ROBSON SESTREM

PROCESSAMENTO DE DADOS COM ALGORITMO FLORESTA ALEATÓRIA:

Uma pesquisa de seu comportamento para prever inadimplência

Augusto Pestana

Março/2020

Resumo

O presente trabalho possui o objetivo de construir um classificador por meio de técnicas de Aprendizado de Máquina que se utiliza do algoritmo de floresta aleatória para prever quantos clientes de uma instituição financeira serão inadimplentes no próximo pagamento. O Aprendizado de Máquina se tornou uma ferramenta fundamental para algumas áreas do conhecimento e sua contribuição nas finanças deve ganhar espaço considerável como solução para prever inadimplência. Como cumprimento da metodologia de cunho mais descritivo foram aplicadas atividades de coleta e processamento de dados realizadas com o ambiente do RStudio para desenvolvimento do modelo preditivo. Afim de validar o modelo criado como solução extraiu-se informações relevantes de grau de importância das variáveis, quantidades de acertos na previsão de bons e maus pagadores, acurácia entre outras métricas. Os resultados se mostraram positivos quanto a precisão geral do modelo, cerca de 81,67% provando sua boa capacidade de classificação para diferentes conjuntos de dados, porém houve uma baixa taxa de sensibilidade de 37,93% para o conjunto de teste, dessa forma foi considerado que dependerá do ponto de vista da empresa validar os resultados como uma solução completa do problema, pois apesar da pouca abrangência na exploração do relacionamento das variáveis do *dataset* a pesquisa demonstrou qualidade na capacidade de generalização de hipótese e proporciona uma futura exploração mais focada na fase de pré-processamento.

Palavras-Chave: machine learning; random forest; inadimplência

1. INTRODUÇÃO

Com o passar dos anos a tecnologia tem nos beneficiado cada vez mais em tarefas complexas, as quais se tem a necessidade de extrair informações que agreguem valor para os mais diversos setores da sociedade, como na indústria automotiva, agroindústria, construção civil, indústria farmacêutica, transporte e logística, entre outros. Na Tecnologia da Informação (TI) o uso de algoritmos em *softwares* como ferramentas para análise e processamento de dados de forma

automática e inteligente, tem demonstrado cada vez mais seu valor prático, pois proporciona soluções nunca antes vista em grandes volumes de dados. Algumas destas soluções estão relacionadas com o termo big data, que pode ser descrito como técnicas de processar estes dados a fim de apresentar informação que venha resolver problemas nos quais a programação geral não se adequa (MUELLER; MASSARON, 2019).

A exemplo de proveito destas soluções, temos a empresa americana de logística United Parcel Service Inc. (UPS), que cruzou dados dos sensores de sua frota, mapas, geolocalização e clientes para reduzir as distâncias percorridas e conseguiu economizar 85 milhões de milhas por ano com a tecnologia de big data. Outro exemplo com a utilização de big data foi com o Grupo Pão de Açúcar, que identificou os produtos preferidos dos consumidores através de seus programas de recompensas, e com essa informação coletada, foi possível personalizar as ofertas geradas aos seu clientes, como resultado disso houve expressivo aumento de fidelização e os dados ainda favoreceram uma melhor gestão de estoque com redução de custos (IBE, 2018).

Temos a disposição o Aprendizado de Máquina (*Machine Learning*, em inglês) que de acordo com Gama et al. (2011), é uma área da Inteligência Artificial (IA) que trabalha com o aprendizado computacional para automatizar um conhecimento adquirido, a qual é uma forma de otimizar uma tarefa através de uma experiencia passada, utilizando meios programados em computadores. No Aprendizado de Máquina (AM) existe o aprendizado supervisionado como método preditivo, neste tipo de conhecimento pode-se estimar valores contidos numa coleção finita de dados, uma das formas de alcançar este resultado é com a tarefa de classificação, onde o que se quer predizer é uma classe desconhecida. É nesta tarefa de classificação que os algoritmos utilizados são capazes de identificar padrões de uma classe por meio de dados históricos e utilizar esse padrão para prever valores ainda desconhecidos desta classe em novas coleções de dados

(MUELLER; MASSARON, 2019). O uso de algoritmos em AM traz soluções bem-sucedidas para as mais diversas áreas, inclusive em sistemas do setor financeiro onde por exemplo pode-se identificar fraude de cartões de crédito.

Neste cenário financeiro existe um problema comum para diversas instituições e empresas que é a inadimplência, e tem por conceito, o não cumprimento de uma obrigação estabelecida para o devedor, ou seja, o cliente deixa de assumir o compromisso de pagamento de bens ou serviços de uma empresa ou instituição até sua data de vencimento. O impacto destas inadimplências nas empresas leva a redução de recursos para arcar com as despesas fixas, renovar o estoque, pagar fornecedores, impede novos investimentos e quando a receita fica reduzida por falta destes pagamentos, pode levar o negócio a falência (ANGELO; BELTRAME; DIAS, 2020). Seguindo a linha destes conceitos apresentados, esse trabalho sugere aplicar o algoritmo floresta aleatória (*Random Forest,* em inglês) utilizado em modelos preditivos de AM, para que seja possível classificar um conjunto de dados que contenha informações financeiras de clientes e assim prever se eles serão inadimplentes. Com isso as vantagens desta previsão são inúmeras, pois é possível se antecipar com o possível cliente inadimplente e mitigar futuras dívidas não pagas, essa tomada de decisão favorece até para análise de crédito mais cautelosa com este cliente, em consequência disso a empresa terá uma melhor gestão de risco de crédito, conseguirá equilíbrio do fluxo de caixa, redução de problemas com clientes, irá favorecer a lucratividade e por fim aumento da competitividade do negócio (ANGELO; BELTRAME; DIAS, 2020). Mas, através da análise preditiva com este algoritmo como saber quais clientes de uma empresa ou instituição financeira que provavelmente seriam inadimplentes no próximo mês?

O objetivo geral desse estudo está baseado em uma técnica para previsão de inadimplência utilizando o RStudio com o algoritmo floresta aleatória. Esta técnica está dentro do contexto de AM supervisionado com tarefas de aprendizagem por classificação, para prever se os clientes de uma instituição financeira ou empresa de

uma fonte de dados que tenha as informações necessárias para se realizar este processo de aprendizagem, serão inadimplentes ou adimplentes no próximo pagamento.

Como o cumprimento do objetivo geral foram necessários os seguintes procedimentos:

- Pesquisa e coleta de dados de uma fonte disponível e confiável para realizar os testes necessários;
- Também foi utilizado o *software* RStudio que é uma ferramenta *open source* própria para análise de dados para aplicar o algoritmo floresta aleatória com a linguagem R;
- Realizado o processamento dos dados de entrada coletados e identificado os atributos mais relevantes para o conjunto de variáveis preditoras do algoritmo;
- Por fim, apresentado os resultados de acurácia da predição no RStudio e verificado o quanto podem ser aceitáveis para a solução do problema.

1.1 JUSTIFICATIVA

A pesquisa sobre o algoritmo floresta aleatória reforça seu valor prático para o uso na análise preditiva de grandes volumes de dados como método de classificação do aprendizado de máquina supervisionado. Contudo, o estudo favorece melhor entendimento sobre o comportamento deste algoritmo para prever se uma pessoa dentro do contexto das finanças tende a ser inadimplente ou não, determinando assim o quanto pode ser viável como solução deste problema de inadimplência, pois é notável o quanto este algoritmo classificador auxiliaria empresas e instituições financeiras.

2. REFERENCIAL TEÓRICO

Os tópicos que seguem neste capítulo exploram uma melhor compreensão dos conceitos a serem tratados neste trabalho.

2.1 GESTÃO DE RISCO DE CRÉDITO

Como conceito defensivo, a gestão de risco é considerada uma continuidade de redução de riscos assumidos pela corporação, como uma proatividade que se concentra mais na seleção do tipo e nível de ameaças para o negócio, destaca-se ainda que para realizar esta gestão, um dos fatores a ser controlado é a inadimplência (NUNES, 2013). Realmente existe a importância da previsão de adimplentes e inadimplentes no setor de crédito de uma instituição financeira, pois um modelo preditivo pode proporcionar vantagem competitiva aos negócios (CAMARGOS, 2012).

Conforme esclarece Araújo e Carmona (2007), a principal finalidade em modelos de risco de crédito é poder prever a inadimplência, ainda fica enfatizado as melhorias em uma empresa que aplica uma abordagem preditiva para bons ou maus pagadores. Um dos destaques é maximizar o desempenho das funções de assistência operacional dos créditos, outra vantagem é a minimização de custos operacionais para instituição financeira, também é enfatizado que com estas informações pode-se assessorar o comitê de créditos fornecendo capacitação nas tomadas de decisão com embasamento maior, ainda com estes benefícios o ganho para realizar a função de análise e concessão de crédito é que no processo de avaliação de risco pode-se ter maior precisão, redução de equívocos e diminuição da subjetividade (ARAÚJO; CARMONA, 2007).

2.2 ALGORITMO FLORESTA ALEATÓRIA

Para chegarmos ao conceito do algoritmo floresta aleatória, é necessário entender a definição de árvores de decisão que de acordo com Abud (2018), é uma

técnica baseada em estrutura de dados de árvores binárias composta por nós pai *p* e nós filho *n*, onde o nó *p* contém o vetor de atributos $[x_1, x_2, x_3]$ a serem calculados para decidir o valor de classe $\{y_1, y_2\}$ do nó *n*. O que pode ser entendido sobre o vetor de atributos no contexto deste trabalho, é que são os valores contidos numa coluna de dados (por exemplo estado civil, idade, nível de escolaridade, etc) de uma tabela contendo a massa de dados de clientes devedores ou não, por meio de um cálculo obtêm-se o valor binário de classe que seria inadimplente ou adimplente e o resultado é atribuído ao nó *n*. De acordo com Abud (2018), para se calcular os valores de um nó na construção de uma árvore de decisão se realizam basicamente dois cálculos, e um deles é a entropia descrito na ilustração 1 abaixo:

Ilustração 1 – Fórmula da entropia

$$\text{entropia}(\mathcal{S}) = -\sum_{k=1}^{K} \left(\frac{freq(Y_k, \mathcal{S})}{|\mathcal{S}|} \right) \log_2 \left(\frac{freq(Y_k, \mathcal{S})}{|\mathcal{S}|} \right)$$

Fonte: Abud (2018)

Neste cálculo temos *S* representando um conjunto de uma coluna de dados, esta entropia busca classificar o nível de impureza relacionado à classe, como exemplo temos um conjunto categórico indicando se cliente é inadimplente ou não, o total de linhas deste grupo representa a divisão por *S* na fórmula e a frequência com que acontece um valor *sim* e *não* neste grupo infere sobre a expressão $freq(Y_k, S)$. Os outros atributos da tabela de dados serão calculados da mesma forma de acordo com a frequência que um grupo destes valores ocorrem relacionados aos valores binários da classe, desta forma os valores variam entre 0 e 1, onde um grupo bem homogêneo é igual ou próximo a 0,5, pois esta separação de valores representa uma classificação binária mais bem dividida. O cálculo que determina a criação do nó raiz terá a entropia do conjunto da variável de decisão relacionada com entropia de outro subconjunto para buscar o que é chamado de ganho de informação (ABUD,

2018). Abaixo na ilustração 2 temos a expressão matemática deste ganho de informação:

Ilustração 2 – Fórmula de ganho de informação

$$\text{ganho}(\mathcal{S}, A) = \text{entropia}(\mathcal{S}) - \sum_{q=1}^{Q} \frac{|\mathcal{S}_{v_q}|}{|\mathcal{S}|} \cdot \text{entropia}(\mathcal{S}_{v_q})$$

Fonte: Abud 2018

Nesta fórmula exposta por Abud (2018), temos a entropia da classe menos o somatório do peso entre as amostras de um subconjunto $|S_{vq}|$ e o conjunto da classe $|S|$, multiplicados pela entropia do subconjunto $entropia(S_{vq})$. O valor encontrado neste ganho de informação indica a pureza nas divisões dos subconjuntos e com isso é possível classificar e determinar o nó raiz da árvore de decisão que será constituído pela variável que trazer o valor mais próximo ou igual a 1, assim todo o processo se repete nos demais subconjuntos criando os outros nós filhos.

Floresta aleatória é um algoritmo de aprendizagem de máquina supervisionada que pode ser utilizado em tarefas de classificação, que cria um conjunto combinado de árvores de decisão de forma aleatória para obter maior precisão e estabilidade, definindo os dados de saída como aqueles pertencentes à coluna da massa de dados com maior frequência (ABUD, 2018). O algoritmo floresta aleatória se utiliza dos métodos *ensemble* que fazem uma combinação de diferentes modelos do processo de AM para conseguir um único resultado, esse método faz com que um único algoritmo aproveite todos os modelos criados (JUNIOR, 2019). O que muda no funcionamento do algoritmo floresta aleatória em relação as árvores de decisão é que a definição da classe que irá constituir o nó raiz da árvore estruturada não acontece com base em todas as variáveis (conjunto das colunas de dados) disponíveis, assim o algoritmo escolhe de forma randômica algumas variáveis (pode

ser definido manualmente duas ou mais) para definir o melhor atributo para classificação (JÚNIOR, 2019).

Testado em diversas bases de dados, este algoritmo consegue trazer uma maior precisão nos resultados de previsão de dados, ainda possui a vantagem de se adaptar num ambiente que necessite processar uma grande quantidade de dados, pois dá ênfase para atributos mais importantes e menor relevância em outros atributos existentes na tarefa de classificação (SILVA, 2014).

2.3 RSTUDIO

Para aplicação do modelo preditivo foi utilizado o software RStudio que é um ambiente de desenvolvimento integrado (IDE) para a linguagem R. O software possui ferramentas para plotagem, depuração, suporte a scripts de análise de dados entre outras características. O software é de código aberto e possui a licença AGPL v3, dessa forma não implica em custos, porém até possui a versão paga RStudio Server Pro que se aplica à ciência de dados no meio empresarial (RSTUDIO, 2020).

A linguagem R que é aplicada no RStudio é destinada para computação e gráficos estatísticos e possui uma ampla variedade de técnicas como modelagem linear e não linear, análise de séries temporais, classificação, agrupamento, etc. R pode ser estendido através de pacotes (bibliotecas com funções específicas) que são fornecidos pela própria distribuição R ou pelo repositório CRAN (do inglês *Comprehensive R Archive Network*). Esses pacotes podem ser importados dentro do ambiente do RStudio e possibilitam o uso de suas funções de acordo com o código desenvolvido (R-PROJECT, 2020).

Para possibilitar a implementação de modelos preditivos se fez necessário o uso dos pacotes *caret, randomForest*, *nnet, ggplo2* para apresentar os resultados em gráficos entre outros conhecidos na distribuição R ou CRAN que contém as funções relevantes para o desenvolvimento deste trabalho. Dessa maneira com todos estes recursos é possível executar o algoritmo desejado e realizar o

processamento, treino e teste dos dados entre outras técnicas de aprendizado de máquina para criação de um modelo preditivo. Um exemplo aplicado destas técnicas por Muller e Massarom (2019), nos mostra como pode-se organizar as previsões de um conjunto de dados de teste em uma matriz de confusão. Essa matriz é uma forma de visualizar os dados reais aos preditos na amostra de validação, ou seja, cruza as informações numa tabela de dupla entrada, com isso pode ser observado o percentual de acertos nas previsões do modelo com o algoritmo floresta aleatória.

3. METODOLOGIA

Para cumprimento dos objetivos deste trabalho, será fundamental a realização das atividades citadas nos objetivos específicos, caracterizando assim uma pesquisa mais de cunho descritivo. Para alcançar os resultados da previsão de inadimplência é importante levantar as técnicas e procedimentos utilizados em cada objetivo.

Atendendo ao primeiro objetivo específico, foram coletados os dados de uma empresa de Ijuí, Rio Grande do Sul. É uma empresa com mais de dez anos no mercado e bem consolidada. A base de dados trabalhada está em um Sistema Gerenciador de Banco de Dados (SGBD) Postgresql e possui um grande volume de dados para processamento. Os dados desta base serão exportados para um arquivo no formato CSV (*comma-separeted values*) que facilitará sua importação com a ferramenta RStudio. As informações contidas desta extração possuem dados de pagamentos de clientes da empresa, é basicamente um histórico contendo vários atributos destas pessoas, inclusive se já estão inadimplentes ou não (classe preditora). Nesta importante etapa será feito uma seleção das colunas de dados relevantes para a construção do modelo preditivo, como por exemplo idade do cliente, estado civil, valor pago no mês atual, valor pago no mês anterior, quantidade de crédito e assim por diante.

O passo seguinte é sobre a utilização da ferramenta Rstudio, que será imprescindível para trabalhar com as informações fornecidas pela empresa. Para dar sequência dos procedimentos deve ser instalado o interpretador da linguagem R, e como o Sistema Operacional utilizado será Windows e deve-se instalar também o Rtools que permitirá compilar os pacotes. Por fim, instala-se o Rstudio e aplicam-se as configurações básicas como, por exemplo, a instalação dos pacotes que contém as funções para análise dos dados e até mesmo para execução do algoritmo floresta aleatória. Essa ferramenta permitirá que o modelo de AM desenvolvido nesse estudo possa ser salvo e reutilizado em novos conjuntos de dados para futuras previsões.

O terceiro objetivo consiste em processar os dados afim de obter resultados que irão corroborar para o AM de classificação e também será definido o modelo preditivo. Dentre essas atividades de processamento serão renomeadas algumas colunas de dados (atributos) para melhor leitura e apresentação e para redução de dimensionalidade, certas variáveis até serão eliminadas como forma de trabalhar só com as mais relevantes e obter melhora no desempenho do modelo induzido, redução no custo computacional e resultados mais compreensíveis. Para o processo de limpeza será aplicado uma análise para explorar possíveis valores faltantes ou nulos em cada uma das variáveis e destas ocorrências será feito a eliminação do registro, pois isso pode implicar numa baixa qualidade dos dados já que o que se quer alcançar é uma melhor precisão das previsões. Os atributos finais tratados desta base de dados serão adicionados a este trabalho com nome e descrição de forma tabular.

Outro passo a se considerar é tratar seletivamente certas variáveis para ficarem com os dados qualitativos e não quantitativos, este é um método de discretização que maximiza a pureza dos intervalos da variável e favorece melhor aplicação do algoritmo de classificação, pois a existência de atributos com uma quantidade muito maior de exemplos que os demais pode levar a indução de classificadores tendenciosos para os atributos majoritários, ou seja, a redução desta

grande variedade de valores evita que um atributo predomine sobre outro e sem este tratamento o conjunto de dados pode ficar desbalanceado (GAMA; et. al, 2011).

Para construção do modelo será utilizado o método de amostragem estratificada proporcional para representar o conjunto de dados original, desta amostra será possível criar o subconjunto de dados de treino que se baseia em valores de saída já conhecidos para cada variável. Como o subconjunto de dados de treino é um percentual do conjunto original por se basear na amostra, o que resta do conjunto original será utilizado como conjunto de dados de teste. Com o objetivo de conseguir bons resultados e evitar que este modelo só fique bom se for treinado no mesmo conjunto de dados será aplicado a técnica de validação cruzada (*cross-validation*, em inglês), que consiste em selecionar e avaliar o modelo de aprendizado criado por meio da divisão do conjunto de dados original em duas ou mais partes, afim de estimar o desempenho esperado dos modelos treinados (GAMA; et. al, 2011).

Em continuidade dos procedimentos será executado então a função que aplica o algoritmo floresta aleatória no conjunto de dados de treino, é nesta parte que o programa irá criar o modelo preditivo, pois ele está aprendendo os relacionamentos a partir desses dados.

Por fim, com o modelo de aprendizado de máquina criado será possível visualizar suas informações e será incluído uma figura com os resultados neste trabalho, a geração deste resultado nada mais é do que uma fórmula matemática que acabou de ser encontrada a partir do relacionamento entre as variáveis preditoras e a variável binária de saída (classe[1]). Será usado o coeficiente Mean Decrease Gini a fim de medir qual a influência de cada variável para homogeneidade dos nós e as folhas na floresta aleatória (conjunto de árvores de decisão[2])

[1] A classe é a variável preditora do conjunto de dados, a qual é a coluna do dataset que o algoritmo fornecerá os valores de classificação para inadimplentes ou não.
[2] Combinação das árvores de decisão criadas de maneira aleatória pelo algoritmo Random Forest onde cada árvore será utilizada na escolha do resultado final.

resultante. Esse coeficiente realiza a soma do decrescimento para cada variável no índice de cada nó relacionado a todas as árvores (CORREIA, 2019).

Com os recursos disponíveis, poderá ser mostrado as taxas de erros do modelo através de gráficos bem como demostrar um ranqueamento das variáveis preditoras com o grau de importância. A exposição destes valores mostrará quais características mais influenciam nas previsões. Em sequência vou aplicar este modelo de AM com algoritmo floresta aleatória no conjunto de dados de teste que fundamenta-se em avaliar o desempenho do modelo gerado na etapa anterior com dados que não foram utilizados no treinamento, exibindo uma estimativa da sua capacidade de classificação, pois até aqui ele ainda estava sendo aplicado no conjunto de treino. O resultado das previsões deste conjunto de teste vai ser exposto por meio da técnica de matriz de confusão, onde serão realizadas algumas análises como sua acurácia, quantidade de acertos e erros de previsão e o ponto principal que é o número de clientes que provavelmente seria inadimplente no próximo mês. Para visualização dos resultados observados analisados e descritos serão adicionadas imagens com esses registros gerados no RStudio.

4. APLICAÇÃO DO AM COM FLORESTA ALEATÓRIA

Dando continuidade dessa pesquisa foi apresentado nas próximas subseções as atividades trabalhadas para satisfazer os objetivos de forma proveitosa e substancial.

4.1 COLETA DE DADOS

Nessa importante etapa foi coletado os dados necessários para o desenvolvimento do trabalho por meio de um *dataset* (conjunto de dados) que contém mais de trinta mil registros dos clientes que possuem cartão de crédito e um perfil de maior movimentação. Para maior obtenção de histórico desses registros e

melhor precisão do modelo preditivo foram selecionados vinte e quatro atributos e um rótulo de classe. Descrição de cada atributo está detalhada na tabela 1 abaixo:

Tabela 1 – Descrição dos atributos

Atributo	**Descrição**
codigo	Código do cliente da empresa
qtdade_credito_cartao	Quantidade de limite no cartão de crédito
sexo	Sexo do cliente, 1 = masculino; 2 = feminino
grau_instrucao	Nível de escolaridade, 1 = pós-graduado; 2 = graduado; 3 = ensino médio; 4 = outros
estado_civil	Estado civil, 1 = casado, 2 = solteiro, 3 = outros
idade	Idade do cliente
status_pag_setembro	Situação de pagamento em setembro
status_pag_agosto	Situação de pagamento em agosto
status_pag_julho	Situação de pagamento em julho
status_pag_junho	Situação de pagamento em junho
status_pag_maio	Situação de pagamento em maio
status_pag_abril	Situação de pagamento em abril
vlr_fatura_setembro	Valor da fatura do cartão em setembro
vlr_fatura_agosto	Valor da fatura do cartão em agosto
vlr_fatura_julho	Valor da fatura do cartão em julho
vlr_fatura_junho	Valor da fatura do cartão em junho
vlr_fatura_maio	Valor da fatura do cartão em maio
vlr_fatura_abril	Valor da fatura do cartão em abril

vlr_pago_setembro	Valor pago em setembro
vlr_pago_agosto	Valor pago em agosto
vlr_pago_julho	Valor pago em julho
vlr_pago_junho	Valor pago em junho
vlr_pago_maio	Valor pago em maio
vlr_pago_abril	Valor pago em abril
inadimplente	Classe preditora, valor 0 ou 1 – 0 Significa não inadimplência e 1 significa inadimplência

Fonte: o autor

Para facilitar interpretação dos dados, as colunas do *dataset* foram renomeadas já na extração das informações. Os atributos status_pag_abril, status_pag_maio, status_pag_junho, status_pag_julho, status_pag_agosto e status_pag_setembro são o estado do pagamento nos meses de abril a setembro respectivamente. O padrão numérico determinado pela empresa para as situações de pagamento se define conforme tabela 2 abaixo:

Tabela 2 – Padrão para definir situação de pagamento

Padrão numérico	**Descrição**
-1	Pagamento em dia
0	Pagamento parcial da fatura
1	Atraso no pagamento por um mês
2	Atraso no pagamento por dois meses
3	Atraso no pagamento por três meses
4	Atraso no pagamento por quatro meses
5	Atraso no pagamento por cinco meses

6	Atraso no pagamento por seis meses
7	Atraso no pagamento por sete meses
8	Atraso no pagamento por oito meses
9	Atraso no pagamento por nove meses ou mais

Fonte: o autor

Existe um forte relacionamento das colunas descritas sobre valor da fatura, valor pago e status de pagamento, por exemplo, se tenho *vlr_fatura_abril* com um valor de 2150, esse mesmo valor aparecerá na coluna *vlr_pago_maio* se valor numérico para situação de pagamento na coluna *status_pag_maio* for de -1. Caso valor na coluna *vlr_pago_maio* fosse inferior a 2150 o valor para situação na coluna *status_pag_maio* seria zero.

4.2 PROCESSANDO OS DADOS COM RSTUDIO

Após realizado as configurações necessárias para o projeto no RStudio e definido a pasta de trabalho foram instalados e carregados os pacotes *Amelia*, *caret*, *ggplot*, *dplyr*, *reshape* e *randomForest*. Assim, com o dataset carregado foi identificado que o atributo *codigo* não é relevante para o modelo, pois essa informação não determina se o cliente será inadimplente, com isso eliminou-se a coluna.

Como todas as colunas desse dataset são formadas por dados do tipo inteiro, ou seja, as variáveis são quantitativas, se fez necessário aplicar a discretização dos dados para deixar o conjunto de dados melhor balanceado. Com isso foi determinado categorias para os intervalos de dados de algumas variáveis. Para o atributo *idade* foram categorizados como *jovem* os clientes de até 30 anos, entre 30 e 50 como *adulto* e para 50 anos ou mais como *idoso*. No atributo *sexo* o valor 1 foi alterado para *masculino* e 2 para *feminino*, já para a coluna *grau_instrucao* os dados foram modificados de 1 para *posgraduado*, 2 para *graduado*, 3 para *ensinomedio* e 4

para *outros*. As classificações para o atributo *estado_civil* se deram com as alterações do valor 1 para *casado*, 2 para *solteiro* e 3 para *outros* conforme o próprio significado dos valores do *dataset* sugere. O que possibilitou o pré-processamento de dados para transformar esses atributos quantitativos em qualitativos foi a utilização do pacote *dplyr* no RStudio. Como forma de permitir classificar melhor os dados também foram alterados as colunas que indicam situação de pagamento e a coluna *inadimplente* para o tipo de dados *fator* da linguagem R. O tipo de dados fator é uma estrutura específica que possibilita definir níveis para uma variável quando se quer expressar ela como categórica (MUELLER; MASSARON, 2019).

Seguindo para o processo de limpeza dos dados e assegurar qualidade do dataset, foi possível observar os valores faltantes com a função *sapply* da linguagem R no RStudio conforme pode ser visto a seguir na ilustração 3:

Ilustração 3 – valores faltantes por coluna de dados

```
> sapply(dataset, function(x) sum(is.na(x)))
qtdade_credito_cartao                  sexo         grau_instrucao           estado_civil
                    0                     0                    345                      0
               idade    status_pag_setembro      status_pag_agosto       status_pag_julho
                    0                     0                      0                      0
     status_pag_junho        status_pag_maio      status_pag_abril    vlr_fatura_setembro
                    0                     0                      0                      0
    vlr_fatura_agosto       vlr_fatura_julho      vlr_fatura_junho        vlr_fatura_maio
                    0                     0                      0                      0
     vlr_fatura_abril      vlr_pago_setembro        vlr_pago_agosto         vlr_pago_julho
                    0                     0                      0                      0
       vlr_pago_junho          vlr_pago_maio        vlr_pago_abril           inadimplente
                    0                     0                      0                      0
```

Fonte: o autor

Logo, os registros que continham esses valores nulos foram removidos do conjunto, o que também pode ser repensado futuramente caso necessite analisar melhorias nessa etapa de processamento para o resultado final. Por fim, pude obter um resumo das transformações com detalhes das variáveis e seus tipos de dados e níveis categóricos trabalhados até o momento com a ilustração 4 capturada do console (resultado do interpretador de comandos R) do RStudio abaixo:

Ilustração 4 – Estrutura do objeto dataset

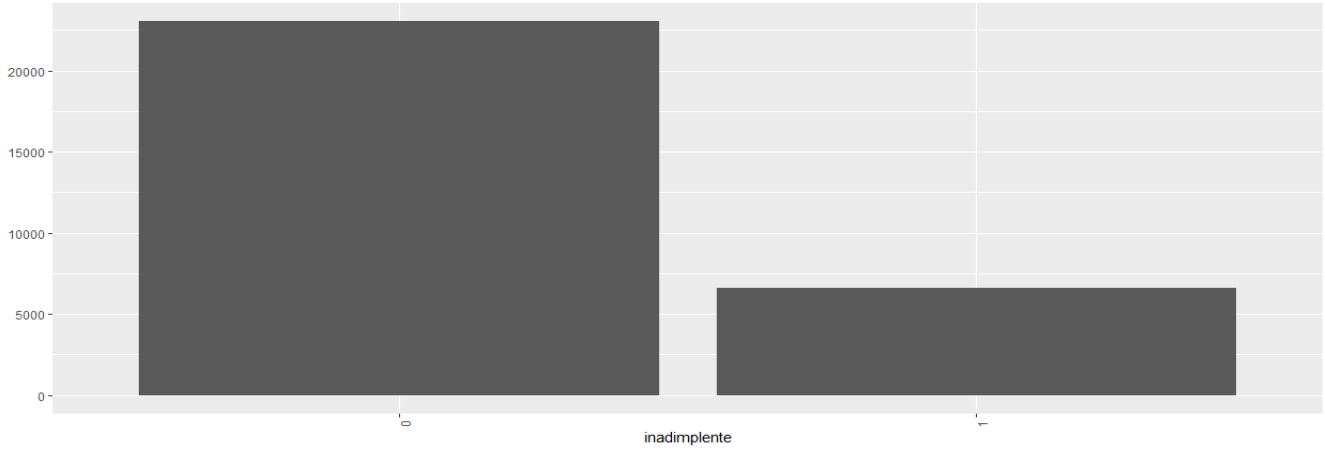
```
> str(dataset)
'data.frame':   29655 obs. of  24 variables:
 $ qtdade_credito_cartao: int  20000 120000 90000 50000 50000 50000 500000 100000 140000 20000 ...
 $ sexo                 : Factor w/ 2 levels "Masculino","Feminino": 2 2 2 2 1 1 1 2 2 1 ...
 $ grau_instrucao       : Factor w/ 4 levels "posgraduado",..: 2 2 2 2 2 1 1 2 3 3 ...
 $ estado_civil         : Factor w/ 4 levels "Desconhecido",..: 2 3 3 2 2 3 3 3 2 3 ...
 $ idade                : Factor w/ 3 levels "Jovem","Adulto",..: 1 1 2 2 3 2 1 1 1 2 ...
 $ status_pag_setembro  : Factor w/ 11 levels "-2","-1","0",..: 5 2 3 3 2 3 3 3 3 1 ...
 $ status_pag_agosto    : Factor w/ 11 levels "-2","-1","0",..: 5 5 3 3 3 3 3 2 3 1 ...
 $ status_pag_julho     : Factor w/ 11 levels "-2","-1","0",..: 2 3 3 3 2 3 3 2 5 1 ...
 $ status_pag_junho     : Factor w/ 11 levels "-2","-1","0",..: 2 3 3 3 3 3 3 3 3 1 ...
 $ status_pag_maio      : Factor w/ 10 levels "-2","-1","0",..: 1 3 3 3 3 3 3 3 3 2 ...
 $ status_pag_abril     : Factor w/ 10 levels "-2","-1","0",..: 1 4 3 3 3 3 3 2 3 2 ...
 $ vlr_fatura_setembro  : int  3913 2682 29239 46990 8617 64400 367965 11876 11285 0 ...
 $ vlr_fatura_agosto    : int  3102 1725 14027 48233 5670 57069 412023 380 14096 0 ...
 $ vlr_fatura_julho     : int  689 2682 13559 49291 35835 57608 445007 601 12108 0 ...
 $ vlr_fatura_junho     : int  0 3272 14331 28314 20940 19394 542653 221 12211 0 ...
 $ vlr_fatura_maio      : int  0 3455 14948 28959 19146 19619 483003 -159 11793 13007 ...
 $ vlr_fatura_abril     : int  0 3261 15549 29547 19131 20024 473944 567 3719 13912 ...
 $ vlr_pago_setembro    : int  0 0 1518 2000 2000 2500 55000 380 3329 0 ...
 $ vlr_pago_agosto      : int  689 1000 1500 2019 36681 1815 40000 601 0 0 ...
 $ vlr_pago_julho       : int  0 1000 1000 1200 10000 657 38000 0 432 0 ...
 $ vlr_pago_junho       : int  0 1000 1000 1100 9000 1000 20239 581 1000 13007 ...
 $ vlr_pago_maio        : int  0 0 1000 1069 689 1000 13750 1687 1000 1122 ...
 $ vlr_pago_abril       : int  0 2000 5000 1000 679 800 13770 1542 1000 0 ...
 $ inadimplente         : Factor w/ 2 levels "0","1": 2 2 1 1 1 1 1 1 1 1 ...
 - attr(*, "na.action")= 'omit' Named int [1:345] 48 70 386 503 505 1074 1266 1283 1370 1832 ...
  ..- attr(*, "names")= chr [1:345] "48" "70" "386" "503" ...
>
```

Fonte: o autor

4.3 CONSTRUINDO O MODELO PREDITIVO

Após ter o conjunto de dados devidamente processado foi analisado a diferença entre clientes inadimplentes a adimplentes e verificado sua proporção para então definir uma amostra estratificada para o subconjunto de dados de treino do modelo de AM. Segue diferença no conjunto original exposta abaixo no gráfico 1:

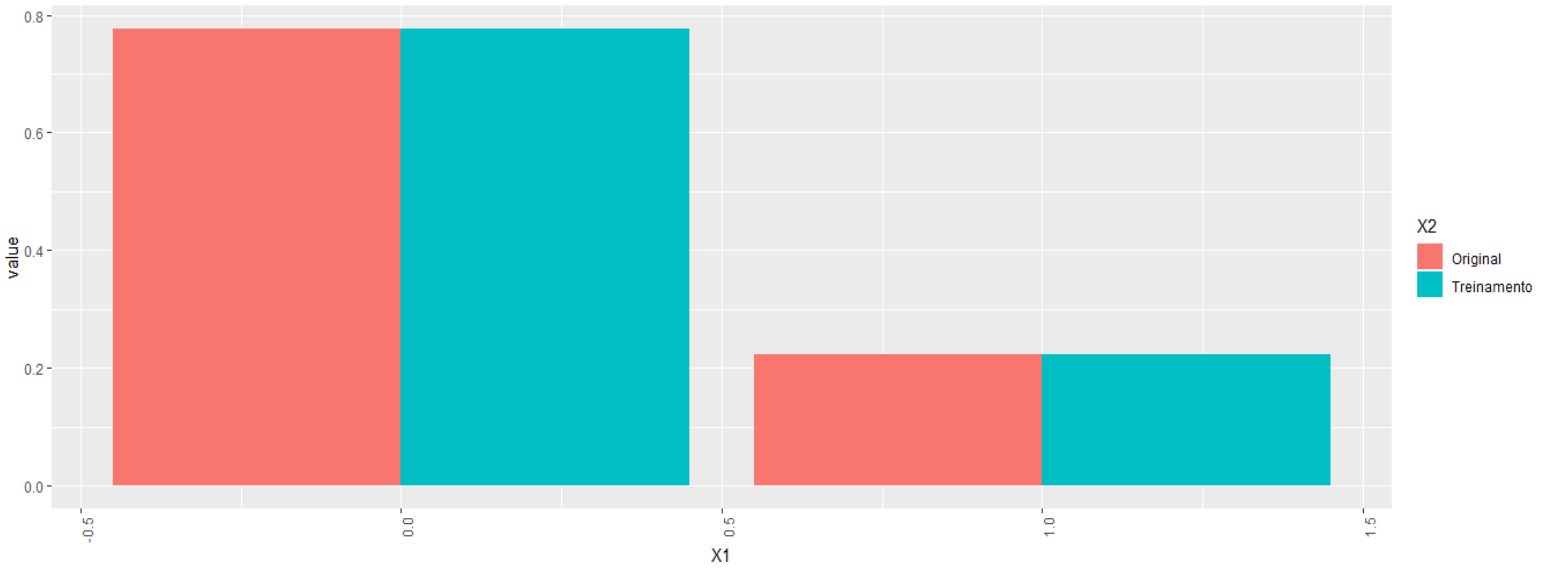
Gráfico 1 – Adimplentes versus inadimplentes



Fonte: o autor

O gráfico de barras representa a variável *inadimplente* com os dados de clientes adimplentes na coluna da esquerda (mais de 23000) e inadimplentes na coluna da direita (mais de 6000). O que se pode observar é que os dados refletem bem a situação atual da empresa pois temos um número de inadimplentes bem inferior ao de adimplentes e não o contrário, senão a empresa estaria com sérios problemas. Nessa mesma proporção foi extraído uma amostra para o subconjunto de treino e assim poder criar o modelo preditivo mantendo quantidade das ocorrências da classe preditora o mais próximo possível do original. O R dispõe de um pacote especializado com funções para divisão de dados, o nome do pacote é o *caret*. Como resultado dessa distribuição de dados de treino, foi possível comparar a proporção das quantidades em cada categoria da variável *inadimplente* entre o conjunto de treino e o original. Para isso foi utilizado funções do pacote *ggplot2,* onde o resultado foi exportado para o gráfico 2 que pode ser observado abaixo:

Gráfico 2 – Conjunto de treino versus dataset original



Fonte: o autor

Temos no gráfico de barras o eixo X1 com os valores de classe 0 e 1 acompanhado das barras duplas dos dois conjuntos de treino e original, as barras duplas do lado esquerdo representam os bom pagadores e do lado direito os maus pagadores, onde no eixo X2 se apresentam os valores percentuais dos dados que se mostram praticamente iguais para inadimplentes com pouco mais de vinte porcento e adimplentes próximo de oitenta porcento.

Em continuidade dos procedimentos foi aplicado a validação cruzada que pode diminuir os riscos oferecidos pela divisão dos conjuntos pois deixa de fora do treinamento alguns exemplos úteis para amostra. As vantagens desse procedimento é que independente do número de observações a tendenciosidade diminui e testando todas as observações verifica-se totalmente sua hipótese de AM, além disso pode-se esperar um desempenho preditivo (MUELLER; MASSARON, 2019).

Seguindo as etapas de aprendizado de máquina executa-se em seguida o algoritmo *Random Forest* nesse subconjunto de dados de treino onde se faz uso do pacote *randomForest* no RStudio. Após criação do modelo com esse algoritmo o console do RStudio traz informações bem interessantes que podem ser visualizadas na ilustração 5 abaixo:

Ilustração 5 – Modelo preditivo nos dados de treino

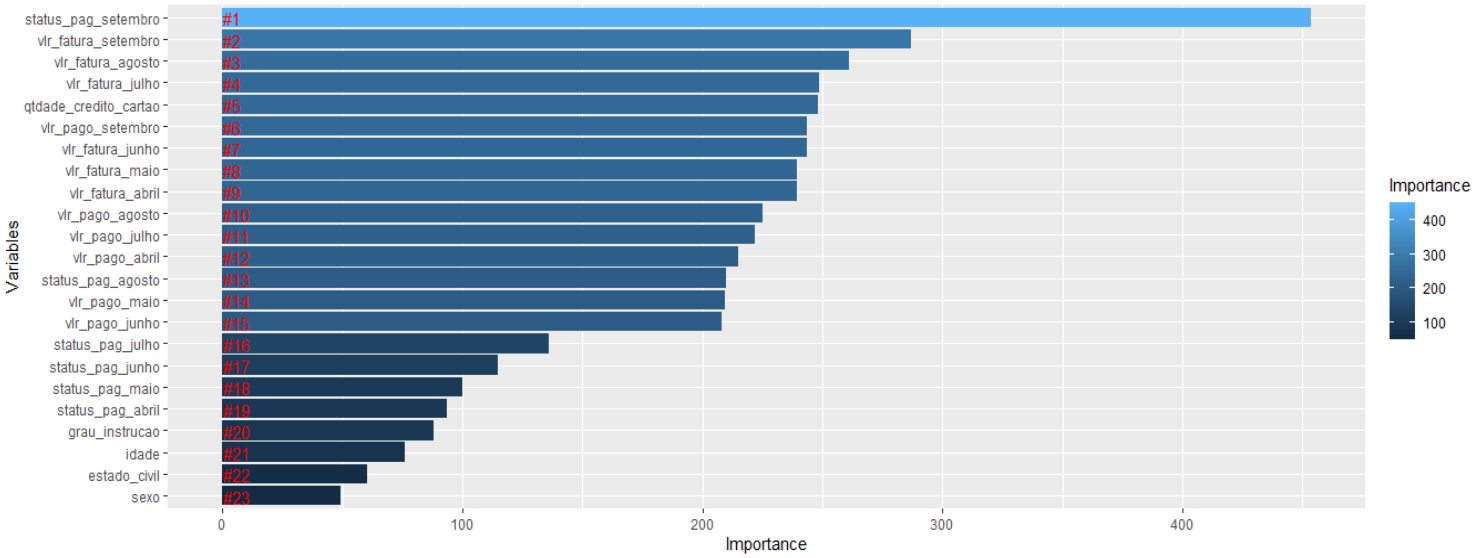
```
> rf_model

Call:
 randomForest(formula = inadimplente ~ ., data = trainData)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 4

        OOB estimate of  error rate: 18.46%
Confusion matrix:
     0    1 class.error
0 9749  622  0.05997493
1 1842 1133  0.61915966
```

Fonte: o autor

Nesse resultado pode-se observar o número de 500 árvores de decisão utilizadas no cálculo de classificação assim como trouxe uma matriz de confusão (*confusion matrix*) que objetivamente tem-se um número de 9749 acertos na previsão para os casos de clientes adimplentes e 1842 erros para esse mesmo valor de classe (zero), no entanto quando os dados do subconjunto de treino eram inadimplentes (1) o modelo errou 622 previsões e acertou 1133.

Com esse modelo já é possível verificar o grau de importância das variáveis e assim identificar qual informação provoca mais influência nas previsões. Dessa forma foi utilizado algumas funções da linguagem R, inclusive com suporte ao uso do coeficiente Mean Decrease Gini, e juntamente com os recursos do pacote *ggplot2* para tornar-se praticável um ranqueamento dos atributos e assim expor através do gráfico 3 demonstrado abaixo:

Gráfico 3 – Importância das variáveis do modelo preditivo



Fonte: o autor

Esse gráfico demonstra como foi classificado a importância das variáveis de cima para baixo, consequentemente observou-se que as menos importantes são de

dados demográficos como idade, grau_instrucao, estado_civil e sexo, o que pode ser bem compreensível pois o sexo da pessoa num primeiro momento não vai determinar se ela deixará de pagar alguma fatura ou não. Em sequência dos procedimentos de AM foi executado esse modelo sobre o subconjunto de teste que é o restante do conjunto original de dados coletados, onde ao finalizar essa rotina evidencia-se os resultados tratados na seção seguinte.

5. RESULTADOS OBTIDOS

Quando criamos o modelo de AM com a divisão do dataset em um subconjunto de treino e testamos de fato esse modelo no subconjunto de teste com o RStudio é possível extrair com recursos do R a matriz de confusão que faz o cruzamento das previsões sobre a classe preditora. Dessa maneira conseguiu-se uma avaliação potencial da classificação do modelo usado anteriormente conforme pode ser observado na ilustração 6 retirada do console do RStudio:

Ilustração 6 – Matriz de confusão nos dados de teste

```
> matrizConf
Confusion Matrix and Statistics

          Reference
Prediction     0     1
         0 11940  2256
         1   734  1379

               Accuracy : 0.8167
                 95% CI : (0.8106, 0.8226)
    No Information Rate : 0.7771
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.3779

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.37937
            Specificity : 0.94209
         Pos Pred Value : 0.65263
         Neg Pred Value : 0.84108
             Prevalence : 0.22288
         Detection Rate : 0.08455
   Detection Prevalence : 0.12956
      Balanced Accuracy : 0.66073

       'Positive' Class : 1
```

Fonte: o autor

A matriz apresentada desse resultado do modelo preditivo no subconjunto de teste é uma informação importante para representar a performance da classificação, pois descreve o esforço para prever o valor atribuído nas amostras. Para interpretação de predição dessa amostra observa-se que quando os dados do conjunto de teste eram zero (coluna 0 - adimplentes) o modelo acertou 11940 previsões e errou 734, já no momento que os dados do conjunto se confirmavam como um (coluna 1 - inadimplentes) o modelo errou 2256 previsões e acertou para clientes adimplentes 1379. Os acertos de predição totalizados em 13319 representam 81,67% do total da amostra e isso caracteriza a acurácia do modelo aplicado. O valor no campo *Sensitivity* (sensibilidade) representa uma taxa de 37,93% de amostras positivas classificadas corretamente sobre o total de inadimplentes, o valor descrito no campo *Specificity* (especificidade) é uma taxa de 94,20% de amostras negativas identificadas corretamente sobre o total de clientes adimplentes. Entre tantas métricas relacionadas com os erros e acertos preditivos o que pode não ser um atrativo em soluções nesse caso talvez seja a quantidade de acertos de previsão serem muito maiores em clientes adimplentes, pois são 84,10% de valor preditivo negativo (*Neg Pred Value*) contra apenas 65,26% de valor preditivo positivo (*Pos Pred Value*).

Embora a acurácia possa parecer uma quantidade grande de acertos nas previsões deve-se considerar outras características que venham sustentar resultados mais satisfatórios do problema proposto.

Uma observação positiva analisada no resultado desse modelo está relacionada com a boa capacidade de generalização de hipótese, que segundo Gama et al. (2011) uma hipótese ou modelo de AM tem como requisito importante a capacidade de trabalhar com dados imperfeitos de forma que minimize distorções nas previsões de uma classe do conjunto de treinamento, e ainda se tem por objetivo que essa hipótese seja capaz de lhe dar com novos dados de mesmo domínio ou problema.

Quando se tem uma boa capacidade de generalização de hipótese é porque o modelo conseguiu extrair os padrões para previsão tanto no conjunto de treino quanto no de teste, no entanto se houver baixa capacidade de generalização significa que o modelo sofreu sobreajuste (*overfitting)*, ou seja, só aprendeu com os dados de treino e não obtém bons resultados em um conjunto de teste. Existe ainda o caso de a hipótese apresentar baixa taxa de acerto até para o conjunto de treino, o que configura a condição de subajuste (*underfitting*) (GAMA; et. al, 2011).

Conforme conceitos mostrados anteriormente o modelo proposto no artigo não sofreu sobreajuste nem subajuste pois apresentou uma acurácia de 83% (11082 de acertos) no conjunto de treino visto na seção 4.2, ilustração 5 e manteve um percentual bem próximo no conjunto de teste.

6. CONSIDERAÇÕES FINAIS

Os resultados apresentados sobre o algoritmo nesse modelo de AM demonstram que ele pode ser utilizado em novos conjuntos de dados de mesma natureza, pois se apresentou estável quanto a sua precisão nas previsões, o que possibilita trabalhar com as quantidades de inadimplentes previstos para o próximo pagamento de uma fatura de cartão de crédito como um indicador preventivo para uma gestão de risco mais assertiva a ser aplicada no setor de crédito da instituição financeira. Para essa pequena amostra o modelo não se adequou totalmente para prever os padrões quanto ao cliente ser inadimplente pois acertou apenas 37,93% do total de não pagantes quando executado no subconjunto de teste, todavia se mostrou muito melhor no subconjunto de treino com 64,55% de acerto.

Mesmo considerando os fatores positivos dos resultados não se pode descrever essa solução de forma binária para o problema exposto, seja que por um lado atingiu bons resultados de acurácia e de outro obteve baixos acertos de maus pagadores, não é assertivo afirmar que o problema estará resolvido somente com esses procedimentos pois trabalhar com essa área de inteligência artificial com

certeza exigiria uma abrangência maior nos procedimentos para concluir um completo exercício do Aprendizado de Máquina. Acredito que de acordo com esse problema de negócio, um classificador seria satisfatório se tivesse um valor de *Sensitivity* mais alto porque a empresa estaria mais interessada em conhecer os aspectos positivos reais, ou seja, o número de clientes que provavelmente seria inadimplente no mês seguinte.

Contudo ressalta-se que este estudo se limitou na exploração de variáveis que melhor poderiam relacionar o modelo para as previsões uma vez que a adequação ou redução de variáveis conseguiriam trazer ganhos consideráveis de predição.

Espera-se que essa pesquisa possa colaborar para melhorar as análises relacionadas com problemas de risco de crédito. Ainda como contribuição do presente estudo em termos de pro atividade, acredito estimular a importância do conhecimento e alerta aos devedores para os problemas que podem ser originados a partir de créditos concedidos equivocadamente.

6.1 TRABALHO FUTURO

Como esse trabalho refere-se a uma área da ciência de dados e possui certa complexidade e grande abrangência, para estudos futuros sugere-se explorar outros modelos de AM a serem utilizados como alternativa para classificação. Ainda assim, a qualidade dos resultados obtidos pelo pré-processamento de dados é amplamente determinada pela qualidade dos dados de entrada, o que torna indispensável uma pesquisa e planejamento mais focada nessa fase crítica do processamento, pois requer mais tempo e conhecimento de domínio para sua correta realização e por fim os modelos podem ser comparados com maior precisão de todo o processo de Aprendizado de Máquina.

Outro ponto a ser tratado no uso do modelo proposto seria realizar o processamento de dados com um *dataset* de volume maior, pois o algoritmo pode

aprender com uma variedade muito maior de exemplos e com isso o resultado da comparação da relação entre as variáveis evidenciaria uma diferença maior ainda no grau de importância e com isso seria possível descartar algum atributo com mais certeza para construir novos modelos e agregar melhores soluções para o problema.

REFERÊNCIAS

GAMA, F. L. et al. **Inteligência Artificial:** Uma Abordagem de Aprendizado de Máquina. Rio de Janeiro: Grupo Gen - LTC, 2011.

MUELLER, John Paul; MASSARON, Luca. **Aprendizado de Máquina Para Leigos**. Rio de Janeiro: Alta Books Editora, 2019.

ANGELO, Claudio Felisoni de; BELTRAME, Nelson Bruxellas; DIAS, Manuel Martins. **Orçamento de lucros e perdas e concessão de crédito Vol.4**: Planejamento orçamentário e análise de riscos. São Paulo: Saint Paul, 2020.

NUNES, Rodrigo Escobar. **A atuação da recuperação de crédito na gestão de risco de crédito: um estudo de caso no Sicred – Sistema de Crédito Cooperativo.** 2013. 66 f. Trabalho de conclusão de curso (Graduação Bacharel em Administração) – Universidade Federal do Rio Grande do Sul, Porto Alegre, 2013.

ABUD, Luciana de Melo e. **Modelos computacionais prognósticos de lesões traumáticas do plexo braquial em adultos.** 2018. 133 f. Dissertação (Mestre em Ciências) – Instituto de Matemática e Estatística da Universidade de São Paulo, São Paulo, 2018.

SILVA, Luiz Otávio Lamardo Alves. **Classificação visual de mudas de plantas ornamentais: Análise da eficácia de técnicas de seleção de atributos.** 2014. 66 f. Dissertação (Mestre em Ciências) – Escola Politécnica da Universidade de São Paulo, São Paulo, 2014.

JUNIOR, João Carlos Pacheco. **Modelos para detecção de fraudes utilizando técnicas de aprendizado de máquina.** 2019. 103 f. Dissertação (Mestre em economia) – Fundação Getúlio Vargas Escola de Economia de São Paulo, São Paulo, 2019.

ARAÚJO, A. E.; CARMONA, M. de U. C. Desenvolvimento de modelos credit scoring com abordagem de regressão logística para a gestão da inadimplência de uma

instituição de microcrédito. **Contabilidade Vista & Revista.** Minas Gerais, v. 18, n. 3, p. 107-131, jul./set., 2007.

CAMARGOS, de A. M.; ARAÚJO, T. A. E.; CAMARGOS, S. C. M. A inadimplência em um programa de crédito de uma instituição financeira pública de Minas Gerais: Uma análise utilizando regressão logística. **Rege**. São Paulo, v. 19, n. 3, p. 473-492, jul./set., 2012.

CORREIA, Catarina Inês Couto. **Previsão da compra de serviços em seguros de vida usando Data Mining**. 2019. 87 f. Dissertação - Mestrado em Engenharia Matemática, Faculdade de ciências Universidade do Porto, Porto, 2019.

Conheça 5 exemplos de sucesso com o Big Data nas empresas. IBE. São Paulo, abr. 2018. Disponível em: https://www.ibe.edu.br/conheca-5-exemplos-de-sucesso-com-o-big-data-nas-empresas/. Acesso em: 16 mar. 2020.

The R Project for Statistical Computing. R-project. Disponível em: https://www.r-project.org. Acesso em: 18 mar. 2020.

R Studio Desktop. Rstudio. Disponível em: https://rstudio.com/products/rstudio/#rstudio-desktop. Acesso em: 18 mar. 2020.

Available CRAN Packages By Date of Publication. CRAN.2020. Disponível em: https://cran.r-project.org/web/packages/available_packages_by_date.html. Acesso em: 18 mar. 2020.