Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Mineração de Dados para Previsão de Renda de Clientes com Contas-Correntes Digitais

Roberto Nunes Mourão

Dissertação apresentada como requisito parcial para conclusão do
Mestrado Profissional em Computação Aplicada

Orientador
Prof. Dr. Guilherme N. Ramos

Brasília
2018

Ficha catalográfica elaborada automaticamente,
com os dados fornecidos pelo(a) autor(a)

NR642m

Nunes Mourão, Roberto
Mineração de Dados para Previsão de Renda de Clientes com Contas-Correntes Digitais / Roberto Nunes Mourão; orientador Guilherme Novaes Ramos. -- Brasília, 2018.
70 p.

Dissertação (Mestrado - Mestrado Profissional em Computação Aplicada) -- Universidade de Brasília, 2018.

1. Mineração de Dados. 2. Classificação. 3. Previsão de Renda. 4. Modelo de Público Semelhante. 5. Indústria Bancária. I. Novaes Ramos, Guilherme, orient. II. Título.

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Mineração de Dados para Previsão de Renda de Clientes com Contas-Correntes Digitais

Roberto Nunes Mourão

Dissertação apresentada como requisito parcial para conclusão do
Mestrado Profissional em Computação Aplicada

Prof. Dr. Guilherme N. Ramos (Orientador)
CIC/UnB

Prof. Dr. Daniel G. e Silva
Universidade de Brasília

Prof. Dr. Donald M. Pianto
Universidade de Brasília

Prof. Dr. Aleteia Patricia Favacho de Araujo
Coordenador do Programa de Pós-graduação em Computação Aplicada

Brasília, 29 de junho de 2018

Dedicatória

Dedico este trabalho a meu pai, Raimundo Coêlho Mourão, que um dia sonhou com um futuro melhor para os filhos em um lugar em que pudessem estudar.

Agradecimentos

Agradeço primeiramente e sempre a Deus, que me ajudou até aqui (Samuel 7.12). Esses últimos dois anos foram um período de extremas dificuldades no âmbito pessoal e profissional. Vi mais uma vez a fidelidade d'Ele no cumprimento de Suas promessas, onde pude me amparar durante devastadoras tempestades, encontrando paz para realizar um sonho de infância.

Não tenho palavras para agradecer o apoio dado pela minha esposa e filhos, que muitas vezes deixaram de passar momentos em família porque meu "dever de casa" era muito grande e que ainda por cima eu tinha que escrever um livro "todinho".

Agradeço ao prof. Dr. Guilherme Ramos cuja dedicação ímpar motivou-me a cada dia nessa árdua tarefa. Certamente sua orientação aumentou em vários graus de qualidade este trabalho e ensinou-me muito para os próximos, que espero não serem poucos.

Agradeço a todos os meus professores do Programa de Pós-graduação em Computação Aplicada (PPCA). Em especial o prof. Dr. Marcelo Ladeira e o prof. Dr. Donald Pianto pelos aconselhamentos e apoio.

Agradeço a meus colegas de empresa, que me auxiliaram apoiando minha participação no Programa de Pós-graduação, na obtenção dos dados usados nessa pesquisa e na orientação sobre questões do problema de negócio. Em especial Roberto Paiva Zorrón, Daniel Regis Filho, Fabiana Lauxen, Rogério Lopes, Alexandre Duarte, Sérgio Diogo Barbosa, Analaura Morais e Diogo Kugler.

Por fim, agradeço aos colegas e funcionários do PPCA.

Resumo

Um banco brasileiro disponibilizou a abertura de conta bancária por meio de um aplicativo móvel, o que geralmente exige muito pouca informação do usuário. Essa falta de dados prejudica os atuais modelos preditivos aplicados na seleção de clientes para campanhas de *marketing*. Com o intuito de atenuar isso, este trabalho investiga o uso da Mineração de Dados a fim de criar um modelo preditivo capaz de identificar a renda desses clientes. Para tanto, como treinamento, usa os dados de um grupo de clientes, os quais, de forma semelhante, utilizam o aplicativo móvel do banco. Todavia, abriram suas contas indo às agências, local onde comprovaram suas rendas. Os dados utilizados incluem informações cadastrais, demográficas e características dos *smartphones* dos clientes. O processo CRISP-DM foi aplicado para comparar várias abordagens, tais como: Regressão Logística, *Random Forest*, Redes Neurais Artificiais, *Gradient Boosting Machine* e *Hill-climbing Ensemble Selection with Bootstrap Sampling*. Os resultados mostraram que o *Gradient Boosting Machine* obteve o melhor resultado com Acurácia de 92 % e *F-Measure* de 62 %.

Palavras-chave: Mineração de Dados, Classificação, Previsão de Renda, Modelo de Público Semelhante, Indústria Bancária

Abstract

Digital bank accounts require little information from customers to enable simple banking services, and the absence of income data hampers a focused targeting of customers for additional products/services.

This study presents a comparison of predictive models to identify a customer's income bracket, by mining digital account data. The information available to build the models includes customers' registered data, demographics, house prices, and smartphone features. The models are applied to a set of customers with regular accounts, who have income data and features similar to those with digital accounts. The models' performances are compared to the model currently in use in a private bank.

Several approaches were used, in a CRISP-DM process: Logistic Regression, Random Forest, Artificial Neural Networks, Gradient Boosting Machine, and Hill-Climbing Ensemble with Bootstrap Sampling. Experimental results indicate the Gradient Boosting Machine model achieved the best results, with a 92% Accuracy and a 62% F-Measure.

Keywords: Data Mining, Classification, Income Prediction, Look-alike Model, Banking

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Abreviaturas e Siglas

ADASYN *Adaptive Synthetic Sampling Approach for Imbalanced Learning.*

ANN *Artificial Neural Networks.*

API *Application Programming Interface.*

AUC *Area Under the Curve.*

BCB Banco Central do Brasil.

CEP Código de Endereçamento Postal.

CPF Cadastro de Pessoas Físicas.

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

CSV *Comma-Separated Values.*

FEBRABAN Federação Brasileira de Bancos.

FPR *False Positive Rate.*

GBM *Gradient Boosting Machine.*

HCES-Bag *Hill-climbing Ensemble Selection with Bootstrap Sampling.*

IBGE Instituto Brasileiro de Geografia e Estatística.

KNN *K-Nearest Neighbors.*

LR *Logistic Regression.*

MD Mineração de Dados.

OLS *Ordinary Least Squares.*

PCA *Principal Component Analysis.*

PIB Produto Interno Bruto.

RF *Random Forest.*

RFB Receita Federal do Brasil.

ROC *Receiver Operating Characteristic.*

SFN Sistema Financeiro Nacional.

SMOTE *Synthetic Minority Over-sampling Technique.*

SMOTEENN *Synthetic Minority Over-sampling Technique and Edited Nearest Neighbors.*

SMOTETomek *Synthetic Minority Over-sampling Technique and Tomek Links.*

SQL *Structured Query Language.*

TPR *True Positive Rate.*

VIF *Variance Inflation Factor.*

Capítulo 1

Introdução

Contas-correntes que podem ser abertas usando somente um *smartphone* são uma grande inovação do setor bancário brasileiro. Essas *contas digitais* possibilitaram que uma maior parte da população pudesse ser inserida no mercado bancário, o qual fornece serviços de intermediação financeira, poupança e crédito. Contudo, a ausência de comprovação de renda desses clientes tem dificultado a adequada oferta de produtos e serviços a eles e, consequentemente, a obtenção de novos ativos por meio desse público-alvo.

1.1 Definição do Problema

No ano de 2016, o Banco Central do Brasil (BCB) permitiu que instituições financeiras disponibilizassem a abertura de contas-correntes por meio eletrônico, sem a necessidade de que os clientes se deslocassem a uma agência física [1]. As chamadas *contas-correntes digitais* ou, simplesmente, *contas digitais* permitem que seus clientes façam transferências financeiras entre si, paguem contas e realizem recarga de crédito em celulares pré-pagos, com um limite de movimentações de R$ 5.000,00 por mês.

Estima-se que cerca de 53% dos domicílios no Brasil se tornaram público-alvo das contas digitais [2]. Até o primeiro trimestre de 2017, segundo apuração[1] da Federação Brasileira de Bancos (FEBRABAN), existiam cerca de 940 mil clientes bancários que efetuavam transações somente utilizando contas-correntes digitais e a expectativa de crescimento para o final de 2017 era de 3,3 milhões de clientes.

Uma conta-corrente, contudo, é somente o primeiro produto disponibilizado aos novos clientes pelas instituições financeiras. Os bancos comerciais, com a intenção de aumentar seus lucros, oferecem aos seus correntistas diversos produtos e serviços, tais como: seguros, consórcios, investimentos e empréstimos.

[1] Disponível em `http://www.ciab.com.br`

Cada um dos produtos oferecidos pelas instituições financeiras segue regras preestabelecidas por órgãos reguladores do governo, em especial o Banco Central do Brasil (BCB). Um dos critérios para concessão de empréstimos, por exemplo, é a análise de crédito do cliente, cálculo efetuado a partir de diversos fatores, dos quais se destaca a comprovação de renda [3]. Além de mitigar o risco de inadimplência em operações de crédito, a correta informação da renda permite que as instituições financeiras direcionem seus produtos e serviços a um público apto a consumí-los.

De igual maneira, a renda dos clientes é base para a estratégia de *marketing* do Banco Alfa, denominado assim por sigilo. Correntistas digitais, no entanto, não tem obrigatoriedade de comprovar sua renda, dificultando o processo de seleção de clientes para oferta de produtos e serviços.

O cadastro reduzido da conta digital torna-se então um empecilho para o Banco Alfa ofertar novos produtos a esses clientes. Um passo natural seria convencer alguns dos novos correntistas a *evoluírem* suas contas digitais para uma conta convencional. Por um lado, os clientes obteriam maiores benefícios como o fim da limitação de R$ 5.000,00 de movimentação mensal e por outro lado, teriam de fornecer uma informação cadastral mais completa. Tornando-se detentor de uma conta convencional, o cliente de origem digital teria acesso a todo o rol de produtos disponibilizados pelo Banco Alfa.

Novamente a seleção de clientes torna-se um problema, dadas as parcas informações disponíveis acerca desse público. Há somente três informações que são realmente necessárias para a abertura de uma conta digital: o número do Cadastro de Pessoas Físicas (CPF), o número do Código de Endereçamento Postal (CEP) e um número de telefone celular. A partir do CPF é possível obter de órgãos governamentais, como o Banco Central do Brasil e a Receita Federal do Brasil, mais informações: data de nascimento, sexo e grau de endividamento. O CEP de residência do cliente permite identificar a Unidade Federativa, o município, a cidade e o bairro. O telefone celular fornece informações do modelo do aparelho. Mesmo usando essas fontes de dados, não é possível obter diretamente a informação sobre os rendimentos desses clientes.

No intuito de obter as informações de renda de seu público-alvo o Banco Alfa adquiriu, de uma empresa de apoio ao crédito, uma base de dados com a renda presumida de várias pessoas físicas. Essa base tem sido utilizada atualmente como referencial de renda na seleção de clientes contactados para a alteração de conta.

Todavia, na Figura 1.1 é possível perceber que a renda presumida indica valores bem menores que a informação comprovada dos clientes convencionais, principalmente para rendas maiores que R$ 10 mil.

Foi também efetuada uma análise de correlação separada por faixas de renda tradicionalmente utilizadas pelo Banco Alfa. A Tabela 1.1 indica uma baixa correlação entre a

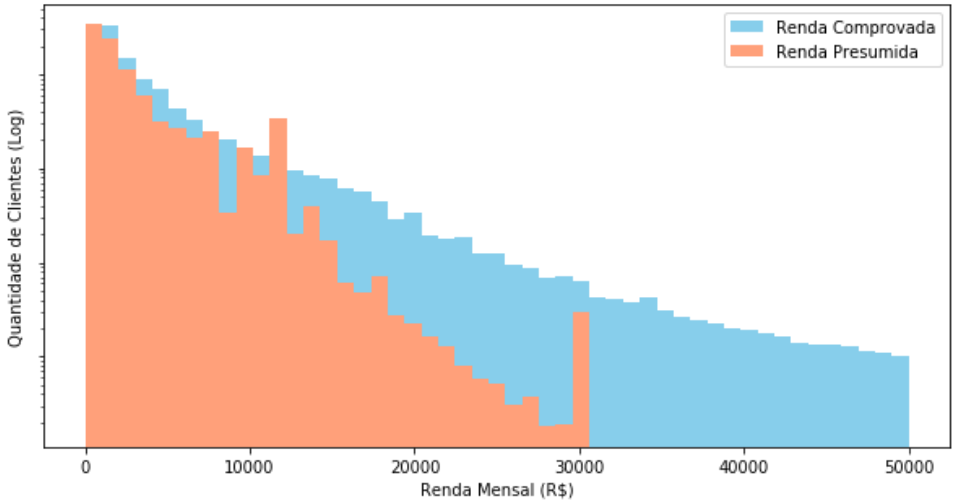


Figura 1.1: Comparativo entre renda presumida da base dados externa e a renda comprovada dos clientes que utilizam o aplicativo móvel do Banco Alfa.

renda comprovada e a renda presumida em todas as faixas, sendo que a faixa de maior renda apresentou a menor correlação.

Tabela 1.1: Correlação entre a renda comprovada e a renda presumida da base de dados externa.

Faixa de Renda (R$)	**Correlação de Pearson**
[0,01, 1.000)	-0,056
[1.000, 4.000)	0,148
[4.000, 10.000)	0,197
[10.000, $+\infty$)	0,029

A Margem de Contribuição é a diferença entre a receita das vendas e os custos e despesas variáveis [4] e é usada pelo Banco Alfa para avaliar o retorno financeiro dos clientes. A Tabela 1.2 mostra a comparação das Margens de Contribuição projetadas ao longo de 6 meses, supondo que 20 mil contas digitais sejam abertas por mês, conforme expectativa de especialistas[2]. Estima-se que a Margem de Contribuição total das pessoas com renda informada pela fonte de dados externa corresponde a menos da metade da projeção da base interna.

Observando a distribuição de renda dos clientes do Banco Alfa com a renda informada pela base de dados externa e as projeções de rentabilidade, entende-se que a primeira não reflete adequadamente a renda dos correntistas desse Banco, podendo causar prejuízos se for utilizada para tomadas de decisão. Assim, o Banco Alfa concluiu que a informação atualmente disponível mostrou-se de pouco valor para identificar a renda dos clientes digitais, sendo necessário procurar uma solução para esse problema.

[2]Disponível em `http://www.ciab.com.br`

Tabela 1.2: Projeção Comparada das Margens de Contribuição Mensal a Partir de Diferentes Fontes de Informação de Renda

Mês	**Contas**	**Margem de Contribuição (R$)**	
		Fonte Externa	**Fonte Interna**
1	20.000	198.272,70	425.081,54
2	40.000	396.543,39	850.161,07
3	60.000	594.815,09	1.275.241,61
4	80.000	793.086,79	1.700.322,14
5	100.000	991.358,49	2.125.402,68
6	120.000	1.189.630,18	2.550.483,21
Total		**4.163.705,65**	**8.926.691,24**

1.2 Justificativa

Em função da problemática apresentada e baseado nos dados levantados, o Banco Alfa necessita de uma informação de renda mais acurada para ofertar produtos e serviços mais adequados aos seus clientes digitais, a fim de obter a máxima Margem de Contribuição e, consequentemente, maior retorno financeiro.

Além disso, os dados atualmente disponíveis ao Banco Alfa para determinar a oferta de produtos a seus clientes, informam rendas inferiores às reais e, segundo projeções, reduziriam o potencial de obtenção de Margem de Contribuição em 50%, ocasionando uma perda potencial de milhões de reais por mês em vendas de produtos e serviços bancários.

1.3 Hipótese

Acredita-se que um mecanismo de predição de renda criado a partir de um público semelhante ao público-alvo, mas que possua renda comprovada, utilizando informações cadastrais comuns nos dois públicos, dados sócio-demográficos dos clientes, comportamentos de movimentação financeira e informações do modelo dos *smartphones* terá maior confiabilidade em comparação à informação de renda presumida utilizada atualmente.

1.4 Objetivos

Esta pesquisa busca uma maneira de identificar clientes detentores de contas digitais com maior renda para uma oferta mais eficiente de produtos e serviços bancários. Este trabalho pretende utilizar uma abordagem de Mineração de Dados para inferir faixas de renda.

1.4.1 Objetivo Geral

Comparar entre si modelos de classificação de faixa de renda, criados a partir de algoritmos de Mineração de Dados, selecionando o modelo que identificar uma quantidade equilibrada de acertos nas diferentes faixas de renda, para finalmente compará-lo com a informação sobre renda adquirida de uma empresa de apoio ao crédito.

1.4.2 Objetivos Específicos

Detalhadamente, objetiva-se:

- definir faixas de renda que sejam relevantes para o negócio e adequadas para o modelo preditivo, baseada na estratégia de oferta de produtos do Banco Alfa e na divisão mais eficaz das faixas para a modelagem estatística;
- testar diferentes algoritmos de classificação de renda, escolhendo o modelo mais equilibrado (*F-Measure*);
- comparar os resultados do modelo escolhido com a solução atualmente utilizada pelo Banco Alfa, isto é, a informação de renda presumida adquirida de fonte externa.
- avaliar a qualidade do modelo escolhido na base de clientes digitais.

Capítulo 2

Revisão Teórica

Empresas costumam organizar suas informações sobre clientes, produtos e forças de vendas em banco de dados, podendo combinar essas informações para montar uma estratégia de oferta de produtos. Ao identificarem os clientes adequados para certos produtos, elas evitam o envio indiscriminado de ofertas, reduzindo o custo de envio da propaganda e aumentando as vendas. Além disso, sugere-se que empresas devam coletar informações sobre o clientes a cada interação, sendo que qualquer dado possui valor, seja ele de origem cadastral ou transacional. Essa quantidade de dados deve ser mantida de forma organizada, como em *data warehouses*, para o uso da equipe de **Mineração de Dados** [5].

A Mineração de Dados é o processo de descobrir informações úteis em grandes repositórios de dados, tudo automaticamente. As técnicas de MD permitem encontrar padrões novos e úteis que, de outra forma, permaneceriam desconhecidos [6]. A MD também pode ser definida como a construção de um modelo estatístico [7].

A resolução de um problema utilizando Mineração de Dados depende de etapas de *pré-processamento* e *pós-processamento*. O pré-processamento corresponde a transformar dados brutos em um formato apropriado para a Mineração de Dados. As etapas de pós-processamento estão relacionadas à avaliação do modelo de Mineração de Dados criado, de forma que seja assegurado que somente resultados válidos e úteis sejam incorporados aos sistemas de produção. As seções a seguir detalham técnicas utilizadas em cada um desses três momentos.

2.1 Pré-processamento

Passos comuns nessa etapa envolvem a obtenção, a limpeza e a seleção de dados relevantes para a tarefa de Mineração de Dados. Grande parte do esforço da criação de um modelo de MD é despendido nessa etapa [6].

Os dados podem vir em muitos formatos diferentes: texto, numéricos, categóricos [6]. Para cada um desses formatos podem ser aplicadas técnicas para torná-los adequados aos requisitos do problema de MD. Uma das transformações possíveis é igualar a escala de todas as variáveis. Uma maneira de realizar isso é redefinir os valores de uma váriavel para o intervalo [0, 1], usando a Equação 2.1 [8].

$$y = \frac{x - min}{max - min}, \tag{2.1}$$

onde:

y: valor transformado;

x: valor original;

min: menor valor da variável;

max: maior valor da variável.

Outro exemplo de transformação consiste em mudar a distribuição de valores para que esta se assemelhe a uma distribuição Normal. A transformação Box-Cox [9], Equação 2.2, pode ser utilizada para esse fim.

$$y = \begin{cases} \dfrac{x^{\lambda} - 1}{\lambda}, & \text{se } \lambda \neq 0. \\ ln\ x, & \text{se } \lambda = 0, \end{cases} \tag{2.2}$$

onde:

y: número transformado;

x: número original;

λ: valor que melhor aproxima o conjunto de valores a uma distribuição normal.

Transformações como as mostradas nas Equações Equação 2.1 e Equação 2.2 não mudam o tipo da variável, que continua numérica [6, 9]. Em outros casos podem ser necessárias transformações que tornam variáveis numéricas em categóricas. Uma das técnicas de categorização consiste no uso do algoritmo de criação de grupos denominado *K-Means* [6] e é detalhado nos procedimentos abaixo:

1. Selecionar K pontos como centróides iniciais.
2. Formar K grupos a partir dos elementos mais próximos a um centróide.

3. Recalcular os centróides.

4. Repetir passos 2 e 3 até que os centróides não mudem.

A avaliação da qualidade dos grupos criados deveria levar em consideração a quantidade ideal de grupos, a coesão e a separação. Coesão consiste em o quanto os elementos dentro de um grupo estão relacionados (ou próximos entre si). Separação, por outro lado, verifica o quanto grupos distintos estão separados. Uma métrica que une esses dois conceitos é o coeficiente *Silhouette*, dado pela Equação 2.3 [6]:

$$S = \frac{\sum_{i=1}^{n} \frac{b_i - a_i}{max(a_i, b_i)}}{n}, \tag{2.3}$$

onde:

n: número de elementos;

a_i: distância média entre o i-ésimo elemento e todos os outros elementos do mesmo grupo;

b_i: menor distância média entre o i-ésimo elemento e os elementos dos grupos vizinhos.

O valor do coeficiente de *Silhouette* varia entre -1 e 1, onde se assume que o menor valor corresponde a um agrupamento incorreto, enquanto que o maior valor sugere um "bom" agrupamento, com máxima coesão e separação [10].

Durante o pré-processamento, pode-se ainda identificar variáveis repetidas, que aumentariam o tempo de processamento da etapa de modelagem ou prejudicariam seus resultados [11]. Uma das situações encontradas é chamada de *colinearidade* e refere-se à situação em que duas ou mais variáveis explicativas são muito relacionadas entre si. Uma das maneiras de se identificar essa situação é utilizando o *Variance Inflation Factor* (VIF) [11]. Para calculá-lo, efetua-se uma Regressão Linear entre uma variável explicativa, definida como alvo, e todas as outras variáveis explicativas. Ao obter o *coeficiente de determinação* R^2, o VIF é dado pela Equação 2.4.

$$VIF = \frac{1}{1 - R^2} \tag{2.4}$$

A seleção de variáveis é dada pelo algoritmo a seguir:

1. calcular o VIF de cada variável;

2. retirar a variável de maior VIF;

3. executar os passos 1 e 2 até que não haja VIF maiores que 5 [11].

A execução desse procedimento tem como vantagem identificar não só a relação entre duas, mas entre várias variáveis explicativas, o que não seria possível se fosse aplicado o coeficiente de correlação de Pearson duas a duas variáveis [11].

Apesar da duplicidade de variáveis ser um problema, registros repetidos podem ajudar na fase de modelagem. Alguns algoritmos de MD obtém melhores resultados quando se encontra um número aproximadamente igual de observações das categorias de uma variável alvo, como algoritmos baseados em regras [6]. São exemplos de técnicas que podem ser utilizadas para balancear categorias [12]:

Random Undersampling: realiza uma amostragem aleatória simples nos dados pertencentes à categoria majoritária, para que a amostra fique com a mesma quantidade de registros que a categoria minoritária;

Random Oversampling: efetua uma amostragem simples com reposição da categoria minoritária até que essa amostra atinja o mesmo número de registros que a categoria majoritária;

Cluster Centroids: realiza um agrupamento *K-Means* com K igual ao número de registros da categoria minoritária na categoria majoritária, fornecendo assim um grupo de centróides na mesma quantidade da categoria minoritária;

Synthetic Minority Over-sampling Technique (SMOTE): gera registros da categoria minoritária por interpolação, até que haja o mesmo número de registros da categoria majoritária;

Synthetic Minority Over-sampling Technique and Edited Nearest Neighbors (SMOTEENN): similar ao SMOTE, mas removendo *outliers* (valores extremos) pela técnica de edição de vizinhos mais próximos [13], onde são removidos da amostra elementos que não pertencem à maioria da categoria majoritária entre os k vizinhos;

Synthetic Minority Over-sampling Technique and Tomek Links (SMOTETomek): similar ao SMOTE, mas removendo *outliers* pela técnica de *links de Tomek*. Um *link* de Tomek ocorre quando os dois vizinhos mais próximos são de categorias diferentes. A remoção ocorre na categoria majoritária;

Adaptive Synthetic Sampling Approach for Imbalanced Learning (ADASYN): similar ao SMOTE, gera registros por interpolação, mas utiliza somente registros considerados mais difíceis de discriminar por estarem muito próximos das outras categorias, identificados por meio de um classificador de vizinhos mais próximos.

O *balanceamento* das categorias pode, no entanto, produzir efeitos indesejáveis na construção do modelo, como o *sobreajuste* [6], que será explicado adiante.

2.2 Mineração de Dados

Após o pré-processamento, segue-se a Mineração de Dados [6]. Problemas de Mineração de Dados podem ser grosseiramente classificados como supervisionados ou não supervisionados. Em um problema supervisionado, espera-se predizer um valor objetivo baseado em valores preditores. Na MD não supervisionada, por outro lado, não há um valor objetivo e espera-se descrever associações e padrões a partir dos dados de entrada [14].

Duas abordagens supervisionadas a serem consideradas para o problema de predição de renda são a **Regressão** e a **Classificação**. A Regressão busca encontrar um valor numérico contínuo para uma variável alvo, usando para isso uma ou mais variáveis preditoras [15]. A Classificação, por sua vez, identifica um valor alvo categórico a partir de variáveis preditoras [6].

Há vários algoritmos de MD que podem ser usados tanto para Regressão quanto para Classificação. A Regressão Linear, por exemplo, pode ser generalizada para lidar com outros tipos de variáveis alvo [16]. Se a variável alvo é binária (ou binária), o modelo generalizado para ela chama-se *Logistic Regression* (LR). Esse modelo é definido pela Equação 2.5 [11].

$$p(X) = \frac{e^{\beta_0+\beta_1\cdot X}}{1+e^{\beta_0+\beta_1\cdot X}}, \tag{2.5}$$

onde:

$p(X)$: a probabilidade de que o evento X ocorra;

β_i: argumentos da função de regressão que explica o evento.

Para encontrar os argumentos β_i do modelo descrito na Equação 2.5, usa-se o método de *máxima verossimilhança* [11], obtendo assim a probabilidade de que o evento observado ocorra.

Algoritmos de MD geralmente permitem customizações na sua execução a partir da modificação de parâmetros [17]. A *Logistic Regression* trabalha essencialmente com dois parâmetros: α, que controla a distribuição entre as regularizações LASSO e *Ridge* e λ que controla o quanto de regularização é aplicada, sendo que $\lambda = 0$ corresponde a um modelo sem regularizações, denominado *Ordinary Least Squares* (OLS) [18]. A relação entre esses dois parâmetros é detalhada na Tabela 2.1.

Tabela 2.1: Controle de regularização na *Logistic Regression*.

λ	α	**Resultado**
0	[0, 1]	Sem regularização. α é ignorado.
(0, 1]	0	Regressão *Ridge*
(0, 1]	1	LASSO
(0, 1]	[0, 1]	Penalização *Elastic Net*

Random Forest (RF), um algoritmo desenvolvido por Breiman [19], utiliza-se de uma técnica distinta a da *Logistic Regression*. *Random Forest* é um algoritmo do tipo *ensemble* (ou composto) que se utiliza de vários outros para gerar melhores resultados [17]. Nesse caso, RF cria múltiplas árvores de decisão que efetuam previsões sobre um evento [6]. O resultado final da previsão é obtido apurando-se os resultados individuais das árvores e optando-se pela previsão da maioria [14].

Parâmetros fundamentais para uma RF são o número de árvores usadas (*ntrees*) e o comprimento máximo de cada árvore (*max_depth*). Parâmetros outros podem ser utilizados com o intuito de generalizar o modelo, reduzindo as informações disponíveis para cada árvore ao omitir registros ou ainda variáveis explicativas [17]. São alguns exemplos de parâmetros que podem ser aplicados em um algoritmo RF [20]:

mtries: número de variáveis que serão aleatoriamente disponibilizados para a formação de cada novo nível em uma árvore;

sample_rate: proporção de registros que cada árvore terá acesso para ser construída;

col_sample_rate_per_tree: proporção de variáveis que são disponibilizadas para a construção de cada árvore;

min_rows: quantidade mínima de registros em um ramo da árvore de decisão para que ocorra divisão em dois ramos;

col_sample_rate_change_per_level: taxa de amostragem das variáveis usada em cada novo nó das árvores;

min_split_improvement: melhora mínima no erro quadrático para que haja ramificação de uma árvore.

Assim como RF, o *Gradient Boosting Machine* (GBM) é também um método composto que se utiliza de árvores de decisão [17]. O algoritmo combina duas técnicas: otimização baseada em gradiente e *boosting*. A primeira refere-se à adição iterativa de árvores para minimizar uma função de perda [21]. O termo *boosting* corresponde ao uso de um processo iterativo usado para mudar adaptativamente a distribuição de exemplos de treinamento de forma que os classificadores base (árvores) foquem em registros difíceis de classificar [6].

Tal como *Random Forest*, o principais parâmetros do algoritmo GBM são o número de árvores usadas e o comprimento máximo de cada árvore [17]. A diferença encontrada em sua parametrização consiste principalmente na possibilidade de definir que taxa de aprendizagem cada árvore fornecerá ao algoritmo como um todo (*learn_rate*) [21].

Inspiradas na rede de neurônios encontrada no cérebro e como ela opera, as *Artificial Neural Networks* são compostas por unidades também denominadas neurônios. As *Artificial Neural Networks* do tipo *Multi-Layer Perceptron with FeedForward*, em especial, consistem em várias camadas de neurônios artificiais interconectados em uma só direção, partindo da camada inicial até a final [22].

A passagem de informações de um neurônio para outro depende de uma função de ativação [6] que, ao receber valores de entrada de outros neurônios, gera um valor para a próxima camada [17]. A Tabela 2.2 mostra alguns exemplos de funções de ativação [22].

Tabela 2.2: Funções de Ativação de ANN utilizadas neste trabalho.

Nome	**Função**	**Imagem**
Tahn	$f(\alpha) = \frac{e^{\alpha} - e^{-\alpha}}{e^{\alpha} + e^{-\alpha}}$	[-1, 1]
Rectified Linear	$f(\alpha) = max(0, \alpha)$	$\mathbb{R}_+$
Maxout	$f(\alpha_1, \alpha_2) = max(\alpha_1, \alpha_2)$	$\mathbb{R}$

A variável α corresponde à soma dos valores pelo neurônio. Na função *Maxout*, contudo, os valores α_1 e α_2 não são somados, mas o maior valor entre os dois é selecionado [17]. Além da função de ativação, existem outros parâmetros customizáveis. São exemplos [22]:

epochs: quantidade de vezes que a base de treinamento passa em uma rede neural;

hidden: quantidade de camadas internas e quantos neurônios há em cada uma;

L_1 e L_2: regularizações *LASSO* e *Ridge*, respectivamente;

input_dropout_ratio: quantidade de neurônios da camada inicial que irão passar informação para a primeira camada oculta.

Desenvolvido por Caruana et al. [23, 24], *Hill-climbing Ensemble Selection with Bootstrap Sampling* (HCES-Bag) é um algoritmo *composto* e heterogêneo, isto é, pode ser formado por diferentes algoritmos. A ideia é que os algoritmos tem diferentes visões sobre os dados e é desta forma que se complementam [25].

Na estratégia de construção do modelo HCES-Bag, modelos individuais ou *candidatos* são reamostrados utilizando a técnica de *bootstrap aggregating* [6] ou *bagging*, gerando múltiplas combinações com reposição dos modelos candidatos. Em cada uma das amostras de modelos candidatos, é iniciada uma uma composição de t melhores modelos individuais.

São geradas então todas as combinações possíveis com $t + 1$ modelos e é selecionado o conjunto de melhor performance de uma métrica previamente definida. Essa adição de modelos se repete até que não haja melhora na métrica. Como pode se observar, nem todos os modelos candidatos são necessariamente utilizados, enquanto outros podem ser usados mais de uma vez [24]. A previsão do HCES-Bag é feita utilizando a média dos modelos compostos criados [25].

Como explicado nos parágrafos anteriores, algoritmos de MD podem receber parâmetros de ajuste. Não se sabendo previamente quais valores para o parâmetros irão produzir os melhores modelos a partir dos dados de treinamento, faz-se necessário o teste de uma gama de valores para esses parâmetros. Tal atividade é onerosa e maçante, dificultando a criação de modelos de alta performance. Para reduzir esse problema, foi criada uma técnica denominada *Grid Search*, onde se pode construir vários modelos a partir da combinação iterativa de diferentes valores para cada parâmetro [17]. Muitas ferramentas de *MD* possuem a técnica de *Grid Search* disponível[1] [2] [3], o que torna mais simples sua implementação.

2.3 Pós-processamento

Essa etapa pode envolver tanto a representação gráfica dos modelos quanto a análise numérica dos resultados. Gráficos permitem que os analistas explorem os dados e os resultados da MD a partir de uma variedade de pontos de vista. A análise numérica, por meio de medidas estatísticas ou testes de hipóteses, pode ser usada para eliminar modelos de MD que produzem resultados inúteis [6].

Um dos problemas, por vezes encontrado na MD, é o *sobreajuste* do modelo, isto é, quando o modelo criado é particularmente suscetível ao ruído na base de treinamento [11]. O sobreajuste torna o modelo preditivo menos capaz de ser usado de forma generalizada e pode ser observado como uma queda brusca do poder preditivo nas etapas de validação frente aos resultados de treinamento [6].

Com o intuito de identificar mais facilmente um possível *sobreajuste* dos modelos preditivos criados, pode ser utilizada a técnica de *validação cruzada* [11] ou *K-Fold Cross-Validation*. Essa técnica corresponde a separar aleatoriamente os dados de treinamento em K partes iguais e, iterativamente, treinar o modelo com um conjunto de $K - 1$ partes, validando-o com a parte restante [11]. A métrica utilizada para validar o modelo, *F-Measure*, por exemplo, é calculada a cada iteração e ao final a soma das predições é

[1] `http://scikit-learn.org/stable/modules/grid_search.html`
[2] `http://docs.h2o.ai/h2o/latest-stable/h2o-docs/grid-search.html`
[3] `https://spark.apache.org/docs/2.2.0/ml-tuning.html`

utilizada para definir a métrica final [6]. A Figura 2.1 ilustra a execução de uma validação cruzada que divide a base de treinamento em quatro partes.

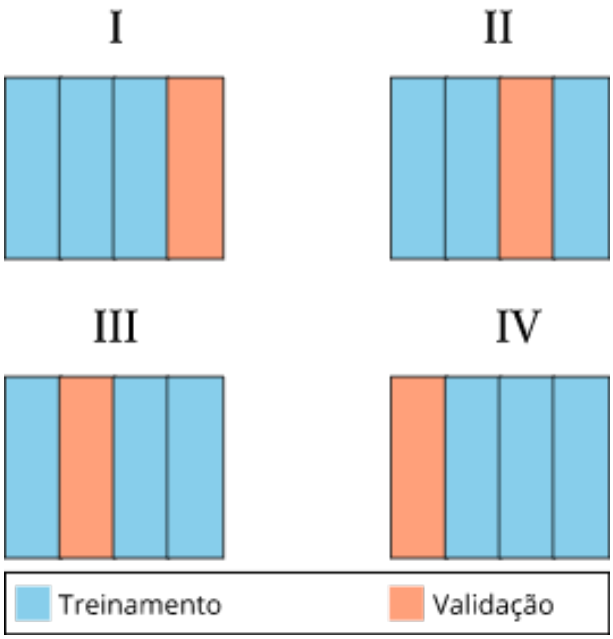


Figura 2.1: Exemplo de validação cruzada. A base de treinamento é divida em partes iguais. As etapas I a IV mostram a mudança da composição das bases de treinamento e validação.

Uma alternativa à validação cruzada é a *validação comum*, que consiste em separar uma parte da base para validação e executar separadamente as etapas de treinamento e validação [14]. Contudo, as ferramentas de MD costumam ter integradas a elas a validação cruzada[4] [5] [6], bastando informar como parâmetro a quantidade K de partes que devem ser utilizadas. Devido a isso e à maior robustez da validação cruzada em comparação à validação comum [11], torna-se mais interessante a utilização da primeira.

Uma matriz de confusão é uma outra maneira conveniente de avaliar um modelo de classificação [11]. Consiste em uma tabela que contém a quantidade de elementos que foram classificados correta e erroneamente como positivos e negativos. A Tabela 2.3 demonstra a estrutura de uma matriz de confusão.

Onde:

TN: verdadeiros negativos ou, particularmente, a quantidade de registros *corretamente* classificados como falsos;

FN: falsos negativos, ou número de registros *erroneamente* classificados como falsos;

[4] `http://scikit-learn.org/stable/modules/cross_validation.html`
[5] `http://docs.h2o.ai/h2o/latest-stable/h2o-docs/cross-validation.html`
[6] `https://spark.apache.org/docs/2.2.0/ml-tuning.html`

Tabela 2.3: Exemplo de matriz de confusão

		predito	
		Falso	**Verdadeiro**
real	**Falso**	TN	FP
	Verdadeiro	FN	TP

FP: falsos positivos, ou número de registros que foram classificados *erroneamente* como verdadeiros;

TP: verdadeiros positivos, ou a quantidade de registros que foram classificados *corretamente* como verdadeiros.

Outra ferramenta de análise no pós-processamento comumente utilizada é a curva ROC, pois resume o equilíbrio entre *True Positive Rate* (TPR), ou taxa de verdadeiros positivos, e *False Positive Rate* (FPR), ou taxa de falsos positivos. Cada ponto na curva representa um ponto de corte das probabilidades encontradas por um modelo preditivo [14]. A TPR e a FPR são dadas, respectivamente, pelas Equações 2.6 e 2.7.

$$TPR = \frac{TP}{TP + FN}. \tag{2.6}$$

$$FPR = \frac{FP}{FP + TN}. \tag{2.7}$$

A área abaixo da curva ROC, ou AUC, fornece uma abordagem para avaliar qual modelo é melhor em média. Se o modelo é perfeito, então a AUC é igual a 1. Se o modelo simplesmente age de forma aleatória, então a área sobre a curva será igual a 0,5. Um modelo é estritamente melhor que outro se ele possuir uma maior AUC [6].

Além de identificar o modelo preditivo com maior AUC, é possível identificar o ponto de corte das probabilidades encontradas para as observações que melhor atinge uma métrica de performance [6]. Dentre as métricas existentes, uma das mais populares é a Acurácia, descrita na Equação 2.8.

$$Acurácia = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.8}$$

Contudo, a Acurácia pode não ser apropriada ao avaliar modelos onde a variável alvo é muito desbalanceda, como em casos que a categoria Positiva é considerada *rara* (<10%) [6]. A *F-Measure* ou Medida F, definida pela Equação 2.9, é uma alternativa nesses casos, pois em sua formulação não leva em conta o número de Verdadeiros Negativos.

$$F - Measure = \frac{2 \cdot TP}{2 \cdot TP + FP + FN} \tag{2.9}$$

Assim, como explicado anteriormente, a avaliação de modelos de MD utilizando a AUC permite identificar o modelo que em média acerta mais vezes. Conjuntamente, a utilização de uma métrica como a Medida F auxilia a fixação de um valor de corte ótimo para as probabilidades determinadas pelo melhor modelo de MD.

Capítulo 3

Estado da Arte

Este capítulo descreve algumas pesquisas científicas que possuem ligação com o presente trabalho e analisa as possíveis contribuições desses estudos na construção da solução para o problema apresentado.

3.1 Trabalhos Relacionados

González et al. [26] monitoraram a trajetória do *smartphone* de 100 mil indivíduos, durante um período de seis meses. Eles descobriram que a trajetória humana não pode ser predita adequadamente utilizando modelos com variância infinita, como Vôos de Lévy, mas que, ao contrário, a trajetória humana possui alto grau de regularidade temporal e espacial, com cada indivíduo podendo ser caracterizado por uma distância de viagem limítrofe e uma grande probabilidade de retornar a poucos locais muito frequentados. Apesar de não tratar propriamente da relação entre renda e geolocalização, o trabalho de González et al. [26] aponta que dados posicionais de *smartphones*, em geral, podem ser boas variáveis preditoras.

Um dos exemplos de utilização de dados de localização para predizer renda é a pesquisa de Frias-Martinez e Virseda [27]. Nele os autores apresentam um modelo preditivo para dados sócio-econômicos (renda e outros) de usuários de telefone celulares em um país da América Latina. Os autores possuíam acesso aos dados das ligações, a geolocalização da torre de celular utilizada para cada ligação, gênero, idade e endereço residencial de cada cliente. Foi construído então um modelo de regressão linear multivariada, utilizando o *Ordinary Least Squares* (OLS) para cada uma das variáveis censitárias. A combinação dos três tipos de variáveis de telefonia resultou em um modelo preditor de faixa de renda com $R^2 = 0,83$, o que é consideravelmente alto, se comparada à estratégia de predição de renda do Banco Alfa.

Li et al. [28], por sua vez, usaram a relação entre preços de apartamentos e a renda de clientes de um site de produtos farmacêuticos para definir um modelo de segmentação para os clientes dessa empresa. Para isso, cruzaram os endereços de entrega dos produtos farmacêuticos com os valores de casas à venda em um site de oferta de imóveis. Para criar o modelo de segmentação, os autores utilizaram Mineração de Dados não supervisionada. O cruzamento dos endereços com os imóveis à venda foi realizado utilizando um algoritmo de similaridade de texto. No caso de não se encontrar endereços com a similaridade mínima definida pelos autores, outro algoritmo era usado: calculava-se a média e o desvio-padrão de casas à venda próximas ao endereço de entrega, aumentando o raio de pesquisa até que o desvio-padrão encontrasse um valor limite.

Lessmann et al. [25] efetuaram um *benchmarking* com 41 algoritmos de classificação para definição de *default*[1] em 8 diferentes bases de dados. Dentre os algoritmos de classificação, foram avaliados algoritmos individuais, técnicas de *ensemble* homogêneo e heterogêneo. Dentre todos os algoritmos testados, três se destacaram ao obter acurácia melhor que a Regressão Logística, técnica mais utilizada no mercado: *Hill-climbing Ensemble Selection with Bootstrap Sampling* (HCES-Bag), *Random Forest* (RF) e *Artificial Neural Networks* (ANN), respectivamente. Apesar do algoritmo HCES-Bag ter sido o melhor em acurácia, os autores verificaram que ANN e RF, respectivamente, obtiveram um menor número de Falsos Negativos, isto é, deixavam passar menos casos de *default*.

Bjorkegren e Grissen [29] criaram um modelo preditivo de risco de crédito tendo como base dados de pessoas que fizeram empréstimos em uma empresa de microcrédito. Foram utilizados dados de ligações telefônicas, da localização das torres e modelos dos *smartphones*. Em contraste, um conjunto de dados contendo idade, sexo, valor do empréstimo e tempo para quitação do empréstimo foi utilizado para comparar a acurácia do primeiro conjunto de dados. Foram utilizados os algoritmos RF e OLS para criação do modelo de Regressão, onde os modelos utilizando dados de telefone obtiveram *Area Under the Curve* (AUC) superior (0,66-0,67) aos mesmos algoritmos utilizando os dados demográficos e relacionados ao empréstimo (0,53).

Toole et al. [30] estudaram a relação entre dados de ligações telefônicas e desemprego. Eles construíram um classificador Bayesiano, baseado em mudanças de ritmo de uso de telefones celulares. Além disso, um resultado da pesquisa foi a identificação de uma relação entre o desemprego e a diminuição da mobilidade e do comportamento social dos indivíduos.

Kim et al. [31] investigaram a correlação entre o uso de *smartphones* e características demográficas de cerca de dez mil respondentes de um questionário na Córeia. Neste

[1] "Atraso ou não pagamento de título ou cupom na data de seu vencimento. Declaração de insolvência do devedor, decretada pelos credores quando as dívidas não são pagas nos prazos estabelecidos". `http://www.bcb.gov.br`, acessado em 03/06/2018.

questionário, além de existirem perguntas relacionadas a dados demográficos (idade, sexo, grau de instrução e renda), havia questões voltadas a quais aplicativos as pessoas utilizam (*e-commerce*, entretenimento, leitura, notícias e redes sociais). Foi identificada uma correlação de 0,54 entre o uso de smartphones e o grau de instrução. A renda e o uso de *smartphones* mostrou uma correlação baixa, de 0,17. Entre o uso de tipos de aplicativos, os softwares relacionados a notícias obtiveram maior correlação com a renda dos indivíduos (0,23).

Blumenstock et al. [32] usaram dados de *smartphones* para identificar características demográficas de indivíduos de Ruanda. Para definir um índice de riqueza, foi efetuada uma pesquisa a 856 cidadãos desse país, perguntando sobre posse de bens, características de suas moradias e outros indicadores de bem-estar social. A correlação entre o índice de riqueza e os dados de ligações telefônicas e mensagens de texto foi de 0,68. Ao separar o estudo em microrregiões, Blumenstock et al. [32] mostraram que os dados de telefonia tem forte correlação com os indicadores de riqueza de cada distrito de Ruanda (0,91).

Sundsøy et al. [33] utilizaram algoritmos de classificação para predizer a renda de 80 mil clientes de uma empresa de telefonia, usando tanto dados cadastrais quanto transacionais. Foram selecionados cerca de 150 variáveis preditoras da base da operadora, como uso de mensagens (SMS), torres de celular utilizadas nas ligações, comportamento de recarga de créditos, frequência de uso de redes sociais, tipo de aparelho entre outros. Foram utilizadas 3 técnicas de classificação: *Random Forest* (RF), *Gradient Boosting Machine* (GBM) e *Artificial Neural Networks* (ANN). Os pesquisadores explicaram que o ANN alcançou acurácia satisfatória (77%), usando somente variáveis associadas ao uso das torres, enquanto que os outros métodos precisaram de todos os outros atributos para alcançarem um resultado próximo (68-72%) ao de ANN.

Kibekbaev e Duman [34] se destacam no teste de diversos algoritmos de Regressão para identificar a renda de clientes. Eles testaram um total de 10 algoritmos: OLS, *Beta Regression*, *Beta-OLS*, *Box-Cox OLS*, *Ridge Regression*, *Robust Regression*, CART, M5P, MARS, ANN. Foram incluídas ainda técnicas combinadas, como OLS + M5P, OLS + MARS, OLS + CART, OLS + ANN, OLS + LSSVM e técnicas de *Ensemble Learning*, como *Random Forest* e *AdaBoost*. As informações utilizadas para testar os modelos consistiam em uma média de registros de 10.000 clientes em 5 bancos turcos. O novo modelo seria usado na análise de concessão de crédito e deveria minimizar a negação de empréstimo para clientes que poderiam recebê-lo e evitar que fossem concedidos valores acima do limite do cliente. O modelo com melhor resultado era formado por uma combinação do *Ordinary Least Squares* com o algoritmo de Árvore de Decisão M5, obtendo R^2 entre 0,38 e 0,59.

Steele et al. [35] utilizaram Regressão por meio de modelos lineares generalizados em

dados de mobilidade e em características de ligações telefônicas para predizer índices de pobreza em Bangladesh. O experimento foi considerado bem-sucedido, obtendo $R^2 = 0,78$ para as regiões urbanas e $R^2 = 0,66$ para as áreas rurais. A mesma abordagem, ao utilizar os dados em conjunto, alcançou um $R^2 = 0,76$.

3.2 Considerações

Em relação ao tipo de Mineração de Dados supervisionada, os trabalhos [27, 34, 35] utilizaram algoritmos de Regressão, enquanto que os trabalhos [25, 29, 30, 33] optaram por algoritmos de Classificação. Algoritmos de Regressão foram testados em maior proporção o *Ordinary Least Squares* [27, 29, 34] e *Random Forest* [29, 34]. Dentre os algoritmos de Classificação, foram testados em maior quantidade *Random Forest* [25, 33] e *Artificial Neural Networks* [25, 33]. Para o problema de determinar a renda de correntistas digitais, optou-se pela utilização de algoritmos de Classificação, pois não há a necessidade de identificar um valor contínuo da renda, sendo suficiente para a estratégia de oferta de produtos do Banco Alfa tão somente a identificação de faixas de renda. De fato, há trabalhos de Classificação suficientes que serviram de referencial para a pesquisa e que forneceram uma lista de cinco algoritmos para comparação: (ANN, GBM, HCES-Bag, RF e LR).

Uma dificuldade encontrada na maioria dos artigos pesquisados sobre predição de renda consistia em obter rendas individuais para construir a variável alvo [27, 28, 32, 33, 35], devido a questões de sigilo da empresa ou pela própria falta do dado. Os pesquisadores então costumavam usar variáveis correlacionadas como subterfúgio da falta de informação. À exceção das pesquisas [25, 30, 31, 34], foram usados, em algum grau, dados de localização dos clientes como indicadores de renda. Os trabalhos [27, 28, 31–33, 35] utilizaram dados censitários ou questionários para obter tais indicadores.

Apesar desta pesquisa sofrer da mesma ausência de informação de renda para os clientes digitais, optou-se por um caminho distinto que, conforme descrito no Capítulo 1, corresponde a utilizar uma população similar para criar o modelo de MD. Mesmo assim, algumas das variáveis correlatas à renda descritas nesses artigos serviram de inspiração na construção das variáveis explicativas. O uso de dados censitários e demográficos oriundos do Instituto Brasileiro de Geografia e Estatística, aliados a informações sobre endividamento obtidas do Banco Central do Brasil, por exemplo, foram utilizados na construção de variáveis preditoras do modelo. Seguindo essa mesma linha, as pesquisas [29, 33], por exemplo, utilizaram o modelo do *smartphone* como variável preditora. O artigo de Li et al. [28], por sua vez, usou valores de venda de imóveis da região. Tais variáveis também foram levadas em consideração durante a construção do modelo.

Capítulo 4

Plano de Trabalho

Uma abordagem para minerar dados comum na indústria é o *Cross Industry Standard Process for Data Mining* (CRISP-DM), um processo iterativo que consiste em 6 etapas, descritas a seguir [36]:

1. **Entendimento do Negócio**: importa-se com o entendimento dos objetivos e requisitos a partir da perpectiva negocial, para então converter este conhecimento na definição de um problema de Mineração de Dados e em um plano preliminar para atingir os objetivos.
2. **Entendimento dos Dados**: começa com uma coleção de dados inicial e prossegue com atividades que tem como intuito tornar os dados familiares, identificar problemas de qualidade e descobrir os primeiros *insights* nos dados e detectar subconjuntos para formular hipóteses sobre a informação escondida.
3. **Preparação dos Dados**: cobre todas as atividades relacionadas à construção do conjunto de dados final, ou seja, o conjunto de dados que será utilizado pelo modelo.
4. **Modelagem**: nessa etapa, vários modelos são escolhidos, aplicados e calibrados em busca do melhor resultado.
5. **Avaliação**: com um ou mais modelos de alta qualidade construídos, faz-se necessário revisitar na etapa de Entendimento do Negócio os pré-requisitos de negócio e verificar se os modelos os atendem corretamente.
6. **Implantação**: a última fase consiste na entrega do modelo no ambiente de produção.

Como se pode observar na Figura 4.1, o CRISP-DM é um processo onde os resultados obtidos em uma etapa podem tornar necessária uma revisão de etapas anteriores. Adicionalmente, o processo pode ser executado repetidas vezes, até que os objetivos de négocio tenham sido atingidos ou se tenham esgotados os recursos disponíveis para a pesquisa.

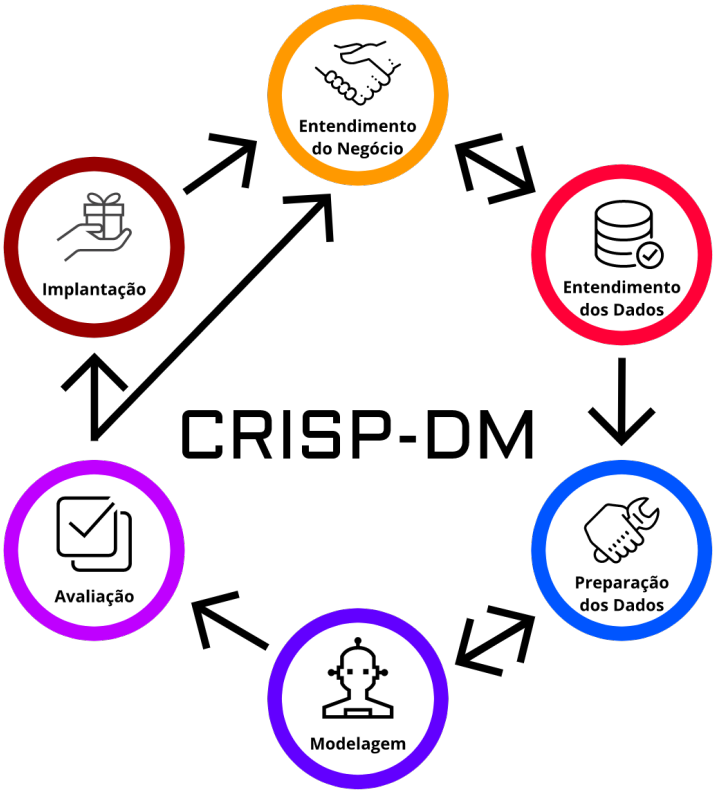


Figura 4.1: Processo Padrão Inter-Indústrias para Mineração de Dados.

Este trabalho foi executado e organizado conforme as etapas propostas pelo CRISP-DM, tendo sido executadas duas iterações do processo. Visando um melhor entendimento da pesquisa, foi criado um capítulo para iteração. O código-fonte e os dados utilizados para a criação dos modelos foram armazenados no GitHub[1]. Por questões de sigilo, os dados e os nomes das variáveis foram anonimizados.

[1]Disponível em `https://github.com/rnmourao/renda_digitais`.

Capítulo 5

Primeira Iteração

Este Capítulo detalha os passos executados para a investigação e a criação de um modelo de Mineração de Dados como possível solução do problema proposto. A organização do Capítulo segue as etapas do CRISP-DM.

5.1 Entendimento do Negócio

Muito do que corresponde a essa etapa foi apresentado no Capítulo 1. O objetivo de negócio é identificar a renda de clientes do Banco Alfa que possuem contas-correntes digitais, de forma que se possa oferecer aos clientes com maior renda a opção de transformar suas contas digitais em contas convencionais. A seleção de uma clientela interessada permite que a abordagem seja mais eficiente, reduzindo tempo e custo e ainda maximizando o retorno [5].

Entende-se que clientes com renda superior adquirem mais produtos. Logo, aumentam sua margem de contribuição mais que outros clientes e que, por isso, devem ser abordados primeiro. Presume-se, por exemplo, que a quantidade de produtos adquiridos seja diretamente relacionada à renda dos clientes. A Tabela 5.1 indica uma relação direta entre a média de produtos adquiridos e as progressivas faixas de renda.

Tabela 5.1: Quantidade de Produtos por Faixa de Renda

Faixa de Renda (R$)	**Média de Produtos**
[0,01, 1.000)	3
[1.000, 4.000)	4
[4.000, 10.000)	6
[10.000, $+\infty$)	7

Percebe-se também uma maior correlação entre a renda e a margem de contribuição em valores acima de R$ 10.000 em comparação às faixas de renda inferiores, conforme mostrado na Tabela 5.2.

Tabela 5.2: Correlação entre Renda e Margem de Contribuição

Faixa de Renda (R$)	**Correlação de Pearson**
[0,01, 1.000)	0,006
[1.000, 4.000)	0,014
[4.000, 10.000)	0,078
[10.000, +Inf)	0,375

Sendo assim, a seleção do público alvo para oferta de alteração de conta deveria levar em consideração uma faixa de renda superior. Essa abordagem pode restringir bastante o público selecionado, pois o percentual de clientes convencionais que recebem mensalmente um salário superior a R$ 10 mil reais, por exemplo, corresponde a cerca de 10% do total.

Para tentar encontrar público semelhante no grupo de clientes digitais, o Banco Alfa se vale de uma base de dados externa, a qual possui a renda presumida não só do público-alvo, mas de cerca de metade da população brasileira.

Conforme apresentado na Figura 1.1 e na Tabela 1.1, essa fonte de informação apresenta rendas aquém do esperado, quando comparadas às rendas comprovadas na base de dados do Banco Alfa. Tal situação prejudica a oferta de evolução de conta via *telemarketing*, afetando assim a correta priorização de clientes e podendo causar diminuição nas vendas.

Propôs-se então, como alternativa, a criação de um modelo de mineração de dados para previsão de renda dos clientes digitais do Banco Alfa. O principal empecilho para a formulação de tal modelo consiste na ausência da informação alvo, ou seja, a renda.

Para suplantar essa dificuldade, decidiu-se por utilizar como base para treinamento clientes semelhantes aos digitais que possuíssem as mesmas informações e, adicionalmente, a renda comprovada. Um modelo criado a partir de uma amostra conveniente de clientes poderia ser usado na previsão de renda dos clientes digitais.

Desta forma, o trabalho de Mineração de Dados consiste em definir um modelo preditivo de faixas de renda para os clientes com contas-correntes digitais a partir de clientes que possuem renda comprovada. É conveniente testar diversos algoritmos de MD para que se selecione o melhor modelo. O classificador com melhores AUC e *F-Measure*, respectivamente, será comparado com a base de dados externa, a fim de verificar se o primeiro supera a última.

5.2 Entendimento dos Dados

As informações utilizadas nesta primeira iteração estavam disponíveis em bancos de dados e foram extraídas de três origens:

1. um Banco de Dados *data warehouse* com diversas informações dos clientes e utilizado pelo Banco Alfa para análises de mercado;
2. uma lista com a identificação de todos os clientes digitais;
3. uma base de dados com a fonte de informação externa sobre renda.

Além de possuir quase todas as variáveis explicativas utilizadas na primeira iteração desse estudo, a primeira base continha uma variável que permitiu identificar os clientes convencionais que utilizavam o aplicativo móvel do Banco Alfa.

A segunda base foi utilizada para remover da primeira todos os clientes digitais. Assim, a base de estudo foi composta somente por clientes convencionais que usavam o aplicativo móvel.

A terceira base de dados era organizada por CPF, empregador e data da informação e ainda continha a informação de renda presumida.

As bases 1 e 3 foram acessadas utilizando *Structured Query Language* (SQL). As informações obtidas no *data warehouse* já se encontravam *desnormalizadas*, o que simplificou bastante a extração dos dados. A segunda base possuía também todas as informações pertinentes em uma só tabela. Além disso, por ser de um volume considerável, foram recuperados da base 3 somente os dados de usuários do aplicativo móvel.

Os dados foram extraídos para arquivos no formato *Comma-Separated Values* (CSV), por ser mais adequado para manipulação com ferramentas de Mineração de Dados. A base 2, contendo a lista de clientes digitais, já se encontrava em formato CSV, não tendo sido necessária qualquer mudança de formatação.

A Tabela 5.3 descreve o volume de dados disponíveis nas bases de dados.

Tabela 5.3: Volume das bases utilizadas.

Base	**Colunas**	**Registros**
Data Warehouse	116	10 milhões
Clientes Digitais	1	1 milhão
Fonte Externa	4	36 milhões

Os dados disponíveis na base de clientes convencionais eram constituídos de informações sócio-demográficas, canais de comunicação utilizados com o banco (aplicativo móvel, ligação telefônica, atendimento presencial etc.), posse de produtos, datas relacionadas

a contatos e aquisição de produtos, detalhes sobre movimentação financeira, métricas de retorno financeiro entre outras. Muitos desses atributos não foram utilizados para o trabalho, devido à ausência de informação equivalente para os clientes digitais.

Após a extração dos dados, tornou-se possível comparar os clientes digitais com os clientes convencionais que usavam o aplicativo móvel, os quais de agora em diante serão referidos somente como *clientes convencionais*.

Os clientes convencionais eram um grupo que possuía certa semelhança com os correntistas digitais, devido ao uso da mesma interface que os últimos para efetuar suas transações bancárias. Os primeiros, por sinal, superavam em número os últimos, como se pode observar na Figura 5.1.

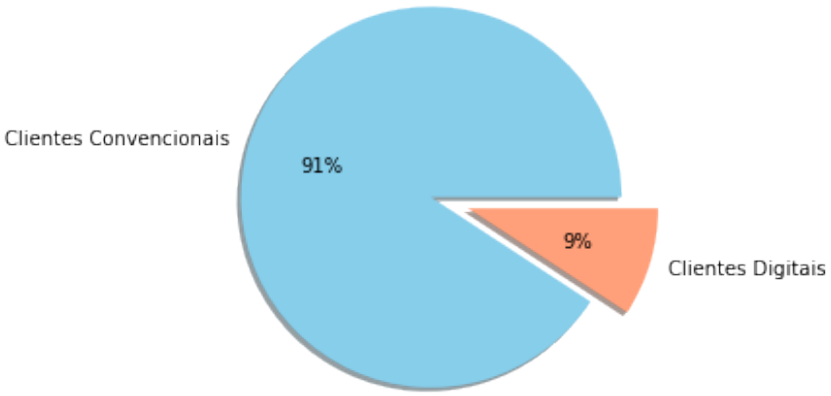
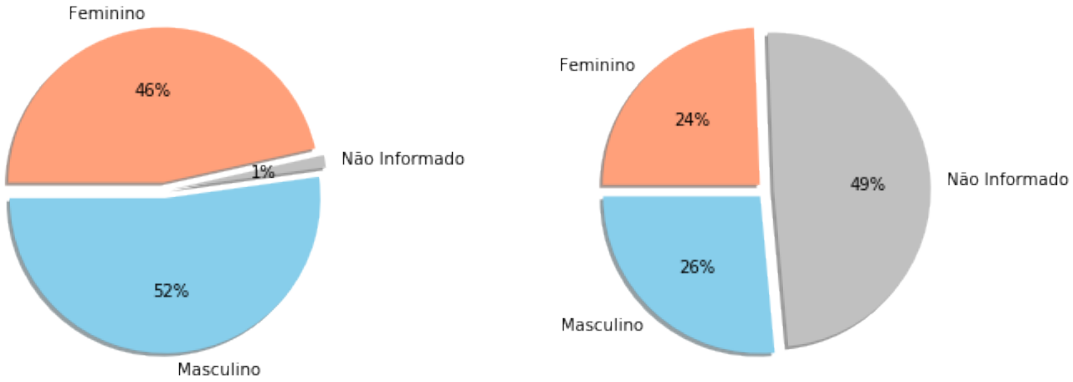


Figura 5.1: Proporção entre clientes convencionais e clientes digitais que utilizam o aplicativo móvel.

Certas diferenças entre os dois grupos de clientes eram esperadas, como os dados sobre gênero, explicitada na Figura 5.2, que mostra o grande número de clientes digitais sem essa informação.

Outra diferença estava na distribuição de idades desses dois públicos. Na Figura 5.3 observa-se um maior percentual de clientes digitais na faixa de 20 anos em comparação aos clientes convencionais. Dentre as prováveis causas, presume-se que a facilidade na abertura da conta digital pode ter motivado a adoção desse produto por parte do público mais jovem. Por outro lado, faixas de idades superiores podem estar menos propensas ao uso de contas digitais, por possuírem contas convencionais há mais tempo. A diferença na distribuição de idade entre esse dois públicos poderia afetar a comparação da distribuição de renda.

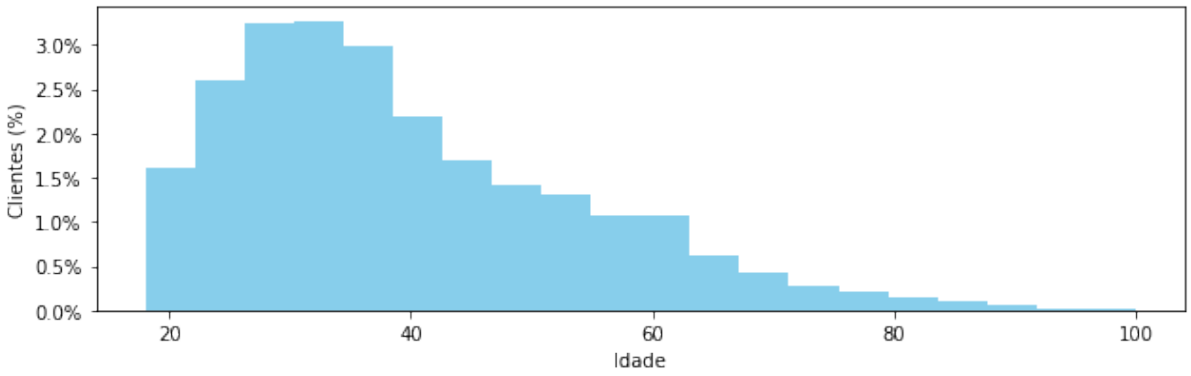
O local de residência dos clientes era outro aspecto onde se esperava encontrar diferenças. Dada novamente a facilidade de criação de uma conta digital, eximindo o público-alvo de se deslocar para uma agência bancária, havia certa expectativa por parte do Banco



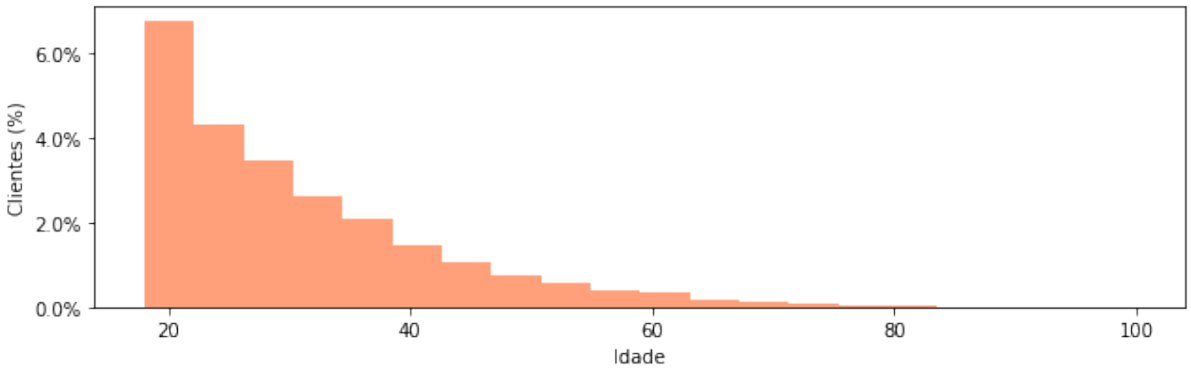
(a) Clientes Convencionais.

(b) Clientes Digitais.

Figura 5.2: Distribuição da variável “sexo” entre clientes que usam o aplicativo móvel.



(a) Clientes Convencionais.



(b) Clientes Digitais.

Figura 5.3: Distribuição de Idade dos Clientes que usam o aplicativo móvel.

Alfa de que um grande número desses novos clientes morasse em localidades distantes dos grandes centros. Contudo, como mostra a Figura 5.4, a proporção de clientes digitais nas capitais não difere da proporção de clientes convencionais tornando-os, sob esse prisma, mais semelhantes entre si.

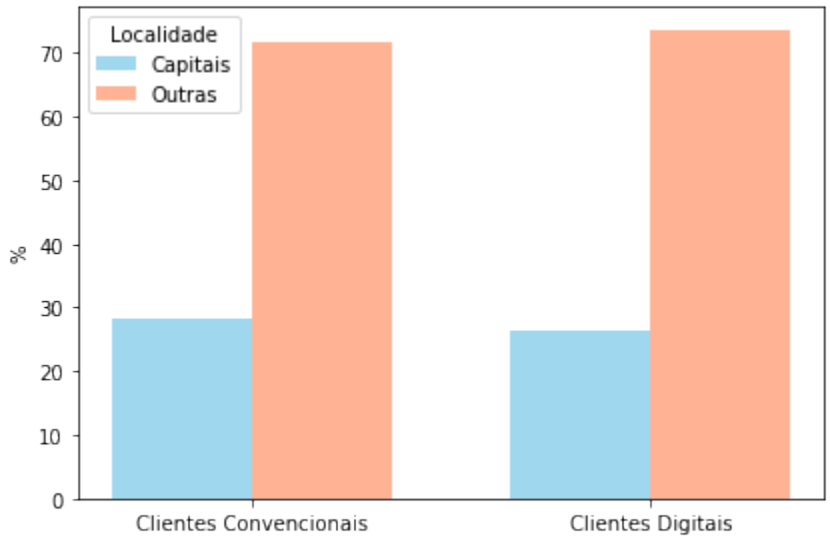


Figura 5.4: Proporção entre clientes convencionais e clientes digitais que utilizam o aplicativo móvel.

Alguns dos atributos obtidos possuíam um grande número de valores nulos para os clientes digitais, como foi exemplificado pela variável "sexo", mostrada na Figura 5.2. Casos como esse forçaram a exclusão do atributo. Em outros casos, nos atributos em que não havia uma grande quantidade de valores nulos, foi realizada imputação dos dados, de acordo com a particularidade de cada variável. Outro problema encontrado foi a existência de *outliers* em algumas variáveis. Transformações estatísticas foram utilizadas para tentar mitigar o problema. Detalhes sobre a preparação final dos dados são explicados na seção a seguir.

5.3 Preparação dos Dados

As tarefas de preparação são, costumeiramente, executadas múltiplas vezes e não possuem uma ordem prescrita. Essas tarefas incluem tabulação, seleção de atributos e registros, assim como a transformação e limpeza de dados para atender pré-requisitos das ferramentas de modelagem [36].

Como explicado na Seção 5.2, três origens de dados foram utilizadas para o estudo. A maioria das informações dos clientes convencionais e digitais encontrava-se em um *data warehouse* e foram extraídas utilizando SQL. Assim também foram os dados de renda presumida. A lista de clientes digitais foi disponibilizada por meio de arquivo de planilha eletrônica.

A extração gerou arquivos com extensão CSV e a atividade de preparação de dados ocorreu utilizando as linguagens *Bash* e *Python*. Com a linguagem *Bash* tratou-se de

retirar espaços duplicados, caracteres não-imprimíveis, acentos e tornou minúsculos todos os caracteres alfabéticos. A execução da linguagem *Python* ocorreu no editor *Jupyter Notebook*[1] e foram utilizadas diversas bibliotecas de manipulação de dados, como *Pandas*[2] e *Apache Spark*[3].

Em primeiro lugar, os clientes foram identificados como convencionais ou digitais, por meio da lista de clientes digitais.

A primeira variável tratada foi o próprio atributo alvo. Esse atributo foi construído a partir da categorização da informação de renda comprovada dos clientes convencionais. O Banco Alfa considera como clientes de alta renda aqueles que recebem um salário mensal maior ou igual a R$ 10.000. A variável alvo deveria ser, desta forma, um atributo binário, indicando se o cliente possui alta renda ou não. Para avaliar o quão bem essa bissecção separava o público-alvo como um todo, foi aplicado o coeficiente de Silhouette, descrito na Equação 2.3. A partição sugerida pelo Banco Alfa alcançou o coeficiente de Silhouette de aproximadamente 0,25.

Para efeito de comparação, aos valores de renda foi aplicado um agrupamento utilizando o algoritmo *K-Means* com $K = 2$, isto é, com dois centróides. As faixas de renda encontradas pelo algoritmo não se mostraram interessantes, como se verifica na Tabela 5.4.

Tabela 5.4: Faixas de renda encontradas pelo aplicativo *K-Means* na primeira rodada.

Faixa de Renda (R$)	**Distribuição de Clientes (%)**
(0, 2.057.679,00]	99,99
[2.057.679,00, $+\infty$)	0,01

O coeficiente de *Silhouette* encontrado (0,69) indicou melhor definição dos conjuntos de dados, mas como a quantidade de pessoas classificadas como tendo uma faixa de renda superior era muito baixa, foi aplicada uma transformação nos dados para que o formato da distribuição da variável renda não prejudicasse a definição dos grupos. Após os valores serem convertidos via Equação 2.2, o algoritmo *K-Means* foi executado novamente, com $K = 2$. Após isso, as rendas foram revertidas para os valores originais, de forma que fosse possível interpretar os dados. O coeficiente de *Silhouette* chegou a 0,57 e as faixas estão descritas na Tabela 5.5.

Desta forma, a categorização da renda dos clientes utilizando o algoritmo *K-Means* encontrou uma faixa de renda melhor distribuída segundo os critérios de coesão e separação. Ainda assim, o estudo prosseguiu com as faixas de valores recomendadas pelo Banco Alfa, por se tratar de estratégia já estabelecida para outros públicos.

[1]Disponível em `http://jupyter.org/`.
[2]Disponível em `https://pandas.pydata.org/index.html`.
[3]Disponível em `https://spark.apache.org/`.

Tabela 5.5: Faixas de renda encontradas pelo aplicativo *K-Means* após transformação Box-Cox.

Faixa de Renda (R$)	**Distribuição de Clientes (%)**
(0, 2.958,28]	59,76
[2.958,28, $+\infty$)	40,24

Após a definição da variável alvo, seguiu-se com a seleção das variáveis explicativas. Muitas variáveis foram removidas do grupo de 116 variáveis disponíveis no *data warehouse*.

A primeira variável removida foi "sexo", porque havia um grande número de valores nulos na base de clientes digitais, conforme mostrado na Figura 5.2. Outras variáveis foram retiradas porque pertenciam somente a clientes convencionais, como a variável Risco de Crédito. Essa variável é obtida por meio de uma análise de crédito, que depende da comprovação de renda do cliente. Variáveis que correspondiam à posse de produtos de crédito ou demais produtos que só poderiam ser adquiridos se o cliente possuísse renda comprovada também foram descartadas.

Outra informação que possuía um número considerável de valores nulos era a renda presumida. Os valores correspondiam a cerca de 2% da base de clientes convencionais, porém excedia 31% na base de clientes digitais. Como foi explicado nas Seções 5.1 e 5.2, essa informação é usada atualmente como preditora de renda pelo Banco Alfa e foi comparada com os modelos de Mineração de Dados gerados. Essa própria variável foi incluída nas base de treinamento para os modelos de MD.

Desta forma, ao invés de excluir a variável, foi-lhe imputada o valor do salário mínimo em 2017 [4], R$ 937,00, nos campos nulos. Dois motivos levaram à utilização do salário mínimo para imputação dos valores nulos. Primeiramente, a distribuição de renda presumida mostrada na Figura 1.1 indica uma assimetria positiva, o que aumenta o valor da média e a distancia dos valores mais comuns [37]. Em segundo lugar, políticas de salário mínimo tem como objetivo dar aos trabalhadores um padrão de vida minimamente adequado, elevando o salário para acima do ponto de equilíbrio entre a oferta e a demanda [38]. Assim, apesar de em alguns casos se observar salários abaixo dessa marca, tal política governamental traz um grande número de pessoas que ganhariam menos para o valor do salário mínimo, tornando o valor mais comum encontrado.

Algumas variáveis foram construídas a partir de datas, como tempo da conta atual, tempo da primeira conta, tempo da primeira transação, tempo da última atualização cadastral, todas contadas em dias.

Outras correspondiam à data de utilização de determinado canal com o Banco Alfa (aplicativo móvel, telefone, terminal de auto-atendimento, atendimento presencial). A

[4]Disponível em `http://www.planalto.gov.br/ccivil_03/_ato2015-2018/2016/decreto/D8948.htm`.

partir dessas variáveis foram criadas variáveis binárias que indicavam se um cliente já foi usuário de cada um desses canais.

As variáveis explicativas estão detalhadas na Tabela 5.6. Como se pode observar, a própria variável de previsão de renda adquirida pela fonte externa foi também utilizada para o estudo, a fim de verificar sua valia em conjunto com outras variáveis para construção de um modelo preditivo.

Tabela 5.6: Variáveis explicativas

Definição	**Tipo**	**Domínio**	**Quantidade**
Indicadores de posse de produtos	Binária	0, 1	17
Indicadores de uso de canais	Binária	0, 1	10
Relacionadas a tempo de posse de produtos em dias	Inteira	$[1, +\infty)$	5
Quantidade de produtos	Inteira	$[0, +\infty)$	1
Valores de endividamento no Sistema Financeiro Nacional	Decimal	$[0, +\infty)$	5
Valor em investimentos	Decimal	$[0, +\infty)$	1
Margem de contribuição	Decimal	$(-\infty, +\infty)$	1
Movimentação financeira	Decimal	$(-\infty, +\infty)$	9
Renda presumida	Decimal	$[0, +\infty)$	1
Idade	Inteira	$[18, +\infty)$	1

Todas as variáveis receberam transformação Box-Cox. Depois disso, foram padronizadas para a faixa [0, 1], a fim de que algoritmos que usam cálculos de distância não gerassem resultados anômalos, devido à diferença de amplitude entre as variáveis [6].

Após a seleção das variáveis, ainda restava tratar o número de registros utilizados. A base de clientes possui milhões de registros e fazer uso de toda a base poderia impactar o tempo de processamento de cada modelo.

Para que os modelos preditivos fossem criados com uma base que representasse a variabilidade da população, foram selecionados clientes convencionais do Banco Alfa que tinham conta na cidade de Joinville (SC), um total de 29.113 pessoas. É possível observar na Figura 5.5, que a distribuição de renda em Joinville é bem similar à distribuição de renda em todo o território nacional.

Em relação à variável alvo, pode-se perceber pela Figura 5.6 uma leve diferença entre as observações de Joinville em relação ao total de clientes do Banco Alfa, indicando a viabilidade dessa amostra para o estudo.

Na Figura 5.7 se vê que a escolha da cidade de Joinville também se manteve semelhante à população em relação à distribuição da renda presumida. Assume-se, desta feita, que

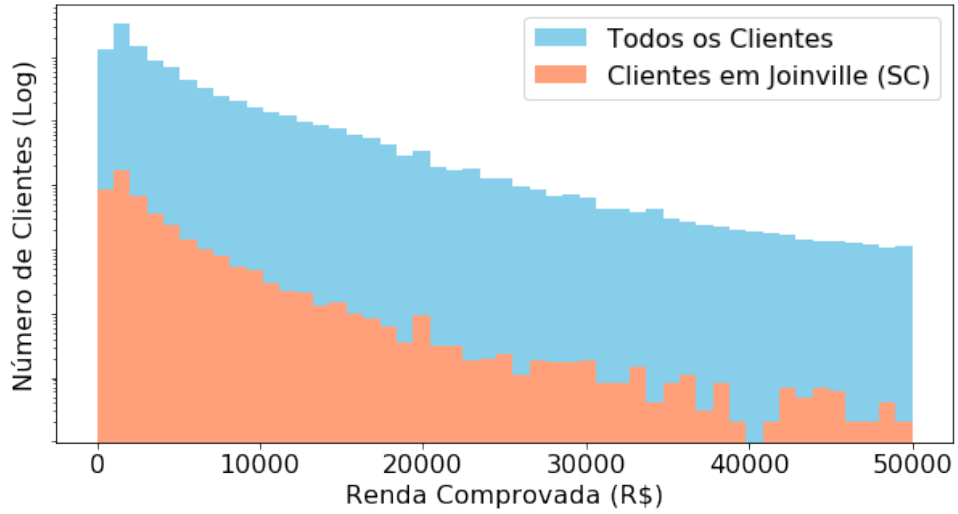


Figura 5.5: Distribuição de renda entre clientes em todo o território nacional e clientes em Joinville (SC).

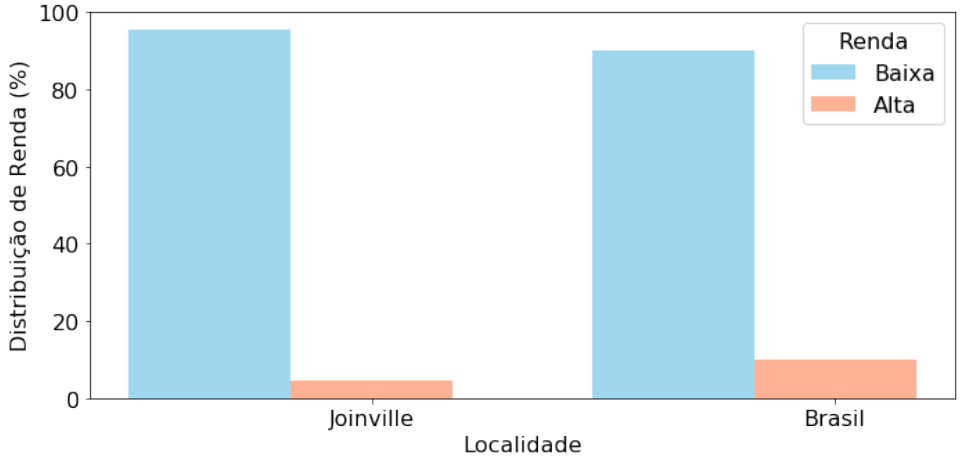


Figura 5.6: Distribuição da variável alvo entre clientes em todo o território nacional e clientes em Joinville (SC).

a amostra não afetou a imparcialidade da comparação entre a estratégia de previsão de renda atualmente utilizada pelo Banco Alfa e os modelos preditivos propostos.

Dado que o percentual de observações marcadas como de alta renda era inferior a 10%, foram testadas diversas técnicas de balanceamento. Para efetuá-las, os registros de Joinville foram separados em dois conjuntos de dados: uma base de treinamento com 16.636 registros e uma base de teste de 12.477 linhas. Somente a base de treinamento foi balanceada, mantendo assim a base de teste com as faixas de renda em suas proporções originais.

As consequentes bases de treinamento balanceadas foram testadas por meio de uma *Logistic Regression*. O valor de *F-Measure* obtido do modelo gerado a partir de cada uma

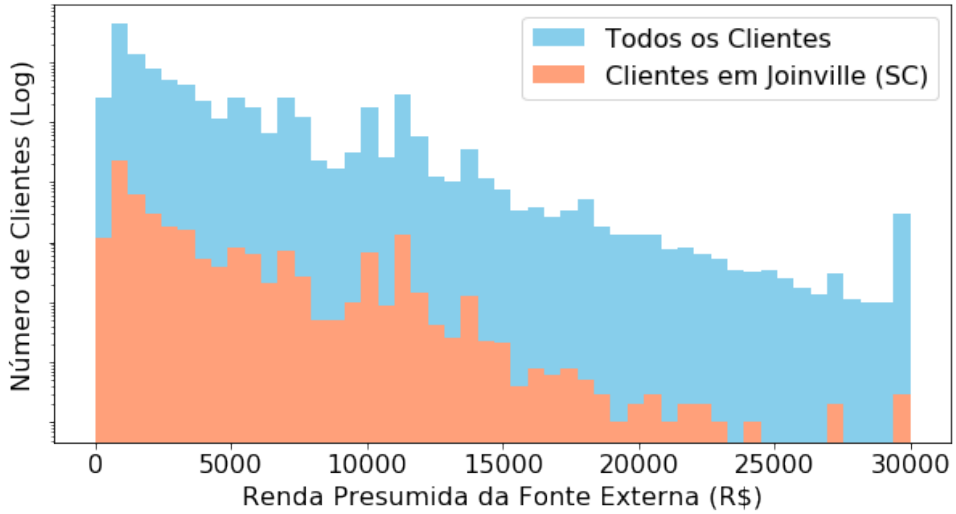


Figura 5.7: Distribuição de renda presumida da fonte externa para clientes em todo o território nacional e clientes em Joinville (SC).

dessas bases sobre a base de teste é representado na Tabela 5.7.

Tabela 5.7: Resultado do teste das técnicas de balanceamento de categorias.

Técnica	**F-Measure**
Random Oversampling	0,514
SMOTEENN	0,514
Random Undersampling	0,513
SMOTE	0,510
SMOTETomek	0,509
ADASYN	0,501
Cluster Centroids	0,415

Apesar de todas as técnicas terem obtido *F-Measure* próximo, *Random Oversampling* e SMOTEENN se destacaram. Para efetuar a etapa de modelagem foi escolhido o balanceamento *Random Oversampling*, devido o princípio da parcimônia [6], pois SMOTEENN possui um maior número de passos para execução e, consequentemente, mais tempo de processamento. Assim, a base de treinamento balanceada ficou com 31.664 registros.

Para efetuar a comparação entre os modelos preditivos e a atual estratégia de previsão de renda do Banco Alfa, foi criado um terceiro arquivo a partir da base teste, mantendo uma variável binária que representava se a renda presumida do cliente era inferior ou superior a R$ 10.000.

Ademais, a fim de executar o algoritmo HCES-Bag foi necessário criar arquivos no formato LIBSVM.

5.4 Modelagem

Esta etapa descreve como os modelos preditivos foram construídos e testados. Seis modelos preditivos foram criados nesta iteração, sendo um modelo correspondendo à abordagem adotada no presente pelo Banco Alfa e cinco modelos criados a partir de algoritmos de MD.

O primeiro modelo preditivo foi definido como linha de base para comparação com os demais. Construído a partir da renda presumida adquirida da fonte externa, caracteriza-se por uma variável explicativa binária, indicando se o cliente tem renda presumida superior a 10 mil reais. Tal variável foi avaliada usando a base de teste criada na etapa 5.3. Como se tratava de uma aplicação direta de um modelo preditivo já criado, não houve utilização da base de treinamento e, consequentemente, não foi realizada validação cruzada. De igual maneira, não houve utilização de *Grid Search*, pois não havia parâmetros a serem definidos.

Cinco algoritmos de Mineração de Dados foram usados para criar modelos preditivos a partir dos dados do Banco Alfa: *Logistic Regression* (LR), *Artificial Neural Networks* (ANN), *Random Forest* (RF), *Gradient Boosting Machine* (GBM) e *Hill-climbing Ensemble Selection with Bootstrap Sampling* (HCES-Bag). Enquanto os quatro primeiros algoritmos foram executados usando a plataforma H2O[5], o HCES-Bag foi executado usando a implementação de Lambert[6]. A parametrização dos primeiros quatro algoritmos está descrita na Tabela 5.8.

No caso da *Logistic Regression*, o *framework* H2O permite uma busca automática pelo valor ótimo do parâmetro λ. Os demais valores utilizados no *Grid Search* dos algoritmos correspondiam ou a valores padrão das ferramentas utilizadas, ou a valores limítrofes [17]. O *Grid Search* desses algoritmos foi executado algumas vezes com o intuito de ajustar os parâmetros que seriam usados. Esse refinamento levou também em consideração o quanto a modificação dos valores afetava o resultado do modelo. Em parâmetros que afetavam o tempo de criação do modelo, se dois valores alcançavam o mesmo *F-Measure*, era escolhido o valor que tornava a criação do modelo mais rápida, conforme princípio da parcimônia [6].

O último modelo avaliado nessa iteração foi o *Hill-climbing Ensemble Selection with Bootstrap Sampling* (HCES-Bag). Neste trabalho, os algoritmos de classificação base para o HCES-Bag foram LR, RF, GBM e ANN. Para essa execução, buscou-se usar os parâmetros de *Grid Search* iniciais descritos na Tabela 5.8. A implementação de Lambert utilizou como base os algoritmos disponíveis no *framework* Scikit-Learn [39]. Diferenças entre a parametrização disponível nesse pacote e no H2O impediram a utilização idêntica

[5]Disponível em `https://www.h2o.ai/`.
[6]Disponível em `https://github.com/dclambert/pyensemble`.

Tabela 5.8: Pârametros avaliados por *Grid Search* dos algoritmos *Logistic Regression*, *Random Forest*, *Gradient Boosting Machine* e *Artificial Neural Networks*.

Algoritmo	**Parâmetro**	**Valores Testados**	**Modelo Final**
LR	α	0,0, 0,1, 0,5, 0,7 e 1,0	0,1
	λ	Busca automática	0,00045
RF	ntrees	50 e 100	100
	max_depth	2 e 20	20
	mtries	1 e 7	7
	sample_rate	0,6 e 1,0	1
	col_sample_rate_per_tree	0,5 e 1,0	0,5
	min_split_improvement	10^{-5} e 10^{-4}	10^{-5}
GBM	ntrees	50 e 100	100
	max_depth	2 e 20	20
	learn_rate	0,5 e 1,0	0,5
	sample_rate	0,6 e 1,0	1,0
	col_sample_rate_per_tree	0,5 e 1,0	0,5
	min_split_improvement	10^{-5} e 10^{-4}	10^{-5}
ANN	activation	tanh, rectifier e maxout	tanh
	epochs	5 e 10	5
	hidden	(10, 10) e (10, 10, 10)	(10, 10)
	L_1	0,0 e 0,5	0,0
	L_2	0,0 e 0,5	0,0
	input_dropout_ratio	0,1 e 0,2	0,1

dos algoritmos nos dois *frameworks*. As documentações das duas bibliotecas tiveram de ser detalhadamente comparadas para efetuar os ajustes necessários. Como exemplo dessas diferenças, na LR o parâmetro α no Scikit-Learn é o λ no H2O, enquanto que o parâmetro α no H2O é o *l1_ratio* no Scikit-Learn[7] [8]. Em outros casos não havia equivalência, a exemplo do parâmetro *col_sample_rate_per_tree* nos algoritmos RF e GBM, disponível somente no pacote H2O [9] [10] [11] [12].

As curvas ROC dos modelos testados nessa primeira iteração usando a base de teste são mostradas na Figura 5.8. É possível observar que o modelo atualmente utilizado pelo Banco Alfa se aproximou bastante do valor mínimo, sendo suas predições comparadas a uma tomada de decisão totalmente aleatória. Os algoritmos de MD obtiveram melhores

[7]Disponível em `http://scikit-learn.org/stable/modules/generated/sklearn.linear_model.SGDClassifier.html`.

[8]Disponível em `http://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/glm.html`.

[9]Disponível em `http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html`.

[10]Disponível em `http://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/drf.html`.

[11]Disponível em `http://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html`.

[12]Disponível em `http://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/gbm.html`.

Tabela 5.9: Resultado da primeira iteração de Mineração de Dados.

Modelo	AUC	F-Measure	Acurácia	TP	TN	FP	FN
RF	0,94	0,56	0,96	350	11.566	331	230
HCES-Bag	0,94	0,51	0,94	415	11.258	639	165
ANN	0,91	0,50	0,96	253	11.719	178	327
LR	0,90	0,51	0,95	304	11.599	298	276
GBM	0,81	0,55	0,96	343	11.583	314	237
Linha de Base	0,56	0,14	0,89	115	10.961	936	465

resultados e ficaram bastante próximos entre si, sendo que o GBM obteve a menor AUC entre eles.

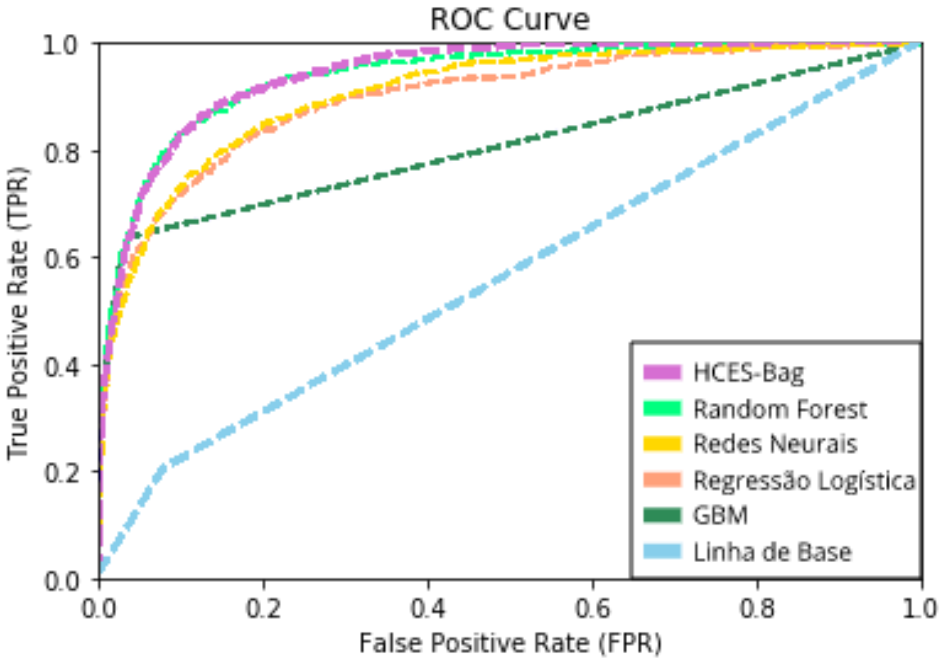


Figura 5.8: Curva ROC dos modelos preditivos sobre a base de teste.

Uma síntese dos resultados obtidos nessa primeira iteração é mostrada na Tabela 5.9. Verifica-se que o modelo gerado por meio de *Random Forest* obteve os melhores resultados.

As dez variáveis apontadas como mais importantes pelo modelo gerado pelo algoritmo RF estão descritas na Tabela 5.10 e correspondem a 64% da redução do erro quadrático médio [20].

5.5 Avaliação

Essa etapa tem como objetivo mostrar os resultados obtidos dos modelos criados e avaliá-los de acordo com os objetivos de negócio [36].

Tabela 5.10: Dez variáveis mais importantes para o modelo vencedor.

Definição	**Quantidade**
Margem de contribuição	1
Movimentação financeira	3
Relacionadas a tempo de posse de produtos em dias	3
Renda presumida	1
Quantidade de produtos	1
Valores de endividamento no Sistema Financeiro Nacional	1

Os resultados alcançados com o estudo mostraram-se satisfatórios para o Banco Alfa, pois obtiveram resultados melhores que a estratégia atual e podem ser implementados com facilidade, devido à disponibilidade dos dados e dos algoritmos utilizados.

Nesse meio tempo, o setor do Banco Alfa conduzia uma campanha direcionada a clientes digitais, com o objetivo de oferecer a mudança para contas convencionais. Uma amostra dos 500 maiores percentuais de confiança do modelo proposto coincidiu com 180 clientes digitais que aceitaram a mudança do tipo de conta.

Contudo, o percentual de clientes digitais identificados como de Alta Renda, pelo modelo preditivo, ficou muito abaixo do percentual de clientes convencionais com a mesma faixa de renda, trazendo dúvidas sobre a qualidade das previsões. Uma das possibilidades era de sobreajuste, que poderia ser causado tanto pelo tamanho da base de treinamento, quanto pela baixa importância das variáveis utilizadas. Porém, dado que o modelo preditivo foi criado a partir de uma base de clientes convencionais, questionou-se também a sua adequação no universo de clientes digitais. Dadas essas indagações, iniciou-se uma segunda iteração de CRISP-DM, ainda que o modelo tradicional tenha sido substituído em parte pelo modelo criado nessa primeira iteração.

5.6 Implantação

Após as etapas de treinamento, validação e teste, foi executado o modelo baseado no algoritmo *Random Forest* sobre a base de clientes digitais. Com o resultado obtido, um arquivo em formato CSV foi disponibilizado ao setor responsável no Banco Alfa. Esse arquivo era formado pelo campo de identificação do cliente digital, uma indicação de Alta Renda e a probabilidade dessa indicação.

Como se mostrou necessário executar mais uma iteração de aprimoramento do modelo, foi somente disponibilizado um arquivo com as predições do grupo de clientes digitais até Outubro de 2017, não produzindo assim previsões para novos clientes digitais. Para os novos entrantes, o método tradicional ainda tem sido utilizado. Espera-se que após o fim da segunda iteração, seja possível criar um aplicativo de uso irrestrito.

Capítulo 6

Segunda Iteração

Após os resultados encontrados no Capítulo 5, mostrou-se necessário buscar melhorias no modelo de Mineração de Dados. A segunda investida sobre o problema proposto é descrito neste Capítulo, que seguiu a mesma organização das etapas propostas pelo CRISP-DM.

6.1 Entendimento do Negócio

As situações descritas na Capítulo 5.1 não mudaram inteiramente com a implantação do modelo criado na primeira iteração. As informações disponíveis sobre os clientes digitais ainda continuaram mínimas e a estratégia de utilizar dados dos clientes convencionais pareceu ainda ser a mais acertada. A criação do primeiro modelo utilizando dados já internalizados pelo Banco Alfa mostrou bons resultados em comparação ao método preditivo tradicional. As informações disponíveis em *data warehouse* já haviam se esgotado e outras informações ainda não exploradas deveriam ser adquiridas na tentativa de melhorar o modelo preditivo.

6.2 Entendimento dos Dados

O aplicativo de *smartphone* do Banco Alfa possui um algoritmo de captura de dados que disponibiliza uma série de informações, como o código do cliente, a latitude e a longitude do celular, o modelo do aparelho, sua resolução de tela, sua quantidade de processadores, entre outras. A coleta ocorre no momento de *login* do cliente no aplicativo, que ocorre sempre que o cliente o utiliza. Essa coleta, contudo, depende da autorização do cliente, o que reduz a gama de informações acessíveis. Dados geográficos e informações detalhadas sobre características do aparelho de *smartphone*, por exemplo, são mínimos. A marca e o modelo do aparelho tem se mostrado como as informações mais disponíveis.

A distribuição de marcas de aparelho entre os clientes do Banco Alfa se dá por três sistemas operacionais: *Android*, *iOS* e *Windows*. A Figura 6.1 mostra a distribuição de clientes do Banco Alfa por sistema operacional.

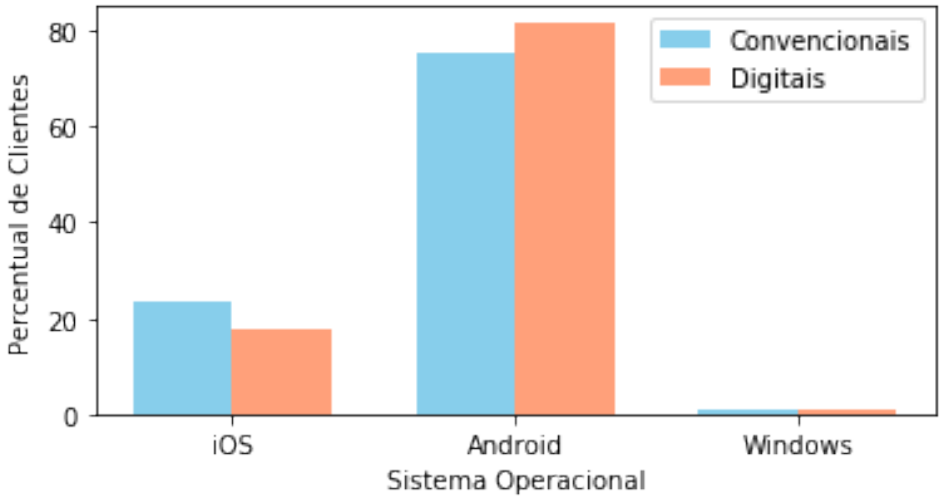


Figura 6.1: Distribuição de *smartphones* dos clientes do Banco Alfa por sistema Operacional.

Ao invés de utilizar o nome do modelo do aparelho do cliente para a modelagem, optou-se por extrair seu preço médio. Para isso, foi usada a *Application Programming Interface* (API) de uma grande loja virtual para efetuar a consulta do preço médio desses aparelhos. Toda a base de nomes de modelos de telefone dos clientes do Banco Alfa foi aplicada a essa API, que retornava diversas ofertas de venda dos aparelhos. Para cada um, foi efetuada uma pesquisa que retornava até 25 resultados, dos quais se apurava o valor médio. Depois disso, o valor do aparelho era vinculado aos demais dados do cliente. A Figura 6.2 mostra um gráfico da distribuição de preços de *smartphones* dos clientes do Banco Alfa, onde se pode observar a semelhança entre os dois públicos nesse quesito.

Outra informação não utilizada na primeira iteração foi o CEP de residência dos clientes. Verificando esse dado junto à base dos Correios[1], foi possível determinar em que Unidade da Federação o cliente morava e se residia em capital de estado ou não.

Além disso, semelhantemente ao que foi feito com os modelos de telefone, obteve-se o preço médio, a metragem quadrada e o valor do metro quadrado de casas à venda em cada CEP. Na Figura 6.3 é possível perceber diferenças maiores nos valores encontrados para clientes convencionais e digitais do que para o preço do celular.

[1] `http://www.correios.com.br/`

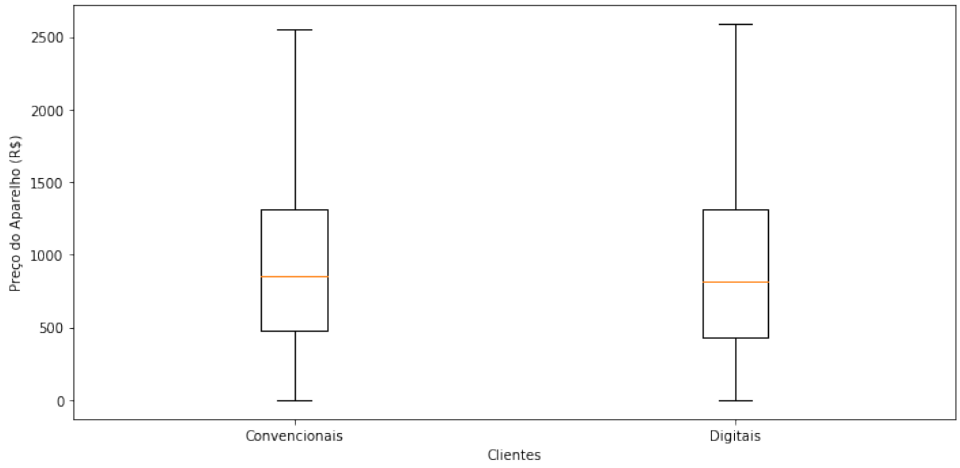
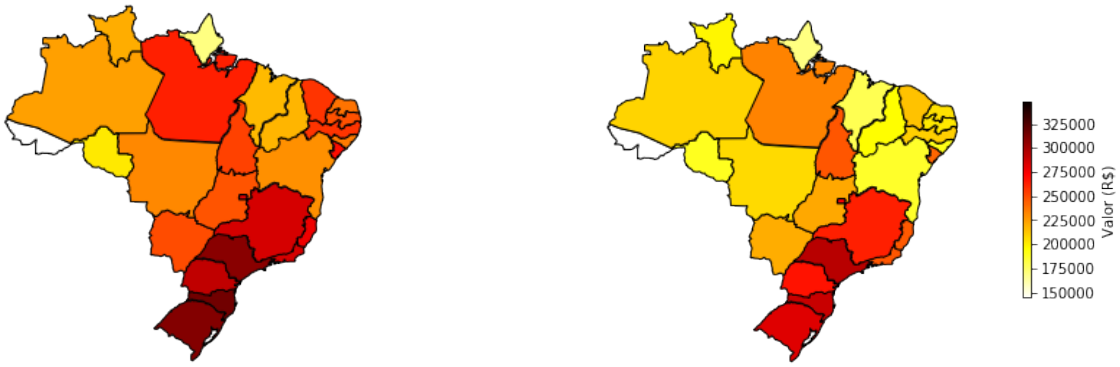


Figura 6.2: Distribuição de preço dos modelos de *smartphones* dos clientes do Banco Alfa.



(a) Clientes Convencionais.

(b) Clientes Digitais.

Figura 6.3: Preço médio de casas no CEP de clientes do Banco Alfa.

Com a utilização do CEP de residência dos clientes, foi possível ainda obter dados de seus municípios, como o Produto Interno Bruto (PIB) e o índice Gini[2], por meio de cruzamento de informações com o sítio do IBGE[3]. Adicionalmente, o IBGE separa porções do território brasileiro em micro e mesorregiões [40]. Havia também em seu endereço *web* a informação do PIB de cada uma dessas regiões.

[2]O índice de Gini mede até que ponto a distribuição da renda entre indivíduos ou famílias dentro de uma economia se desvia de uma distribuição perfeitamente igual. Fonte: `http://databank.worldbank.org`, em 01/06/2018.

[3]Disponível em `https://www.ibge.gov.br/`.

6.3 Preparação dos Dados

Esta seção descreve a preparação dos novos dados obtidos, sua inclusão à base criada na primeira iteração e como as variáveis explicativas e os registros foram selecionados no intuito de reduzir a possibilidade de sobreajuste, cogitada na Seção 5.5.

A informação do sistema operacional do *smartphone* dos clientes foi transformada em três variáveis binárias que indicavam com 1 se o celular do cliente era *Android*, *iOS* ou *Windows*.

O preço dos aparelhos foi capturado via API de uma loja virtual que vende grande variedade de produtos novos e usados. Como alguns modelos de aparelhos não foram encontrados, foi realizada uma imputação de dados com o valor médio dos celulares dos clientes, que resultou em R$ 1.081,00. A Figura 6.4 mostra a diferença de valores entre clientes convencionais por faixa de renda.

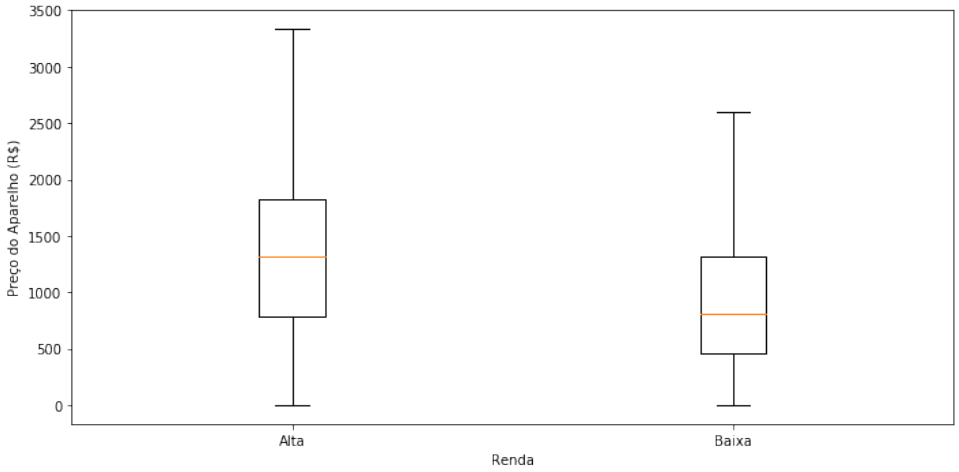


Figura 6.4: Preço dos celulares por faixa de renda.

As informações sobre preço e tamanho de imóveis foram obtidas por meio de um *crawler*[4]. A partir de uma pesquisa pelo nome de bairro vinculado a um CEP, o programa varria a página de resultados e recuperava o valor e a metragem dos imóveis anunciados para venda. Após isso, era calculado o preço, a metragem e o preço do metro quadrado médios para cada bairro. A imputação dos valores faltantes valeu-se da construção do código de CEP, como mostrado na Figura 6.5. Esse código de natureza posicional indica 5 níveis de regiões, sendo que os algarismos da esquerda para a direita correspondem a regiões cada vez menores.

Assim, a imputação de valores utilizou o número formado para o CEP como variável explicativa de um Regressão baseada nos vizinhos mais próximos KNN [14]. O algoritmo KNN foi configurado para encontrar os dois CEP mais próximos, utilizando a distância

[4]Programa que navega automaticamente em uma determinada página *web* capturando dados.

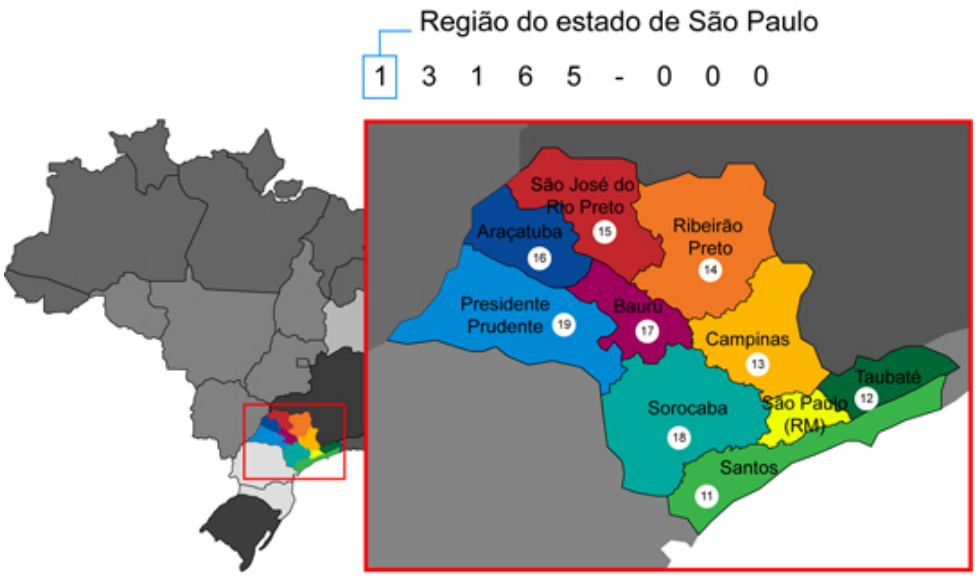
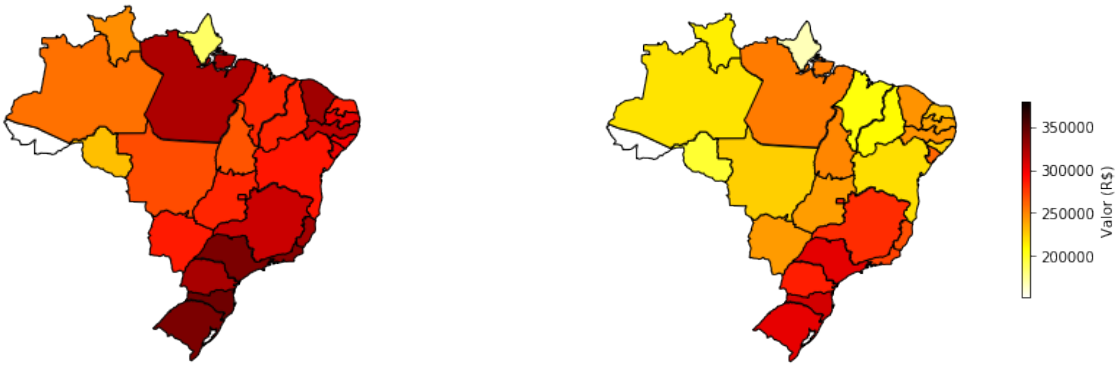


Figura 6.5: Exemplo de estrutura do CEP. Fonte: `https://www.correios.com.br`, acessada em 01/06/2018.

de *Manhatan* [6], isto é a diferença absoluta entre os dois valores. Ao encontrar esses dois vizinhos, foi efetuada a média aritmética para imputar o valor faltante para aquele CEP. Esse procedimento foi adotado tanto para o valor quanto para a metragem quadrada dos imóveis. O preço do metro quadrado foi obtido divindo o preço do imóvel pela metragem quadrada.

Conforme mostra a Figura 6.6, as diferenças encontradas nos preços dos imóveis de clientes convencionais de Baixa e Alta renda indicam uma forte relação entre renda e moradia, o que motivou sua utilização.



(a) Alta Renda.

(b) Baixa Renda.

Figura 6.6: Preço médio de casas no CEP de clientes convencionais do Banco Alfa por faixa de renda.

Cada cliente também foi identificado como morador de capital de estado por meio de uma variável binária. Foram também agregados os valores de PIB do município, da microrregião e da mesorregião. O índice Gini do município do cliente também foi acrescentado. Para mais, foram criadas variáveis indicadoras de Unidade da Federação e das meso e microrregiões, todas binárias.

As variáveis numéricas inteiras e decimais sofreram transformação Box-Cox e foram padronizadas dentro da faixa [0, 1]. As variáveis explicativas adicionais estão detalhadas na Tabela 6.1. Após serem unificadas, contava-se com 227 variáveis explicativas.

Tabela 6.1: Variáveis explicativas

Definição	**Tipo**	**Domínio**	**Quantidade**
Dados sobre imóveis da vizinhança	Decimal	$[0, +\infty)$	3
Preço do celular	Decimal	$[0, +\infty)$	1
Sistemas operacionais de celulares	Binária	0, 1	3
Indicadores Demográficos (IBGE)	Decimal	$[0, +\infty)$	4
Residência em Capital	Binária	0, 1	1
Residência em Unidade da Federação	Binária	0, 1	27
Residência em Mesorregião	Binária	0, 1	138

Devido ao grande número de variáveis e à suspeita de sobreajuste descrita na Seção 5.5, foram executadas ações para avaliação da importância das variáveis explicativas utilizadas para o novo modelo preditivo.

Primeiramente foi feita uma análise de correlação de Pearson, a fim de verificar o quanto uma variável explicativa estava correlacionada com a variável alvo, além de saber se haviam variáveis explicativas correlacionadas. A Figura 6.7 mostra em amarelo as variáveis correlacionadas. A linha 0 (zero) corresponde à correlação da variável alvo com as demais.

Além da análise de correlação, foi realizada uma comparação entre os valores médios das variáveis contínuas, separadas pelas categorias da variável alvo. Assim cada variável possuía dois valores médios, um de Baixa Renda e outro de Alta Renda. Foram executados testes T de Student para as variáveis, com a hipótese nula rejeitada em todos os casos. Tal situação pode ter ocorrido devido ao grande volume de observações, cerca de 2 milhões, que aumentaria bastante o poder do teste, mesmo para diferenças mínimas entre as médias [41]. Desta forma, a comparação de médias foi executada de maneira mais simples. Como os dados estavam padronizados para a escala [0, 1], foram mantidas, para estudo, as variáveis que possuíam diferenças entre as médias maiores ou iguais a 0,1. A Figura 6.8 mostra uma representação de duas das variáveis analisadas, usando essa comparação.

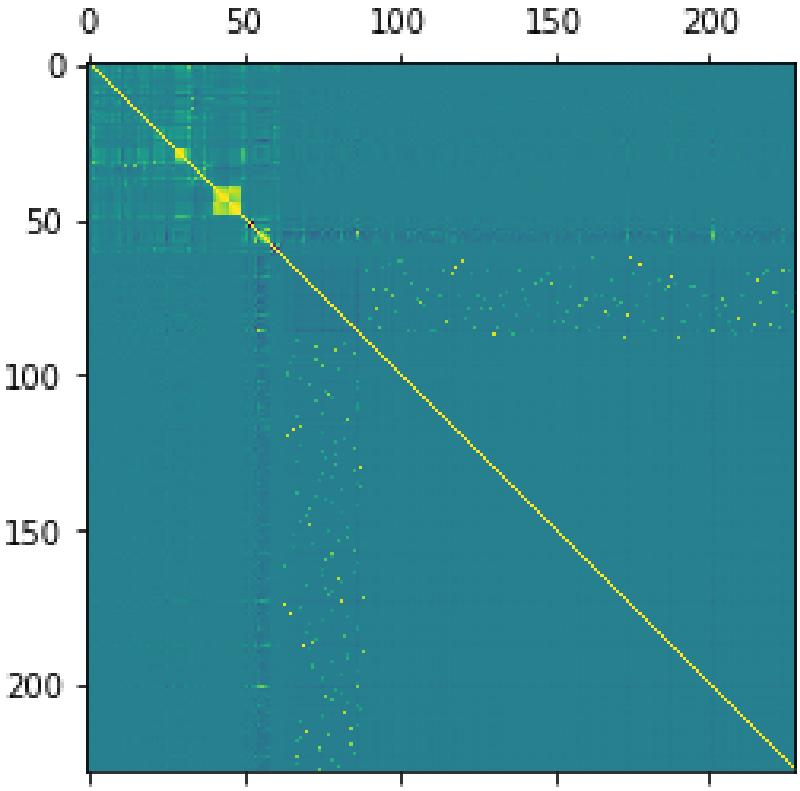
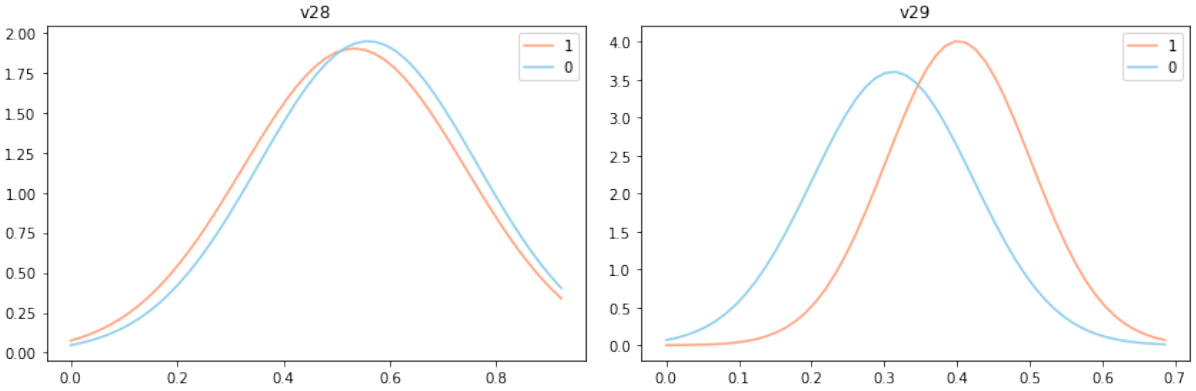


Figura 6.7: Matriz de correlação entre todos os atributos.



(a) Médias iguais.

(b) Médias diferentes.

Figura 6.8: Representação das distribuições de variáveis contínuas com dois tratamentos relativos aos valores da variável alvo (legenda).

Variáveis contínuas que possuíam variância distinta em relação aos dois grupos de renda, como a mostrada na Figura 6.9, não foram descartadas de pronto, mas lhes foi aplicada uma *Principal Component Analysis* (PCA) [11]. Diferentemente da comparação de médias, foi aplicado um PCA separado para cada faixa de renda, de forma a saber quais variáveis demonstravam maior variância em cada grupo. Como resultado, foram

geradas componentes formadas pela composição de diversas variáveis. As 10 primeiras variáveis da primeira componente em cada grupo foram selecionadas.

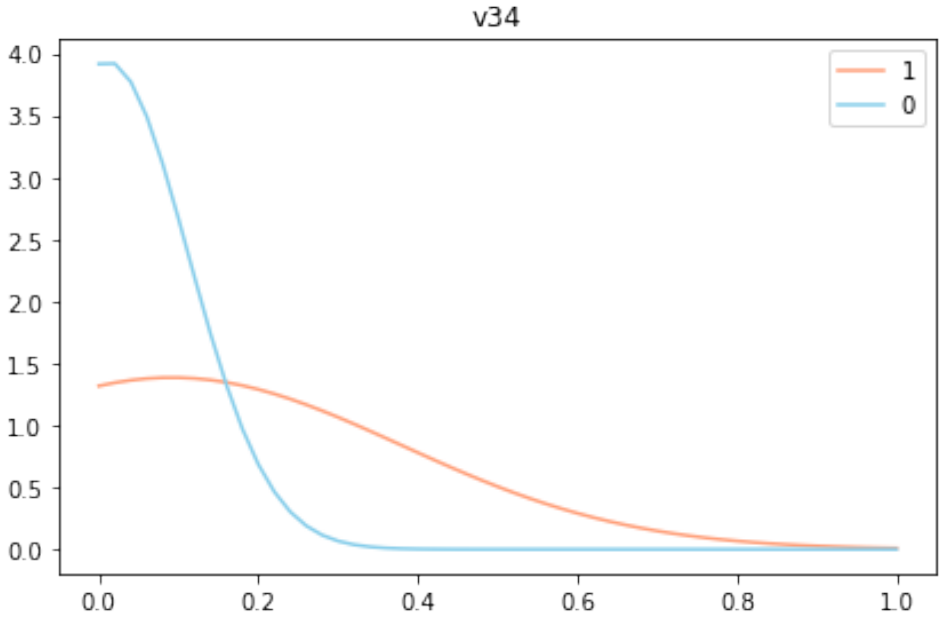
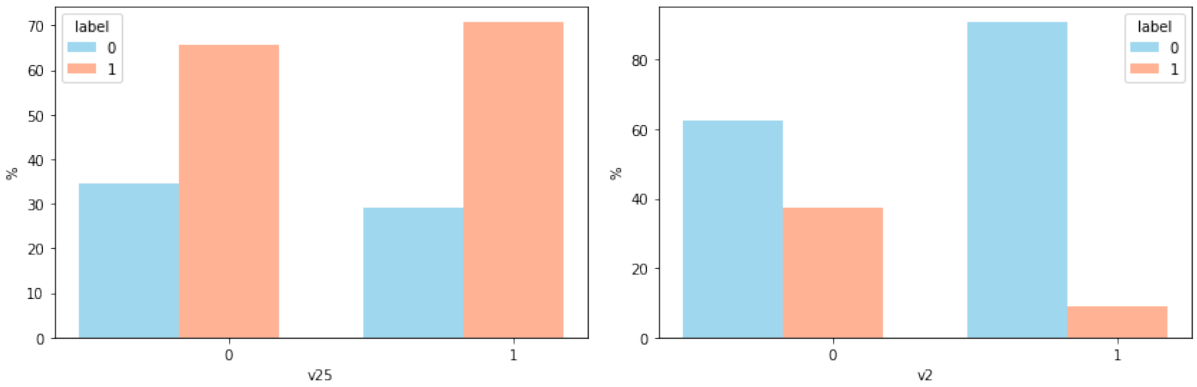


Figura 6.9: Variâncias distintas entre os dois tratamentos.

Para as variáveis binárias, foi aplicado um Teste Qui-Quadrado [42] a fim de determinar se havia diferença entre as proporções de 0 (Falso) e 1 (Verdadeiro) de tais variáveis em relação à variável alvo, também binária. Um valor p abaixo de 0,05 foi usado para indicar se a diferença de proporções era significativa. Variáveis abaixo desse valor de corte foram mantidas. A Figura 6.10 mostra uma representação gráfica de uma variável mantida e outra descartada por esse teste.



(a) Proporções iguais.

(b) Proporções diferentes.

Figura 6.10: Representação do teste de homogeneidade, ou Qui-Quadrado, para variáveis binárias, com dois tratamentos baseados nos valores da variável alvo (label).

Tabela 6.2: Variáveis selecionadas por diversos métodos discriminantes.

Análise	**Critério**	**Variáveis Selecionadas**
Correlação	Correlação absoluta maior que 0,15	14
Comparação de Médias	$\|\bar{x_0} - \bar{x_1}\| < 0,1$	10
Teste de Homogeneidade	Valor p $<$ 0,05	8
Principal Component Analysis	Primeira Componente	22

Os testes aplicados e os critérios de seleção dos atributos estão resumidos na Tabela 6.2. Pode-se observar que apesar do grande número de variáveis criadas até esse ponto (227), os diversos testes selecionaram menos de 10% do total, o que torna mais simples a futura implantação do modelo. Ainda assim, as variáveis selecionadas foram testadas em relação à colinearidade.

Para tratar a multicolinearidade de variáveis explicativas, foi calculado o VIF. Após sua execução, restaram 20 variáveis explicativas. Foi executada uma nova análise de correlação com a variável alvo, que está ilustrada na Figura 6.11.

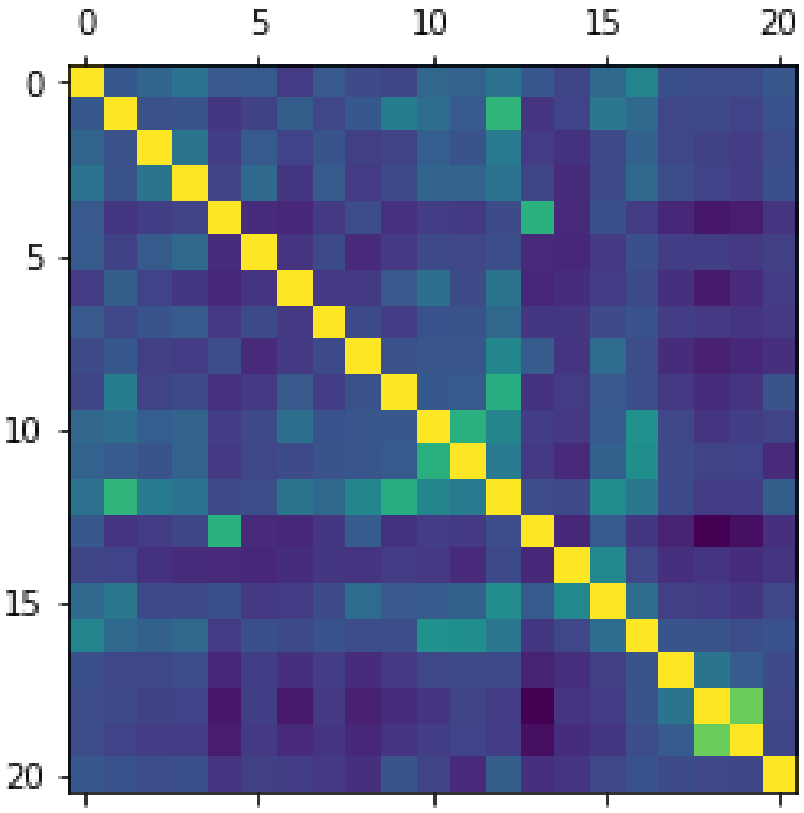


Figura 6.11: Matriz de correlação entre atributos selecionados.

Após a seleção das variáveis explicativas, foram escolhidos os registros a serem usados

para a modelagem. Como foi explicado na Seção 5.3, na primeira iteração foram utilizados para esse fim dados de clientes convencionais residentes na cidade de Joinville (SC). Dadas as suspeitas de sobreajuste na Seção 5.5 e que a maioria das variáveis desta segunda iteração estão relacionadas a localização, decidiu-se por usar toda a base de clientes convencionais para o estudo, que compreende todos os estados brasileiros.

Contudo, devido ao problema de desbalanceamento de dados explicado na Seção 5.2, foram aplicadas novamente técnicas de balanceamento. A Tabela 6.3 mostra o *F-Measure* obtido com cada uma dessas bases ao executar uma LR. Técnicas mais elaboradas, como ADASYN, *Cluster Centroids*, SMOTE e suas variantes foram executadas, mas a grande quantidade de registros tornou o tempo de execução proibitivo.

Tabela 6.3: Resultado do teste das técnicas de balanceamento de categorias.

Técnica	**F-Measure**
Random Oversampling	0,549
Random Undersampling	0,549
Dados Desbalanceados	0,551

Devido à pequena diferença entre os valores e à grande quantidade de dados, optou-se por utilizar a base de treino criada com a técnica *Random Undersampling*. Desta forma, a base de treinamento ficou com 1.307.599 registros e a base de teste com 2.852.461 registros.

6.4 Modelagem

Em comparação à primeira iteração, não foram realizadas muitas mudanças na etapa de modelagem. O modelo de linha de base foi novamente incluído, principalmente para servir de comparação em relação aos algoritmos de MD. Os algoritmos LR, RF, GBM e ANN foram executados utilizando novamente o *framework* H2O. O algoritmo HCES-Bag valeu-se novamente da implementação de Lambert.

Foram incluídos dois parâmetros no *Grid Search*, para os algoritmos RF e GBM. O parâmetro *min_rows* foi utilizado para evitar erros de execução, devido à grande quantidade de registros pois, como foi explicado no Capítulo 2, ele é usado para determinar a quantidade mínima de registros em um ramo da árvore de decisão, para que ocorra divisão em dois ramos. O parâmetro *col_sample_rate_change_per_level* não foi utilizado na primeira iteração porque apresentava erro no *framework* H2O. Porém, durante a segunda iteração, foi disponibilizada uma versão com a correção. Alguns valores finais divergem dos valores iniciais do *Grid Search* porque foram, novamente, executadas rodadas de refinamento. Os detalhes podem ser observados na Tabela 6.4.

Tabela 6.4: Pârametros avaliados por *Grid Search* dos algoritmos *Logistic Regression*, *Random Forest*, *Gradient Boosting Machine* e *Artificial Neural Networks*.

Algoritmo	**Parâmetro**	**Valores Testados**	**Modelo Final**
LR	α	0,0, 0,1, 0,5, 0,7 e 1,0	1,0
	λ	Busca automática	0,00082
RF	ntrees	50 e 100	100
	max_depth	2 e 20	20
	mtries	1 e 4	4
	sample_rate	0,6 e 1,0	1
	col_sample_rate_per_tree	0,5 e 1,0	1,0
	col_sample_rate _change_per_level	1 e 2	2
	min_rows	5.000 e 50.000	100
	min_split_improvement	10^{-5} e 10^{-4}	10^{-5}
GBM	ntrees	5 e 10	20
	max_depth	20	20
	learn_rate	0,5 e 1,0	0,5
	sample_rate	0,6 e 1,0	1,0
	col_sample_rate_per_tree	0,5 e 1,0	1,0
	col_sample_rate _change_per_level	1 e 2	2
	min_rows	500 e 1000	500
	min_split_improvement	10^{-5} e 10^{-4}	10^{-5}
ANN	activation	tanh, rectifier e maxout	maxout
	epochs	5 e 10	10
	hidden	(10, 10) e (10, 10, 10)	(20, 20, 20)
	L_1	0,0 e 0,5	0,0
	L_2	0,0 e 0,5	0,0
	input_dropout_ratio	0,1 e 0,2	0,15

O modelo baseado no algoritmo HCES-Bag foi criado da mesma maneira que na primeira iteração, isto é, usando a parametrização inicial dos algoritmos LR, RF, GBM e ANN. As curvas ROC dos modelos gerados a partir da base de teste nessa iteração podem ser vistas na Figura 6.12.

As métricas alcançadas pelos algoritmos estão descritas na Tabela 6.5. Desta vez os modelos ficaram bastante parecidos, sendo que o modelo criado pelo algoritmo *Gradient Boosting Machine* se destacou, por muito pouco, dos modelos baseados em *Random Forest* e HCES-Bag.

As dez variáveis apontadas como mais importantes pelo modelo gerado pelo algoritmo RF estão descritas na Tabela 6.6 e são responsáveis por 97% da redução do erro quadrático médio [20].

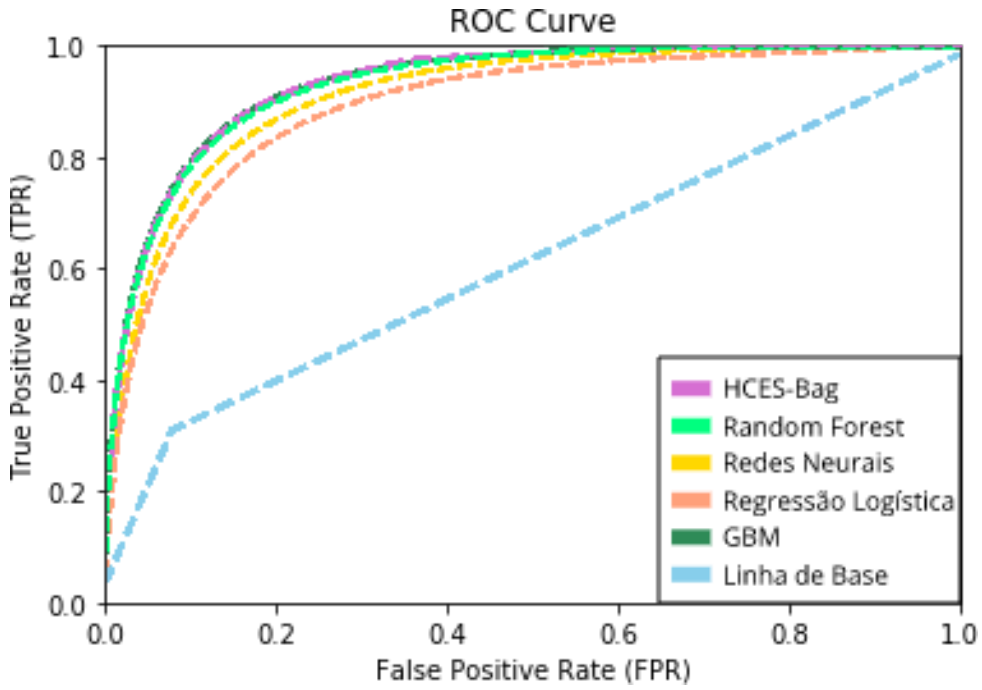


Figura 6.12: Curva ROC dos modelos criados usando a base de teste.

Tabela 6.5: Resultado da segunda iteração de Mineração de Dados.

Modelo	**AUC**	**F-Measure**	**Acurácia**	**TP**	**TN**	**FP**	**FN**
GBM	0,93	0,62	0,92	195.577	2.421.430	135.261	100.193
RF	0,93	0,62	0,91	194.954	2.414.171	142.520	100.816
HCES-Bag	0,93	0,52	0,83	259.540	2.115.323	441.368	36.230
ANN	0,91	0,58	0,90	191.274	2.382.598	174.093	104.496
LR	0,89	0,55	0,90	180.144	2.376.269	180.422	115.626
Linha de Base	0,61	0,30	0,86	86.045	2.355.689	200.859	209.868

6.5 Avaliação

A fim de verificar se houve melhora nos modelos da segunda iteração em relação ao primeiro, foram efetuadas algumas comparações nas previsões desses modelos sobre o público de clientes digitais. A primeira verificação foi a quantidade de clientes digitais que os modelos indicaram como de Alta Renda. O modelo da primeira iteração classificou menos de 0,1% dos clientes como ricos, havendo um forte indício de sobreajuste. O modelo da segunda iteração, por outro lado, marcou 3,7% de clientes como de Alta Renda, ou seja, um percentual muito mais próximo dos 10% referentes ao clientes convencionais. Não é possível afirmar se há entre os clientes digitais um percentual de pessoas de Alta Renda semelhante ao precentual dos clientes convencionais, mas ainda assim espera-se que esse valor seja bem maior do que o apresentado pelo modelo da primeira iteração.

Os modelos da segunda iteração tem outras vantagens em relação aos modelos da

Tabela 6.6: Dez variáveis mais importantes para o modelo vencedor.

Definição	**Quantidade**
Dados sobre imóveis da vizinhança	1
Idade	1
Indicadores de posse de produtos	1
Indicadores demográficos (IBGE)	1
Quantidade de produtos	1
Relacionadas a tempo de posse de produtos em dias	1
Renda presumida	1
Sistemas operacionais de celulares	1
Valores de endividamentono no Sistema Financeiro Nacional	2

primeira, como uma menor quantidade de variáveis explicativas e o uso de dados de todo o Brasil. Adicionado a isso, a variedade de modelos com bons resultados permite a escolha daquele de implantação mais fácil.

6.6 Implantação

Durante a escrita desse trabalho, o Banco Alfa ainda verificava os resultados da segunda iteração. Espera-se que o modelo criado substitua finalmente o modelo de linha de base. A criação de um aplicativo a partir do modelo permitirá que o setor de *marketing* inclua regularmente novos clientes em suas campanhas.

Capítulo 7

Conclusão

Contas bancárias digitais são uma realidade brasileira. A facilidade proporcionada pela abertura de conta por meio de um aplicativo de *smartphone* se tornou bastante popular, atraindo milhões de pessoas para esse produto.

Apesar das contas digitais serem vantajosas para os clientes, instituições financeiras tem dificuldade de conhecer melhor esses clientes, devido ao cadastro mínimo necessário para sua contratação. Devido à falta de detalhes como a renda comprovada dos clientes, torna-se difícil definir uma ordem de priorização nas ações de *telemarketing*, podendo prejudicar a venda de novos produtos a esses clientes.

O presente trabalho propôs a criação de um modelo preditivo de renda para correntistas digitais a partir de informações oriundas de fontes diversas, tais como: características dos *smartphones*, locais de residência, grau de endividamento no Sistema Financeiro Nacional, entre outras.

Foram utilizados, como base de estudo, dados de correntistas detentores de contas convencionais, mas que usavam o mesmo aplicativo de *smartphone* que os correntistas digitais. As rendas foram divididas em duas faixas, Alta e Baixa, sendo considerado como divisor o valor de dez mil reais. Foram testados diversos algoritmos de Mineração de Dados: *Logistic Regression*, *Random Forest*, *Gradient Boosting Machine*, *Artificial Neural Networks* e *Hill-climbing Ensemble Selection with Bootstrap Sampling*. Esses modelos foram comparados entre si e entre o modelo preditivo até então em uso por uma instituição financeira. Os modelos de MD obtiveram resultados muito superiores em comparação à solução anterior, sendo que o melhor modelo criado obteve AUC de 0,93, Acurácia de 0,92 e *F-Measure* de 0,62.

Além da criação de um modelo preditivo superior para a instituição bancária que financiou o projeto, foi possível contribuir[1] para o *framework* de Mineração de Dados

[1] A indicação da falha encontrada e a descrição da solução podem ser encontradas em `https://0xdata.atlassian.net/browse/PUBDEV-5334`.

H2O, usado extensivamente nesse trabalho.

Uma das questões levantadas, durante a execução da pesquisa, consistia na melhor renda de corte para separar os clientes de alta e de baixa renda. No Capítulo 5 observou-se, utilizando uma análise baseada somente na renda, que valores superiores a R$ 2.958,28 já definiam bem os dois grupos. Trabalhos futuros podem abordar esse problema mais detalhadamente, utilizando mais variáveis para isso ou identificando a melhor quantidade de faixas de renda.

A tarefa de descoberta e utilização de novas variáveis também deve ser levada adiante, a fim de melhorar ainda mais o poder preditivo do modelo de MD. Não foram exploradas, por exemplo, características dos *smartphones* dos clientes, tais como: tamanho e resolução da tela, tamanho da memória, tipo de processador, entre outros dados de uso autorizado. Conforme descrito no Capítulo 3 em diversos trabalhos, a informação de geolocalização dos *smartphones* também pode ser utilizada para construção de variáveis explicativas de um modelo preditivo. Essas informações não estavam disponíveis durante a execução dessa pesquisa, mas trabalhos futuros podem se beneficiar de tais dados.

Outra dificuldade encontrada durante a pesquisa foi a criação do modelo a partir do algoritmo HCES-Bag. Por se basear na biblioteca *Scikit-Learn*, os parâmetros de configuração para os algoritmos utilizados no HCES-Bag diferiam entre os mesmos algoritmos da biblioteca *H2O*, usada na criação dos outros modelos. Criar uma implementação do algoritmo a partir da API do *H2O* pode tornar mais direta a inclusão de outros algoritmos no *ensemble*, pois não seria necessário fazer adaptações às diferenças de parâmetros entre as bibliotecas.

O desbalanceamento entre as duas categorias de renda levou à tomada de duas decisões. A primeira foi usar técnicas de balanceamento como *undersampling* e *oversampling*. A outra decisão foi selecionar o melhor modelo levando em consideração as métricas AUC e *F-Measure* em oposição à Acurácia. Contudo, como explicado no 2, técnicas de balanceamento podem causar sobreajuste. Uma alternativa é usar *Aprendizagem Sensível ao Custo* [6]. Ela mantém as classes desbalanceadas, mas determina custos específicos para cada tipo de erro de classificação. Essa abordagem pode se mostrar vantajosa para o Banco Alfa, caso sejam utilizados os custos referentes, tanto ao contato com clientes não propensos ao produto ofertado, quanto à perda potencial de não contactar clientes propensos. Assim, o modelo selecionado poderia indicar de antemão seu custo potencial, tornando mais claro os riscos e os benefícios de sua adoção.

Referências

[1] B. C. do Brasil, "RESOLUÇÃO N. 4.480," Apr. 2016. [Online]. Available: https://www.bcb.gov.br/pre/normativos/busca/downloadNormativo.asp?arquivo=/Lists/Normativos/Attachments/50185/Res_4480_v1_O.pdf 1

[2] Instituto Brasileiro de Geografia e Estatística, Ed., *Acesso à Internet e à televisão e posse de telefone móvel celular para uso pessoal, 2015: Pesquisa Nacional por Amostra de Domicílios*. Rio de Janeiro: IBGE, Instituto Brasileiro de Geografia e Estatística, 2016. 1

[3] B. C. do Brasil, "RESOLUÇÃO N. 3.721," 2009. [Online]. Available: http://www.lume.ufrgs.br/handle/10183/60749 2

[4] I. P. Dias, "Algumas observações sobre a margem de contribuição," *Revista de Administração de Empresas*, vol. 7, no. 24, pp. 79–101, 1967. [Online]. Available: http://www.scielo.br/scielo.php?pid=S0034-75901967000300003&script=sci_arttext&tlng=pt 3

[5] P. Kotler and K. L. Keller, *Administração de marketing*. Prentice Hall, 2002. 6, 23

[6] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction To Data Mining*. Pearson Education, 2006. 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 31, 33, 34, 42, 52

[7] J. Leskovec, A. Rajaraman, and J. D. Ullman, *Mining of massive datasets*. Cambridge university press, 2014. 6

[8] I. Witten, E. Frank, and M. Hall, *Data Mining: Practical Machine Learning Tools and Techniques*, ser. The Morgan Kaufmann Series in Data Management Systems. Elsevier Science, 2011. [Online]. Available: https://books.google.com.br/books?id=bDtLM8CODsQC 7

[9] G. E. P. Box and D. R. Cox, "An Analysis of Transformations," *Research Methods Meeting of the Society*, vol. 26, no. 2, pp. 211–252, 1967. 7

[10] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *Journal of Computational and Applied Mathematics*, vol. 20, no. C, pp. 53–65, nov 1987. 8

[11] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning: with Applications in R*. Springer Science & Business Media, Jun. 2013, google-Books-ID: qcI_AAAAQBAJ. 8, 9, 10, 13, 14, 44

[12] G. Lemaître, F. Nogueira, and C. K. Aridas, "Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning," *Journal of Machine Learning Research*, vol. 18, no. 17, pp. 1–5, 2017. [Online]. Available: http://jmlr.org/papers/v18/16-365.html 9

[13] D. L. Wilson, "Asymptotic Properties of Nearest Neighbor Rules Using Edited Data," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 2, no. 3, pp. 408–421, 1972. 9

[14] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*, 2nd ed., ser. Springer Series in Statistics. New York, NY, USA: Springer New York Inc., 2008. 10, 11, 14, 15, 41

[15] P. Harrington, *Machine Learning in Action*. Manning Publications Company, Dec. 2011, google-Books-ID: 2d7RXwAACAAJ. 10

[16] A. B. Downey, *Think Stats*, 2011. [Online]. Available: http://books.google.com/books?hl=en{&}lr={&}id=TCfZ7d6skT4C{&}oi=fnd{&}pg=PR5{&}dq=Think+Stats{&}ots=LxYK9ntvRZ{&}sig=UchH878Uf04hPLGPmOwnNHBGmdk 10

[17] D. Cook, *Practical Machine Learning with H2O - Powerful, Scalable Techniques for AI and Deep Learning*, 1st ed. Sebastopol, CA: O'Reilly Media, 2017. 10, 11, 12, 13, 34

[18] T. Nykodym, T. Kraljevic, A. Wang, and A. Bartz, *Generalized Linear Modeling with H2O*, 6th ed. Mountain View, CA: H2O.ai, Inc., 2017. [Online]. Available: http://www.h2o.ai/wp-content/uploads/2018/01/GLM-BOOKLET.pdf 10

[19] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001. [Online]. Available: http://www.springerlink.com/index/U0P06167N6173512.pdf 11

[20] "Distributed Random Forest (DRF) — H2O 3.18.0.9 documentation." [Online]. Available: http://docs.h2o.ai/h2o/latest-stable/h2o-docs/data-science/drf.html 11, 36, 48

[21] C. Click, M. Malohlava, A. Candel, H. Roark, and V. Parmar, *Gradient Boosting Machine with H2O*, 6th ed. H2O.ai, 2016. 11, 12

[22] A. Candel and E. LeDell, *Deep Learning With H2O*, 6th ed., A. Bartz, Ed. Mountain View, CA: H2o.ai, Inc., 2018, no. May. [Online]. Available: http://docs.h2o.ai/h2o/latest-stable/h2o-docs/booklets/DeepLearningBooklet.pdf 12

[23] R. Caruana, A. Niculescu-Mizil, G. Crew, and A. Ksikes, "Ensemble selection from libraries of models," in *Proceedings of the Twenty-first International Conference on Machine Learning*, ser. ICML '04. New York, NY, USA: ACM, 2004, pp. 18–. [Online]. Available: http://doi.acm.org/10.1145/1015330.1015432 12

[24] R. Caruana, A. Munson, and A. Niculescu-Mizil, "Getting the most out of ensemble selection," in *Proceedings of the Sixth International Conference on Data Mining*, ser. ICDM '06. Washington, DC, USA: IEEE Computer Society, 2006, pp. 828–833. [Online]. Available: https://doi.org/10.1109/ICDM.2006.76 12, 13

[25] S. Lessmann, B. Baesens, H.-V. Seow, and L. C. Thomas, "Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research," *European Journal of Operational Research*, vol. 247, no. 1, pp. 124–136, Nov. 2015. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0377221715004208 12, 13, 18, 20

[26] M. C. González, C. A. Hidalgo, and A.-L. Barabási, "Understanding individual human mobility patterns," *Nature*, vol. 453, no. 7196, pp. 779–782, Jun. 2008. [Online]. Available: http://www.nature.com/doifinder/10.1038/nature06958 17

[27] V. Frias-Martinez and J. Virseda, "On the relationship between socio-economic factors and cell phone usage," in *Proceedings of the fifth international conference on information and communication technologies and development*. ACM, 2012, pp. 76–84. [Online]. Available: http://dl.acm.org/citation.cfm?id=2160684 17, 20

[28] R. Li, H. Xiong, and H. Zhao, "More than address: Pre-identify your income with the open data," in *2015 International Conference on Cloud Computing and Big Data (CCBD)*, Nov 2015, pp. 193–200. 18, 20

[29] D. Bjorkegren and D. Grissen, "Behavior revealed in mobile phone usage predicts loan repayment," 2015. [Online]. Available: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2611775 18, 20

[30] J. L. Toole, Y.-R. Lin, E. Muehlegger, D. Shoag, M. C. González, and D. Lazer, "Tracking employment shocks using mobile phone data," *Journal of The Royal Society Interface*, vol. 12, no. 107, p. 20150185, Jun. 2015. [Online]. Available: http://rsif.royalsocietypublishing.org/lookup/doi/10.1098/rsif.2015.0185 18, 20

[31] Y. Kim, D. A. Briley, and M. G. Ocepek, "Differential innovation of smartphone and application use by sociodemographics and personality," *Computers in Human Behavior*, vol. 44, pp. 141–147, Mar. 2015. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0747563214006694 18, 20

[32] J. Blumenstock, G. Cadamuro, and R. On, "Predicting poverty and wealth from mobile phone metadata," *Science*, vol. 350, no. 6264, pp. 1073–1076, 2015. [Online]. Available: http://science.sciencemag.org/content/350/6264/1073.short 19, 20

[33] P. Sundsøy, J. Bjelland, B. Reme, A. Iqbal, and E. Jahani, "Deep learning applied to mobile phone data for Individual income classification," in *Proceedings of the 2016 International Conference on Artificial Intelligence: Technologies and Applications*, 2016. 19, 20

[34] A. Kibekbaev and E. Duman, "Benchmarking regression algorithms for income prediction modeling," *Information Systems*, vol. 61, pp. 40–52, Oct. 2016. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0306437916300151 19, 20

[35] J. E. Steele, P. R. Sundsøy, C. Pezzulo, V. A. Alegana, T. J. Bird, J. Blumenstock, J. Bjelland, K. Engø-Monsen, Y.-A. d. Montjoye, A. M. Iqbal, K. N. Hadiuzzaman, X. Lu, E. Wetter, A. J. Tatem, and L. Bengtsson, "Mapping poverty using mobile phone and satellite data," *Journal of The Royal*

Society Interface, vol. 14, no. 127, p. 20160690, Feb. 2017. [Online]. Available: http://rsif.royalsocietypublishing.org/content/14/127/20160690 19, 20

[36] P. Chapman, J. Clinton, R. Kerber, T. Khabaza, T. Reinartz, C. Shearer, and R. Wirth, *CRISP-DM 1.0: Step-by-step Data Mining Guide*. SPSS, 2000. 21, 28, 36

[37] J. da Fonseca and G. de Andrade Martins, *Curso de estatística*, 3rd ed. São Paulo: Atlas, 1996. 30

[38] N. G. Mankiw, *Introdução à economia: princípios de micro e macroeconomia*. Rio de Janeiro: Campus, 1999. 30

[39] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011. 34

[40] J. R. Portela, Ed., *Divisão Regional do Brasil em mesorregioes e microrregiões geográficas*. Rio de Janeiro: IBGE, 1990, vol. 1. 40

[41] G. Sullivan and R. Feinn, "Using effect size—or why the p value is not enough," *Journal of graduate medical education*, vol. 4, pp. 279–82, 09 2012. 43

[42] D. Moore, G. McCabe, W. Duckworth, and L. Alwan, *The Practice of Business Statistics: Using Data for Decisions*, 2nd ed. New York: W. H. Freeman and Company, 2009. 45