UNIVERSIDADE FEDERAL DO MARANHÃO
Programa de Pós-Graduação em Ciência da Computação

Daniel Moreira Guilhon

Classificação de Risco em Transferências Voluntárias Federais Utilizando XGBoost

São Luís - MA
2020

Daniel Moreira Guilhon

Classificação de Risco em Transferências Voluntárias Federais Utilizando XGBoost

Dissertação apresentada como requisito parcial para obtenção do título de Mestre em Ciência da Computação, ao Programa de Pós-Graduação em Ciência da Computação, da Universidade Federal do Maranhão.

Universidade Federal do Maranhão

Centro de Ciências Exatas e Tecnologia

Programa de Pós-Graduação em Ciência da Computação

Orientador: Prof. Dr. Anselmo Cardoso Paiva

Coorientador: Prof. Dr. Daniel Lima Gomes Júnior

São Luís - MA

2020

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).
Núcleo Integrado de Bibliotecas/UFMA

Guilhon, Daniel Moreira.
Classificação de risco em transferências voluntárias federais utilizando XGBoost / Daniel Moreira Guilhon. - 2020.
84 p.

Coorientador(a): Daniel Lima Gomes Júnior.
Orientador(a): Anselmo Cardoso de Paiva.
Dissertação (Mestrado) - Programa de Pós-graduação em Ciência da Computação/ccet, Universidade Federal do Maranhão, São Luis, 2020.

1. Aprendizagem computacional. 2. Predição de risco. 3. Transferências voluntárias. 4. XGBoost. I. Gomes Júnior, Daniel Lima. II. Paiva, Anselmo Cardoso de. III. Título.

Daniel Moreira Guilhon

Classificação de Risco em Transferências Voluntárias Federais Utilizando XGBoost

Dissertação apresentada como requisito parcial para obtenção do título de Mestre em Ciência da Computação, ao Programa de Pós-Graduação em Ciência da Computação, da Universidade Federal do Maranhão.

Trabalho Aprovado. São Luís - MA, 16 de julho de 2020:

Prof. Dr. Anselmo Cardoso Paiva
Orientador
Universidade Federal do Maranhão

Prof. Dr. Daniel Lima Gomes Júnior
Coorientador
Instituto Federal do Maranhão

Prof. Dr. Geraldo Braz Júnior
Examinador Interno
Universidade Federal do Maranhão

Prof. Dr. Cláudio de Souza Baptista
Examinador Externo
Universidade Federal de Campina Grande

São Luís - MA
2020

Aos meus pais, esposa e filhos
que, com muita paciência e compreensão, me apoiaram para que eu pudesse realizar esta conquista.

Agradecimentos

O mundo está transformado. Todos fomos surpreendidos com uma ameaça invisível, que nos forçou a um isolamento social, onde um toque, um carinho, um abraço, ficou limitado a poucas pessoas. Mas essas poucas pessoas me deram as forças necessárias para esta empreitada que, neste momento de pandemia, foi uma jornada de autoconhecimento.

Primeiramente, agradeço a Deus, que manteve minha garra nesse momento tão difícil, e me guiou até o fim do caminho.

Aos meus pais, Afonso e Virgínia, por terem me ensinado os valores que são os pilares de um bom caráter, como honestidade e perseverança.

À minha amada esposa Vivian, pela paciência e companheirismo, e pela compreensão pelas horas de necessária ausência.

Aos meus filhos, André e Maria Luisa, que sempre me deram motivação e para quem sempre dediquei parte de todas as minhas conquistas.

Ao meu orientador Anselmo Paiva, pelas discussões e direcionamentos nos momentos mais difíceis.

Ao meu coorientador Daniel Lima, pelo acompanhamento e incentivo constantes.

Aos professores do curso de Pós-Graduação em Ciência da Computação da Universidade Federal do Maranhão, que tiveram a disposição de compartilhar um pouco de seus conhecimentos.

À Universidade Federal do Maranhão, que fez parte da minha formação e me acolheu novamente para esta nova jornada.

Por fim, a todos que contribuíram, direta ou indiretamente, para a concretização dos esforços deste trabalho.

"Não é porque certas coisas são difíceis que nós não ousamos; é justamente porque não ousamos que tais coisas são difíceis."

Sêneca

Resumo

Com a redemocratização no Brasil, estados e municípios passaram a contar com transferências voluntárias de recursos por parte do Governo Federal para a consecução de suas políticas públicas. Para uma maior tempestividade na recuperação de recursos eventualmente gastos de forma inadequada, é necessária uma ferramenta de classificação para atribuir perfis de risco de sucesso ou fracasso dessas transferências. Neste trabalho, propomos o uso do algoritmo *eXtreme Gradient Boosting* (XGBoost) usando conjuntos de dados balanceados e desbalanceados, com técnicas de otimização de hiperparâmetros *Tree-structured Parzen Estimator* bayesiano (TPE). Os resultados alcançaram boas taxas de sucesso. Os resultados do *XGBoost* mostraram uma taxa de sensibilidade usando dados balanceados de 89,3% e dados desbalanceados 87,8%. No entanto, para os dados desbalanceados, a AUC foi de 98,1%, contra 97,9% para os dados balanceados. Incorporar dados como informações acerca do objeto pactuado utilizando-se técnicas de processamento de linguagem natural pode melhorar os resultados obtidos.

Palavras-chave: transferências voluntárias, aprendizagem computacional, *XGBoost*, predição de risco

Abstract

After the Brazilian re-democratization, states and municipalities had to rely on federal government's voluntary transfers of resources to achieve their public policies. For greater timeliness in the recovery of resources that may have been spent inappropriately, it is necessary to assign risk profiles of success or failure of these transfers. In this work, we propose a methodology that uses eXtreme Gradient Boosting (XGBoost) algorithm, using balanced and unbalanced data sets, with the use of hyperparameter optimization techniques, such as Tree-structured Parzen Bayesian Estimator (TPE). The results achieved good success rates. Results for XGBoost using balanced data showed a recall of 89.3% and unbalanced data a recall of 87.8%. However, for unbalanced data, the AUC score was 98.1%, against 97.9% for balanced data. Incorporating information data about the agreed object using natural language processing techniques can improve the results obtained.

Keywords: Voluntary Transfers; Machine Learning; XGBoost; Risk Prediction;

Lista de ilustrações

Figura 1 – Volume financeiro disponibilizado pela União, por meio de transferências voluntárias 18
Figura 2 – Fluxo anual de prestação de contas - entradas x estoque x saídas . . . 19
Figura 3 – Visão geral dos passos do processo de KDD 29
Figura 4 – Principais metodologias utilizadas em projetos de *data mining* 30
Figura 5 – Visão geral da metodologia CRISP-DM. 31
Figura 6 – Macroprocesso de celebração de convênios 33
Figura 7 – Quantidade de instrumentos de repasse do MDR 34
Figura 8 – Árvore do modelo CART para *credit scoring* 36
Figura 9 – Esquema simplificado de uma rede MLP. 39
Figura 10 – Evolução do XGBoost 41
Figura 11 – Exemplo de espaço de busca explorado pelo método busca em grade. . 46
Figura 12 – Métodos busca em grade (a) e busca aleatória (b) 47
Figura 13 – Fluxo da metodologia proposta 49
Figura 14 – Modelo de dados do Siconv 51
Figura 15 – Análise gráfica de variáveis 53
Figura 16 – Esquema *k-fold cross-validation* 56
Figura 17 – Representação Visual dos resultados de classificação 59
Figura 18 – Pesos das principais características utilizadas pelo *XGBoost*, sem otimização. 62
Figura 19 – Pesos das principais características utilizadas pelo *XGBoost*, otimizado. 63
Figura 20 – Otimzação de Hiperparâmetros do modelo. 64
Figura 21 – *Area Under Curve* dos classificadores para dados desbalanceados. . . . 65
Figura 22 – Densidade Cumulativa de Previsões Aprovados e Reprovados 65
Figura 23 – Curva *Precisão x Sensibilidade* dos classificadores para dados desbalanceados. 66
Figura 24 – Pesos das principais características utilizadas pelo *XGBoost*. 69
Figura 25 – Análise de variáveis que impactaram nas previsões. 71

Lista de tabelas

Tabela 1 – Faixa de valores adotada para criação de nova característica. 54
Tabela 2 – Proporção da distribuição dos dados de treino e teste *k-fold*, para k=10 55
Tabela 3 – Espaço de busca de hiperparâmetros utilizados no *Hyperopt*. 57
Tabela 4 – Espaço de busca de hiperparâmetros - Regressão Logística e Multilayer Perceptron. 57
Tabela 5 – Métricas importantes para avaliação da performance da metodologia. . 58
Tabela 6 – Resumo das métricas para dados desbalanceados sem otimização de hiperparâmetros 61
Tabela 7 – Resumo das métricas para dados desbalanceados após otimização de hiperparâmetros 62
Tabela 8 – Comparativo dos resultados dos modelos com diferentes métodos de *resampling* 67
Tabela 9 – Caso de Falso Positivo - Contribuição das Variáveis 70
Tabela 10 – Caso de Falso Negativo - Contribuição das Variáveis 70
Tabela 11 – Artigos produzidos referentes ao tema de classificação de riscos em transferências voluntárias. 74

Lista de abreviaturas e siglas

ACU	Acurácia
AG	Algoritmo Genético
AOP	Algoritmo de Otimização de Parâmetros
API	Application Programming Interface
AUC	Area Under the Curve
CPU	Computer Processing Unit
CST	Column Sample by Tree
ESP	Especificidade
FN	Falso Negativo
FP	Falso Positivo
GA	Genetic Algorithm
GB	Gigabyte
GPU	Graphics Processing Unit
KNN	K-Nearest Neighbors
MCW	Min Child Weight
MLP	Multilayer Perceptron
MXD	Max Depth
PRE	Precisão
RAM	Random Access Memory
ReLU	Rectified Linear Unit
RF	Random Forest
RNA	Redes Neurais Artificiais
ROC	Receiver Operating Characteristic

SEN	Sensibilidade
SPW	Scale Positive Weight
SS	SumSample
SVM	Support Vector Machine
TN	True Negative
TP	True Positive

Sumário

1 INTRODUÇÃO 16
1.1 Justificativa 17
1.2 Objetivos 19
1.3 Contribuições 20
1.4 Organização do Trabalho 20

2 TRABALHOS RELACIONADOS 22
2.1 *Credit scoring* por meio da aprendizagem computacional 22
2.2 Desbalanceamento de dados no domínio de *Credit Scoting* 24
2.3 Seleção de características no domínio de *Credit scoring* 25
2.4 Otimização de hiperparâmetros 25
2.5 Métricas de avaliação dos modelos 26
2.6 Algoritmos com estruturas de *Gradient Boosting* 27

3 FUNDAMENTAÇÃO TEÓRICA 29
3.1 *Cross Industry Standard Process for Data Mining* 29
3.2 Transferências Voluntárias 32
3.3 *Credit Scoring* 35
3.4 Técnicas de *Amostragem* 36
3.5 *Regressão Logística* 37
3.6 *Multilayer Perceptron* 39
3.7 *eXtreme Gradient Boosting* 40
3.8 Otimização de hiperparâmetros 44

4 METODOLOGIA 49
4.1 Extração dos Dados 50
4.2 Pré-processamento e Seleção 51
4.3 Execução e Otimização de Hiperparâmetros do *XGBoost* 55
4.4 Validação do Modelo 58

5 RESULTADOS 61

6 CONCLUSÃO 72
6.1 Trabalhos Futuros 73
6.2 Produções Científicas 73

REFERÊNCIAS 75

1 Introdução

Com a promulgação da Constituição Federal de 1988 no Brasil, estabeleceu-se um processo de descentralização político-administrativa dos entes federados, impondo maior responsabilidade a estados e municípios, que se tornaram os principais responsáveis pela alocação eficiente de seus recursos para a consecução das políticas públicas.

Ocorre que este mesmo pacto federativo, ao descentralizar as responsabilidades administrativas a estados e municípios, criou uma desigualdade em termos de arrecadação e disponibilidade financeira. Segundo dados de estudos da carga tributária de 2017, feitos pela Receita Federal do Brasil (BRASIL, 2017), a arrecadação dos municípios ficou em cerca de 6,26% da receita tributária total, ou 2,03% do Produto Interno Bruto (PIB), o equivalente a R$ 133.189,98 milhões, enquanto que estados ficaram com 25,72% da arrecadação e a União com 68,02%.

Conforme Moutinho e Kniess (2017), o processo de redefinição dos papéis dos entes da federação trouxe para os municípios a responsabilidade de oferecer um conjunto de serviços à comunidade que, até então, era de responsabilidade de outras esferas federativas. Em que pese o montante financeiro mencionado, com a descentralização promovida pela chamada redemocratização, esses recursos não são suficientes para fazer frente à entrega eficiente das novas políticas públicas.

Para financiar suas novas despesas, é necessário que os entes federados se valham de outras fontes de financiamento, via descentralização de recursos da União para estados e municípios. Essas transferências podem ser classificadas em obrigatórias ou discricionárias, as chamadas transferências voluntárias. As primeiras compreendem aquelas decorrentes de imposição normativa, sejam constitucionais ou legais, enquanto que as últimas abrangem aqueles repasses que devem observar critérios específicos de cada órgão repassador. Entre os principais instrumentos utilizados para as transferências voluntárias estão o convênio, o contrato de repasse e os termos de parceria. A Lei Complementar 101/2000, denominada Lei de Responsabilidade Fiscal, caracteriza as transferências voluntárias como “a entrega de recursos correntes ou de capital a outro ente da Federação, a título de cooperação, auxílio ou assistência financeira, que não decorra de determinação constitucional, legal ou os destinados ao Sistema Único de Saúde”.

No caso de convênios, os recursos são transferidos diretamente da União para o município ou entidade convenente. O contrato de repasse equipara-se ao convênio, no entanto, há a intermediação de instituições ou agências financeiras oficiais federais, destinados à execução de programas governamentais. Por fim, o termo de parceria é um instrumento para transferência de recursos a entidades qualificadas como Organizações da

Sociedade Civil de Interesse Público (OSCIP) para o fomento e a execução de atividades de interesse público.

Nesta dissertação, iremos nos concentrar nas transferências voluntárias efetuadas por meio de convênios e contratos de repasse. Todo o acompanhamento, desde à formalização e execução, até a prestação de contas, deve ser realizado por meio do Sistema de Gestão de Convênios e Contratos de Repasse (Siconv), que faz parte da plataforma +Brasil (ECONOMIA, 2019b), do Governo Federal.

O Siconv disponibiliza livremente as informações de transferências voluntárias da União com o objetivo de facilitar o acesso aos dados do sistema pela sociedade e outras esferas de Governo, promovendo, assim, a transparência. Conforme obrigação estabelecida pela Constituição Federal, é dever de qualquer pessoa que venha a receber recursos públicos federais a prestação de contas quanto ao seu bom e regular uso.

A Portaria Interministerial 424/2016 (BRASIL, 2016) também estabelece a competência para acompanhamento e fiscalização dos convênios e contratos de repasse, determinando que responderá o convenente pelos danos causados a terceiros, decorrentes de culpa ou dolo na sua execução. Referida norma, cria, ainda, mecanismos e condições para a prestação de contas da boa e regular aplicação dos recursos recebidos, atribuindo ao órgão concedente responsabilidade por decidir sobre a regularidade da aplicação dos recursos transferidos e, se extinto, ao seu sucessor, registrando todos os seus atos no Siconv.

Após a análise da prestação de contas dos recursos, caso esta não seja aprovada, e exauridas todas as providências cabíveis para a regularização de eventuais pendências ou reparação do dano apurado, a autoridade competente, sob pena de responsabilização solidária, registrará o fato no Siconv e adotará as providências necessárias à instauração do processo de Tomada de Contas Especial (TCE), para apurar responsabilidade por ocorrência de dano, com apuração de fatos, quantificação do dano, identificação dos responsáveis e obtenção do respectivo ressarcimento dos recursos transferidos.

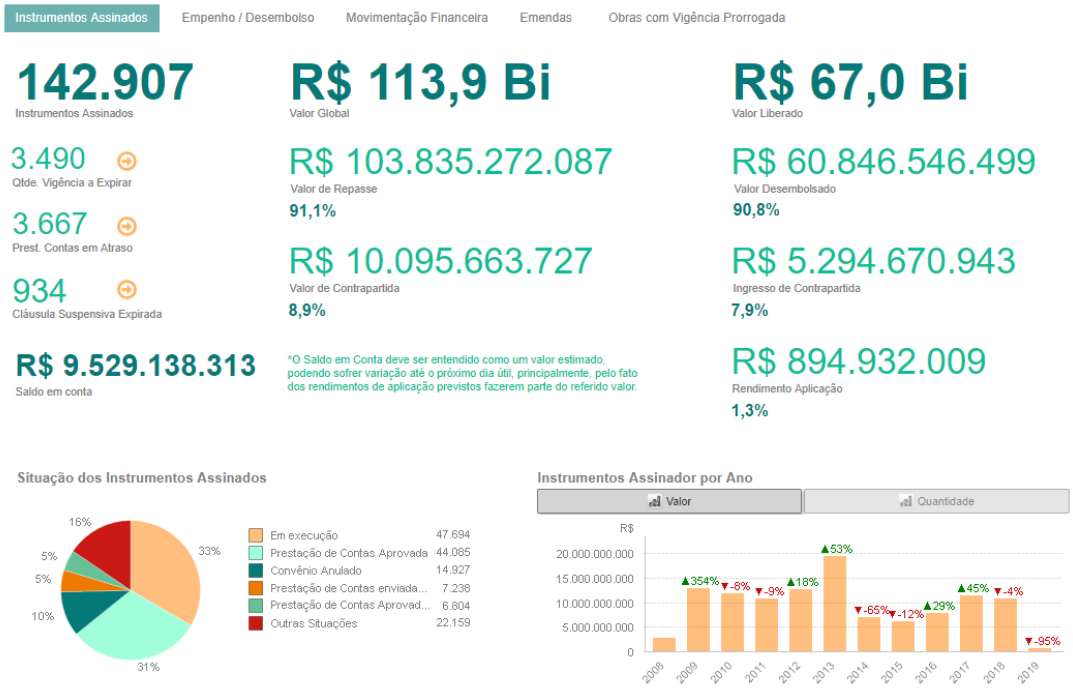
Ressalte-se que não é objeto deste trabalho a análise minuciosa do processo de acompanhamento, fiscalização e prestação de contas dos instrumentos em tela. Analisaremos, tão somente, os resultados das análises das prestações de contas quanto ao atingimento do objeto proposto no plano de trabalho, e ulterior parecer pela aprovação ou reprovação das contas, portanto, uma análise quantitativa do processo de controle e prestação de contas.

1.1 Justificativa

Em consulta ao painel de transferências abertas, apresentado na figura 1, verifica-se o volume financeiro que é disponibilizado pela União para estados e municípios por meio de transferências voluntárias. Ao todo, são 142.907 instrumentos assinados, e que somam

cerca de R$ 133,9 bilhões de reais.

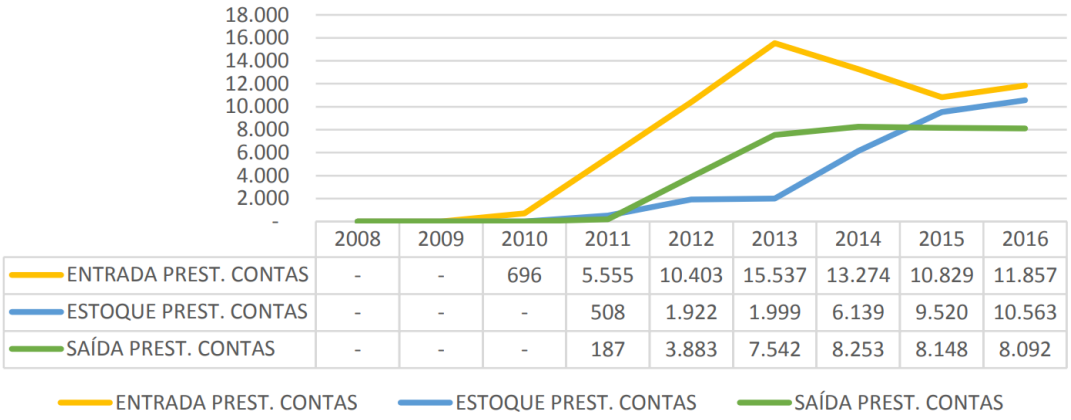
Figura 1 – Volume financeiro disponibilizado pela União, por meio de transferências voluntárias



Fonte: Economia (2019a)

Desta forma, o volume de recursos, além da quantidade de instrumentos de repasse, torna evidente a importância de um bom processo de verificação das prestações de contas. Não obstante a evidente materialidade, o esforço despendido na análise da regular aplicação desses recursos não tem sido suficiente para dar vazão ao fluxo dos processos de prestação de contas. Consultando dados do Siconv, pode-se verificar que é crescente a quantidade de processos de prestação de contas que ficam em estoque, aguardando sua conclusão, conforme apresentado na figura 2, o que pode favorecer a má gestão de tais recursos.

Figura 2 – Fluxo anual de prestação de contas - entradas x estoque x saídas



	2008	2009	2010	2011	2012	2013	2014	2015	2016
ENTRADA PREST. CONTAS	-	-	696	5.555	10.403	15.537	13.274	10.829	11.857
ESTOQUE PREST. CONTAS	-	-	-	508	1.922	1.999	6.139	9.520	10.563
SAÍDA PREST. CONTAS	-	-	-	187	3.883	7.542	8.253	8.148	8.092

Fonte: Ministério da Transparência e Controladoria-Geral da União (2018)

Tal entendimento foi corroborado pela Controladoria-Geral da União (CGU) em relatório que avalia a gestão das transferências voluntárias da União (MINISTÉRIO DA TRANSPARÊNCIA E CONTROLADORIA-GERAL DA UNIÃO, 2018), o qual constatou preocupante evolução do quantitativo de prestações de contas fora do prazo de análise e manifestação definitiva do repassador de recursos.

Portanto, para assegurar a tempestividade na análise das prestações de contas, e permitir que os esforços sejam direcionados para aqueles trabalhos que têm o condão de gerar os melhores resultados, é necessária uma ferramenta capaz de classificar os processos de prestação de contas em perfis de risco, para que as instâncias de controle envidem esforços naqueles processos que possuem o maior risco de fracasso, facilitando, assim, a recuperação dos recursos.

1.2 Objetivos

Diante do contexto apresentado, o objetivo principal deste trabalho é propor uma metodologia para a classificação de transferências voluntárias do governo federal em perfis de risco de não cumprimento do seu objetivo, utilizando o *XGBoost*, com o propósito de aumentar a eficiência dos órgãos de controle nas análises de prestações de contas e recuperação dos recursos financeiros transferidos.

Mais especificamente, o presente trabalho busca os seguintes objetivos aplicados ao problema de classificação de transferências voluntárias do governo federal:

- Estudar os conceitos e principais problemas nas prestações de contas dos instrumentos de repasse;
- Estudar e investigar a utilização dos algoritmos *XGBoost*, *Logistic Regression* e

Multilayer Perceptron na construção de modelos de classificação para a metodologia proposta;

- Estudar e investigar as técnicas de *oversampling* e *undersampling* para rebalanceamento de dados : *Synthetic Minority Over-sampling Technique* (SMOTE) e *Near Miss Undersampling*;
- Estudar e investigar as técnicas de otimização de hiper-parâmetros *random search (RS)* e *tree-structured Parzen estimator(TPE)* bayesiano;
- Analisar as vantagens e limitações do modelo proposto;
- Contribuir com o controle externo dos repasses de recursos federais por meio de proposta de utilização técnica de aprendizagem computacional para classificação destas transferências em perfis de risco de não cumprimento dos objetivos propostos nos instrumentos de repasse;

1.3 Contribuições

As principais contribuições deste trabalho são:

- Utilização de algoritmo de *boosting* com busca de parâmetros ideais para a classificação de transferência voluntárias em perfis de risco;
- Desenvolvimento de um método automatizado capaz de auxiliar órgãos de controle na análise das prestações de contas de transferências voluntárias de recursos federais.

Os resultados obtidos com o uso do *XGBoost* aplicado à classificação de risco de transferências voluntárias mostraram-se efetivos ao capturar padrões em instrumentos de repasse que possuem diversidade de características. Apesar de haver iniciativas de órgãos de controle para a automatização desta atividade, poucas publicações desses resultados podem ser encontradas na literatura.

1.4 Organização do Trabalho

Além da Introdução, esta dissertação está organizada em 6 capítulos, apresentados a seguir.

O **Capítulo 2** (Trabalhos Relacionados) descreve os trabalhos relacionados ao tema de classificação e atribuição de perfis de risco, principalmente relacionados a técnicas de *credit scoring*, expondo metodologias e resultados, utilizadas para a classificação em

perfis de risco, além de técnicas de busca de hiperparâmetros e seleção de características ideais para a construção do modelo.

O **Capítulo 3** (Fundamentação Teórica) apresenta a fundamentação teórica do modelo adotado, descrevendo conceitos importantes, além do embasamento teórico das principais técnicas e algoritmos utilizados neste trabalho.

O **Capítulo 4** (Metodologia) apresenta uma descrição detalhada sobre o desenvolvimento da metodologia proposta, a qual é baseada no modelo de referência *Cross Industry Standard Process for Data Mining* (CRISP-DM), que resultou na criação do modelo de classificação de riscos de transferências voluntárias.

No **Capítulo 5** (Resultados), os resultados são apresentados e discutidos em relação às técnicas aplicadas.

Por fim, o **Capítulo 6** (Conclusão) apresenta as considerações finais sobre o trabalho realizado, as propostas de trabalhos futuros e artigos científicos publicados, relacionados ao tema.

2 Trabalhos Relacionados

Neste capítulo, serão apresentados trabalhos já desenvolvidos utilizando diversas técnicas de inteligência computacional com aplicação tanto no auxílio para classificação de riscos em operações financeiras e recuperação de crédito quanto para avaliações voltadas ao setor público. Desta forma, a proposta é analisar os melhores modelos de classificação de risco para pontuação de créditos (*credit scoring*) como forma de avaliar o sucesso ou fracasso das transferências voluntárias, quanto à sua prestação de contas.

Os trabalhos descritos a seguir estão organizados em duas partes: primeiramente são abordados os trabalhos que utilizam diversas técnicas de aprendizado de máquina para o problema proposto (pontuação de crédito); e em seguida, relacionam-se os trabalhos que utilizam apenas técnicas de *gradient boosting*, as quais foram utilizadas para a condução dos experimentos, e suas técnicas de otimização, bem como formas de melhor comparar os modelos produzidos.

2.1 *Credit scoring* por meio da aprendizagem computacional

Uma das formas de se compreender as transferências voluntárias é entendê-las como uma espécie de operação de crédito em sentido amplo, um empréstimo, feito pela União a estados e municípios (BRASIL, 2004), portanto, exposta a determinados riscos de problemas. Contudo, o que as diferencia é que os seus resultados devem ser avaliados em termos de benefícios sociais e não meramente financeiros. A avaliação do risco de crédito (*credit scoring*) é a análise do risco associado ao emprestar recursos para empresas e indivíduos, e envolve o uso de ferramentas de gerenciamento de risco desde a pré-triagem de um potencial mutuário, até o gerenciamento da conta durante sua vida útil e possível baixa contábil (CROOK; EDELMAN; THOMAS, 2007). Vários trabalhos abordam o tema por diferentes perspectivas, sendo as principais a classificação discreta e binária, ou uma atribuição de escala contínua de risco de calote.

O trabalho proposto por Romanyuk (2015) apresenta um método para avaliar o risco de crédito com base em uma escala contínua. Sua aplicação pode, por exemplo, atribuir custos individuais de empréstimo de uma pessoa a depender do risco, em vez de definir um preço de empréstimo padrão para todos. O modelo usa coeficientes de ponderação e funções de qualidade, que auxiliam no cálculo da qualidade da pontuação de crédito em uma escala contínua. O trabalho mencionado aplicou a abordagem em dados da base de crédito alemã e mostrou a viabilidade desse tipo de abordagem para a finalidade proposta.

Na perspectiva discreta de categorização, outro trabalho (BRAVO; THOMAS; WEBER, 2015) propôs um estudo do comportamento dos mutuários e sua classificação em "bons pagadores", "não podem pagar", devido a problemas de fluxo de caixa, por exemplo, ou "não vão pagar", estes por ação deliberada de deixar de pagar. Outros estudos também avaliaram o comportamento de inadimplentes, e as razões que levam à inadimplência, demonstrando que parte dos mutuários não age de forma racional, e outros acabam deixando de pagar suas dívidas de forma estratégica, a depender das condições contratuais (GUISO; SAPIENZA; ZINGALES, 2013; JANGER; BLOCK-LIEB, 2006; ALARY; GOLLIER, 2004).

No contexto do presente trabalho, entende-se que não há espaço na legislação para discriminar custos individuais para transferências de recursos para os entes federados, e nem avaliar as razões que levam à possível rejeição das contas de determinada transferência voluntária. Nesse sentido, *credit scoring* também pode ser avaliado em uma perspectiva de classificação binária, ou seja, discriminando-se entre bons e maus pagadores, abordagem similar à utilizada nesta pesquisa.

Vários trabalhos analisaram a avaliação de risco de crédito sob a perspectiva de classificação binária. Louzada, Ara e Fernandes (2016) apresentaram uma extensa revisão de literatura (1992-2015) acerca da aplicação de técnicas de classificação binária para *credit scoring*. Entre os principais objetivos apontados no estudo estão a proposição de novos métodos de avaliação, comparativos entre técnicas de classificação, seleção de características, entre outros.

Várias técnicas de aprendizagem computacional já foram utilizadas para a classificação de crédito. Entre elas, redes neurais (AKKOÇ, 2012), programação genética (ONG; HUANG; TZENG, 2005), máquinas de vetores de suporte (HARRIS, 2015), regressão logística (NIKOLIC et al., 2013), árvores de decisão (XIA et al., 2017), entre outras. Em recente análise de literatura, Dastile, Celik e Potsane (2020) apontaram inúmeras técnicas de aprendizado computacional utilizadas para *credit scoring*, sendo regressão logística, máquinas de vetores suporte e redes neurais as mais comuns para classificadores individuais, para arranjos de classificadores, técnicas de *Boosting* foram as mais utilizadas.

Lessmann et al. (2015) realizaram extenso *benchmarking* dos principais algoritmos de classificação para pontuação de crédito como forma de atualizar um estudo anterior realizado por Baesens et al. (2003). Os 41 trabalhos analisados foram categorizados em 3 grandes grupos de classificadores: individuais, arranjos homogêneos (*homogeneous ensemble*), e arranjos heterogêneos (*heterogeneous ensemble*). Ao final de sua comparação, os autores mostraram que arranjos heterogêneos tiveram, no geral, melhores performances que os demais.

Em outro trabalho recente, Paula et al. (2019) compararam a utilização das técnicas de regressão logística à utilização de *random forests*. A análise revela que os métodos

estatísticos melhoram a previsibilidade do padrão quando comparado ao uso de técnicas subjetivas. O estudo mostrou, também, a superioridade do modelo de *random forests* na estimativa da pontuação de crédito e pontuação de lucro quando comparado ao método de regressão logística dos mínimos quadrados ordinários (*Ordinary Least Squares* - OLS). Além disso, a pesquisa mostra que as duas análises podem ser usadas em conjunto para uma tomada de decisão ainda mais eficiente.

Nessa mesma linha, Lopes et al. (2016) propuseram um estudo acerca da recuperação de operações de crédito em um banco brasileiro, no qual desenvolveram modelos utilizando *Generalized Linear Models* (GLM), *Gradient Boosted Methods* (GBM) e *Distributed Random Forest* (DRF). Como resultado, o GBM apresentou as melhores performances para os indicadores utilizados.

Uma abordagem do uso da tecnologia para mineração de dados visando detectar a lavagem de dinheiro também foi analisada (PAULA, 2016). Apesar de não estar relacionada ao contexto de *credit scoring* - abordagem usada em nosso trabalho - mostra que a utilização de técnicas computacionais permite a identificação ou avaliação de situações relacionadas a recursos públicos.

2.2 Desbalanceamento de dados no domínio de *Credit Scoting*

Outro aspecto muito abordado na literatura é o desbalanceamento dos dados entre as classes. Dados relativos a risco de crédito geralmente apresentam um forte desbalanceamento. Os mutuários adimplentes geralmente são em maior quantidade que os mutuários inadimplentes. Nos casos em que há um grande desequilíbrio nos dados, muitos modelos passam a ser tendenciosos para a classe majoritária (BRANCO; TORGO; RIBEIRO, 2016). Para mitigar esse comportamento, frequentemente são utilizadas técnicas como superamostragem (*oversampling*) ou subamostragem (*undersampling*). Em sua pesquisa, Dastile, Celik e Potsane (2020) verificaram que apenas 18% dos estudos primários da literatura consultada balancearam seus conjuntos de dados. Segundo eles, a técnica mais utilizada é a subamostragem da classe majoritária.

Na literatura, podem ser encontradas técnicas de *oversampling*, aplicadas à classe minoritária, ou *undersampling*, aplicadas à classe majoritária. Entre as técnicas de *oversampling*, já foram utilizadas técnicas como *Self-Organizing Map-based Oversampling* (SOMO) (DOUZAS; BACAO, 2017), e *Synthetic Minority Over-sampling Technique* (SMOTE) (SUN et al., 2018). Entre as técnicas de *undersampling*, alguns estudos compararam técnicas de *k-Nearest Neighbors* (kNN) (CHYI, 2003; MANI; ZHANG, 2003) e *under-Sampling Based on Clustering* (SBC) (YEN; LEE, 2009) com a subamostragem aleatória. Em seu estudo, Sun et al. (2018) também compararam técnicas combinadas de *oversampling* e *undersampling* aleatórios com SMOTE e para gerar seus dados de treinamento, chegando

a resultados promissores.

Crone e Finlay (2012) compararam o impacto do desbalanceamento de dados em diferentes classificadores, como Regressão Logística, *Linear Discriminant Analysis*, árvores de decisão CART e redes neurais. Os resultados do estudo feito em dados de *credit scoring* mostram que o desbalanceamento entre classes, de fato, afeta o desempenho dos classificadores, indicando que Regressão Logística é mais robusta e é menos afetada pelo desbalanceamento do que as demais técnicas em análise.

De forma similar, em um estudo realizado com *Random Forests* e *Gradient Boosting*, Brown e Mues (2012) mostraram que tais técnicas têm um desempenho muito bom em problemas de *credit scoring* e são capazes de lidar bem com o desbalanceamento acentuado de dados. Entretanto, outras técnicas como *Decision Trees* e *k-Nearest Neighbors*, não conseguiram lidar tão bem com o desbalanceamento de dados.

2.3 Seleção de características no domínio de *Credit scoring*

Outra preocupação comum encontrada na literatura diz respeito às técnicas de seleção de características, importante etapa de pré-processamento no processo de descoberta de conhecimento em bases de dados (LIANG; TSAI; WU, 2015). Hajek e Michalak (2013) mostraram que a seleção de características no âmbito de *credit scoring* aumentou a performance de vários classificadores testados, como Redes Neurais, Máquinas de Vetores de Suporte, *Random Forests*, entre outros, permitindo a redução no número de *features* utilizadas pelos modelos. Porém, conforme estudo conduzido por Liang, Tsai e Wu (2015), nem sempre utilizar técnicas de seleção de características leva a um aumento significativo na performance dos classificadores.

Entre as principais técnicas de seleção de características utilizadas, destacam-se: *Stepwise* (WONGCHINSRI; KURATACH, 2017; BROWN; MUES, 2012; BIJAK; THOMAS, 2012), *F-score* (CHEN; LI, 2010), *Rough Set* (CHEN; LI, 2010), *Genetic Algorithm* (JADHAV; HE; JENKINS, 2018), *Principal Component Analysis* (PCA) (MANCISIDOR et al., 2018), *AutoEncoder* (MANCISIDOR et al., 2018), *Linear Discriminant Analysis* (LDA) (ZHANG; YANG; ZHOU, 2018) e *Least Absolute Shrinkage and Selection Operator* (LASSO) (MALDONADO; PÉREZ; BRAVO, 2017).

2.4 Otimização de hiperparâmetros

Além de técnicas que lidam com os dados utilizados no treinamento dos modelos, como rebalanceamento dos dados e seleção de características, também é dada bastante importância à otimização dos hiperparâmetros dos modelos avaliados, como forma de encontrar os melhores parâmetros para o problema proposto. O processo de otimização

pode ser visto como um problema de otimização do tipo caixa preta, cuja função objetivo está associada ao desempenho preditivo do modelo induzido pelo algoritmo (Mantovani et al., 2016). Em termos de custos computacionais, a otimização de hiperparâmetros pode ser uma tarefa bastante onerosa (THORNTON et al., 2013) e, ao mesmo tempo, pode ser ineficaz em seu objetivo, além de trazer pouca ou nenhuma melhora na performance dos modelos (RIDD; GIRAUD-CARRIER, 2014).

É possível encontrar na literatura várias técnicas de otimização de hiperparâmetros. Desde as mais simples, como *Grid-Search* – abordagem determinística cartesiana que reduz a um número finito o espaço de busca de hiperparâmetros (BRAGA et al., 2013) – e *Random-Search* – abordagem também determinística, porém com exploração randômica do espaço de busca, proposta por Bergstra e Bengio (2012) – até algumas mais complexas como meta-heurísticas (FRIEDRICHS; IGEL, 2005) e meta-aprendizado (FEURER; SPRINGENBERG; HUTTER, 2015) – abordagens de otimização *Bayesiana*.

No âmbito de *credit scoring*, recentemente, Wang e Ni (2019) propuseram uma modelagem de risco utilizando um algoritmo de *gradient boosting*, *XGBoost*, e otimização de hiperparâmetros utilizando a abordagem de *tree-structured Parzen Estimators* (TPE) Bayesiana, abordagem proposta por Bergstra et al. (2011). Nesta mesma linha, Xia et al. (2017) também propuseram um método de *credit scoring* utilizando *XGBoost* e otimização com TPE, e verificaram que esta abordagem de otimização é superior a abordagens que utilizam *Grid Search* ou *Random Search*.

2.5 Métricas de avaliação dos modelos

Quanto às métricas de avaliação dos modelos, Dastile, Celik e Potsane (2020) apresentam em sua pesquisa as métricas mais comumente utilizadas em problemas do domínio de *credit scoring*: percentual classificado corretamente (PCC), ou acurácia, *Recall*, ou Sensibilidade, Especificidade, F-*Score* e Área Sob a Curva ROC, do inglês, *Area Under Curve* (AUC), Erro Tipo I, ou Taxa de Falso Positivo, Erro Tipo II, ou Taxa de Falso Negativo.

Em (THARWAT, 2018) é verificado o grau de sensibilidade ao desbalanceamento de dados de algumas métricas de avaliação, muito comuns no domínio de *credit scoring*. Métricas como precisão e sensibilidade utilizam dados tanto dos casos positivos quanto dos casos negativos, o que leva a um viés para o lado da classe dominante. Porém, métricas como *Geometric Mean*, sensibilidade e especificidade são menos sensíveis ao desbalanceamento de dados. Outra métrica muito utilizada para a comparação de modelos é a curva Precisão-Sensibilidade, que analisa o *trade-off* entre precisão e sensibilidade, à medida em que se varia o limiar de classificação do modelo (FLACH; KULL, 2015).

Além de utilizarem técnicas de *gradient boosting*, com técnicas de otimização de

hiperparâmetros, em seu trabalho de *credit scoring*, Wang e Ni (2019) utilizaram como métricas de avaliação do modelo acurácia, sensibilidade (*recall*), F-*Score* e área sob a curva ROC.

Considerando que modelos de arranjo (*ensemble*), em especial, métodos de *gradient boosting* utilizando árvores de decisão, apresentaram, em geral, melhores performances e lidam bem com desbalanceamento de dados, e levando em conta que árvores de decisão são mais robustas a ruídos nos dados, possuem baixo custo computacional, e lidam bem com atributos redundantes (BARROS et al., 2011) , buscamos analisar os recentes avanços de pesquisa nas referidas técnicas.

2.6 Algoritmos com estruturas de *Gradient Boosting*

Wang e Ni (2019) propuseram a utilização de um modelo de arranjo relativamente novo, que utiliza um algoritmo de *gradient boosting*, chamado *extreme gradient boosting (XGBoost)*, via seleção de características e otimização bayesiana de hiperparâmetros. Em sua pesquisa, eles fizeram uma comparação com modelos de regressão logística.

Os autores lançaram mão de várias técnicas para buscar melhorar a performance do modelo proposto. Entre elas, seleção de características e otimização de hiperparâmetros. Para a seleção de características, foram utilizados cinco métodos, a saber: pesos por índice gini, pesos por chi-quadrado, agrupamento hierárquico de variáveis, pesos por correlação e pesos por taxa de ganho de informação. No que se refere à otimização de hiperparâmetros, foram utilizadas estratégias de busca randômica (*Random Search* - RS) e *Tree-structured Parzen Estimator* (TPE) bayesiano.

Nessa pesquisa, os autores buscaram avaliar como a seleção de características e a otimização de hiperparâmetros afetam a performance tanto do *XGBoost* quanto da regressão logística, bem como se o *XGBoost* é melhor na tarefa de previsão de risco para o domínio proposto.

Após selecionar os modelos que obtiveram os melhores resultados depois da etapa de seleção de características, os autores fizeram a comparação destes modelos utilizando as técnicas de otimização de hiperparâmetros. Como resultados, os modelos de *XGBoost*, tanto utilizando RS como TPE para otimização de hiperparâmetros, obtiveram resultados superiores à regressão logística. Entre estes, o modelo que utiliza TPE teve performance superior ao modelo de RS.

Ainda nesse contexto da área de negócios de gerenciamento de risco de crédito, o trabalho de Jiang, Ji e Li (2011) apresenta uma análise no cenário chinês. O problema de atualização do *credit scoring* das pessoas restringe e impede o desenvolvimento sólido dos negócios de crédito ao consumidor. O referido trabalho propõe a otimização com

combinação de modelos usando *Simulated Annealing - Genetic Algorithm* (SA-GA) e mostra que o modelo combinado pode ser efetivamente usado para explicar efetivamente o impacto das variáveis sobre a inadimplência.

Em outro estudo no domínio de riscos financeiros, Zięba, Tomczak e Tomczak (2016) também propuseram a utilização de um modelo com *XGBoost* para a previsão de falências bancárias. Em suas pesquisas, eles utilizaram a métrica AUC em razão do alto grau de desbalanceamento de seus dados.

Apesar de o *XGBoost* ter sido objeto de bastantes pesquisas na comunidade acadêmica, recentemente outro algoritmo de *gradient boosting* tem ganhado bastante atenção. O *LightGBM* foi proposto por Ke et al. (2017) como uma alternativa mais eficiente ao *XGBoost*, utilizado como parâmetro de avaliação pelos autores. Um trabalho recente comparou a performance de 3 algoritmos de *gradient boosting* no domínio de análise de créditos (*LightGBM*, *XGBoost* e *CatBoost*), mostrando uma vantagem para o *LightGBM* (DAOUD, 2019). Porém o autor faz uma ressalva acerca da generalização dos resultados para outras bases de dados.

A pesquisa conduzida por Ke et al. (2017) mostra que o *LightGBM* é mais rápido que o *XGBoost* e consome menos memória. Porém, apesar de mais lento, o *XGBoost* atingiu métricas de performance como acurácia e AUC similares ao *LightGBM*. Apesar de ter obtido vantagens, o *LightGBM* ainda não foi tão explorado academicamente quanto o *XGBoost*, sendo difícil encontrarmos pesquisas científicas que utilizam este algoritmo, em especial, no domínio de *credit scoring*.

É importante ressaltar que o *XGBoost* vem ganhando bastante atenção da comunidade acadêmica em razão de suas recentes conquistas em vários domínios e em vários desafios de aprendizado de máquina e mineração de dados. Olhemos o site de competições de aprendizado de máquina *Kaggle*, por exemplo. Entre as 29 soluções vencedoras do desafio publicadas no *blog* da *Kaggle* em 2015, 17 soluções usaram o *XGBoost*. Entre essas soluções, oito usaram apenas o *XGBoost* para treinar o modelo, enquanto outras combinaram o *XGBoost* em arranjos com redes neurais (CHEN; GUESTRIN, 2016).

Os trabalhos relacionados, apresentados neste capítulo, indicam que, para o domínio da presente pesquisa, a utilização do *XGBoost* é válida na tarefa de classificação de riscos em transferências voluntárias. Juntamente com esta técnica, faz-se viável a utilização de técnicas de *undersampling* e *oversampling* para o tratamento do desbalanceamento dos dados, e a utilização de TPE para a otimização dos hiperparâmetros do modelo.

3 Fundamentação Teórica

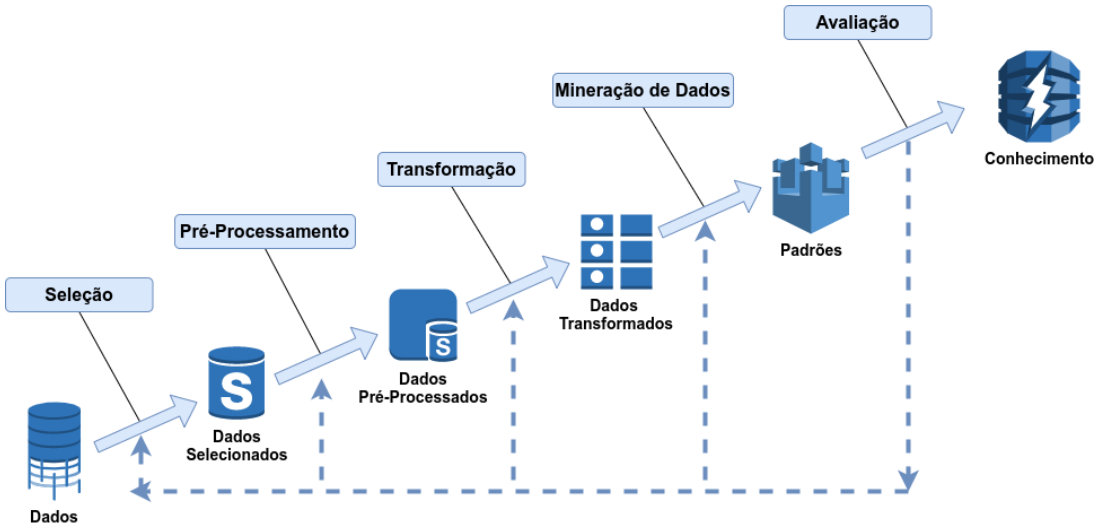
O presente capítulo descreve conceitos sobre as transferências voluntárias e suas prestações de contas, bem como os conceitos fundamentais do modelo de referência metodológica de *Data Mining*, *Cross Industry Standard Process for Data Mining* (CRISP-DM), das técnicas de aprendizagem computacional utilizadas *Gradient Boosting Machine* (GBM), *Logistic Regression* (LR), e *Multilayer Perceptron* (MLP), além das técnicas utilizadas para balanceamento dos dados e otimização de hiperparâmetros dos modelos.

Para melhor situar o leitor acerca das etapas conduzidas neste trabalho, iremos apresentar inicialmente o modelo CRISP-DM, no qual é baseada a metodologia proposta.

3.1 *Cross Industry Standard Process for Data Mining*

Mineração de dados é o processo de descoberta de padrões interessantes em grandes massas de dados e, como um processo de descoberta de conhecimento, tipicamente envolve a limpeza, seleção e transformação dos dados e a descoberta e evolução de padrões (HAN; KAMBER; PEI, 2012). A mineração de dados pode ser entendida como uma etapa de um processo de descoberta de conhecimento em bases de dados, conhecido como Knowledge Discovery in Databases (KDD), que aplica técnicas de análise de dados e aprendizagem computacional para extrair conhecimento dos dados (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996). A figura a seguir melhor situa a mineração de dados no processo de KDD.

Figura 3 – Visão geral dos passos do processo de KDD

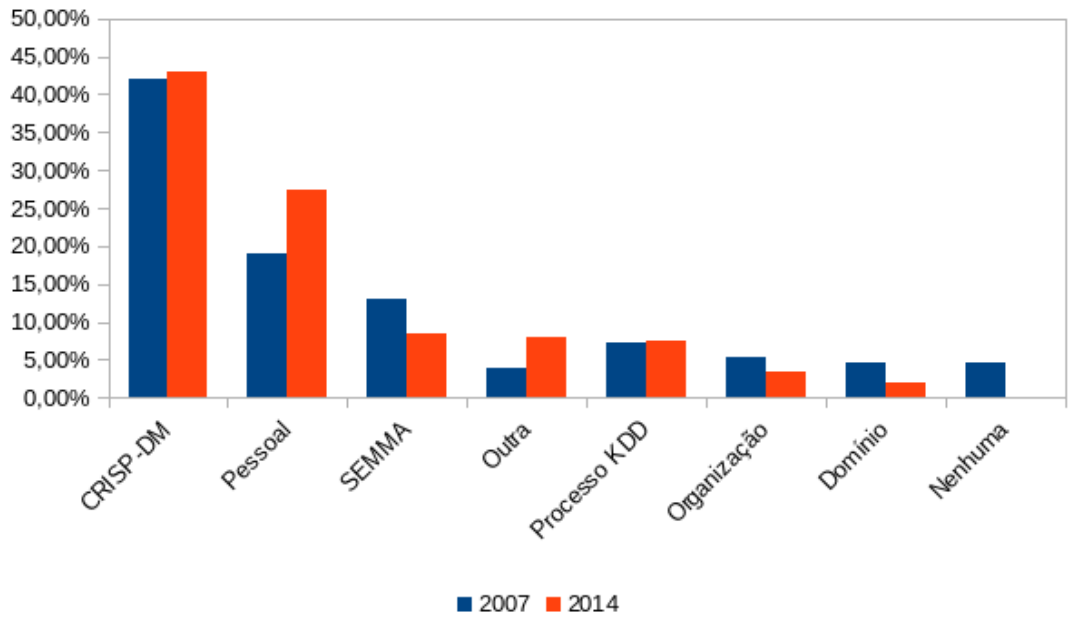


Fonte: Adaptado de Fayyad, Piatetsky-Shapiro e Smyth (1996)

Em outras palavras, a mineração de dados é o processo pelo qual se podem extrair padrões significativos dos dados, por meio de modelos representativos de relações existentes entre esses dados, valendo-se da utilização de técnicas estatísticas, aprendizagem computacional e reconhecimento de padrões (KOTU; DESHPANDE, 2015). Portanto, os problemas de mineração de dados podem ser agrupados de várias formas, como classificação, regressão, análises associativas, detecção de anomalias, séries temporais e mineração de textos.

Existem várias metodologias de *data mining* disponíveis, entre as quais se destacam DMAIC, SEMMA e CRISP-DM. Segundo Nisbet, Elder e Miner (2009), destas, se destaca a CRISP-DM por ser a metodologia que expressa de forma mais completa o processo de *data mining*. Em pesquisa conduzida pelo site KDnuggets[1], a metodologia CRISP-DM foi a mais utilizada em projetos de *data mining*, segundo a pergunta feita: "Qual a principal metodologia que você está usando nos seus projetos de *analytics*, *data mining* e ciência de dados?".

Figura 4 – Principais metodologias utilizadas em projetos de *data mining*

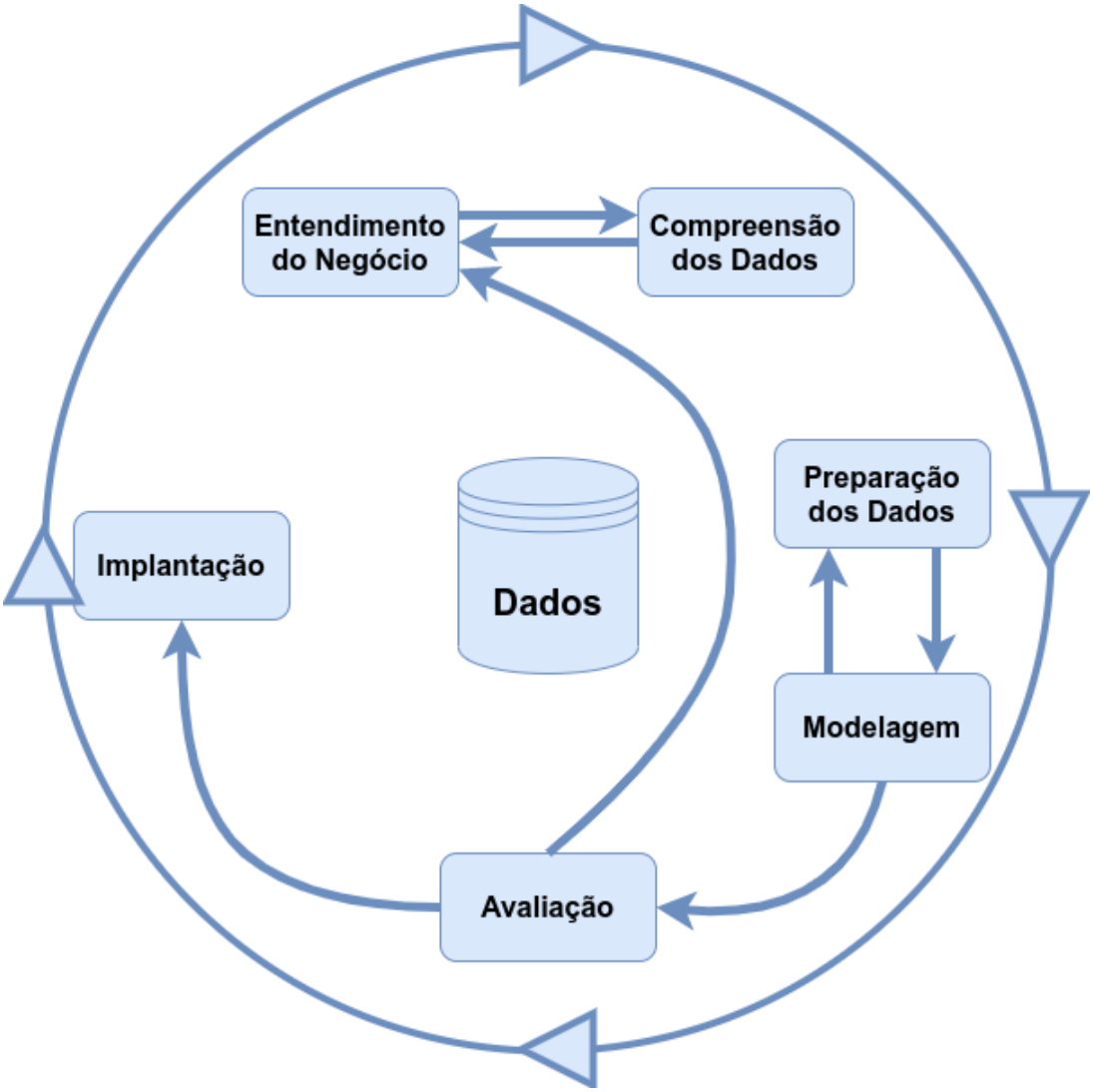


Fonte: Adaptado de KDnuggets

O presente trabalho foi conduzido com base no modelo de referência da metodologia para mineração de dados *Cross Industry Standard Process for Data Mining* (CRISP-DM)(CHAPMAN et al., 2000). Esse modelo fornece uma visão geral do ciclo de vida de um projeto de mineração de dados, conforme apresentado na figura 5.

[1] <https://www.kdnuggets.com/2014/10/crisp-dm-top-methodology-analytics-data-mining-data-science-projects.html>

Figura 5 – Visão geral da metodologia CRISP-DM.



Fonte: Adaptado de Chapman et al. (2000)

A metodologia CRISP-DM divide um projeto de mineração de dados em 6 grandes fases, a saber:

- **Entendimento do Negócio**: Nesta fase inicial, o foco é entender os problemas de negócio a serem resolvidos e traduzi-los em termos de objetivos do projeto de mineração de dados e quais resultados são esperados após a conclusão do projeto;
- **Compreensão dos Dados**: Nesta fase há um primeiro contato com os dados a serem trabalhados. Inicialmente, busca-se conhecer quais dados estão disponíveis, entender sua estrutura, modelos de dados e faz-se a coleta inicial dos dados para familiarizar-se com estes;
- **Preparação dos Dados**: A fase de preparação de dados engloba as atividades de limpeza e tratamento dos dados para a construção dos conjuntos de dados que serão

utilizados pelos modelos. Nesta etapa também são feitas algumas transformações e normalizações, além da geração de novos atributos e seleção de características, caso necessário;

- **Modelagem**: Esta é a fase em que as várias técnicas de modelagem são selecionadas e aplicadas, bem como é feita a otimização de seus hiperparâmetros. Dependendo das técnicas a serem avaliadas, é necessário voltar à fase de preparação de dados;
- **Avaliação**: Após a construção dos modelos, nesta fase é necessária a utilização de métricas capazes de inferir a qualidade dos modelos produzidos e dos resultados alcançados. É importante avaliá-los levando em conta os objetivos de negócios e verificar se os resultados são, de fato, resultados de generalizações adequadas dos modelos ou se houve um sobreajustamento dos dados, ou *overfit* (HAWKINS, 2004);
- **Implantação**: O projeto não acaba após a criação e seleção do melhor modelo. É necessário que se crie uma estratégia para a sua utilização em um ambiente real do negócio, para a situação para a qual o modelo foi concebido.

Cada uma dessas fases, por sua vez, é dividida em subníveis, que vão de atividades, operações e tarefas executadas em cada fase do projeto. É importante ressaltar que no Capítulo 4 será apresentado um maior detalhamento das tarefas da metodologia do presente trabalho.

3.2 Transferências Voluntárias

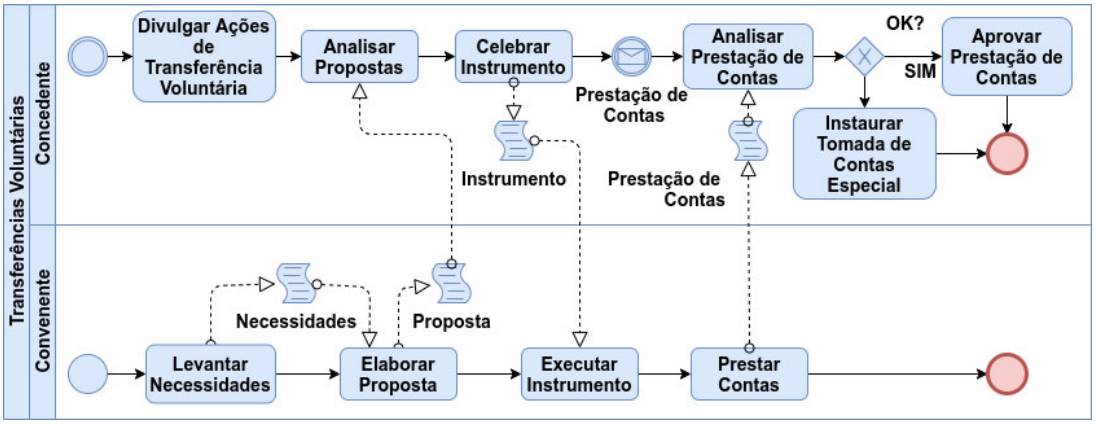
É importante entender a estrutura das transferências voluntárias, bem como as suas etapas e particularidades, em especial a sua prestação de contas, tendo em vista que este conhecimento facilita o desenvolvimento de técnicas de aprendizagem computacional para a tarefa de classificação de riscos destas transferências em perfis de risco.

A Lei Complementar 101/2000(BRASIL, 2000), denominada Lei de Responsabilidade Fiscal, caracteriza as transferências voluntárias como "a entrega de recursos correntes ou de capital a outro ente da Federação, a título de cooperação, auxílio ou assistência financeira, que não decorra de determinação constitucional, legal ou os destinados ao Sistema Único de Saúde".

Portanto, transferência voluntária é um instrumento legal firmado de comum acordo entre as partes para a execução de programas, projetos e atividades de interesse recíproco, no qual há clara definição das responsabilidades dos partícipes, sendo uma delas a obrigação da prestação de contas dos recursos transferidos.

A figura 6 apresenta uma simplificação do processo de negócio[2], com as etapas necessárias à celebração de um dos instrumentos de repasse, o convênio.

Figura 6 – Macroprocesso de celebração de convênios



Fonte: Adaptado pelo autor[2]

Para os convênios e contratos de repasse, a Portaria Interministerial 424/2016 (BRASIL, 2016), dos ministérios do planejamento, orçamento e gestão, da fazenda e da transparência, fiscalização e controladoria-geral da União, estabelece que os atos e os procedimentos relativos à formalização, execução, acompanhamento, prestação de contas e as informações acerca de tomadas de contas especial dos convênios e contratos de repasse devem ser registrados no Sistema de Gestão de Convênios e Contratos de Repasse (Siconv). É por meio do Sincov que um município, por exemplo, cadastra a proposta de trabalho, registra as informações sobre execução físico-financeira e a posterior prestação de contas.

Para a operacionalização de uma transferência voluntária como convênio, por exemplo, o processo de trabalho é separado em 3 etapas: 1) atos preparatórios, como manter os cadastros dos órgãos envolvidos, identificação das necessidades da sociedade, e celebração do instrumento de repasse; 2) execução, na qual são feitas os processos de compra, execução física e financeira e fiscalização da execução do objeto; e 3) prestação de contas, na qual é analisada a prestação de contas e instaurada, caso necessária, a tomada de contas especial para recuperação dos recursos repassados.

É importante ressaltar que neste trabalho não foram feitos estudos acerca das etapas 1 e 2 mas, tão somente, da etapa relativa à prestação de contas, no que diz repeito ao resultado das análises destas prestações, apesar de algumas informações utilizadas pelo modelo serem produzidas ao longo de todo o processo de trabalho. Quanto a esta etapa, é necessário esclarecer que cada órgão concedente possui discricionariedade para definir normativamente o procedimento interno adotado para análise das prestações de contas.

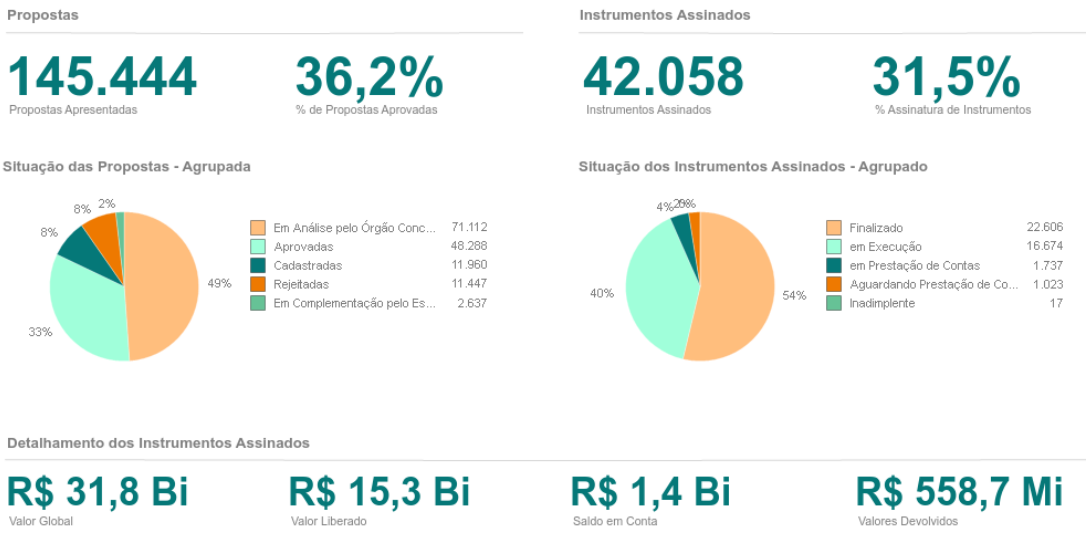
[2] Adaptado de <http://plataformamaisbrasil.gov.br/images/docs/mapeamento_de_processos/convenio/Visao_Geral-Macroprocesso-Convenio.png>

Nesse sentido, o Ministério do Desenvolvimento Regional (MDR) publicou recente Portaria 2.906/2019, a qual destaca em seu art. 2º:

> VII - análise da prestação de contas técnica: procedimento de análise do conjunto de documentos que buscam comprovar a compatibilidade entre o objeto pactuado e o executado, assim como o alcance dos resultados previstos, após a conclusão do objeto ou encerramento da vigência do instrumento; e
>
> VIII - análise da prestação de contas financeira: procedimento de análise do conjunto de documentos que buscam comprovar a conformidade da execução financeira, após a conclusão do objeto ou encerramento da vigência do instrumento.

Assim, para a emissão de parecer quanto à aprovação ou reprovação das contas das transferências voluntárias sob sua gestão, o MDR deverá analisar o conjunto de documentos técnicos e financeiros de todos os seus instrumentos. Em consulta ao painel gerencial de transferências abertas[3], de um total 42.058 instrumentos assinados, o MDR possui cerca de 2.760 instrumentos que ainda estão ou em fase de prestação de contas ou aguardando a prestação de contas. A figura 7 apresenta um panorama da situação dos instrumentos de repasse do MDR. Segundo os dados do relatório de gestão da CGU de 2018[4], naquele ano, o governo federal possuía um estoque de prestação de contas pendentes de 15,3 mil, cujo montante somava cerca de R$ 16,7 bilhões.

Figura 7 – Quantidade de instrumentos de repasse do MDR



Fonte: Plataforma +Brasil

[3] <http://plataformamaisbrasil.gov.br/paineis-gerenciais-brasil>
[4] <https://repositorio.cgu.gov.br/handle/1/38861>

3.3 *Credit Scoring*

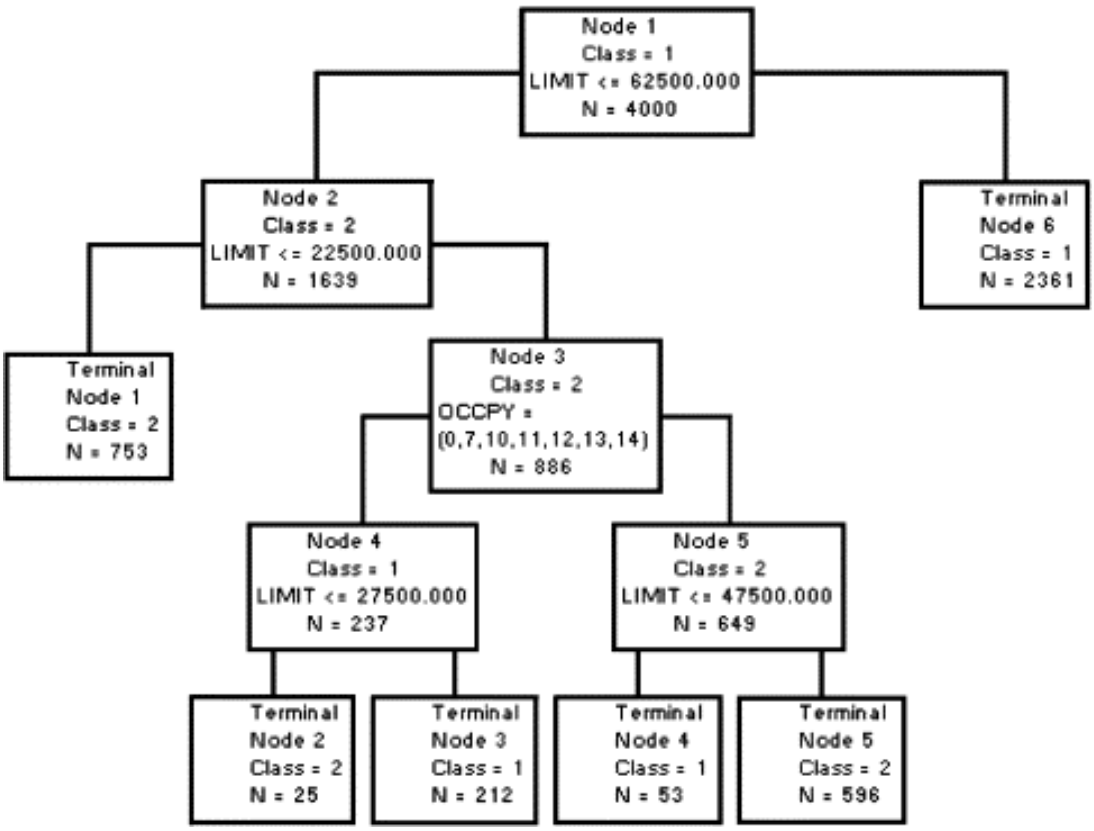
A previsão de riscos financeiros vem crescendo e ganhando importância no campo da probabilidade e estatística ao longo dos anos. Uma das áreas de análise de riscos financeiros é a pontuação de crédito, ou *credit scoring*, um conceito bastante difundido que pode ser caracterizado como um processo de modelagem da qualidade das operações de crédito por instituições financeiras (HAND; JACKA, 1998). Segundo Thomas (2000), *credit scoring* é a aplicação da previsão de risco financeiro em operações de empréstimos.

No início de sua utilização, os métodos usados para concessão de crédito eram julgamentos puramente subjetivos. Os analistas observavam características dos tomadores de empréstimo, como capital, garantias, capacidade de pagamento, e condições gerais do mercado, e tomavam suas decisões (THOMAS, 2000). Atualmente, modelos de *credit scoring* utilizados para analisar riscos financeiros utilizam várias técnicas estatísticas e de aprendizagem computacional, como regressão logística, redes neurais, máquinas de vetores de suporte e árvores de decisão (SOHN; KIM; YOON, 2016).

Independentemente das técnicas utilizadas, o problema é basicamente obter as informações coletadas nos formulários de avaliação, acompanhar o comportamento do tomador de crédito ao longo de um período predeterminado (entre 12 e 24 meses) e atribuir um rótulo de bom ou mau pagador ao mutuário, conforme seu comportamento ao longo do contrato (THOMAS, 2000). Portanto, *credit scoring* pode ser caracterizado basicamente como um problema de classificação em que se divide as respostas aos formulários em dois subconjuntos, sendo A_B aquelas respostas dos bons pagadores, e A_M aquelas cujos mutuários se mostraram maus pagadores. Novos empréstimos somente poderão ser concedidos, caso as respostas dos novos tomadores sejam pertencentes ao conjunto A_B.

Um exemplo de modelo de classificação de *credit scoring*, utilizando árvores de classificação, está representado na figura 8, que apresenta uma árvore e seus conjuntos de regras para a atribuição de um rótulo de bom ou mal pagador.

Figura 8 – Árvore do modelo CART para *credit scoring*



Fonte: (LEE et al., 2006)

Portanto, *credit scoring* é uma forma de classificar indivíduos por meio de características que estão associadas entre si, porém sem uma relação de causalidade com a classe atribuída, não sendo possível precisar seus atributos marcantes. Portanto, trata-se de um problema de classificação no qual as variáveis do modelo são as características dos tomadores de crédito, e o resultado é uma classificação entre "bons" e "maus" pagadores. Como um método automático de modelar o risco potencial de tomadores de crédito, *credit scoring* pode empregar várias técnicas estatísticas e de aprendizagem computacional em dados históricos (TIAN; YONG; LUO, 2018).

3.4 Técnicas de *Amostragem*

Entre as técnicas mais utilizadas para lidar com desbalanceamento de dados estão as técnicas de *resampling*, ou reamostragem (JUNIOR et al., 2020). As técnicas de reamostragem são usadas para rebalancear um conjunto de dados desbalanceado a fim de mitigar o efeito da distorção entre as classes no processo de aprendizagem. Os métodos de reamostragem são independentes do modelo de classificação adotado e podem ser divididas em três grupos, dependendo do método usado para equilibrar o balanceamento de classes (HAIXIANG et al., 2017):

- Métodos de superamostragem (*oversampling*): aborda o problema criando novas amostras de classes minoritárias. Dois métodos amplamente utilizados para criar amostras minoritárias sintéticas são duplicar aleatoriamente as amostras da classe minoritária e o SMOTE (*synthetic minority over-sampling technique* (CHAWLA et al., 2002).

- Métodos de subamostragem, (*undersampling*): aborda o desbalanceamento descartando amostras da classe majoritária. O método mais simples, porém mais eficaz, é o *Random Sub-Sampling* (RUS), que envolve a eliminação aleatória de exemplos da classe majoritária (TAHIR et al., 2009).

- Métodos híbridos: são uma combinação do método de superamostragem e o método de subamostragem.

Em seu estudo, Haixiang et al. (2017) mostram que reamostragem é uma estratégia popular para lidar com o desbalanceamento de dados, com vangagem para métodos de superamostragem. Métodos de superamostragem e subamostragem utilizam técnicas baseadas em *cluster* (por exemplo, k-*means*), técnicas baseados em distância (por exemplo, k-*nearest-neigbohrs*) e métodos evolutivos (por exemplo, algoritmo genético).

O SMOTE utiliza técnica de *nearest-neigbohrs*. A ideia básica é que existe uma amostra virtual positiva entre as duas amostras positivas reais que estão próximas uma da outra. Portanto, o algoritmo SMOTE tenta inventar artificialmente uma nova amostra positiva entre as duas amostras positivas reais que estão próximas. Desta forma, para cada exemplo positivo x^{pos}, deve-se encontrar m vizinho positivos próximos, $x^{pos-prox}$, e gerar com uma distância aleatória de cada exemplo positivo, um novo exemplo sintético, conforme a equação 3.1

$$x_{ik}^{pos-SMOTE} = x_i^{pos} + rand(0,1) \times (x_{ik}^{pos-prox} - x_i^{pos}) \tag{3.1}$$

onde $i \in \{1, 2, ..., S_{pos}\}, k \in \{1, 2, ..., m\}$

3.5 *Regressão Logística*

Regressão Logística é um modelo estatístico de classificação com grande popularidade, utilizado por muitos pesquisadores por sua simplicidade e transparência nas predições de seus domínios de pesquisa (DASTILE; CELIK; POTSANE, 2020). Conforme visto no capítulo 2, é um modelo de referência no domínio de riscos financeiros, em especial *credit scoring*.

Considerando a essência de problemas de classificação, o objetivo do algoritmo é encontrar superfícies de decisão, ou *decision boundaries*, que separam as observações de uma classe da outra. No caso da regressão logística, estas superfícies são hiperplanos no espaço de características do modelo, onde a dimensão desse espaço pode ser determinada pelo número de elementos no vetor de características dos exemplos de treinamento. Assim, um modelo de Regressão Logística pode ser definido como tendo a seguinte hipótese:

$$h_\beta(X) = g(\beta^T X) = g(\sum_{i=0}^{n} \beta_i x_i) \text{ onde } x_0 = 1 \tag{3.2}$$

De forma simplificada, os parâmetros do modelo de regressão logística são os pesos para as características (*features*). Para mantermos a simplicidade do nosso trabalho, em problemas de classificação binário, para que o resultado do modelo seja interpretado como a probabilidade de que um exemplo pertença a uma classe específica, é necessária uma restrição para um valor entre 0 e 1, feita por meio de uma função logística em formato de S. A aplicação da função logística à equação 3.2 é apresentada a seguir:

$$g(z) = \frac{1}{1 + e^{(-z)}} \tag{3.3}$$

O algoritmo de aprendizado ajusta os pesos para classificar corretamente os exemplos de treinamento. Para realizar a estimativa desses parâmetros do modelo, usualmente é feita uma estimativa por máxima verossimilhança, ou *maximum likelihood estimation*. O método de descida de gradiente e outras variantes são populares para ajustar os pesos. A função custo para a estimativa dos parâmetros é apresentada na equação a seguir:

$$J(\beta) = \frac{1}{m} \sum_{i=1}^{m} [-y_i log(h_\beta(x_i) - (1 - y_i) log(1 - h_\beta(x_i))] \tag{3.4}$$

Se considerarmos a utilização do método de gradiente descendente para a solução da equação 3.4, nosso problema de otimização é a solução da seguinte equação:

$$\frac{dJ(\beta)}{d\beta} = \frac{1}{m} \sum_{i=1}^{N} x_i (h_\beta(x_i) - y_i) = 0 \tag{3.5}$$

Devido à simplicidade da hipótese de superfícies de decisão lineares, a regressão logística é muitas vezes um dos primeiros algoritmos a serem usados para problemas de classificação. Além disso, devido às superfícies de decisão serem lineares e não complexas, sabe-se que a regressão logística é menos propensa ao sobreajuste, ou *overfit* (GUDIVADA

et al., 2016). Além disso, o algoritmo de descida de gradiente é de fácil convergência, tornando a fase de treinamento da regressão logística rápida, o que justifica a aplicação popular da regressão logística a uma variedade de problemas de classificação. Por outro lado, a simplicidade das hipóteses de superfícies de decisão lineares podem levar a um ajuste insuficiente, ou *underfit* para conjuntos de dados mais complexos.

3.6 *Multilayer Perceptron*

Redes neurais artificiais (RNA) são estruturas computacionais que tentam simular, a grosso modo, as redes de células nervosas (neurônios) do sistema nervoso central do ser humano (MUKHOPADHYAY, 2011). *Multilayer Perceptron* (MLP) é uma rede neural *feed-forward* e, de forma similar à regressão logística, possui a seguinte hipótese:

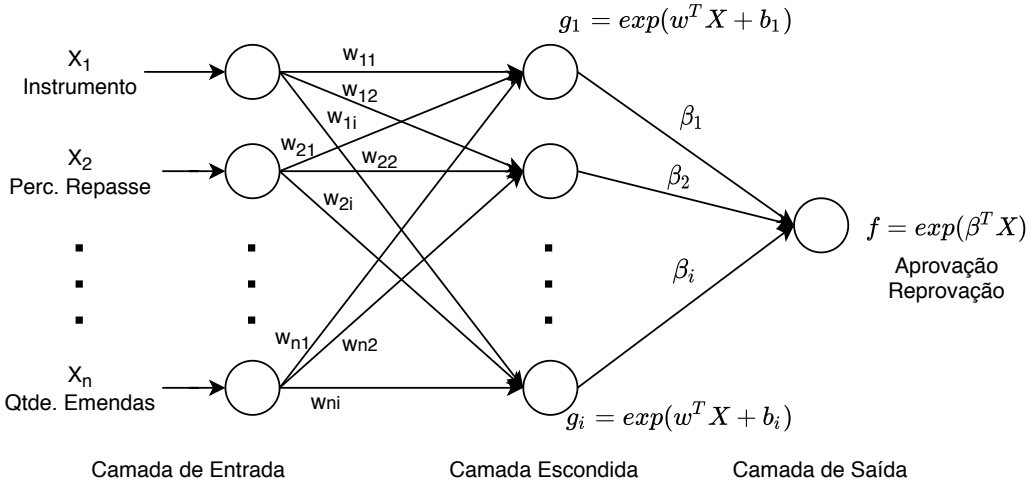
$$f(X) = \sum_{i=0}^{L} G_i(X, w_i, b_i) * \beta_i, \text{ com } w_i \in \mathbb{R}^d; b_i, \beta_i \in \mathbb{R} \tag{3.6}$$

Para redes *feed-forward*, G_i é a função de ativação do nó, e assume a forma

$$G_i(X, w_i, b_i) = g(w_i * X + b_i) \tag{3.7}$$

A representação de uma rede neural artificial *feed forward* é apresentada na figura 9, a qual apresenta n nós na camada de entrada à esquerda, interligados aos nós da camada escondida com seus respectivos pesos, que, por sua vez, está interligada à camada de saída.

Figura 9 – Esquema simplificado de uma rede MLP.



Autoral

Para aprender, as redes *feed forward* também utilizam o método de descida de gradiente, e utilizam o algoritmo de *backpropagation* para ajustar todos os pesos dos seus nós. O algoritmo de *backpropagation* calcula o gradiente da função de perda com relação aos pesos e propaga pela rede, por meio da regra da cadeia, os pesos na direção do gradiente mais íngreme. Considerando uma superfície de erro relativamente suave, espera-se que os pesos convirjam para um mínimo da superfície de erro.

O algoritmo de *backpropagation* pode ser resumido conforme abaixo (BISHOP, 1995).

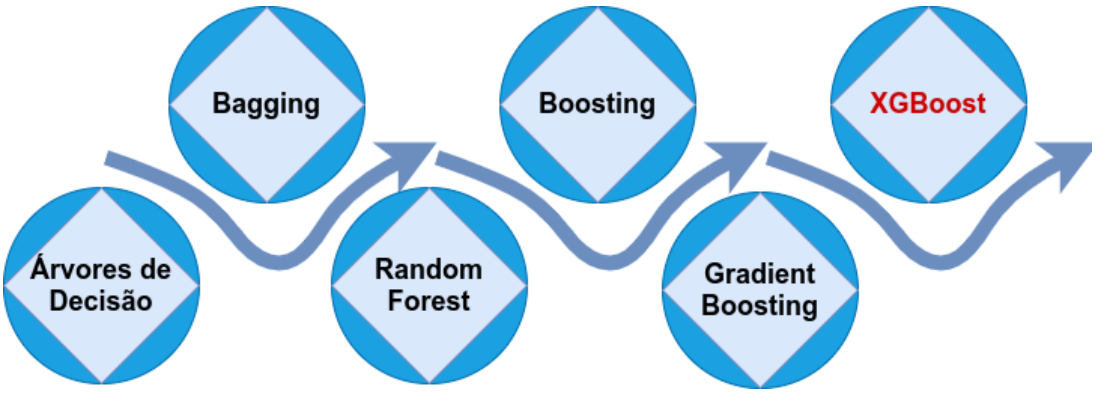
1. inicializar os pesos da rede;
2. apresentar o vetor de entrada, dos dados de treinamento à rede;
3. propagar o vetor de entrada pela rede para obter um valor de saída;
4. calcular o erro comparando a saída obtida com o valor real;
5. propagar o sinal de erro de volta pela rede;
6. ajustar os pesos para minimizar o erro geral;
7. repetir os passos 2 a 6 com o próximo vetor de entrada, até que o erro atinja um valor aceitável.

3.7 *eXtreme Gradient Boosting*

O algoritmo *XGBoost* foi originalmente proposto por Chen e Guestrin (2016). Na publicação original, os autores atribuem o sucesso de aplicações que fazem uso da aprendizagem de máquina a dois fatores: uso de modelos efetivos (estatísticos) que capturam as dependências complexas dos dados e sistemas de aprendizado escaláveis que aprendem o modelo de interesse a partir de grandes conjuntos de dados.

O *XGBoost* é um modelo de aprendizagem de máquina baseado em árvore, que aperfeiçoa a técnica de *tree boosting*. Para melhor entendermos o funcionamento deste método, é importante que vejamos como se deu o processo evolutivo destes tipos de modelo.

Figura 10 – Evolução do XGBoost



Fonte: Autoral

As árvores de decisão ainda são amplamente utilizadas para aprendizagem computacional. Elas possuem simples entendimento e apresentação do modelo proposto, que é facilmente compreendido por seres humanos. Por ser não-paramétrico, seus algoritmos não necessitam de informações prévias acerca da distribuição dos dados, sendo adequados para a descoberta de conhecimento, além de possuírem bom desempenho na construção do modelo (BREIMAN et al., 1984; MEHTA; AGRAWAL; RISSANEN, 1996).

Desde o surgimento das árvores de decisão, com os primeiros algoritmos ID3 (QUINLAN, 1986) e, posteriormente, o *Classifier 4.5* (C4.5) (QUINLAN, 1993), houve grande evolução na área. Árvores de decisão podem ser compreendidas como um conjunto de critérios utilizado para separar os dados de acordo com suas características, resultando em regras para classificar determinada instância em um ou mais grupos, ou prever um determinado valor. Ocorre que para muitos domínios, as árvores de decisão, utilizadas isoladamente, não apresentam os melhores resultados, podendo ser classificadas como *weak learners*.

Por isso, muitos pesquisadores criaram meios de arranjar várias árvores, com o objetivo de melhorar a performance de seus modelos. Esses arranjos (*ensembles*) podem ser divididos em *bagging* e *boosting*.

Na técnica de *bagging*, acrônimo de *bootstrap aggregating*, a ideia é criar vários subconjuntos aleatórios dos dados de treinamento, ou *bootstraps*. Cada subconjunto é usado para treinar uma árvore de decisão. Como resultado deste treinamento, acabamos com um conjunto de árvores, ou modelos, diferentes. A média de todas as previsões das diferentes árvores é usada como resultado, que é mais robusto que uma única árvore de decisão (BREIMAN, 1996).

Boosting, por sua vez, é outra técnica de arranjo para criar uma coleção de árvores. Nessa técnica, porém, os modelos são aprendidos sequencialmente com os *learners* iniciais, que podem ser árvores, se ajustando aos dados, e os demais buscando minimizar os erros

de seus antecessores, ajustando as árvores consecutivamente a cada passo (FREUND; SCHAPIRE, 1995).

Vários algoritmos de *boosting* foram propostos, sendo os mais comuns:

- *AdaBoost* (FREUND; SCHAPIRE, 1995)
- *Gradient boosting machine* (FRIEDMAN; HASTIE; TIBSHIRANI, 2001)
- *Stochastic gradient boosting* (FRIEDMAN, 2002); e
- *XGBoost* (CHEN; GUESTRIN, 2016).

Em problemas de aprendizagem preditiva, ou estimativa de funções, é frequente a utilização de exemplos de treinamento $\{y_i, x_i\}_1^N$ com valores conhecidos de (x, y) para obter uma aproximação $\hat{F}(x)$, da função verdade $F^*(x)$ que mapeia x para y, por meio da minimização de uma função de perda ou custo $L(y, F(x))$, que podem incluir erros quadráticos ou *log likelihood* a depender do problema a ser solucionado. É comum a restrição na escolha de $F(x)$ como sendo de uma classe de funções parametrizadas $F(x; \mathbf{P})$, onde $\mathbf{P} = \{P_1, P_2, ...\}$, o que torna o problema de aproximação em um de otimização dos parâmetros

$$\mathbf{P}^* = \arg\min_{\mathbf{P}} \Phi(\mathbf{P}) \tag{3.8}$$

onde

$$\Phi(\mathbf{P}) = E_{x,y} L(y, F(x; \mathbf{P})) \tag{3.9}$$

de forma que

$$\mathbf{P}^* = \sum_{m=0}^{M} p_m \tag{3.10}$$

sendo p_0 uma inicialização dos parâmetros e $\{p_m\}_1^M e$ são sucessivos incrementos, passos ou *boosts*, realizados sequencialmente após os passos anteriores, e são calculados conforme o método de otimização adotado.

No *XGBoost*, os autores, ao revisarem a literatura acerca de técnicas de *gradient tree boosting*, em especial (FRIEDMAN; HASTIE; TIBSHIRANI, 2000), propuseram melhorias na função objetivo regularizada. Para um determinado conjunto de dados com n exemplos

e m características, $D = \{(x_i, y_i)\}$ ($|D| = n, x_i \in \mathbb{R}^m, y_i \in \mathbb{R}$), um arranjo de árvores de decisão usa K funções aditivas para prever um resultado:

$$\widehat{y}_i = \phi(x_i) = \sum_{k=1}^{K} f_k(x_i), f_k \in \digamma \tag{3.11}$$

onde $\digamma = \left\{f(x) = w_{q(x)}\right\} \left(q : \mathbb{R}^m \to T, w \in \mathbb{R}^T\right)$ é o espaço de árvores de regressão. Nesta representação, q é a estrutura de cada árvore que mapeia um exemplo para o índice de folha correspondente. T é o número de folhas na árvore, ou a complexidade do modelo. Cada f_k corresponde a uma estrutura de árvore independente q com peso w em suas folhas. Como as árvores de regressão possuem um *score* em cada folha, é utilizado w_i para representar o *score* da i-ésima folha. A previsão é a soma dos *scores* de cada folha, classificadas conforme as regras de q.

Para aprender o conjunto de funções usadas no modelo, deve-se minimizar a seguinte função objetivo regularizada.

$$\mathcal{L}(\phi) = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k) \tag{3.12}$$

onde $\Omega(f) = \gamma T + \frac{1}{2}\lambda \|w\|^2$.

A proposta dos autores é utilizar aproximação de segunda ordem para otimizar a função descrita em (3.12). Como resultado, expandindo Ω, eles obtêm:

$$\widehat{\mathcal{L}}^{(t)} = \sum_{j=1}^{T}[(\sum_{i \in I_j} g_i)w_j + \frac{1}{2}(\sum_{i \in I_j} h_i + \lambda)w_j^2] + \gamma T \tag{3.13}$$

Por fim, para uma estrutura *q(x)* fixa, o cálculo do peso w_j^* ideal para uma folha j é dado por

$$w_j^* = -\frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda} \tag{3.14}$$

E o seu correspondente valor ótimo, o qual é usado como *scoring* de cada estrutura *q(x)*.

$$\widehat{\mathcal{L}}^{(t)}(q) = -\frac{1}{2}\sum_{j=1}^{T}\frac{(\sum_{i \in I_j} g_i)^2}{\sum_{i \in I_j} h_i + \lambda} + \gamma T \tag{3.15}$$

Além da função objetivo regularizada, os autores propõem duas técnicas adicionais para evitar o *overfitting*. O *shrinkage*, ou encolhimento, introduzido por (FRIEDMAN, 2002). O encolhimento dimensiona os pesos recém-adicionados por um fator η após cada etapa de aumento da árvore. A outra técnica é a subamostragem de características.

Um dos principais problemas na aprendizagem por árvores é encontrar a melhor divisão. Desta forma, também foi proposto um algoritmo *exact greedy* de busca de *splits* que procura por todas as possíveis divisões em todas as características do conjunto de dados.

Este algoritmo se mostrou bastante robusto, pois busca de forma agressiva todos os possíveis pontos de divisão das árvores. Porém, possui baixa eficiência quando o conjunto de dados é muito grande, pois necessitaria de grandes quantidades de memória. Para contornar a situação, um algoritmo de aproximação é utilizado.

O XGBoost se mostrou bastante robusto no processamento de grandes massas de dados usando recursos computacionais limitados. De acordo com os resultados de seus experimentos, os autores obtêm um aprendizado escalável, explorando a computação paralela e distribuída.

3.8 Otimização de hiperparâmetros

O objetivo final de um problema de aprendizado de máquina é determinar uma relação na forma $y = G(x, \theta)$, onde y é a saída, x é o vetor de entrada e a função G é parametrizada por um vetor θ. Por exemplo, um dos parâmetros em uma árvore de decisão são os pesos dos nós folha, que podem ser aprendidos por algum método de otimização, como descida de gradiente. No entanto, o próprio algoritmo de aprendizado também pode ter parâmetros que não são aprendidos diretamente dos dados de entrada e precisam ser definidos. Esses parâmetros são chamados de hiperparâmetros porque são considerados de nível superior em relação aos parâmetros estimados a partir dos dados de entrada. A profundidade da árvore ou valor mínimo dos pesos são hiperparâmetros típicos para árvores de decisão. De um modo geral, os algoritmos de aprendizado aprendem parâmetros do modelo, enquanto os hiperparâmetros controlam a forma com que o modelo irá aprender (XIA et al., 2017).

Os valores escolhidos para os hiperparâmetros de um modelo podem impactar diretamente a performance preditiva de algoritmos de aprendizagem de máquina. Para um mesmo conjunto de dados de treinamento, com diferentes hiperparâmetros, um algoritmo pode aprender modelos com desempenhos significativamente diferentes no conjunto de dados de teste (WANG; GONG, 2018). Portanto, a escolha de hiperparâmetros de um modelo é de extrema importância para que se alcance bons resultados.

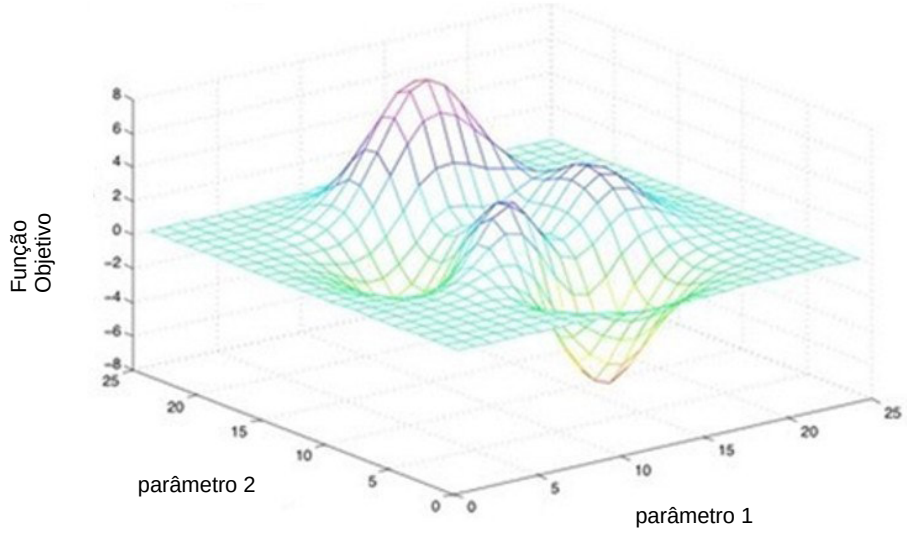
A otimização de hiperparâmetros é um processo que tenta encontrar os melhores valores para um conjunto de parâmetros de um algoritmo de aprendizagem de forma a gerar um modelo que apresente melhor performance. O objetivo de um modelo de aprendizagem $\mathcal{M}$ é minimizar a função perda $\mathcal{L}(X^{test}; \mathcal{M})$ sobre os dados de teste. O modelo $\mathcal{M}$, então, é construído por meio de um algoritmo de aprendizagem $\mathcal{A}$ utilizando os dados de treinamento X^{treino}. Por seu turno, o algoritmo de aprendizagem $\mathcal{A}$ pode ser parametrizado por um conjunto de hiperparâmetros $\lambda \in \Lambda$, de forma que $\mathcal{M} = \mathcal{A}(X^{treino}; \lambda)$, onde Λ é o espaço de busca de hiperparâmetros. Desta forma, o objetivo da otimização de hiperparâmetros é achar um conjunto λ^* que resulte no melhor modelo $\mathcal{M}^*$, que minimize $\mathcal{L}(X^{test}; \mathcal{M})$. Desta forma

$$\lambda^* = \underset{\boldsymbol{\lambda}}{\operatorname{argmin}}\ \mathcal{L}(X^{teste}; \mathcal{A}(X^{treino}; \lambda)) = \underset{\boldsymbol{\lambda}}{\operatorname{argmin}}\ F(\lambda; \mathcal{A}, X^{treino}, X^{teste}, \mathcal{L}) \tag{3.16}$$

Onde a função objetivo F recebe o conjunto de parâmetros λ e retorna o erro associado calculado pelo treinamento do modelo $\mathcal{M}$ gerado pelo algoritmo de treinamento $\mathcal{A}$. A avaliação da função objetivo F requer a avaliação da performance do modelo com cada conjunto de parâmetros λ, o que, a depender da complexidade do problema e da quantidade de dados (X^{treino}, X^{teste}) disponível, pode consumir bastante tempo. Associado a essa dificuldade, alguns algoritmos de treinamento podem possuir um conjunto significativo de hiperparâmetros o que torna o seu espaço de busca complexo e custoso (CLAESEN; MOOR, 2015). Entre os métodos disponíveis para a realização da tarefa de busca no espaço de hiperparâmetros, três são mais comuns: pesquisa em grade, pesquisa aleatória e otimização Bayesiana (SNOEK; LAROCHELLE; ADAMS, 2012).

A busca em grade, também conhecida por *grid search*, pode se tornar um método muito custoso, porém eficaz em pesquisar todo o espaço de busca de hiperparâmetros. Considerando que λ é um conjunto indexado por K parâmetros de configuração (por exemplo, para árvores de decisão poderia ser a taxa de aprendizado, profundidade máxima da árvore, peso mínimo de nós folha, etc.), a pesquisa em grade exige que se escolha um conjunto de valores para cada variável $(L^1), L^2, ..., L^K)$. Assim, o número de tentativas é formado pela combinação de todos os possíveis valores das variáveis, e é composto por $S = \prod_{k=1}^{K} L_k$ elementos. Esse produto sobre K conjuntos faz com que a pesquisa em grade sofra com a maldição da dimensionalidade pois esse valor cresce exponencialmente com o número de hiperparâmetros (BERGSTRA; BENGIO, 2012). Um exemplo ilustrativo é apresentado na Figura 11, em que é considerada a busca pelos valores dos parâmetros I e II. Na imagem, o plano inferior representa o espaço de busca e os picos e vales as regiões onde os resultados são piores e melhores, respectivamente.

Figura 11 – Exemplo de espaço de busca explorado pelo método busca em grade.



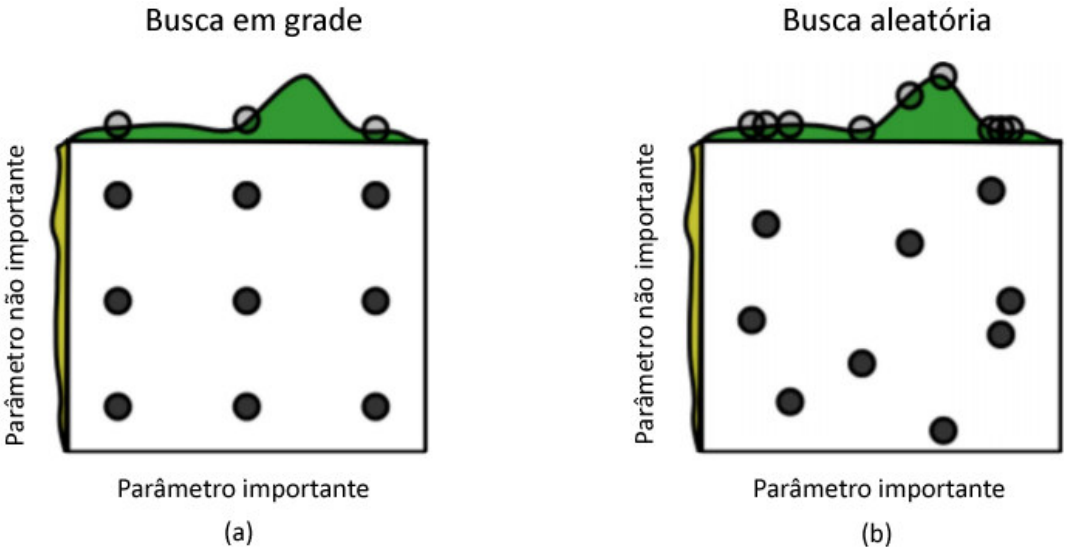
Fonte: (DANGETI, 2017)

Por outro lado, a busca aleatória (do inglês, *random search*), proposta por Bergstra e Bengio (2012) é mais simples e rápida, tendo os autores demonstrado a ineficiência da busca em grade em relação à busca aleatória, porém esta pode não explorar importantes regiões do espaço de busca. Nesse método, a diferença em relação à busca em grade é a realização de sorteios independentes de uma densidade uniforme do mesmo espaço de busca utilizado por uma grade regular, mantendo a mesma simplicidade, porém melhorando a eficiência em espaços de busca de alta dimensionalidade.

Bergstra e Bengio (2012) apontam que a pesquisa aleatória é mais eficiente que a pesquisa em grade em espaços de alta dimensionalidade pois a função objetivo F tem uma baixa dimensionalidade efetiva, ou seja, é mais sensível a mudanças em algumas dimensões do espaço de busca do que outras. Eles afirmam que se uma função f de duas variáveis pode ser aproximada por outra função g de uma variável ($f(x_1, x_2) \approx g(x_1)$), logo a função f tem uma baixa dimensionalidade efetiva.

A figura 12 mostra como a situação descrita acima é atacada pelas duas abordagens, em um espaço de busca de dois hiperparâmetros. A função a ser otimizada toma a forma $f(x, y) = g(x) + h(y) \approx g(x)$, com baixa dimensionalidade efetiva. Acima de cada espaço de busca está a função $g(x)$ e ao lado esquerdo a função $h(x)$. Percebe-se claramente que a função $g(x)$ exerce maior influência no espaço de buscas, e é mais bem explorada pela abordagem de busca aleatória.

Figura 12 – Métodos busca em grade (a) e busca aleatória (b)



Fonte: Adaptado de (BERGSTRA; BENGIO, 2012)

Por fim, otimização sequencial baseada em modelo, ou sequential model-based optimization (SMBO) e, mais precisamente, a otimização Bayesiana, é uma abordagem probabilística eficaz para otimização de funções objetivas do tipo caixa preta, as quais você não tem conhecimento da sua forma (convexidade, linearidade, etc.) e tem se tornado a abordagem padrão para otimização de hiperparâmetros (CANDELIERI; ARCHETTI, 2019). A otimização SMBO é composta, basicamente, por três componentes. Primeiro, um modelo delegatário de regressão probabilístico, que prevê a performance de cada possível configuração de hiperparâmetros. Segundo, uma função de aquisição, que usa o modelo delegatário para propor a nova configuração de parâmetros a ser avaliada. Em terceiro, opcionalmente, uma técnica de inicialização que utiliza parâmetros previamente vencedores em problemas anteriores (WISTUBA; SCHILLING; SCHMIDT-THIEME, 2015).

De acordo com Shahriari et al. (2016) vários modelos de regressão probabilística podem ser utilizados no contexto da otimização bayesiana:

- Processos Gaussianos: método de aprendizado de máquina baseado em *kernel* para problemas de regressão não linear. Para inferir uma relação funcional desconhecida de um conjunto de dados de treinamento, é assumida uma distribuição *a priori* do tipo gaussiana, para restringir as suas possíveis formas, e atualizando seus formato á medida que exemplos de treinamento são apresentados, gerando um processo gaussiano *a posteriori*. Enquanto a função *kernel* controla a suavidade e a amplitude das amostras do GP, a média *a priori* fornece um possível *offset* (Shahriari et al., 2016).

- *Random Forests*: conjunto de árvores de decisão com boas performances tanto

para predições lineares, quanto não-lineares, encontrando um equilíbrio entre viés e variância. Esses conjuntos podem ser construídos de forma que aprendam a ignorar variáveis inativas. Neste método de aprendizado, o crescimento ocorre de forma conjunta, por meio do empilhamento de árvores sucessivas e não dependem de árvores anteriores. Cada árvore é determinada de forma independente usando uma amostra do conjunto de dados e uma votação por maioria simples é tomada para a previsão final (HULTQUIST; CHEN; ZHAO, 2014).

- *Tree-structured Parzen Estimator* (TPE): O TPE é uma abordagem sequencial de otimização baseada em modelos. Tais métodos constroem sequencialmente modelos para aproximar o desempenho de hiperparâmetros com base em medições históricas e, em seguida, selecionam novos hiperparâmetros para teste com base nesse modelo. A abordagem do TPE modela $P(x|y)$ e $P(y)$ onde x representa os hiperparâmetros e y, o índice de qualidade associado. $P(x|y)$ é modelado pela transformação do processo gerador de hiperparâmetros, substituindo as distribuições da configuração anterior por densidades não paramétricas (BERGSTRA et al., 2011).

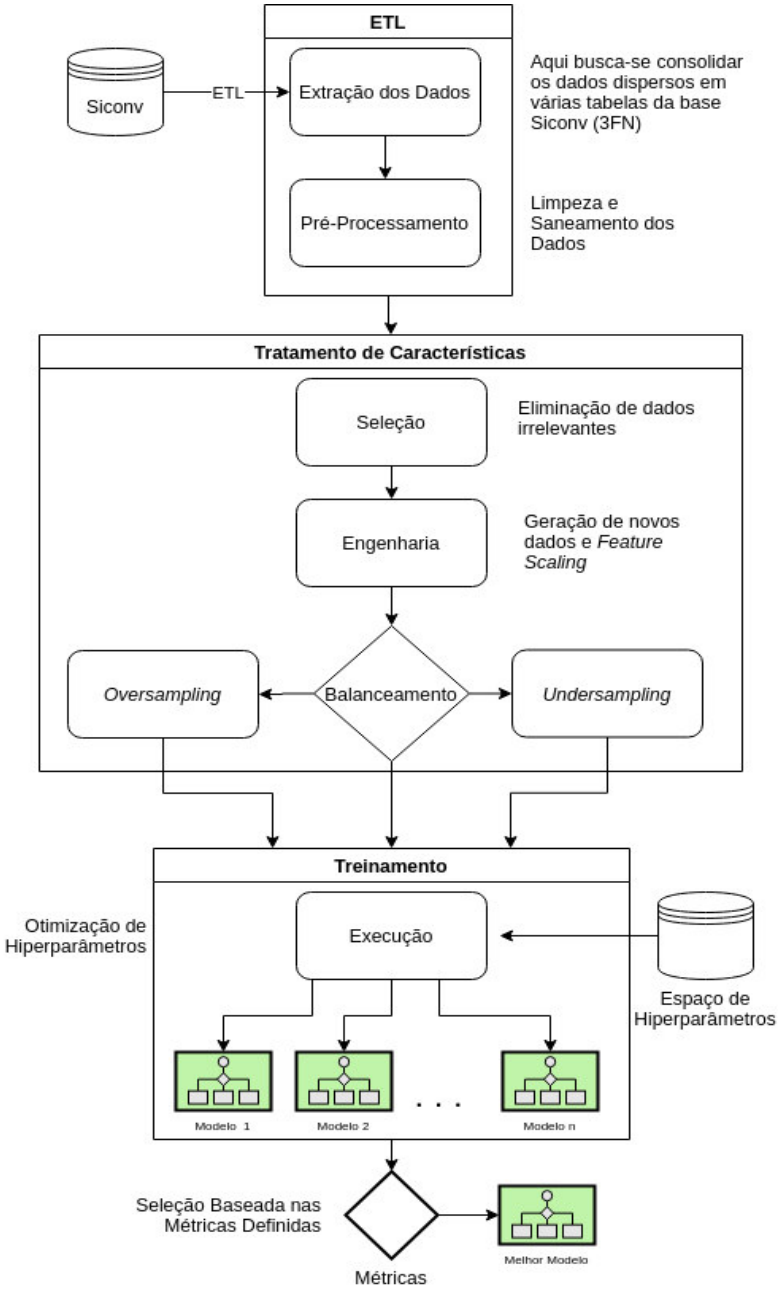
Com relação às funções de aquisição, vários métodos podem ser utilizados, tais como amostragem de Thompson, probabilidade de melhoria, expectativa de melhoria (EI), limites superiores de confiança e pesquisa por entropia. Geralmente essas funções de aquisição fazem um *trade off* entre investigação (exploration) - nas quais os pontos ótimos estão localizados onde a incerteza no modelo delegatário é grande - e exploração (exploitation) - nos quais a previsão do modelo é alta (exploração). Os algoritmos de otimização bayesiana selecionam o próximo ponto de busca do espaço maximizando essas funções de aquisição (Shahriari et al., 2016).

Este capítulo apresentou os principais embasamentos teóricos que foram adotadas tanto na metodologia quanto na avaliação dos resultados alcançados. A metodologia proposta, bem como seu detalhamento, é apresentado na capítulo a seguir.

4 Metodologia

A metodologia deste trabalho é uma adaptação da metodologia CRISP-DM, e foi dividida em três etapas: extração dos dados; seleção e pré-processamento dos dados; execução e otimização de hiperparâmetros do algoritmo *XGBoost* para obtenção dos resultados de classificação de riscos. Os passos da metodologia proposta são apresentados na figura 13 e detalhados nas seções posteriores.

Figura 13 – Fluxo da metodologia proposta



Fonte: Autoral

As duas primeiras etapas da metodologia estão relacionadas ao processo conhecido como ETL (*Extraction-Transformation-Loading*) em sistemas de *Business Intelligence* (BI), que realiza a extração dos dados da origem, realizando as transformações e calculando novos dados quando necessário, isolando e limpando dados que apresentem problemas e carregando esses novos dados em uma base de destino (VASSILIADIS, 2009).

4.1 Extração dos Dados

A primeira etapa da metodologia proposta é uma adaptação das etapas de entendimento do negócio, compreensão dos dados e preparação dos dados do modelo CRISP-DM. A etapa de entendimento do negócio já foi abordada com detalhes na seção 3.2. Ao longo do processo de celebração e execução dos instrumentos de repasse de recursos, várias informações são produzidas. Algumas destas informações são conhecidas *a priori*, antes do encerramento do instrumento, como as informações do proponente, valores pactuados, datas inciais previstas, etc. Outras informações só serão conhecidas ao longo da execução do instrumento ou quando do momento da prestação de contas, como a quantidade de prorrogações e termos aditivos, data final de prestação de contas, etc. Os dados utilizados neste trabalho são os dados de celebração e acompanhamento dos instrumentos de repasse disponíveis publicamente para *download* na Plataforma +Brasil (ECONOMIA, 2019b).

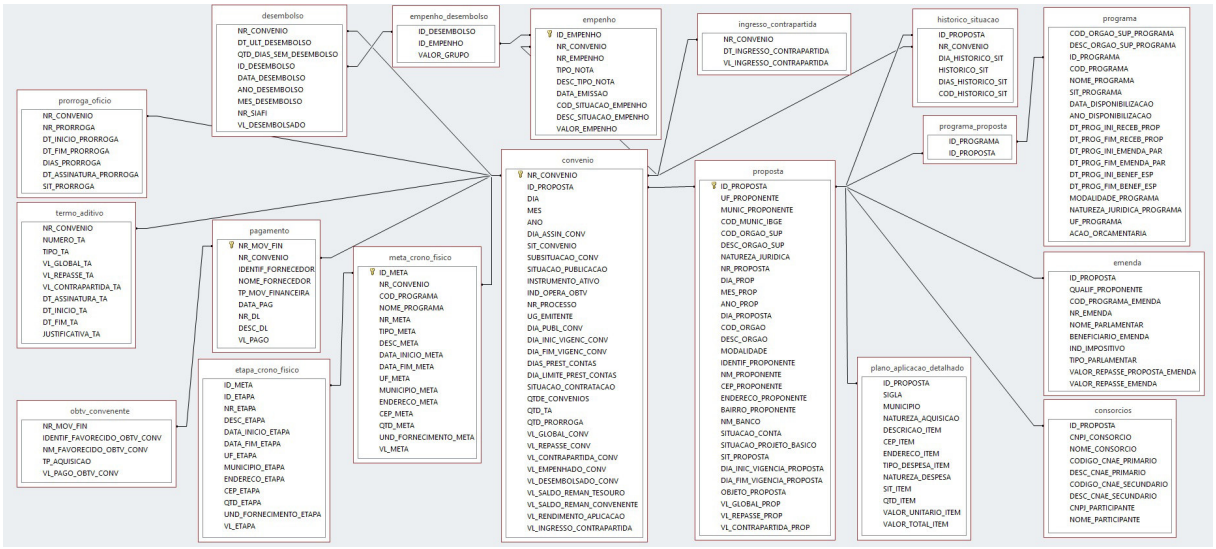
Inicialmente, foi necessário um melhor entendimento do modelo de negócios utilizado pelo sistema, a fim de se compreender o papel de cada informação disponibilizada pela plataforma. Buscou-se, nesta etapa, uma maior compreensão do modelo de dados[1] utilizado e identificar o universo de características a serem utilizadas pelo modelo aqui desenvolvido, a fim de realizar a extração dos dados.

Os dados são disponibilizados em formato CSV (SHAFRANOVICH, 2005) e foram importados para uma base de dados MySQL, replicando o modelo de dados originalmente utilizado pelo sistema, utilizando-se *scripts* em SQL para juntar informações que estavam normalizadas nas várias tabelas do banco de dados em uma única tabela.

O modelo de dados é composto por 22 tabelas, das quais se destacam as tabelas "convenio", "proposta" e "proponentes", de onde foram retiradas as principais características utilizadas, dentre as 212 disponíveis. Como os dados estão modelados na 3ª Forma Normal (CODD, 1970), foi preciso realizar um processo de desnormalização do modelo de dados a fim de se ter, em cada linha, todas as informações referentes a um instrumento, as quais serão utilizadas para treinamento e teste do modelo.

[1] <http://plataformamaisbrasil.gov.br/images/docs/CGSIS/modelo_dados_siconv.zip>

Figura 14 – Modelo de dados do Siconv



Fonte: (ECONOMIA, 2019b)

Muitos dos dados disponíveis são informações meramente administrativas, como identificação dos vários atores do processo, e outros são referentes às fases internas de execução do instrumento de repasse, como etapas do projeto e descrição dos itens de cada etapa, e não foram incluídas no escopo deste trabalho, por terem relação direta com a etapa de prestação de contas.

Após a etapa de extração dos dados, procedeu-se à etapa de pré-processamento e seleção, descritos na próxima seção.

4.2 Pré-processamento e Seleção

A base de dados de transferências voluntárias utilizada[2] possui um total de 172.366 instrumentos cadastrados. Destes, muitos ainda estão em fase inicial de execução, ou ainda estão em ciclo interno de análise das prestações de contas. Para a condução dos experimentos, foi utilizado apenas um subconjunto dos instrumentos disponíveis, conforme descrito a seguir.

Durante a etapa de pré-processamento buscou-se verificar a qualidade dos dados extraídos, fazendo-se a sua limpeza e saneamento, e removendo-se os registros com datas de fim anteriores às datas de início dos instrumentos e dados de valores sem preenchimento (valores iguais a 0) ou com valores negativos, possivelmente por problemas de preenchimento.

Considerando as transferências voluntárias como um problema de *credit scoring*,

[2] Siconv - Dados disponíveis em 17/07/2019

conforme abordado no capítulo anterior, e considerando as principais características utilizadas para modelagem deste tipo de problema (THOMAS, 2000), foram extraídas as seguintes informações para desnormalização dos dados:

- Unidade da Federação do Proponente;
- Município do Proponente;
- Órgão Superior vinculador;
- Órgão executor (concedente);
- Modalidade;
- Situação da conta bancária;
- Situação do projeto básico;
- Quantidade de termos aditivos e prorrogações;
- Datas de início e fim da vigência do instrumento;
- Mês de assinatura do instrumento;
- Valores pactuados;

O universo de transferências utilizadas como sucesso e fracasso foram aquelas em que já houve a manifestação completa do ciclo interno de controle, com os respectivos pareceres emitidos. É importante ressaltar que os casos de interesse, os casos positivos, são aqueles em que houve manifestação do controle pela reprovação das contas. Para os casos negativos, o parecer do controle foi pela aprovação das contas.

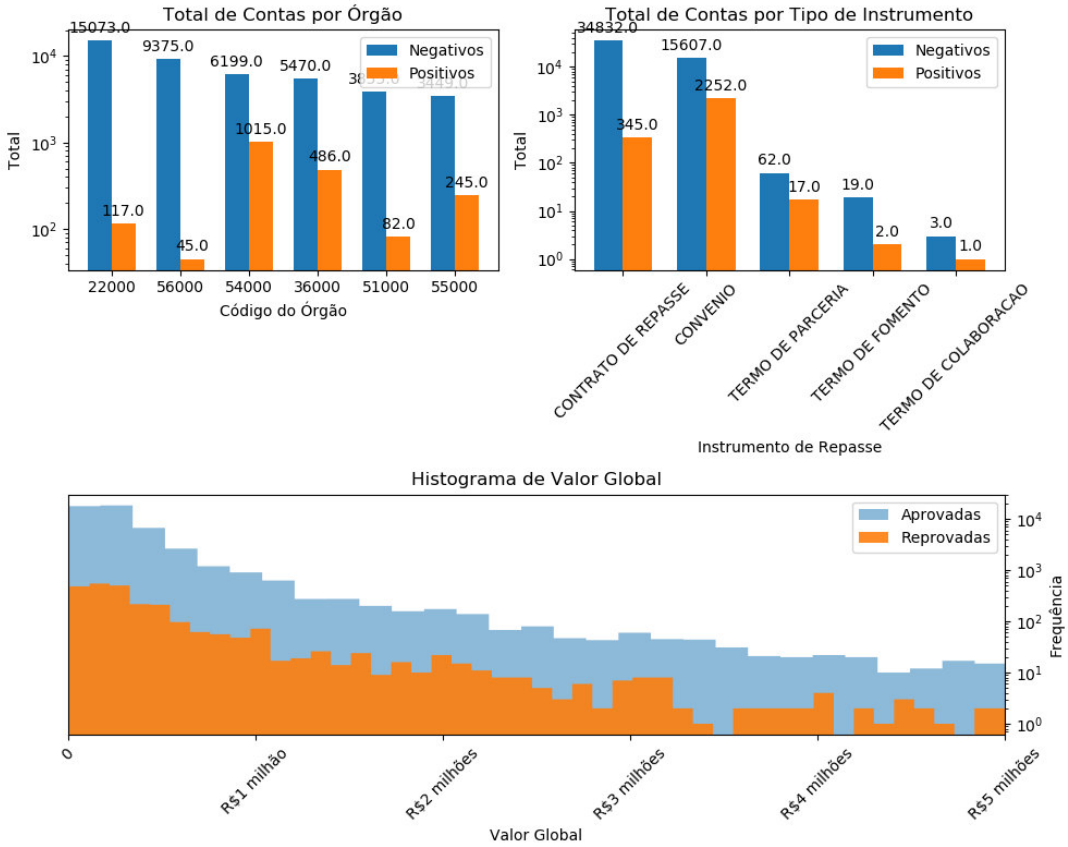
Para essas transferências, foram consideradas negativas aquelas em que a manifestação do controle interno foi "prestação de contas aprovada" ou "prestação de contas aprovada com ressalvas". Quanto aos casos positivos, foram consideradas as contas com manifestação "prestação de contas rejeitada", "inadimplente" ou "convenio rescindido".

Deste universo, após a etapa de saneamento dos dados, resultou um total de 53.581 transferências voluntárias com manifestação conclusiva. Destas, 51.353 com pareceres pela aprovação das contas (negativos) e 2.228 pela reprovação (positivos). O que se percebe, de pronto, é o grande desbalanceamento que há entre exemplos positivos e negativos, sendo aqueles em proporção de mais de 23 vezes superior a estes.

A seguir, um resumo gráfico de algumas variáveis extraídas são apresentados na figura 15. Apresenta-se uma estratificação de casos positivos e negativos por código do

órgão e por tipo de instrumento de repasse, evidenciando concentração de casos em determinados órgãos ou tipo de instrumentro.

Figura 15 – Análise gráfica de variáveis



Fonte: Autoral

É importante ressaltar que nesta etapa também foi aplicado um processo de transformação de alguns dados, para geração de novas *features*. É o caso das informações relativas a emendas e consórcios. Apesar de estarem disponíveis, eram informações analíticas, que continham muitos dados acerca da tramitação das emendas ou composição dos consórcios. Para esses casos, foram criados campos que indicavam se determinado instrumento derivou de emendas orçamentárias, visto que podem ser recursos direcionados que acabem não trazendo os resultados adequados, e a quantidade de órgãos que compunham o consórcio.

Outra transformação necessária diz respeito aos valores dos instrumentos, os quais apresentam grande variância. Há instrumentos de repasse de R$ 2.000,00 até R$ 170 milhões. Considerando que estes valores podem, de alguma forma, enviesar o modelo favorecendo aqueles em que os valores são mais vultosos em detrimento dos demais, buscou-se uma normalização destes valores, fazendo com que os valores de todos os instrumentos ficassem na faixa entre 0 e 1. Porém, entendendo que, de fato, pode haver significado embutido nos valores pactuados, foram criadas faixas de valores, para tentar capturar essa

informação. Para esta finalidade, foram utilizados os valores definidos pelo Decreto nº 9.412/2018, conforme a tabela 1.

Tabela 1 – Faixa de valores adotada para criação de nova característica.

	Faixa de Valor
Dispensa	R$ 0 < Valor ≤ R$ 33 mil
Convite	R$ 33 mil < Valor ≤ R$ 330 mil
Tomada de Preços	R$ 330 mil < Valor ≤ R$ 3,30 milhões
Concorrência	Valor > R$ 3,3 milhões

Apesar de a técnica adotada nesta metodologia (*XGBoost*), que utiliza *gradient descent*, não sofrer de problemas relativos a *vanishing gradients* (XIA et al., 2017), como iremos compará-la a modelos de regressão logística e redes neurais artificiais, que podem sofrer problemas de gradiente (IOFFE; SZEGEDY, 2015), procedeu-se à normalização dos dados relativos aos dados não categóricos, como valores, número de prorrogações (termos aditivos) e quantidade de órgãos de consórcio utilizando-se a técnica de normalização Min-Max (SINGH; VERMA; THOKE, 2015), com $min = 0$ e $max = 1$, utilizando a fórmula definida na equação

$$\hat{x}_i = \frac{x_i - min(x_i)}{max(x_i) - min(x_i)} * (maxx_{new} - minx_{new}) + minx_{new} \tag{4.1}$$

onde $min_{new} = 0$ e $max_{new} = 1$.

Para as demais características, como se tratam de variáveis categóricas, não embutindo significação numérica em seus valores, iremos utilizar a codificação *onehot encoding*, a qual utiliza uma vetor esparso de dimensão d, sendo esta dimensão a quantidade de valores distintos que a variável pode assumir. Por exemplo, para a variável "modalidade", que pode ser "CONVENIO" ou "CONTRATO DE REPASSE", nossa representação seria um vetor de 2 dimensões, que assumiria os valores [1,0] para "CONVENIO" e [0,1] para "CONTRATO DE REPASSE".

Além desses dados, utilizar datas em sua forma original, apenas com a informação de ano, mês e dia de celebração e prestação de contas não permitiria o confronto direto entre os instrumentos. Desta forma, procedeu-se à transformação dessa informação em diferença de dias entre as datas inicial e final do instrumento, bem como a data entre a prestação de contas e o término do instrumento.

Desta forma, além dos dados anteriormente apresentados, foram acrescentados os seguintes:

- Percentual total pactuado, percentual de contrapartida, e percentual desembolsado;

- Se instrumento deriva de emenda;
- Diferença, em dias, entre início e fim da vigência, original e modificado, quando aplicável;
- Quantidade de órgãos proponentes, quando se tratarem de órgãos em consórcio;

4.3 Execução e Otimização de Hiperparâmetros do *XGBoost*

Este trabalho foi desenvolvido utilizando-se a linguagem *Python*, juntamente com o módulo *Scikit-Learn* (PEDREGOSA et al., 2011) e as bibliotecas *XGBoost*, que implementam o algoritmo de *tree boosting*, e *Hyperopt* (BERGSTRA et al., 2015), responsável pela otimização dos hiperparâmetros.

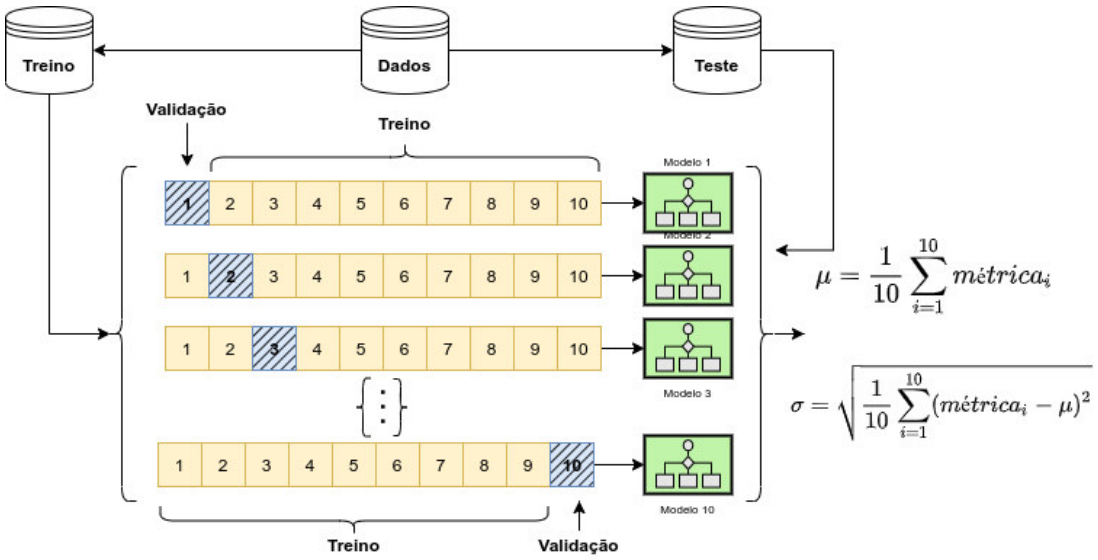
Para a realização dos experimentos, utilizou-se o método *k-fold cross-validation* de 10 segmentos (KOHAVI et al., 1995). Para isso, particionou-se igualmente os dados em 10 segmentos ou *folds*. Portanto, para cada execução, os dados de cada segmento ficaram assim distribuídos:

Tabela 2 – Proporção da distribuição dos dados de treino e teste *k-fold*, para k=10

Dados	**Positivos**	**Negativos**
Treino	2005	46.217
Teste	223	5.136

Nesses *folds*, treino e teste são feitos em 10 iterações, de modo que, em cada iteração, o treinamento do modelo é feito com 9 segmentos e o teste é feito no *fold* restante (Yadav; Shukla, 2016). As métricas de avaliação são obtidas em cada iteração, e ao final das iterações, são calculados média e desvio-padrão, para podermos avaliar o modelo. Um esquema deste processo é apresentado na figura 16, que apresenta a divisão dos dados em treino e teste, validação do modelo com os dados de validação e extração das métricas.

Figura 16 – Esquema *k-fold cross-validation*



Fonte: Autoral

Todo esse processo é executado para a obtenção de um modelo de classificação de riscos de transferências voluntárias. Após cada execução, são feitas buscas no espaço de buscas de hiperparâmetros para otimização do modelo. A otimização de hiperparâmetros foi realizada com *Hyperopt*, biblioteca baseada no modelo de regressão probabilística *tree-structured Parzen estimator* (BERGSTRA et al., 2011).

Para que o *Hyperopt* consiga otimizar e selecionar os melhores parâmetros para o modelo, foi necessária a configuração dos seguintes parâmetros:

- Função objetivo: a função que se pretende otimizar, no presente caso, o erro de validação dos modelos;
- Espaço de busca de hiperparâmetros: valores que podem ser assumidos pelos hiperparâmetros do modelo, descritos a seguir, e seus valores apresentados na tabela 3:
  - *eta*: Taxa de aprendizagem. Em problemas de *gradient boosting*, este parâmetro controla o peso dado a cada nova árvore adicionada ao modelo;
  - *max_depth* (MXD): Máxima profundidade das árvores para os *base learners*[3];
  - *min_child_weight* (MCW): Somatório mínimo dos pesos de um nó folha.
  - *subsample* (SS): Taxa de subamostragem de cada instância de treinamento;
  - *gamma*: Redução mínima na perda necessária para realizar novos particionamentos em nós folhas da árvore;

[3] Classificadores que, individualmente, não possuem boa performance.

- *colsample_bytree* (CST): Taxa de subamostragem das colunas ao construir novas árvores;
- *alpha*: Regularização L1 para os pesos;
- *lambda*: Regularização L2 para os pesos;
- *scale_pos_weight* (SPW): Balanceamento dos pesos positivos e negativos;

Tabela 3 – Espaço de busca de hiperparâmetros utilizados no *Hyperopt*.

Regressão Logística **Hiperparâmetro**	**Domínio** **Domínio**
eta	$0,025 + (n-1) * 0,025, n = [1..10]$
MXD	$1 + (n-1), n = [1..14]$
MCW	$1 + (n-1), n = [1..10]$
SS	$0,7 + (n-1) * 0,05, n = [1..7]$
gamma	$0,5 + (n-1) * 0,05, n = [1..11]$
CST	$0,7 + (n-1) * 0,05, n = [1..7]$
alpha	$1 + (n-1), n = [1..11]$
lambda	$1 + (n-1) * 0,1, n = [1..11]$
SPW	$50 + (n-1) * 10, n = [1..16]$

De igual modo, também foram configurados espaços de busca para os algoritmos de comparação, RL e MLP, conforme descritos na tabela 4.

Tabela 4 – Espaço de busca de hiperparâmetros - Regressão Logística e Multilayer Perceptron.

Regressão Logística		
Hiperparâmetros	**Descrição**	**Domínio**
C	Termo inversamente proporcional à regularização utilizada.	$[0, 1000]$
Scaling	Caso utilizada uma variável bias, qual o peso a ser atribuído a ela.	$[0, 1000]$
L1Ratio	Taxa de regularização L1.	$[0, 1]$
Multilayer Perceptron		
Hiperparâmetros	**Descrição**	**Domínio**
LearningRate	A taxa de aprendizagem em redes neurais é a taxa de variação dos pesos entre os nós da rede.	$[0\,001, 1]$
HiddenLayers	Quantidade de nós presentes na camada escondida.	$[10, 100]$
Alpha	Termo de regularização L2 para os pesos.	$[0, 1000]$
Activation	Função de ativação a ser utilizada.	[ReLu, Sigmoid, Tanh]

É importante observar que, como a base de dados possui alto grau de desbalanceamento, os dados devem ser estratificados antes de serem divididos em segmentos.

Estratificação é o processo de reorganizar os dados de maneira que cada *fold* seja uma amostra representativa do todo, de maneira que cada segmento contenha a mesma proporção dos casos positivos e negativos (DIAMANTIDIS; KARLIS; GIAKOUMAKIS, 2000).

Além do teste com a base altamente desbalanceada, utilizou-se os métodos de amostragem para o tratamento do balanceamento. Foram feitos testes com SMOTE, para a geração de novos exemplos sintéticos positivos, bem como método *NearMiss* de *undersampling*, para a redução no número de exemplos negativos. Também foram utilizadas combinações dos métodos para atacar o problema do desbalanceamento.

4.4 Validação do Modelo

Para a avaliação dos resultados de classificação dos modelos, as métricas utilizadas foram acurácia, precisão, sensibilidade, especificidade e *f-measure*. Buscamos por um método que tenha boa capacidade de classificar corretamente as contas reprovadas (sensibilidade) e, ao mesmo tempo, conseguir identificar as contas aprovadas (especificidade).

Para melhor visualização dos resultados da aprendizagem, a tabela 5 apresenta uma matriz de confusão. Cada linha da matriz representa uma ocorrência específica de uma classe prevista pelo modelo, enquanto que cada coluna representa uma ocorrência da verdadeira classe (THARWAT, 2018). Nesta tabela estão representadas as classes utilizadas para a classificação das transferências voluntárias quanto à sua prestação de contas.

Tabela 5 – Métricas importantes para avaliação da performance da metodologia.

		Verdadeiro	
		Reprovação	Aprovação
Predição	Reprovação	TP	FP
	Aprovação	FN	TN

Toda prestação de contas, após ser analisada, deverá receber um parecer pela sua aprovação ou reprovação. Desta forma, como se trata de um problema de classificação binário, as reprovações de contas foram consideradas os eventos de interesse, portanto, classificadas como positivos. Para os demais casos, aprovação de contas, foi atribuído o valor negativo. Dessa forma, as medidas TP, FN, FP e TN possuem a seguinte interpretação (THARWAT, 2018):

- Verdadeiros Positivos (TP - *True Positives*): referem-se aos casos em que o parecer foi pela reprovação das contas, classe positiva, e que o modelo classificou corretamente como positivo, ou seja, reprovação.

- Falsos Negativos (FN - *False Negatives*): referem-se aos casos em que o parecer foi pela reprovação, ou seja, classe positiva, porém o modelo a classificou de forma equivocada como sendo da classe negativa (aprovação);
- Falsos Positivos (FP - *False Positives*): referem-se aos casos em que o parecer foi pela aprovação das contas (classe negativa) porém previstos incorretamente como da classe positiva;
- Verdadeiros Negativos (TN - *True Negatives*): referem-se aos casos em que a classe é negativa e foram previstos corretamente como sendo desta classe, ou contas aprovadas.

Figura 17 – Representação Visual dos resultados de classificação

Fonte: Autoral

Analisando-se os valores da matriz de confusão, é possível obter métricas capazes de comparar os resultados obtidos pelos modelos, validando seus resultados. A acurácia (ACU) representa a proporção de todos os resultados verdadeiros (verdadeiros positivos e verdadeiros negativos) em relação à população utilizada. Quanto maior for o valor da acurácia, mais preciso será o teste (THARWAT, 2018). A forma de realizar seu cálculo é apresentada na Equação 4.2:

$$ACU = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.2}$$

Para o presente caso, considerando o desbalanceamento da base, temos uma relação positivos/negativos da ordem de 5%/95%. Um classificador que atribuísse a classe negativa a todos os casos, ainda assim, obteria uma acurácia de 95%. Portanto, principalmente para classificadores binários, a acurácia por si só não é um bom indicador de performance, em especial para dados altamente desbalanceados, como no presente caso (STONE, 2014). Desta forma, métricas de performance que sofrem menos este tipo de problema serão utilizadas (BATISTA; PRATI; MONARD, 2004).

A sensibilidade (SEN), também chamada de *true positive rate* (TPR), ou taxa de verdadeiros positivos, é a taxa de acertos reais considerando todos os casos positivos. Representa a probabilidade de um resultado positivo identificar, de fato, prestações de contas que foram reprovadas. Quanto maior o valor numérico da sensibilidade, menos

provável a classificação da conta gerar um falso positivos (FP) (THARWAT, 2018). O cálculo da SEN é apresentado na Equação 4.3:

$$SEN = \frac{TP}{TP + FN} \tag{4.3}$$

A especificidade (ESP), *true negative rate* (TNR), ou taxa de verdadeiros negativos, representa a probabilidade de uma predição classificar pela aprovação sem a presença de resultados falso positivos (FP). É a proporção dos verdadeiros negativos (TN) corretamente identificados pelo classificador, ou seja, quanto maior o valor da especificidade, mais provável que as contas, de fato, sejam aprovadas (THARWAT, 2018). Seu cálculo é baseado na Equação 4.4:

$$ESP = \frac{TN}{TN + FP} \tag{4.4}$$

Por seu turno, a precisão (PRE) é a proporção de casos positivos preditos (contas reprovadas) que são positivos reais, ou seja, quanto maior a precisão mais acertos dos casos em que houve reprovação das contas (THARWAT, 2018). A precisão é definida na Equação 4.5:

$$PRE = \frac{TP}{TP + FP} \tag{4.5}$$

O *f-measure* (F) é a média ponderada de precisão e sensibilidade. Leva em conta tanto os falsos positivos (FP) quanto os falsos negativos (FN). Quando há um desbalanceamento entre classes, como é o caso dos dados utilizados no presente trabalho, quanto maior o valor de *f-measure*, mais precisa é a classificação. Seu cálculo se baseia na equação 4.6 (THARWAT, 2018):

$$\textit{f-measure} = \frac{2 \cdot PRE \cdot SEN}{PRE + SEN} \tag{4.6}$$

Este capítulo apresentou os principais detalhes quanto à metodologia proposta, expondo todas as etapas do fluxo e seus materiais principais, como a configuração do *hardware* e os *softwares* usados, além da base de dados utilizada, sua extração e preparação, bem como os métodos utilizados e a forma de validação da metodologia. A seguir, serão apresentados e discutidos os resultados obtidos com a aplicação da metodologia proposta.

5 Resultados

Este capítulo tem como objetivo apresentar os resultados alcançados pela metodologia proposta, a qual investiga a eficiência da utilização do *XGBoost* na classificação de transferências voluntárias federais em perfis de risco de reprovação em comparação com outras técnicas de aprendizagem computacional - *Logistic Regression* (LR) e uma rede neural *Multilayer Perceptron* (MLP). Conforme descrito no Capítulo 4, para a realização dos experimentos, utilizou-se o método *k-fold* de *cross-validation*, com $k = 10$, sendo 9 *folds* para treinamento e 1 para validação. Ao final de cada bloco, extraiu-se a média e desvio-padrão das métricas escolhidas a fim de analisar a generalização do modelo em análise. Após isso, o modelo foi utilizado para classificação dos dados de teste, obtendo-se as métricas para sua avaliação. Para validar a metodologia, considerou-se as métricas de acurácia, sensibilidade, especificidade, precisão, *f-measure* e área sob a curva ROC.

Segundo já descrito na metodologia, o *dataset* utilizado é desbalanceado na proporção de 24 para 1. Os testes iniciais foram realizados utilizando-se os dados nestas mesmas proporções. Na segunda bateria de avaliações, foram feitos testes com estratégias de rebalanceamento dos dados de treinamento, com SMOTE e NearMiss. Porém, frise-se que os testes dos modelos foram realizados mantendo-se, nos dados de teste, a proporção original de desbalanceamento entre exemplos positivos e negativos. No decorrer dos experimentos, também foram avaliados os impactos que determinadas variáveis trazem às previsões dos modelos. Assim, testes foram feitos com um subconjunto de determinadas variáveis.

Quanto à otimização de hiperparâmetros, esta foi feita em todos os experimentos, de forma a avaliar os impactos tanto do balanceamento dos dados, quanto das características utilizadas pelos modelos. Inicialmente, são apresentados os dados com os hiperparâmetros padrão de cada modelo e, em seguida, o melhor resultado obtido após a sua otimização.

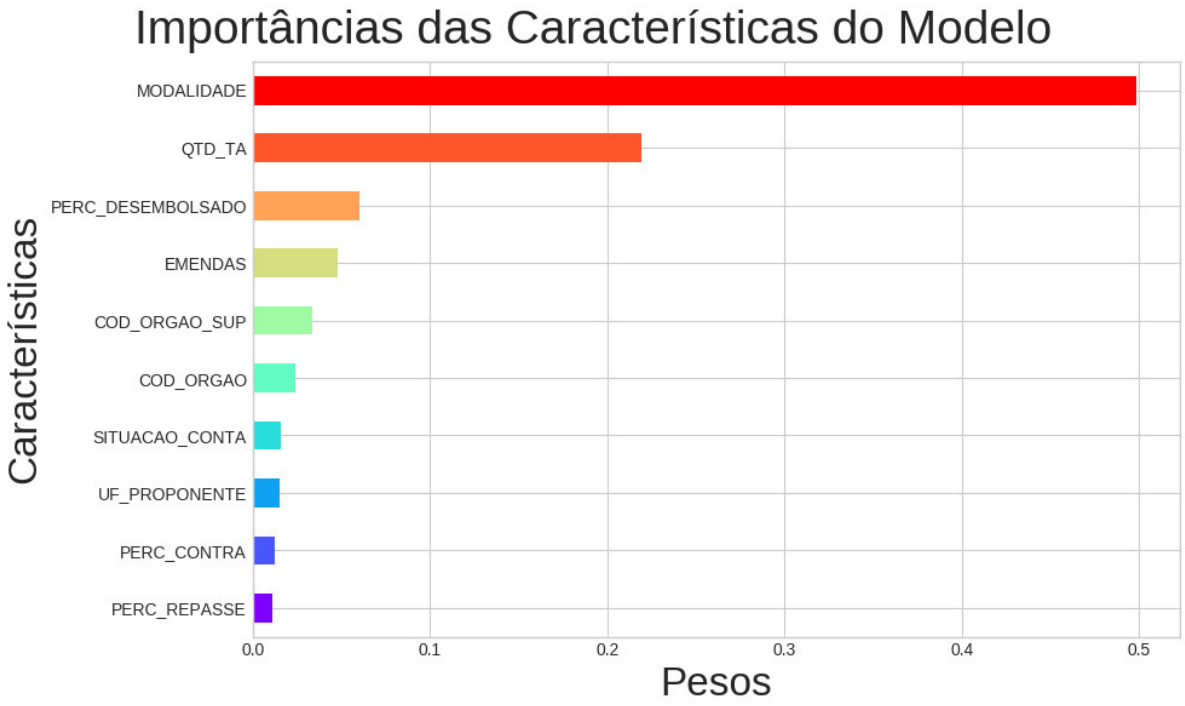
É importante relembrar que os exemplos positivos são considerados aqueles em que a manifestação do controle interno é pela reprovação das contas. Analisando-se as métricas obtidas nos testes com dados desbalanceados e hiperparâmetros padrão, verificam-se os resultados conforme apresentados na tabela 6, entre os quais o *XGBoost* obteve uma taxa de sensibilidade de 0,878.

Tabela 6 – Resumo das métricas para dados desbalanceados sem otimização de hiperparâmetros

	ACU	SEN	ESP	PRE	F-measure	AUC
XGBClassifier	0.952	**0.878**	0.956	0.510	**0.645**	**0.979**
LogisticRegression	0.919	0.710	0.930	0.345	0.464	0.937
MLPClassifier	**0.961**	0.485	**0.986**	**0.641**	0.552	0.936

Para esses resultados, as principais características obtidas pelo *XGBoost*, e seus respectivos pesos, são apresentados na figura 18:

Figura 18 – Pesos das principais características utilizadas pelo *XGBoost*, sem otimização.



Fonte: Autoral

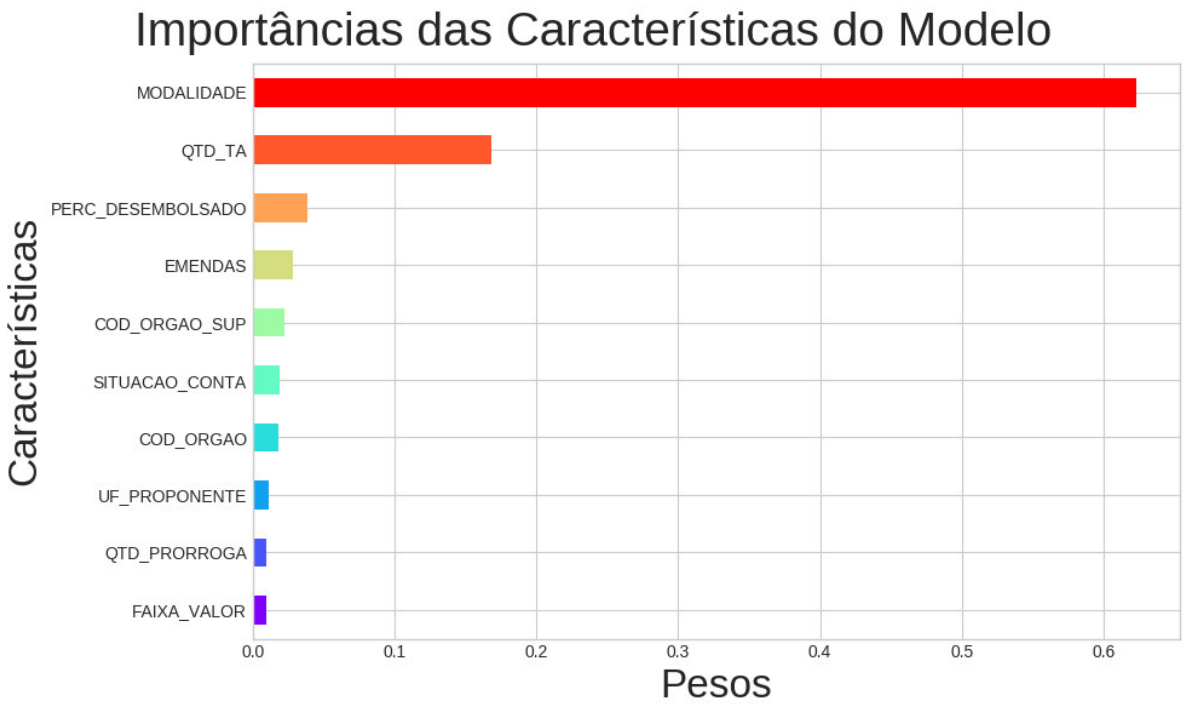
Ao procedermos à otimização dos hiperparâmetros dos três algoritmos de aprendizagem, obtemos significativa melhora nos resultados, que são apresentados na tabela 7.

Tabela 7 – Resumo das métricas para dados desbalanceados após otimização de hiperparâmetros

	ACU	SEN	ESP	PRE	F-measure	AUC
XGBClassifier	0.965	0.805	0.973	0.606	**0.692**	**0.981**
LogisticRegression	0.891	**0.924**	0.889	0.302	0.455	0.961
MLPClassifier	**0.967**	0.523	**0.990**	**0.721**	0.606	0.973

Novamente, o *XGBoost* obteve bons resultados, principalmente na métrica de f-measure, que captura a relação entre precisão e sensibilidade. Conforme apresentado na figura 19, as características mais importantes para o modelo não se alteraram, permanecendo similares ao resultado anterior, com modificações apenas em seus pesos. Houve ligeira redistribuição dos pesos, e as características de quantidade de prorrogações e faixa de valor ganharam relevância.

Figura 19 – Pesos das principais características utilizadas pelo *XGBoost*, otimizado.



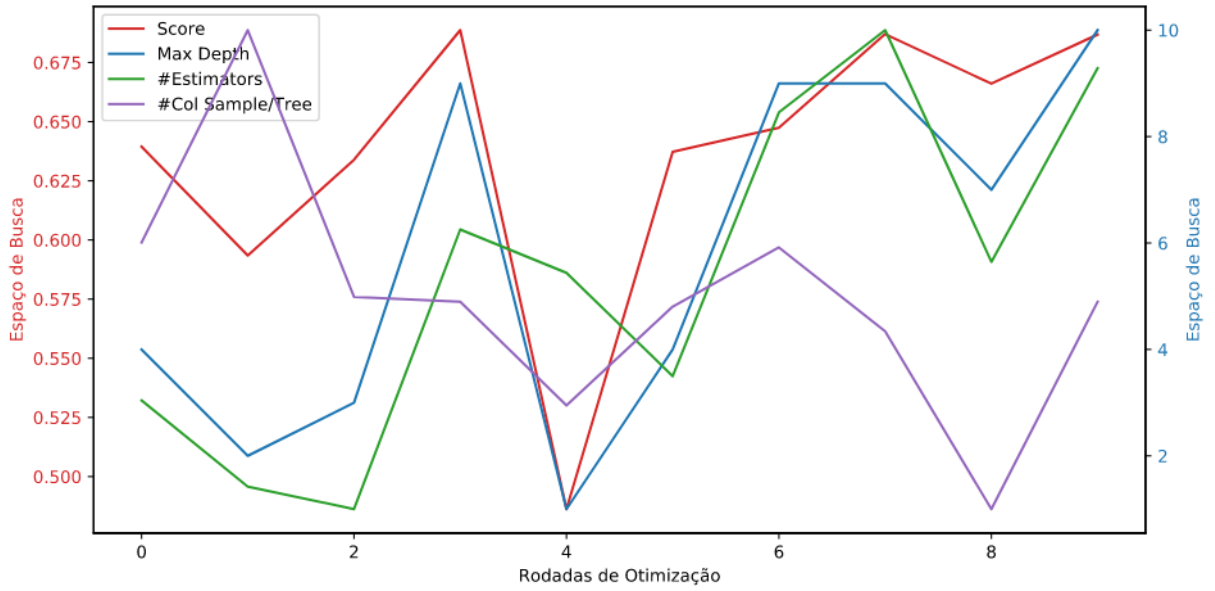
Fonte: Autoral

Verifica-se que entre as características mais influentes estão a modalidade de repasse e as que identificam o órgão repassador ou seu órgão vinculante. Estas características possuem grande relevância para o modelo, que consegue um ganho de informação pela diminuição da entropia do conjunto, conforme apresentado na figura 15. Juntamente com o código do município, estas características tem granularidade relativamente baixa, o que pode levar a um sobreajuste, ou *overfit*, do modelo. Ao removermos estas características, o resultado obtido pelo *XGBoost* alcança uma acurácia de 0,955, sensibilidade de 0,733, especificidade de 0,967, precisão de 0,533, *f-measure* de 0,617 e AUC de 0,968.

Apesar de não apresentar a melhor leitura, essa pode ser considerada uma interessante medida de generalização, tendo em vista que estas características identificam com certa precisão um Município específico ou um órgão repassador. Um ponto a ser observado é que apesar de termos cerca de 53 mil observações em nossos dados, apenas 2 mil são de casos positivos. Desses, aparecem 1.269 dos 5.570 municípios brasileiros. Brasília, Rio de Janeiro e Recife concentram uma grande parte dos casos positivos, somando 293 casos. Ainda, 374 municípios tem mais de uma reprovação, sendo 227 apenas 2. Mais da metade dos municípios brasileiros não tem estatística de reprovação de contas, o que indica que faz mais sentido manter o modelo sem esta informação para a classificação. Essa modificação no modelo deve ser avaliada à medida em que surgirem novos municípios com contas reprovadas, o que fará com que possamos testar o modelo já treinado com informações de um município que ainda não participou do treinamento.

Durante o processo de otimização do modelo, foram analisados os impactos que esta traz. Entre os hiperparâmetros, o que mais influencia o modelo é a profundidade máxima da árvore (MXD). Na figura 20, percebe-se claramente a característica da baixa dimensionalidade efetiva (BERGSTRA; BENGIO, 2012), apesar da quantidade de hiperparâmetros a serem otimizados. Verifica-se que alguns hiperparâmetros tem alta taxa de variação, porém sem impactar significativamente na performance do modelo.

Figura 20 – Otimzação de Hiperparâmetros do modelo.

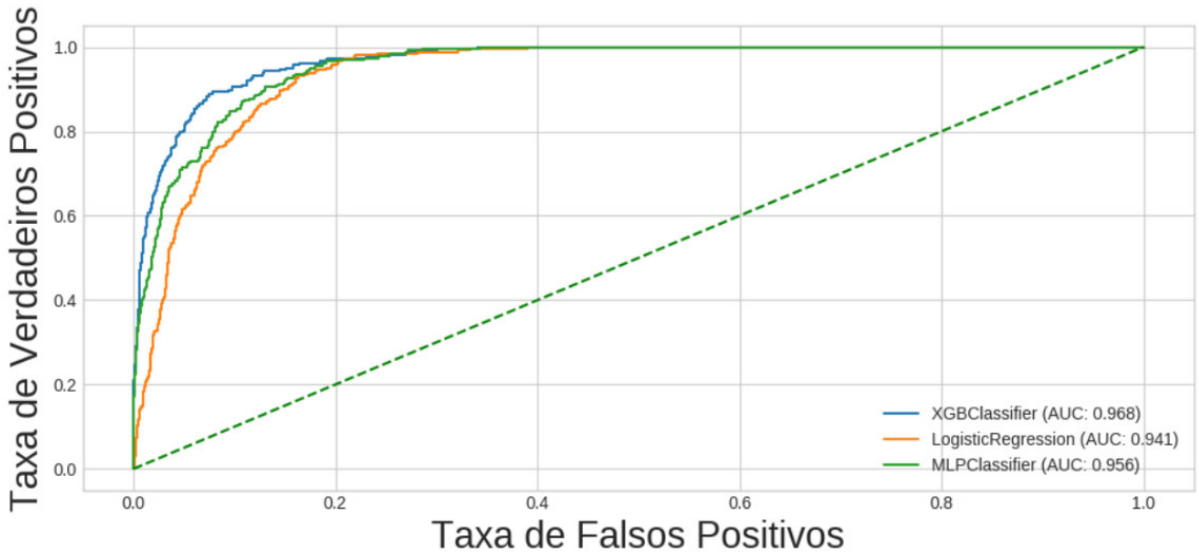


Fonte: Autoral

Uma outra maneira de avaliarmos a performance de modelos preditivos é utilizando a Curva ROC (Receiver Operating Characteristic) (FAWCETT, 2006). Por meio desta, pode-se avaliar a capacidade do modelo de separar os casos positivos e negativos, sendo esta análise independente de desbalanceamento (LASKO et al., 2005). Seu uso é especialmente importante em classificadores binários, quando este atribui uma probabilidade de um exemplo ser de uma classe ou de outra. Variando o limiar de decisão entre as classes, pode-se avaliar o que os autores chamam de *trade-off* entre sensibilidade e especificidade.

Para se realizar comparações objetivas entre classificadores, podemos utilizar a área sob a curva ROC, do inglês *Area Under Curve* (AUC). A AUC pode ser interpretada como sendo a probabilidade de que um classificador irá atribuir a um exemplo positivo aleatório um *score* superior a um exemplo negativo aleatório (FAWCETT, 2006). A figura 21 compara a curva ROC dos 3 classificadores com parâmetros otimizados. Percebe-se que o *XGBoost* tem a melhor relação entre Taxa de Verdadeiros Positivos e seu *trade-off*, entre Taxa de Falsos Positivos.

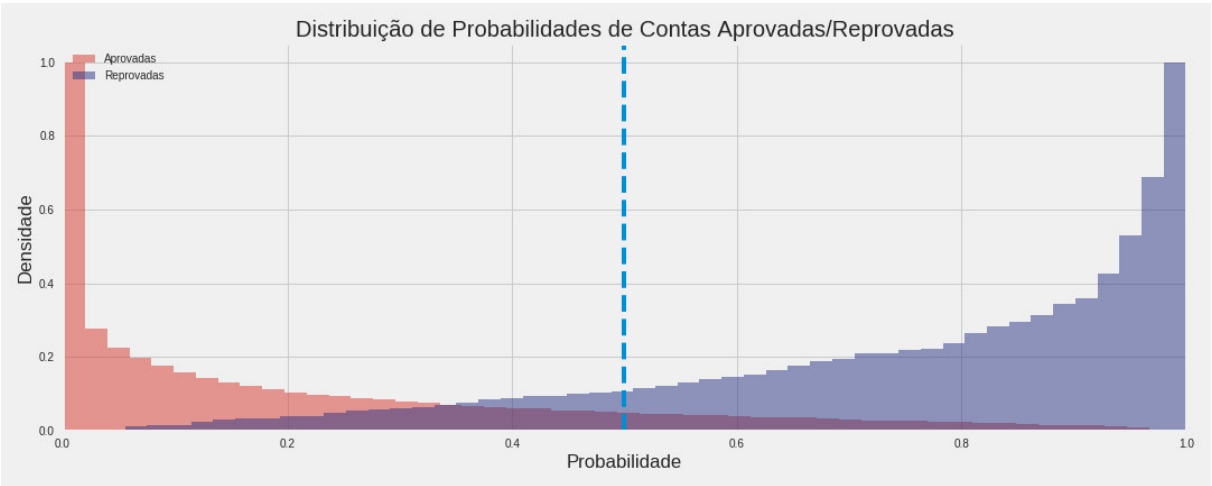
Figura 21 – *Area Under Curve* dos classificadores para dados desbalanceados.



Fonte: Autoral

Também podemos analisar o nosso classificador ao verificarmos a distribuição de frequência acumulada dos casos positivos e negativos, conforme figura 22. O limiar de decisão de 0,5 está marcado na figura. Os casos da distribuição de probabilidade dos positivos que estão abaixo de 0,5 são falsos negativos. Já os casos da distribuição de negativos acima de 0,5 são falsos positivos. Visualmente, podemos verificar o que ocorreria com os casos de falsos positivos e falsos negativos caso mudássemos o *threshold* de classificação.

Figura 22 – Densidade Cumulativa de Previsões Aprovados e Reprovados



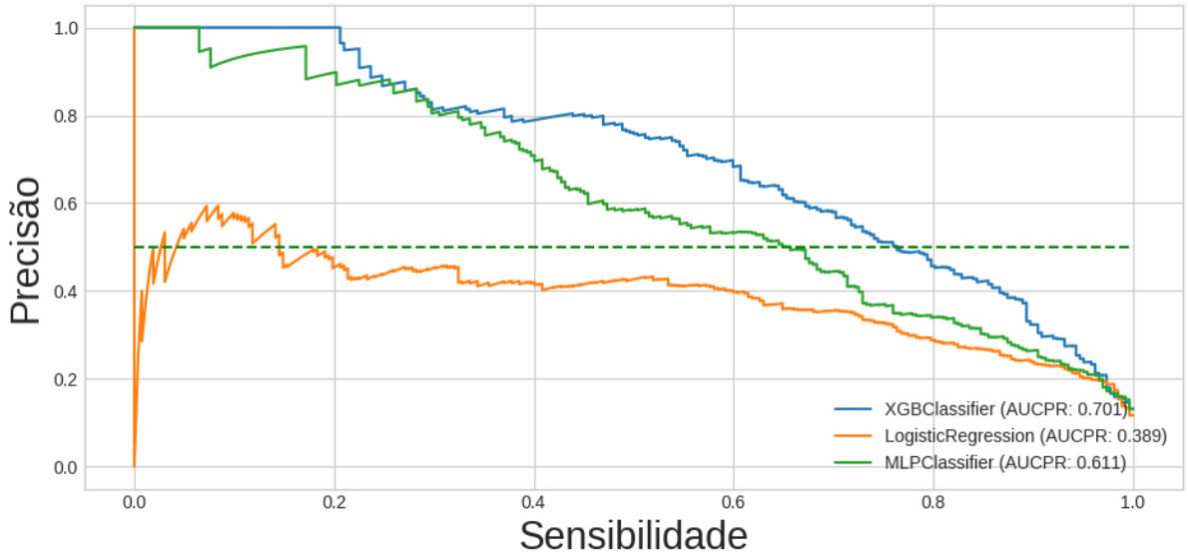
Fonte: Autoral

Há, ainda, outra questão a ser abordada. Observe-se que o objetivo da utilização de

um classificador de riscos de transferências voluntárias é assegurar maior tempestividade na análise de contas possivelmente problemáticas, o que aumenta as chances de os recursos financeiros serem recuperados. Desta forma, o aumento na quantidade de contas analisadas pelo controle interno, mesmo que não resultem em reprovação das contas, portanto falsos positivos, evita que contas flagrantemente reprováveis passem no crivo de contenção de desperdício de recursos públicos. Portanto, considerando a figura 22, caso o *threshold* fosse diminuído para 0,4, teríamos uma quantidade maior de casos de falsos positivos, porém, também teríamos um aumento nos casos de verdadeiros positivos, o que poderia atender aos objetivos propostos.

O *trade-off* entre aumentar a taxa de falsos positivos, diminuindo o limiar de classificação, em contraste com uma menor precisão, dado que a precisão é a taxa entre verdadeiros positivos sobre todos os previstos como positivos, fará com que se aumente a taxa de sensibilidade, uma vez que esta é a razão entre verdadeiros positivos e todos os positivos de fato. Nesse sentido, de acordo com Saito e Rehmsmeier (2015), pode ser mais interessante a análise da curva *Precision x Recall* (Precisão x Sensibilidade), especialmente ao avaliarmos classificadores binários com alto desbalanceamento entre classes. Esta curva é apresentada na figura 23 para os três classificadores otimizados.

Figura 23 – Curva *Precisão x Sensibilidade* dos classificadores para dados desbalanceados.



Fonte: Autoral

Percebe-se nesta figura que ao aumentarmos a taxa de verdadeiros positivos (sensibilidade), acabamos por diminuir a precisão. Entende-se que adotar um limiar de decisão que seja adequado passa por uma análise de apetite ao risco, que deve ser empreendida pelo órgão que irá utilizar essas medidas. Conforme se verifica no portal da plataforma

Mais Brasil [1], a análise informatizada deve passar por um processo de avaliação de limites de tolerância a riscos, com publicação da respectiva norma. Desta forma, cada órgão definirá quantas contas a mais ou a menos estaria disposto a analisar na tentativa de ser mais tempestivo na análise de contas a serem possivelmente reprovadas.

Em outra esteira, experimentos também foram realizados quanto à performance dos modelos com dados balanceados. Foram adotados três métodos. Optou-se por utilizar o rebalanceamento em 3 novas proporções, a saber, 10x1, 5x1 e 1x1. Utilizou-se os métodos SMOTE para geração aleatória de novos exemplos positivos, NearMiss para a eliminação de exemplos negativos e, por fim, utilizou-se uma combinação desses métodos de *undersampling* e *oversampling*. Os dados utilizados excluíram as informações de municípios e órgãos vinculados. A tabela a seguir sintetiza os resultados obtidos pelos métodos.

Tabela 8 – Comparativo dos resultados dos modelos com diferentes métodos de *resampling*

Taxa	Método	Modelo	ACU	SEN	ESP	PRE	F-M	AUC
10 x 1	**SMOTE**	XGBoost	0.967	0.775	0.977	0.636	0.699	0.977
		LR	0.894	0.920	0.893	0.307	0.461	0.961
		MLP	0.966	0.584	0.986	0.677	0.627	0.970
	NearMiss	XGBoost	0.932	0.782	0.940	0.404	0.533	0.963
		LR	0.930	0.855	0.934	0.401	0.546	0.964
		MLP	0.960	0.565	0.981	0.604	0.584	0.954
	SMO+NM	XGBoost	0.967	0.779	0.977	0.639	**0.702**	0.978
		LR	0.901	**0.924**	0.900	0.324	0.480	0.963
		MLP	0.967	0.553	**0.988**	0.704	0.620	0.970
5 x 1	**SMOTE**	XGBoost	0.967	0.737	0.979	0.641	0.686	0.978
		LR	0.896	0.920	0.895	0.312	0.466	0.961
		MLP	0.965	0.592	0.984	0.662	0.625	0.968
	NearMiss	XGBoost	0.855	0.805	0.857	0.226	0.353	0.918
		LR	0.874	0.824	0.877	0.258	0.393	0.927
		MLP	0.923	0.626	0.939	0.347	0.446	0.915
	SMO+NM	XGBoost	0.968	0.744	0.980	0.659	0.699	0.978
		LR	0.905	0.920	0.904	0.333	0.489	0.963
		MLP	0.964	0.565	0.985	0.658	0.608	0.968
1 x 1	**SMOTE**	XGBoost	**0.970**	0.626	**0.988**	**0.726**	0.672	**0.979**
		LR	0.902	0.920	0.901	0.325	0.480	0.962
		MLP	0.965	0.565	0.986	0.673	0.614	0.968
	NearMiss	XGBoost	0.690	0.893	0.680	0.126	0.221	0.859
		LR	0.256	0.798	0.228	0.051	0.096	0.520
		MLP	0.237	0.737	0.211	0.046	0.087	0.480
	SMO+NM	XGBoost	**0.970**	0.626	**0.988**	**0.726**	0.672	**0.979**
		LR	0.902	0.920	0.901	0.325	0.481	0.962
		MLP	0.966	0.546	0.987	0.691	0.610	0.967

No presente caso, observa-se que os melhores resultados foram obtidos com o reba-

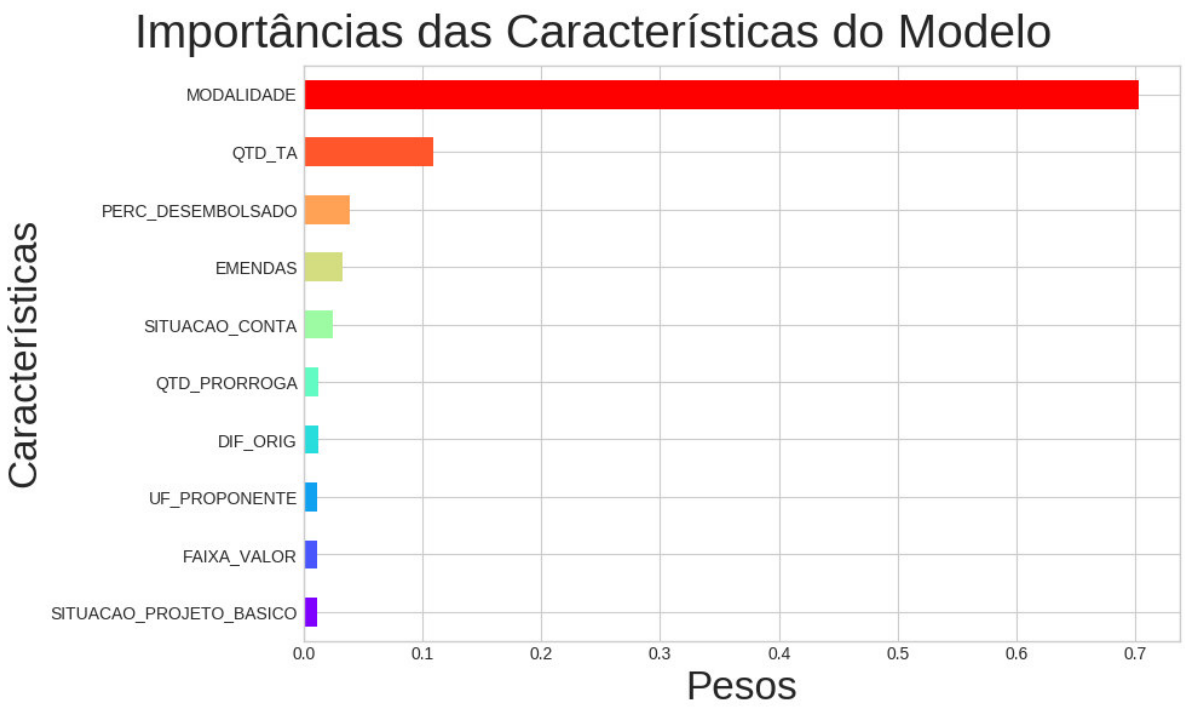
[1] <http://plataformamaisbrasil.gov.br/analise-informatizada>

lanceamento utilizando-se a combinação dos métodos SMOTE e NearMiss, na proporção de 1x1. Uma curiosidade que podemos observar é a redução da precisão dos modelos à medida em que fomos reduzindo a quantidade de casos negativos (NearMiss) no rebalanceamento. Na prática, num cenário de alta dimensão de características, como observado por Lusa et al. (2013), classificadores *k-NN* baseados em distância euclidiana parecem se beneficiar mais do uso do SMOTE, com benefícios proporcionais à utilização de mais vizinhos. No presente caso, se consideramos que há necessidade de o órgão repassador definir o limiar de risco que está disposto a aceitar para analisar as contas, o indicador que melhor captura essa variação é a AUC. O resultado obtido pelo melhor modelo, 0,979, não superou o melhor resultado obtido pelo melhor modelo com dados desbalanceados e hiperparâmetros otimizados, que alcançou o valor de 0,980.

Um dos principais problemas de classificação quando se tratam de dados altamente desbalanceados, como na situação em tela, é que a acurácia não traduz fielmente a qualidade do classificador. No presente caso, as observações positivas representam cerca de 5% da base completa. Um modelo que classificasse qualquer exemplo como negativo, ainda assim teria uma taxa de acurácia de cerca de 95%. Nos testes com a base desbalanceada original o melhor modelo superou ligeiramente esse resultado, atingindo 96,1% de taxa de acurácia. O melhor modelo para a base balanceada, proporção 1x1, atingiu acurácia de 97%, uma ligeira vantagem que não foi alcançada em outras métricas que capturam melhor as características do desbalanceamento da base.

Para o melhor caso balanceado, alcançado utilizando-se o método SMOTE+NearMiss 1x1, as principais características utilizadas pelo *XGBoost*, e seus respectivos pesos, são apresentados na figura 24.

Figura 24 – Pesos das principais características utilizadas pelo *XGBoost*.



Fonte: Autoral

Percebe-se uma homogeneidade das principais características nas avaliações dos modelos, independentemente de ser com dados balanceados ou desbalanceados. Porém, verifica-se uma melhor distribuição nos pesos das características, quando para os dados desbalanceados, havia maior prevalência para a quantidade de termos aditivos e percentual desembolsado, sugerindo que, além da modalidade de repasse, para um maior número de prestações de contas, os valores desembolsados separam melhor os casos.

Uma vantagem da utilização de árvores de decisão é a transparência dos modelos gerados, com regras claras que podem ser validadas por especialistas, o que não acontece com as redes neurais, por exemplo. Tendo em vista que se trata de um modelo de *gradient boosting*, que gera um conjunto de várias árvores de decisão, não poderíamos reproduzir, neste espaço, todas as regras do modelo. O melhor modelo para dados balanceados, por exemplo, sem as características que identificam o município e o órgão, após otimização, gerou um modelo com 169 *boosters* e com uma profundidade de árvore de 13 níveis.

Porém, pode-se verificar os resultados obtidos pela aplicação das regras, bem como as características que contribuíram para estes resultados. Analisemos dois resultados falsos de nossa base de teste, um falso positivo e outro falso negativo. Conforme tabela 9, é possível observar como o modelo se comportou. Um fator determinante para o falso positivo foi a presença das características "Situação da Conta"como "Registrada", que, conforme se verifica na figura 25, é o valor cuja concentração de casos positivos é maior. Quanto ao caso

de falso negativo descrito na tabela 10, a situação do projeto básico teve fator determinante para que o modelo atribuísse classificação como negativa, tendo em vista a concentração de casos negativos para a situação "aguardando", apenas 180 em em total de 3493, cerca de 5%. Estas regras podem ser facilmente analisadas e validadas por especialistas da área de negócios, que podem, inclusive, propor melhorias na forma de abordar o tema.

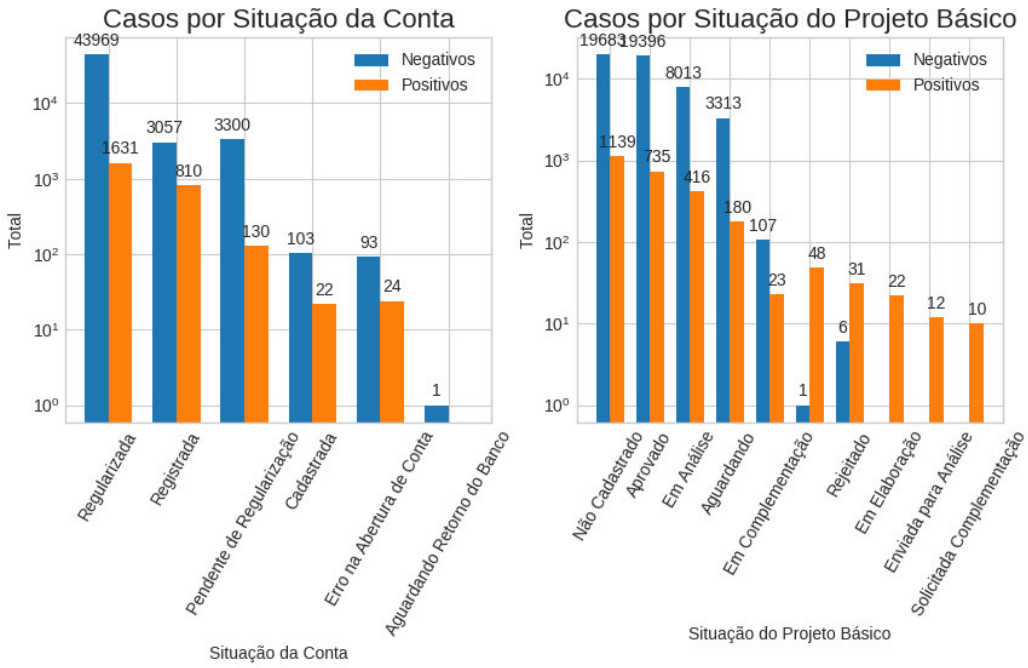
Tabela 9 – Caso de Falso Positivo - Contribuição das Variáveis

Falso Positivo	**(p = 0.847)**	
Contribuição	**Característica**	**Valor**
+1.215	SITUACAO_CONTA	Aguardando
+1.162	MODALIDADE	Convênio
+0.633	<BIAS>	1.000
+0.376	FAIXA_VALOR	Tomada de Contas
+0.158	PERC_REPASSE	0.949
+0.127	EMENDAS	0.000
+0.093	PERC_CONTRA	0.051
-0.005	DIF_PRORROG	680.000
-0.033	UF_PROPONENTE	TO
-0.083	MES	DEZ
-0.127	DIF_ORIG	365.000
-0.161	SITUACAO_PROJETO_BASICO	Não Cadastrado
-0.261	QTD_PRORROGA	1.000
-0.444	PERC_DESEMBOLSADO	0.854
-0.936	QTD_TA	1.000

Tabela 10 – Caso de Falso Negativo - Contribuição das Variáveis

Falso Negativo	**(p = 0.810)**	
Contribuição	**Característica**	**Valor**
+1.006	QTD_TA	1.000
+0.880	PERC_DESEMBOLSADO	0.980
+0.402	SITUACAO_PROJETO_BASICO	Aguardando
+0.374	SITUACAO_CONTA	Regularizada
+0.363	PERC_REPASSE	0.980
+0.194	DIF_PRORROG	605.000
+0.143	MES	DEZ
+0.112	QTD_PRORROGA	1.000
+0.051	PERC_CONTRA	0.020
+0.047	DIF_ORIG	367.000
+0.020	UF_PROPONENTE	RJ
-0.037	EMENDAS	0.000
-0.328	FAIXA_VALOR	Tomada de Preço
-0.633	<BIAS>	1.000
-1.147	MODALIDADE	Convênio

Figura 25 – Análise de variáveis que impactaram nas previsões.



Fonte: Autoral

Há carência de publicações de estudos que comparem resultados de vários algoritmos de aprendizagem de máquina no domínio proposto, em especial de bases de dados de transferências voluntárias do governo. Este capítulo apresentou os principais resultados obtidos na aplicação da metodologia proposta. Foram discutidos os resultados obtidos com dados desbalanceados, métodos utilizados para rebalanceamento, e os resultados obtidos. Foram feitas comparações entre classificadores que utilização Regressão Logística, Redes Neurais MLP e Árvores de Decisão.

6 Conclusão

As transferências voluntárias da União a estados e municípios somam, desde a implantação do sistema de gestão de convênios, Siconv, cerca de R$ 133 bilhões de reais. Esses recursos são transferidos aos entes federados para fazer frente às novas despesas criadas com o processo de descentralização administrativa promovido pelo novo pacto federativo de 1988, que lhes atribuiu um conjunto de serviços à comunidade que, até então, era de responsabilidade das esferas superiores.

Entre os instrumentos utilizados para as transferências voluntárias, que devem observar critérios específicos de cada órgão repassador, estão os convênios e os contratos de repasse, que deram origem aos dados, objeto do presente estudo. Com a transferência de recursos, é criada também a responsabilidade, para o ente recebedor, de prestação de contas do montante recebido. Após apresentação das contas, o órgão repassador emite um parecer sobre as contas, aprovando-a ou reprovando-a. Em caso de reprovação, os recursos transferidos devem ser devolvidos para que possam ser utilizados novamente para o benefício da sociedade.

Técnicas de aprendizagem computacional, como as que foram propostas neste trabalho, em especial algoritmos de *gradient boosting* como o *XGBoost*, podem ser utilizadas no auxílio aos órgãos de controle para a seleção das contas a serem analisadas para que se tenha uma maior tempestividade e possivelmente maior eficácia na recuperação de recursos que tenham, eventualmente, sido gastos de forma inadequada.

A presente dissertação apresentou uma metodologia de classificação de transferências voluntárias em perfis de riscos utilizando o *XGBoost*, comparando seus resultados com outras técnicas geralmente utilizadas em problemas de *credit scoring*, como Regressão Logística e Multilayer Perceptron. Os dados utilizados, disponíveis publicamente no sistema Siconv, possuem alto desbalanceamento entre número de contas aprovadas e reprovadas. Foram utilizadas técnicas de rebalanceamento de dados, como SMOTE e NearMiss, para verificar o impacto do desbalanceamento do treinamento dos modelos.

Os resultados obtidos comprovam a efetividade da metodologia proposta, com boa vantagem para a utilização do *XGBoost* sem grandes impactos, mesmo com a utilização de dados desbalanceados. Após a otimização dos hiperparâmetros do modelo, os resultados alcançados foram de acurácia de 96,5%, sensibilidade de 80,5%, especificidade de 97,3%, precisão de 60,6%, *f-measure* de 69,2% e AUC de 98,1%.

Os resultados apresentados para a classificação de riscos de transferências voluntárias mostraram-se bastantes promissores pelo fato de apresentarem valores elevados e consistentes. No entanto, entende-se que é uma decisão do órgão repassador adotar

um limiar de decisão que seja adequado ao seu apetite a risco, o que pode ocasionar a ocorrência de mais ou menos falsos positivos.

Como a metodologia proposta foi baseada no modelo de referência CRISP-DM, ainda resta a etapa de implantação do modelo. Esta etapa busca, efetivamente, implantar o modelo que apresentou os melhores resultados em ambiente de produção. Este trabalho, apesar de sua aplicabilidade prática, não colocou em produção a ferramenta de classificação de transferências voluntárias. Não obstante, a confecção de uma ferramenta mais robusta está em análise para a sua utilização por parte dos principais órgãos de controle e da sociedade em geral, para que se obtenha um maior controle efetivo dos gastos públicos.

6.1 Trabalhos Futuros

Embora os resultados apresentados para a classificação de riscos em transferências voluntárias tenham se mostrado satisfatórios, se vislumbram algumas oportunidades de melhoria visando a aumentar sua efetividade. Algumas melhorias são listadas abaixo:

- Executar a metodologia proposta em *datasets* de outras transferências voluntárias, a exemplo das transferências dos programas efetuadas pelo Fundo Nacional de Desenvolvimento da Educação (FNDE);
- Avaliar a generalização do modelo sem a utilização da característica de municípios, à medida em que surgirem novos municípios com contas reprovadas;
- Aumentar qualitativamente as características utilizadas pelo modelo, como a inclusão de dados de indicadores sociais, como IDH, e indicadores de Governança dos órgãos;
- Utilizar técnicas de seleção de características a fim de verificar o impacto de *features* que possuem altas taxas de correlação;
- Experimentar técnicas de Processamento de Linguagem Natural, do inglês *Natural Language Processing* (NLP) nas informações relativas tanto aos objetos pactuados nos instrumentos, quanto nas constantes da própria prestação de contas apresentada;
- Analisar outras técnicas de aprendizagem computacional como o *LightGBM* e SVM, bem como outras técnicas de *resampling* dos dados como Self-Organizing Map Oversampling (SOMO) e *Adaptive Synthetic* (ADASYN).

6.2 Produções Científicas

A Tabela 11 lista os artigos aceitos, produtos da metodologia proposta, no ano de 2019 e 2020 na área da ciência da computação. Os trabalhos apresentados a seguir foram produzidos como autor principal.

Tabela 11 – Artigos produzidos referentes ao tema de classificação de riscos em transferências voluntárias.

Título	**Congresso**	**Qualis**
Classificação de Risco em Transferências Voluntárias Federais Utilizando XGBoost	XIV CBIC	B5
Técnicas de Aprendizagem Computacional para Classificação de Riscos em Transferências Voluntárias Federais (a ser submetido)	Inderscience Electronic Government Journal	B1

Referências

AKKOÇ, S. An empirical comparison of conventional techniques, neural networks and the three stage hybrid adaptive neuro fuzzy inference system (anfis) model for credit scoring analysis: The case of turkish credit card data. *European Journal of Operational Research*, v. 222, n. 1, p. 168 – 178, 2012. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0377221712002858>. Citado na página 23.

ALARY, D.; GOLLIER, C. Debt Contract, Strategic Default, and Optimal Penalties with Judgement Errors. *Annals of Economics and Finance*, v. 5, n. 2, p. 357–372, November 2004. Disponível em: <https://ideas.repec.org/a/cuf/journl/y2004v5i2p357-372.html>. Citado na página 23.

BAESENS, B.; GESTEL, T. V.; VIAENE, S.; STEPANOVA, M.; SUYKENS, J.; VANTHIENEN, J. Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the Operational Research Society*, Taylor & Francis, v. 54, n. 6, p. 627–635, 2003. Citado na página 23.

BARROS, R. C.; BASGALUPP, M. P.; CARVALHO, A. C. D.; FREITAS, A. A. A survey of evolutionary algorithms for decision-tree induction. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, IEEE, v. 42, n. 3, p. 291–312, 2011. Disponível em: <https://ieeexplore.ieee.org/abstract/document/5928432/>. Citado na página 27.

BATISTA, G. E. A. P. A.; PRATI, R. C.; MONARD, M. C. A study of the behavior of several methods for balancing machine learning training data. *SIGKDD Explor. Newsl.*, Association for Computing Machinery, New York, NY, USA, v. 6, n. 1, p. 20–29, jun. 2004. Disponível em: <https://doi.org/10.1145/1007730.1007735>. Citado na página 59.

BERGSTRA, J.; BENGIO, Y. Random search for hyper-parameter optimization. *The Journal of Machine Learning Research*, v. 13, p. 281–305, 2012. Citado 5 vezes nas páginas 26, 45, 46, 47 e 64.

BERGSTRA, J.; KOMER, B.; ELIASMITH, C.; YAMINS, D.; COX, D. D. Hyperopt: a python library for model selection and hyperparameter optimization. *Computational Science & Discovery*, IOP Publishing, v. 8, n. 1, p. 014008, jul 2015. Disponível em: <https://doi.org/10.1088%2F1749-4699%2F8%2F1%2F014008>. Citado na página 55.

BERGSTRA, J. S.; BARDENET, R.; BENGIO, Y.; KÉGL, B. Algorithms for hyper-parameter optimization. In: *Advances in neural information processing systems*. [s.n.], 2011. p. 2546–2554. Disponível em: <http://papers.nips.cc/paper/4443-algorithms-for-hyper>. Citado 3 vezes nas páginas 26, 48 e 56.

BIJAK, K.; THOMAS, L. C. Does segmentation always improve model performance in credit scoring? *Expert Systems with Applications*, Elsevier, v. 39, n. 3, p. 2433–2442, 2012. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417411012243>. Citado na página 25.

BISHOP, C. M. *Neural Networks for Pattern Recognition*. USA: Oxford University Press, Inc., 1995. Citado na página 40.

BRAGA, I.; CARMO, L. P. do; BENATTI, C. C.; MONARD, M. C. A note on parameter selection for support vector machines. In: SPRINGER. *Mexican International Conference on Artificial Intelligence*. 2013. p. 233–244. Disponível em: <https://link.springer.com/chapter/10.1007/978-3-642-45111-9_21>. Citado na página 26.

BRANCO, P.; TORGO, L.; RIBEIRO, R. P. A survey of predictive modeling on imbalanced domains. *ACM Computing Surveys (CSUR)*, ACM New York, NY, USA, v. 49, n. 2, p. 1–50, 2016. Disponível em: <https://dl.acm.org/doi/epdf/10.1145/2907070>. Citado na página 24.

BRASIL. Lei complementar nº 101, de 4 de maio de 2000. estabelece normas de finanças públicas voltadas para a responsabilidade na gestão fiscal e dá outras providências. *Diário Oficial da União*, Brasília, DF, 2000. Citado na página 32.

BRASIL. *Parecer AGU/MC-02/04. Limitações impostas pela Lei nº 9.504, de 30 de setembro de 1997*. [S.l.], 2004. Citado na página 22.

BRASIL. *Portaria Interministerial nº 424/2016, de 24 de novembro de 2011. Estabelece normas para execução do disposto no Decreto no 6.170, de 25 de julho de 2007, que dispõe sobre as normas relativas às transferências de recursos da União mediante convênios e contratos de repasse*. 2016. Citado 2 vezes nas páginas 17 e 33.

BRASIL, R. F. d. *Carga Tributária no Brasil*. 2017. Disponível em: <http://receita.economia.gov.br/dados/receitadata/estudos-e-tributarios-e-aduaneiros/estudos-e-estatisticas/carga-tributaria-no-brasil/carga-tributaria-2017.pdf>. Citado na página 16.

BRAVO, C.; THOMAS, L. C.; WEBER, R. Improving credit scoring by differentiating defaulter behaviour. *Journal of the Operational Research Society*, Taylor & Francis, v. 66, n. 5, p. 771–781, 2015. Disponível em: <https://doi.org/10.1057/jors.2014.50>. Citado na página 23.

BREIMAN, L. Bagging predictors. *Machine Learning*, v. 24, p. 123–140, 1996. Citado na página 41.

BREIMAN, L.; FRIEDMAN, J. H.; OLSHEN, R. A.; STONE, C. J. *Classification and regression trees*. [S.l.]: Taylor & Francis Group, 1984. Citado na página 41.

BROWN, I.; MUES, C. An experimental comparison of classification algorithms for imbalanced credit scoring data sets. *Expert Systems with Applications*, Elsevier, v. 39, n. 3, p. 3446–3453, 2012. Disponível em: <https://www.sciencedirect.com/science/article/pii/S095741741101342X>. Citado na página 25.

CANDELIERI, A.; ARCHETTI, F. Sequential model based optimization with black-box constraints: Feasibility determination via machine learning. In: AIP PUBLISHING LLC. *AIP Conference Proceedings*. 2019. v. 2070, n. 1, p. 020010. Disponível em: <https://aip.scitation.org/doi/abs/10.1063/1.5089977>. Citado na página 47.

CHAPMAN, P.; CLINTON, J.; KERBER, R.; KHABAZA, T.; REINARTZ, T.; SHEARER, C.; WIRTH, R. *CRISP-DM 1.0. Step-by-step data mining guide. Edited by CRISP-DM Consortium.* 2000. Acessado em 06/02/2020. Disponível em: <ftp://ftp.software.ibm.com/software/analytics/spss/support/Modeler/Documentation/14/UserManual/CRISP-DM.pdf>. Citado 2 vezes nas páginas 30 e 31.

CHAWLA, N. V.; BOWYER, K. W.; HALL, L. O.; KEGELMEYER, W. P. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, v. 16, p. 321–357, 2002. Disponível em: <https://www.jair.org/index.php/jair/article/view/10302>. Citado na página 37.

CHEN, F.-L.; LI, F.-C. Combination of feature selection approaches with svm in credit scoring. *Expert systems with applications*, Elsevier, v. 37, n. 7, p. 4902–4909, 2010. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417409010719>. Citado na página 25.

CHEN, T.; GUESTRIN, C. Xgboost: A scalable tree boosting system. In: *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. [S.l.]: ACM, 2016. p. 785–794. Citado 3 vezes nas páginas 28, 40 e 42.

CHYI, Y.-M. Classification analysis techniques for skewed class distribution problems. *Department of Information Management, National Sun Yat-Sen University*, 2003. Citado na página 24.

CLAESEN, M.; MOOR, B. D. Hyperparameter search in machine learning. arxiv 2015. *arXiv preprint arXiv:1502.02127*, 2015. Disponível em: <https://arxiv.org/abs/1502.02127>. Citado na página 45.

CODD, E. F. A relational model of data for large shared data banks. *Commun. ACM*, Association for Computing Machinery, v. 13, n. 6, p. 377–387, jun. 1970. Disponível em: <https://doi.org/10.1145/362384.362685>. Citado na página 50.

CRONE, S. F.; FINLAY, S. Instance sampling in credit scoring: An empirical study of sample size and balancing. *International Journal of Forecasting*, Elsevier, v. 28, n. 1, p. 224–238, 2012. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0169207011001403>. Citado na página 25.

CROOK, J. N.; EDELMAN, D. B.; THOMAS, L. C. Recent developments in consumer credit risk assessment. *European Journal of Operational Research*, v. 183, n. 3, p. 1447 – 1465, 2007. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0377221706011866>. Citado na página 22.

DANGETI, P. *Statistics for Machine Learning*. [S.l.]: Packt Publishing, 2017. Citado na página 46.

DAOUD, E. A. Comparison between xgboost, lightgbm and catboost using a home credit dataset. *International Journal of Computer and Information Engineering*, v. 13, n. 1, p. 6–10, 2019. Citado na página 28.

DASTILE, X.; CELIK, T.; POTSANE, M. Statistical and machine learning models in credit scoring: A systematic literature survey. *Applied Soft Computing*, v. 91, p. 106263, 2020. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1568494620302039>. Citado 4 vezes nas páginas 23, 24, 26 e 37.

DIAMANTIDIS, N.; KARLIS, D.; GIAKOUMAKIS, E. Unsupervised stratification of cross-validation for accuracy estimation. *Artificial Intelligence*, v. 116, n. 1, p. 1 – 16, 2000. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0004370299000946>. Citado na página 58.

DOUZAS, G.; BACAO, F. Self-organizing map oversampling (somo) for imbalanced data set learning. *Expert systems with Applications*, Elsevier, v. 82, p. 40–52, 2017. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417417302324>. Citado na página 24.

ECONOMIA, B. M. da. *Painéis +Brasil*. Brasília, 2019. Disponível em: <http://www.transferenciasabertas.planejamento.gov.br>. Acesso em: 16 de jul. de 2019. Citado na página 18.

ECONOMIA, B. M. da. *Plataforma +Brasil*. Brasília, 2019. Disponível em: <http://plataformamaisbrasil.gov.br/>. Acesso em: 16 de jul. de 2019. Citado 3 vezes nas páginas 17, 50 e 51.

FAWCETT, T. An introduction to roc analysis. *Pattern recognition letters*, Elsevier, v. 27, n. 8, p. 861–874, 2006. Disponível em: <https://www.sciencedirect.com/science/article/pii/S016786550500303X>. Citado na página 64.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From data mining to knowledge discovery in databases. *AI magazine*, v. 17, n. 3, p. 37–37, 1996. Disponível em: <https://wvvw.aaai.org/ojs/index.php/aimagazine/article/view/1230>. Citado na página 29.

FEURER, M.; SPRINGENBERG, J. T.; HUTTER, F. Initializing bayesian hyperparameter optimization via meta-learning. In: *Twenty-Ninth AAAI Conference on Artificial Intelligence*. [s.n.], 2015. Disponível em: <https://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/viewPaper/10029>. Citado na página 26.

FLACH, P.; KULL, M. Precision-recall-gain curves: Pr analysis done right. In: *Advances in neural information processing systems*. [s.n.], 2015. p. 838–846. Disponível em: <https://papers.nips.cc/paper/5867-precision-recall-gain-curves-pr-analysis-done-right.pdf>. Citado na página 26.

FREUND, Y.; SCHAPIRE, R. E. A decision-theoretic generalization of on-line learning and an application to boosting. *Proc. 2nd European Conf. on Computational Learning Theory*, p. 23–37, 1995. Citado na página 42.

FRIEDMAN, J.; HASTIE, T.; TIBSHIRANI, R. Additive logistic regression: a statistical view of boosting. *Annals of Statistics*, v. 28, p. 337–407, 2000. Citado na página 42.

FRIEDMAN, J.; HASTIE, T.; TIBSHIRANI, R. Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, v. 29, p. 1189–1232, 2001. Disponível em: <https://statweb.stanford.edu/~jhf/ftp/trebst.pdf>. Citado na página 42.

FRIEDMAN, J. H. Stochastic gradient boosting. *Computational Statistics and Data Analysis*, v. 38, p. 367–378, 2002. Citado 2 vezes nas páginas 42 e 44.

FRIEDRICHS, F.; IGEL, C. Evolutionary tuning of multiple svm parameters. *Neurocomputing*, Elsevier, v. 64, p. 107–117, 2005. Citado na página 26.

GUDIVADA, V.; IRFAN, M.; FATHI, E.; RAO, D. Chapter 5 - cognitive analytics: Going beyond big data analytics and machine learning. In: GUDIVADA, V. N.; RAGHAVAN, V. V.; GOVINDARAJU, V.; RAO, C. (Ed.). *Cognitive Computing: Theory and Applications*. Elsevier, 2016, (Handbook of Statistics, v. 35). p. 169 – 205. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0169716116300517>. Citado na página 39.

GUISO, L.; SAPIENZA, P.; ZINGALES, L. The determinants of attitudes toward strategic default on mortgages. *The Journal of Finance*, v. 68, n. 4, p. 1473–1515, 2013. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1111/jofi.12044>. Citado na página 23.

HAIXIANG, G.; YIJING, L.; SHANG, J.; MINGYUN, G.; YUANYUE, H.; BING, G. Learning from class-imbalanced data: Review of methods and applications. *Expert Systems with Applications*, v. 73, p. 220 – 239, 2017. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0957417416307175>. Citado 2 vezes nas páginas 36 e 37.

HAJEK, P.; MICHALAK, K. Feature selection in corporate credit rating prediction. *Knowledge-Based Systems*, Elsevier, v. 51, p. 72–84, 2013. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0950705113002104>. Citado na página 25.

HAN, J.; KAMBER, M.; PEI, J. *Data Mining*. Third edition. [S.l.]: Morgan Kaufmann, 2012. Citado na página 29.

HAND, D. J.; JACKA, S. D. *Statistics in finance*. [S.l.]: John Wiley & Sons, 1998. Citado na página 35.

HARRIS, T. Credit scoring using the clustered support vector machine. *Expert Systems with Applications*, v. 42, n. 2, p. 741 – 750, 2015. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0957417414005119>. Citado na página 23.

HAWKINS, D. M. The problem of overfitting. *Journal of Chemical Information and Computer Sciences*, v. 44, n. 1, p. 1–12, 2004. Disponível em: <https://doi.org/10.1021/ci0342472>. Citado na página 32.

HULTQUIST, C.; CHEN, G.; ZHAO, K. A comparison of gaussian process regression, random forests and support vector regression for burn severity assessment in diseased forests. *Remote sensing letters*, Taylor & Francis, v. 5, n. 8, p. 723–732, 2014. Citado na página 48.

IOFFE, S.; SZEGEDY, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015. Disponível em: <https://arxiv.org/pdf/1502.03167.pdf>. Citado na página 54.

JADHAV, S.; HE, H.; JENKINS, K. Information gain directed genetic algorithm wrapper feature selection for credit rating. *Applied Soft Computing*, Elsevier, v. 69, p. 541–553, 2018. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1568494618302242>. Citado na página 25.

JANGER, E. J.; BLOCK-LIEB, S. The myth of the rational borrower: Behaviorism, rationality and the misguided reform of bankruptcy law. *Texas Law Review*, v. 84, n. 6, p. 1481, 2006. Citado na página 23.

Jiang, M.; Ji, F.; Li, R. The applied research of credit scoring combination model based on sa-ga algorithm. In: *2011 Fourth International Conference on Business Intelligence and Financial Engineering*. [s.n.], 2011. p. 491–494. Disponível em: <https://ieeexplore.ieee.org/document/6121187>. Citado na página 27.

JUNIOR, L. M.; NARDINI, F. M.; RENSO, C.; TRANI, R.; MACEDO, J. A. A novel approach to define the local region of dynamic selection techniques in imbalanced credit scoring problems. *Expert Systems with Applications*, v. 152, p. 113351, 2020. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0957417420301767>. Citado na página 36.

KE, G.; MENG, Q.; FINLEY, T.; WANG, T.; CHEN, W.; MA, W.; YE, Q.; LIU, T.-Y. Lightgbm: A highly efficient gradient boosting decision tree. In: GUYON, I.; LUXBURG, U. V.; BENGIO, S.; WALLACH, H.; FERGUS, R.; VISHWANATHAN, S.; GARNETT, R. (Ed.). *Advances in Neural Information Processing Systems 30*. Curran Associates, Inc., 2017. p. 3146–3154. Disponível em: <http://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree.pdf>. Citado na página 28.

KOHAVI, R. et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: MONTREAL, CANADA. *Ijcai*. [S.l.], 1995. v. 14, n. 2, p. 1137–1145. Citado na página 55.

KOTU, V.; DESHPANDE, B. *Predictive Analytics and Data Mining. Concepts and Practice with Rapidminer*. [S.l.]: Morgan Kaufmann, 2015. Citado na página 30.

LASKO, T. A.; BHAGWAT, J. G.; ZOU, K. H.; OHNO-MACHADO, L. The use of receiver operating characteristic curves in biomedical informatics. *Journal of biomedical informatics*, Elsevier, v. 38, n. 5, p. 404–415, 2005. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1532046405000171>. Citado na página 64.

LEE, T.-S.; CHIU, C.-C.; CHOU, Y.-C.; LU, C.-J. Mining the customer credit using classification and regression tree and multivariate adaptive regression splines. *Computational Statistics & Data Analysis*, v. 50, n. 4, p. 1113 – 1130, 2006. Citado na página 36.

LESSMANN, S.; SEOW, H.-V.; BAESENS, B.; THOMAS, L. C. Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. In: *European Journal of Operational Research*. Elsevier, 2015. v. 247, p. 124–136. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0377221715004208>. Citado na página 23.

LIANG, D.; TSAI, C.-F.; WU, H.-T. The effect of feature selection on financial distress prediction. *Knowledge-Based Systems*, Elsevier, v. 73, p. 289–297, 2015. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0950705114003773>. Citado na página 25.

LOPES, R. G.; CARVALHO, R. N.; LADEIRA, M.; CARVALHO, R. S. Predicting recovery of credit operations on a brazilian bank. In: *15th International Conference on Machine Learning and Applications (ICMLA)*. [S.l.]: IEEE, 2016. Citado na página 24.

LOUZADA, F.; ARA, A.; FERNANDES, G. B. Classification methods applied to credit scoring: Systematic review and overall comparison. *Surveys in Operations Research and Management Science*, Elsevier, v. 21, n. 2, p. 117–134, 2016. Disponível em: <https://arxiv.org/pdf/1602.02137.pdf>. Citado na página 23.

LUSA, L. et al. Smote for high-dimensional class-imbalanced data. *BMC bioinformatics*, Springer, v. 14, n. 1, p. 106, 2013. Disponível em: <https://link.springer.com/article/10.1186/1471-2105-14-106>. Citado na página 68.

MALDONADO, S.; PÉREZ, J.; BRAVO, C. Cost-based feature selection for support vector machines: An application in credit scoring. *European Journal of Operational Research*, Elsevier, v. 261, n. 2, p. 656–665, 2017. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0377221717301595>. Citado na página 25.

MANCISIDOR, R. A.; KAMPFFMEYER, M.; AAS, K.; JENSSEN, R. Segment-based credit scoring using latent clusters in the variational autoencoder. *arXiv preprint arXiv:1806.02538*, 2018. Disponível em: <https://arxiv.org/abs/1806.02538>. Citado na página 25.

MANI, I.; ZHANG, I. knn approach to unbalanced data distributions: a case study involving information extraction. In: *Proceedings of workshop on learning from imbalanced datasets*. [s.n.], 2003. v. 126. Disponível em: <https://www.site.uottawa.ca/~nat/Workshop2003/jzhang.pdf>. Citado na página 24.

Mantovani, R. G.; Horváth, T.; Cerri, R.; Vanschoren, J.; Carvalho, A. C. P. L. F. d. Hyper-parameter tuning of a decision tree induction algorithm. In: *2016 5th Brazilian Conference on Intelligent Systems (BRACIS)*. [S.l.: s.n.], 2016. p. 37–42. Citado na página 26.

MEHTA, M.; AGRAWAL, R.; RISSANEN, J. Sliq: A fast scalable classifier for data mining. In: SPRINGER. *International conference on extending database technology*. [S.l.], 1996. p. 18–32. Citado na página 41.

MINISTÉRIO DA TRANSPARÊNCIA E CONTROLADORIA-GERAL DA UNIÃO. Relatório de auditoria. In: _____. *Avaliação da Gestão das Transferências Voluntárias da União*. Brasília, 2018. Disponível em: <https://auditoria.cgu.gov.br/download/11014.pdf>. Acesso em: 25 de set. de 2019. Citado na página 19.

MOUTINHO, J.; KNIESS, C. Transferências voluntárias da união para municípios brasileiros: Identificação de correlação entre variáveis. *Revista de Gestão e Projetos*, v. 8, n. 1, p. 90–101, 2017. Citado na página 16.

MUKHOPADHYAY, S. 13 - artificial neural network applications in textile composites. In: MAJUMDAR, A. (Ed.). *Soft Computing in Textile Engineering*. Woodhead Publishing, 2011, (Woodhead Publishing Series in Textiles). p. 329 – 349. Disponível em: <http://www.sciencedirect.com/science/article/pii/B9781845696634500139>. Citado na página 39.

NIKOLIC, N.; ZARKIC-JOKSIMOVIC, N.; STOJANOVSKI, D.; JOKSIMOVIC, I. The application of brute force logistic regression to corporate credit scoring models: Evidence from serbian financial statements. *Expert Systems with Applications*, v. 40, n. 15, p. 5932 – 5944, 2013. Citado na página 23.

NISBET, R.; ELDER, J.; MINER, G. *Handbook of statistical analysis and data mining applications*. [S.l.]: Academic Press, 2009. Citado na página 30.

ONG, C.-S.; HUANG, J.-J.; TZENG, G.-H. Building credit scoring models using genetic programming. *Expert Systems with Applications*, v. 29, n. 1, p. 41 – 47, 2005. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0957417405000059>. Citado na página 23.

PAULA, D. A. V. d.; ARTES, R.; AYRES, F.; MINARDI, A. M. A. F. Estimating credit and profit scoring of a brazilian credit union with logistic regression and machine-learning techniques. *RAUSP Management Journal*, scielo, v. 54, p. 321 – 336, 09 2019. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S2531-04882019000300321&nrm=iso>. Citado na página 23.

PAULA, E. L. de. *Mineração de dados como suporte à detecção de lavagem de dinheiro.* Dissertação (Mestrado) — Universidade de Brasília, 2016. Citado na página 24.

PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V. et al. Scikit-learn: Machine learning in python. *Journal of machine learning research*, v. 12, n. Oct, p. 2825–2830, 2011. Citado na página 55.

QUINLAN, J. The morgan kaufmann series in machine learning. *San Mateo*, 1993. Citado na página 41.

QUINLAN, J. R. Induction of decision trees. *Machine Learning*, v. 1, p. 81–106, 1986. Citado na página 41.

RIDD, P.; GIRAUD-CARRIER, C. G. Using metalearning to predict when parameter optimization is likely to improve classification accuracy. In: *MetaSel@ ECAI*. [s.n.], 2014. p. 18–23. Disponível em: <https://repositorio.inesctec.pt/bitstream/123456789/4540/1/P-00G-699.pdf#page=23>. Citado na página 26.

Romanyuk, K. Credit scoring based on a continuous scale for on-line credit quality control. In: *2015 IEEE International Conference on Evolving and Adaptive Intelligent Systems (EAIS)*. [s.n.], 2015. p. 1–5. Disponível em: <https://ieeexplore.ieee.org/document/7368796>. Citado na página 22.

SAITO, T.; REHMSMEIER, M. The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets. *PloS one*, Public Library of Science, v. 10, n. 3, 2015. Citado na página 66.

SHAFRANOVICH, Y. *Common Format and MIME Type for Comma-Separated Values (CSV) Files*. [S.l.], 2005. Disponível em: <https://tools.ietf.org/html/rfc4180>. Citado na página 50.

Shahriari, B.; Swersky, K.; Wang, Z.; Adams, R. P.; de Freitas, N. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, v. 104, n. 1, p. 148–175, 2016. Disponível em: <https://ieeexplore.ieee.org/document/7352306>. Citado 2 vezes nas páginas 47 e 48.

SINGH, B. K.; VERMA, K.; THOKE, A. Investigations on impact of feature normalization techniques on classifier's performance in breast tumor classification. *International Journal of Computer Applications*, Foundation of Computer Science, v. 116, n. 19, 2015. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.695.1851&rep=rep1&type=pdf>. Citado na página 54.

SNOEK, J.; LAROCHELLE, H.; ADAMS, R. P. Practical bayesian optimization of machine learning algorithms. In: *Advances in neural information processing systems*. [s.n.], 2012. p. 2951–2959. Disponível em: <http://papers.nips.cc/paper/4522-practical-bayesian-optimization>. Citado na página 45.

SOHN, S. Y.; KIM, D. H.; YOON, J. H. Technology credit scoring model with fuzzy logistic regression. *Applied Soft Computing*, v. 43, p. 150 – 158, 2016. Disponível em: <https://www.sciencedirect.com/science/article/pii/S156849461630076X>. Citado na página 35.

STONE, E. A. Predictor performance with stratified data and imbalanced classes. *Nature methods*, Nature Publishing Group, v. 11, n. 8, p. 782, 2014. Citado na página 59.

SUN, J.; LANG, J.; FUJITA, H.; LI, H. Imbalanced enterprise credit evaluation with dte-sbd: Decision tree ensemble based on smote and bagging with differentiated sampling rates. *Information Sciences*, v. 425, p. 76 – 91, 2018. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0020025517310083>. Citado na página 24.

TAHIR, M. A.; KITTLER, J.; MIKOLAJCZYK, K.; YAN, F. A multiple expert approach to the class imbalance problem using inverse random under sampling. In: SPRINGER. *International workshop on multiple classifier systems*. 2009. p. 82–91. Disponível em: <http://epubs.surrey.ac.uk/733262/1/MCS09.pdf>. Citado na página 37.

THARWAT, A. Classification assessment methods. *Applied Computing and Informatics*, 2018. Disponível em: <http://www.sciencedirect.com/science/article/pii/S2210832718301546>. Citado 4 vezes nas páginas 26, 58, 59 e 60.

THOMAS, L. C. A survey of credit and behavioural scoring: forecasting financial risk of lending to consumers. *International Journal of Forecasting*, v. 16, n. 2, p. 149 – 172, 2000. Citado 2 vezes nas páginas 35 e 52.

THORNTON, C.; HUTTER, F.; HOOS, H. H.; LEYTON-BROWN, K. Auto-weka: Combined selection and hyperparameter optimization of classification algorithms. In: *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. [S.l.: s.n.], 2013. p. 847–855. Citado na página 26.

TIAN, Y.; YONG, Z.; LUO, J. A new approach for reject inference in credit scoring using kernel-free fuzzy quadratic surface support vector machines. *Applied Soft Computing*, v. 73, p. 96 – 105, 2018. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1568494618304812>. Citado na página 36.

VASSILIADIS, P. A survey of extract-transform-load technology. *International Journal of Data Warehousing and Mining*, v. 5, p. 1–27, 07 2009. Citado na página 50.

WANG, B.; GONG, N. Z. Stealing hyperparameters in machine learning. In: *2018 IEEE Symposium on Security and Privacy (SP)*. [s.n.], 2018. p. 36–52. Disponível em: <https://ieeexplore.ieee.org/abstract/document/8418595>. Citado na página 44.

WANG, Y.; NI, X. S. A xgboost risk model via feature selection and bayesian hyper-parameter optimization. *arXiv e-prints*, p. arXiv:1901.08433, 2019. Disponível em: <https://arxiv.org/abs/1901.08433>. Citado 2 vezes nas páginas 26 e 27.

WISTUBA, M.; SCHILLING, N.; SCHMIDT-THIEME, L. Hyperparameter search space pruning – a new component for sequential model-based hyperparameter optimization. In: APPICE, A.; RODRIGUES, P. P.; COSTA, V. S.; GAMA, J.; JORGE, A.; SOARES, C. (Ed.). *Machine Learning and Knowledge Discovery in Databases*. Cham: Springer International Publishing, 2015. p. 104–119. ISBN 978-3-319-23525-7. Disponível em: <https://link.springer.com/chapter/10.1007/978-3-319-23525-7_7>. Citado na página 47.

WONGCHINSRI, P.; KURATACH, W. Sr-based binary classification in credit scoring. In: *2017 14th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*. [s.n.], 2017. p. 385–388. Disponível em: <https://ieeexplore.ieee.org/document/8096254>. Citado na página 25.

XIA, Y.; LIU, C.; LI, Y.; LIU, N. A boosted decision tree approach using bayesian hyper-parameter optimization for credit scoring. *Expert Systems with Applications*, v. 78, p. 225 – 241, 2017. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0957417417301008>. Citado 4 vezes nas páginas 23, 26, 44 e 54.

Yadav, S.; Shukla, S. Analysis of k-fold cross-validation over hold-out validation on colossal datasets for quality classification. In: *2016 IEEE 6th International Conference on Advanced Computing (IACC)*. [S.l.: s.n.], 2016. p. 78–83. Citado na página 55.

YEN, S.-J.; LEE, Y.-S. Cluster-based under-sampling approaches for imbalanced data distributions. *Expert Systems with Applications*, v. 36, n. 3, Part 1, p. 5718 – 5727, 2009. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0957417408003527>. Citado na página 24.

ZHANG, X.; YANG, Y.; ZHOU, Z. A novel credit scoring model based on optimized random forest. In: IEEE. *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)*. 2018. p. 60–65. Disponível em: <https://ieeexplore.ieee.org/abstract/document/8301707/>. Citado na página 25.

ZIĘBA, M.; TOMCZAK, S. K.; TOMCZAK, J. M. Ensemble boosted trees with synthetic features generation in application to bankruptcy prediction. *Expert Systems with Applications*, Elsevier, v. 58, p. 93–101, 2016. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0957417416301592>. Citado na página 28.