenho e eficiência. Além disso, uma estratégia eficiente de fusão de *ensemble* é um campo importante de investigação para a construção de modelos de *ensemble* para avaliação de crédito.

A implementação de mecanismos como *early stopping* e *cost sensitive learning* poderão contribuir para um melhoramento dos resultados em trabalhos futuros, tendo em conta os resultados baixos obtidos por classificadores (AUC e sensibilidade) no *dataset* alemão, que poderá indicar um provável *overfitting* dos modelos. O que também poderá ser uma solução de como tratar o problema do desequilíbrio das classes em datasets.

Finalmente, deve ser dada especial atenção à compreensão da influencia das *features* no treino de modelos. Seria interessante um maior foco na implementação de *feature selection*, por exemplo, uma analise as correlações entre *features* de forma a ser possível concluir a existência de *features* redundantes e por sua vez contribuir para um melhoramento do desempenho dos modelo.

## 6.4 Conclusões

A classificação de crédito é um sistema de avaliação de risco de crédito para orientar a tomada de decisões para bancos e instituições financeiras. A sua importância tem suscitado grande atenção junto dos decisores financeiros na procura de tecnologia avançada para melhorar o desempenho da classificação de empréstimos e na obtenção de lucro. Devido ao seu desempenho superior, métodos de *ensemble* têm atraído muita atenção de investigadores na área de classificação de crédito.

Esta tese, baseou-se num planeamento de modelação de forma a conseguir delinear, as fases necessárias para chegar aos resultados pretendidos, as quais são, investigação de quais os *datasets* mais adequados, análise exploratória de dados, preprocessamento dos dados, seleção dos modelos a utilizar, otimização dos hiperparametros e por fim uma análise dos resultados.

O objetivo principal da tese é conseguir analisar os resultados obtidos, de forma a concluir quais os classificadores de ML mais relevantes contribuindo assim para o melhoramento do sistema de avaliação de crédito. Para tal, foram empregue diversos modelos, representando vários tipos de classificadores: Classificadores individuais, Classificadores de *ensemble* homogéneos e classificadores de *ensemble* heterogéneos.

Na análise realizada pela tese, foi possível concluir que os classificadores individuais, que são formados pelos modelos tradicionais mais comummente usados na avaliação de crédito, apresentam resultados inferiores aos dos classificadores de *ensemble*, o que demonstra o potencial deste tipo de classificadores em comparação aos usados como padrão nas instituições financeiras.

Classificadores de *ensemble* heterogéneos desenvolvidos e testados pela tese, ao serem comparados com os modelos tradicionais apresentaram resultados positivos apesar de ligeiros. A tese recomenda aos futuros estudos, a se focarem em classificadores de *ensemble*, visto que, ainda existem muitas possibilidades de melhoria, a serem explorados.

# Referências


[1] Hariom Tatsat, Sahil Puri, and Brad Lookabaugh. *Machine Learning and Data Science Blueprints for Finance*. O'Reilly Media, Inc., October 2020.

[2] Inteligência artificial vs. Machine Learning | Microsoft Azure.

[3] Yufei Xia, Junhao Zhao, Lingyun He, Yinguo Li, and Mengyi Niu. A novel tree-based dynamic heterogeneous ensemble method for credit scoring. *Expert Systems with Applications*, 159:113615, November 2020.

[4] Xolani Dastile, Turgay Celik, and Moshe Potsane. Statistical and machine learning models in credit scoring: A systematic literature survey. *Applied Soft Computing*, 91:106263, June 2020.

[5] Wanan Liu, Hong Fan, and Min Xia. Step-wise multi-grained augmented gradient boosting decision trees for credit scoring. *Engineering Applications of Artificial Intelligence*, 97:104036, January 2021.

[6] Swati Jadhav, Hongmei He, and Karl Jenkins. Information gain directed genetic algorithm wrapper feature selection for credit rating. *Applied Soft Computing*, 69:541–553, August 2018.

[7] Feng Shen, Xingchao Zhao, Gang Kou, and Fawaz E. Alsaadi. A new deep learning ensemble credit risk evaluation model with an improved synthetic minority oversampling technique. *Applied Soft Computing*, 98:106852, January 2021.

[8] Andro Merćep, Lovre Mrčela, Matija Birov, and Zvonko Kostanjčar. Deep Neural Networks for Behavioral Credit Rating. *Entropy*, 23(1):27, January 2021. Number: 1 Publisher: Multidisciplinary Digital Publishing Institute.

[9] Pedro Miguel Pinhal Pereira. Análise de risco de crédito usando algoritmos de Machine Learning. 2021. Accepted: 2021-05-21T14:31:03Z.

[10] Shigeyuki Hamori, Minami Kawai, Takahiro Kume, Yuji Murakami, and Chikara Watanabe. Ensemble Learning or Deep Learning? Application to Default Risk Analysis. *Journal of Risk and Financial Management*, 11(1):12, March 2018. Number: 1 Publisher: Multidisciplinary Digital Publishing Institute.

[11] Flavio Barboza, Herbert Kimura, and Edward Altman. Machine learning models and bankruptcy prediction. *Expert Systems with Applications*, 83:405–417, October 2017.

[12] Give Me Some Credit.

[13] UCI Machine Learning Repository.

[14] Jason Brownlee. SMOTE for Imbalanced Classification with Python, January 2020.

[15] Migran N. Gevorkyan, Anastasia V. Demidova, Tatiana S. Demidova, and Anton A. Sobolev. Review and comparative analysis of machine learning libraries for machine learning. *Discrete and Continuous Models and Applied Computational Science*, 27(4):305–315, December 2019.

[16] Giang Nguyen, Stefan Dlugolinsky, Martin Bobák, Viet Tran, Álvaro López García, Ignacio Heredia, Peter Malík, and Ladislav Hluchý. Machine Learning and Deep Learning frameworks and libraries for large-scale data mining: a survey. *Artificial Intelligence Review*, 52(1):77–124, June 2019.

[17] scikit-learn, June 2021. Page Version ID: 61397551.

[18] scikit-learn: machine learning in Python — scikit-learn 0.24.2 documentation.

[19] TensorFlow- Introdução ao TensorFlow.

[20] Introdução aos Tensores | TensorFlow Core.

[21] Matplotlib: Python plotting — Matplotlib 3.4.3 documentation.

[22] Visualization with Matplotlib | Python Data Science Handbook.

[23] Jason Brownlee. Supervised and Unsupervised Machine Learning Algorithms, March 2016.

[24] Stefan Lessmann, Bart Baesens, Hsin-Vonn Seow, and Lyn C. Thomas. Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research*, 247(1):124–136, November 2015.

[25] Jason Brownlee. How to Develop a Bagging Ensemble with Python, April 2020.

[26] Jason Brownlee. Essence of Boosting Ensembles for Machine Learning, May 2021.

[27] Gradient-boosted decision trees (GBDT).

[28] What is XGBoost? | data science | NVIDIA glossary.

[29] Pushkar Mandot. What is LightGBM, how to implement it? how to fine tune the parameters?

[30] Welcome to LightGBM's documentation! — LightGBM 3.3.2.99 documentation.

[31] CatBoost vs XGBoost and LighGBM: When to choose CatBoost?

[32] Jason Brownlee. How to Develop Voting Ensembles With Python, April 2020.

[33] Jason Brownlee. How to Develop a Weighted Average Ensemble With Python, May 2021.

[34] 14 Popular Machine Learning Evaluation Metrics, October 2020.

[35] Melissa Assel, Daniel D. Sjoberg, and Andrew J. Vickers. The Brier score does not evaluate the clinical utility of diagnostic tests or prediction models. *Diagnostic and Prognostic Research*, 1(1):19, December 2017.

[36] Mark S. Roulston. Performance targets and the Brier score. *Meteorological Applications*, 14(2):185–194, 2007. _eprint: https://rmets.onlinelibrary.wiley.com/doi/pdf/10.1002/met.21.

[37] Jason Brownlee. Why Data Preparation Is So Important in Machine Learning, June 2020.

[38] What is Exploratory Data Analysis? | IBM.

[39] Hyperparameter tuning for machine learning models., November 2017.

[40] Jason Brownlee. Scikit-Optimize for Hyperparameter Tuning in Machine Learning, September 2020.

[41] Jason Brownlee. A Gentle Introduction to k-fold Cross-Validation, May 2018.

[42] 3.1. Cross-validation: evaluating estimator performance — scikit-learn 0.24.2 documentation.

[43] Rahil Shaikh. Choosing the right Encoding method-Label vs OneHot Encoder, November 2018.

[44] Baijayanta Roy. All about Feature Scaling, April 2020.