Instituto de MATEMÁTICA E ESTATÍSTICA UFRGS

DEPARTAMENTO DE ESTATÍSTICA
E-mail: dest@mat.ufrgs.br

Trabalho de Conclusão de Curso

Estudo comparativo entre abordagens de Aprendizado de Máquina em modelos de Credit Scoring

Cinthia Becker

22 de dezembro de 2018

Cinthia Becker

Estudo comparativo entre abordagens de Aprendizado de Máquina em modelos de Credit Scoring

Trabalho de Conclusão apresentado à comissão de Graduação do Departamento de Estatística da Universidade Federal do Rio Grande do Sul, como parte dos requisitos para obtenção do título de Bacharel em Estatística.

Orientador(a): Profa. Dra. Lisiane Priscila Roldão Selau

Porto Alegre
Dezembro de 2018

Cinthia Becker

Estudo comparativo entre abordagens de Aprendizado de Máquina em modelos de Credit Scoring

Este Trabalho foi julgado adequado para obtenção dos créditos da disciplina Trabalho de Conclusão de Curso em Estatística e aprovado em sua forma final pelo Orientador(a) e pela Banca Examinadora.

Orientador(a):_____
Profa. Dra. Lisiane Priscila Roldão Selau, UFRGS
Doutora pela Universidade Federal do Rio Grande do Sul, Porto Alegre, RS

Banca Examinadora:

Prof. Dra. Lisiane Priscila Roldão Selau, UFRGS
Doutora pela Universidade Federal do Rio Grande do Sul – Porto Alegre, RS

Prof. Dr. Rodrigo Citton Padilha dos Reis, UFRGS
Doutor pela Universidade Federal de Minas Gerais – Minas Gerais, MG

Porto Alegre
Dezembro de 2018

"Sonhos determinam o que você quer. Ação determina o que você conquista.".
(Aldo Novak)

Agradecimentos

À UFRGS, pela oportunidade de ensino, aprendizado, trabalho e crescimento profissional e pessoal.

Aos professores do Instituto de Matemática e Estatística da UFRGS por todo o aprendizado oferecido nesses quatro anos de graduação.

Aos amigos que tive a oportunidade de conhecer e conviver ao longo da faculdade. O apoio de vocês foi fundamental nesta jornada.

Aos demais amigos e familiares que sempre me incentivaram durante esta trajetória.

Aos meus pais (Darci e Janete) e irmão Willian, por todo o amor, incentivo e compreensão nos momentos de ausência. Amo muito vocês.

A minha orientadora, a professora Dra. Lisiane Priscila Roldão Selau, pelos ensinamentos, companheirismo, empenho e exemplo de profissional. Muito obrigada por confiar e acreditar em mim durante este processo.

Resumo

Com o crescimento da demanda e popularização do mercado de crédito no Brasil, as empresas estão buscando maneiras de aprimorar a assertividade na hora de conceder crédito. Há técnicas quantitativas amplamente utilizadas para a construção de modelos de previsão de risco de crédito que, baseadas em informações cadastrais, predizem o comportamento padrão de risco. Porém, estudos recentes mostram que os métodos de Inteligência Artificial têm alcançado melhor desempenho que os métodos estatísticos tradicionais, sendo assim, este trabalho introduz técnicas de Aprendizado de Máquina ainda pouco estudadas em crédito (Árvore de Decisão, *Random Forest, Bagging, Adaboost* e *Support Vector Machine*), a fim de fornecer um modelo com melhor poder explicativo. Para fins de comparação, adotou-se a abordagem tradicional de Regressão Logística. Os modelos foram desenvolvidos em uma base de dados real com 9110 clientes, e foram avaliados em um conjunto de validação de 2279 clientes. Todos os modelos foram analisados com base em três indicadores: percentual de acerto, área abaixo da curva ROC e teste KS. O modelo que apresentou melhor desempenho nos três indicadores avaliados e em ambas amostras de estudo foi o *Adaboost*, sendo esta uma técnica a ser levada em consideração na hora da criação de um modelo de *Credit Scoring*. No entanto, a superioridade encontrada na técnica mencionada pode ser considerada pouco significativa, isso sugere que pode não valer a pena usá-la quando comparada com a técnica padrão de Regressão Logística, devido a sua dificuldade de interpretação e implementação.

Palavras-Chave: Credit Scoring, Aprendizado de Máquina, Regressão Logística.

Abstract

With the growing demand and popularization of the credit market in Brazil, companies are looking for ways to improve assertiveness when it comes to lending credit. There are quantitative techniques widely used for the construction of predictive models of credit risk that, based on cadastral information, predict the standard risk behavior. However, recent studies show that Artificial Intelligence methods have achieved better performance than traditional statistical methods, therefore this work introduces Machine Learning techniques that are not yet studied in credit (Decision Tree, Random Forest, Bagging, Adaboost and Support Vector Machine), in order to provide a model with better explanatory power. For purposes of comparison, the traditional approach of Logistic Regression was adopted. The models were developed based on a database with 9110 clients, and they were evaluated in a validation set of 2279 clients. All models were analyzed based on three indicators: hit percentage, area below the ROC curve and KS test.The model that presented the best performance in the three indicators evaluated and in both samples of study was the Adaboost, which is a technique to be taken into account in the creation of a model of Credit Scoring. However, the superiority found in the technique mentioned before can not be considered significant. This suggests that it may not be worth using it when compared to the standard Logistic Regression technique, due to its difficulty of interpretation and implementation.

Keywords: Credit Scoring, Machine Learning, Logistic Regression.

Sumário

Lista de Figuras

Lista de Tabelas

1 Introdução

Concessão de crédito pode ser definida como a atividade de colocar um valor à disposição de um tomador de recursos, com o compromisso de pagamento do mesmo em uma data futura (Brito e Neto, 2008). Dessa forma, com o crescimento do mercado e da demanda por crédito, faz-se necessária uma tomada de decisão assertiva para quais clientes conceder-se-á o crédito, uma vez que este processo envolve o risco da contraparte não ser capaz ou ser relutante em fazer o pagamento de suas obrigações contratuais (Gregory, 2012).

O sucesso das instituições financeiras portadoras de crédito está diretamente associado ao controle de risco que utilizam. Segundo Steiner et al. (1999), em apenas uma tomada de decisão feita de forma errônea, pode-se levar a empresa a perder o ganho obtido em dezenas de operações bem sucedidas. Sendo assim, a previsão e controle da inadimplência mostram-se de extrema importância para a longevidade de uma companhia.

Sicsú (2010) revela que existem duas desvantagens quando a análise de crédito é feita por um analista. A primeira delas é a subjetividade: se a mesma solicitação de crédito for submetida a diferentes analistas, os mesmos podem chegar a conclusões bem distintas. A segunda é a ausência de quantificação para o risco das operações. Sendo assim, o uso de técnicas de modelagem mostra-se benéfico para a administração de crédito por diversos fatores, como por exemplo: automatização dos processos, rapidez nas análises, padronização e consistência nas decisões, sem falar na capacidade de administrar e monitorar o risco ao qual a carteira está exposta.

Segundo Thomas (2000), *Credit Scoring* é uma das técnicas de previsão mais importantes na área financeira, tendo como seu principal uso, a previsão de inadimplência. Hand e Henley (1997) definem como um processo que determina a chance dos clientes honrarem o crédito concedido. É comum a divisão dos clientes entre "bons" e "maus", sendo "bons" aqueles que não atrasam, ou atrasam seus pagamentos até um ponto definido como aceitável pela empresa concedente, e "maus" aqueles que não honram suas dúvidas e acarretam algum prejuízo para a empresa.

Além disso, conforme Crook et al. (2007), *Credit Scoring* também é usado para decidir quais consumidores são mais propensos a sanar as dívidas contraídas, quem deve receber uma mala direta e ainda, para identificar potenciais clientes que podem receber um serviço ou produto adicional. Em outras palavras, o modelo de *Credit Scoring* pode ser utilizado para além da concessão, sendo útil como ferramenta para administração do crédito.

Na literatura, não há um consenso sobre qual o melhor método para obter um bom modelo de *Credit Scoring*. Segundo Altman e Saunders (1997), somente após

a década de 70 que as técnicas estatísticas começaram a ser utilizadas para modelagem, e as que dominavam eram a Regressão Logística e a Análise Discriminante. Recentemente, Redes Neurais estão tornando-se também uma alternativa muito popular nas tarefas de análise de crédito (Corrar et al., 2007) e apresentam vantagens em relação às técnicas convencionais (West, 2000; Selau e Ribeiro, 2011).

Com o intuito de superar os métodos estatísticos para análise de crédito, tecnologias de Inteligência Artificial começaram a ser estudadas na década de 90, especialmente o Aprendizado de Máquina. *Artificial Neural Networks* e *Support Vector Machine* são consideradas as duas abordagens de Aprendizado de Máquina mais utilizadas (Zhong et al., 2014). Além disso, ao contrário das técnicas estatísticas tradicionais, os métodos de Aprendizado de Máquina não exigem o conhecimento das relações entre as variáveis de entrada e de saída dos modelos (Aniceto, 2016).

Por se tratar de um processo que envolve um grande volume de dinheiro, melhorias na concessão de crédito precisam ser amplamente discutidas. Sendo assim, o objetivo deste trabalho é comparar o desempenho dos algoritmos de Aprendizado de Máquina para a análise de crédito com relação à abordagem tradicional de Regressão Logística. Dentre os diversos métodos de Aprendizado de Máquina existentes, serão apresentados neste trabalho os seguintes: Árvore de Decisão, *Random Forest, Bagging, Adaboost* e *Support Vector Machine*.

O restante deste trabalho está organizado da seguinte forma: na seção 2, apresenta-se o referencial teórico, o qual relata tecnicamente as abordagens estudadas. Depois, tem-se a metodologia, que apresenta o passo a passo realizado para obtenção dos resultados, os quais serão apresentados na seção seguinte. Por fim, apresenta-se as considerações finais do estudo realizado, bem como sugestões para estudos futuros.

2 Referencial Teórico

Esta seção está dividida em três partes: a primeira apresenta uma revisão bibliográfica sobre o desenvolvimento e utilização de *Credit Scoring*, a segunda relata as abordagens tradicionais utilizadas no desenvolvimento de modelos de *Credit Scoring*, mais especificamente sobre a Regressão Logística, e a terceira mostra os conceitos por trás dos algoritmos de Aprendizado de Máquina que serão utilizados neste trabalho.

2.1 Credit Scoring

Pode-se considerar a análise de crédito como um processo de classificação de clientes entre bons e maus pagadores, onde o principal objetivo é prever se um indivíduo irá se tornar mau pagador, em um determinado período de tempo. A fim de aprimorar este processo, cada vez mais estão sendo utilizadas ferramentas matemáticas e estatísticas, pois, conforme Tsai et al. (2014), uma pequena melhora na precisão da classificação de crédito pode resultar numa grande redução do risco, e além disso, gerar uma significativa economia para a instituição.

Pode-se dividir os modelos de *Credit Scoring* em dois grandes grupos. O primeiro composto por modelos de *Application Scoring*, que objetivam a concessão de crédito para novos clientes e tomam como referência basicamente dados cadastrais. No segundo grupo estão os modelos de *Behavioral Scoring*, utilizados para administrar o crédito daqueles que já são clientes, estes modelos utilizam, além de informações cadastrais, o histórico comportamental do cliente na companhia. Sicsú (2010) relata que este segundo grupo tende a fornecer modelos com maior poder de discriminação entre bons e maus pagadores do que o primeiro, pois incorpora um número maior de variáveis e melhor qualidade de informações. Neste trabalho, optou-se por utilizar um modelo de *Application Scoring*, dado que somente informações cadastrais foram disponibilizadas para análise.

O primeiro método para segregar os grupos foi proposto originalmente por Fisher (1936) em um problema de classificação de variedades de plantas, onde desenvolveu-se os princípios da Análise Discriminante. Durand (1941) aplicou esta metodologia no setor financeiro a fim de distinguir os bons e maus pagadores de um empréstimo. Outra técnica amplamente utilizada é a Regressão Logística, a qual não se sabe ao certo o ano em que foi utilizada pela primeira vez, porém Cox e Snell (1989) e Hosmer e Lemeshow (1989) concordam que a técnica ficou conhecida após o trabalho de Truett et al. (1967), que analisava o risco de doença coronária em um estudo denominado de "*Framingham heart study*".

2.2 Regressão Logística em modelos de Credit Scoring

Decisões são bastante frequentes na vida de um ser humano e, muitas delas, acabam envolvendo dicotomias, ou seja, decidir entre duas opções. Por exemplo, decidir entre ir à praia ou serra, direita ou esquerda, ou ainda, muitas vezes se espera por uma resposta como sucesso ou fracasso em um tratamento médico, a aprovação ou não em um exame (Lopes, 2004).

Ainda, segundo a autora, o modelo de Regressão Logística é útil para expressar a relação entre uma variável dependente (resposta) dicotômica e uma ou mais variáveis independentes, que podem ser quantitativas ou categóricas, pois permite estimar a magnitude e a direção dos efeitos preditores.

Em análise de crédito, a variável resposta determina se um tomador de empréstimo é "bom" ou "mau" pagador. Logo, pode-se definir a probabilidade de um cliente ser "mau" pagador, pelo modelo de Regressão Logística dado por:

$$\pi_i = P(y_i = 1 \mid x_i) = \frac{e^{\beta_0+\beta_1 x_{1i}+\beta_2 x_{2i}+...+\beta_p x_{pi}}}{1 + e^{\beta_0+\beta_1 x_{1i}+\beta_2 x_{2i}+...+\beta_p x_{pi}}}$$

Aplicando a transformação logito, obtém-se um modelo linearizado:

$$ln(\frac{\pi_i}{1-\pi_i}) = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2i} + ... + \beta_p x_{pi}$$

onde $i = 1, 2, .., n$, $y_i \in \{0, 1\}$, x_i são as covariáveis, n é o número total de observações, e os β_j são os parâmetros do modelo e são em geral, estimados via método de máxima verossimilhança. Existem muitos métodos de otimização numérica que podem ser utilizados para encontrar a máxima verossimilhança. Um dos mais antigos e importantes é o método de *Newton-Raphson* (Olivera, 2016).

Bittencourt (2003) relata que a literatura sobre Regressão Logística é bastante vasta e apresentou um crescimento muito rápido. Além das diversas aplicações na área da saúde, esta técnica de modelagem tem sido utilizada também no campo da econometria, administração e educação, possivelmente devido a sua facilidade de execução e interpretação.

2.3 Algoritmos de Aprendizado de Máquina

Nesta seção, será apresentado o raciocínio de cada abordagem de Aprendizado de Máquina utilizada neste trabalho.

2.3.1 Árvore de Decisão

Segundo Lantz (2013), Árvore de Decisão compreende a uma série de decisões lógicas, semelhantes a um fluxograma, com nós de decisão indicando uma decisão a ser tomada em um atributo. Já os ramos, indicam as escolhas de cada decisão.

A construção de uma árvore de classificação binária começa pela identificação da variável independente que melhor segregue a amostra em grupos distintos em relação a variável dependente (Zekic-Susac et al., 2004). Estes modelos utilizam a estratégia de dividir para conquistar: um problema complexo é decomposto em subproblemas

mais simples e recursivamente esta técnica é aplicada a cada subproblema (Gama, 2002). Para o caso da análise de crédito, a primeira divisão ou corte distingue entre bons e maus pagadores, o segundo corte identifica a variável que diferencia bons e maus pagadores, o terceiro corte estabelece outra variável que diferencia o comportamento da variável anterior, e assim sucessivamente (Reis Filho, 2006).

Pela Figura (2.1) é possível visualizar a representação de uma Árvore de Decisão, onde o primeiro nó da árvore é chamado de raiz. Denomina-se de "nó" cada teste de atributo que é feito na árvore, e a sua resposta será uma decisão binária entre "sim" ou "não", a qual denominamos de "folha". A ligação entre os nós ou entre um nó e uma folha é chamada de "aresta".

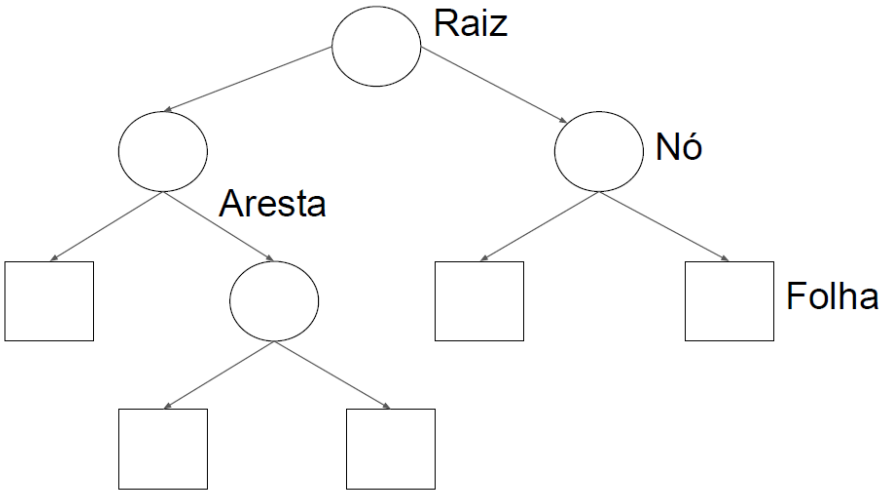

Figura 2.1: Ilustração de uma Árvore de Decisão
Fonte: Reis Filho (2006).

Conforme Silva (2005), o critério utilizado para realizar as partições é o da utilidade do atributo para a classificação em questão. Posteriormente, aplica-se, por este critério, um determinado ganho de informação a cada atributo. A escolha do atributo teste para o corrente nó é aquele que possui o maior ganho de informação e, a partir desta aplicação, inicia-se um novo processo de partição.

Para os casos em que a árvore é usada para classificação, os critérios de partição mais conhecidos são baseados na Entropia e Índice Gini (Onoda, 2001). Segundo Tsai et al. (2014), ao selecionar-se um caso aleatório de um conjunto S de casos e estabelecer que ele pertence a alguma classe C_j, a probabilidade de que uma amostra arbitrária pertence à classe C_j é estimada por:

$$P_i = \frac{freq(C_j, S)}{|S|}$$

onde $|S|$ é o número de amostras no conjunto S e assim as informações que transmite é $-log_2\ p_i$ bits.

E para uma dada distribuição discreta de probabilidade $P = p_1, p_2, ..., p_n$, a informação transmitida por esta distribuição, também chamada de entropia de P, é conhecida como:

$$Info(P) = \sum_{i=1}^{n} -p_i log_2\ p_i$$

Se a partição de um conjunto de amostras T é feita com base no valor de um atributo não-categórico X em conjuntos de $T_1, T_2, ..., T_m$, então a informação necessária para identificar a classe de um elemento de T passa a ser a média ponderada da informação necessária para identificar a classe de um elemento de T_i, ou seja, a média ponderada de $Info(T_i)$.

$$Info(X,T) = \sum_{i=1}^{m} \frac{|T_i|}{T} Info(T_i)$$

Dessa forma, pode-se definir o ganho de informação $Gain(X,T)$ como:

$$Gain(X,T) = Info(T) - Info(X,T)$$

E esta função representa a diferença entre a informação necessária para identificar um elemento de T e a informação necessária para identificar um elemento de T após o valor do atributo X ter sido avaliado. Dessa forma, $Gain(X,T)$ é o ganho de informação devido ao atributo X (Tsai et al., 2014).

Por fim, conforme Silva (2005), pode-se dizer também que a construção de uma Árvore de Decisão baseia-se em três objetivos: diminuição da entropia, consistência em relação ao conjunto de dados e um número pequeno de nós.

2.3.2 *Bagging*

Bagging é considerado um método de classificadores *ensemble* que, segundo Breiman (1996), são classificadores treinados de forma independente por diferentes conjuntos de treinamento por meio de um método de inicialização. Assim, várias Árvores de Decisão são criadas de forma aleatória para posteriormente serem combinadas. A formação dessas árvores é feita por amostragem *bootstrap*: a partir do conjunto de treinamento inicial, subconjuntos são criados de forma aleatória, com reposição.

Cada subconjunto gerado possui o mesmo tamanho (número de exemplos) do conjunto original. Considerando um conjunto de treinamento T com n exemplos, T_k é uma amostra *bootstrap* do conjunto de treinamento a partir de T com reposição, contendo n exemplos. Cada subconjunto T_k é usado para treinar um classificador diferente ($h_{k(x)}$), e a estratégia de combinação dos classificadores é o voto majoritário (Oshiro, 2013). Essa estratégia é simples, mas pode reduzir a variância do classificador final quando combinado com as estratégias de geração de bases de aprendizado (Wang et al., 2011). A Figura (2.2) apresenta uma visualização do funcionamento do método.

Segundo Acuna e Rojas (2001), Breiman (1996) e Freund e Schapire (1996), *Bagging* é muito eficaz quando os classificadores utilizados possuem um comportamento instável (Árvores de Decisão, por exemplo). Um classificador é conhecido como instável quando pequenas mudanças no conjunto de treinamento podem causar grandes mudanças no classificador gerado. Quando isso ocorre, um único classificador instável não é capaz de oferecer uma resposta confiável, ao contrário de um conjunto de classificadores, uma vez que um classificador composto pode ter maior chance de acerto (Lopes, 2007).

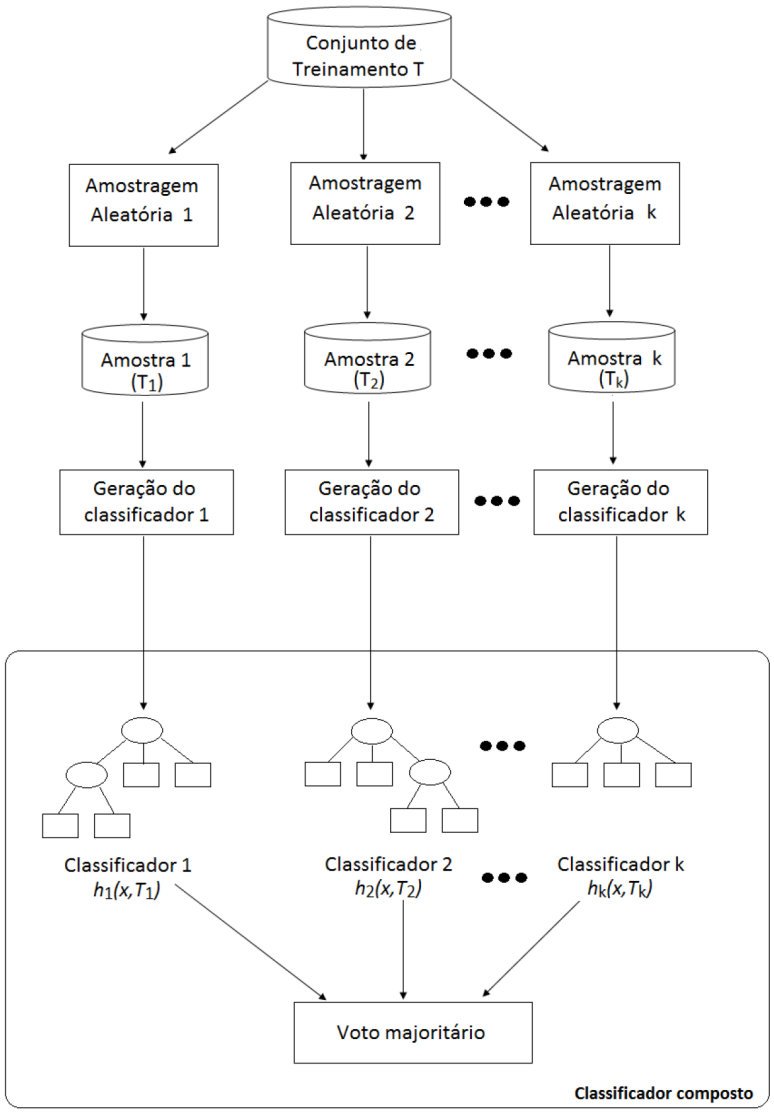

Figura 2.2: Ilustração do funcionamento do classificador *Bagging*
Fonte: Oshiro (2013).

2.3.3 *Random Forest*

Random Forest pode ser definido como uma evolução do modelo de *Bagging* (Breiman, 1996), e também faz parte dos classificadores *ensemble*, que cria diferentes Árvores de Decisão que serão usadas posteriormente na classificação de um novo exemplo por meio do voto majoritário.

Este método, segundo Lantz (2013), baseia-se em um conjunto de Árvores de Decisão, que combina versatilidade e potência em uma abordagem de Aprendizado de Máquina única. O método utiliza apenas uma parte das variáveis independentes disponíveis no conjunto de dados, e as mesmas são selecionadas de forma aleatória para a construção de cada Árvore de Decisão.

Conforme Breiman (2001), pode-se definir o *Random Forest* como um classificador que consiste em uma coleção de árvores classificadoras estruturadas $h(x, \theta_n)$, $n = 1, ..., k$ onde os θ_n são os vetores aleatórios independentes e identicamente distribuídos e, a partir dos dados de entrada x, cada árvore lança um único voto para

a classe mais popular. Tratando-se de crédito, devido as diferentes sub-amostras geradas tanto no *Random Forest*, quanto no *Bagging*, uma das formas de classificar o novo cliente é pela identificação da maioria das classificações obtidas em cada uma das árvores.

Como já mencionado, Oshiro (2013) aponta que *Random Forest* aplica o mesmo método que o *Bagging* para produzir amostras *bootstraps* de conjuntos de treinamento para cada Árvore de Decisão. A única diferença entre os métodos, é que no *Random Forest* as m covariáveis em cada nó das árvores, são selecionadas de forma aleatória e o valor de m é fixo para todos os nós.

Na Figura (2.3) pode-se verificar o funcionamento do algoritmo.

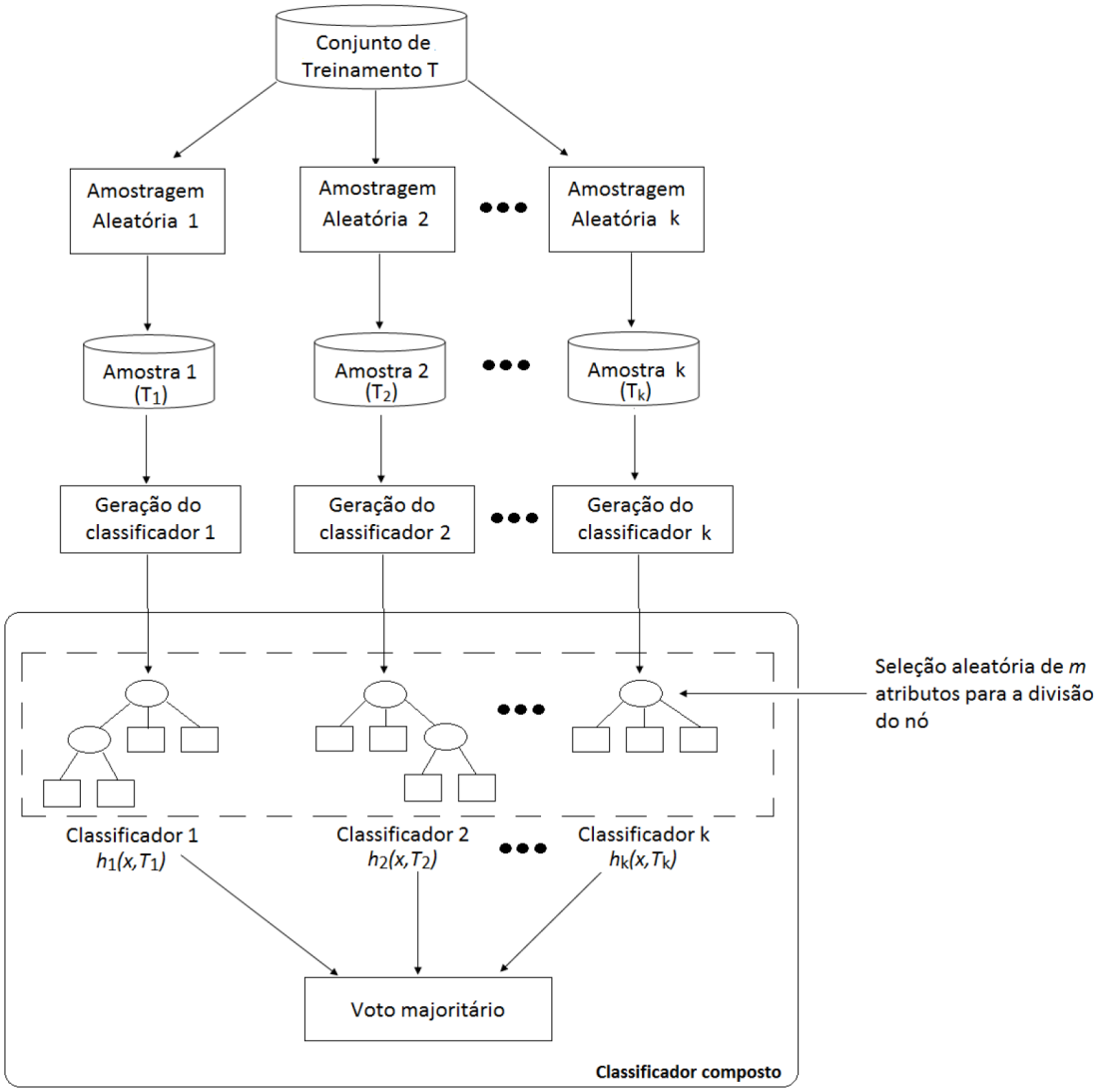

Figura 2.3: Ilustração do funcionamento do classificador *Random Forest*
Fonte: Oshiro (2013).

Ainda, Breiman (2001) justifica o uso de *Bagging* em *Random Forest* pelas seguintes razões: o uso do *Bagging* aparenta melhorar o desempenho quando atributos aleatórios são tomados e também, esta técnica ser usada para fornecer estimativas do erro de generalização do conjunto combinado de árvores. Dessa forma, por gerar diferentes Árvores de Decisão, a aleatorização implica numa correlação baixa entre

as mesmas, o que diminui o erro de classificação em ambos os algoritmos.

2.3.4 *Boosting* e *Adaboost*

O método de *Boosting*, proposto inicialmente por Schapire (1990), é semelhante ao *Bagging*, pois cada classificador é construído com base num conjunto de treinamento diferente. Conforme Lantz (2013), os conjuntos de dados reamostrados em *Boosting* são construídos com o intuito de gerar aprendizados complementares e a importância do voto é ponderada seguindo o desempenho de cada modelo.

Pela capacidade de aumentar o desempenho de um limiar arbitrário com a adição de *learners* mais fracos, o *Boosting* é considerado uma das descobertas mais significativas em Aprendizado de Máquina (Lantz, 2013). Porém, como todo algoritmo de Aprendizado de Máquina, dificuldades de implementação e/ou execução aparecem. Logo, com o intuito de contornar esta situação, o *Adaboost* surgiu. O mesmo foi apresentado pela primeira por Freund e Schapire (1996) e tem originado crescente número de pesquisas e aplicações em várias áreas.

Segundo Tsai et al. (2014), para treinar o $k - ésimo$ classificador como um "classificador fraco", são utilizados para treiná-lo n conjuntos de amostras da base de treinamento ($n < m$) entre S. Após, o classificador treinado é avaliado por S a fim de identificar situações de treinamento que foram classificadas erroneamente, então a árvore $k + 1$ é treinada por um conjunto de treinamento modificado, o que reforça a importância de exemplos classificados incorretamente.

Além disso, segundo os autores citados acima, o procedimento de amostragem é repetido até que k amostras de treinamento sejam criadas para a construção da $k - ésima$ árvore e, portanto, a decisão final é baseada na votação ponderada dos classificadores individuais. O funcionamento do algoritmo pode ser visualizado na Figura (2.4).

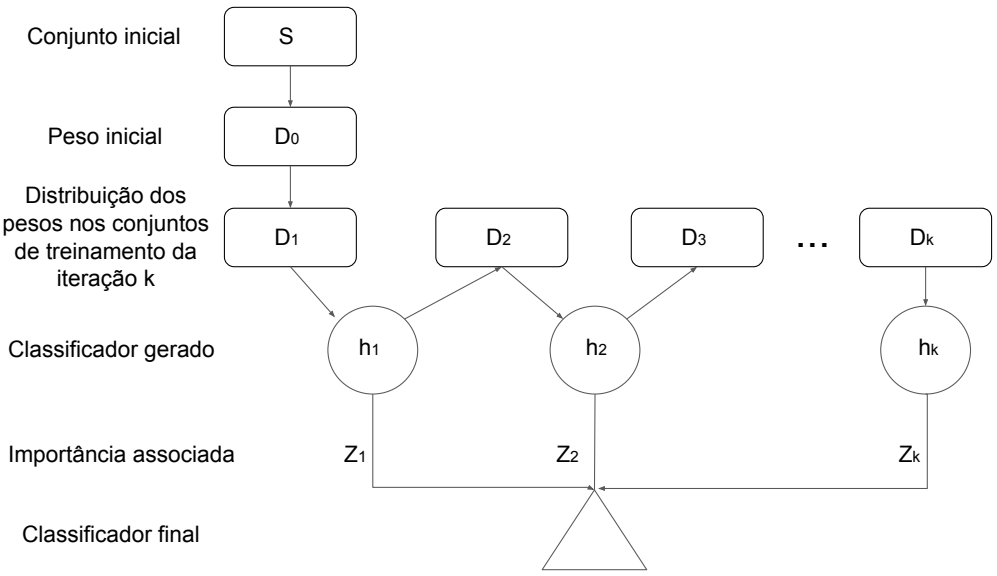

Figura 2.4: Ilustração do funcionamento do classificador *Adaboost*
Fonte: Adaptado de Chaves (2012).

Após a proposta original do *Adaboost* ter sido apresentada aos cientistas e pesquisadores, muitas variações e extensões deste algoritmo foram sugeridas e desen-

volvidas como alternativas para fornecerem melhores resultados para problemas específicos (Chaves, 2012).

2.3.5 *Support Vector Machine*

Uma das estratégias de maior sucesso no equacionamento de problemas de classificação é a denominada Máquina de Vetor Suporte, mais conhecida pela denominação em inglês *Support Vector Machine* (SVM) (Monard e Baranauskas, 2003). Esta técnica é capaz de produzir classificadores com uma boa capacidade de predição de dados não presentes na amostra de treinamento.

O algoritmo de SVM foi proposto por Vapnik (1998) a fim de resolver problemas de classificação binários, tendo sido utilizado com sucesso em aplicações de reconhecimento de padrões, tais como categorização de textos, reconhecimento de caracteres manuscritos, reconhecimento de textura, análise de expressões de genes, reconhecimento de objetos em três dimensões, etc (Rodrigues, 2012).

De forma simplificada, pode-se dizer que SVM é uma técnica de Aprendizado de Máquina capaz de produzir classificadores com a máxima capacidade de generalização e tem como objetivo a construção de uma fronteira plana, chamada de hiperplano. Uma reta é feita com relação a maior distância possível entre as classes do espaço vetorial (-1 e +1), e são criadas com base nos vetores de suporte, que maximizam esta distância.

Porém, há casos em que não é possível traçar uma reta para separar linearmente os dados. Assim, utiliza-se uma função ϕ (conhecida como *kernel*) que mapeia os vetores para uma dimensão de ordem maior, possibilitando assim a sua separação linear (Becker, 2017). O seu funcionamento pode ser visualizado na Figura (2.5).

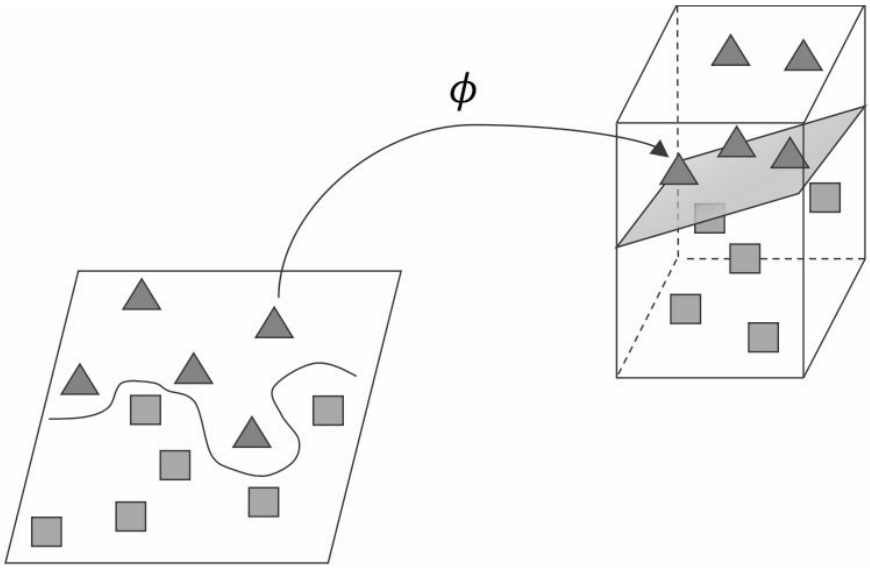

Figura 2.5: Ilustração do funcionamento do classificador *SVM*
Fonte: Becker (2017).

Zhou et al. (2010) relatam que o SVM apresenta grandes benefícios quando empregado em problemas de análise de crédito, o que é o caso deste estudo. Para os autores, pode-se definir 3 principais benefícios desta técnica:

i) Necessita menos concepções sobre os dados, como linearidade e continuidade, por exemplo;

ii) Possibilita a realização de um mapeamento não linear da estrutura dos dados;

iii) Capacidade de implementação de uma estrutura de minimização de risco (SEM – *Structure Risk Minimization*) a partir de algoritmos que buscam aprender o hiperplano que maximiza as margens.

Porém, conforme Silva et al. (2017), uma dificuldade para a aplicação do SVM para a concessão de crédito vem de suas características de estimação de parâmetros, que não são probabilísticos, mas sim medidas de distâncias (lineares ou não lineares). Dessa forma, o cálculo das probabilidades é feito por meio de uma aproximação das distribuições de suas estimativas para alguma distribuição de probabilidades.

3 Metodologia

Este trabalho foi elaborado com base nas etapas de criação de modelos de *Credit Scoring* (Figura (3.1)) sugeridas por Selau (2008). As informações utilizadas são provenientes de uma base de dados real, contendo informações de clientes de uma rede de farmácias com unidades espalhadas pelo Rio Grande do Sul. Esta rede oferece um cartão de crédito para os clientes a fim de facilitar o pagamento de suas compras. Vale salientar ainda, que os dados apresentados neste trabalho foram parcialmente transformados, a fim de preservar o sigilo dos mesmos.

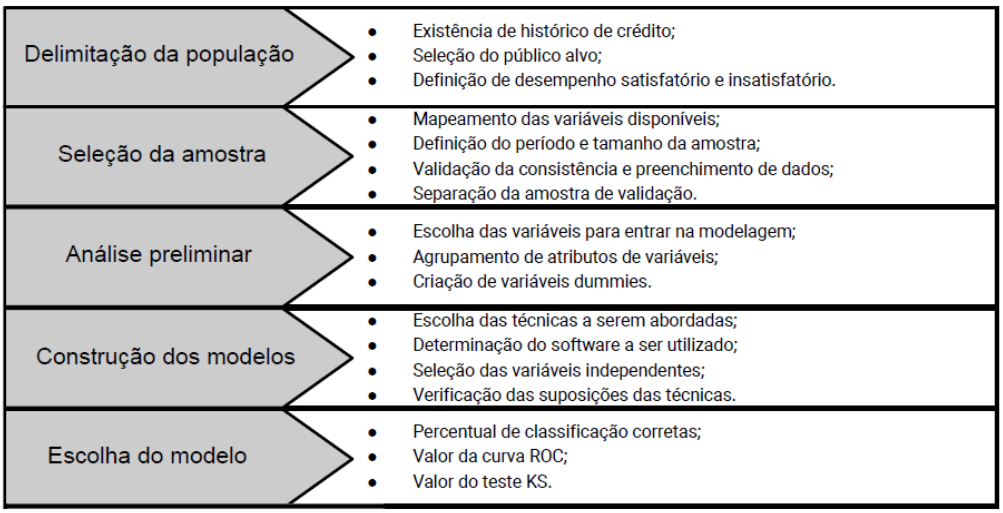

Figura 3.1: Etapas para a construção de um modelo de *Credit Scoring*
Fonte: Adaptado de Selau (2008).

Para a empresa estudada, o cliente bom é definido como aquele que apresenta atraso de até 30 dias, e os maus, são aqueles que apresentam atraso superior a 60 dias. Clientes com atrasos na faixa de 31 a 60 dias são classificados como indefinidos e os mesmos são excluídos do processo de modelagem, a fim de conseguir um maior poder de discriminação entre bons e maus. Além destes três grupos de clientes, há um quarto grupo composto por clientes que ainda não utilizaram o cartão no período do estudo. Este grupo foi denominado de clientes sem uso.

O período analisado compreende a clientes aprovados em dezembro de 2005 a junho de 2006. O total de clientes na amostra é de 11.681, excluindo os casos de clientes classificados como indefinidos ou sem uso.

As informações disponibilizadas pela empresa para a criação dos modelos, dispostas na Figura (3.2), são oriundas da proposta que é preenchida pelos clientes no

momento em que solicitam crédito.

Variável	Descrição
Sexo	Feminino ou masculino
Idade	Idade do cliente no dia do cadastro (em anos)
Estado Civil	Casado, solteiro, divorciado, viúvo, etc
Escolaridade	Fundamental, médio, superior incompleto ou completo
Renda	Valor da renda (R$)
Tipo de Renda	Renda declarada ou comprovada
Profissão	Profissão ou cargo do cliente
Tipo Ocupação	Assalariado, autônomo, profissional liberal, etc
CEP Residencial	CEP do local onde reside
CEP Comercial	CEP do local onde trabalha
Tempo Serviço	Tempo no emprego atual (em meses)
Crédito 3ºs	Tem crédito em outros estabelecimentos?
Tipo Residência	Própria, alugada, cedida ou com pais
Cidade Nascimento	Cidade de naturalidade do cliente
Filho	Tem filhos?
Pensão	Paga pensão alimentícia?

Figura 3.2: Proposta de crédito da empresa (2004).

Analisou-se também, cada variável cadastral disponível no banco de dados, a fim de retirar casos que apresentaram algum erro de digitação, *missing* ou *outlier*. Verificou-se que nas variáveis Idade e Tempo de Serviço, ocorreram alguns valores negativos e também casos com clientes menores de 18 anos, o que demonstra erro de preenchimento no momento do cadastro. Na variável CEP Residencial, notou-se a presença de valores gerais, zerados ou referentes a outros estados. Diante disso, optou-se pela exclusão destes casos, fazendo com que a amostra disponível fosse de 11.389 clientes.

Separou-se de forma aleatória as amostras de treinamento e validação na proporção de 80% e 20%, respectivamente. Esta separação tem por objetivo utilizar uma parcela dos dados para a criação dos modelos, e a outra, para verificar o desempenho dos mesmos em uma amostra diferente. Sendo assim, a amostra de desenvolvimento ficou composta de 5509 clientes bons e 3601 maus, com um total de 9110 clientes. Já a amostra de validação foi formada de 1378 bons e 901 maus, totalizando 2279 clientes.

Primeiramente fez-se uma análise da escolha das variáveis por meio do cálculo do Risco Relativo (RR), onde dividiu-se o percentual de bons clientes pelo percentual de maus para cada um dos atributos. Por meio deste cálculo, excluiu-se duas variáveis do processo de análise (Tipo Renda e Crédito 3ºs), pois ambas apresentaram um baixíssimo poder de discriminação, ou seja, um Risco Relativo próximo de 1. Outra variável excluída neste processo foi Renda, pois não apresentou um comportamento linear quanto ao valor do Risco Relativo. Também, devido à presença de poucos clientes com a característica, a variável com a informação do pagamento de pensão alimentícia foi retirada da análise.

Devido à grande quantidade de atributos das variáveis Profissão, Cidade de Nascimento, CEP Residencial e Comercial, foram criados grupos conforme o seu Risco Relativo. Sendo assim, os que apresentaram resultados próximos foram agrupados, seguindo a escala: péssimo (RR $<$ 0,50); muito mau (RR entre 0,50 e 0,67); mau

(RR entre 0,67 e 0,90); neutro (RR entre 0,90 e 1,10); bom (RR entre 1,10 e 1,50); muito bom (RR entre 1,50 e 2,00) e excelente (RR maior que 2,00). Estipulou-se também a ocorrência de no mínimo 30 observações em cada atributo.

Nas variáveis de CEP Residencial e Comercial, ambas compostas por 8 dígitos, fez-se necessário desagregar a informação em passos. Primeiro, partiu-se os atributos nos dois primeiros, posteriormente em três e em seguida, quatro dígitos. Sempre em busca de uma representação de no mínimo 30 casos em cada um. Por exemplo, caso o total de casos com o CEP comercial inicial 914 for representativo, analisa-se a partição de CEP de 9140 a 9149.

Assim realizou-se o agrupamento das referidas variáveis, criando-se sete grupos, do cliente péssimo ao excelente, embasados pelo seu Risco Relativo. Cada grupo foi transformado em uma variável *dummy* (0 ou 1), as quais serão testadas posteriormente como variáveis independentes na construção dos modelos, assim como as demais variáveis originais.

As relações dos atributos classificados em cada um dos sete grupos para as variáveis profissão, cidade de nascimento, CEP residencial e comercial são apresentados no Anexo I, Anexo II, Anexo III e Anexo IV, respectivamente.

Os resultados da construção dos modelos de previsão de risco de crédito com utilização das técnicas de Árvore de Decisão, *Random Forest, Bagging, Adaboost, Support Vector Machine* e Regressão Logística são apresentados na próxima seção. Todas as seis técnicas de modelagem utilizadas neste trabalho foram construídas no *software R*, versão 3.5.1, e foram aplicadas conforme a lógica descrita na sequência deste texto. A *sintaxe* utlizada está disponível no Anexo V deste trabalho.

O algoritmo de Árvore de Decisão foi implementado pela função *rpart()*, do pacote *rpart* no *software R*, onde criou-se uma árvore a partir de todas as covariáveis disponíveis no banco de desenvolvimento.

A implementação do *Random Forest* deu-se por meio da função *randomForest()*, disponível no pacote de mesmo nome. Definiu-se o número de árvores a serem criadas pelo método em 100, análise feita a partir da visualização gráfica da variação do erro ao longo do número de árvores. Esta análise pode ser feita por meio da função *plot()* com o modelo criado pelo método.

No pacote *ipred* é encontrada a função *bagging()*, utilizada para o desenvolvimento do método de *Bagging*. Já para o *Adaboost*, o comando utilizado foi o *boosting()*, disponível no pacote *adabag*.

O *Support Vector Machine* foi executado via função *svm()*, disponível no pacote *e1071*, e além disso, os parâmetros de ajuste do modelo foram definidos por meio da função *tune()*, a qual encontra os melhores parâmetros para o algoritmo.

Para a Regressão Logística, criou-se um modelo com todas as covariáveis disponíveis, por meio da função *glm()*.

Após a construção de cada modelo a partir da amostra de desenvolvimento, é necessário analisar o comportamento dos escores na amostra de validação, a fim de verificar a adequação dos modelos encontrados. Este processo é importante para evitar que as decisões sejam viciadas e que os modelos sejam bons apenas para dados semelhantes aos quais foram treinados. Dessa forma, espera-se que a distribuição dos escores nesta nova amostra não seja muito diferente da encontrada na amostra de desenvolvimento.

Os critérios de avaliação considerados neste trabalho foram: percentual de acerto geral, valor do Teste KS e área sob a Curva ROC (AUC, do inglês *Area Under the*

Curve). O primeiro indicador, é oriundo de uma matriz de confusão, a qual apresenta um cruzamento das classificações reais *versus* preditas, e pela sua diagonal principal é possível identificar as classificações corretas feitas pelo modelo. A AUC apresenta a sensibilidade e a especificidade do modelo ao mesmo tempo. Esta medida varia de 0 a 1 (ou de 0 a 100%) e é utilizada para descrever a capacidade discriminativa de um modelo, logo quanto maior o seu valor, melhor. O teste KS (Kolmogorov-Smirnov), segundo Selau (2008), é uma técnica não paramétrica utilizada para verificar se duas amostras foram extraídas de uma mesma população. Este teste se baseia na distribuição acumulada dos escores dos clientes classificados como "bons"e "maus", e seu valor é dado pela maior diferença entre estas duas distribuições, logo, quanto maior o seu valor, melhor. Quanto à definição de um valor ideal para este teste, Picinini et al. (2003) revela que, para o mercado financeiro, um bom modelo de *Credit Scoring* é aquele que apresenta um valor de KS igual ou superior a 30.

4 Resultados

Nesta seção são apresentados os resultados dos seis modelos criados conforme cada abordagem utilizada, e ao final dela, é feita uma discussão sobre os resultados encontrados.

4.1 Árvore de Decisão

O algoritmo utilizado para gerar a Árvore de Decisão produz apenas dois subgrupos em cada divisão, ou seja, só são aceitas decisões binárias. Com base nos dados disponíveis na base de desenvolvimento, o algoritmo construiu uma árvore de tamanho igual a 3, o que indica que foram feitas somente 3 decisões para a classificação dos clientes entre bons e maus. A ilustração da árvore construída pelo algoritmo é dada na Figura (4.1). Por meio dela, identifica-se que a variável com maior ganho de informação é o estado civil "solteiro", seguido pelo grupo 2 de cidades de naturalidade dos clientes, e por último, o grupo de idade de até 20 anos.

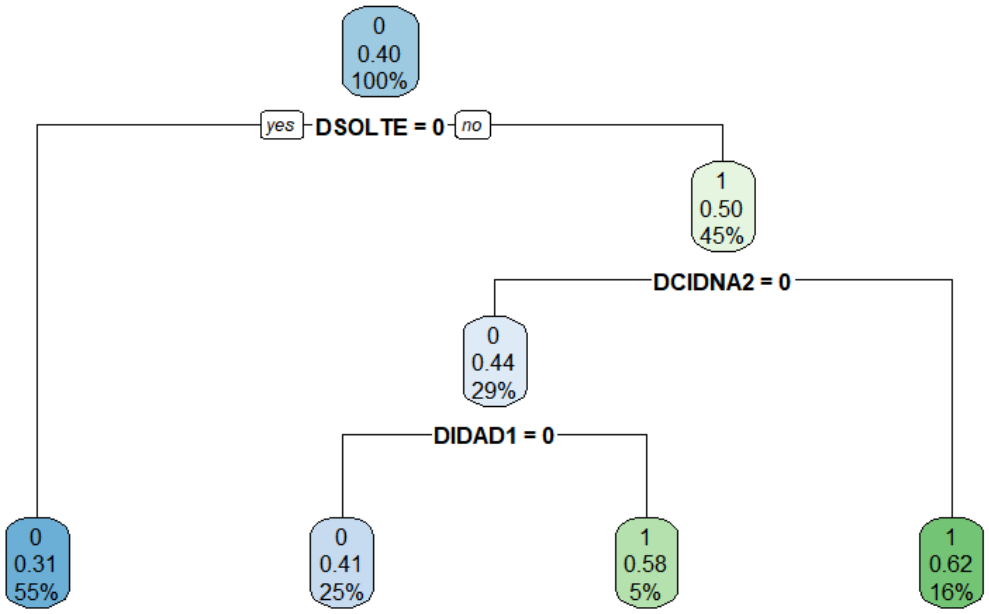

Figura 4.1: Árvore de decisão obtida pelo algoritmo.

Avaliou-se o percentual de classificações corretas dentro do banco de desenvolvimento, o qual atingiu o valor de 66,14% de predições corretas. Posteriormente,

a Árvore de Decisão foi aplicada no conjunto de validação, e resultou na matriz de confusão dada na Tabela (4.1).

Tabela 4.1: Matriz de confusão da amostra de validação - Árvore de Decisão

	Resposta Real		
Resposta Predita	Bom	Mau	Total
Bom	1203	614	1817
Mau	175	287	462
Total	1378	901	2279

Das 2279 observações contidas na amostra de validação, a árvore resultante foi capaz de prever corretamente que 1203 clientes são bons e que 287 são maus, o que resulta num percentual de acerto de 65,38%. Este percentual foi inferior ao encontrado na base de desenvolvimento, o que é esperado, uma vez que o desempenho de um modelo tende a diminuir em dados fora do conjunto de treinamento.

As medidas referentes à AUC e ao teste KS nos dois cenários são dadas na Tabela (4.2) e, por meio desta, é possível identificar que a discriminação do modelo, dada pela AUC, no conjunto de desenvolvimento foi de 64,14% e teve uma leve queda na amostra de validação, atingindo 63,26%. A capacidade de diferenciação da curva de bons pagadores para a de maus é apresentada pelo KS, e o mesmo atingiu 25,30% e 22,27% nos cenários de desenvolvimento e validação, respectivamente.

Tabela 4.2: Indicadores de desempenho - Modelo via Árvore de Decisão

Medida	Desenvolvimento	Validação
AUC	64,14%	63,26%
KS	25,30%	22,27%

4.2 *Random Forest*

Dado que a técnica de *Random Forest* consiste num conjunto de árvores, inicialmente fixou-se em 100 o número de árvores a serem criadas, valor definido a partir da análise da variação do erro ao longo do número de árvores criadas.

A Figura (4.2) apresenta a relação de importância das variáveis dentro do modelo. Por meio dela, é possível notar que o grupo 2 de cidade de naturalidade é a variável que apresenta maior importância na discriminação dos clientes "bons"e "maus"no método citado, seguido pela variável estado civil "solteiro"e assim sucessivamente. Ainda, a variável com menor importância utilizada neste método é o grupo de profissões 7.

Então aplicando o modelo encontrado nos dados utilizados para gerá-lo, o percentual de acerto foi de 80,03%, já na amostra de validação, chegou-se num total de 68,80% de classificações corretas, as quais 1139 foram feitas para bons clientes e 429 para os maus. Estes valores são apresentados na Tabela (4.3).

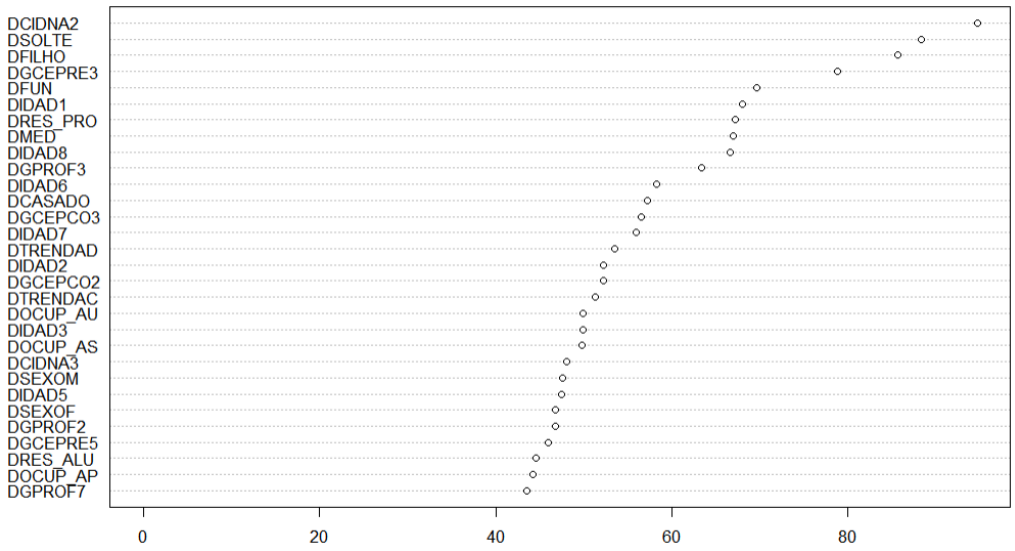

Figura 4.2: Gráfico de importância das variáveis - *Random Forest.*

Tabela 4.3: Matriz de confusão da amostra de validação - *Random Forest*

	Resposta Real		
Resposta Predita	Bom	Mau	Total
Bom	1139	472	1611
Mau	239	429	668
Total	1378	901	2279

As demais medidas de desempenho do modelo nos dois conjuntos de dados são dadas na Tabela (4.4).

Tabela 4.4: Indicadores de desempenho - Modelo via *Random Forest*

Medida	Desenvolvimento	Validação
AUC	80,05%	72,75%
KS	37,91%	35,55%

4.3 *Bagging*

Semelhante à ideia do *Random Forest*, no *Bagging* criaram-se diversas árvores a partir da amostra de desenvolvimento. Pela Figura (4.3), identifica-se que o estado civil "solteiro"é de longe, a variável com maior grau de importância dentro do método.

Em seguida, testou-se o poder de classificação do modelo resultante do algoritmo, onde ele atingiu 65,03% de acertos. Posteriormente aplicou-se o mesmo na amostra de validação, e observou-se que dos 2279 clientes disponíveis, o modelo classificou corretamente 984 bons e 452 maus, alcançando um percentual de acerto de 63,01%, valores que são observados na Tabela (4.5).

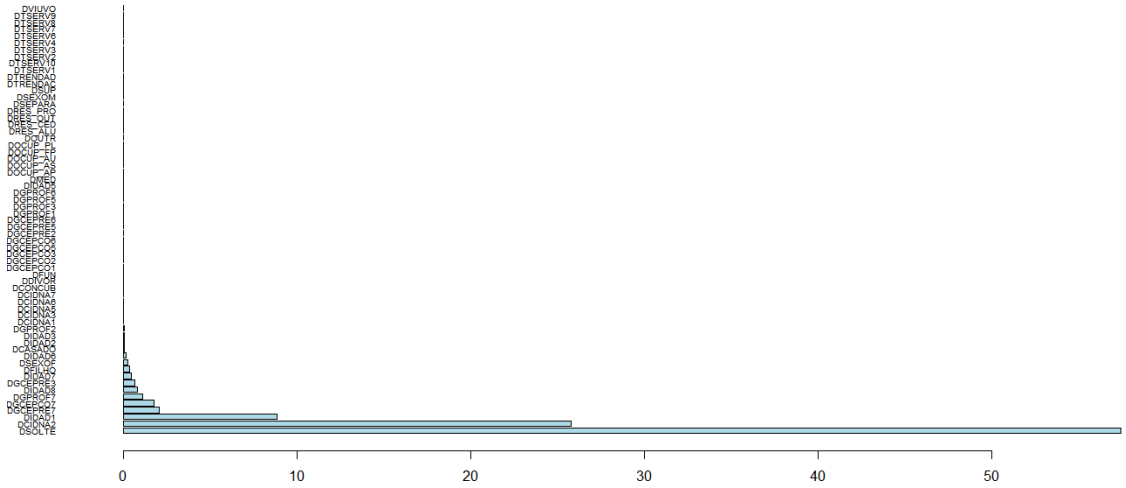

Figura 4.3: Gráfico de importância das variáveis - *Bagging.*

Tabela 4.5: Matriz de confusão da amostra de validação - *Bagging*

	Resposta Real		
Resposta Predita	Bom	Mau	Total
Bom	984	449	1433
Mau	394	452	846
Total	1378	901	2279

As medidas referentes à AUC e ao teste KS são apresentadas na Tabela (4.6), onde nota-se que a precisão do modelo praticamente manteve-se estável nas duas amostras, já a capacidade de diferenciação bons pagadores para os maus diminuiu cerca de 3 pontos percentuais na amostra de validação quando comparada com a de desenvolvimento.

Tabela 4.6: Indicadores de desempenho - Modelo via *Bagging*

Medida	Desenvolvimento	Validação
AUC	66,81%	66,16%
KS	28,47%	25,61%

4.4 *Adaboost*

A partir dos diferentes conjuntos criados por meio da amostra de desenvolvimento, várias árvores são construídas, e a melhor classe para cada conjunto de dados é selecionada e passa a compor o modelo final.

O melhor modelo encontrado via *Adaboost* foi com 100 tentativas. A importância das variáveis dentro do mesmo é apresentada na Figura (4.4), e por meio dela, identifica-se que novamente o estado civil "solteiro"está entre as variáveis mais significativas, ficando atrás somente do grupo de idades superior a 60 anos.

O modelo criado resultou na classificação correta de 70,6% dos 9110 clientes disponíveis no conjunto de desenvolvimento. Na amostra de validação, ocorreu a classificação correta de 1128 clientes bons e 436 maus, atingindo um percentual de acerto de 68,63%, o que pode ser visualizado na Tabela (4.7), e as demais medidas de ajuste do modelo são dadas na Tabela (4.8).

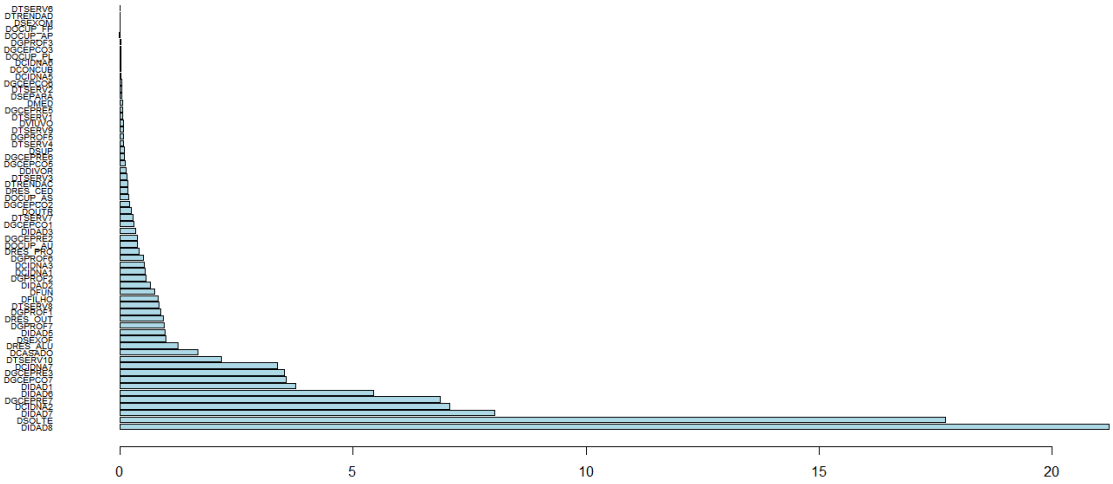

Figura 4.4: Gráfico de importância das variáveis - *Adaboost.*

Tabela 4.7: Matriz de confusão da amostra de validação - *Adaboost*

	Resposta Real		
Resposta Predita	Bom	Mau	Total
Bom	1128	465	1593
Mau	250	436	686
Total	1378	901	2279

Tabela 4.8: Indicadores de desempenho - Modelo via *Adaboost*

Medida	Desenvolvimento	Validação
AUC	78,12%	73,04%
KS	37,05%	36,72%

4.5 *Support Vector Machine*

Com o intuito de buscar o hiperplano que melhor separa os clientes bons dos maus, o modelo de SVM com *kernel* linear foi desenvolvido e utilizou como auxílio 6430 vetores de suporte. Quando avaliado em relação ao percentual de classificações corretas, o modelo apresentou 68,65% de acerto. Já para a amostra de validação, conforme a Tabela (4.9), o modelo alcançou 67,99% de acerto, classificando corretamente 1114 clientes bons e 439 maus.

Tabela 4.9: Matriz de confusão da amostra de validação - *SVM*

	Resposta Real		
Resposta Predita	Bom	Mau	Total
Bom	1114	462	1576
Mau	264	439	703
Total	1378	901	2279

As medidas referentes à AUC e ao teste KS nos dois cenários estudados são dadas na Tabela (4.10), e por meio desta, identifica-se que a precisão do modelo no conjunto de desenvolvimento foi de 72,85% e teve uma queda na amostra de validação, atingindo 64,78%. A capacidade de diferenciação da curva de bons pagadores para a de maus é apresentada pelo KS, e o mesmo manteve-se em torno de 29% em ambos os conjuntos.

Tabela 4.10: Indicadores de desempenho - Modelo via *SVM*

Medida	Desenvolvimento	Validação
AUC	72,85%	64,78%
KS	29,89%	29,57%

4.6 Regressão Logística

A técnica de Regressão Logística é a mais utilizada no ramo de análise de crédito. Dessa forma, criou-se um modelo com todas as variáveis disponíveis a fim de compará-lo com as técnicas de Aprendizado de Máquina. Os indicadores de desempenho do modelo na capacidade de predição foram avaliados na amostra de validação e são apresentados na Tabela (4.11).

Tabela 4.11: Matriz de confusão da amostra de validação - Regressão Logística

	Resposta Real		
Resposta Predita	Bom	Mau	Total
Bom	1114	458	1572
Mau	264	443	707
Total	1378	901	2279

Assim, dos 2279 dados contidos na amostra de validação, o modelo previu corretamente que 443 são inadimplentes e 1114 são adimplentes, o que resulta numa precisão de 68,27%, indicador o qual mantém-se bem próximo do encontrado na amostra de desenvolvimento (68,43%). A Tabela (4.12) apresenta os demais indicadores utilizados para fins de comparação de desempenho, não apresentando grandes diferenças entre os conjuntos testados.

Tabela 4.12: Indicadores de desempenho - Modelo via Regressão Logística

Medida	Desenvolvimento	Validação
AUC	72,89%	72,74%
KS	34,98%	34,86%

4.7 Discussão

A Tabela (4.13) apresenta a comparação dos indicadores de todos os métodos abordados neste trabalho.

Tabela 4.13: Comparação dos Indicadores de desempenho

	Desenvolvimento			Validação		
Método	% acerto	AUC	KS	% acerto	AUC	KS
Árvore de Decisão	66,14%	64,14%	25,30%	65,38%	63,26%	22,27%
Random Forest	80,03%	80,05%	37,91%	68,80%	72,75%	35,55%
Bagging	65,03%	66,81%	28,47%	63,01%	66,16%	25,61%
Adaboost	70,60%	78,12%	37,05%	68,63%	73,04%	36,72%
SVM	68,65%	72,85%	29,89%	67,99%	64,78%	29,57%
Regressão Logística	68,43%	72,89%	34,98%	68,27%	72,74%	34,86%

O percentual de acerto nas classificações é um indicador amplamente utilizado na comparação de modelos, sendo assim, pela Tabela (4.13), é possível identificar que na amostra de desenvolvimento o método que obteve o maior percentual foi o *Random Forest*, seguido pelo *Adaboost* e Regressão Logística. O mesmo comportamento se repete no conjunto de validação. A alta capacidade de classificação correta das observações do algoritmo de *Random Forest* também foi encontrada por De Morais (2017) num trabalho que utilizou a técnica para fazer classificações sensoriais em amostras de arroz.

As técnicas de Árvore de Decisão e *Bagging* obtiveram desempenho inferior à abordagem tradicional de Regressão Logística. Já o *Support Vector Machine*, apesar de demonstrar desempenho inferior, tem um indicador muito próximo do encontrado com o uso da Regressão Logística na amostra de desenvolvimento. Porém, no conjunto de validação, esta diferença entre os dois aumenta consideravelmente a favor da Regressão Logística.

Como discutido anteriormente, a AUC é um critério que mede a discriminação entre as classes estudadas, neste caso, bons e maus clientes. Na amostra de desenvolvimento, novamente o *Random Forest* e o *Adaboost* apresentaram desempenho superior à Regressão Logística e no conjunto de validação, este resultado se manteve, porém com uma diferença menor, o que corrobora com o encontrado por Aniceto (2016) e Dias (2012), que utilizaram a técnica em modelos de crédito.

Segundo Picinini et al. (2003), para modelos de *Credit Scoring* espera-se que a estatística do Teste KS seja maior ou igual a 30 para o modelo ser considerado bom. Diante disto, em ambos os conjuntos de amostras utilizados, os algoritmos de

Árvore de Decisão, *Bagging* e *Support Vector Machine* foram os que apresentaram desempenho inferior ao desejado, sendo este último o que apresentou o valor mais próximo do estabelecido pelos autores.

Novamente na amostra de desenvolvimento, os algoritmos de *Random Forest* e *Adaboost* apresentaram valor do teste KS superior à Regressão Logística. No segundo cenário testado, o *Random Forest* obteve desempenho semelhante ao da Regressão Logística, assumindo um valor de 34,86% quando se trata da maior diferença entre as distribuições de bons e maus pagadores. A técnica que apresentou melhor desempenho em relação ao KS foi a *Adaboost*, nos dois conjuntos amostrais testados.

Na amostra de desenvolvimento, a técnica que apresentou melhor desempenho foi a *Random Forest*. Já no conjunto de validação, este algoritmo perde desempenho e quem assume a primeira colocação é o *Adaboost*. Esta queda de desempenho, segundo Silva (2005), pode ser justificada pelo *overfitting* do modelo nos dados de desenvolvimento, o que é comum de acontecer em algoritmos que utilizam Árvore de Decisão em sua metodologia.

Na literatura, o bom desempenho do *Random Forest* foi observado nos estudos sobre análise de crédito dos autores Brown e Mues (2012), Ali et al. (2012) e Bhattacharyya et al. (2011). Já a boa capacidade de predição do algoritmo *Adaboost* é apontada nos estudos de Randhawa et al. (2018) e Finlay (2011).

5 Considerações Finais

Técnicas de Aprendizado de Máquina, um sub-campo da Inteligência Artificial, têm sido bastante utilizadas na análise de risco de crédito por apresentarem resultados competitivos com as abordagens tradicionais utilizadas. Segundo Huang et al. (2004), diversos pesquisadores têm obtido resultados promissores com o uso de diferentes abordagens de Inteligência Artificial e técnicas estatísticas.

Este trabalho teve como propósito principal comparar o desempenho de diferentes métodos de Aprendizado de Máquina por meio de um estudo empírico em um banco de dados real utilizado na concessão de crédito. As técnicas discutidas no trabalho foram Árvore de Decisão, *Random Forest, Bagging, Adaboost* e *Support Vector Machine*. Utilizou-se também a Regressão Logística para fins de comparação dos modelos.

Conforme o estudo empírico, algumas técnicas de Aprendizado de Máquina apresentaram desempenho superior à abordagem tradicional de Regressão Logística. *Random Forest* e *Adaboost* apresentaram um desempenho superior à Regressão nos dois cenários testados, e em todas as métricas estudadas: percentual de acerto, AUC e KS.

Os resultados deste trabalho corroboram com outros já realizados sobre análise de crédito, dessa forma, pode-se concluir que a utilização de algumas técnicas de Aprendizado de Máquina em modelos de *Credit Scoring* pode gerar um maior poder de predição quando comparado a técnicas tradicionais.

Porém, além do menor esforço computacional para o desenvolvimento do modelo de Regressão Logística, outra desvantagem que os métodos da Inteligência Artificial apresentam é a dificuldade de implementação dos algoritmos e interpretação dos parâmetros do modelo, sendo questionáveis ganhos de *performance* que não sejam significativamente superior.

Como sugestão de trabalhos futuros, indica-se o estudo do uso de podas em alguns algoritmos de Aprendizado de Máquina, a fim de verificar se há uma melhora de desempenho. Além disso, é interessante analisar o custo de cada classificação errada por meio de alguma medida monetária, como forma de melhorar a avaliação e comparação de desempenho dos modelos construídos. A verificação do desempenho das técnicas estudadas neste trabalho em um conjunto de dados com diferentes variáveis também pode ser interessante, bem como a combinação de algumas técnicas a fim de obter uma melhor predição.

Referências Bibliográficas

Acuna, E. e Rojas, A. (2001). Bagging classifiers based on kernel density estimators. In *Proceedings of the International Conference on New Trends in Computational Statistics with Biomedical Applications*, pages 343–350.

Ali, J., Khan, R., Ahmad, N., e Maqsood, I. (2012). Random forests and decision trees. *International Journal of Computer Science Issues (IJCSI)*, 9(5):272.

Altman, E. I. e Saunders, A. (1997). Credit risk measurement: Developments over the last 20 years. *Journal of banking & finance*, 21(11-12):1721–1742.

Aniceto, M. C. (2016). Estudo comparativo entre técnicas de aprendizado de máquina para estimação de risco de crédito. Master's thesis, Universidade de Brasília.

Becker, W. E. (2017). Uma abordagem de redes neurais convolucionais para análise de sentimento multi-lingual. Master's thesis, Pontifícia Universidade Católica do Rio Grande do Sul.

Bhattacharyya, S., Jha, S., Tharakunnel, K., e Westland, J. C. (2011). Data mining for credit card fraud: A comparative study. *Decision Support Systems*, 50(3):602–613.

Bittencourt, H. R. (2003). Regressão logística politômica: revisão teórica e aplicações. *Acta Scientiae*, 5(1):77–86.

Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2):123–140.

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32.

Brito, G. A. S. e Neto, A. A. (2008). Modelo de classificação de risco de crédito de empresas. *Revista Contabilidade & Finanças*, 19(46):18–29.

Brown, I. e Mues, C. (2012). An experimental comparison of classification algorithms for imbalanced credit scoring data sets. *Expert Systems with Applications*, 39(3):3446–3453.

Chaves, B. B. (2012). *Estudo do algoritmo AdaBoost de aprendizagem de máquina aplicado a sensores e sistemas embarcados*. Master's thesis, Universidade de São Paulo.

Corrar, L. J., Paulo, E., e Dias Filho, J. M. (2007). Análise multivariada: para cursos de administração, ciências contábeis, atuariais e financeiras. *São Paulo: Atlas*.

Cox, D. R. e Snell, E. J. (1989). Analysis of binary data (vol. 32). *Monographs on Statistics and Applied Probability*.

Crook, J. N., Edelman, D. B., e Thomas, L. C. (2007). Recent developments in consumer credit risk assessment. *European Journal of Operational Research*, 183(3):1447–1465.

De Morais, R. L. (2017). Uso de árvores aleatórias para classificação sensorial de arroz cozido. Trabalho de Conclusão de Curso, Universidade de Brasília.

Dias, A. A. D. (2012). Previsão do incumprimento no crédito a empresas com classificadores múltiplos. Master's thesis, Universidade Técnica de Lisboa.

Durand, D. (1941). *Risk elements in consumer installment financing*. National Bureau of Economic Research, New York.

Finlay, S. (2011). Multiple classifier architectures and their application to credit risk assessment. *European Journal of Operational Research*, 210(2):368–378.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188.

Freund, Y. e Schapire, R. E. (1996). Experiments with a new boosting algorithm. In *Icml*, volume 96, pages 148–156. Citeseer.

Gama, J. (2002). Árvores de decisão. *Palestra ministrada no Núcleo da Ciência de Computação da Universidade do Porto*, Porto.

Gregory, J. (2012). *Counterparty credit risk and credit value adjustment: A continuing challenge for global financial markets*. John Wiley & Sons.

Hand, D. J. e Henley, W. E. (1997). Statistical classification methods in consumer credit scoring: a review. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 160(3):523–541.

Hosmer, D. e Lemeshow, S. (1989). Applied logistic regression. 1989. *New York: Johns Wiley & Sons*.

Huang, Z., Chen, H., Hsu, C.-J., Chen, W.-H., e Wu, S. (2004). Credit rating analysis with support vector machines and neural networks: a market comparative study. *Decision support systems*, 37(4):543–558.

Lantz, B. (2013). *Machine learning with R*. Packt Publishing Ltd.

Lopes, L. (2007). Aprendizagem de máquina baseada na combinação de classificadores em bases de dados da área da saúde. Master's thesis, Pontifícia Universidade Católica do Paraná.

Lopes, L. d. S. (2004). Definição de um modelo de cobrança (collection score) utilizando regressão logística multinomial. Trabalho de Conclusão de Curso, Universidade Federal do Rio Grande do Sul.

Monard, M. C. e Baranauskas, J. A. (2003). Conceitos sobre aprendizado de máquina. *Sistemas Inteligentes-Fundamentos e Aplicações*, 1(1):32.

Olivera, A. R. (2016). Comparação de algoritmos de aprendizagem de máquina para construção de modelos preditivos de diabetes não diagnosticado. Master's thesis, Pontífica Universidade Católica do Rio Grande do Sul.

Onoda, M. (2001). Estudo sobre um algoritmo de árvores de decisão acoplado a um sistema de banco de dados relacional. 2001. 110p. Master's thesis, Universidade Federal do Rio de Janeiro.

Oshiro, T. M. (2013). *Uma abordagem para a construção de uma única árvore a partir de uma Random Forest para classificação de bases de expressão gênica*. PhD thesis, Universidade de São Paulo.

Picinini, R., Oliveira, G., e Monteiro, L. (2003). Mineração de critério de credit scoring utilizando algoritmos genéticos. In *Simpósio Brasileiro de Automação Inteligente*, volume 6.

Randhawa, K., Loo, C. K., Seera, M., Lim, C. P., e Nandi, A. K. (2018). Credit card fraud detection using adaboost and majority voting. *IEEE ACCESS*, 6:14277–14284.

Reis Filho, J. (2006). Sistema inteligente baseado em árvore de decisão, para apoio ao combate às perdas comerciais na distribuição de energia elétrica. Master's thesis, Universidade Federal de Uberlândia.

Rodrigues, F. A. A. (2012). Modelo de análise de risco de crédito utilizando máquina de vetor suporte. Master's thesis, Universidade Federal do Rio de Janeiro.

Schapire, R. E. (1990). The strength of weak learnability. *Machine learning*, 5(2):197–227.

Selau, L. P. R. (2008). Construção de modelos de previsão de risco de crédito. Master's thesis, Universidade Federal do Rio Grande do Sul.

Selau, L. P. R. e Ribeiro, J. L. D. (2011). A systematic approach to construct credit risk forecast models. *Pesquisa Operacional*, 31(1):41–56.

Sicsú, A. L. (2010). *Credit Scoring: desenvolvimento, implantação, acompanhamento*. Blucher.

Silva, L. M. (2005). *Uma aplicação de Árvores de Decisão, Redes Neurais e KNN para a Identificação de Modelos ARMA não Sazonais e Sazonais*. PhD thesis, Pontifícia Universidade Católica do Rio de Janeiro.

Silva, R. A., Ara, A., e Ribeiro, E. M. S. (2017). Desempenho financeiro como critério de decisão para seleção modelos de análise e concessão de crédito. *RECADM*, 16(1):25–39.

Steiner, M. T. A., Carnieri, C., Kopittke, B. H., e Neto, P. J. S. (1999). Sistemas especialistas probabilísticos e redes neurais na análise do crédito bancário. *Revista de Administração da Universidade de São Paulo*, 34(3).

Thomas, L. C. (2000). A survey of credit and behavioural scoring: forecasting financial risk of lending to consumers. *International journal of forecasting*, 16(2):149–172.

Truett, J., Cornfield, J., e Kannel, W. (1967). A multivariate analysis of the risk of coronary heart disease in framingham. *Journal of chronic diseases*, 20(7):511–524.

Tsai, C.-F., Hsu, Y.-F., e Yen, D. C. (2014). A comparative study of classifier ensembles for bankruptcy prediction. *Applied Soft Computing*, 24:977–984.

Vapnik, V. (1998). *Statistical learning theory. 1998*, volume 3. Wiley, New York.

Wang, G., Hao, J., Ma, J., e Jiang, H. (2011). A comparative assessment of ensemble learning for credit scoring. *Expert systems with applications*, 38(1):223–230.

West, D. (2000). Neural network credit scoring models. *Computers & Operations Research*, 27(11-12):1131–1152.

Zekic-Susac, M., Sarlija, N., e Benšić, M. (2004). Small business credit scoring: a comparison of logistic regression, neural network, and decision tree models. In *26th International Conference on Information Technology Interfaces*, pages 265 – 270.

Zhong, H., Miao, C., Shen, Z., e Feng, Y. (2014). Comparing the learning effectiveness of bp, elm, i-elm, and svm for corporate credit ratings. *Neurocomputing*, 128:285–295.

Zhou, L., Lai, K. K., e Yu, L. (2010). Least squares support vector machines ensemble models for credit scoring. *Expert Systems with Applications*, 37(1):127–133.

6 Anexo I - Agrupamento de profissões

Péssimo Desempenho	BABA	PROMOTOR VENDAS
	COZINHEIRO	ALMOXARIFE
	PINTOR	
Muito Mau Desempenho	AUX PRODUCAO	PEDREIRO
	CABELEIREIRO	PORTEIRO
	CONFEITEIRO	RECEPCIONISTA
	GERENTE	VENDEDOR
	PADEIRO	
Mau Desempenho	AUTONOMO	MANICURE
	AUX ADMINISTRATIVO	MECANICO
	AUX COZINHA	TEC ENFERMAGEM
	AUX SERVICOS GERAIS	VIGILANTE
	COMERCIANTE	
Desempenho Neutro	ATENDENTE	INDUSTRIARIO
	COMERCIARIO	MOTORISTA
	DOMESTICA	
Bom Desempenho	CAIXA	SECRETARIA
	DO LAR	SERVENTE
	PENSIONISTA	
Muito Bom Desempenho	AGRICULTOR	OPERADOR
	BALCONISTA	METALUGICO
	COSTUREIRO	AUX ENFERMAGEM
	DIARISTA	
Excelente Desempenho	APOSENTADO	PROFESSOR

7 Anexo II - Agrupamento de cidades de nascimento

Péssimo Desempenho	ALVORADA	
Muito Mau Desempenho	CRUZ ALTA	RIO GRANDE
	ESTEIO	TRAMANDAI
	PORTO ALEGRE	
Mau Desempenho	CANOAS	SAO BORJA
	GRAVATAI	SAO GABRIEL
	IJUI	SAPIRANGA
	NOVO HAMBURGO	SAPUCAIA DO SUL
	PELOTAS	URUGUAIANA
Desempenho Neutro	ALEGRETE	SANTO ANGELO
	CAMAQUA	SAO FRANCISCO DE PAULA
	CANELA	SAO LOURENCO DO SUL
	SANTANA DO LIVRAMENTO	VIAMAO
Bom Desempenho	BAGE	PASSO FUNDO
	BUTIA	RIO PARDO
	CACAPAVA DO SUL	SANTA CRUZ DO SUL
	GUAIBA	SAO JERONIMO
	MONTENEGRO	SAO LEOPOLDO
	OSORIO	SAO LUIZ GONZAGA
Muito Bom Desempenho	CACHOEIRA DO SUL	SANTA ROSA
	CAXIAS DO SUL	SANTA VITORIA DO PALMAR
	PALMEIRA DAS MISSOES	TAQUARA
	SANTA MARIA	
Excelente Desempenho	CANGUCU	SAO SEPE
	ENCRUZILHADA DO SUL	TAPES
	GIRUA	TORRES
	HORIZONTINA	TRES DE MAIO
	ROLANTE	TRIUNFO
	SANTO ANTONIO DA PATRULHA	

8 Anexo III - Agrupamento de CEP residencial

	2 PRIMEIRAS POSIÇÕES	3 PRIMEIRAS POSIÇÕES	4 PRIMEIRAS POSIÇÕES
Péssimo Desempenho			**9670** SÃO JERÔNIMO
Muito Mau Desempenho			**9175** Aberta dos Morros - POA **9179** Restinga - POA **9191** Camaquã - POA **9192** Cavalhada/Camaquã - POA **9440** Águas Claras - VIAM **9449** NS Aparecida/Pq Índio Jari - VIAM **9481** Formozo/Passo Feijó - ALVO **9493** Dist Industrial/Cohab - CACH
Mau Desempenho		**902** Farrapos/Navegantes/Humaitá - POA **906** Partenon/Jardim Botânico - POA **912** Protásio Alves/Rubem Berta - POA **915** Lomba do Pinheiro/Agronomia - POA **923** Mathias Velho/Harmonia - CANO **934** Lomba Grande/Santo Afonso - NH **941** GRAVATAÍ **945** Vila Augusta/Jd Universit. - VIAM	**9117** Rubem Berta - POA **9172** Nonoai/Teresópolis - POA **9174** Cavalhada/Vila Nova - POA **9190** Tristeza/Vila Assunção - POA **9326** Centro/Vila Teópolis - EST **9329** Pq Primavera/Pq St Inácio - EST **9353** São Jorge/Vila Diehl - NH **9400** GRAVATAÍ **9441** Centro/Tarumã - VIAM **9442** Vila Elsa/Estalagem - VIAM **9444** Jd Krahe/St Onofre - VIAM **9482** Maria Regina/Sumaré - ALVO **9483** Tijuca/Piratini - ALVO **9485** Aparecida/Jd Algarve - ALVO **9490** Jardim América/Vila City - CACH **9483** Tijuca/Piratini - ALVO **9485** Aparecida/Jd Algarve - ALVO **9490** Jardim América/Vila City - CACH **9607** Porto/Três Vendas - PEL **9618** CAMAQUÃ **9750** URUGUAIANA
Desempenho Neutro		**908** Santa Tereza/Medianeira - POA **913** Vila Jardim/Vila Ipiranga - POA **914** Protásio Alves/Jardim Carvalho - POA	**9332** Industrial/Ouro Branco - NH **9445** São Lucas/Florescente - VIAM **9447** St Cecília/Viamópolis - VIAM **9480** Maringá/Sumaré - ALVO **9494** Vila Vista Alegre - CACH **9495** Vila Bom Princípio/Pq Matriz - CACH
Bom Desempenho		**922** Fátima/Rio Branco - CANO **924** Igara/São José/Guajuviras - CANO **925** GUAÍBA **930** SÃO LEOPOLDO **938** NOVA HARTZ, SAPIRANGA **955** OSÓRIO, CAPÃO DA CANOA **956** TAQUARA, CANELA, GRAMADO **957** BENTO GONÇALVES, GARIBALDI	**9178** Lami/Belém Novo - POA **9328** Vila Esperança/Pq Amador - EST **9330** Centro - NH **9333** Liberdade/Ideal - NH **9334** Primavera/Petrópolis - NH **9354** Canudos/Mauá - NH **9443** Jardim Krahe/Sítio S.José - VIAM **9496** Pq Granja Esperança - CACH **9601** Centro - PEL **9617** SÃO LOURENÇO DO SUL **9674** ARROIO RATOS e CHARQUEADAS **9754** ALEGRETE
Muito Bom Desempenho	**99** PASSO FUNDO	**950** CAXIAS DO SUL **962** RIO GRANDE, STA VITÓRIA PALMAR **965** CACHOEIRA DO SUL, CAÇAPAVA **970** SANTA MARIA	**9407** GRAVATAÍ
Excelente Desempenho	**98** CRUZ ALTA	**937** CAMPO BOM **958** ESTRELA, TAQUARI, VENÂNCIO AIRES **964** BAGÉ, DOM PEDRITO **966** RIO PARDO, PÂNTANO GRANDE **968** SANTA CRUZ DO SUL **971** SANTA MARIA, ITAARA **973** SÃO GABRIEL, LAVRAS DO SUL	**9352** Guarani/Vila Nova - NH

9 Anexo IV - Agrupamento de CEP comercial

	2 PRIMEIRAS POSIÇÕES	3 PRIMEIRAS POSIÇÕES	4 PRIMEIRAS POSIÇÕES
Péssimo Desempenho		**912** Protásio Alves/Rubem Berta - POA	**9670** SÃO JERÔNIMO
Muito Mau Desempenho		**900** Centro/Farroupilha/Bom Fim - POA **906** Partenon/Jardim Botânico - POA **932** ESTEIO, SAPUCAIA DO SUL **948** ALVORADA	**9174** Cavalhada/Vila Nova - POA **9190** Tristeza/Vila Assunção - POA **9192** Cavalhada/Camaquã - POA
Mau Desempenho		**901** Azenha/Menino Deus/Praia Belas - POA **902** Farrapos/Navegantes/Humaitá - POA **908** Santa Tereza/Medianeira - POA **910** Passo D'Areia/Jardim Lindóia - POA **911** Sarandi/Rubem Berta - POA **913** Vila Jardim/Vila Ipiranga - POA **915** Lomba do Pinheiro/Agronomia - POA **933** Rio Branco/Primavera/Industrial - NH **940** GRAVATAÍ **949** CACHOEIRINHA **961** CAMAQUÃ, CAPÃO DO LEÃO	**9175** Aberta dos Morros - POA **9179** Restinga - POA **9191** Camaquã - POA **9351** Centro/Hamburgo Velho - NH **9353** São Jorge/Vila Diehl - NH **9602** Fragata/Três Vendas - PEL
Desempenho Neutro		**904** Auxiliadora/Petrópolis - POA	**9380** SAPIRANGA
Bom Desempenho		**905** São João/Floresta/Higienópolis - POA **925** GUAÍBA **934** Lomba Grande/Santo Afonso - NH **955** OSÓRIO, CAPÃO DA CANOA **956** TAQUARA, CANELA, GRAMADO **957** BENTO GONÇALVES, GARIBALDI	**9178** Lami/Belém Novo - POA **9389** NOVA HARTZ **9441** Centro/Tarumã - VIAM **9601** Centro - PEL **9674** ARROIO RATOS e CHARQUEADAS
Muito Bom Desempenho	**99** PASSO FUNDO	**959** LAJEADO, ENCANTADO, PROGRESSO **962** RIO GRANDE, STA VITÓRIA PALMAR	
Excelente Desempenho	**97** SANTA MARIA **98** CRUZ ALTA	**937** CAMPO BOM **950** CAXIAS DO SUL **958** ESTRELA, TAQUARI, VENÂNCIO AIRES **964** BAGÉ, DOM PEDRITO **965** CACHOEIRA DO SUL, CAÇAPAVA **966** RIO PARDO, PÂNTANO GRANDE **968** SANTA CRUZ DO SUL	

10 Anexo V - *Sintaxe* utilizada

```
    #ARVORE DE DECISAO
arvore = rpart(resp ~ . , method="class", data = treino)
rpart.plot(arvore)

    #RANDOM FOREST
rf = randomForest(resp ~ . , method="class", data = treino, ntree = 100)
varImpPlot(rf)

    #BAGGING
bag = bagging(resp ~ . , data = treino, probability=TRUE, coob=TRUE)
importanceplot(bag)

    #ADABOOST
adab = boosting(resp ~ . , data=treino, boos=TRUE, mfinal=100, coeflearn='Breiman')
importanceplot(adab)

    #SUPPORT VECTOR MACHINE
tune_svm = tune(svm, resp ~ . , data = treino, ranges = list(gamma = 2 ^ (-1:1),
cost = 2 ^ (2:4)))
svm = svm(resp ~ . , data = treino, na.action=na.omit(treino), kernel="radial",
cost=4, gamma=0.5)

    #REGRESSAO LOGISTICA
reg = glm(resp ~ . , data = treino, family = binomial)


    #Aplicando o modelo na amostra de validação
fit.ar = predict(arvore, newdata = validacao, type = "prob")

    #Matriz de confusão
fit.ar = predict(arvore, newdata = validacao, type = "class")
confusion.matrix = table(pred = fit.ar, true = validacao$resp)

    #Percentual de acerto
acerto.ar = (confusion.matrix[1,1] + confusion.matrix[2,2])/(confusion.matrix[1,1]
+ confusion.matrix[1,2] + confusion.matrix[2,1] + confusion.matrix[2,2])
```

```
    #AUC
pred = prediction(fit.ar[, 2], validacao$resp)
auc.ar = performance(pred, 'auc')
slot(auc.ar, 'y.values')

    #Teste KS
ks.ar = max(attr(performance(pred, "tpr", "fpr"), "y.values")[[1]] - (attr(performance(pred,
"tpr", "fpr"), "x.values")[[1]]))
```