MESTRADO EM ENGENHARIA E GESTÃO INDUSTRIAL

Projeto de Análise de Dados e Gestão do Conhecimento

AUTORAS

Bruna Neiva | **1220524@isep.ipp.pt**
Bárbara Falcão | **1210184@isep.ipp.pt**

DOCENTE
Maria de Fátima Coutinho Rodrigues

Unidade Curricular

Análise de Dados e Gestão do Conhecimento

3 de dezembro de 2023

1 Introdução

Este trabalho surge no âmbito da unidade curricular Análise de Dados e Gestão do Conhecimento inserido no mestrado em Engenharia e Gestão Industrial do Instituto Superior de Engenharia do Porto (ISEP) cujo objetivo passa por desenvolver modelos preditivos capazes de prever se os clientes de um banco entrarão em cumprimento ou não de um empréstimo. Para tal serão utilizados métodos e técnicas de *data mining* que tem como objetivo a extração de informações e padrões de comportamento de uma grande quantidade de dados.

2 Introdução Teórica

O processo de Data Mining é frequentemente utilizado para apoiar na tomada de decisões a nível empresarial e tem crescido consideravelmente nas últimas décadas. Esse crescimento é especialmente notável nos setores de serviços, como o setor da distribuição, onde o uso de Data Mining se tornou uma prática empresarial abrangente. De modo a garantir que os projetos de Data Mining entreguem consistentemente os resultados pretendidos, as empresas usam processos padronizados, como o CRISP-DM para gerir projetos de Data Mining.

O CRISP-DM é um modelo de processo independente que consiste em seis fases iterativas, que vão desde a compreensão do negócio até à implementação.

O CRISP-DM é um processo iterativo, o que significa que é possível retroceder e reavaliar fases anteriores se necessário tornando-se, assim, um método flexível. As fases que constituem o CRISP-DM são as seguintes:

1. **Compreensão do Negócio (*Business Understanding*):** O objetivo desta fase é compreender o objetivo do "negócio"assim como os requisitos do projeto. A fase de *Business Understanding* o envolve o entendimento do problema e dos objetivos do projeto. Requer a compreensão do contexto do negócio, do ambiente competitivo e da necessidade do cliente. Deve ser definido um plano inicial, que permita orientar todo o processo de análise. (Shearer, 2000)

2. Compreensão dos Dados (*Data Understanding*)

O principal propósito desta fase é a recolha e exploração de dados de modo a melhor entender as suas carateristicas e verificar a relevância dos mesmos para os objetivos do projeto. A fase de compreensão dos dados envolve a recolha de dados assim como a sua descrição e exploração. Esta fase ajuda a identificar problemas de qualidade dos dados e a entender a natureza dos dados. A partir da análise inicial do banco de dados é possível obter *insights* sobre os dados, formando hipóteses sobre as informações que não estão claras. (Witten, I. H., Frank, E., & Hall, M. A., 2016)

3. Preparação dos Dados (*Data Preparation*)

Nesta fase ocorre a preparação dos dados para a modelação dos mesmos. Esta fase consiste na limpeza de dados, verificação da existência de valores nulos, entre outros. É também na presente fase que são selecionadas as variáveis relevantes para a análise e transformados os dados conforme a necessidade.

De acordo com o artigo publicado em 2012 por Han, J., Kamber, M., & Pei, J., a fase de preparação dos dados envolve a seleção, limpeza, transformação e formatação dos dados. É a mais demorada e trabalhosa, mas é essencial para a obtenção de bons resultados.

4. Modelação (*Modeling*) O principal foco desta fase é a construção de modelos de *Data Mining*. Nesta etapa são selecionadas as técnicas de modelação que mais se adequam ao contexto em questão. A construção e treino dos modelos assim como a avaliação e validação do mesmo constituem esta fase.

Witten, I. H., Frank, E., & Hall, M. A., no seu livro publicado em 2016, afirmam que a modelação envolve a construção de modelos a partir dos dados preparados. A modelação é uma fase iterativa e envolve a experimentação com diferentes modelos e parâmetros. Experimentam-se várias técnicas de modelação e seleciona-se a que melhor se ajusta aos dados.

5. Avaliação (*Evaluation*)

Esta etapa consiste na avaliação do desempenho do nosso modelo.

A fase de avaliação envolve a avaliação dos modelos construídos na fase anterior. É importante ter em mente que a escolha do modelo depende do objetivo do projeto e da natureza dos dados (IBIDEM). A avaliação dos modelos é feita com base em vários critérios de desempenho, como a precisão, a sensibilidade e a especificidade.

6. Implementação (*Deployment*) A última etapa envolve a implementação do modelo.

A fase de implantação envolve a integração do modelo escolhido na solução de negócio e sua disponibilização para uso pelos utilizadores finais. Essa fase também envolve a monitorização contínua do modelo em questão. (IBIDEM)

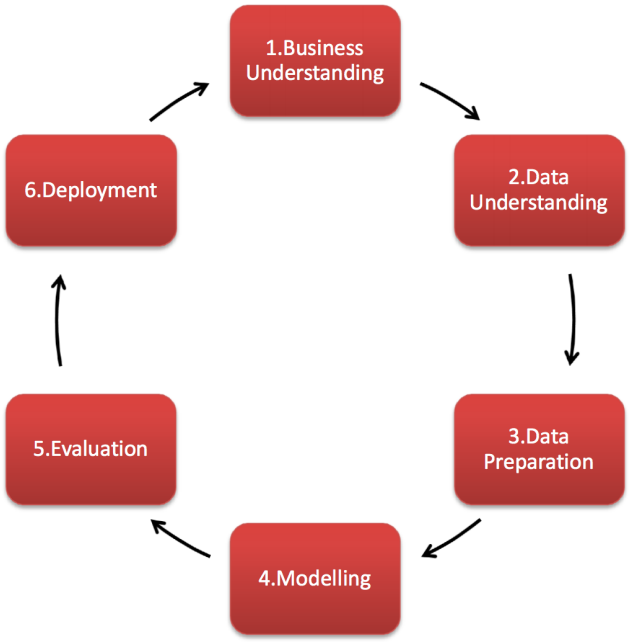

Figura 1: Etapas do CRISP-DM

3 Metodologia

3.1 *Business Understanding*

Nesta fase, o foco principal reside na compreensão aprofundada dos objetivos do negócio associados ao projeto de *Data Mining*. Neste caso em concreto, o objetivo consiste em desenvolver modelos capazes de prever se os clientes irão cumprir o pagamento dos seus empréstimos. O não pagamento de empréstimos pelos clientes resulta em perdas financeiras significativas para o banco, o que resultará em prejuízo no seu desempenho económico. Outros objetivos do trabalho consistem na análise de possíveis correlações entre as variáveis e que podem vir a gerar *insights* para o negócio.

3.2 *Data Understanding*

O objetivo do Data Understanding é, de modo geral, a compreensão da base de dados, identificando critérios que estudem a qualidade desses dados e procurem eliminar fatores que não são importantes para a análise. A partir da análise inicial do *dataset* é possível obter *insights* relacionados aos dados e gerar hipóteses relacionadas a informações não esclarecedoras do *dataset*. Os principais passos para realização do *data understanding* são a descrição, exploração e verificação da qualidade do *dataset*.

Primeiramente foi feita a recolha dos dados através da leitura do ficheiro facultado pela docente em formato csv. (Figura 2)

```
caminho_arquivo = '/Users/brunaneiva/Desktop/ANDOC/train.csv'
dados = pd.read_csv(caminho_arquivo)
```

Figura 2: Recolha dos dados

Seguiu-se a exploração dos dados extraindo informações como o nome das colunas (variáveis), o tipo de dados de cada variável, a quantidade de valores ausentes em cada variável e o número de linhas e colunas. A partir da Figura 3 conseguimos ter uma visão geral dos nossos dados como a quantidade de variáveis (35), quantidade de linhas (67463), a inexistência de *outliers* em todas as variáveis e o tipo de dados (9 variáveis do tipo float64, 17 do tipo int64 e 9 do tipo *object*.

```
13   print(dados.info())

PROBLEMS 14   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS

 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   ID                            67463 non-null  int64
 1   Loan Amount                   67463 non-null  int64
 2   Funded Amount                 67463 non-null  int64
 3   Funded Amount Investor        67463 non-null  float64
 4   Term                          67463 non-null  int64
 5   Batch Enrolled                67463 non-null  object
 6   Interest Rate                 67463 non-null  float64
 7   Grade                         67463 non-null  object
 8   Sub Grade                     67463 non-null  object
 9   Employment Duration           67463 non-null  object
 10  Home Ownership                67463 non-null  float64
 11  Verification Status           67463 non-null  object
 12  Payment Plan                  67463 non-null  object
 13  Loan Title                    67463 non-null  object
 14  Debit to Income               67463 non-null  float64
 15  Delinquency - two years       67463 non-null  int64
 16  Inquires - six months         67463 non-null  int64
 17  Open Account                  67463 non-null  int64
 18  Public Record                 67463 non-null  int64
 19  Revolving Balance             67463 non-null  int64
 20  Revolving Utilities           67463 non-null  float64
 21  Total Accounts                67463 non-null  int64
 22  Initial List Status           67463 non-null  object
 23  Total Received Interest       67463 non-null  float64
 24  Total Received Late Fee       67463 non-null  float64
 25  Recoveries                    67463 non-null  float64
 26  Collection Recovery Fee       67463 non-null  float64
 27  Collection 12 months Medical  67463 non-null  int64
 28  Application Type              67463 non-null  object
 29  Last week Pay                 67463 non-null  int64
 30  Accounts Delinquent           67463 non-null  int64
 31  Total Collection Amount       67463 non-null  int64
 32  Total Current Balance         67463 non-null  int64
 33  Total Revolving Credit Limit  67463 non-null  int64
 34  Loan Status                   67463 non-null  int64
dtypes: float64(9), int64(17), object(9)
memory usage: 18.0+ MB
```

Figura 3: Informações dos dados

Para as variavéis numericas (26) fez-se uma analise estatistica descritiva como é possível observar na figura 4.

```
15  estatisticas_descritivas = dados.describe()
16  print(estatisticas_descritivas)

PROBLEMS 15   OUTPUT   DEBUG CONSOLE   TERMINAL   PORTS                Python

(base) brunaneiva@Air-de-Bruna ~ % /usr/local/bin/python3 "/Users/brunaneiva/andoc 2.py"
                 ID   Loan Amount  Funded Amount  Funded Amount Investor  ...  Total Collection Amount  Total Current Balance  Total Revolving Credit Limit   Loan Status
count  6.746300e+04  67463.000000   67463.000000            67463.000000  ...             67463.000000           6.746300e+04                  67463.000000  67463.000000
mean   2.562761e+07  16048.902776   15770.599114            14621.799323  ...               146.467990           1.595739e+05                  23123.405544      0.892510
std    2.109155e+07   8367.865726    8150.992662             6785.345170  ...               744.382233           1.390332e+05                  20916.699999      0.289747
min    1.297933e+06   1014.000000    1014.000000             1114.590204  ...                 1.000000           6.170000e+02                   1000.000000      0.000000
25%    6.570288e+06  10012.000000    9266.500000             9831.684984  ...                24.000000           5.037500e+04                   8155.500000      0.000000
50%    1.791565e+07  16073.000000   13042.000000            12793.682170  ...                36.000000           1.183698e+05                  16733.000000      0.000000
75%    4.271521e+07  22106.000000   21793.000000            17807.594120  ...                46.000000           2.283750e+05                  32146.500000      0.000000
max    7.224578e+07  35000.000000   34999.000000            34999.746430  ...             16421.000000           1.177412e+06                 201169.000000      1.000000

[8 rows x 26 columns]
(base) brunaneiva@Air-de-Bruna ~ %
```

Figura 4: Estatísticas descritivas

Esta análise permitiu assim visualizar as principais caraterísticas das nossas variáveis nu-

méricas como a contagem das linhas, a média, o desvio padrão, os quartis, o valor minimo e máximo. Seguiu-se a verificação da existência de dados ausentes (novamente mas de forma isolada) visto que estes representam um entrave para os métodos de análise, aplicando a função *isnull().sum()*. Desta forma constatou-se (e confirmou-se)a inexistência de valores nulos.

Para melhor compreendermos as variáveis numéricas construiu-se 2 tipos de gráficos: histogramas e *boxplots*.

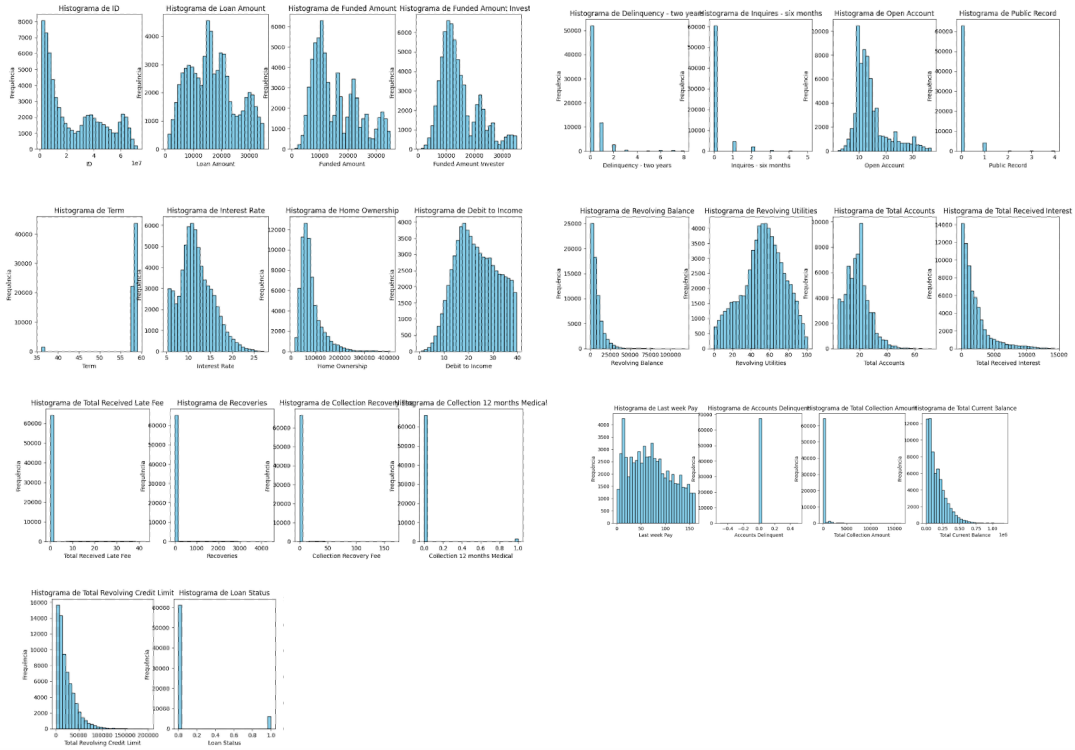

Figura 5: Histogramas Variáveis Numéricas

Através dos histogramas observamos que o histograma "Accounts Deliquent"apresenta apenas um valor(0) pelo que se trata de uma variável poouco relevante.

Na seguinte figura constam os *boxplots* das variáveis númericas uma vez que este tipo de gráfico fornece uma representação visual da distribuição estatística de um conjunto de dados. Deste modo constatamos a existência de outliers em várias variáveis (Funded Amount Investor, Term, Interest Rate, Home Ownership, Delinquency - two years, Inquires - six months, Open Account, Public Record, Revolving Balance, Total Accounts, Total Received Interest, Total Received Late Fee, Recoveries, Collection Recovery Fee, Collection 12 months Medical, Total Collection Amount, Total Current Balance, Total Revolving Credit Limit, Loan Status).

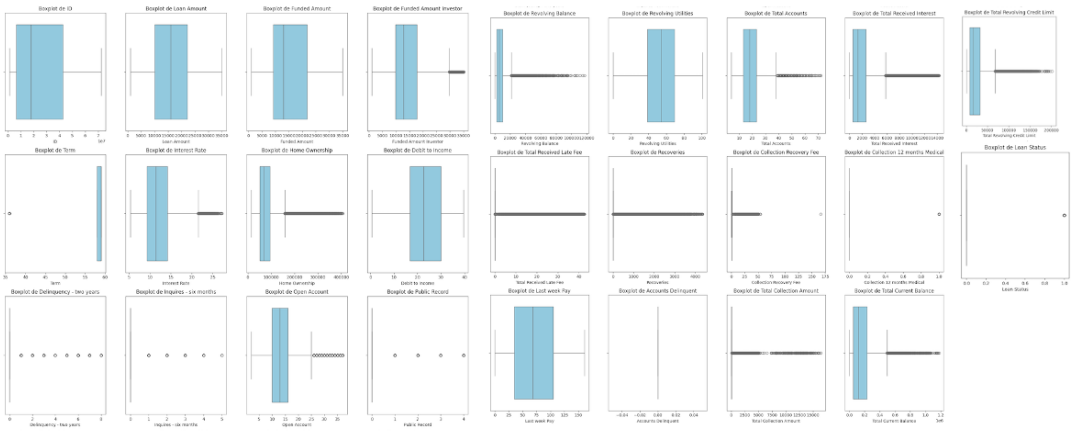

Figura 6: Boxplots das Variáveis Numéricas

De modo a verificar se as variáveis numéricas apresentavam alguma correlação entre si procedeu-se à implementação de uma matriz correlação, na qual se cosntatou-se que nenhuma das variáveis estão correlacionadas entre si.

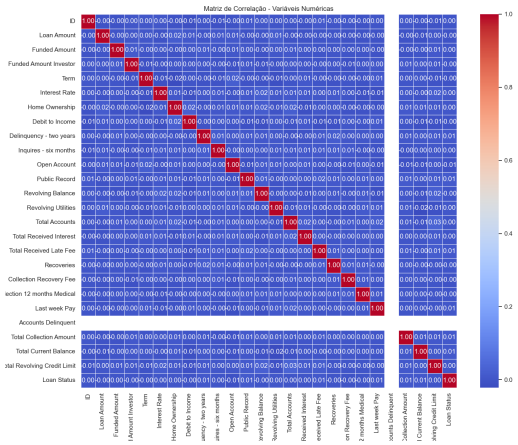

Figura 7: Matriz Correlação das variáveis numéricas

Já para a análise das variáveis categóricas utilizou-se o seguinte código(Figura 8) obtendo informações sobre cada uma delas.

```
41  # visualizar quais sao as variaveis categoricas
42  categoricas = dados.select_dtypes(include=['object']).columns
43  print(categoricas)
44  # visualizar quais as categorias de cada variavel categorica e quantas são
45  for coluna in categoricas:
46      print(f'Categorias em {coluna}:\n{dados[coluna].value_counts()}\n')
47      print(f'{coluna} tem {dados[coluna].nunique()} categorias únicas.')
```

Figura 8: Análise Variáveis Categóricas

A partir do código da figura supramencionada obteve-se a informação da quantidade de categorias que cada variável categórica contém assim como a descrição de cada uma. As variáveis categóricas são: *'Batch Enrolled', 'Grade', 'Sub Grade', 'Employment Duration','Verification Status', 'Payment Plan', 'Loan Title','Initial List Status', 'Application Type'*e contêm, respetivamente, 41, 7,35,3,3,1,109,2 e 2 categorias. Aplicando o código da figura 9 obteve-se informações sobre quais as categorias desta variável (Inidivual e Joint) e também quanto à sua distribuição pelas 2 categórias. Neste caso, 67340 aplicações são do tipo inidivual e as restantes 123 são aplicações conjuntas.

```
Categorias em Application Type:
Application Type
INDIVIDUAL    67340
JOINT           123
Name: count, dtype: int64

Application Type tem 2 categorias únicas.
```

Figura 9: Análise Variável Categórica:Application Type

Seguidamente, de modo a melhor perceber a distribuição das categorias fez-se um gráfico de barras para cada variável categórica como é possível observar na figura seguinte (Figura 10).

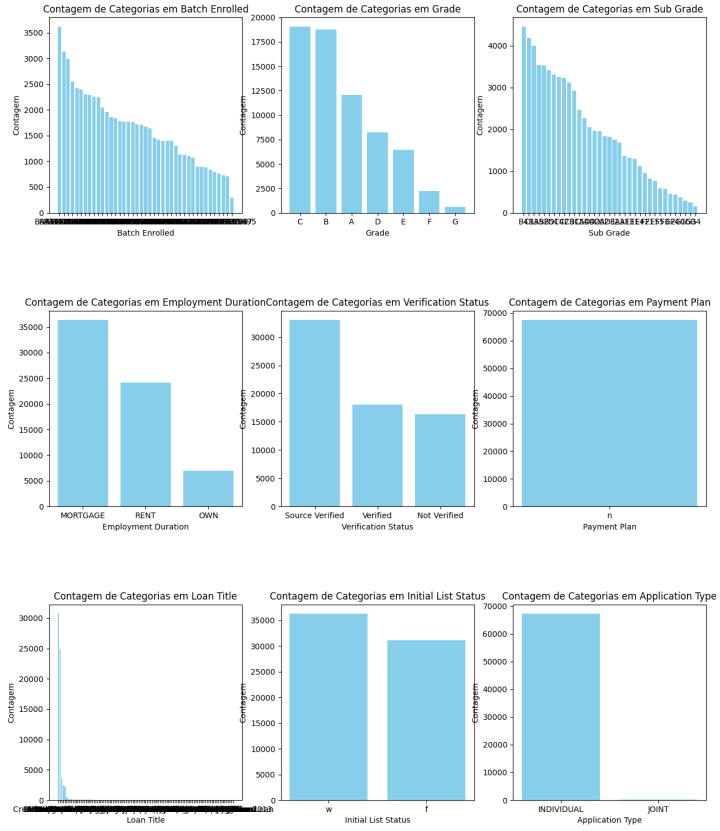

Figura 10: Gráfico de Barras das 9 Variáveis Categóricas

A partir da análise dos gráficos de barras é possível inferir que a variável "Payment Plan"tem apenas uma categoria e portanto não possui relevancia. Como existem 3 variavéis categóricas com uma grande quantidade de categorias foi necesário fazer um gráfico com o Top 10 das categorias.

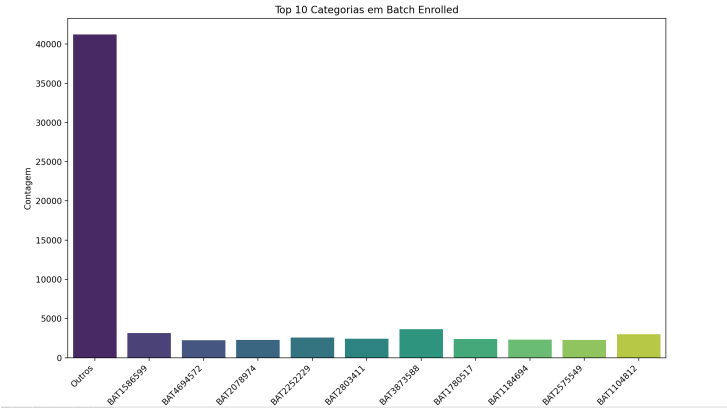

Figura 11: Gráfico de Barras da variável categórica Batch Enroled

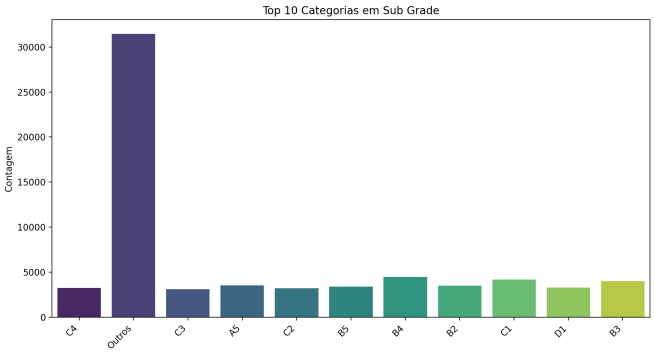

Figura 12: Gráfico de Barras da variavél categórica Sub Grade

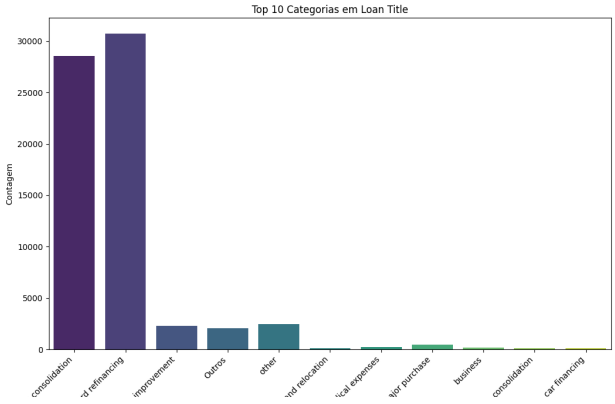

Figura 13: Gráfico de Barras da variável categórica Loan Title

A partir dos gráficos de barras com o Top 10 (Figura 11 e 12) verificou-se que as categorias das variáveis categóricas *Batch Enroled* e *Sub Grades* estão aproximadamente igualmente distribuidas pelas suas categorias porém para a variável categórica *Loan Title* (figura 13) sobresaiam 2 categórias: *credit card refinancing* e *debt consolidation*.

Para compreender a relação entre as várias variáveis categóricas procedeu-se ao teste Chi-Square verificando que existe uma associação significativa entre as variáveis:

```
Teste Qui-Quadrado entre Batch Enrolled e Grade: p-valor = 3.3569893462167676e-26
Teste Qui-Quadrado entre Batch Enrolled e Sub Grade: p-valor = 0.00063557584341905 37
Teste Qui-Quadrado entre Batch Enrolled e Employment Duration: p-valor = 4.2932240832829357e-38
Teste Qui-Quadrado entre Batch Enrolled e Verification Status: p-valor = 8.456454145526474e-29
Teste Qui-Quadrado entre Batch Enrolled e Payment Plan: p-valor = 1.0
Teste Qui-Quadrado entre Batch Enrolled e Loan Title: p-valor = 0.09682228064265154
Teste Qui-Quadrado entre Batch Enrolled e Initial List Status: p-valor = 1.9335664893218855e-17
Teste Qui-Quadrado entre Batch Enrolled e Application Type: p-valor = 0.5702875835269952
Teste Qui-Quadrado entre Grade e Sub Grade: p-valor = 2.9253049030321995e-61
Teste Qui-Quadrado entre Grade e Employment Duration: p-valor = 1.0771260954004472e-246
Teste Qui-Quadrado entre Grade e Verification Status: p-valor = 4.88046049545632e-86
Teste Qui-Quadrado entre Grade e Payment Plan: p-valor = 1.0
Teste Qui-Quadrado entre Grade e Loan Title: p-valor = 2.4523635870458066e-05
Teste Qui-Quadrado entre Grade e Initial List Status: p-valor = 2.1338005157450123e-128
Teste Qui-Quadrado entre Grade e Application Type: p-valor = 6.469629902523045e-06
Teste Qui-Quadrado entre Sub Grade e Employment Duration: p-valor = 4.5092354846125707e-41
Teste Qui-Quadrado entre Sub Grade e Verification Status: p-valor = 4.120181249822814e-51
Teste Qui-Quadrado entre Sub Grade e Payment Plan: p-valor = 1.0
Teste Qui-Quadrado entre Sub Grade e Loan Title: p-valor = 3.2112525438416183e-06
Teste Qui-Quadrado entre Sub Grade e Initial List Status: p-valor = 1.3342965569936317e-46
Teste Qui-Quadrado entre Sub Grade e Application Type: p-valor = 0.0863322996293686 7
Teste Qui-Quadrado entre Employment Duration e Verification Status: p-valor = 2.236016433530157e-193
Teste Qui-Quadrado entre Employment Duration e Payment Plan: p-valor = 1.0
Teste Qui-Quadrado entre Employment Duration e Loan Title: p-valor = 2.7054027595490687e-05
Teste Qui-Quadrado entre Employment Duration e Initial List Status: p-valor = 8.323252022864482e-123
Teste Qui-Quadrado entre Employment Duration e Application Type: p-valor = 0.028542902080085 79
Teste Qui-Quadrado entre Verification Status e Payment Plan: p-valor = 1.0
Teste Qui-Quadrado entre Verification Status e Loan Title: p-valor = 0.09556081276702803
Teste Qui-Quadrado entre Verification Status e Initial List Status: p-valor = 6.1820168928820325e-139
Teste Qui-Quadrado entre Verification Status e Application Type: p-valor = 0.2805132784326254
Teste Qui-Quadrado entre Payment Plan e Loan Title: p-valor = 1.0
Teste Qui-Quadrado entre Payment Plan e Initial List Status: p-valor = 1.0
Teste Qui-Quadrado entre Payment Plan e Application Type: p-valor = 1.0
Teste Qui-Quadrado entre Loan Title e Initial List Status: p-valor = 0.05765671972798365
Teste Qui-Quadrado entre Loan Title e Application Type: p-valor = 9.121488507071438e-09
Teste Qui-Quadrado entre Initial List Status e Application Type: p-valor = 0.1852074863279897
```

Figura 14: Teste Qui-Quadrado aplicado às variáveis categóricas

- Batch Enrolled e Grade
- Batch Enrolled e Sub Grade
- Batch Enrolled e Employment Duration
- Batch Enrolled e Verification Status
- Batch Enrolled e Initial List Status
- Grade e Sub Grade
- Grade e Employment Duration
- Grade e Verification Status
- Grade e Loan Title
- Grade e Initial List Status
- Grade e Application Type
- Sub Grade e Employment Duration
- Sub Grade e Verification Status
- Sub Grade e Loan Title
- Sub Grade e Initial List Status
- Employment Duration e Verification Status
- Employment Duration e Loan Title
- Employment Duration e Initial List Status
- Employment Duration e Application Type
- Verification Status e Initial List Status
- Loan Title e Application Type

Uma vez que quando é realizado um grande número de testes estatísticas há um aumento do risco de erro do tipo I isto é, a rejeição incorreta da hipótese nula, aplicou-se o método de Bonferroni de modo a mitigar este problema ajustando o nível de significância dividindo-o pelo número de testes. Após a aplicação do método referido houve uma relação que outrora tinha uma associação significativa e passou a não ter associação significativa (Employment Duration e Application Type).

3.3 *Data Preparation*

Após analise dos dados, surge a fase de preparação dos dados para a modelação. Uma vez que a base de dados não possui valores ausentes não foi necessário corrigir. Nesta etapa serão retiradas do conjunto de dados variáveis que não influenciam a análise de dados e portanto são irrelavantes. Retirou-se as seguintes variáveis: *ID, Account Delinquent e Playment Plan* . Seguidamente procedeu-se à normalização dos dados recorrendo às funções *StandardScaler* (variáveis contínuas) exceto a variável dependente (*Loan Status*) e *LabelEncoder* (variáveis categóricas).

3.4 *Modeling*

Nesta etapa serão demonstrados os modelos implementados:Regressão Logística, Árvores de decisão e floresta aleatória, Gradient Boosting, Máquinas de Vetores de Suporte(SVM), Redes Neurais e Naive Bayes.

```python
# Separar as features (X) e a variável alvo (y)
X = dados.drop('Loan Status', axis=1)
y = dados['Loan Status']

# Dividir os dados em conjuntos de treinamento e teste
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Certificar se os rótulos são inteiros
y_train = y_train.astype(int)
y_test = y_test.astype(int)

# Modelos
modelos = {
    'Regressão Logística': LogisticRegression(solver='liblinear'),
    'Árvore de Decisão': DecisionTreeClassifier(),
    'Floresta Aleatória': RandomForestClassifier(),
    'Gradient Boosting': GradientBoostingClassifier(),
    'Máquina de Vetores de Suporte (SVM)': SVC(probability=True),
    'Redes Neurais': MLPClassifier(),
    'Naive Bayes': GaussianNB()
}

# Treinar e avaliar cada modelo
for nome, modelo in modelos.items():
    print(f"\n{nome}:")
    modelo.fit(X_train, y_train)
    previsoes = modelo.predict(X_test)
    print("Matriz de Confusão:")
    print(confusion_matrix(y_test, previsoes))
    print("\nRelatório de Classificação:")
    print(classification_report(y_test, previsoes, zero_division=1))
```

Figura 15: Código implementação Modelos

3.5 *Evaluation*

1.Regressão Logística

```
Regressão Logística:
Matriz de Confusão:
[[12275     0]
 [ 1218     0]]

Relatório de Classificação:
              precision    recall  f1-score   support

           0       0.91      1.00      0.95     12275
           1       1.00      0.00      0.00      1218

    accuracy                           0.91     13493
   macro avg       0.95      0.50      0.48     13493
weighted avg       0.92      0.91      0.87     13493
```

Figura 16: Resultados obtidos pela Regressão Logística

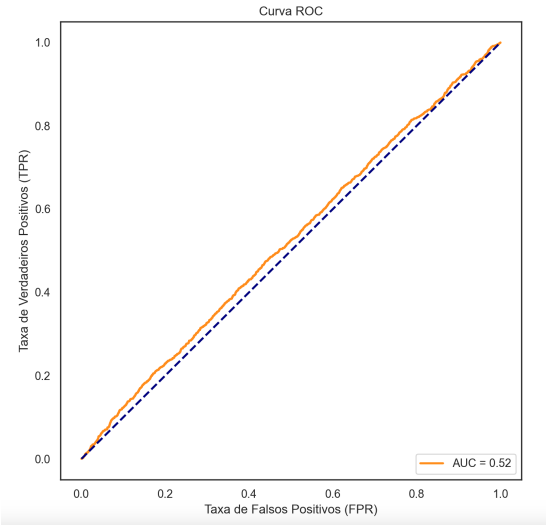

Figura 17: Curva ROC

A Matriz de Confusão devolveu: 0 Verdadeiros Positivos (TP); 12275 Verdadeiros Negativos (TN); 0 Falsos Positivos (FP) e 1218 Falsos Negativos. Relativamente ao relatório de classificação este dá-nos a acurácia global do modelo que é 91% o que à partida parece ser um bom desempenho.A classe 0 (cumprimento do empréstimo) possui uma precisão de 0,91 o que significa que das instâncias previstas como "cumprimento de empréstimo", 91% dos empréstimos foram efetivamente pagos. Em contrapartida a classe 1 (incumprimento do empréstimo)

possui um *recall* de 0.00 o que indica uma falha no modelo na identificação de instâncias reais de "incumprimento do empréstimo". Todas as instâncias desta classe foram mal classificadas como classe 0.O F1-score para a classe 1 é 0.00 o que denota um baixo desempenho geral para esta classe. A *macro avg e weighted avg* para *precision, recall e f1-score* indicam uma média das métricas para ambas as classes.A *macro avg* atribui o mesmo peso a todas as classes, enquanto que a *weighted avg* pondera as classes com base no *support*.

Para complementar a análise obteve-se a Curva ROC e parâmetro AUC que são frequentemente utilizadas para avaliar o desempenho de modelos de classificação. Uma AUC de 0,52 sugere um desempenho muito próximo ao aleatório o que pode significar que o modelo não está a fazer bem a distinção entre classes.

2. Árvores de decisão

```
Árvore de Decisão:
Matriz de Confusão:
[[10888  1387]
 [ 1065   153]]

Relatório de Classificação:
              precision    recall  f1-score   support

           0       0.91      0.89      0.90     12275
           1       0.10      0.13      0.11      1218

    accuracy                           0.82     13493
   macro avg       0.51      0.51      0.50     13493
weighted avg       0.84      0.82      0.83     13493
```

Figura 18: Resultados árvore de decisão

A Matriz de Confusão devolveu: 153 Verdadeiros Positivos (TP); 10888 Verdadeiros Negativos (TN); 1387 Falsos Positivos (FP) e 1065Falsos Negativos. O modelo apresenta uma precisão razoável para a classe 0 o que indica que quando faz previsões para a classe 0, essas previsões são na sua maioritariamente acertivas. No entanto, a precisão para a classe 1 é baixa o que se traduz numa previsão errada muitas das vezes. O *recall* para a classe 1 é baixo o que é indicador de que o modelo não consegue identificar corretamente muitas instâncias dessa classe. A acurácia geral do modelo é razoável porém deve ser tido em considerado a desporporção presente nas classes. Em suma este modelo necessita de ser ajustado de modo a melhorar o seu desempenho especialmente no que diz respeito ao *recall* e precisão para a classe 1. A Curva

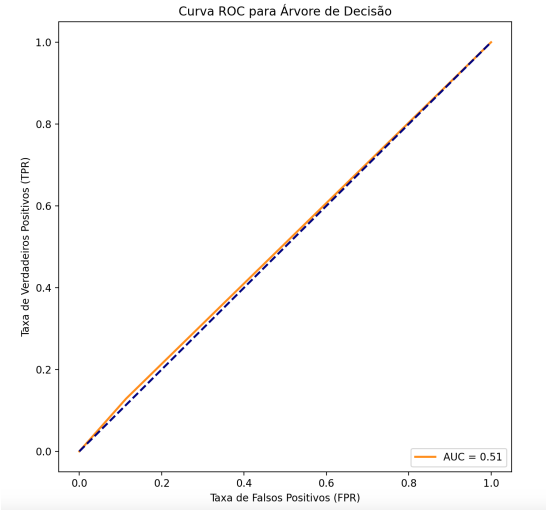

Figura 19: Curva ROC referente à árvore de decisão

ROC e a métrica AUC(0,51) vêm, uma vez mais, reforçar a necessidade de ajuste deste modelo pelo que o seu desempenho está muito próximo ao aleatório.

3. Floresta Aleatória

```
Floresta Aleatória:
Matriz de Confusão:
[[12275     0]
 [ 1218     0]]

Relatório de Classificação:
              precision    recall  f1-score   support

           0       0.91      1.00      0.95     12275
           1       1.00      0.00      0.00      1218

    accuracy                           0.91     13493
   macro avg       0.95      0.50      0.48     13493
weighted avg       0.92      0.91      0.87     13493
```

Figura 20: Resultados floresta aleatória

A Matriz de Confusão devolveu: 0 Verdadeiros Positivos (TP); 12275 Verdadeiros Negativos (TN); 0 Falsos Positivos (FP) e 1218 Falsos Negativos. A classe 0 apresenta um excelente desempenho por apresentar uma precisão de 0,91, um recall de 1,00 e um f1-score de 0,95, o que indica que o modelo está a prever bem a classe 0 quando esta ocorre. Em contrapartida, a classe 1 não está a ser considerada pelo modelo pois os parâmetros são todos 0. Uma razão plau-

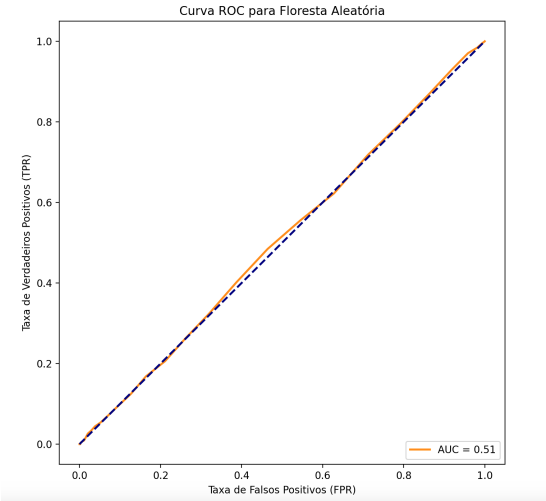

Figura 21: Curva ROC floresta aleatória

sivel para tal ocorrer é a desproporção entre as classes. A acurácia geral do modelo (0,91) pode originar conclusões precipitadas pois está a ser fortemente influenciada pela predominância da classe 0 porém não está a ter em consideração a classe 1, pelo que não é possível afirmar que o modelo apresenta bom desempenho e uma vez mais a Curva ROC e a AUC(0,51) confirmam este cenário.

4. Gradiente Boosting

```
Gradient Boosting:
Matriz de Confusão:
[[12272     3]
 [ 1218     0]]

Relatório de Classificação:
              precision    recall  f1-score   support

           0       0.91      1.00      0.95     12275
           1       0.00      0.00      1.00      1218

    accuracy                           0.91     13493
   macro avg       0.45      0.50      0.98     13493
weighted avg       0.83      0.91      0.96     13493
```

Figura 22: Resultados Gradient Boosting

A Matriz de Confusão devolveu: 0 Verdadeiros Positivos (TP); 12272 Verdadeiros Negativos (TN); 3 Falsos Positivos (FP) e 1218 Falsos Negativos.Assim como a floresta aleatória, o modelo Gradient Boosting apresenta um desempenho excelente na classe 0. À semelhança do

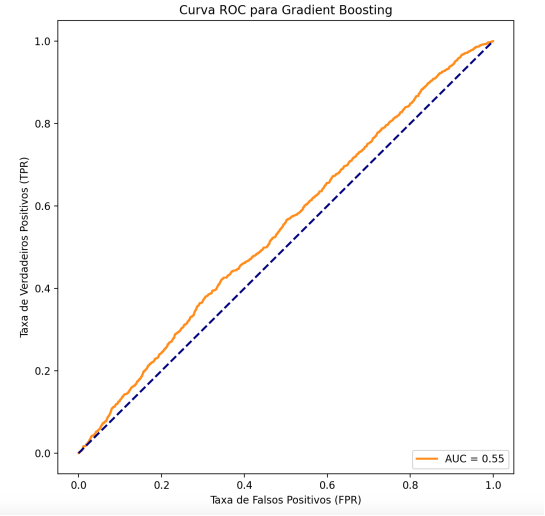

Figura 23: Curva ROC Gradient Boost

que acontece na floresta aleatória a classe 1 é desconsiderada pelo modelo tendo por essa razão todos os parametros a 0. Uma vez mais é realçado o desiquilibrio entre classes. A acurácia geral (0.91) pode também originar equivocos, pois não reflete a capacidade do modelo em identificar a classe minoritária. Uma vez que acurácia do modelo não corresponde de todo ao desmepenho do modelo baseamo-nos na curva ROC e o AUC para avaliar o modelo obtendo um AUC de 0,55 que sugere também um desempenho próximo ao aleatório sendo porém o modelo que obteve maior valor de AUC até ao momento.

5. Máquinas de Vetores de Suporte(SVM)

```
Máquina de Vetores de Suporte (SVM):
Matriz de Confusão:
[[12275     0]
 [ 1218     0]]

Relatório de Classificação:
              precision    recall  f1-score   support

           0       0.91      1.00      0.95     12275
           1       1.00      0.00      0.00      1218

    accuracy                           0.91     13493
   macro avg       0.95      0.50      0.48     13493
weighted avg       0.92      0.91      0.87     13493
```

Figura 24: Resultados SVM

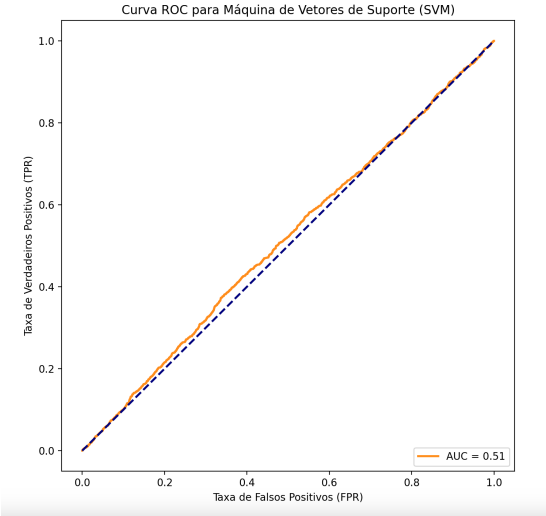

Figura 25: Curva ROC SVM

A Matriz de Confusão devolveu: 0 Verdadeiros Positivos (TP); 12275 Verdadeiros Negativos (TN); 0 Falsos Positivos (FP) e 1218 Falsos Negativos. O modelo apresenta uma precisão razoável para a classe 0 o que indica que quando faz previsões para a classe 0, essas previsões são na sua maioritariamente acertivas. No entanto, a precisão para a classe 1 é baixa o que se traduz numa previsão errada muitas das vezes. O *recall* para a classe 1 é baixo o que é indicador de que o modelo não consegue identificar corretamente muitas instâncias dessa classe. A acurácia geral do modelo é razoável porém deve ser tido em considerado a desporporção presente nas

classes. Em suma este modelo necessita de ser ajustado de modo a melhorar o seu desempenho especialmente no que diz respeito ao *recall* e precisão para a classe 1. A Curva ROC e a métrica AUC(0,51) vêm, uma vez mais, reforçar a necessidade de ajuste deste modelo pelo que o seu desempenho está muito próximo ao aleatório.

6. Redes Neurais

```
Redes Neurais:
Matriz de Confusão:
[[12146   129]
 [ 1207    11]]

Relatório de Classificação:
              precision    recall  f1-score   support

           0       0.91      0.99      0.95     12275
           1       0.08      0.01      0.02      1218

    accuracy                           0.90     13493
   macro avg       0.49      0.50      0.48     13493
weighted avg       0.83      0.90      0.86     13493
```

Figura 26: Resultados Redes Neurais

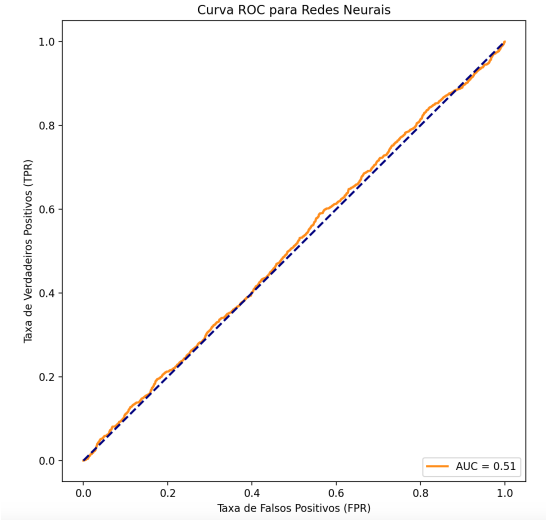

Figura 27: Curva ROC Redes Neurais

A Matriz de Confusão devolveu: 11 Verdadeiros Positivos (TP); 12146 Verdadeiros Negativos (TN); 129 Falsos Positivos (FP) e 1207 Falsos Negativos. Relativamente à precisão esta é consideravelmente boa para a classe 0 em contrapartida a classe 1 possui um *recall* de 0.00 o

que indica uma falha no modelo na identificação de instâncias reais de "incumprimento do empréstimo". Todas as instâncias desta classe foram mal classificadas como classe 0.O F1-score para a classe 1 é 2% o que denota um fraco desempenho geral para esta classe, no entanto para a classe 1 apresenta um valor de 95% o que é bastante satisfatório. A acurácia geral do modelo é 90% o que uma vez mais nos leva a pensar ser um modelo com bom desempenho pelo facto desta métrica estar fortemente influenciada pela predominância da classe 0. O melhor critério para avaliar então o desempenho do modelo será a conjugação da curva roc, o parâmetro AUC que neste caso é 0,51, e a *precisão, recall e f1-score.*. Todos os parâmetros anteriormente referidos sugererm que o desempenho do modelo está muito próximo ao aleatório o que se pode traduzir na incapacidade do modelo em fazer distinção entre classes.

7. Naive Bayes

```
Naive Bayes:
Matriz de Confusão:
[[11667   608]
 [ 1145    73]]

Relatório de Classificação:
              precision    recall  f1-score   support

           0       0.91      0.95      0.93     12275
           1       0.11      0.06      0.08      1218

    accuracy                           0.87     13493
   macro avg       0.51      0.51      0.50     13493
weighted avg       0.84      0.87      0.85     13493
```

Figura 28: Resultados Naive Bayes

A Matriz de Confusão devolveu: 73 Verdadeiros Positivos (TP); 11667 Verdadeiros Negativos (TN); 608 Falsos Positivos (FP) e 1145 Falsos Negativos. Quanto à precisão para a classe 0 foi uma vez mais boa porém para a classe 1 não apresenta um valor satisfatório apesar de ligeiramente superior relativamente aos restantes modelos. Quanto à *recall* é bastante satisfatória para a classe 0 e novamente para a classe 1 apesar de superior aos outros modelos é praticamente nula. O mesmo cenário se repete para o F1-score o que seria de esperar pois este parâmetro é a média da precisão e *recall*. A acurácia do modelo é de 87% o que mais uma vez induz em erro pois o modelo apenas apresenta um desempenho razoável para a classe 0, com boa precisão e *recall*. Relativamente à classe 0, o modelo apresenta um desempenho significativamente inferior com valores bastante baixos de precisão, *recall e f1-score*. Concluimos novamente que

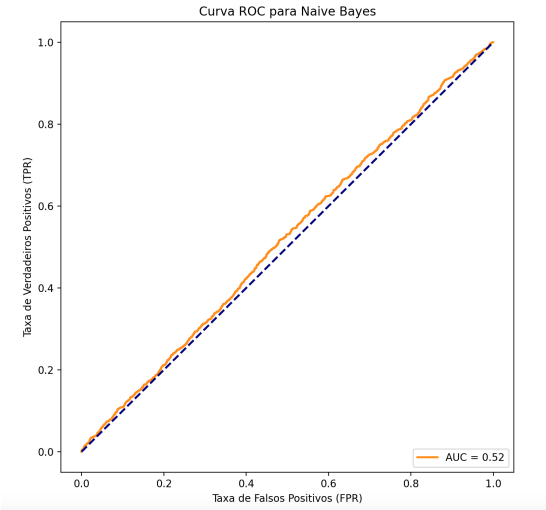

Figura 29: Curva ROC Naive Bayes

o que pode estar a infuenciar negativamente o desempenho global do modelo possa ser o forte desiquilibrio entre classes. A Curva ROC e o parâmetro AUC parecem ser as métricas indicadas para a avaliação do modelo e pelo gráfico e o valor obtido de AUC (0,52) o modelo apresenta um desempenho bastante próximo do aleatório retratando o problema da ausência de distinção entre as classes.

Para ter uma visão consolidada dos diferentes modelos construiu-se uma tabela com o todas as métricas de avaliação utilizadas para analisar o desempenho do modelo. Ao nível da acurácia todos os modelos apresentam um bom desempenho assim como a precisão para a classe 0 (classe maioritária). O fraco desempenho dos modelos começa a refletir-se na precisão da classe 1 assim como o *recall e o f1-score* correspondentes a esta classe. A recall e a precisão para a classe 0 é bastante satisfatória rondando, em média, os 100% e os 95% respetivamente. O AUC dos modelos rondam a casa dos 0,5-0,6 destacando-se o modelo *Gradient Boosting* que possui o maior valor de AUC (0,55) o que ainda assim é um valor bastante próximo do aleatório reforçado a necessidade de ajuste dos modelos. A regressão logística, o Gradient Boosting e a SVM apresentam uma precisão elevada mas uma recall nula para a classe minoritária. A árvore de decisão e a floresta aleatória proporcionam um desempenho mais equilibrado, mas ainda há margem para melhorias, especialmente para a classe minoritária. As redes neuronais

parecem encontrar um equilíbrio com a capacidade de identificar algumas instâncias da classe minoritária. Naive Bayes tem uma melhor *recall* para a classe minoritária. Dado o desequilíbrio na distribuição das classes, os modelos que alcançam um melhor equilíbrio entre a precisão e a *recall* para a classe 1 devem ser considerados mais eficazes para este conjunto de dados. As estratégias para melhorar estes modelos podem incluir o ajuste da dimensão das classes ou a reestruturação do *data set*.

Parâmetros	Regressão Logística	Decision Tree	Random Forest	Gradient Boosting	SVM	Redes Neurais	Naive Bayes
Accuracy	91%	82%	91%	91%	91%	90%	87%%
Precision							
classe 0	91%	91%	91%	91%	91%	91%	91%
classe 1	100%	10%	100%	0%	100%	8%	11%
Recall							
classe 0	100%	89%	100%	100%	100%	99%	95%
classe 1	0%	13%	0%	0%	0%	1%	6%
F1 score							
classe 0	95%	90%	95%	95%	95%	95%	93%
classe 1	0%	11%	0%	100%	0%	2%	8%
TP	0	153	0	0	0	11	73
TN	12275	10888	12275	12272	12275	12146	11667
FP	0	1387	0	3	0	129	608
FN	1218	1065	1218	1218	1218	1207	1145
AUC	0,52	0,51	0,51	0,55	0,51	0,51	0,52

Figura 30: Tabela comparativa de todos os modelos

3.6 *Conclusão*

O presente relatório apresenta uma análise de dados de um banco contendo um grande número de variáveis que vão influenciar o pagamento ou não do empréstimo. O principal objetivo cingiu-se no desenvolvimento de um modelo capaz de fazer previsões a cerca da liquidação do empréstimo. Ao nível de interpretação de dados foi possível verificar que maior parte das variáveis apresenta outliers na sua composição, o que de certa forma não torna os resultados 100% credíveis. A matriz de correlação demonstrou-nos que as variáveis numéricas não estão correlacionadas entre si. Quanto aos modelos e à sua avaliação, todos os modelos são semelhantes isto é, apresentam uma boa capacidade preditiva em relação à classe 0 porém desconsideram por completo a classe 1 o que gera uma falsa acurácia alta pois está fortemente influenciada pela predominância da classe 0. Deste modo recorremos à curva ROC e ao parâmetro AUC que são mais fidedignos quando há a existência de desiquilibrios entre classes. Estes comprovaram que todos os modelos por apresentarem valores a rondar os 0,5 possuem um desempenho muito próximo ao aleatório significando que o modelo não está a fazer a distinção entre classes. Em

suma, para que o desempenho do modelo melhore significativamente é crucial a identificação da classe minoritária (classe 1). Uma vez que a acurácia neste caso em particular pode não traduzir exatamente o desempenho do modelo pelos motivos já referidos, o AUC e o F1-score serão parâmetros que realmente traduzem o desempenho do sistema,ou seja, este está longe do desejado. Uma sugestão para melhorar o desempenho do modelo pode passar pelo ajuste dos hiperparâmetros do modelo especialmente em relação à classe 1. De modo a tentar entender o que pudesse estar na origem deste problema eliminou-se parte das variáveis mantendo apenas as seguintes: 'Loan Status', 'Loan Amount', 'Funded Amount', 'Interest Rate', 'Term','Employment Duration', 'Debit to Income', 'Delinquency - two years', 'Open Account', 'Revolving Balance', 'Total Accounts', 'Last week Pay'. Seria de esperar que o desempenho do modelo melhorasse pois muitas das variáveis foram eliminadas mas os resultados foram praticamente inalterados pelo que não será este fator que esta na origem do problema. Provavelmente deveriamos aplicar técnicas de balanceamente de classe ou experimentar modelos mais avançados de modo a melhorar o desempenho. Concluindo, o modelo que melhor se adapta à nossa base de dados será o *Gradient Boosting* apesar de não apresentar diferenças significativas em relação aos restantes.

3.7 *Referências*

Shearer, C. (2000). The CRISP-DM model: the new blueprint for data mining. Journal of data warehousing, 5(4), 13-22.

Data Mining: Practical Machine Learning Tools and Techniques", de Witten, I. H., Frank, E., Hall, M. A

Cross-Industry Standard Process for Data Mining (CRISP-DM): Towards a Standard Process for Data Mining", de Shearer, C.