Identificador de Evasão de Clientes para Banco Digital

Luciano Munhoz Silva
Miguel de Oliveira Costa
Carlos Henrique Valério de Moraes (Orientador)

Resumo — A evasão de clientes, no inglês *churn*, é uma métrica empregada pelas empresas para avaliar problemas em suas operações. Além disso, a perda de assinaturas resulta em custos, pois fidelizar clientes é menos oneroso que captar novos. Portanto, no setor bancário esse fator ainda é crítico. Dessa forma, o objetivo do trabalho é identificar futuros cancelamentos de contas, possibilitando a realização de ações para reter os clientes que viriam a encerrar a conta no futuro. Para isso, foi utilizado como ferramenta o aprendizado de máquina, do inglês *machine learning*, para treinar algoritmos capazes de identificar padrões nas características dos clientes e de realizar a predição sobre a saída dos clientes. O melhor algoritmo treinado obteve uma acurácia de 100% na identificação dos clientes que viriam a encerrar a conta no futuro.

Palavras-Chave — Evasão de clientes, previsão, identificação de padrões, aprendizado de máquina.

Abstract — Churn is a metric used by companies to assess problems in their operations. In addition, the loss of subscriptions results in costs, as retaining customers is less expensive than acquiring new ones. Therefore, in the banking sector this factor is still critical. In this way, the objective of the study is to identify future account cancellations, making it possible to carry out actions to retain customers who would close the account in the future. For this, machine learning was used as a tool to train algorithms capable of identifying patterns in customer characteristics and predicting customer exit. The best trained algorithm was 100% accurate in identifying customers who would deactivate their accounts in the future.

Keywords — Churn, predict, pattern identification, machine learning.

I. Introdução

Evitar a evasão de clientes no setor bancário é um trabalho de grande importância para os resultados de um banco. De acordo com a pesquisa de *Reichheld* e *Sasser*, uma redução de 5% da evasão de clientes pode aumentar em até 85% o lucro de um banco [1]. A importância é ainda maior quando falamos de um banco digital, pois a quantidade de clientes é um grande indicador para demonstrar a eficiência do banco e, além disso, aumenta o potencial de capitalização da instituição. Por exemplo, o Nubank possui 59,6 milhões de clientes [2] e um valor de mercado de aproximadamente R$ 85,7 bilhões nas cotações do dia 21/05/2022 [3], enquanto o Banco Inter, que tem 18,6 milhões de clientes [4], possui um valor de mercado de R$ 13,2 bilhões nas cotações do dia 21/05/2022 [5]. Com isso, é possível observar o tamanho da importância da quantidade de clientes de um banco digital.

Nesse sentido, estudar mecanismos que possam contribuir na questão do *churn*, termo em inglês que se refere à evasão de clientes, possuem um alto potencial financeiro. Além disso, num ambiente de transformação digital que o Brasil se encontra e, com isso, a chegada de novas empresas no setor bancário, a concorrência está crescendo. Esse fato requer decisões estratégicas mais elaboradas para se destacar. Com o avanço da tecnologia, novas metodologias passaram a ser viabilizadas com baixo custo para refinar a maneira com que a evasão de clientes é abordada [6].

Nesse contexto, o uso de técnicas de inteligência artificial, como o aprendizado de máquina, do inglês *machine learning*, pode ajudar a estabelecer um padrão para os clientes que possuem a intensão de encerrar suas contas bancárias.

Para que essas técnicas possam ser utilizadas, é importante ter uma base de dados equilibrada para que o algoritmo possa ser mais assertivo. Portanto, a primeira etapa é certificar-se de que uma base de dados não esteja desequilibrada, para que o algoritmo não seja tendencioso. Isso pode ser feito através da análise dos dados que serão usados para aplicar um algoritmo de aprendizado de máquina. Caso a base não esteja equilibrada, será preciso manipular a base de dados com técnicas como a subamostragem, do inglês *undersampling*, ou a sobre-amostragem, do inglês *oversampling*. Além disso, nessa etapa também é feito um tratamento dos dados, de forma com que sejam selecionados os dados relevantes para o estudo [7].

Uma vez que a base de dados esteja pronta, a próxima etapa é o treinamento dos algoritmos de aprendizado de máquina. O algoritmo tem como objetivo analisar o padrão de comportamento dos clientes no banco para categorizá-los corretamente entre aqueles que devem permanecer com suas contas e aqueles que devem encerrar suas contas. Para isso, o algoritmo reserva uma parcela dos dados para controle da acurácia. Em seguida, com outra parcela, realiza iterações para atingir o melhor resultado possível dentro de um limite de processamento pré-estabelecido [8].

Para validar o experimento, foi comparada a performance de treinamento dos algoritmos de aprendizado de máquina escolhidos com técnicas já consolidadas, como a aprendizagem profunda (*deep learning*, em inglês). A aprendizagem profunda é um algoritmo de aprendizado de máquina que atua

com diversas camadas de aprendizado. Sendo assim, seu teste também foi realizado com um limite de processamento pré-determinado [9].

Por fim, com um algoritmo treinado, será realizado uma aplicação prática do algoritmo. O algoritmo será utilizado na base de dados de um banco digital para que o experimento pudesse ser validado em um ambiente real.

A. Objetivos

Neste projeto, será desenvolvido um identificador de evasão de clientes que utilizará técnicas de aprendizado de máquina, com o objetivo de indicar quais clientes possuem uma tendência para encerrarem suas contas bancárias. Logo, o banco poderá agir de forma prévia nesses clientes já mapeados para aumentar a retenção dos mesmos.

II. Fundamentação Teórica

A. Aprendizado de Máquina

O método de aprendizado de máquina consiste na análise de dados com base em modelos analíticos, de forma a automatizar a construção desses modelos. O aprendizado de máquina é um ramo dentro do estudo da inteligência artificial que se baseia na ideia de que sistemas podem aprender com dados e com o mínimo de intervenção humana [10]. A concepção do aprendizado de máquina atual evoluiu bastante nos últimos anos, esse avanço se deu graças à melhora do processamento computacional o que possibilitou no desenvolvimento de modelos cada vez mais complexos e precisos. Um dos pontos chave da técnica vem da capacidade de adaptação da máquina quando os dados de entrada do sistema podem mudar de padrão e agir de forma aleatória, o que normalmente acontece na natureza. Assim a máquina aprende por si só o novo padrão e como reagir a ele. Por conta disso a técnica é recomendada em diversas áreas do conhecimento como medicina, finanças, engenharia, segurança, etc [11].

Os dois métodos mais utilizados de aprendizado de máquina são o aprendizado supervisionado e o aprendizado não-supervisionado. O aprendizado supervisionado se dá quando o modelo já sabe qual deve ser seu resultado de saída de acordo com cada entrada. Com isso o modelo consegue medir o erro e ir se reajustando para obter resultados mais precisos, esse método é bastante empregado em problemas de previsão, utilizando dados passados para prever acontecimentos futuros. Já no aprendizado não-supervisionado não existem resultados pré-definidos, ou seja, o modelo é responsável por definir sua análise identificando semelhanças entre os dados, e assim, classificando-os em grupos com base em suas características. Normalmente o aprendizado não-supervisionado é utilizado quando não se tem todos os dados necessários para obter o resultado desejado, e assim, o operador recebe o auxílio da ferramenta para tomar suas decisões [12].

Existe também o método de aprendizagem por reforço, nesse método o sistema de inteligência artificial recebe recompensas ou penalidades por suas ações e, por meio de tentativa e erro, busca maximizar sua recompensa e diminuir suas penalidades, para assim atingir sua otimização. Mesmo sendo um método de tentativa e erro, o programador não fornece dicas ao programa, apenas define as recompensas e penalidades para que o programa se guie sozinho a partir de tentativas aleatórias. Portanto após varias tentativas o programa tende a alcançar seu máximo potencial, mas podendo cair em máximos locais e não encontrar o real máximo potencial do sistema.

Neste trabalho optamos por utilizar técnicas de aprendizado supervisionado. O objetivo é utilizar técnicas modernas e que tenham alta performance para obter resultados ótimos em nossas análises.

B. Técnicas de Aprendizado de Máquina

1) Classificador por Árvore de Decisão: O modelo de classificação por Árvore de Decisão (*Decision Tree*, em inglês) funciona através de uma amostra desconhecida que é classificada em uma classe. Essa estratégia envolve uma ou várias funções de decisão sucessivas para avaliar os melhores resultados, e as classes são agrupadas em um diagrama que representa a árvore. A classificação por Árvore de Decisão consiste em um nó raiz, seguido por nós interiores que são ligados em estágios de decisão e terminados em nós folhas. A distância de um nó até a raiz é chamada de camada. Sendo assim, todos os nós que possuem a mesma distância até a raiz se encontram em uma mesma camada [13]. Para a execução, esse classificador utiliza um conjunto de treinamento e um conjunto de teste para verificar a precisão do algoritmo, como indicado na Figura 1.

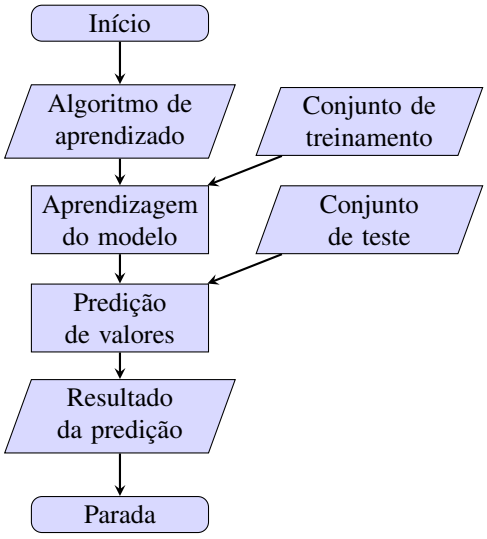


Figura 1. Classificação e treinamento

Essa técnica possui uma fraqueza em sua estrutura de classificação que é a demanda de processamento para classificar um novo valor. O custo computacional aumenta conforme aumentam o número de camadas na árvore, pois para classificar uma consulta de entrada, todos os componentes do classificador devem ser processados [14].

2) Classificador por Floresta Aleatória: A proposta do classificador por Floresta Aleatória (*Random Forest*, em inglês) consiste em combinar modelos aleatórios de aprendizagem baseados na Árvore de Decisão. Essa estratégia recebe

o nome de *Bagging* [15]. Nesse sentido, a combinação de modelos diferentes de aprendizado de máquina trazem mais robustez e precisão para os resultados. Outra vantagem desse classificador é a capacidade de rodar de forma eficaz para uma grande quantidade de dados [16]. Na prática, a classificação por Floresta Aleatória utiliza diversas árvores para realizar o treinamento do algoritmo.

Um ponto importante é que esse modelo de classificação requer que as amostras de treino estejam balanceadas para diminuir a sensibilidade do classificador em relação às variáveis selecionadas [17].

3) Classificador por K-Vizinhos Mais Próximos: O modelo de classificação de K-Vizinhos Mais Próximos (K-*Neighbors*, em inglês) é um método de aprendizado de máquina supervisionado que funciona através do cálculo das distâncias entre pontos de dados não classificados, denominado conjunto de treinamento, com um grupo de dados já classificados, denominado conjunto de teste. A distância pode ser calculada através da Distância Euclidiana (Equação 1), onde p e q são os pontos de dados a serem analisados.

$$d\left(p,q\right)=\sqrt{\sum_{i=1}^{n}\left(q_i-p_i\right)^2} \tag{1}$$

A classificação e treinamento desse algoritmo consiste em selecionar os dados com menor distância dos dados de referência e classificá-los em classes. Então, essas classes são comparadas para avaliar os melhores resultados [18]. Uma vantagem desse classificador é a robustez contra ruído nos dados. Entretanto, ele requer muita memória e processamento para realizar a classificação de banco de dados grandes e para classificar dados com muitas dimensões [19].

4) Classificador por Regressão de Cume: O classificador por Regressão de Cume (*Ridge*, em inglês) é um modelo de aprendizado supervisionado que reduz para próximo de zero os coeficientes de regressão, através da penalização dos coeficientes quadrados. A regularização de coeficientes é conhecida como norma L2. Uma das vantagens do modelo de regressão por Cume é que ele tem um bom desempenho quando comparado com o método dos mínimos quadrados em situações de uma quantidade grande de dados multivariados [20].

5) Classificador Passivo-Agressivo: O algoritmo de aprendizado Passivo-Agressivo (*Passive Aggressive*, em inglês) observa as instâncias de maneira sequencial. E, a cada observação, o algoritmo prevê um resultado. A decisão do resultado, quando se trata de uma classificação binária, é uma escolha de sim ou não. Uma vez que o algoritmo define uma previsão, ele recebe um *feedback* que indica se o resultado está correto ou não. Então, o classificador passivo agressivo pode modificar seu mecanismo de previsão, de acordo com o *feedback* recebido. Dessa forma, ele irá melhorar as chances de fazer uma previsão mais precisa sobre as iterações subsequentes [21].

C. Balanceamento de Bases

O bom desempenho de um algoritmo de aprendizado de máquina está atrelado à uma base de dados que proporcione um universo de dados que não trarão conclusões enviesadas. Sendo assim, a análise sobre esse quesito é crucial para que o experimento seja válido. Afinal, em um cenário no qual a base de dados não está balanceada, o treinamento do algoritmo pode exibir resultados errôneos, ainda que o algoritmo tente otimizar seu desempenho a cada iteração. Nesses casos, deve ser feito um balanceamento na base de dados antes de iniciar o treinamento do algoritmo.

*1) **Subamostragem**:* O método de balanceamento chamado subamostragem (*Undersampling*, em inglês) consiste na redução dos dados que estão em maior número. Esse método pode ser aplicado de forma manual, com a remoção dos dados direto na base de dados ou de forma aleatória, através da criação de um algoritmo que remova aleatoriamente os dados em maior número. Vale ressaltar que, como esse método reduz a base de dados, é necessário avaliar se não houve perda de dados que seriam relevantes para o aprendizado de máquina.

*2) **Sobre-amostragem**:* O método de balanceamento chamado sobre-amostragem (*Oversampling*, em inglês) consiste no aumento dos dados que estão em menor número. A execução desse método requer que os dados existentes em menor número sejam clonados de forma aleatória. Essa técnica funciona bem com situações em que o parâmetro que está sendo balanceado não possui uma variação quantitativa grande.

D. Métricas

A predição dos algoritmos é focada em atingir os melhores resultados para determinadas métricas. Portanto, é necessário entender o que cada métrica representa e qual se enquadra melhor no problema proposto:

1) Acurácia: A Acurácia (*Accuracy*, em inglês) representa a performance geral do modelo, ou seja, dentre todas as classificações, quantas o modelo classificou corretamente. Por ser uma métrica geral, pode-se utilizá-lo para identificar a qualidade do modelo para o problema em questão. Porém, para que o melhor modelo seja utilizado, é importante analisar também a *Precisão* ou o *Reposição*.

2) Precisão: A Precisão (*Precision*, em inglês) indica a assertividade do modelo quando ele classifica o indivíduo como Positivo. Ou seja, dentre todas as classificações "Positivas"que o modelo fez, quantas de fato são Positivas. A precisão costuma ser utilizada em problemas que Falsos Positivos são mais prejudiciais que os Falsos Negativos. Por exemplo, um modelo utilizado para direcionar anúncios de *marketing*, nesse cenário, se o modelo apresentar Falso Positivo, a empresa estará utilizando mal seu capital, já um Falso Negativo, apenas fará com que o anúncio não seja feito, sem perda monetária.

3) Reposição: A Reposição (*Recall*, em inglês) indica a assertividade do modelo quando ele deveria classificar os indivíduo como Positivo. Essa métrica costuma ser utilizado quando os Falsos Negativos são mais prejudiciais que os Falsos Positivos. Essa é a métrica que melhor representa o cenário tratado neste artigo, já que deixar de tomar alguma ação com um cliente que vai encerrar a conta é mais prejudicial do que realizar uma ação com o cliente que não vai encerrar.

4) Métrica F: A Métrica F (*F1-Score*, em inglês) é utilizada para identificar se a *Precisão* ou a *Reposição* está baixa, já

que é calculado a partir da média harmônica entre os dois indicadores.

III. Desenvolvimento

Neste trabalho foi utilizado uma base de dados de um banco europeu com dados de dez mil clientes, disponibilizado pela plataforma *Kaggle* [22]. Essa base de dados contém clientes que estão retirando sua conta do banco devido a alguma possível perda e outros possíveis problemas. Para o desenvolvimento do estudo foi realizado um tratamento desse banco de dados e, então, foram aplicados algoritmos de aprendizado de máquina aos dados tratados, a fim de obter informações úteis sobre como reter os clientes no banco. Além disso, a performance do modelo de Aprendizado de Máquina varia para cada problema e, por conta disso, precisamos utilizar diversos modelos para ver qual se aplica melhor ao problema proposto. Por fim, foram utilizados buscadores de parâmetros para otimizar o resultado dos algoritmos. O desenvolvimento também pode ser visualizado pelo repositório [23] que contém o ambiente no qual foi realizado o desenvolvimento deste trabalho.

A. Base de Dados para Treinamento

A primeira etapa do algoritmo consistiu em importar o banco de dados. Essa tarefa foi feita através do método *read_csv* da biblioteca *pandas* para importar o banco de dados do *Gitub* no formato CSV *(comma-separated values)*. Com isso, os dados ficam disponíveis para serem manipulados no ambiente de desenvolvimento.

Em seguida, os dados foram tratados para que houvesse um equilíbrio entre os dois grupos principais na observação do experimento: os clientes que permaneceram e os clientes que saíram do banco. Originalmente, o banco de dados usado é composto por 7.963 clientes que permanecem no banco e 2.037 clientes que encerraram suas contas (Figura 2). Por conta dessa disparidade no número entre esses dois grupos, foi aplicada a técnica de sobre-amostragem. Sendo assim, a base de dados passou a conter 7.963 clientes que encerraram suas contas, totalizando um universo de 15.926 clientes.

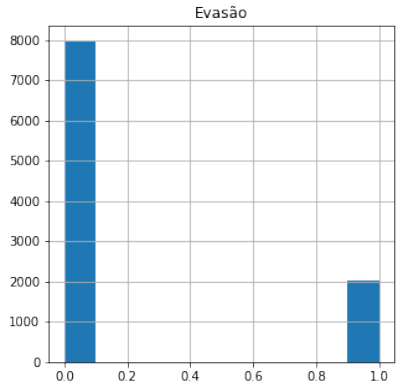


Figura 2. Base de dados desbalanceada com apenas 20% clientes evadindo.

Antes de realizar o treinamento dos algoritmos foram feitas análises em cima da base de dados. Com isso, foi possível confirmar se os procedimentos realizados para balancear as bases não estava prejudicando a estrutura da base de dados.

B. Análise dos dados

Com a análise foi possível destacar quais características tem maior impacto na decisão do cliente. Ao analisar os histogramas de crédito bancário (Figura 3) e a idade (Figura 4), fica claro que os fatores mais relevantes para o encerramento da conta são a idade do cliente e o crédito bancário liberado, já que existem padrões nesses gráficos quando os dados são divididos entre contas ativas e contas encerradas. O mesmo não ocorre para o gráfico de saldo em conta (Figura 5), por exemplo, indicando que essa informação não é determinante para manter a conta ativa.

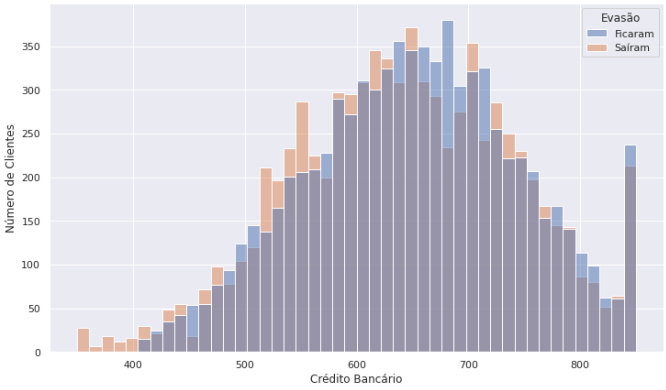


Figura 3. Crédito bancário

O histograma de crédito bancário (Figura 3), é possível observar que os clientes com menos de 400 dólares em crédito encerram suas contas.

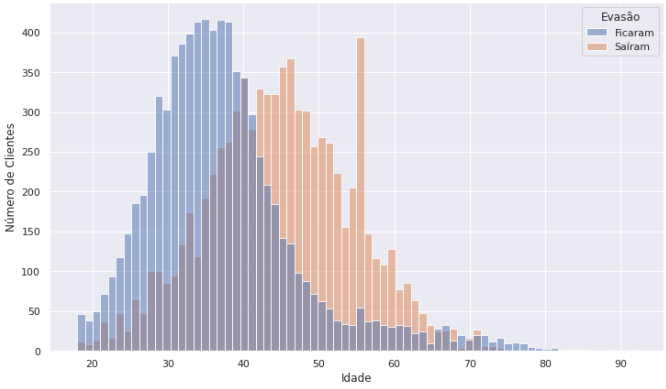


Figura 4. Idade do cliente

Com o histograma da idade (Figura 4), é possível notar que quanto maior a idade do cliente, maior a tendência de encerrar a conta.

Com relação ao saldo em conta, representado na Figura 5, não é possível tomar nenhuma conclusão, já que a curva de quem encerrou a conta é similar a de quem manteve ativa.

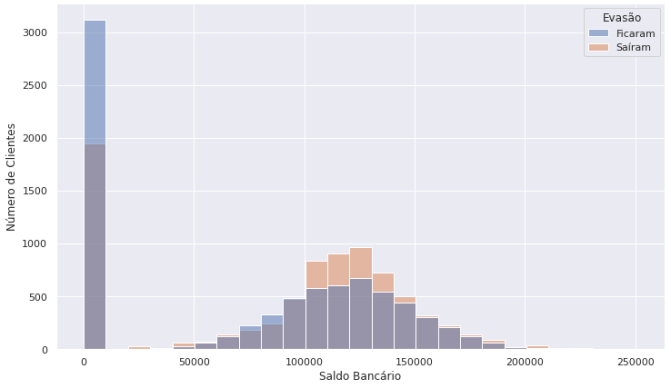


Figura 5. Saldo bancário

C. Algoritmos de Aprendizado de Máquina

Após realizar a análise e identificar quais dados mais impactam a saída do cliente, deve-se iniciar o treinamento da inteligência artificial com diversos modelos, para utilizar o de melhor desempenho para o problema em questão. Dessa forma, foram utilizados os modelos:

- *Classificador por Árvore de Decisão*
- *Classificador por Floresta Aleatória*
- *Classificador por K-Vizinhos Mais Próximos*
- *Classificador por Regressão de Cume*
- *Classificador Passivo-Agressivo*

Desses, o que obteve melhor desempenho foi a Classificação por K-Vizinhos Mais Próximo, e por conta disso, é o modelo que será utilizado para a aplicação na base de dados do banco escolhido para validar o modelo. Além disso, foi utilizado o método de Aprendizado Profundo para comparação de resultados, já que trata-se de um método de aprendizado de máquina muito difundido pela sua eficácia.

D. Buscadores

A classificação realizada pelos algoritmos teve os parâmetros de cada algoritmo ajustados manualmente. Sendo assim, ainda existe a possibilidade de otimizar o resultado obtido. Essa otimização pode ser feita através dos buscadores, que são responsáveis por testar combinações de parâmetros de forma mais eficiente do que o método manual feito previamente. Os buscadores utilizados neste trabalho foram:

1) Busca por Grade: A técnica da Busca por Grade (*GridSearchCV*, em inglês) consiste em uma busca exaustiva de valores de um parâmetro específico para um classificador. Os parâmetros são otimizados por meio de uma busca de grade de validação cruzada, dado uma grade de parâmetros.

2) Busca Aleatória: A técnica de Busca Aleatória (*RandomizedSearchCV*, em inglês) consiste em testar parâmetros com base em uma quantidade pré-determinada de variações de valores para os parâmetros. Assim como a Busca por Grade, a otimização também é feita por validação cruzada, dada uma grade de parâmetros.

IV. Testes e Resultados

Os testes realizados com as técnicas de aprendizado de máquina supervisionado estão apresentados nas Tabelas I à V.

	Precisão	Reposição	Métrica F
Ficou	0.51	0.84	0.64
Saiu	0.55	0.20	0.29
Acurácia	0.52	0.52	0.52
Média macro	0.53	0.52	0.46
Tempo [s]	0.06	0.06	0.06

Tabela I
Classificador Passivo-Agressivo

	Precisão	Reposição	Métrica F
Ficou	0.70	0.72	0.71
Saiu	0.71	0.69	0.70
Acurácia	0.70	0.70	0.70
Média macro	0.70	0.70	0.70
Tempo [s]	0.01	0.01	0.01

Tabela II
Classificador por Regressão de Cume

	Precisão	Reposição	Métrica F
Ficou	0.77	0.59	0.67
Saiu	0.67	0.83	0.74
Acurácia	0.71	0.71	0.71
Média macro	0.72	0.71	0.70
Tempo [s]	0.03	0.03	0.03

Tabela III
Classificador por K-Vizinhos Mais Próximos

	Precisão	Reposição	Métrica F
Ficou	0.77	0.78	0.77
Saiu	0.77	0.76	0.77
Acurácia	0.77	0.77	0.77
Média macro	0.77	0.77	0.77
Tempo [s]	0.05	0.05	0.05

Tabela IV
Classificador por Árvore de Decisão

	Precisão	Reposição	Métrica F
Ficou	0.77	0.82	0.79
Saiu	0.81	0.75	0.78
Acurácia	0.79	0.79	0.79
Média macro	0.79	0.79	0.79
Tempo [s]	1.42	1.42	1.42

Tabela V
Classificador por Floresta Aleatória

O teste realizado com a técnica de aprendizagem profunda está apresentado na Tabela VI.

Após a realização dos testes feitos com os algoritmos de aprendizado de máquina, foi feita uma otimização com os buscadores. E os resultados dessa otimização, aplicada nos três melhores algoritmos, estão representados nas tabelas VII à XII.

	Precisão	Reposição	Métrica F
Ficou	0.79	0.87	0.83
Saiu	0.85	0.77	0.81
Acurácia	0.82	0.82	0.82
Média macro	0.82	0.82	0.82
Tempo [s]	24.33	24.33	24.33

Tabela VI
Aprendizagem Profunda

	Precisão	Reposição	Métrica F
Ficou	0.94	0.92	0.93
Saiu	0.92	0.94	0.93
Acurácia	0.93	0.93	0.93
Média macro	0.93	0.93	0.93
Tempo [s]	2307.86	2307.86	2307.86

Tabela VII
Busca por Grade aplicada na Floresta Aleatória

	Precisão	Reposição	Métrica F
Ficou	0.80	0.82	0.81
Saiu	0.82	0.80	0.81
Acurácia	0.81	0.81	0.81
Média macro	0.81	0.81	0.81
Tempo [s]	2.68	2.68	2.68

Tabela VIII
Busca por Grade aplicada na Árvore de Decisão

	Precisão	Reposição	Métrica F
Ficou	1.00	1.00	1.00
Saiu	1.00	1.00	1.00
Acurácia	1.00	1.00	1.00
Média macro	1.00	1.00	1.00
Tempo [s]	189.56	189.56	189.56

Tabela IX
Busca por Grade aplicada no K-Vizinhos Mais Próximos

	Precisão	Reposição	Métrica F
Ficou	0.94	0.91	0.93
Saiu	0.91	0.94	0.93
Acurácia	0.93	0.93	0.93
Média macro	0.93	0.93	0.93
Tempo [s]	73.57	73.57	73.57

Tabela X
Busca Aleatória aplicada na Floresta Aleatória

	Precisão	Reposição	Métrica F
Ficou	0.80	0.82	0.81
Saiu	0.82	0.80	0.81
Acurácia	0.81	0.81	0.81
Média macro	0.81	0.81	0.81
Tempo [s]	0.66	0.66	0.66

Tabela XI
Busca Aleatória aplicada na Árvore de Decisão

Na figura 6 é possível verificar a taxa de verdadeiro positivo e a taxa de falso positivo com a utilização do algoritmo Classificador por K-Vizinhos Mais Próximos combinado com a Busca por Grade. A Matriz de Confusão resultante dessa predição está representada na tabela XIII.

	Precisão	Reposição	Métrica F
Ficou	0.99	0.88	0.93
Saiu	0.89	0.99	0.94
Acurácia	0.94	0.94	0.94
Média macro	0.94	0.94	0.94
Tempo [s]	21.65	21.65	21.65

Tabela XII
Busca Aleatória aplicada no K-Vizinhos Mais Próximos

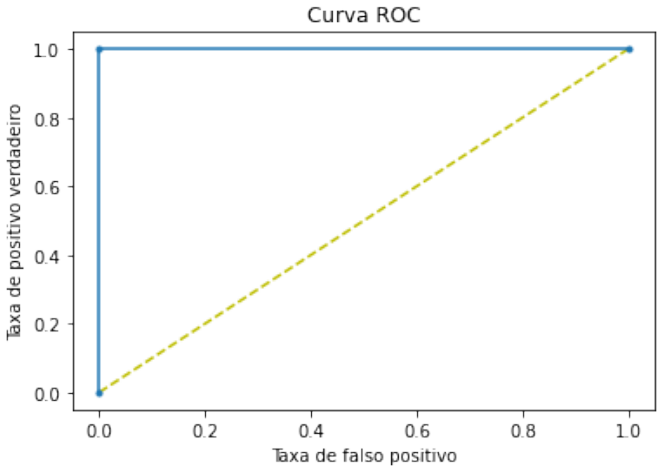


Figura 6. *Curva ROC*

		Predito	
		Positivo	Negativo
Real	Positivo	7963	0
	Negativo	0	7963

Tabela XIII
MATRIZ DE CONFUSÃO

V. CONCLUSÃO

A base de dados de evasão apresentou características que foram melhor identificadas pela técnica ajustada de classificação do voto dos K-Vizinhos Mais Próximos (*K-Neighbors Classifier*) apresentando acurácia, precisão, reposição e métrica F iguais a 100% (Tabela IX) para a média nos valores dos conjuntos de testes obtidos do selecionador de modelos por k-Fold de grupos não sobrepostos. As demais técnicas apresentaram resultados acima de 80% com tempos de treinamento reduzido, mas no contexto de utilização periódica, não contínua, não é necessário velocidade na obtenção do modelo de identificador para classificar novos clientes que desejam cancelar sua assinatura.

Destaca-se aqui a importância no ajuste das técnicas de aprendizado de máquina, devido a diferença de qualidade do modelo obtido antes (Tabela III) e depois do seu ajuste de parâmetros para treinamento (Tabela IX). Essa melhoria garantiu a obtenção do melhor modelo incontestável em relação às demais técnicas avaliadas.

A. Trabalhos Futuros

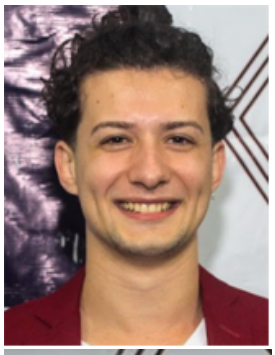
Ainda existe espaço para a utilização de aprendizados de máquina não supervisionado para detecção de anomalias para a identificação de evasão de clientes, pois como o número de casos é significativamente menor que os clientes cativos, torna as técnicas de aprendizado supervisionado classificador problemático devido ao desequilíbrio de classes dessa base de dados.

Esse trabalho poderá ser integrado a um sistema de REST API para fornecer uma constante previsão de clientes que podem evadir, servindo um serviço essencial para o setor bancário, identificando continuamente a qualidade das operações realizadas na empresa.

AGRADECIMENTOS

Agradecemos ao professor Carlos Valério pela colaboração e orientação neste projeto e a todos professores que fizeram parte da nossa formação.

BIOGRAFIA



Luciano Munhoz Silva Nascido em Araçatuba (SP), em 1995. Ingressou na Universidade Federal de Itajubá em 2016, no curso de Engenharia de Controle e Automação. Foi membro do projeto Ex-Machina e membro fundador da empresa júnior Asimov Jr. Atuou como estagiário de engenharia na empresa Energia Automação no ano de 2021.

Miguel de Oliveira Costa Nascido em Borda da Mata (MG), em 1999. Ingressou na Universidade Federal de Itajubá em 2017, no curso de Engenharia de Controle e Automação, foi membro do Centro Acadêmico de Engenharia de Controle e Automação e da Diretor Comercial da Asimov Jr. Está atualmente trabalhando como Analista Jr. de Backoffice Onboarding no C6 Bank em São Paulo (SP).

REFERÊNCIAS

[1] F. F. Reichheld and W. E. Sasser, "Zero defeofions: Quoliiy comes to services," *Harvard business review*, vol. 68, no. 5, pp. 105–111, 1990.
[2] R. Nubank. Nu holdings ltd. divulga os resultados do primeiro trimestre de 2022. [Online]. Available: https://blog.nubank.com.br/resultados-financeiros-1o-tri-2022/
[3] Nu: Nu holdings ltd, cotação dividendos e indicadores. [Online]. Available: https://statusinvest.com.br/acoes/eua/nu
[4] Banco inter (bidi4) tem lucro líquido de r$27,5mi no 1t22, alta de 31,8%. [Online]. Available: https://www.suno.com.br/noticias/banco-inter-bidi4-lucro-liquido-27-mi-1t22/
[5] Bidi11: Banco inter unit: cotação e indicadores. [Online]. Available: https://statusinvest.com.br/acoes/bidi11
[6] P. R. d. Franceschi, "Modelagens preditivas de churn: o caso do banco do brasil," 2019.
[7] G. Barbosa, P. de Miranda, R. Mello, and R. Silva, "Sequenciamento de algoritmos de amostragem para aumentar o desempenho de classificadores em conjuntos de dados desequilibrados," in *Anais do XVI Encontro Nacional de Inteligência Artificial e Computacional*. SBC, 2019, pp. 413–423.
[8] T. M. Mitchell and T. M. Mitchell, *Machine learning*. McGraw-hill New York, 1997, vol. 1, no. 9.
[9] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.
[10] J. G. Carbonell, R. S. Michalski, and T. M. Mitchell, "An overview of machine learning," *Machine learning*, pp. 3–23, 1983.
[11] P. P. Shinde and S. Shah, "A review of machine learning and deep learning applications," in *2018 Fourth international conference on computing communication control and automation (ICCUBEA)*. IEEE, 2018, pp. 1–6.
[12] R. Saravanan and P. Sujatha, "A state of art techniques on machine learning algorithms: a perspective of supervised learning approaches in data classification," in *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*. IEEE, 2018, pp. 945–949.
[13] P. H. Swain and H. Hauska, "The decision tree classifier: Design and potential," *IEEE Transactions on Geoscience Electronics*, vol. 15, no. 3, pp. 142–147, 1977.
[14] S. B. Kotsiantis, "Decision trees: a recent overview," *Artificial Intelligence Review*, vol. 39, no. 4, pp. 261–283, 2013.
[15] M. Pal, "Random forest classifier for remote sensing classification," *International journal of remote sensing*, vol. 26, no. 1, pp. 217–222, 2005.
[16] V. F. Rodriguez-Galiano, B. Ghimire, J. Rogan, M. Chica-Olmo, and J. P. Rigol-Sanchez, "An assessment of the effectiveness of a random forest classifier for land-cover classification," *ISPRS journal of photogrammetry and remote sensing*, vol. 67, pp. 93–104, 2012.
[17] M. Belgiu and L. Drăguţ, "Random forest in remote sensing: A review of applications and future directions," *ISPRS journal of photogrammetry and remote sensing*, vol. 114, pp. 24–31, 2016.
[18] A. Giri, M. V. V. Bhagavath, B. Pruthvi, and N. Dubey, "A placement prediction system using k-nearest neighbors classifier," in *2016 Second International Conference on Cognitive Computing and Information Processing (CCIP)*. IEEE, 2016, pp. 1–4.
[19] A. A. Soofi and A. Awan, "Classification techniques in machine learning: applications and issues," *Journal of Basic & Applied Sciences*, vol. 13, pp. 459–465, 2017.
[20] C. Peng and Q. Cheng, "Discriminative ridge machine: A classifier for high-dimensional data or imbalanced data," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 6, pp. 2595–2609, 2020.
[21] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer, "Online passive aggressive algorithms," 2006.
[22] Bank customers churn. [Online]. Available: https://www.kaggle.com/datasets/santoshd3/bank-customers
[23] Repositório do notebook desenvolvido para identificar a evasão de clientes para banco digital. [Online]. Available: https://github.com/MunhozSilva/TFG