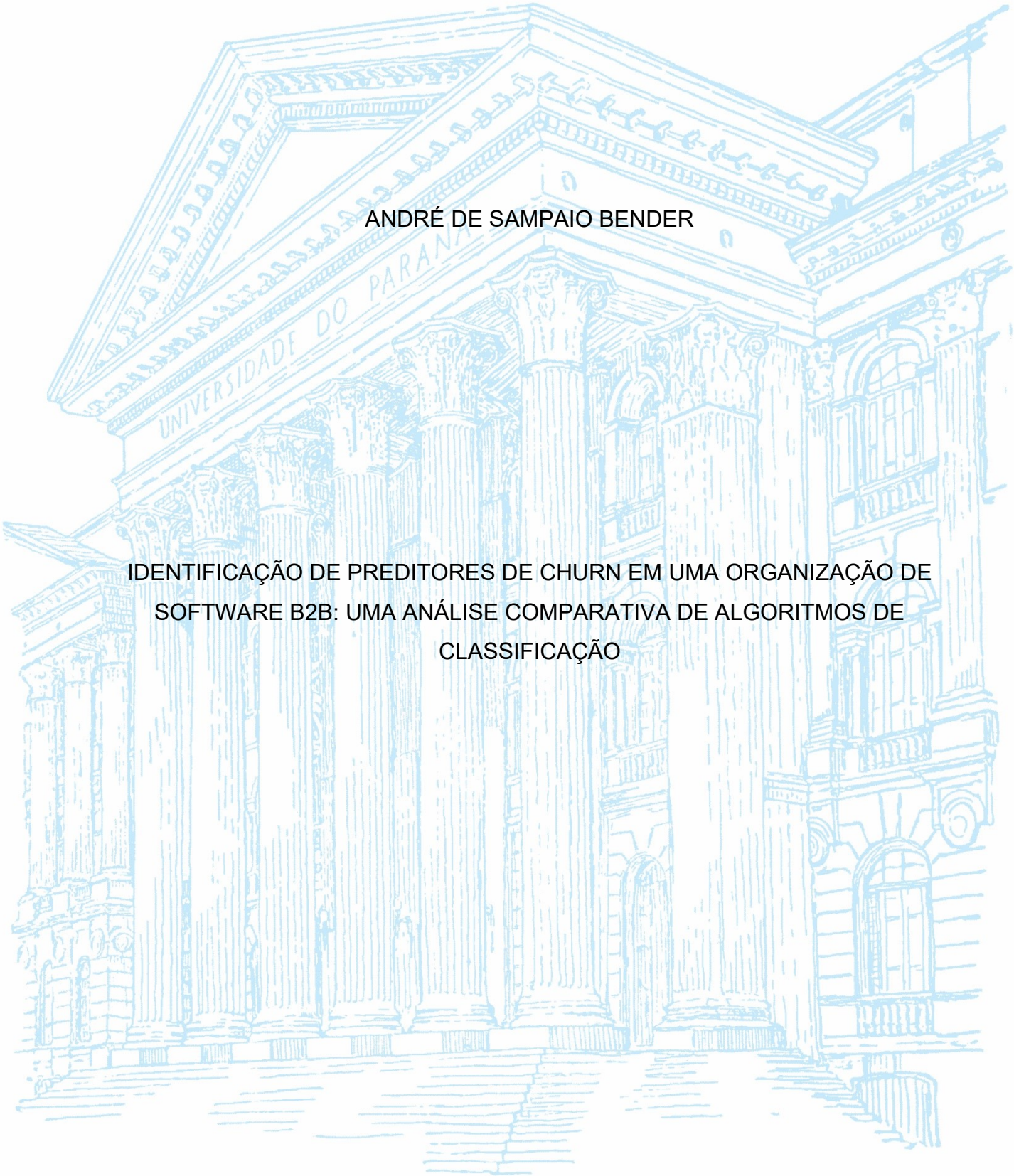
UNIVERSIDADE FEDERAL DO PARANÁ

ANDRÉ DE SAMPAIO BENDER



IDENTIFICAÇÃO DE PREDITORES DE CHURN EM UMA ORGANIZAÇÃO DE SOFTWARE B2B: UMA ANÁLISE COMPARATIVA DE ALGORITMOS DE CLASSIFICAÇÃO

CURITIBA

2024

ANDRÉ DE SAMPAIO BENDER

IDENTIFICAÇÃO DE PREDITORES DE CHURN EM UMA ORGANIZAÇÃO DE SOFTWARE B2B: UMA ANÁLISE COMPARATIVA DE ALGORITMOS DE CLASSIFICAÇÃO

Trabalho de Conclusão de Curso apresentado ao curso de Graduação em Gestão da Informação, Setor de Ciências Sociais Aplicadas, Universidade Federal do Paraná, como requisito parcial à obtenção do título de Bacharel em Gestão da Informação.

Orientadora: Prof. Dra. Denise Fukumi Tsunoda.

CURITIBA

2024

RESUMO

A predição de *churn* (cancelamento de serviços) em organizações é um tema crucial devido ao alto custo de aquisição de novos clientes, em comparação à retenção de clientes existentes. E o uso de algoritmos de *machine learning* para prever *churn* mostra-se uma abordagem eficaz. Esta pesquisa utiliza dados de uma organização de *Software as a Service* (Saas), que atua no mercado B2B de gestão de qualidade, no setor alimentício, para prever *churn* a partir de algoritmos de *machine learning*. O objetivo é identificar os principais preditores de cancelamento de assinaturas, a partir do uso de algoritmos de classificação. A metodologia da investigação é fundamentada em revisão sistemática da literatura científica, nas bases de dados Web of Science e Scielo. O resultado inicial de 6.516 artigos científicos, após aplicação de filtros, é reduzido a 32 artigos, que possibilitam o conhecimento e a comparação das práticas atuais sobre *churn* e a utilização de classificadores para a prevenção de *churn*; a análise de dados históricos de clientes atuais e que cancelaram os contratos na organização analisada; o desenvolvimento de um modelo quantitativo; e a avaliação dos classificadores e predicadores que influenciam no cancelamento de serviços. Para a coleta de dados, utilizam-se registros de clientes da organização estudada, analisados com técnicas de pré-processamento, balanceamento de classes, e modelos de *machine learning*, como Random Forest, Regressão Logística, SVM e Redes Neurais. A implementação segue o modelo CRISP-DM, incluindo as fases de entendimento do negócio, entendimento dos dados, preparação dos dados, modelagem e avaliação. Os resultados mostram que o modelo *Random Forest* apresenta o melhor desempenho em termos de acurácia, precisão, *recall* e *F1-score*. É possível determinar os comportamentos de clientes com maior influência no cancelamento ao aplicar *Random Forest*, ANOVA e Regressão Logística. Conclui-se que a utilização de algoritmos, como Random Forest, ANOVA e RFE, são capazes de identificar os principais preditores que influenciam no cancelamento dos serviços e, assim, contribuir à retenção de clientes em organização de SaaS, atuantes no mercado B2B.

Palavras-chave: previsão de churn; algoritmos de classificação; preditores de churn; machine learning; software as a service.

ABSTRACT

Predicting churn (service cancellation) in organizations is a crucial topic due to the high cost of acquiring new customers compared to retaining existing ones. The use of machine learning algorithms to predict churn has proven to be an effective approach. This research utilizes data from a Software as a Service (SaaS) organization operating in the B2B quality management market within the food sector to predict churn using machine learning algorithms. The objective is to identify the main predictors of subscription cancellations using classification algorithms. The investigation methodology is based on a systematic review of scientific literature from the Web of Science and Scielo databases. The initial result of 6,516 scientific articles, after applying filters, is reduced to 32 articles that provide knowledge and allow the comparison of current practices on churn and the use of classifiers for churn prevention; the analysis of historical data of current and former clients who canceled their contracts in the analyzed organization; the development of a quantitative model; and the evaluation of classifiers and predictors that influence service cancellations. For data collection, client records from the studied organization are used, analyzed with preprocessing techniques, class balancing, and machine learning models, including Random Forest, Logistic Regression, SVM, and Neural Networks. The implementation follows the CRISP-DM model, including the phases of business understanding, data understanding, data preparation, modeling, and evaluation. The results show that the Random Forest model presents the best performance in terms of accuracy, precision, recall, and F1-score. It is possible to determine the client behaviors with the greatest influence on cancellations by applying Random Forest, ANOVA, and Logistic Regression. It is concluded that the use of algorithms such as Random Forest, ANOVA, and RFE can identify the main predictors that influence service cancellations and thus contribute to client retention in SaaS organizations operating in the B2B market.

Keywords: churn prediction; classification algorithms; churn predictors; machine learning; software as a service.

LISTA DE FIGURAS

LISTA DE GRÁFICOS

LISTA DE QUADROS

LISTA DE TABELAS

LISTA DE ABREVIATURAS OU SIGLAS

ANOVA	- ANÁLISE DE VARIÂNCIA
AUC	- AREA UNDER THE CURVE
B2B	- BUSINESS TO BUSINESS
CART	- CLASSIFICATION AND REGRESSION
CRM	- CUSTOMER RELATIONSHIP MANAGEMENT
LIME	- LOCAL INTERPRETABLE MODEL-AGNOSTIC EXPLANATIONS
RFE	- ELIMINAÇÃO RECURSIVA DE CARACTERÍSTICAS
ROC	- RECEIVER OPERATING CHARACTERISTIC
RFE	- ELIMINAÇÃO RECURSIVA DE CARACTERÍSTICAS
ROC	- RECEIVER OPERATING CHARACTERISTIC
SAAS	- SOFTWARE AS A SERVICE
SHAP	- SHAPLEY ADDITIVE EXPLANATIONS
SVM	- SUPPORT VECTOR MACHINES

SUMÁRIO

1 INTRODUÇÃO

O cenário de *Software as a Service* (SaaS) – ou *Software* como Serviço, em português – expande-se rapidamente e oferece vantagens, como flexibilidade e escalabilidade, que são essenciais às organizações modernas.

Maheshwari, Toshniwal e Dubey (2020) defendem que *Software* como Serviço é como um modelo de distribuição de *software* em que programas são gerenciados por um fornecedor terceirizado e disponibilizados pela *web* para os clientes. Eles afirmam que no modelo SaaS não há necessidade de pagamento antecipado ou investimento em servidores, licenciamento de software e bancos de dados, pois o serviço é pago conforme o uso e acessado com uma conexão à internet.

Contudo, a retenção de clientes ainda é um desafio significativo no ambiente SaaS, levando a um fenômeno conhecido como *churn* em SaaS: percentual de clientes que cancelam seu serviço em um período específico (Strouse, 1999, p. 271).

Estudos recentes indicam que empresas SaaS enfrentam dificuldades com retenção de clientes devido a diversos fatores, como a competição acirrada e a rápida evolução das expectativas dos consumidores (Rivera *et al.*, 2020). Exemplos em escala global incluem as organizações Salesforce e Dropbox, que implementaram estratégias de retenção para mitigar taxas de *churn* elevadas (Hall, 2019).

Estratégias eficazes para retenção de clientes são importantes para empresas de *Software* como Serviço enfrentarem a alta competitividade do mercado. De acordo com Kumar e Petersen (2022), a personalização do serviço é uma das estratégias mais efetivas, pois permite que as empresas atendam às necessidades específicas de seus clientes, aumentando a satisfação e a lealdade. Outra estratégia é o fornecimento de um excelente suporte ao cliente, que ajuda a resolver problemas rapidamente e a melhorar a experiência do usuário (Hall, 2019). Estudos indicam que a análise preditiva e o uso de *big data* podem identificar padrões de comportamento que precedem o *churn*, permitindo intervenções proativas para manter os clientes (Rivera *et al.*, 2020). A combinação destas estratégias não apenas reduz as taxas de *churn*, mas também promove um crescimento sustentável às organizações SaaS (Hall, 2019; Rivera *et al.*, 2020).

Frente a isso existe o desafio de identificar quais os predicadores que mais influenciam o cancelamento de assinatura de clientes de uma organização de *software* no modelo *SaaS*. Assim, inicialmente é preciso conhecer os fatores que indicam um

possível cancelamento para posterior adoção de estratégias e medidas que mitiguem o *churn*.

1.1 PROBLEMATIZAÇÃO

Pesquisas indicam que adquirir um novo cliente pode ser de cinco a 25 vezes mais caro do que reter um cliente existente – a depender do setor. Isto ocorre porque manter um cliente satisfeito requer menos tempo e recursos que conquistar um novo. A investigação de Frederick Reichheld, da Bain & Company, mostra que aumentar as taxas de retenção de clientes em 5% pode elevar os lucros de 25% a 95% (Reichheld, 1996; Gallo, 2014).

Compreender as particularidades que podem influenciar a decisão de um cliente em cancelar sua assinatura torna-se imperativo à sobrevivência de organizações de SaaS, pois as assinaturas dos clientes envolvem complexas negociações. E a previsibilidade dos cancelamentos de assinaturas pode ser labiríntica devido à variedade de fatores envolvidos, que incluem desde a adequação às necessidades do cliente e satisfação com o produto e suporte, até mudanças no mercado (Singh; Samalia, 2014).

Adicionalmente, estudos demonstram que a compreensão e previsão da taxa de *churn* (taxa de cancelamento) é essencial para modelos de negócios baseados em SaaS. A taxa de *churn* impacta diretamente na capacidade da empresa em projetar receitas futuras com precisão (Sukow; Grant, 2013). A abordagem preditiva do *churn* combina variáveis críticas que permitem às empresas SaaS projetar receitas futuras com base no comportamento histórico e esperado dos clientes.

Essa problemática também se aplica às empresas de software B2B (*Business to Business*: empresas que vendem produtos e serviços para outras empresas), que fornecem sistemas de gestão da qualidade para o setor alimentício por meio do modelo de negócios SaaS. Estas organizações oferecem soluções que ajudam outras empresas a manterem a conformidade com regulamentações de segurança alimentar, melhorar a eficiência operacional e garantir a qualidade do produto final (Pacana; Ulewicz, 2020).

Dessa forma, a questão de investigação proposta nesta pesquisa é: como identificar, por meio da aplicação de classificadores, quais preditores de cancelamentos de assinaturas influenciam a decisão dos clientes em cancelar seus

contratos em uma organização de SaaS B2B atuante em gestão de qualidade, no setor alimentício?

1.2 OBJETIVOS

Em resposta à questão de investigação, define-se o objetivo geral, que será alcançado a partir da execução dos objetivos específicos.

1.2.1 Objetivo geral

O objetivo geral da investigação é identificar os principais preditores de cancelamentos de assinaturas, utilizando algoritmos de classificação de aprendizagem de máquina ou classificadores.

1.2.1.1 Objetivos específicos

O objetivo geral desta pesquisa é composto por cinco objetivos específicos, a saber:

a) Conhecer as práticas atuais sobre *churn* e a utilização de classificadores para a sua prevenção;
b) Comparar as abordagens existentes sobre *churn* e a utilização de classificadores;
c) Analisar dados históricos de clientes atuais e que cancelaram os contratos na organização;
d) Desenvolver um modelo quantitativo baseado em dados históricos e em padrões identificados;
e) Avaliar os classificadores e preditores que influenciam no cancelamento de serviços.

1.3 JUSTIFICATIVAS

Esta pesquisa é fundamentada em quatro justificativas, detalhadas a seguir.

1.3.1 JUSTIFICATIVA ORGANIZACIONAL

A aplicação desta investigação em uma organização de software focada na gestão da qualidade para o setor alimentício é particularmente relevante, não somente

à organização analisada, como também a outras organizações de SaaS B2B. Este setor é crucial para a saúde pública e a segurança alimentar, e softwares de gestão eficientes podem garantir a manutenção dos padrões de qualidade ao reduzirem riscos e, por meio da rastreabilidade dos ingredientes, garantirem a saúde do consumidor.

A previsibilidade nos cancelamentos de assinaturas permitirá à organização manter sua base de clientes, e garantir adequação às necessidades das empresas contratantes.

A aplicação de sistemas de informação de qualidade significa contribuir para a aplicação consistente de práticas seguras e de alta qualidade na produção de alimentos. Isto, por sua vez, pode levar à redução da incidência de problemas de saúde decorrentes de alimentos contaminados ou mal processados, além de contribuir para uma maior conformidade às regulamentações governamentais e expectativas da sociedade (Pacana; Ulewicz, 2020).

1.3.2 JUSTIFICATIVA PESSOAL

Em nível pessoal, a motivação para esta investigação reside no desejo pessoal e necessidade profissional de desenvolvimento de competências e experiências avançadas em análise preditiva e modelos estatísticos.

Ademais, a integração de métodos qualitativos e quantitativos representa um desafio acadêmico e profissional significativo, oferecendo a oportunidade de aprofundar a compreensão das técnicas de análise de dados e de sua aplicação prática (Hair *et al.*, 2019). Esta pesquisa contribuirá para o desenvolvimento de competências valiosas no mercado de trabalho, proporcionando uma base sólida para futuras iniciativas profissionais e acadêmicas.

1.3.3 JUSTIFICATIVA CIENTÍFICA

No âmbito científico, esta pesquisa justifica-se pela contribuição ao conhecimento existente sobre retenção de clientes em organizações de SaaS, pois ao aplicar técnicas de análise preditiva e modelagem estatística, o estudo poderá oferecer novas perspectivas sobre como diferentes fatores influenciam a decisão dos clientes de cancelar suas assinaturas. Além disto, ao explorar a integração de

modelos qualitativos e quantitativos, a pesquisa pode contribuir com metodologias híbridas que são de crescente interesse em atividades relacionadas à análise de dados (Sukow; Grant, 2013).

A aplicação desta pesquisa é extremamente relevante para a Ciência da Informação, pois aborda a utilização de técnicas avançadas de análise de dados e modelagem preditiva em um contexto empresarial específico. A Ciência da Informação tem como um de seus principais objetivos a otimização da gestão dos fluxos informacionais e a melhoria dos processos de tomada de decisão através do uso de tecnologias da informação. Neste sentido, a presente investigação contribui significativamente para este objetivo ao aplicar técnicas de mineração de dados e de algoritmos de aprendizado de máquina para prever o cancelamento de assinaturas de serviços de *software*.

1.3.4 JUSTIFICATIVA ACADÊMICA

Na conjuntura do curso de bacharelado em Gestão da Informação, ofertado na Universidade Federal do Paraná, onde a pressente investigação está inserida, a pesquisa mostra-se alinhada aos objetivos educacionais do curso: formar profissionais capazes de aplicar técnicas avançadas de análise de dados em contextos empresariais.

Dentre as disciplinas do curso que contribuem diretamente à investigação, destacam-se nove disciplinas:

a) Fundamentos da Ciência da Informação, com os princípios básicos da gestão e análise de dados;
b) Fundamentos da Gestão Organizacional, com a apresentação de estratégias de retenção de clientes;
c) Metodologia da Pesquisa, e Introdução à Estatística, com técnicas e ferramentas condução de pesquisas, e para coleta e análise de dados;
d) Tecnologias da Informação e da Comunicação, e Sistemas de Informação, com tecnologias e sistemas para coleta e análise de dados de clientes;
e) Mineração de Dados, e Análise de Dados, com técnicas para extrair informações e tomar decisões baseadas em evidências;

f) Inteligência Artificial Aplicada à Gestão, com algoritmos avançados para prever *churn*.

A justificativa desta pesquisa está na combinação de aspectos organizacionais, pessoais, científicos e acadêmicos, pois a aplicação de um modelo preditivo híbrido pode não apenas melhorar a retenção de clientes e a sustentabilidade de organizações de SaaS B2B, mas também promover a segurança alimentar e proporcionar um crescimento significativo nas habilidades analíticas e preditivas do autor da pesquisa.

1.4 DELIMITAÇÃO DA PESQUISA

A pesquisa concentra-se na identificação dos principais preditores de cancelamento de assinaturas (*churn*) em uma empresa de software de SaaS B2B, atuante no setor alimentício, utilizando algoritmos de classificação de aprendizado de máquina. São desconsiderados outros aspectos relacionados à gestão de clientes ou às métricas de desempenho indiretamente ligados ao *churn*. Além disto, a pesquisa exclui a análise de dados qualitativos, como a satisfação dos clientes com o serviço ou a qualidade percebida dos produtos oferecidos pela empresa.

Em função da delimitação, a pesquisa utiliza exclusivamente dados históricos de interação dos clientes com a plataforma da organização analisada, excluindo quaisquer dados qualitativos ou subjetivos. Também são desconsiderados dados de outras fontes ou sistemas indiretamente relacionados ao uso da plataforma fornecida pela organização.

A pesquisa é de natureza quantitativa e, portanto, não abordará aspectos qualitativos relacionados ao comportamento dos clientes ou às razões subjetivas para o cancelamento de assinaturas. Questões como a influência de fatores socioeconômicos, culturais ou históricos sobre o comportamento dos clientes não são discutidas neste estudo.

1.5 ESTRUTURA DO DOCUMENTO

Este documento está dividido em cinco seções principais.

A primeira seção corresponde à Introdução, e apresenta o problema de pesquisa, os objetivos, a delimitação e as justificativas desta pesquisa.

A segunda seção é a Revisão da Literatura, com a reunião e análise das pesquisas pertinente relacionadas ao tema, onde há os conceitos de *churn* e de *Customer Relationship Management* (CRM); bem como as técnicas de aprendizado de máquina, os algoritmos classificadores e a mineração de dados; além dos tipos de aplicações e de classificadores utilizados na previsão de *churn*.

A terceira seção tem o rótulo Material e Métodos, e consiste nos encaminhamentos metodológicos. Nesta seção, apresenta-se a caracterização da pesquisa e os materiais e métodos utilizados, incluindo a coleta e o pré-processamento de dados, a descrição dos algoritmos de classificação aplicados e as técnicas de avaliação dos modelos.

A quarta seção (Apresentação de Resultados) abarca os resultados obtidos após a aplicação dos algoritmos de aprendizado de máquina nos dados coletados. Há a descrição dos dados coletados, os resultados da análise preditiva, a comparação entre os algoritmos utilizados e a discussão dos principais preditores de *churn* identificados.

Por fim, a quinta seção, chamada de Considerações finais, destaca reflexões sobre a investigação, e possibilidades de continuidade da pesquisa.

2 REVISÃO DE LITERATURA

Para a revisão de literatura opta-se pela revisão sistemática, que é uma metodologia de pesquisa rigorosa e estruturada, utilizada para sintetizar evidências científicas sobre uma determinada questão de pesquisa. Este processo envolve identificação, seleção, avaliação crítica e análise dos estudos relevantes disponíveis na literatura científica. A revisão sistemática segue um protocolo pré-definido que busca minimizar vieses e aumentar a reprodutibilidade dos resultados. Este método é amplamente nas Ciências Sociais Aplicadas para fornecer uma visão abrangente e confiável sobre o estado atual do conhecimento em um domínio, pois a revisão sistemática permite a identificação de lacunas no conhecimento, oferecendo uma base sólida para futuras pesquisas (Okoli *et al.*, 2019).

Realizaram-se pesquisas específicas por artigos científicos nas bases de dados acadêmicas *Web of Science* e Scielo, a respeito do tema, escolhendo-se a seguinte estratégia de busca: ("*churn prediction*" OR “*customer retention*” OR "*classification algorithms*") AND "machine learning".

A estratégia de busca resultou em 6.516 artigos científicos. Com aplicação de filtros de pesquisa, o resultado foi reduzido para 2.894 artigos (filtro de período de 5 anos: 2019 a 2023); depois para 1.746 artigos (artigos em acesso aberto); depois para 474 artigos (artigos relacionados à Gestão, Inteligência Artificial e Aprendizagem de Máquina); depois para 331 artigos (temas correlatos a classificadores e sua aplicação em organizações).

Houve leitura dos títulos, resumos e palavras-chave dos 331 artigos científicos resultantes, para seleção final dos 32 artigos utilizados nesta revisão sistemática de literatura, cujo objetivo é conhecer os principais conceitos, aplicações, tecnologias, extensão do tema e histórico das pesquisas existentes.

O referencial teórico está estruturado de modo a abordar os temas que se interconectam neste projeto: 1) *Customer Relationship Management* (CRM) e *Churn* de clientes; 2) Aprendizagem de máquina, algoritmos, classificadores e mineração de dados; 3) Tipos de classificadores para previsão de *churn*, áreas de aplicação e tipos de atributos; 4) Comparativo de estudos recentes; e 5) *Software* e linguagem de programação utilizados em pesquisas científicas.

Assim, apresenta-se a seguir os cinco temas integrantes desta investigação, conforme localizados nesta revisão sistemática.

2.1 CUSTOMER RELATIONSHIP MANAGEMENT (CRM) E CHURN DE CLIENTES

Nesta subseção abordam-se os conceitos de *churn*, *Customer Relationship Management* (CRM) e previsão de *churn*, que integram a gestão de negócios.

O gerenciamento do relacionamento com o cliente (equivalente em português para a sigla CRM) é uma estratégia de negócios que foca na interação entre uma organização e seus clientes (Payne; Frow, 2005). No contexto do CRM, o conceito de c*hurn* refere-se à taxa de perda de clientes (Zeithaml; Bitner; Gremler, 2018), sendo uma métrica importante para a sustentabilidade e o sucesso das organizações.

O *churn* é um termo usado no ambiente empresarial para descrever a métrica que registra os clientes que param de fazer negócios com uma organização. É uma métrica crítica, pois geralmente é mais caro adquirir novos clientes que manter os existentes (Gupta; Lehmann, 2005). O termo *churn* também é relacionado à quantidade de funcionários que saem de uma organização frente à quantidade contratada. É um indicador de satisfação do cliente ou funcionário; é a qualidade de uma organização; e, em última análise, é a longevidade de um negócio (Reichheld, 1996).

Payne e Frow (2005) argumentam que o CRM deve ser uma estratégia orientada ao cliente, que busca criar valor para ambos: o cliente e a organização. Isto é alcançado ao identificar, atrair, reter e desenvolver clientes para maximizar o benefício mútuo. Através da aplicação de uma estratégia de CRM eficaz, a organização pode entender melhor seus clientes, suas necessidades e comportamentos e, desta forma, melhorar a satisfação e a retenção dos clientes.

No contexto do CRM, a literatura sugere a existência de quatro níveis essenciais na interação entre a organização e o cliente: a identificação, a atração, a retenção e o desenvolvimento do cliente (Sivasankar; Vijaya, 2018).

O primeiro nível, a identificação, refere-se ao processo de reconhecer e entender quem são os clientes. Em seguida, há a atração, que se detém em captar o interesse dos clientes identificados e incentivá-los a interagir com a organização. O terceiro nível, a retenção, foca na manutenção do relacionamento existente com os clientes. Por fim, o desenvolvimento de clientes envolve aumentar o valor do cliente para a organização, por meio da venda cruzada (*cross-selling*), do incentivo a comprar de uma versão mais completa ou de maior valor de um produto ou serviço (*up-selling*) ou melhoria da lealdade do cliente. Estes quatro níveis, em conjunto, são projetados

para maximizar o entendimento do cliente, a fim de alcançar a retenção de longo prazo e maximizar os benefícios mútuos para o cliente e a organização (Sivasankar; Vijaya, 2018).

A previsão de *churn* é um aspecto crítico do CRM (Naidu; Zuva; Sibanda, 2022), pois ao prever quais clientes estão em risco de cancelar seus serviços, as organizações podem tomar medidas para reter os clientes, melhorar sua satisfação e, em última instância, reduzir a taxa de *churn*. Por isto, previsão de *churn* é uma parte fundamental da estratégia, porque ela permite que a organização antecipe e mitigue o risco de perda de clientes. Isto, por sua vez, impulsiona os lucros e a estabilidade da base de clientes.

Para fazer essas previsões de *churn*, a organização pode usar uma variedade de técnicas de mineração de dados. São métodos aplicáveis em diversas fases dos procedimentos de mineração de dados, como eliminar ruído e *outliers*, reduzir o espaço de características por meio da seleção de atributos mais relevantes e a escolha de amostras (Sivasankar; Vijaya, 2018).

2.2 APRENDIZAGEM DE MÁQUINA, ALGORITMOS, CLASSIFICADORES E MINERAÇÃO DE DADOS

Antes de discorrer sobre os tipos de classificadores aplicadores para previsão de *churn*, faz-se necessário compreender os conceitos de linguagem de máquina, algoritmo, classificadores, e mineração de dados, que se aplicam ao tema e ao problema em questão nesta pesquisa.

A linguagem de máquina é a forma mais fundamental de linguagem de programação composta por instruções binárias que podem ser executadas diretamente por um processador de computador. Cada instrução em linguagem de máquina corresponde a uma operação específica, como adição, subtração, multiplicação, divisão, carregamento de dados e armazenamento de dados (Tanenbaum; Woodhull, 2006).

Na previsão de *churn*, a linguagem de máquina possuí, em algoritmos, sequência finita e bem definida de instruções computacionais destinadas a executar uma tarefa ou resolver um problema específico. Os algoritmos são projetados para aceitar um conjunto de entradas, processá-las de acordo com um conjunto predefinido de regras e produzir um resultado ou saída. Algoritmos podem ser expressos em

muitas formas, incluindo linguagens de programação, pseudocódigo ou fluxogramas, e podem variar em complexidade (Cormen *et al.*, 2009).

Um classificador, no contexto de aprendizado de máquina, é um modelo ou algoritmo que é treinado para categorizar ou rotular uma entrada desconhecida em uma das classes predefinidas com base em suas características. Os classificadores são amplamente utilizados em uma variedade de aplicações, incluindo detecção de *spam*, diagnóstico médico e reconhecimento de imagem. Eles podem ser divididos em dois tipos principais: classificadores binários, que distinguem entre duas classes; e classificadores multiclasses, que distinguem entre mais de duas classes. Além disto, os classificadores podem ser baseados em diferentes tipos de algoritmos de aprendizado de máquina, incluindo regressão logística, árvores de decisão, máquinas de vetores de suporte e redes neurais, que serão abordados na próxima sessão (Hastie; Tibshirani; Friedman, 2009).

Nesta discussão, a mineração de dados (ou *data mining*, em inglês) insere-se por ser um processo que envolve a descoberta de padrões em grandes conjuntos de dados, a partir da utilização de métodos que abrangem aprendizado de máquina, estatísticas e sistemas de banco de dados. A mineração de dados é uma etapa importante na análise de dados e na extração de informações úteis, que podem ser convertidas em estratégias de negócios eficazes (Han; Kamber; Pei, 2011).

Na literatura científica específica, a previsão de *churn* refere-se ao uso de métodos analíticos para prever a probabilidade de um cliente abandonar um serviço ou produto. Os classificadores, como parte do aprendizado de máquina, são frequentemente usados para esta mesma finalidade. Estes modelos são treinados para identificar padrões nos dados que indicam um risco aumentado de *churn*. E uma vez identificados os padrões, as organizações podem tomar medidas para reter clientes, como oferecer descontos ou melhorar o atendimento ao cliente. A previsão de *churn* é especialmente importante em setores altamente competitivos, como telecomunicações, onde a aquisição de novos clientes pode ser significativamente mais cara que a retenção de clientes existentes (Umayaparvathi; Iyakutti, 2012; Zhu *et al.*, 2017; Verbraken; Verbeke; Baessens, 2014; Jamalian; Foukerdi, 2018).

No contexto da previsão de *churn*, a mineração de dados desempenha um papel fundamental, pois é com a análise de grandes conjuntos de dados de clientes, que os algoritmos de mineração de dados podem identificar padrões e tendências que indicam a probabilidade de um cliente abandonar um serviço ou produto. Estas

informações podem então ser usadas para criar estratégias de retenção de clientes com vistas a reduzir a taxa de *churn* (Naidu; Zuva; Sibanda, 2022).

A correlação entre a mineração de dados e a previsão de *churn* é evidente na aplicação de classificadores. Estes modelos de aprendizado de máquina, treinados para identificar padrões nos dados que indicam um risco aumentado de *churn*, são um exemplo de como a mineração de dados pode ser usada para informar e melhorar as estratégias de retenção de clientes. Através da identificação de tais padrões, as organizações podem tomar medidas proativas para reter esses clientes, como oferecer descontos ou melhorar o atendimento ao cliente. Isto demonstra a importância da mineração de dados na previsão de *churn* e, em última análise, na melhoria da retenção de clientes e na redução da taxa de *churn* (Umayaparvathi; Iyakutti, 2012; Zhu *et al.*, 2017; Verbraken; Verbeke; Baesens, 2014; Jamalian; Foukerdi, 2018).

2.3 CLASSIFICADORES PARA PREVISÃO DE CHURN, ÁREAS DE APLICAÇÃO E TIPOS DE ATRIBUTOS

Nesta subseção, discutem-se os diferentes tipos de classificadores usados na previsão de *churn*, os atributos que eles usam e como estes fatores variam em diferentes áreas de aplicação.

Como argumentado anteriormente, a rotatividade de clientes torna-se um componente crítico para a sustentabilidade e o crescimento de várias indústrias. A previsão de *churn* é particularmente relevante em setores como telecomunicações, negócios B2B, instituições de crédito e empresas de *software*. A aplicação de classificadores para prever o *churn* pode variar conforme o contexto e o setor, e diferentes atributos ou características podem ser aplicados a estes classificadores.

Em configurações de negócios B2B não contratuais, os dados de nível de fatura são frequentemente utilizados para a engenharia de recursos para prever o *churn* de clientes. Estes dados podem incluir informações sobre pagamentos, frequência de compras, valor médio de compra, entre outros (Mirković *et al.*, 2022). A previsão de *churn* neste contexto pode ajudar as organizações na identificação de clientes em risco e na implementação de estratégias de retenção eficazes.

Na indústria de telecomunicações, o histórico de utilização da rede de um cliente pode ser um recurso valioso para prever o *churn*. Isto pode incluir dados sobre

o uso de dados, chamadas, mensagens de texto e outros serviços relacionados (Sudharsa; Ganesh, 2022). A previsão de *churn* é uma preocupação significativa neste setor, onde a retenção de clientes é relevante para a sustentabilidade do negócio.

Os dados de CRM também podem ser usados para prever o *churn* de clientes na indústria de telecomunicações. Isto pode incluir informações sobre interações com clientes, reclamações, solicitações de serviço e outros dados relacionados ao relacionamento com clientes (Sana *et al.,* 2022).

Em organizações de crédito, os dados de transações de cartão de crédito podem ser usados como recursos para prever o *churn*. Isto pode incluir informações sobre o valor das transações, a frequência das transações, o tipo de transações e outros dados relacionados (Fatima *et al.*, 2021). A previsão de *churn* neste contexto também pode ajudar na identificação de comportamentos suspeitos que podem indicar atividades fraudulentas e evitar cobranças indevidas.

Nas organizações de *software*, a previsão de *churn* pode ser aplicada para identificar usuários ou clientes que estão em risco de cancelar suas assinaturas ou parar de usar um *software* ou serviço. Isto pode ser feito a partir da análise de dados de uso do *software, feedback* de clientes, interações de suporte aos clientes e outros dados relevantes.

Existem vários tipos de classificadores utilizados para a previsão de *churn* que são aplicados aos diversos setores empresariais.

No Quadro 1 há uma visão geral dos diferentes tipos de classificadores utilizados na previsão de *churn*. Cada classificador tem suas próprias características e é adequado para diferentes tipos de problemas e conjuntos de dados.

QUADRO 1 – CLASSIFICADORES DE PREDIÇÃO DE CHURN

(continua)

Classificador	**Descrição**	**Autor**
Regressão Logística	Método usado para prever a probabilidade de um evento específico, como o *churn*. A regressão logística usa uma função logística para modelar a probabilidade de *churn* com base em várias variáveis independentes.	Hosmer, Lemeshow, Sturdivant *(*2013)
Classification and Regression Trees (CART)	Algoritmo que cria modelos preditivos usando árvores de decisão binárias, particionando iterativamente os dados para maximizar a homogeneidade. Utiliza o índice Gini para classificação e o erro quadrático médio para regressão, com um processo de poda para evitar *overfitting.*	Breiman *et al.* (1986)

QUADRO 1 – CLASSIFICADORES DE PREDIÇÃO DE CHURN

Árvores de Decisão	Tipo de modelo de aprendizado de máquina que usa uma estrutura de árvore para representar uma série de decisões possíveis. Cada nó na árvore representa uma pergunta ou teste de uma variável específica, e cada ramo representa o resultado deste teste.	Quinlan (1986)
Random Forest	Método de aprendizado de máquina que combina várias árvores de decisão para fazer previsões. Cada árvore é construída a partir de uma amostra aleatória dos dados, e a previsão final é feita por votação majoritária das previsões de todas as árvores.	Breiman (2001)
Support Vector Machines (SVM)	Modelo de aprendizado de máquina que tenta encontrar o hiperplano que melhor separar as classes nos dados. Eles são particularmente úteis quando os dados não são linearmente separáveis.	Cortes, Vapnik (1995)
Redes Neurais Artificiais (ANN)	Modelos de aprendizado de máquina inspirados pela estrutura e função do cérebro humano. São compostos por uma série de nós, ou "neurônios", que são organizados em camadas e conectados por "sinapses". Cada nó recebe entrada de vários outros nós e produz uma saída com base em uma função de ativação.	Haykin (2009)
Gradient Boosting	Método de aprendizado de máquina que constrói um modelo preditivo na forma de um conjunto de modelos de previsão fracos, tipicamente árvores de decisão. Ele constrói o modelo em etapas, como outros métodos de *boosting*, e generaliza-os permitindo a otimização de uma função de perda arbitrária.	Friedman (2001)

Fonte: O autor (2024).

A partir do Quadro 1, observa-se que:

a) Cada um dos classificadores tem vantagens e desvantagens, e a escolha do classificador depende do problema específico e dos dados disponíveis;

b) A eficácia de cada classificador pode ser influenciada por vários fatores, incluindo a qualidade dos dados, a seleção de recursos e a sintonia de parâmetros;

c) Algoritmos baseados em árvores de decisão, por exemplo, podem ter folhas com a mesma probabilidade de classe e são sensíveis a ruídos;

d) Redes neurais buscam soluções subótimas e podem sofrer de *overfitting* quando o número de parâmetros aumenta;

e) Algoritmos genéticos, apesar de fornecerem modelos de previsão precisos, não esclarecem a probabilidade de ocorrência;

f) Métodos como as máquinas de vetores de suporte geralmente não produzem os melhores resultados.

No Quadro 2 há algumas das principais estratégias estudadas na literatura científica acerca dos classificadores usados para a previsão de *churn.*

QUADRO 2 – ESTRATÉGIAS DE APLICAÇÃO DE CLASSIFICADORES

Nº	Classificador	Descrição	Referência
1	Modelo híbrido de agrupamento e classificação	Utiliza um conjunto de algoritmos de agrupamento e classificação para melhorar o desempenho do modelo de previsão de *churn* de clientes. Algoritmos de agrupamento como *k-means*, *k-medoids* e *random* são empregados para testar conjuntos de dados de previsão de *churn*. Em seguida, uma técnica de hibridização é aplicada usando diferentes algoritmos de conjunto para avaliar o desempenho do sistema proposto. Os algoritmos de agrupamento mencionados, e integrados com diferentes classificadores, incluindo *Gradient Boosted Tree* (GBT)*, Decision Tree* (DT)*, Random Forest* (RF)*, Deep Learning* (DL) e Naive Bayes (NB), são avaliados em dois conjuntos de dados de telecomunicações padrão.	Liu *et al. (*2022)
2	Classificadores estabelecidos	Propõe uma abordagem para a previsão de *churn* de clientes em configurações de negócios B2B não contratuais que dependem exclusivamente de dados de nível de fatura para engenharia de recursos e usa *multi-slicing* para utilizar ao máximo os dados disponíveis. O *churn* é lançado como um problema de classificação binária e a capacidade de três classificadores estabelecidos para prever isso ao usar diferentes definições de *churn* é avaliada.	Mirković *et al.* (2022)
3	Técnicas de classificação como SVM, *Random Forest* e *Naives Bayes*	Discute uma maneira estruturada de prever a taxa de c*hurn* dos funcionários implementando várias técnicas de classificação como SVM, *Random Forest* e *Naives Bayes*. A performance dos classificadores foi comparada usando métricas como Matriz de Confusão, *Recall*, Taxa de Falso Positivo, e Acurácia para determinar o melhor modelo para a previsão de *churn*.	Bandyopadhyay, Jadhav (2021)

Fonte: O autor (2024).

Entende-se que a previsão de *churn* é uma área de pesquisa importante e com aplicações em variados setores. A escolha dos atributos ou características aplicados aos classificadores para prever o *churn* pode variar a depender do contexto e do setor; e a eficácia dos classificadores pode impactar significativamente na retenção de clientes e na sustentabilidade de negócios.

2.4 COMPARATIVO DE ESTUDOS

Artigos científicos recentes sobre previsão de *churn* envolvem o uso de diversas técnicas, incluindo métodos de conjunto, aprendizado semi-supervisionado, otimização dos grupos ou enxames, classificação baseada em regras e esquemas híbridos que integram técnicas de aprendizado de agrupamento e classificação (Assi *et al.*, 2023; Bogaert; Delaere, 2023; Banu *et al.* 2022; Liu *et al.*, 2022).

Essas técnicas são aplicadas em diversos contextos, incluindo sistemas de *software* de código aberto, e empresas de telecomunicações. Cada contexto apresenta seus próprios desafios e requer uma abordagem ligeiramente diferente para a previsão de *churn*.

As pesquisas atuais são focadas em melhorar a precisão da previsão de *churn*, identificando os recursos mais informativos, otimizando os parâmetros do modelo e explorando novas abordagens para a modelagem de *churn*. Além disto, a interoperabilidade dos modelos de previsão de *churn* é uma área de interesse crescente, à medida em que os pesquisadores tornam os modelos mais transparentes e compreensíveis para os usuários finais.

No Quadro 3, há um resumo dos estudos recentes sobre classificadores de previsão de *churn*, conforme a revisão sistemática realizada.

QUADRO 3 – COMPARATIVO DE ESTUDOS

(continua)

Tema	**Problema**	**Objetivos**	**Métodos**	**Resultados Obtidos**	**Autor**
Previsão de *Churn* em Sistemas de *Software* de Código Aberto.	Prever o impacto da mudança ao resolver defeitos.	Melhorar a previsão do impacto da mudança.	Modelo de Tópicos Incorporado (ETM) e XGBoost.	AUC de 0,84, 0,76 e 0,73 para o churn de código e 0,82, 0,71 e 0,73 para o número de arquivos alterados.	Assi *et al.* (2023)
Métodos de Conjunto na Previsão de *Churn.*	Comparar a eficácia de diferentes classificadores na previsão de *churn.*	Identificar o melhor classificador para previsão de *churn.*	33 classificadores, incluindo 6 classificadores únicos, 14 conjuntos homogêneos e 13 conjuntos heterogêneos.	Conjuntos heterogêneos com seleção de classificador de recozimento simulado apresentaram o melhor desempenho.	Bogaert, Delaere (2023)

QUADRO 3 – COMPARATIVO DE ESTUDOS

Previsão de *Churn* em Mercados de Negócios.	Prever o *churn* de clientes em mercados de negócios.	Controlar a existência de *churners* e não-*churners.*	Seleção de Características com Otimização de Enxame de Salp e Classificador Baseado em Regras Difusas.	Precisão de até 97,5%	Banu et al. (2022)
Esquema Híbrido Inteligente para Previsão de *Churn.*	Prever o *churn* de clientes em empresas de telecomuni-cações.	Melhorar o desempenho do modelo de previsão de *churn.*	Conjunto de algoritmos de agrupamento e classificação.	Precisões de 96% e 93,6% em dois conjuntos de dados de telecomunica-ções padrão.	Liu *et al.* (2022)

Fonte: O autor (2024).

Os resultados dos quatro estudos apresentados no Quadro 3 têm abordagens diferentes para a previsão de *churn* com taxas de acertos acima dos 70% verificadas.

No primeiro estudo, Assi *et al.* (2023) focam na previsão de *churn* em sistemas de *software* de código aberto. Eles usaram um Modelo de Tópicos Incorporado (ETM) e XGBoost para prever o impacto da mudança ao resolver defeitos. O estudo conseguiu uma AUC de 0,84, 0,76 e 0,73 para o *churn* de código; e 0,82, 0,71 e 0,73 para o número de arquivos alterados, demonstrando a eficácia do método.

No segundo estudo, Bogaert e Delaere (2023) mostram uma análise comparativa de 33 classificadores na previsão de *churn*. Os autores descobriram que os conjuntos heterogêneos com seleção de classificador de recozimento simulado apresentaram o melhor desempenho, destacando a importância dos métodos de conjunto na previsão de *churn*.

No terceiro estudo, Banu *et al.* (2022) enfatizam a previsão de *churn* em mercados de negócios. Eles usaram a Seleção de Características com Otimização de Enxame de Salp, e um classificador baseado em regras difusas, alcançando uma precisão de até 97,5%. Isto mostra a eficácia das técnicas de otimização e classificação baseada em regras na previsão de *churn*

Por fim, no estudo de Liu *et al.* (2022) propôs-se um esquema híbrido inteligente para a previsão de *churn*, que integra técnicas de aprendizado de agrupamento e classificação. Eles alcançaram precisões de 96% e 93,6% em dois conjuntos de dados de telecomunicações padrão, demonstrando a eficácia dos esquemas híbridos na previsão de *churn*.

Em resumo, observa-se que há uma variedade de abordagens sendo usadas para prever o *churn*, incluindo métodos de conjunto, técnicas de otimização, classificação baseada em regras e esquemas híbridos. Cada abordagem tem suas próprias vantagens e desvantagens, e a escolha da abordagem depende do contexto específico e dos dados disponíveis.

2.5 SOFTWARE EM ESTUDOS DE PREVISÃO DE CHURN COM CLASSIFICADORES

Em estudos de previsão de *churn*, que envolvem a identificação de clientes propensos a deixar um serviço, diversos *software* e linguagens de programação são amplamente utilizados devido à capacidade de processar grandes volumes de dados e aplicar algoritmos de classificação eficientes. Entre os mais populares, destacam-se R, Python (e suas bibliotecas, como scikit-learn, TensorFlow e Keras), Minitab, RapidMiner e SAS.

O R é um software livre amplamente utilizado para análise estatística e modelagem preditiva. Ele possui diversas bibliotecas como caret e randomForest, que são específicas para a construção de modelos de classificação, incluindo previsões de *churn* (Wickham, 2016).

A linguagem de programação Python é outra escolha popular devido à sua simplicidade e à vasta gama de bibliotecas disponíveis, como a Scikit-learn, usada para construção de modelos de aprendizado de máquina, oferecendo implementações de diversos classificadores como árvores de decisão, regressão logística e redes neurais (Pedregosa *et al.*, 2011).

Também existem ferramentas comerciais, como RapidMiner e SAS, amplamente utilizadas em contextos empresariais. O RapidMiner, por exemplo, oferece uma interface intuitiva para a construção de modelos de classificação sem a necessidade de programação extensiva, facilitando a análise preditiva para usuários não técnicos (Hofmann; Klinkenberg, 2013).

2.6 CONSIDERAÇÕES FINAIS DA REVISÃO DE LITERATURA

A revisão de literatura usa a metodologia de revisão sistemática, reconhecida por sua abordagem estruturada na síntese de evidências científicas sobre uma questão de pesquisa específica. Este método envolve a identificação, seleção, avaliação crítica e análise dos estudos pertinentes disponíveis na literatura científica, seguindo um protocolo pré-definido com o objetivo de minimizar vieses e aumentar a reprodutibilidade dos resultados (Okoli *et al.*, 2019). A pesquisa foi realizada nas bases de dados *Web of Science* e Scielo, utilizando palavras-chave específicas, operadores booleanos e filtros criteriosos para refinar os resultados, culminando em uma seleção final de artigos científicos relevantes para o estudo de previsão de *churn,* abrangendo os principais conceitos, aplicações e tecnologias associadas ao tema.

Os resultados da revisão são sistematicamente organizados em torno de três temas centrais: *Customer Relationship Management* (CRM) e *churn* de clientes; aprendizagem de máquina, algoritmos, classificadores e mineração de dados; e os tipos de classificadores utilizados para previsão de *churn*. A análise detalhada destes temas evidencia a relevância das técnicas de previsão de *churn* em diversos setores, tais como telecomunicações, negócios B2B e instituições de crédito. Dentre os diferentes classificadores e algoritmos, os que possuem as maiores taxas de acerto são regressão logística, árvore de decisão, random forest, Support Vector Machines (SVM), e Redes Neurais Artificiais (ANN).

Observa-se, ainda, uma série de *software* e linguagens de programação amplamente utilizados devido à sua capacidade de processar grandes volumes de dados e aplicar algoritmos de classificação eficientes. Entre os mais populares, destacam-se R, Python, RapidMiner e SAS.

3 MATERIAL E MÉTODOS

Esta seção apresenta a caracterização da pesquisa e os materiais e métodos necessários para a análise preditiva de *churn,* utilizando algoritmos de aprendizado de máquina. Inicialmente, detalha-se a natureza quantitativa da pesquisa e os critérios de seleção de dados. Em seguida, descrevem-se os procedimentos de coleta e pré-processamento de dados – essenciais para garantir a integridade e a qualidade dos dados utilizados. A seção também inclui a descrição dos algoritmos de classificação empregados, e as técnicas de avaliação de desempenho aplicadas para comparar a eficácia destes algoritmos. Por fim, apresenta-se a base de dados utilizada no estudo, especificando suas características e a origem dos dados coletados na organização analisada.

3.1 CARACTERIZAÇÃO DA PESQUISA

Esta pesquisa tem natureza quantitativa, na qual os dados de eventos de uma organização de SaaS são coletados e analisados, com vistas a atingir o objetivo geral desta investigação.

Segundo García e Hernández (2020), a abordagem quantitativa é essencial para estudos que envolvem a análise estatística de grandes volumes de dados, permitindo a identificação de padrões e correlações significativas.

Pelo seu caráter prático, trata-se de uma pesquisa aplicada, uma vez que tem como fim solucionar um problema existente por meio da prática e não apenas teórica (GIL, 2008). Neste sentido, García e Hernández (2020) afirmam que a pesquisa aplicada é relevante em contextos onde a solução de problemas específicos pode levar a inovações e melhorias práticas em determinado campo de estudo.

Esta investigação é caracterizada também como exploratória, uma vez que pretende discutir a respeito de um tema pouco estudado na Gestão da Informação. A pesquisa pode suscitar novas questões e contribuir ao aprofundamento do tema (GIL, 2008), pois a exploração inicial de novas áreas de conhecimento é fundamental para o desenvolvimento de hipóteses robustas que podem ser testadas em pesquisas futuras (García; Hernández, 2020).

Por fim, quanto aos procedimentos, este estudo é caracterizado como experimental, uma vez que ao longo da investigação debatem-se os estudos e os

métodos existentes (GIL, 2008); e as variáveis são elencadas para posterior análise acerca da existência de uma influência no objeto de investigação: a taxa de cancelamento.

Ademias, estudos experimentais são relevantes na identificação de relações causais e na validação de modelos teóricos através da manipulação controlada de variáveis (García; Hernández, 2020).

3.2 FASES DA METODOLOGIA

A metodologia adotada nesta pesquisa segue um fluxo estruturado e sistemático baseado na abordagem CRISP-DM (*Cross-Industry Standard Process for Data Mining*), que é utilizada na literatura cientifica para projetos de mineração de dados. O processo CRISP-DM, desenvolvido pelo CRISP-DM Consortium, é composto por seis fases principais: entendimento do negócio, entendimento dos dados, preparação dos dados, modelagem, avaliação, e implementação (Wirth; Hipp, 2000).

A implementação da metodologia CRISP-DM, nesta investigação, segue as seis fases principais.

Na primeira fase – entendimento do negócio – o objetivo é compreender os requisitos do projeto do ponto de vista do negócio (Chapman *et al.*, 2000). O objetivo principal é identificar os preditores de cancelamento de serviços (*churn*) em uma empresa de software B2B, que tem uma aplicação de gestão de documentos legais para empresas de alimentos. Para isto, é essencial definir claramente as metas de negócio, que incluíam melhorar a retenção de clientes e compreender seu comportamento para posteriormente criar meios para aumentar a satisfação dos clientes.

A segunda fase – entendimento dos dados – envolve a coleta inicial e a familiarização com os dados. Os dados são obtidos a partir de eventos na aplicação que são registros de clientes. Esta fase inclui a inspeção dos dados para entender sua estrutura e qualidade, identificando quaisquer problemas como dados ausentes ou inconsistentes (Chapman *et al.*, 2000).

Durante a fase de preparação dos dados, são realizadas várias etapas para preparar o conjunto de dados final. Isto incluiu a limpeza dos dados, remoção de registros com valores ausentes e a transformação de variáveis categóricas em

numéricas. Técnicas de balanceamento de classes, como up-sampling, foram aplicadas para lidar com o problema de desbalanceamento dos dados, garantindo que os modelos de *machine learning* pudessem ser treinados de maneira eficaz (Chapman *et al.*, 2000).

Na fase de modelagem, diversos algoritmos de classificação são aplicados para prever o *churn*. Os algoritmos utilizados incluíram *Random Forest*, Regressão Logística, SVM (Support Vector Machines) e Redes Neurais. Cada modelo é treinado usando validação cruzada, e sua performance é avaliada com base em métricas como acurácia, precisão, recall e F1-score. Este processo garante que os modelos desenvolvidos sejam robustos e eficazes (Wirth; Hipp, 2000).

A fase de avaliação envolve a comparação dos modelos para determinar qual apresentava o melhor desempenho. Adicionalmente, foram utilizadas técnicas de análise de características, como ANOVA e RFE (Recursive Feature Elimination), para identificar os preditores mais influentes no *churn* dos clientes (Shearer, 2000).

Finalmente, na fase da implementação prática do modelo tem por finalidade que as previsões pudessem ser utilizadas em um ambiente real, contribuindo diretamente para os objetivos de negócio da empresa (Azevedo; Santos, 2008). Porém, como o objetivo deste estudo é estabelecer o modelo e analisar os preditores a parte da implementação não será abordada. Mesmo assim, ideia é que o modelo final seja integrado à plataforma da organização SaaS analisada nesta investigação. Este modelo permitirá à empresa prever quais clientes estão em risco de cancelar seus serviços, possibilitando a adoção de medidas proativas para retenção desses clientes.

3.2.1 *Software* utilizados

Os seguintes *software* e pacotes foram utilizados no desenvolvimento da pesquisa:

a) Análise e modelagem: R Programming Language 4.4.3;
b) Ambiente de desenvolvimento: Python;
c) Feramentas de visualização: Jupyter Notebook, e PyCharm;

Destaca-se que devido a erros constantes causados pelas características dos dados, a linguagem R é mencionada na seção Material e Métodos, entretanto não é inclusa na seção Resultados.

3.2.2 Pacotes de R

O pacote caret (Kuhn, 2020) é essencial para pré-processamento de dados, seleção de características, ajuste e avaliação de modelos preditivos, oferecendo uma interface unificada para diversas tarefas de modelagem. O tidyverse (Wickham *et al.*, 2019) é uma coleção de pacotes integrados que facilitam a manipulação, transformação e visualização de dados, proporcionando uma abordagem coesa e intuitiva para a análise de dados.

Para tarefas específicas de aprendizado de máquina, o pacote nnet (Venables; Ripley, 2002) permite a criação e treinamento de redes neurais artificiais. O e1071 (Meyer *et al.*, 2019) inclui funções para máquinas de vetor de suporte (SVM), além de outros algoritmos de aprendizado de máquina, sendo útil para classificação e regressão. O glmnet (Friedman; Hastie; Tibshirani, 2010) implementa modelos de regressão linear e logística com regularização, utilizando métodos de Lasso e Ridge para evitar *overfitting*.

Para análise de performance de modelos de classificação, o pROC (Roberts; Walter, 2021) é utilizado para calcular e visualizar a curva ROC (Receiver Operating Characteristic) e a AUC (Área Sob a Curva). Em cenários de desbalanceamento de classes, o ROSE (Menard, 2020) oferece métodos para balancear conjuntos de dados, melhorando a performance de modelos de classificação.

Na construção de modelos interpretáveis, o rpart (Therneau & Atkinson, 2019) é utilizado para criar árvores de decisão e regressão, enquanto o rpart.plot (Milborrow, 2020) facilita a visualização dessas árvores, tornando os modelos mais compreensíveis. Para visualizações aprimoradas, o ggpubr (Kassambara, 2020) adiciona funcionalidades ao ggplot2, permitindo a criação de gráficos prontos para publicação.

Finalmente, o skimr (Wrang & Lee, 2021) é útil para a análise descritiva dos dados, fornecendo resumos estatísticos rápidos e completos que facilitam a exploração inicial e a preparação dos dados para análises mais detalhadas. A seguir, no Quadro 4, explica-se o uso de cada pacote em R, anteriormente mencionado.

QUADRO 4 – PACOTES EM R

Pacote	Explicação	Autoria
caret	Ferramentas para pré-processamento, seleção de características, ajuste e avaliação de modelos.	Kuhn (2020)
tidyverse	Coleção de pacotes para manipulação, transformação e visualização de dados.	Wickham *et al.* (2019)
nnet	Funções para criar e treinar redes neurais artificiais.	Venables, Ripley (2002)
e1071	Inclui funções para máquinas de vetor de suporte (SVM) e outros algoritmos de aprendizado de máquina.	Meyer *et al.* (2019)
glmnet	Implementa modelos de regressão linear e logística regularizados.	Friedman, Hastie, Tibshirani (2010)
pROC	Análise e visualização da curva ROC (Receiver Operating Characteristic) e cálculo da AUC.	Roberts, Walter (2021)
ROSE	Métodos para balanceamento de classes em conjuntos de dados desbalanceados.	Menard (2020)
rpart	Ferramentas para criar árvores de decisão e regressão.	Therneau, Atkinson (2019)
rpart.plot	Visualização de árvores de decisão criadas com rpart.	Milborrow (2020)
ggpubr	Funções adicionais para criar gráficos prontos para publicação com ggplot2.	Kassambara (2020)
skimr	Ferramentas para resumo estatístico e análise descritiva dos dados.	Wrang, Lee (2021)

Fonte: O autor (2024).

Observa-se que a ferramenta R possuí uma extensa lista de pacotes que são úteis na análise de dados e aprendizagem de máquina.

3.2.3 Bibliotecas de Python

O uso de diversas bibliotecas e funções é fundamental para o aprimoramento de modelos de machine learning. NumPy e Pandas são essenciais para a manipulação e análise de dados, oferecendo suporte para *arrays* e estruturas de dados eficientes (Van der Walt; Colbert; Varoquaux, 2011; McKinney, 2010). A função train_test_split do scikit-learn facilita a divisão dos dados em conjuntos de treino e teste, enquanto cross_val_score permite a avaliação do modelo através de validação cruzada (Pedregosa *et al.*, 2011).

Para a otimização de hiperparâmetros, utiliza-se GridSearchCV, pois realiza uma busca em grade para encontrar as melhores combinações de parâmetros (Pedregosa *et al.*, 2011). Modelos como RandomForestClassifier**,** LogisticRegression, SVC, e MLPClassifier são amplamente utilizados para diferentes tarefas de classificação, cada um com suas próprias vantagens (Breiman, 2001; Pedregosa *et al.*, 2011; Cortes, Vapnik, 1995).

A avaliação do desempenho dos modelos pode ser realizada utilizando várias métricas disponíveis no scikit-learn, como accuracy_score, roc_auc_score, roc_curve, precision_recall_fscore_support, confusion_matrix, e classification_report (Pedregosa *et al.*, 2011; Bradley, 1997; Fawcett, 2006; Pituch; Stevens, 2012). Para a seleção de características, técnicas como SelectKBest e RFE são úteis para identificar as características mais relevantes (Pedregosa *et al.*, 2011; Guyon *et al.*, 2002).

A visualização dos dados é facilitada por bibliotecas como matplotlib.pyplot e seaborn, que permitem a criação de gráficos informativos (Hunter, 2007; Waskom *et al.*, 2021). A LabelBinarizer é utilizada para a binarização de rótulos, e a função precision_recall_curve é útil para calcular curvas de precisão-recall, oferecendo uma visão detalhada da performance do modelo (Pedregosa *et al.*, 2011; Davis, Goadrich, 2006). A seguir, no Quadro 5 explica-se de forma detalhada cada biblioteca de Python anteriormente mencionada.

QUADRO 5 – BIBLIOTECAS DE PYTHON

(continua)

Biblioteca/Função	**Descrição**	**Referência**
NumPy	Biblioteca fundamental para computação numérica em Python, oferecendo suporte para arrays e matrizes de grande dimensão, além de uma coleção de funções matemáticas.	Van der Walt, Colbert, Varoquaux (2011)
Pandas	Biblioteca de software para manipulação e análise de dados, que oferece estruturas de dados e operações para manipular tabelas numéricas e séries temporais.	McKinney (2010)
train_test_split	Função do scikit-learn que divide matrizes ou matrizes esparsas em subconjuntos aleatórios de treino e teste.	Pedregosa *et al.* (2011)
cross_val_score	Função do scikit-learn que avalia um score através da validação cruzada, dividindo o conjunto de dados em múltiplas partes.	Pedregosa *et al.* (2011)
GridSearchCV	Função do scikit-learn que realiza uma busca em grade para otimização de hiperparâmetros de um modelo.	Pedregosa *et al.* (2011)
RandomForestClassifier	Classe do scikit-learn para classificação utilizando o algoritmo de Random Forest, que combina múltiplas árvores de decisão.	Breiman (2001)
LogisticRegression	Classe do scikit-learn para realizar regressão logística, um modelo estatístico para prever a probabilidade de um evento.	Pedregosa *et al.*, (2011)
SVC	Classe do scikit-learn para Support Vector Classification, uma técnica de machine learning para classificação.	Cortes, Vapnik (1995)
MLPClassifier	Classe do scikit-learn para redes neurais perceptron multicamadas.	Pedregosa *et al.* (2011)

QUADRO 5 – BIBLIOTECAS DE PYTHON

accuracy_score	Função do scikit-learn que calcula a acurácia da classificação.	Pedregosa *et al.* (2011)
roc_auc_score	Função do scikit-learn que calcula a área sob a curva ROC (Receiver Operating Characteristic).	Bradley (1997)
roc_curve	Função do scikit-learn que calcula a curva ROC para a medição da performance de um modelo de classificação.	Fawcett (2006)
precision_recall_fscore_support	Função do scikit-learn que calcula precisão, recall, F1-score e suporte.	Pedregosa *et al.* (2011)
confusion_matrix	Função do scikit-learn que calcula a matriz de confusão para avaliar a performance de um algoritmo de classificação.	Pituch, Stevens (2012)
classification_report	Função do scikit-learn que constrói um relatório de classificação.	Pedregosa *et al.* (2011)
SelectKBest	Função do scikit-learn para selecionar as melhores características baseadas em testes estatísticos.	Pedregosa *et al.* (2011)
f_classif	Função do scikit-learn para teste ANOVA F-valor.	Miller Jr (1997)
RFE	Técnica de Eliminação Recursiva de Atributos do scikit-learn para selecionar características por importância.	Guyon *et al.* (2002)
matplotlib.pyplot	Biblioteca para criação de gráficos em Python.	Hunter (2007)
seaborn	Biblioteca para visualização de dados baseada em matplotlib.	Waskom *et al.* (2021)
LabelBinarizer	Classe do scikit-learn para binarização de rótulos em arrays de classe binária.	Pedregosa *et al.* (2011)
precision_recall_curve	Função do scikit-learn que calcula a curva de precisão-recall para um conjunto de previsões.	Davis, Goadrich (2006)

Fonte: O autor (2024).

Nota-se que existe uma coleção abrangente de bibliotecas e funções amplamente utilizadas em Python para computação numérica, manipulação e análise de dados, aprendizado de máquina e visualização de dados. Cada entrada inclui uma descrição e a referência relevante, fornecendo uma visão clara sobre o uso e a importância de cada ferramenta.

3.3 MODELO DE NEGÓCIO DA EMPRESA E COLETA DE DADOS

Esta seção aborda as características do modelo de negócio, os passos para a coleta e as características da amostra que foram utilizados nesta pesquisa.

3.3.1 MODELO DE NEGÓCIO

Os dados foram coletados a partir de registros de uma empresa de *software* estadunidense, fundada em 2018, especializada em fornecer soluções para a gestão de documentos legais de empresas de alimentos em sua cadeia de suprimentos. A organização analisada foca em indústrias que exigem rigorosas normas de conformidade e rastreabilidade, como a alimentícia e a farmacêutica. Através de sua plataforma baseada em nuvem, a organização analisada oferece ferramentas para a digitalização e automação de processos de gerenciamento de qualidade, *compliance* e documentação entre fornecedores e fabricantes.

3.3.2 COLETA DE DADOS

Os registros de eventos foram coletados utilizando-se o serviço da Mixpanel – uma plataforma avançada de análise de dados que permite às empresas monitorarem e entenderem o comportamento dos usuários em tempo real, através de diversas aplicações digitais. A Mixpanel permite o monitoramento detalhado de eventos definidos pelo usuário, como cliques, visitas a páginas, transações, e outras interações relevantes. Isto possibilita uma compreensão granular de como os usuários interagem com a aplicação ou site.

Nesta investigação, são necessárias as sete etapas a seguir, para extração de dados por meio do serviço Mixpanel, a saber:

a) Na Dashboard da empresa criam-se cinco *Insights Reports* com o objetivo de selecionar os registros de monitoramento produto;
b) Em cada *Insight Report* foram selecionados registros de eventos das empresas. Foram aplicados filtros para retirar usuários de teste ou de administradores e para segmentar os dados por empresa;
c) Foi selecionado um intervalo de tempo de 365 dias, de 5 de junho de 2023 até 6 de junho de 2024;
d) Devido ao fato de haver 9 clientes foram selecionados dados de clientes gratuitos que utilizam os mesmos serviços dentro da empresa. Totalizando 28 empresas;
e) Após este processo foram extraídos os dados de cada Insight Report da plataforma no formato .csv;

f) Os dados coletados foram agrupados em uma planilha de Excel para cada empresa onde cada atributo era uma coluna e cada empresa uma entidade;
g) A planilha foi adicionada duas colunas contendo preditores categóricos para diferenciar os clientes que cancelaram os serviços e tipo de conta da empresa (paga ou grátis).

3.3.3 CARACTERÍSTICAS DOS DADOS

A população (Apêndice 1) com amostra não probabilística e tipo de amostra intencional é composta por 28 organizações do setor alimentício, que interagiram com a aplicação e contém 336 registros. Das 28 organizações, 11 são assinantes que pagam para utilizar a plataforma. E dentre as 11 assinantes, duas cancelarão a assinatura no segundo semestre de 2024. Existem, ainda, outras 17 organizações que abriram uma conta grátis, por meio de uma parceria com organizações que auditam certificados religiosos.

A distribuição da população contém um conjunto diversificado de 28 organizações atuantes em diferentes segmentos da indústria alimentícia. O grupo inclui grandes produtores e processadores de alimentos (30% das empresas); empresas de médio porte (30%) e pequenas empresas especializadas (40%). As empresas dividem-se em categorias principais, sendo 20% produtores de ingredientes básicos (grãos e laticínios); 25% fabricantes de alimentos prontos para consumo; 15% empresas de bebidas; 10% distribuidores internacionais de produtos alimentícios; 20% fornecedores de ingredientes funcionais e naturais; e 10% empresas inovadoras no campo da nutrição. Além disso, cerca de 15% das empresas focam em nichos específicos, como alimentos para dietas especiais, enquanto os outros 85% abrangem variados produtos. A lista inclui empresas tradicionais, com longa trajetória no mercado (aproximadamente 35%), startups (cerca de 20%), e empresas desconhecidas pelo consumidor final de alimentos (45%).

A diversidade dessas empresas reflete a complexidade e amplitude da indústria alimentícia, que atende variadas necessidades e preferências dos consumidores.

O Quadro 6 descreve quais eventos são coletados na amostra.

QUADRO 6 – DESCRIÇÃO DOS PREDITORES

Atributo	Descrição	Tipo de Preditor
NumeroDeLogins	Número total de logins realizados pelo usuário durante o período de análise.	Contínuo
SessaoMediaEmMinutos	Duração média das sessões do usuário, medida em minutos.	Contínuo
CriarSolicitacao	Número de solicitações criadas pelo usuário.	Contínuo
UsuariosUnicos	Número de usuários únicos que interagiram com o mesmo serviço durante o período de análise.	Contínuo
AdicionarProduto	Número de vezes que o usuário adicionou produtos ao serviço.	Contínuo
AdicionarFabrica	Número de vezes que o usuário adicionou fábricas ao serviço.	Contínuo
EditarInformacoes	Número de vezes que o usuário editou informações no serviço.	Contínuo
ArquivarSolicitacao	Número de solicitações arquivadas pelo usuário.	Contínuo
AdicionarFornecedor	Número de vezes que o usuário adicionou fornecedores ao serviço.	Contínuo
Busca	Número de buscas realizadas pelo usuário dentro do serviço.	Contínuo
CancelouServico	Indicador com três classes 0 (Não cancelou), 1 (Cancelou), 2 (Não assinante).	Categórico
TipodeConta	Indicador com duas classes 0 (Gratuita) e 1 (Paga).	Categórico

Fonte: O autor (2024).

Conforme citado no processo de extração dados, os registros foram coletados ao longo de um período específico de 365 dias: de 10 de junho de 2023 a 9 de junho de 2024. Este período foi escolhido devido à implementação do Mixpanel, que ocorreu em abril de 2023, e os dados só começaram a ser coletados em sua totalidade a partir de maio. Na seleção dos eventos constantes no Quadro 6, considerou-se os eventos com mais interações dentro da aplicação da organização analisada.

3.3.4 ANÁLISE DESCRITIVA DOS DADOS

Nesta seção, apresenta-se a análise descritiva dos principais atributos citados acima. A população contém eventos ou dados das interações dos clientes com os serviços, incluindo atividades como buscas, adições de fábricas, produtos e fornecedores, criação de solicitações, edições de informações, arquivamento de solicitações, número de usuários únicos, número de logins, tempo médio de sessão, e tipo de conta. Além disto, o atributo meta "Cancelou Serviço" foi incluído para identificar se um cliente cancelou ou não o serviço conforme observado na Tabela 1, a seguir.

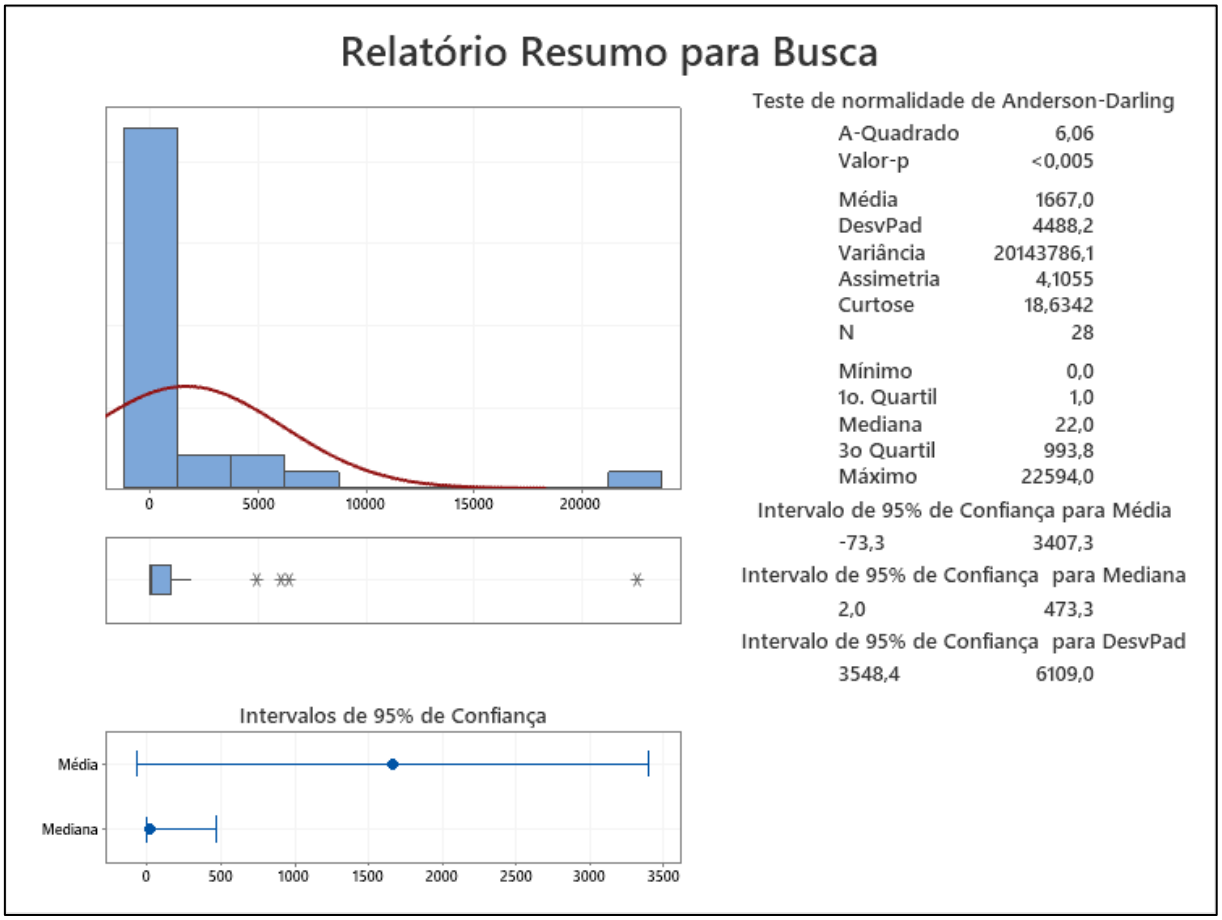
TABELA 1 – MÉDIA, MEDIANA, VALOR MÍNIMO E MÁXIMO DA AMOSTRA

Atributo	Média	Mediana	Mínimo	Máximo	Valor-P	Observações
Busca	1320,89	112	0	22594	<0,005	Alta variação
Adicionar Fábrica	3,64	0	0	38	<0,005	Muitos usuários não utilizam
Adicionar Produto	2,39	0	0	37	<0,005	Uso esporádico
Adicionar Fornecedor	3,68	1	0	25	<0,005	Variação considerável
Criar Solicitação	77,46	6	0	853	<0,005	Uso intensivo por alguns usuários
Editar Informações	161,96	2	0	3566	<0,005	Alta variabilidade
Arquivar Solicitação	14,32	0	0	180	<0,005	Uso esporádico
Usuários Únicos	7,25	6	2	50	<0,005	Distribuição mais concentrada
Número de Logins	58,25	7	1	456	<0,005	Grande variação no número de logins
Sessão Média em Minutos	394,75	91	11	2547	<0,005	Alta variabilidade
Cancelou Serviço (0)	9	-	-	-	<0,005	0 representa "Não Cancelou"
Cancelou Serviço (1)	2	-	-	-	<0,005	1 representa "Cancelou"
Cancelou Serviço (Não Esp)	17	-	-	-	<0,005	2 representa "Não Especificado"
Tipo de Conta (Pago)	11	-	-	-	<0,005	1 representa "Enterprise"
Tipo de Conta (Grátis)	17	-	-	-	<0,005	0 representa "Freemium"

Fonte: O autor (2024).

A análise descritiva inclui medidas de tendência central, dispersão e a distribuição dos dados, que são essenciais para entender as características fundamentais do conjunto de dados utilizado. Para permitir uma análise visual, alguns atributos, como na Figura 1, com o resumo da variável “Busca” serão apresentados.

FIGURA 1 – RESUMO BUSCA

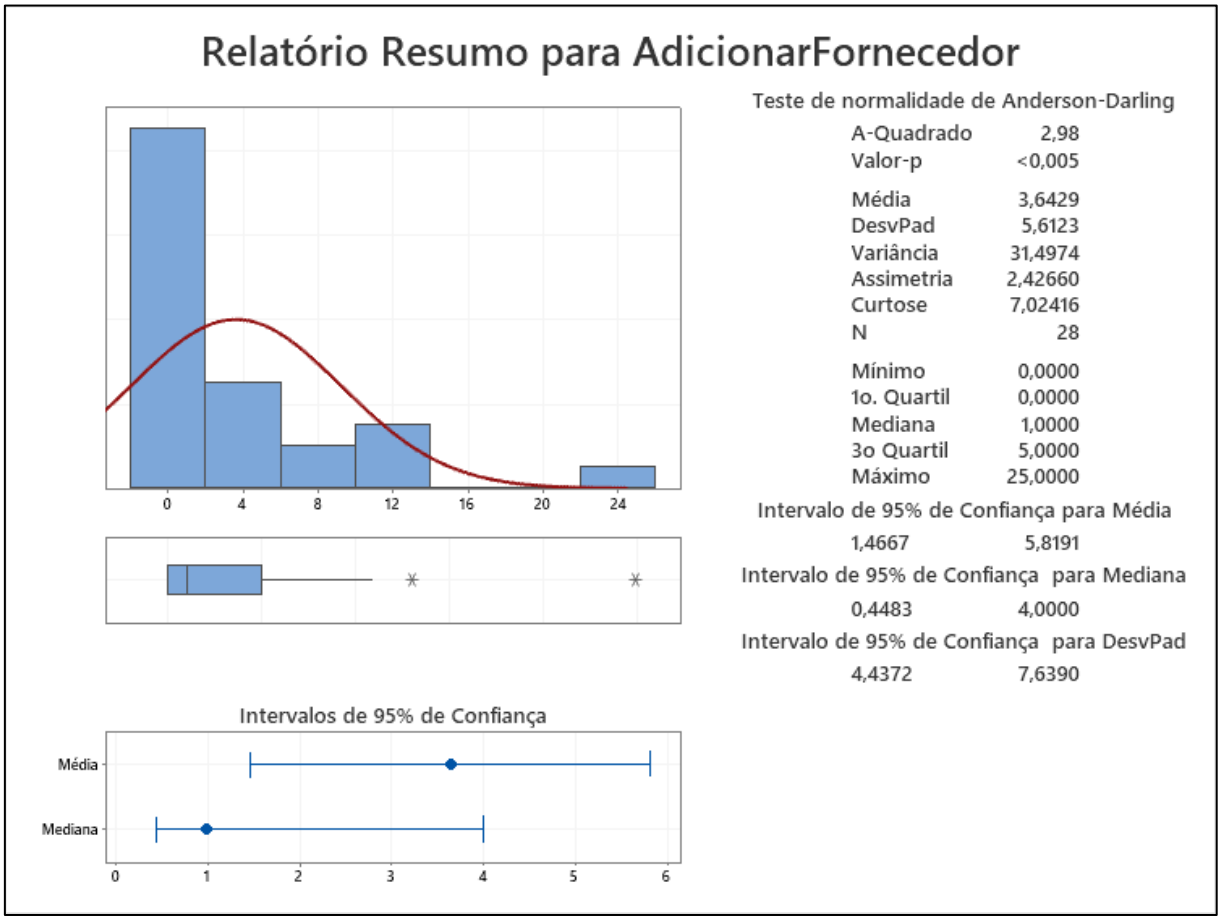


Fonte: O autor (2024).

Ao se tratar de uma análise descritiva da população, os valores para o atributo "Busca" variam de 0 a 22.594, com uma média de 1.320,89 e uma mediana de 112, indicando uma alta variação. Já "Adicionar Fábrica" apresenta valores de 0 a 38, com média de 3,64 e mediana de 0, mostrando que muitos usuários não utilizam essa funcionalidade. O atributo "Adicionar Produto" varia de 0 a 37, com média de 2,39 e mediana de 0, refletindo um uso esporádico, uma vez que os valores mais elevados estão concentrados em empresas que pagam pelos serviços.

No caso do “AdidionarFornecedor” na Figura 2, a distribuição não se concentra tanto no valor de 0 a 1.

FIGURA 2 – RESUMO ADICIONAR FORNECEDOR



Fonte: O autor (2024).

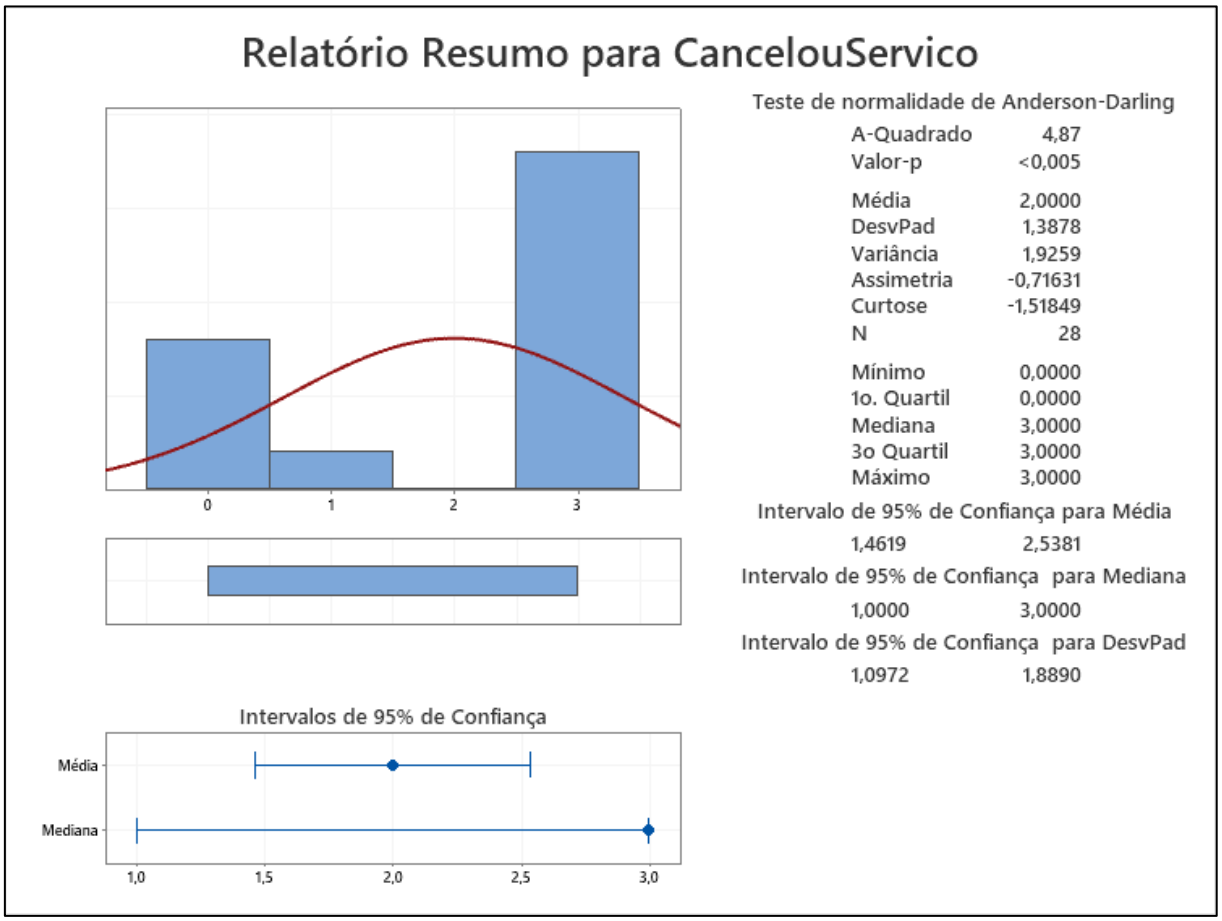
Pode se observar que para "Adicionar Fornecedor", os valores vão de 0 a 25, com média de 3,68 e mediana de 1, indicando uma variação considerável. "Criar Solicitação" apresenta valores de 0 a 853, com média de 77,46 e mediana de 6, mostrando um uso intensivo por alguns usuários. Para "Editar Informações", os valores variam de 0 a 3.566, com média de 161,96 e mediana de 2, indicando alta variabilidade.

Já o atributo "Arquivar Solicitação" varia de 0 a 180, com média de 14,32 e mediana de 0, mostrando uso esporádico. "Usuários Únicos" tem valores de 2 a 50, com média de 7,25 e mediana de 6, mostrando uma distribuição mais concentrada. "Número de Logins" varia de 1 a 456, com média de 58,25 e mediana de 7, indicando grande variação no número de logins.

No quesito tempo na plataforma, "Sessão Média em Minutos" apresenta valores de 11 a 2.547, com média de 394,75 e mediana de 91, mostrando alta variabilidade no tempo médio de sessão. Abaixo a Figura 3 é uma representação do atributo meta

e o que se observa é que a maioria dos dados está concentrada no número 3 que indica empresas que este atributo não se aplica, seguido ao número 0 que indica que este a dos que se aplica a maioria é composta por assinantes.

FIGURA 3 – RESUMO CANCELOU SERVIÇO



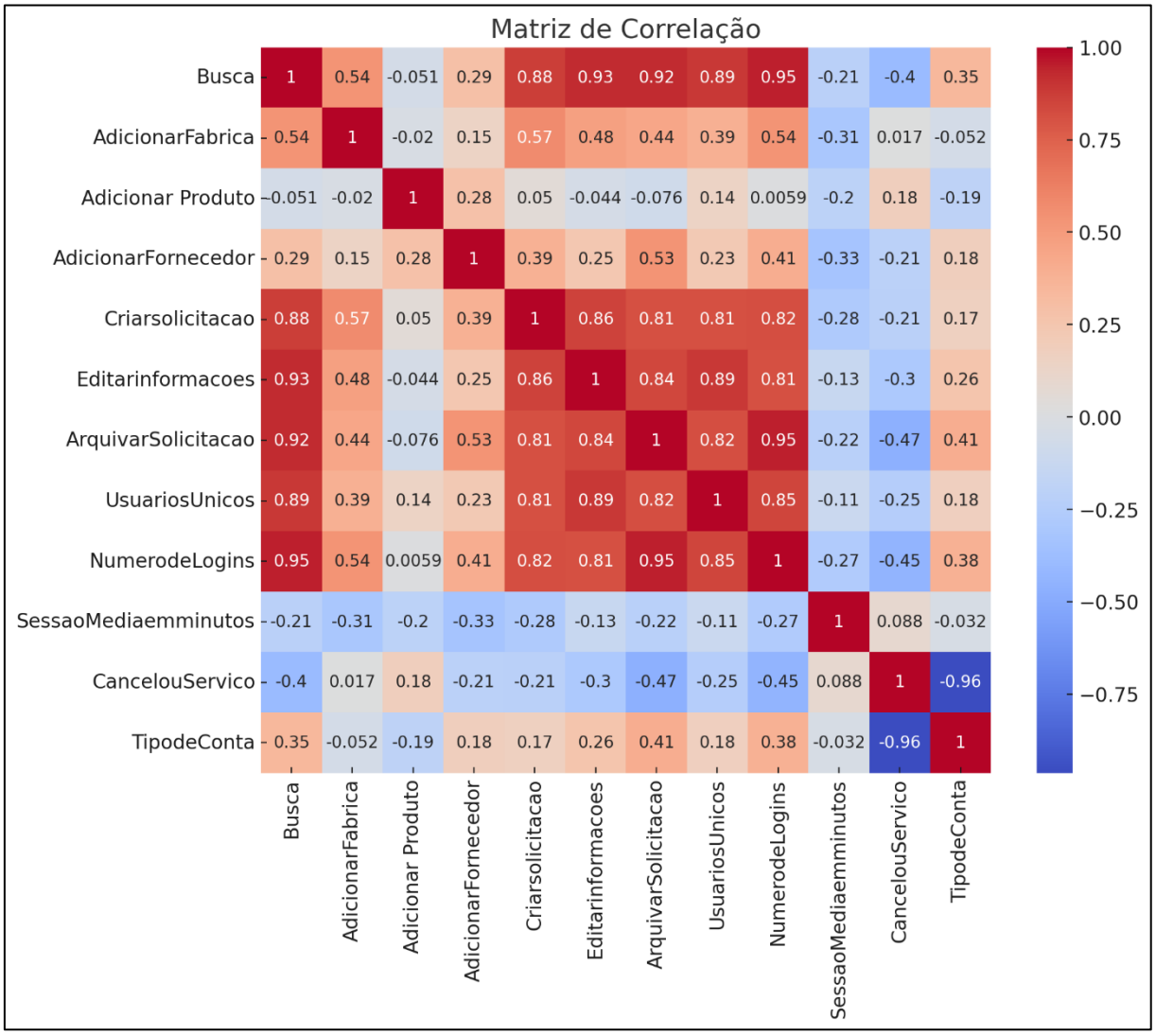
Fonte: O autor (2024).

Para o atributo meta "Cancelou Serviço", temos 9 usuários que não cancelaram, 17 sem especificação uma vez que não se aplica e 2 que cancelaram. No "Tipo de Conta", 11 usuários têm conta "Enterprise", 17 "Freemium". Outro ponto importante é que nenhum atributo passou no teste de normalidade o que quer diz que não é uma distribuição normal.

3.3.4.1 Matriz de Correlação

A matriz de correlação mostra a relação entre os atributos. A Figura 4 apresenta a matriz de correlação visualizada como um mapa de calor.

FIGURA 4 – MATRÍZ DE CORRELAÇÃO



Fonte: O autor (2024).

A matriz de correlação (Figura 4) mostra a relação entre os diferentes atributos do conjunto de dados. Os coeficientes de correlação variam de -1 a 1, onde valores próximos de 1 ou -1 indicam uma forte correlação positiva ou negativa, respectivamente, enquanto valores próximos de 0 indicam pouca ou nenhuma correlação.

A análise da matriz de correlação revela que há correlações significativas entre alguns atributos. Por exemplo, "UsuariosUnicos" e "NumerodeLogins" apresentam uma correlação positiva. Isto significa que existe uma influência entre o número de *logins* e o número de usuários únicos. Por outro lado, "SessaoMediaemminutos" e "Criarsolicitacao" mostram uma correlação negativa, o que indica que não existe uma influência entre os dois fatores. Já sobre o atributo meta “CancelouServico” inexiste

correlação significativa com as outras variáveis. Por outro lado, o atributo “Busca” tem uma correlação forte com “CriarSolicitacao”, “EditarInformacao”, “Arquivarsolicitação”, “Usuariosunicos” e “NumerodeLogins”. Isto não quer dizer que inexista uma correlação entra os preditores e o atributo meta, porém significa que será necessário realizar testes com outros algoritmos.

3.3.5 DESAFIOS DA POPULAÇÃO

Durante a fase de pré-processamento e análise, diversas dificuldades foram encontradas devido a fatores específicos que impactaram o desenvolvimento e a eficácia dos modelos de aprendizado de máquina.

Um dos principais desafios foi adaptar a população aos algoritmos de classificação, uma vez que modelos de aprendizado de máquina geralmente se beneficiam de grandes volumes de dados, pois isto permite melhor captura de padrões subjacentes e maior precisão nas previsões.

Outro problema significativo foi a presença de dados faltantes. Alguns registros apresentavam valores ausentes em um ou mais atributos, o que poderia comprometer a análise se não fosse devidamente tratado.

Além disso, o desequilíbrio dos dados representa uma dificuldade adicional. Conforme observado no Apêndice 1 como destacado anteriormente a distribuição da amostra mostra-se assimétrica algumas classes de usuários tinham um número muito maior de instâncias, o que poderia influenciar negativamente a performance dos modelos. Técnicas como a reamostragem (*oversampling* ou *undersampling*) e o uso de algoritmos específicos para dados desbalanceados como SMOTE e Rose foram utilizadas para tentar mitigar esse problema na modelagem em R e em Python.

A multicolinearidade entre alguns atributos também foi identificada como um desafio. A alta correlação entre variáveis pode impactar negativamente modelos lineares como a Regressão Logística, tornando difícil distinguir os efeitos individuais de cada variável. Para abordar esse problema, métodos como a análise de componentes principais (PCA) foram considerados, permitindo a redução da dimensionalidade dos dados e a mitigação da multicolinearidade.

Por fim, a complexidade e a multidimensionalidade dos dados exigiram a utilização de modelos avançados e ajustes de hiperparâmetros para obter resultados. Este processo está presente na sessão 4.3 dos Resultados. Modelos complexos como

Redes Neurais Artificiais e *Random Forests*, embora poderosos, requerem um ajuste meticuloso para evitar *overfitting* e garantir a generalização dos resultados.

3.4 TÉCNICAS DE PRÉ PROCESSAMENTO, PROCESSAMENTO E ANÁLISE DE DADOS

A análise e processamento de dados são etapas cruciais em projetos de ciência de dados e aprendizado de máquina. Essas etapas garantem que os dados estejam preparados e otimizados para o desenvolvimento de modelos preditivos eficazes. As técnicas utilizadas nesse processo incluem a Análise Exploratória de Dados (EDA), seleção de características, normalização de dados, divisão dos dados em conjuntos de treinamento e teste, treinamento e avaliação de modelos preditivos e ajuste de hiperparâmetros (Tukey, 1977).

A seleção de características é o processo de escolher um subconjunto relevante de variáveis preditoras para usar em um modelo. Isso pode melhorar a precisão do modelo e reduzir a complexidade computacional. Métodos comuns incluem seleção baseada em correlação, análise de variância (ANOVA) e algoritmos como Random Forest e LASSO (Least Absolute Shrinkage and Selection Operator) (Guyon; Elisseeff, 2003).

A normalização de dados é a transformação dos dados para que eles fiquem em uma escala comum, o que é importante para muitos algoritmos de aprendizado de máquina que dependem da distância entre os dados, como K-means e SVM. Métodos comuns incluem Min-Max Scaling, que transforma os dados para um intervalo específico (normalmente 0 a 1), e Z-score normalization, que transforma os dados para que tenham média zero e desvio padrão igual a um (Han; Kamber; Pei, 2011).

Dividir os dados em conjuntos de treinamento e teste é essencial para avaliar o desempenho de um modelo de aprendizado de máquina. O conjunto de treinamento é usado para treinar o modelo, enquanto o conjunto de teste é usado para avaliar sua capacidade de generalização para novos dados. Uma divisão comum é 70% dos dados para treinamento e 30% para teste, mas isso pode variar (Hastie; Tibshirani; Friedman, 2009).

A validação cruzada é uma técnica essencial para avaliar a robustez e a capacidade de generalização de um modelo de aprendizado de máquina. No nosso estudo, utiliza-se a validação cruzada com 10 dobras, onde os dados são divididos

em 10 partes iguais. Em cada iteração, uma parte é utilizada para teste, enquanto as outras nove são usadas para treinamento. Este método é conhecido por fornecer uma estimativa mais confiável da performance do modelo em dados não vistos, pois todos os dados são usados tanto para treinamento quanto para teste (Kohavi, 1995).

O treinamento de modelos preditivos envolve o uso de dados de treinamento para ajustar um modelo que possa fazer previsões. A avaliação envolve o uso de métricas como acurácia, precisão, recall e F1-score para medir o desempenho do modelo em dados de teste. Isso ajuda a garantir que o modelo possa generalizar bem para novos dados não vistos (Bishop, 2006).

A validação cruzada é uma técnica essencial para avaliar a robustez e a capacidade de generalização de um modelo de aprendizado de máquina. No nosso estudo, utiliza-se a validação cruzada com 10 dobras, onde os dados são divididos em 10 partes iguais. Em cada iteração, uma parte é utilizada para teste, enquanto as outras nove são usadas para treinamento. Este método é conhecido por fornecer uma estimativa mais confiável da performance do modelo em dados não vistos, pois todos os dados são usados tanto para treinamento quanto para teste (Kohavi, 1995).

Os hiperparâmetros são parâmetros definidos antes do processo de treinamento de um modelo de machine learning, e são essenciais para ajustar o comportamento do algoritmo. Diferente dos parâmetros aprendidos durante o treinamento (como os coeficientes de uma regressão linear), os hiperparâmetros são configurados pelo usuário e podem influenciar significativamente a performance do modelo. Um exemplo de hiperparâmetro é o número de árvores em uma floresta aleatória (Random Forest) ou o valor de regularização em uma regressão logística (Müller; Guido, 2016).

O GridSearch é uma técnica utilizada para ajustar hiperparâmetros de modelos de machine learning. O processo envolve a definição de um conjunto de valores possíveis para cada hiperparâmetro e a avaliação exaustiva de todas as combinações possíveis desses valores. O objetivo é encontrar a combinação que resulta na melhor performance do modelo com base em uma métrica de avaliação especificada, como a acurácia ou a precisão.

3.5 ALGORITMOS CLASSIFICADORES

Os classificadores selecionados para a previsão de *churn*, conforme discutido na revisão de literatura, incluem uma variedade de métodos amplamente utilizados em aprendizado de máquina. A seleção dos algoritmos foi realizada levando em consideração os classificadores predominantes na literatura. Cada um desses métodos possui características específicas que os tornam adequados para diferentes tipos de dados e problemas de previsão.

A Regressão Logística é um dos métodos mais simples e comumente utilizados para prever a probabilidade de um evento específico, como o *churn*. Este método utiliza uma função logística para modelar a probabilidade de *churn* com base em várias variáveis independentes. A simplicidade da regressão logística facilita a interpretação dos resultados, permitindo que se identifiquem quais variáveis têm maior impacto na probabilidade de churn (Hosmer, Lemeshow, Sturdivant, 2013).

O *Random Forest*, por sua vez, é um método que combina várias árvores de decisão para fazer previsões. Cada árvore é construída a partir de uma amostra aleatória dos dados, e a previsão final é feita por votação majoritária das previsões de todas as árvores. Este método é particularmente eficaz em lidar com grandes conjuntos de dados e em minimizar o risco de *overfitting*, uma vez que as árvores individuais podem ser mais suscetíveis a variabilidades nos dados (Breiman, 2001).

Os *Support Vector Machines (SVM)* são modelos que buscam encontrar o hiperplano que melhor separa as classes nos dados. Este método é especialmente útil quando os dados não são linearmente separáveis, pois utiliza técnicas como o kernel trick para transformar os dados em um espaço de maior dimensão onde uma separação linear se torna possível. SVMs são conhecidos por sua robustez e capacidade de lidar com problemas complexos de classificação (Cortes, Vapnik, 1995).

Por fim, as Redes Neurais Artificiais (ANN) são modelos inspirados pela estrutura e função do cérebro humano, compostos por uma série de nós (neurônios) organizados em camadas e conectados por sinapses. As ANN são capazes de capturar padrões complexos nos dados e são eficazes em diversas aplicações, incluindo a previsão de *churn*. No entanto, requerem uma quantidade significativa de dados para treinar adequadamente e podem ser computacionalmente intensivas (Haykin, 2009).

Esses classificadores foram selecionados devido à sua eficácia comprovada em diversas aplicações de previsão de *churn*, variando em complexidade e adequação conforme o contexto e os dados disponíveis.

3.6 ALGORITMOS CLASSIFICADORES DE PREDITORES

Para avaliar a importância de cada atributo foram aplicados *Random Forest*, ANOVA e RFE com Regressão Logística. O método *Random Forest* é amplamente utilizado para a classificação e a avaliação da importância das características. Ele fornece uma medida de importância baseada na diminuição da impureza ou no aumento da precisão de previsão, quando uma variável é incluída no modelo. Breiman (2001) discute que a importância de uma característica em uma floresta aleatória é avaliada pela quantidade de ganho de informação (redução de impureza) que ela proporciona.

O método ANOVA é uma técnica estatística usada para comparar as médias de três ou mais grupos, verificando se pelo menos um deles é significativamente diferente dos outros. No contexto de seleção de características, ANOVA pode ser utilizada para determinar quais características são mais relevantes para a variável de resposta. Segundo Montgomery (2017), a ANOVA ajuda a identificar quais variáveis possuem um impacto significativo na variável dependente ao comparar as variâncias dentro dos grupos e entre os grupos.

O método RFE com Regressão Logística (*Recursive Feature Elimination*) é utilizado para selecionar características através de um processo recursivo de eliminação. Neste processo, a Regressão Logística é aplicada para classificar os atributos e, em cada iteração, as características menos importantes são removidas. Guyon *et al.* (2002) destacam que o RFE é eficaz para melhorar a precisão do modelo e a interpretação dos resultados ao eliminar variáveis irrelevantes ou redundantes.

Esses métodos são cruciais para a análise de dados em aprendizado de máquina, permitindo a identificação das variáveis mais relevantes para a previsão de *churn* e otimizando a performance dos modelos preditivos.

4 RESULTADOS

Na parte de criação de um modelo foram realizados testes de diversos algoritmos de classificação foram aplicados aos dados preparados, incluindo Regressão Logística, Árvores de Decisão, Florestas Aleatórias, Máquinas de Vetores de Suporte (SVM) e Redes Neurais em Python e R e suas respectivas bibliotecas. Cada modelo foi avaliado usando métricas como acurácia, precisão, recall, F1-score e área sob a curva ROC (AUC), com a validação cruzada de 10 dobras para testar a robustez e generalização dos modelos.

4.1 MODELAGEM DE DADOS EM R

Como se pode observar, o processo se inicia carregando as bibliotecas necessárias para a análise de dados e modelagem preditiva. São elas: caret, usada para facilitar a criação de modelos preditivos e validação cruzada; tidyverse, um conjunto de pacotes para manipulação de dados e visualização; nnet, para criar modelos de redes neurais; e1071, que contém funções para máquinas de vetores de suporte (SVM) e outras técnicas de machine learning; glmnet, para regressão logística com regularização (Lasso e Ridge); pROC, utilizada para calcular a AUC (Área Sob a Curva) e avaliar o desempenho de modelos de classificação; ROSE, para balanceamento de classes em datasets desbalanceados; rpart, para criar árvores de decisão; rpart.plot, que facilita a plotagem de árvores de decisão; ggpubr, um pacote para publicação de gráficos de alta qualidade; e skimr, que gera resumos estatísticos dos dados.

Depois os dados são carregados a partir de um arquivo CSV localizado no caminho especificado. Em seguida, as linhas com valores ausentes (NA) são removidas para garantir que os dados estejam completos para análise. A coluna CancelouServico é convertida para o tipo fator, para ser usada como variável de resposta em modelos de classificação. Os níveis da variável de resposta são então convertidos em nomes válidos para serem usados em modelos preditivos.

Por fim, a distribuição das classes na variável de resposta CancelouServico é impressa para verificar se há desbalanceamento entre as classes. Este passo é crucial para entender se há uma predominância de uma classe sobre a outra, o que pode impactar a performance dos modelos preditivos e pode requerer técnicas de

balanceamento, como o uso da biblioteca ROSE. Este código estabelece o ambiente para uma análise de machine learning em R, preparando os dados e carregando as bibliotecas necessárias para várias técnicas de modelagem e avaliação. A verificação da distribuição das classes é um passo crítico para identificar potenciais problemas de desbalanceamento que podem afetar os resultados dos modelos preditivos.

Antes de realizar a análise, os dados pré-processados, preparados e limpos. Todos os registros contendo valores nulos foram removidos para garantir a integridade dos dados.

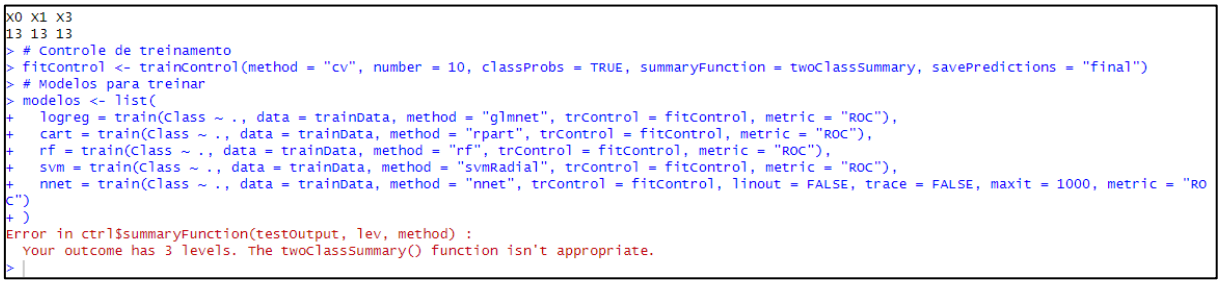
O pré-processamento foi realizado para evitar problemas durante a análise e o treinamento dos modelos, uma vez que valores nulos podem introduzir vieses e reduzir a qualidade do modelo. As variáveis categóricas, como "Cancelou Serviço", foram convertidas em valores numéricos 0 (Não cancelou), 1 (Cancelou), 2 (Não aplicável) para facilitar a análise. Este passo é essencial para que os algoritmos de machine learning possam interpretar corretamente essas variáveis. As colunas que não apresentavam variação foram removidas, uma vez que não contribuem para a análise. Esta remoção ajuda a reduzir a complexidade do modelo e melhora a eficiência do treinamento.

Devido ao desequilíbrio das classes (ou seja, um número significativamente menor de cancelamentos em comparação aos não cancelamentos), utiliza-se a técnica de *up-sampling* para balancear as classes no conjunto de treinamento. Este processo envolveu a replicação das amostras minoritárias até que houvesse um número aproximadamente igual de instâncias para ambas as classes.

Depois disso, um controle de treinamento é definido para usar validação cruzada com 10 divisões, calculando probabilidades de classe e a AUC como métrica. Vários modelos são treinados: regressão logística com regularização (glmnet), árvore de decisão (rpart), floresta aleatória (rf), SVM (svmRadial) e redes neurais (nnet). Os modelos são avaliados quanto à acurácia e AUC no conjunto de teste, gerando uma matriz de confusão e a AUC para cada modelo. O controle de treinamento incluiu a definição da métrica ROC (*Receiver Operating Characteristic*) como principal critério de avaliação.

É importante ressaltar que após o código rodar os algoritmos de treino ocorrem erros constantes que impedem a continuidade da pesquisa. Após diversas tentativas de correção e adequação do código, percebe-se que o problema é da amostra conforme pode ser observado na Figura 5.

FIGURA 5 – RESULTADO EM R



Fonte: O autor (2024).

Os resultados são exibidos e representados graficamente, comparando a acurácia e a AUC dos modelos. A árvore de decisão deveria ser plotada para visualização. Uma análise de correlação deveria realizada entre variáveis numéricas, seguida por testes de normalidade de Shapiro-Wilk e testes de correlação de Kendall entre as variáveis. Por fim, seriam gerados *boxplots* para visualizar a distribuição de várias variáveis em relação à variável.

As principais hipóteses giram em torno do fato do atributo meta CancelouServico não ser binário e a amostragem ter registros com menos de 50 pontos de observação, haver assimetrias nos registros dos eventos. Todo o processo em R está na seção Apêndice, do número 1 ao 13.

4.2 MODELAGEM DE APRENDIZAGEM DE MÁQUINA DE DADOS EM PYTHON

O primeiro passo no desenvolvimento de um projeto de análise e modelagem de dados é garantir que todas as bibliotecas necessárias estejam importadas. Neste caso, utiliza-se bibliotecas como numpy e pandas para manipulação de dados, sklearn para modelagem e avaliação de algoritmos de machine learning, e matplotlib e seaborn para visualização de dados. As bibliotecas importadas incluem modelos como RandomForestClassifier, LogisticRegression, SVC, e MLPClassifier para criar e avaliar diferentes algoritmos de classificação. Também importar as funções para seleção de características e avaliação de métricas, como SelectKBest, f_classif, RFE, accuracy_score, roc_auc_score, entre outras.

O próximo passo é carregar os dados de um arquivo CSV e preparar esses dados para modelagem. Utiliza-se a função pd.read_csv do pandas para carregar o dataset. Após isso, separa-se as características (features) do alvo (target). No exemplo fornecido, a coluna CancelouServico é definida como a variável alvo,

enquanto as outras colunas são usadas como características. Em seguida, divide-se os dados em conjuntos de treino e teste utilizando train_test_split, garantindo assim que 80% dos dados sejam utilizados para treino e 20% para teste, mantendo a aleatoriedade com random_state=42.

Para determinar qual modelo de machine learning performa melhor com os dados da amostra, define-se e avalia-se múltiplos algoritmos: Regressão Logística, *Random Forest*, SVM, e Redes Neurais. Por meio da a validação cruzada (cross-validation) com 10 folds para avaliar a performance inicial desses modelos. A função *cross_val_score* é usada para calcular a acurácia média e o desvio padrão de cada modelo. Esses resultados são armazenados em um dicionário e apresentados em um DataFrame para fácil visualização.

4.3 Análise dos resultados depois da validação cruzada

Os resultados apresentados nas Tabela 2 mostram as métricas de acurácia média e desvio padrão dos modelos antes e depois da validação cruzada.

TABELA 2 – ANTES DA VALIDAÇÃO CRUZADA

Modelo	**Acurácia Média**	**Desvio Padrão**
Logistic Regression	0,616667	0,236291
Random Forest	0,933333	0,133333
SVM	0,633333	0,233333
Neural Network	0,7	0,276887

Fonte: O autor (2024).

Nessa etapa inicial, a validação cruzada foi usada para avaliar a performance dos modelos com seus parâmetros padrão. O modelo *Random Forest* apresentou a maior acurácia média de 0,93 com um desvio padrão relativamente baixo, indicando uma boa performance consistente. A Regressão Logística, SVM e Rede Neural apresentaram acurácias médias mais baixas, com desvio padrão variando entre 0,23 e 0,27, indicando variabilidade nos resultados

Após a avaliação inicial dos modelos, foca-se em otimizar os hiperparâmetros do modelo *Random Forest* utilizando GridSearchCV. Define-se uma grade de parâmetros (param_grid) e, então, aplica-se o GridSearchCV para identificar a combinação de hiperparâmetros que proporciona o melhor desempenho. Em seguida,

treina-se o modelo Random Forest com os melhores parâmetros encontrados e, assim, valida-se sua eficácia.

O ajuste de hiperparâmetros é utilizado para melhorar a performance dos modelos de aprendizado de máquina. Utiliza-se o GridSearchCV para realizar uma busca exaustiva dos melhores valores para os hiperparâmetros do modelo Random Forest (Bergstra & Bengio, 2012).

GridSearchCV é uma técnica utilizada em Machine Learning para encontrar a melhor combinação de hiperparâmetros de um modelo. Seu principal objetivo é otimizar o desempenho do modelo, realizando uma busca exaustiva sobre um espaço especificado de parâmetros e utilizando validação cruzada para avaliar a performance das diferentes combinações.

O funcionamento do GridSearchCV envolve várias etapas. Primeiro, define-se o modelo e a grade de hiperparâmetros a serem otimizados. Em seguida, cria-se uma instância do GridSearchCV com o modelo, a grade de parâmetros e configurações de validação cruzada. O GridSearchCV então treina o modelo para cada combinação de hiperparâmetros, utilizando validação cruzada para garantir uma avaliação robusta. Após a busca, a melhor combinação de hiperparâmetros é identificada e o modelo otimizado é extraído.

As vantagens do GridSearchCV incluem a busca exaustiva que garante encontrar a melhor combinação possível dentro do espaço de busca e da validação cruzada que reduz a chance de overfitting e a automatização do processo de ajuste de hiperparâmetros, tornando-o mais eficiente. Na Tabela 3 há os resultados do GridSearchCV para otimização do hiperparâmetro.

TABELA 3 – HIPERPARÂMETRO

Hiperparâmetro	**Valores**
n_estimators	50, 200
max_features	'sqrt', log2
max_depth	10, none
min_samples_split	2, 5
min_samples_leaf	1, 2

Fonte: O autor (2024).

Após a validação cruzada com ajuste de hiperparâmetros, todos os modelos mostraram uma queda na acurácia média e um desvio padrão de 0, indicando que a

variabilidade nos resultados foi eliminada, mas com perda significativa de performance.

Como observado na Tabela 3, o ajuste de hiperparâmetros resulta em um modelo Random Forest com uma acurácia perfeita de 1,00 no conjunto de teste. Embora isso possa parecer ideal, uma acurácia de 100% pode indicar overfitting, onde o modelo se ajusta tão bem aos dados de treinamento que perde a capacidade de generalizar para novos dados. A matriz de confusão e o relatório de classificação devem ser examinados para verificar a generalização do modelo.

TABELA 4 – DEPOIS DA VALIDAÇÃO CRUZADA

Modelo	Acurácia Média	Desvio Padrão
Logistic Regression	0,666667	0
Random Forest	0,666667	0
SVM	0,5	0
Neural Network	0,5	0

Fonte: O autor (2024).

A distribuição do conjunto de dados pode variar entre treino e teste, resultando em *overfitting* durante o ajuste de hiperparâmetros. As características dos dados podem não ser capturadas adequadamente pelos modelos ajustados, levando a uma performance inferior nos dados de teste. Modelos complexos como Random Forest podem se adaptar excessivamente aos dados de treino, mas falhar em generalizar para novos dados devido à configuração dos hiperparâmetros.

O desbalanceamento de classes no conjunto de dados também pode influenciar negativamente as métricas de performance, como a acurácia, que pode não refletir a verdadeira eficácia do modelo. A validação cruzada com um pequeno número de amostras por classe pode não fornecer uma avaliação robusta, resultando em métricas que não refletem a capacidade de generalização dos modelos.

Para resolver esta questão foi utilizado o SMOTE (*Synthetic Minority Over-sampling Technique*) ao fluxo de trabalho de modelagem é uma etapa fundamental para lidar com o desbalanceamento de classes em conjuntos de dados de classificação. O SMOTE gera novas amostras sintéticas para a classe minoritária, equilibrando a distribuição das classes e melhorando a performance do modelo. Porém, a remodelagem com SMOTE resulta em um erro.

Por isso, é essencial considerar outras métricas e técnicas de validação, além da acurácia, para garantir que o modelo escolhido generalize bem para novos dados. Reavaliar os modelos com técnicas adicionais de balanceamento de dados e métodos de validação mais robustos pode ser necessário para melhorar os resultados e assegurar uma performance consistente.

Com os modelos ajustados e treinados, avalia-se o desempenho utilizando várias métricas como acurácia, precisão, recall, F1-score e AUC. Também se gera a matriz de confusão e o relatório de classificação para cada modelo. Como se pode observar na Tabela 5, as matrizes de confusão fornecem uma visão mais detalhada dos erros de classificação de cada modelo.

TABELA 5 – MATRÍZ DE CONFUSÃO

Modelo	**Previsão Negativa (Neg, Neg)**	**Previsão Positiva (Neg, Pos)**	**Previsão Negativa (Pos, Neg)**	**Previsão Positiva (Pos, Pos)**
Regressão Logística	1	2	0	3
Random Forest	2	1	0	3
SVM	0	3	0	3
Neural Network	0	3	0	3

Fonte: O autor (2024).

O *Random Forest* apresenta a melhor distribuição, com apenas um falso positivo e nenhum falso negativo, o que contribui para sua alta precisão e *recall*. A Regressão Logística teve dois falsos positivos e nenhum falso negativo, o que explica sua alta precisão, mas um recall ligeiramente menor em comparação com o *Random Forest*. Os modelos SVM e Rede Neural não conseguiram classificar corretamente nenhum dos negativos, resultando em uma matriz de confusão que mostra todos os exemplos negativos sendo classificados como positivos.

Finalmente, forma comparados os modelos utilizando curvas de precisão-recall. Essas curvas são úteis para avaliar o desempenho dos modelos em problemas de classificação desbalanceada. Ademais, elaborou-se o Gráfico 1 com um

comparativo das com as curvas para cada modelo, permitindo uma comparação visual de suas performances em termos de precisão e recall.

A partir das matrizes de confusão extraiu-se outras métricas importantes como precisão, *recall* e F1-score, como observado na Tabela 6.

TABELA 6 – MÉTRICAS DE CLASSIFICADORES

Modelo	Acurácia	Precisão	Recall	F1-score	AUC
Regressão Logística	0,67	0,87	0,78	0,75	-
Random Forest	0,67	0,87	0,78	0,75	-
SVM	0,5	0,87	0,78	0,75	-
Neural Network	0,5	0,83	0,67	0,56	-

Fonte: O autor (2024).

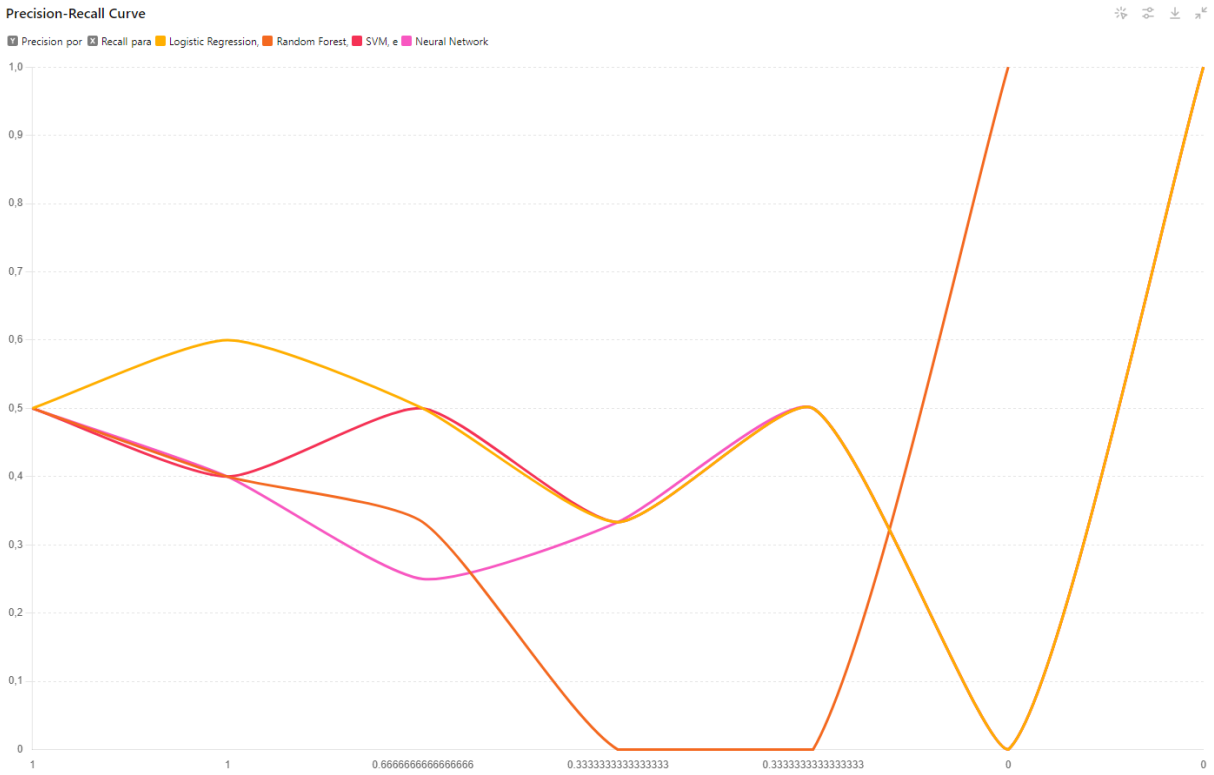
Ao analisar a Tabela 6, pode-se afirmar que os modelos Regressão Logística e *Random Forest* têm o melhor desempenho geral, com as maiores pontuações em todas as métricas (acurácia, precisão, *recall* e F1-score). Isto sugere que ambos modelos são mais eficazes para a previsão de cancelamento de serviço neste conjunto de dados.

No entanto, o *recall* é ligeiramente inferior ao *Random Forest*, indicando que pode não ser tão eficaz em identificar todos os casos positivos. Os modelos SVM e Rede Neural tiveram desempenho semelhante, ambos com acurácia, *recall* e F1-score mais baixos. Estes modelos podem não ser os mais adequados para este problema específico, dado o desempenho inferior em comparação com os outros modelos.

Destaca-se, ainda, que a curva ROC (*Receiver Operating Characteristic*) e a área sob a curva (AUC - *Area Under the Curve*) não puderam ser calculadas para os modelos devido à ausência de previsões probabilísticas em alguns casos.

Para a obtenção de uma perspectiva diferente gera-se o Gráfico 1, contendo a curva de Recall com os quatro modelos analisados, onde o eixo Y é o indicador Precisão, e o eixo X é indicador *Recall*.

GRÁFICO 1 – CURVA DE PRECISÃO RECALL



Fonte: O autor (2024).

No Gráfico 1, a curva mostra a relação entre recall e precisão para diferentes limiares de classificação. As curvas de precisão-recall são úteis quando existe uma distribuição de classes desequilibrada e se pretende focar em minimizar falsos negativos e falsos positivos. O que se observa é que as curvas mais próximas do canto superior direito indicam melhores desempenhos. Nota-se, ainda, que os modelos Regressão Logística e *Random Forest* parecem ter o desempenho mais consistente em termos de precisão e *recall*, seguido pelos outros modelos.

4.4 MÉTODO PARA A ANALISE DE CARACTERÍSTICAS EM PYTHON

Na segunda parte da pesquisa, a análise de importância das características foi realizada usando três métodos: Importância das Características aplicando os classificadores *Random Forest*, ANOVA e RFE com Regressão Logística.

O classificador *Random Forest* avalia a importância de cada característica com base na redução da impureza que cada característica proporciona ao ser usada para dividir os nós das árvores. Primeiro, os dados são preparados através da limpeza, tratamento de valores ausentes, codificação de variáveis categóricas e normalização.

Em seguida, o modelo *Random Forest* é treinado construindo múltiplas árvores de decisão a partir de diferentes subconjuntos dos dados e características. Após o treinamento, a importância das características é calculada com base na redução da impureza (índice Gini), acumulando a redução da impureza em todas as árvores e normalizando-a pelo número total de árvores (Breiman, 2001).

Após realizar a análise com os resultados foi possível listar o grau de importância de cada uma das variáveis onde a importância é medida pelo número maior. Como é visto na Tabela 7, nota-se que a característica mais relevante no teste *Random Forest* é Adicionar Fornecedor, a segunda é Busca e logo a seguir há CriarSolicitacao.

TABELA 7 – CARACTERÍSTICA RANDOM FOREST

Característica	**Importância**
AdicionarFornecedor	0,201
Busca	0,17
CriarSolicitacao	0,141
NumeroDeLogins	0,129
AdicionarFabrica	0,119
AdicionarProduto	0,093
EditarInformacoes	0,062
ArquivarSolicitacao	0,056
UsuariosUnicos	0,03
SessaoMediaEmMinutos	0
TipoDeConta	0

Fonte: O autor (2024).

A análise da Tabela 7 revela que três ações dos usuários – adicionar fornecedores, buscar informações e criar solicitações – são as mais influentes no modelo *Random Forest*, indicando que estas atividades são relevantes para prever futuros cancelamentos. A frequência de logins também é significativa, refletindo o nível de engajamento dos usuários. Por outro lado, a duração média das sessões e o tipo de conta não impactam as previsões.

O segundo método, ANOVA (Análise de Variância), é utilizado para avaliar a relevância de cada característica comparando as médias e variâncias entre diferentes

grupos. Este método baseia-se em medir quanta variação nas respostas pode ser atribuída às diferentes características em vez da variação aleatória. No contexto do classificador, o algoritmo ANOVA realiza uma listagem das características com base nas diferenças nas médias e variâncias entre os grupos distintos. Isto ajuda a identificar quais características têm um impacto significativo nas diferenças observadas nos dados, permitindo determinar a relevância de cada uma com maior precisão (Fisher, 1925).

A partir dos resultados gera-se a Tabela 8, na qual as pontuações mais elevadas de cada característica mostram que existe uma diferença maior entre as médias dos grupos, enquanto valores menores resultam em diferença insignificativa.

TABELA 8 – CARACTERÍSTICA ANOVA

Característica	**Pontuação**
Busca	7,338
NumeroDeLogins	4,185
AdicionarFabrica	3,32
CriarSolicitacao	3,205
EditarInformacoes	2,935
UsuariosUnicos	2,353
AdicionarFornecedor	1,618
ArquivarSolicitacao	0,878
SessaoMediaEmMinutos	0,256
AdicionarProduto	0,195
TipoDeConta	0

Fonte: O autor (2024).

O quadro de pontuações ANOVA revela *insights* significativos sobre a relevância das características no modelo. Observa-se que "Busca" é a característica mais relevante, apresentando a maior diferença entre as médias dos grupos, indicando sua influência crítica nas variáveis dependentes. Em contraste, "Tipo de Conta" é considerada a menos importante, com uma pontuação de zero, indicando que não há diferenças significativas entre os grupos em relação a essa característica. Esta análise destaca a importância de focar em otimizar a funcionalidade de busca para melhorar a eficácia do sistema, enquanto características como o tipo de conta podem ser consideradas menos prioritárias para ajustes ou melhorias.

A Eliminação Recursiva de Características (RFE) é uma técnica de seleção de características que identifica e elimina recursivamente as características menos importantes para construir um modelo preditivo mais eficiente. Quando aplicada com Regressão Logística, o RFE começa treinando um modelo de Regressão Logística com todas as características disponíveis. Em seguida, avalia-se a importância de cada característica com base nos coeficientes absolutos da Regressão Logística. A característica menos importante (aquela com o menor coeficiente) é então eliminada. Este processo é repetido iterativamente, removendo uma característica por vez, até que o conjunto desejado de características seja alcançado. Este método ajuda a identificar o subconjunto de características mais relevante, melhorando a precisão e a interpretabilidade do modelo (Guyon *et al.*, 2002).

A Eliminação Recursiva de Características (RFE) com Regressão Logística para identificar as características mais importantes no conjunto de dados de treinamento. Inicialmente, o modelo de Regressão Logística é configurado para um máximo de 1000 iterações e, em seguida, o RFE é aplicado para eliminar recursivamente as características menos relevantes até que reste apenas uma. Após ajustar o modelo, o ranking das características é obtido e organizado em um DataFrame, que é então ordenado para exibir as características em ordem de importância, conforme observado na Tabela 9, a seguir.

TABELA 9 – CARACTERÍSTICA RFE

Característica	**Rank**
Busca	1
NumeroDeLogins	2
AdicionarFabrica	3
CriarSolicitacao	4
EditarInformacoes	5
UsuariosUnicos	6
AdicionarFornecedor	7
ArquivarSolicitacao	8
SessaoMediaEmMinutos	9
AdicionarProduto	10
TipoDeConta	11

Fonte: O autor (2024).

No caso da Tabela 9, a RFE classifica "Busca" como a característica mais relevante (rank 1), seguida por "NumerodeLogins" (rank 2) e "AdicionarFabrica" (rank 3), indicando sua alta importância para o modelo. As características menos relevantes são "Adicionar Produto" (rank 10) e "TipodeConta" (rank 11), sendo as primeiras a serem eliminadas durante o processo, o que sugere que elas contribuem pouco para o desempenho do modelo.

Observando os resultados, *Random Forest* e Regressão Logísitca apresentam melhorers acurácia média. No entanto, antes da validação cruzada todos os modelos mostram desvios padrões relativamente altos, indicando que os resultados podem variar bastante dependendo do conjunto de dados ou das divisões de treinamento/teste. Seria útil investigar mais sobre a consistência desses modelos em diferentes subconjuntos de dados ou considerar ajustes nos parâmetros para melhorar a estabilidade.

Os resultados da validação cruzada também evidenciam que a amostra precisaria ter mais observações de empresas pagantes. Este indício sugere que, para o conjunto de dados específico, modelos mais complexos não necessariamente garantem melhor desempenho, uma descoberta que está alinhada com o fenômeno de "*No Free Lunch*" em *machine learning* (Wolpert, 1996). De acordo com este teorema, não existe um modelo universalmente superior; a eficácia de um modelo depende muito das características específicas dos dados. Ainda assim, a superioridade da *Random Forest* neste aspecto pode ser atribuída à sua capacidade de lidar com interações não lineares e sua robustez ao *overfitting*, especialmente em conjuntos de dados com muitas variáveis preditivas (Breiman, 2001).

A análise de importância das características revela que "AdicionarFornecedor", "Busca" e "CriarSolicitacao" são consistentemente as características mais influentes para o cancelamento de serviço. Esses resultados são consistentes entre os métodos utilizados, o que reforça a confiabilidade dessas características como indicadores importantes. Este resultado enfatiza a necessidade de focalizar nos comportamentos dos usuários, como a frequência e duração das sessões, que são indicativos de engajamento e possível satisfação com o serviço, aspectos estes frequentemente vinculados à retenção de clientes em muitos estudos de *churn* (Xie *et al.*, 2009).

Portanto, a pesquisa alcança seus objetivos por meio de uma revisão sistemática da literatura, coleta e preparação cuidadosa de dados, aplicação rigorosa

de múltiplos algoritmos de classificação, e uma comparação detalhada do desempenho dos modelos, resultando na identificação do *Random Forest* como o método mais adequado para previsão de *churn* no contexto da organização analisada. Além disto, foi possível determinar por meio dos classificadores os preditores que o preditor “Busca” é o que mais influenciam na taxa de desistência em dois dos três classificadores (ANOVA e RFE) e o segundo no *Random Forest*.

5 CONSIDERAÇÕES FINAIS

Após os resultados obtidos neste estudo, conclui-se que a previsão de churn em empresas de software B2B é uma tarefa desafiadora para gestores de informação. Isto porque requer a aplicação e compreensão de uma gama de conhecimentos específicos e aprofundados em Ciência da Informação, Ciência da Computação e Estatística.

A identificação de preditores de *churn* apresenta-se como um “problema interessante” para trabalhos que envolvem técnicas de *machine learning*, uma vez que os comportamentos dos clientes podem ser influenciados por uma combinação de múltiplos fatores, dificultando a diferenciação entre eles.

5.1 VERIFICAÇÃO DOS OBJETIVOS PROPOSTOS

Para atingir o objetivo geral deste estudo — identificar os principais preditores de cancelamento de assinaturas utilizando algoritmos de classificação — foi necessário alcançar cinco objetivos específicos. Abaixo, detalha-se como cada um destes objetivos foi verificado e alcançado.

5.1.1 Conhecimento das práticas

Para alcançar o primeiro objetivo específico – Conhecer as práticas atuais sobre churn e a utilização de classificadores para a sua prevenção – realizou-se uma revisão sistemática da literatura sobre *churn* e a utilização de classificadores para a sua prevenção. Este objetivo foi completado por meio de uma busca nas bases de dados acadêmicas *Web of Science* e Scielo, que permitiu a identificação dos principais conceitos, aplicações, tecnologias, extensão do tema e histórico das pesquisas realizadas. O resultado foi a seleção e análise de 32 artigos científicos relevantes, proporcionando uma base sólida para a compreensão dos métodos e algoritmos utilizados na previsão de *churn*.

5.1.2 Comparação de abordagens

O segundo objetivo específico – Comparar as abordagens existentes sobre churn e a utilização de classificadores – foi atingindo ao se comparar os diversos algoritmos de classificação, incluindo Regressão Logística, Árvores de Decisão, *Random Forest*, SVM e Redes Neurais Artificiais. Cada algoritmo foi avaliado quanto ao seu desempenho por meio de métricas como acurácia, precisão, *recall* e F1-score. O *Random Forest* destaca-se como o algoritmo com melhor desempenho, apresentando alta acurácia e robustez na previsão de *churn*. Esta comparação criteriosa permitiu a seleção dos algoritmos mais adequados para aplicação na base de dados da pesquisa.

5.1.3 Coleta e análise de dados

O terceiro objetivo específico – Analisar dados históricos de clientes atuais e que cancelaram os contratos na organização – foi alcançado a partir da coleta e análise de dados dos clientes da organização analisada. A coleta de dados foi realizada através da plataforma Mixpanel, onde foram extraídos registros de eventos das interações dos clientes com o serviço oferecido pela organização. A população incluiu 26 empresas do setor alimentício, totalizando 336 registros. Técnicas de pré-processamento, como limpeza de dados e balanceamento de classes, foram aplicadas para garantir a qualidade e representatividade dos dados. A análise descritiva dos dados permitiu identificar os padrões de uso e comportamento dos clientes, estabelecendo uma base sólida para a construção do modelo preditivo.

5.1.4 Construção de um modelo quantitativo

O quarto objetivo específico – Desenvolver um modelo quantitativo baseado em dados históricos e em padrões identificados – envolvia a construção de um modelo preditivo. A partir do uso dos dados coletados e pré-processados, diversos algoritmos de classificação foram aplicados, incluindo *Random Forest*, Regressão Logística, SVM e Redes Neurais. A performance de cada modelo foi rigorosamente avaliada, e a Regressão Logística e o Random Forest novamente destacam-se como métodos mais eficazes, com alta acurácia na previsão de *churn*. Este processo de modelagem

quantitativa foi fundamental para identificar os preditores mais relevantes e construir um modelo robusto e confiável.

5.1.5 Análise dos Classificadores e Preditores

O quinto e último objetivo específico – Avaliar os classificadores e predicadores que influenciam no cancelamento de serviços – foi alcançado e a análise dos resultados permitiu identificar comportamentos dos clientes que estão fortemente associados ao *churn*, como a frequência de logins, a duração média das sessões, o número de solicitações criadas e a interação com o serviço. Por meio do uso de técnicas de análise de características como ANOVA e RFE (Recursive Feature Elimination), foi possível destacar os preditores mais influentes, e fornecer *insights* valiosos para a implementação de estratégias de retenção de clientes na organização analisada.

Todos os objetivos específicos foram alcançados com sucesso, permitindo a identificação dos principais preditores de *churn* em empresas de SaaS no mercado B2B, e a construção de um modelo quantitativo eficaz utilizando algoritmos de classificação. Estes resultados contribuem significativamente para a área de Gestão da Informação, oferecendo novas possibilidades para a aplicação de técnicas de mineração de dados na previsão de *churn* e na retenção de clientes.

5.2 CONTRIBUIÇÃO

Esta pesquisa é uma contribuição para a área de Gestão da Informação, demonstrando a aplicação prática de técnicas de mineração de dados para fins de classificação em um contexto empresarial B2B. A pesquisa mostra como algoritmos de *machine learning* podem ser eficazmente utilizados para prever *o churn* de clientes, um desafio crítico para empresas de software que operam sob o modelo SaaS (Software as a Service). Esta abordagem não apenas reforça a importância da análise de dados avançada na gestão de informações, mas também oferece um modelo prático e replicável para outros setores empresariais que enfrentam problemas similares de retenção de clientes.

Este trabalho abre novas possibilidades para a aplicação de modelos preditivos em diferentes setores. A metodologia e os resultados apresentados podem

ser adaptados e aplicados em diversas indústrias, como telecomunicações, finanças e comércio eletrônico, onde a retenção de clientes é igualmente crítica. Ao demonstrar a eficácia dos algoritmos de *machine learning* na previsão de *churn*, o estudo promove a adoção dessas tecnologias em contextos variados, contribuindo para a sustentabilidade das empresas e a melhoria da satisfação do cliente.

Durante a condução deste estudo, foi realizada uma reunião com os tomadores de decisão da organização analisada, na qual foi possível observar um interesse significativo em implementar o modelo preditivo desenvolvido. Os gestores da organização expressaram a intenção de coletar amostras de dados com períodos de tempo mais curtos, de um a três meses, o que pode proporcionar *insights* mais dinâmicos e oportunos sobre o comportamento dos clientes. Além disto, houve discussões sobre a criação de mais eventos de monitoramento e a melhoria da profundidade das análises. Estas ações visam aumentar a precisão e a relevância dos modelos preditivos, permitindo intervenções mais eficazes e personalizadas para a retenção de clientes.

Desse modo, este estudo não apenas alcança seus objetivos, como também gera um impacto prático significativo, oferecendo ferramentas e *insights* que podem ser imediatamente aplicados pela organização analisada, e servir como modelo para ser aplicado em organizações similares, além de servir como base ser para futuras pesquisas na Gestão da Informação.

5.3 PESQUISAS FUTURAS

Embora os resultados sejam promissores, as limitações do estudo incluem a dependência de hiperparâmetros que foram ajustados apenas para o *Random Forest*. Estudos futuros poderiam explorar ajustes de hiperparâmetros mais extensos para outros modelos para garantir uma comparação justa.

Além disso, a aplicabilidade dos modelos poderia ser testada em conjuntos de dados externos para validar a robustez dos modelos. O uso de técnicas de interpretabilidade de modelos, como SHAP (SHapley Additive exPlanations) ou LIME (Local Interpretable Model-agnostic Explanations), poderia também oferecer *insights* mais profundos sobre como os modelos estão tomando decisões, crucial para aplicações em ambientes regulados.

REFERÊNCIAS

ASSI, M. *et al.* Predicting the change impact of resolving defects by leveraging the topics of issue reports in open source software systems. **ACM Transactions on Software Engineering and Methodology**, v.32, n. 6, p. 1-34, 2023. DOI: https://dx.doi.org/10,1145/3593802. Disponível em: https://dl.acm.org/doi/10,1145/3593802. Acesso em: 14 jun. 2023.

AZEVEDO, A.; SANTOS, M. F. KDD, SEMMA and CRISP-DM: a parallel overview. *In*: IADIS EUROPEAN CONFERENCE ON DATA MINING, 2008, Amsterdam. **Proceedings**… Amsterdam: IADIS, 2008. p. 182-185. Disponível em: https://recipp.ipp.pt/bitstream/10400,22/136/3/KDD-CRISP-SEMMA.pdf. Acesso em: 7 mar. 2024.

BANDYOPADHYAY, N.; JADHAV, A. S. Churn prediction of employees using machine learning techniques. **Tehnički glasnik**, v. 15, n. 1, p. 51-59, 2021. DOI: https://dx.doi.org/10,31803/TG-20210204181812. Disponível em: https://hrcak.srce.hr/253021. Acesso em: 2 jul. 2023.

BANU, F. *et al.* Artificial Intelligence Based Customer Churn Prediction Model for Business Markets. **Computational Intelligence Neuroscience**, Bethesda, v. 2022, n. 1, p. 1-14, 2022. DOI: https://doi.org/10.1155%2F2022%2F1703696. Disponível em: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9552693. Acesso em: 13 abr. 2024.

BISHOP, C. M. **Pattern recognition and machine learning**. Cidade de Nova Iorque: Springer, 2006.

BOGAERT, M.; DELAERE, L. Ensemble methods in customer churn prediction: a comparative analysis of the state-of-the-art. **Mathematics**, v. 11, n. 5, p. 1-28, 2023. DOI: https://dx.doi.org/10,3390/math11051137. Disponível em: https://www.mdpi.com/2227-7390/11/5/1137. Acesso em: 2 jul. 2023.

BRADLEY, A. P. The use of the area under the ROC curve in the evaluation of machine learning algorithms. **Pattern Recognition**, v. 30, n. 7, p. 1.145-1.159, 1997. DOI: https://doi.org/10.1016/S0031-3203(96)00142-2. Disponível em: https://www.sciencedirect.com/science/article/abs/pii/S0031320396001422?via%3Dihub. Acesso em: 13 abr. 2024.

BREIMAN, L. *et al.* **Classification and regression trees**. Belmont: Wadsworth, 1986.

BREIMAN, L. Random forests. **Machine learning**, v. 45, n. 1, p. 5-32, 2001. DOI: https://doi.org/10,1023/A:1010933404324. Disponível em: https://link.springer.com/article/10,1023/A:1010933404324. Acesso em: 10 mar. 2024.

CHAPMAN, P. *et al.* **CRISP-DM 1.0**: step-by-step data mining guide. SPSS, 2000, Disponível em: https://www.kde.cs.uni-kassel.de/wp-content/uploads/lehre/ws2012-13/kdd/files/CRISPWP-0800,pdf. Acesso em: 10 mar. 2024.

CORMEN, T. H. *et al.* **Algoritmos**: teoria e prática. Rio de Janeiro: Elsevier, 2009.

CORTES, C.; VAPNIK, V. Support-vector networks. **Machine learning**, v. 20, n. 3, p. 273-297, 1995. DOI: https://doi.org/10,1007/BF00994018. Disponível em: https://link.springer.com/article/10,1007/BF00994018. Acesso em: 12 mar. 2024.

DAVIS, J.; GOADRICH, M. The relationship between precision-recall and ROC curves. *In*: INTERNATIONAL CONFERENCE ON MACHINE LEARNING, 23., 2006, Pittsburgh. **Proceedings**… Pittsburgh: ICML, 2006, p. 233-240. DOI: https://doi.org/10,1145/1143844.1143874. Disponível em: https://dl.acm.org/doi/10,1145/1143844.1143874. Acesso: 12 mar. 2024.

FATIMA, E. B. *et al.* Minimizing the overlapping degree to improve class-imbalanced learning under sparse feature selection: application to fraud detection. **IEEE Access**, vol. 9, n. 1, p. 28.101-28.110, 2021. DOI: https://dx.doi.org/10,1109/ACCESS.2021.3056285. Disponível em: https://ieeexplore.ieee.org/document/9343840. Acesso em: 2 jul. 2023.

FAWCETT, T. An introduction to ROC analysis. **Pattern Recognition Letters**, v. 27, n. 8, p. 861-874, 2006. DOI: https://doi.org/10,1016/j.patrec.2005.10,010. Disponível em: https://www.sciencedirect.com/science/article/abs/pii/S016786550500303X. Acesso em: 12 maio 2024.

FISHER, R. A. **Statistical methods for research workers**. Edinburgh: Oliver and Boyd, 1925.

FRIEDMAN, J. H. Greedy function approximation: a gradient boosting machine. **Annals of Statistics**, v. 29, n. 5, p. 1.189-1.232, 2001. DOI: https://doi.org/10,1214/aos/1013203451. Disponível em: https://projecteuclid.org/journals/annals-of-statistics/volume-29/issue-5/Greedy-function-approximation-A-gradient-boosting-machine/10,1214/aos/1013203451.full. Acesso em: 28 abr. 2024.

FRIEDMAN, J.; HASTIE, T.; TIBSHIRANI, R. Regularization Paths for Generalized Linear Models via Coordinate Descent. **Journal of Statistical Software**, v. 33, n. 1, p. 1-22, 2010, DOI: https://doi.org/10,18637/jss.v033.i01. Disponível em: https://www.jstatsoft.org/article/view/v033i01. Acesso em: 13 abr. 2024.

GALLO, A. The value of keeping the right customers. **Harvard Business Review**, 29 out. 2014. Disponível em: https://hbr.org/2014/10/the-value-of-keeping-the-right-customers. Acesso em: 28 abr. 2024.

GARCÍA, J. M.; HERNÁNDEZ, L. M. Características y aplicaciones de la investigación cuantitativa. **Revista Cubana de Medicina General Integral**, Havana, v. 36, n. 6, p. 1-12, 2020, Disponível em:

http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S1990-86442020000600065. Acesso em: 20 mar. 2024.

GIL, A. C. **Métodos e técnicas de pesquisa social**. São Paulo: Atlas, 2008.

GUPTA, S.; LEHMANN, D. **Managing customers as investments**: the strategic value of customers in the long run. Cidade de Nova Iorque: FT Press, 2005.

GUYON, I. *et al.* Gene selection for cancer classification using support vector machines. **Machine Learning**, v. 46, n. 1-3, p. 389-422, 2002. DOI: https://doi.org/10,1023/A:1012487302797. Disponível: https://link.springer.com/article/10,1023/A:1012487302797. Acesso em: 28 abr. 2024.

GUYON, I.; ELISSEEFF, A. An introduction to variable and feature selection. **JMLR:** Journal of Machine Learning Research, v. 3, n. 1, p. 1.157-1.182, 2003. Disponível em: https://www.jmlr.org/papers/volume3/guyon03a/guyon03a.pdf. Acesso em: 28 abr. 2024.

HAIR, J. F. *et al.* When to use and how to report the results of PLS-SEM. **European Business Review**, v. 31, n. 1, p. 2-24, 2019. DOI: https://doi.org/10,1108/EBR-11-2018-0203. Disponível em: https://www.emerald.com/insight/content/doi/10,1108/EBR-11-2018-0203/full/html. Acesso em: 24 mar. 2024.

HALL, J. The role of customer retention in SaaS companies. **Journal of Business Strategy**, v. 40, n. 6, p. 30-37, 2019. Disponível em: https://www.emerald.com/insight/content/doi/10,1108/JBS-08-2019-0158/full/html. Acesso em: 20 jun. 2023.

HAN, J.; KAMBER, M.; PEI, J. **Data mining**: concepts and techniques. San Francisco: Morgan Kaufmann, 2011.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. **The elements of statistical learning**: data mining, inference, and prediction. Cidade de Nova Iorque: Springer, 2009.

HAYKIN, S. **Neural networks and learning machines**. Washington: Pearson, 2009.

HOFMANN, M.; KLINKENBERG, R. **RapidMiner**: data mining use cases and business analytics applications. Boca Raton: Chapman and Hall/CRC, 2013.

HOSMER JR, D. W., LEMESHOW, S., STURDIVANT, R. X. **Applied logistic regression**. New Jersey: John Wiley & Sons, 2013.

HUNTER, J. D. Matplotlib: a 2D graphics environment. **Computing in Science & Engineering**, v. 9, n. 3, p. 90-95, 2007. Disponível em: https://ieeexplore.ieee.org/document/4160265. Acesso em: 3 abr. 2024.

JAMALIAN, E.; FOUKERDI, R. A hybrid data mining method for customer churn prediction. **Engineering, Technology & Applied Science Research**, Atenas, v. 8,

n. 3, p. 2.991-2.997, 2018. DOI: https://doi.org/10,48084/etasr.2108. Disponível em: https://etasr.com/index.php/ETASR/article/view/2108. Acesso em: 23 mar. 2024.

KASSAMBARA, A. **'ggplot2' Based Publication Ready Plots**. R package version 0,4.0, 2020, Disponível em: https://cran.r-project.org/package=ggpubr. Acesso em: 23 mar. 2024.

KOHAVI, R. A study of cross-validation and bootstrap for accuracy estimation and model selection. *In*: INTERNATIONAL JOINT CONFERENCE ON ARTIFICIAL INTELLIGENCE, 14., 1995, Montreal. **Proceedings**… San Francisco: Morgan Kaufmann, 1995, p. 1137-1145. Disponível em: https://dl.acm.org/doi/10,5555/1643031.1643047. Acesso em: 24 mar. 2024.

KUHN, M. **Classification and regression training**. R package version 6.0-86, 2020, Disponível em: https://cran.r-project.org/package=caret. Acesso em: 19 jun. 2024.

KUMAR, V.; PETERSEN, J. A. Customer Relationship Management in SaaS: Trends and Practices. **Journal of Marketing Research**, v. 59, n. 1, p. 123-137, 2022. Disponível em: https://journals.sagepub.com/doi/abs/10,1177/00222437211052744. Acesso em: 20 jun. 2024.

LIU, R. *et al.* An Intelligent Hybrid Scheme for Customer Churn Prediction Integrating Clustering and Classification Algorithms. **Applied Science**, Basel, v. 12, n. 18, p. 1-17, 2022. DOI: https://doi.org/10,3390/app12189355. Disponível em: https://www.mdpi.com/2076-3417/12/18/9355. Acesso em: 17 jun. 2023.

MAHESHWARI, R.; TOSHNIWAL, A.; DUBEY, A. Software as a Service architecture and its security issues: a review. *In:* INTERNATIONAL CONFERENCE ON INVENTIVE SYSTEMS AND CONTROL (ICISC), 4., 2020, Coimbatore. **Proceedings**... Coimbatore: IEEE, 2020, p. 766-770, DOI: https://doi.org/10,1109/ICISC47916.2020,9171145. Disponível em: https://ieeexplore.ieee.org/document/9171145. Acesso em: 8 mar. 2024.

MCKINNEY, W. Data Structures for Statistical Computing in Python. *In*: PYTHON IN SCIENCE CONFERENCE, 9., 2010, Austin. **Proceedings**... Austin: SciPy, 2010, p. 51-56. DOI: https://doi.org/10,25080/MAJORA-92BF1922-00A. Disponível em: http://conference.scipy.org.s3-website-us-east-1.amazonaws.com/proceedings/scipy2010/mckinney.html. Acesso em: 8 mar. 2024.

MENARD, S. **Random over-sampling examples**. R package version 0,0-3, 2020. Disponível em: https://cran.r-project.org/package=ROSE. Acesso em: 19 jun. 2024.

MEYER, D. *et al.* **Misc functions of the department of statistics**. R package version 1.7-4, 2019. Disponível em: https://cran.r-project.org/package=e1071. Acesso em: 19 jun. 2024.

MILBORROW, S. **Plot 'rpart' models**. R package version 3.1.0, 2020, Disponível em: https://cran.r-project.org/package=rpart.plot. Acesso em: 19 jun. 2024.

MILLER JR, R. G. **Beyond ANOVA**: basics of applied statistics. Cidade de Nova Iorque: Routledge, 1997.

MIRKOVIĆ, M. *et al.* Customer Churn Prediction in B2B Non-Contractual Business Settings Using Invoice Data. **Applied Sciences**, Basel, v. 12, n. 10, p. 1-18, 2022. DOI: https://dx.doi.org/10,3390/app12105001. Disponível em: https://www.mdpi.com/2076-3417/12/10/5001 Acesso em: 2 jul. 2023.

MONTGOMERY, D. C. **Design and analysis of experiments**. New Jersey: John Wiley & Sons, 2017.

MÜLLER, A. C.; GUIDO, S. **Introduction to machine learning with Python**: a guide for data scientists. Sebastopol: O'Reilly Media, 2016.

NAIDU, G.; ZUVA, T.; SIBANDA, E. M. Systematic Review on Churn Prediction Systems in Telecommunications. *In*: INTERNATIONAL CONFERENCE ON COMMUNICATION, COMPUTING AND ELECTRONICS SYSTEMS, 3., 2022, cidade. **Proceedings**... Singapore: Lecture Notes in Electrical Engineering, v. 844, 2022, p. 983-995. DOI: https://doi.org/10.1007/978-981-16-8862-1_64. Disponível em: https://link.springer.com/article/10.1007/s00170-014-6626-6. Acesso em: 23 mar. 2024.

OKOLI, C. *et al.* Guia para realizar uma revisão sistemática da literatura. **EaD em Foco**, v. 9, n. 1, p. 1-40, 2019. DOI: https://doi.org/10.18264/eadf.v9i1.748. Disponível em: https://eademfoco.cecierj.edu.br/index.php/Revista/article/view/748. Acesso em: 10 mar. 2024.

PACANA, A.; ULEWICZ, R. Analysis of causes and effects of implementation of the quality management system compliant with ISO 9001. **Polish Journal of Management Studies**, v. 21, n. 1, p. 283-296, 2020. DOI: https://doi.org/10.17512/pjms.2020.21.1.21. Disponível em: https://pjms.zim.pcz.pl/resources/html/article/details?id=206243. Acesso em: 10 mar. 2024.

PAYNE, A.; FROW, P. A strategic framework for customer relationship management. **Journal of Marketing**, v. 69, n. 4, p. 167-176, 2005. Disponível em: https://journals.sagepub.com/doi/10,1509/jmkg.2005.69.4.167. Acesso em: 8 mar. 2024.

PEDREGOSA, F. *et al.* Scikit-learn: machine learning in Python. **JMLR**: Journal of Machine Learning Research, v. 12, n. 85, p. 2.825-2.830, 2011. Disponível em: https://www.jmlr.org/papers/v12/pedregosa11a.html. Acesso em: 19 mar. 2024.

PITUCH, K. A.; STEVENS, J. P. **Applied multivariate statistics for the Social Sciences**. Cidade de Nova Iorque: Routledge, 2012.

QUINLAN, J.R. Induction of decision trees. **Machine learning**, v. 1, n. 1, p. 81-106, 1986. DOI: https://doi.org/10,1007/BF00116251. Disponível em: https://link.springer.com/article/10,1007/BF00116251. Acesso em: 19 mar. 2024.

REICHHELD, F.F. **A estratégia da lealdade**: a força invisível que mantém clientes e funcionários e que sustenta crescimento, lucros e valor. Rio de Janeiro: Campus, 1996.

RIVERA, J. *et al.* Challenges in customer retention for SaaS companies. **Journal of Business & Industrial Marketing**, v. 35, n. 3, p. 455-466, 2020, Disponível em: https://www.emerald.com/insight/content/doi/10,1108/JBIM-09-2019-0357/full/html. Acesso em: 20 jun. 2024.

ROBERTS, B.; WALTER, G. **Display and analyze ROC curves**. R package version 1.17.0,1, 2021. Disponível em: https://cran.r-project.org/package=pROC. Acesso em: 15 jun. 2023.

SANA, J. K. *et al.* A novel customer churn prediction model for the telecommunication industry using data transformation methods and feature selection. **Plos One**, v. 17, n. 12, p. 1-21, 2022. DOI: https://dx.doi.org/10,1371/journal.pone.0278095. Disponível em: https://journals.plos.org/plosone/article?id=10,1371/journal.pone.0278095. Acesso em: 2 jul. 2023.

SHEARER, C. The CRISP-DM model: the new blueprint for data mining. **Journal of Data Warehousing**, v. 5, n. 4, p. 13-22, 2000, Disponível em: https://mineracaodedados.wordpress.com/wp-content/uploads/2012/04/the-crisp-dm-model-the-new-blueprint-for-data-mining-shearer-colin.pdf. Acesso em: 12 jul. 2023.

SINGH, H.; SAMALIA, H. V. A business intelligence perspective for churn management. **Procedia**: Social and Behavioral Sciences, v. 109, n. 1, p. 51-56, 2014. DOI: https://doi.org/10.1016/j.sbspro.2013.12.420. Disponível em: https://www.sciencedirect.com/science/article/pii/S1877042813050520. Acesso em: 23 maio 2024.

SIVASANKAR, E., VIJAYA, J. Hybrid PPFCM-ANN model: an efficient system for customer churn prediction through probabilistic possibilistic fuzzy clustering and artificial neural network. **Neural Computer & Application**, v. 31, p. 7.181-7.200, 2018. DOI: https://doi.org/10,1007/s00521-018-3548-4. Disponível em: https://link.springer.com/article/10.1007/s00521-018-3548-4. Acesso em: 17 de jun. 2023.

STROUSE, K. G. **Marketing telecommunications services**: new approaches for changing environment. Norwood: Artech House, 1999.

SUKOW, A. E. R.; GRANT, R. Forecasting and the role of churn in Software-as-a-Service business models. **iBusiness**, v. 5, n. 1, p. 49-57, 2013. DOI: https://doi.org/10.4236/ib.2013.51A006. Disponível em: https://www.scirp.org/journal/paperinformation?paperid=29333. Acesso em: 23 maio 2024.

TANENBAUM, A. S.; WOODHULL, A. S. **Sistemas operacionais**: projeto e implementação. Porto Alegre: Bookman, 2006.

THERNEAU, T.; ATKINSON, B. **Recursive partitioning and regression trees**. R package version 4.1-15, 2019. Disponível em: https://cran.r-project.org/package=rpart. Acesso em: 19 mar. 2024.

TUKEY, J. W. **Exploratory data analysis**. Cidade de Nova Iorque: Pearson, 1977.

UMAYAPARVATHI, V.; IYAKUTTI, K. Applications of Data Mining Techniques in Telecom Churn Prediction. **International Journal of Computer Applications**, v. 42, n. 1, p. 5-9, 2012. DOI: https://doi.org/10,5120/5814-8122. Disponível em: https://www.semanticscholar.org/paper/Applications-of-Data-Mining-Techniques-in-Umayaparvathi-Iyakutti/4f96e06db144823d16516af787e96d13073b4316. Acesso em: 17 jun. 2023.

VAN DER WALT, S.; COLBERT, S. C.; VAROQUAUX, G. The NumPy array: a structure for efficient numerical computation. **Computing in Science & Engineering**, v. 13, n. 2, p. 22-30, 2011. DOI: https://doi.org/10,1109/MCSE.2011.37. Disponível em: https://ieeexplore.ieee.org/document/5725236. Acesso em: 11 abr. 2024.

VENABLES, W. N.; RIPLEY, B. D. **Modern applied statistics with S**. Cidade de Nova Iorque: Springer, 2002.

VERBRAKEN, T.; VERBEKE, W.; BAESSENS, B. Profit optimizing customer churn prediction with Bayesian network classifiers. **Intelligent Data Analysis**, v. 18, n. 1, p. 3-24, 2014. DOI: https://doi.org/10,3233/IDA-130625. Disponível em: https://www.semanticscholar.org/paper/Profit-optimizing-customer-churn-prediction-with-Verbraken-Verbeke/b8551945584abcdc2610f30dc493714d256417e3. Acesso em: 11 abr. 2024.

WASKOM, M. *et al.* Seaborn: statistical data visualization. **JOSS:** Journal of Open Source Software, v. 6, n. 60, p. 1-4, 2021. DOI: https://doi.org/10,21105/joss.03021. Disponível em: https://joss.theoj.org/papers/10,21105/joss.03021. Acesso em: 21 mar. 2024.

WICKHAM, H. *et al.* Welcome to the tidyverse. **JOSS:** Journal of Open Source Software, v. 4, n. 43, p. 1686, 2019. DOI: https://doi.org/10,21105/joss.01686. Disponível em: https://joss.theoj.org/papers/10,21105/joss.01686. Acesso em: 19 mar. 2024.

WICKHAM, H. **ggplot2**: elegant graphics for data analysis. Cidade de Nova Iorque: Springer, 2016.

WIRTH, R.; HIPP, J. CRISP-DM: towards a standard process model for data mining. *In*: INTERNATIONAL CONFERENCE ON THE PRACTICAL APPLICATION OF KNOWLEDGE DISCOVERY AND DATA MINING, 4., 2000, Manchester. **Proceedings**… Manchester: 2020, p. 29-40, Disponível em: https://www.semanticscholar.org/paper/CRISP-DM%3A-Towards-a-Standard-Process-Model-for-Data-Wirth-Hipp/48b9293cfd4297f855867ca278f7069abc6a9c24. Acesso em: 21 mar. 2024.

WOLPERT, D. H. The lack of a priori distinctions between learning algorithms. **Neural Computation**, v. 8, n. 7, p. 1.341-1.390, 1996. DOI: https://doi.org/10.1162/neco.1996.8.7.1341. Disponível em: https://direct.mit.edu/neco/article-abstract/8/7/1341/6016/The-Lack-of-A-Priori-Distinctions-Between-Learning. Acesso em: 21 mar. 2024.

WRANG, E.; LEE, A. **Compact and flexible summaries of data**. R package version 2.1.5, 2021. Disponível em: https://cran.r-project.org/package=skimr. Acesso em: 5 abr. 2024.

XIE, Y. *et al.* Customer churn prediction using improved balanced random forests. **Expert Systems with Applications**, v. 36, n. 3, p. 5.445-5.449, 2009. DOI: https://doi.org/10.1016/j.eswa.2008.06.121. Disponível em: https://www.sciencedirect.com/science/article/abs/pii/S0957417408004326. Acesso em: 11 abr. 2024.

ZEITHAML, V. A.; BITNER, M. J.; GREMLER, D. D. **Services marketing**: integrating customer focus across the firm. Cidade de Nova Iorque: McGraw Hill, 2018.

ZHU, B. *et al.* Benchmarking sampling techniques for imbalance learning in churn prediction. **Journal of the Operational Research Society**, v. 69, n. 1, p. 49-65, 2017. DOI: https://doi.org/10,1057/s41274-016-0176-1. Disponível em: https://www.tandfonline.com/doi/full/10,1057/s41274-016-0176-1. Acesso em: 17 jun. 2023.

APÊNDICE 1 – AMOSTRA

Busca	AdicionarFabrica	Adicionar Produto	AdicionarFornecedor	Criarsolicitacao	Editarinformacoes	ArquivarSolicitacao	UsuariosUnicos	NumerodeLogins	SessaoMediaemminutos	CancelouServico	TipodeConta
22594	25	0	10	853	3566	180	50	456	130	0	1
6451	7	0	7	27	206	87	20	289	470	0	1
6024	5	2	0	84	11	21	8	147	89	0	1
4945	17	3	4	343	19	38	12	132	69	3	0
1914	0	0	25	159	5	80	4	108	55	0	1
1620	13	11	8	385	226	10	14	105	91	3	0
1003	38	0	0	41	58	4	2	47	45	3	0
966	13	0	13	20	91	5	4	36	153	3	0
766	0	0	5	120	12	12	6	39	11	3	0
113	0	0	0	18	2	0	6	29	16	0	1
112	0	37	11	13	3	1	14	34	47	3	0
60	0	0	0	0	0	0	10	8	1409	3	0
37	2	1	1	1	1	6	12	13	255	3	0
30	1	0	4	5	2	0	4	6	345	3	0
14	0	0	0	0	0	0	12	7	124	3	0
10	0	0	0	0	0	0	2	4	94	3	0
8	0	0	5	6	0	0	2	5	356	1	1
3	0	0	1	0	0	0	2	1	1365	1	1
2	0	0	2	3	3	0	8	6	1027	0	1
2	0	0	0	0	0	0	8	4	622	0	1
1	0	0	0	0	0	0	8	5	2547	3	1
1	0	0	0	0	0	0	2	2	415	0	1
0	0	0	1	1	0	0	10	6	712	3	0
0	0	1	1	2	0	0	8	6	88	3	0

0	0	0	1	1	0	0	8	4	627	3	0
0	0	1	1	1	0	0	8	4	1107	3	0
0	0	0	1	1	0	0	2	1	651	3	0
0	0	0	1	1	0	0	2	1	481	3	0

APÊNDICE 2 – R: BIBLIOTECAS NECESSÁRIAS

```
library(caret)
library(tidyverse)
library(nnet)  # Para redes neurais
library(e1071) # Para SVM
library(glmnet) # Para regressão logística com regularização
library(pROC) # Para calcular AUC
library(ROSE) # Para balanceamento de classes
library(rpart) # Para árvore de decisão
library(rpart.plot) # Para plotar árvore de decisão
library(ggpubr)
library(skimr)
```

APÊNDICE 3 – R: PREPARAÇÃO DE DADOS

```
dados <- read.csv("C:/Users/andre/Downloads/TCC André - Sheet2 (10).csv")
dados <- na.omit(dados)
dados$CancelouServico <- as.factor(dados$CancelouServico)
levels(dados$CancelouServico) <- make.names(levels(dados$CancelouServico))
```

APÊNDICE 4 – R: DISTRIBUIÇÃO DAS CLASSES

```
print(table(dados$CancelouServico))
```

APÊNDICE 5 – R: REMOÇÃO DE COLUNAS COM VARIÂNCIA ZERO

```
dados <- dados[, sapply(dados, function(x) length(unique(x)) > 1)]
```

APÊNDICE 6 – R: DIVISÃO DOS DADOS EM TREINAMENTO E TESTE

```
set.seed(123)
index <- createDataPartition(dados$CancelouServico, p = 0,8, list = FALSE)
trainData <- dados[index,]
testData <- dados[-index,]
```

APÊNDICE 7 – R: BALANCEAMENTO DAS CLASSES NO CONJUNTO DE TREINAMENTO

```
trainData <- upSample(x = trainData[, -ncol(trainData)], y =
trainData$CancelouServico)
```

APÊNDICE 8 – R: DISTRIBUIÇÃO DAS CLASSES APÓS BALANCEAMENTO

```
print(table(trainData$Class))
```

APÊNDICE 9 – R: CONTROLE DE TREINAMENTO

```
fitControl <- trainControl(method = "cv", number = 10, classProbs = TRUE,
summaryFunction = twoClassSummary, savePredictions = "final")
```

APÊNDICE 10 – R: MODELOS PARA TREINAR

```
modelos <- list(
  logreg = train(Class ~ ., data = trainData, method = "glmnet", trControl = fitControl,
metric = "ROC"),
  cart = train(Class ~ ., data = trainData, method = "rpart", trControl = fitControl,
metric = "ROC"),
  rf = train(Class ~ ., data = trainData, method = "rf", trControl = fitControl, metric =
"ROC"),
  svm = train(Class ~ ., data = trainData, method = "svmRadial", trControl = fitControl,
metric = "ROC"),
  nnet = train(Class ~ ., data = trainData, method = "nnet", trControl = fitControl, linout
= FALSE, trace = FALSE, maxit = 1000, metric = "ROC")
)
```

APÊNDICE 11 – R: AVALIAÇÃO DOS MODELOS

```
resultados <- lapply(modelos, function(model) {
  predictions <- predict(model, testData)
  prob <- predict(model, testData, type = "prob")
  cm <- confusionMatrix(predictions, testData$CancelouServico)
  auc <- roc(testData$CancelouServico, prob[,2], plot = FALSE)$auc
  return(list(model = model, accuracy = cm$overall['Accuracy'], auc = auc, cm =
cm$table))
})
```

APÊNDICE 12 – R: EXIBIR RESULTADOS

```
print(resultados)
```

APÊNDICE 13 – R: REPRESENTAÇÃO GRÁFICA DOS RESULTADOS

```
resultados_df <- data.frame(
  Model = names(resultados),
  Accuracy = sapply(resultados, function(res) res$accuracy),
  AUC = sapply(resultados, function(res) res$auc)
)

ggplot(resultados_df, aes(x = Model, y = Accuracy)) +
  geom_bar(stat = "identity", fill = "blue") +
  geom_text(aes(label = round(Accuracy, 2)), vjust = -0,5) +
  ggtitle("Model Accuracy Comparison") +
  ylim(0, 1)

ggplot(resultados_df, aes(x = Model, y = AUC)) +
  geom_bar(stat = "identity", fill = "green") +
  geom_text(aes(label = round(AUC, 2)), vjust = -0,5) +
  ggtitle("Model AUC Comparison") +
  ylim(0, 1)
```

APÊNDICE 14 – R: ÁRVORE DE DECISÃO

```
rpart_model <- modelos$cart$model
rpart.plot(rpart_model)
```

APÊNDICE 15 – R: ANÁLISE DE CORRELAÇÃO

```
correlacoes <- cor(dados[, sapply(dados, is.numeric)])
print(correlacoes)
```

APÊNDICE 16 – PYTHON: BIBLIOTECAS NECESSÁRIAS

```
import numpy as np
import pandas as pd
from sklearn.model_selection import train_test_split, cross_val_score, GridSearchCV
from sklearn.ensemble import RandomForestClassifier
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
from sklearn.neural_network import MLPClassifier
from sklearn.metrics import accuracy_score, roc_auc_score, roc_curve,
precision_recall_fscore_support, confusion_matrix, classification_report
from sklearn.feature_selection import SelectKBest, f_classif, RFE
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.preprocessing import LabelBinarizer
from sklearn.metrics import precision_recall_curve
```

APÊNDICE 17 – PYTHON: PREPARAÇÃO DOS DADOS

```
# Carregar dados
df = pd.read_csv('caminho/para/seu/arquivo.csv')  # Substitua pelo caminho do seu arquivo

# Separar características e alvo
X = df.drop('CancelouServico', axis=1)  # Características
y = df['CancelouServico']  # Alvo

# Dividir os dados em conjunto de treino e teste
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0,2, random_state=42)
```

APÊNDICE 18 – PYTHON: DEFINIÇÃO E AVALIAÇÃO DOS MODELOS

```
# Modelos
models = {
    'Logistic Regression': LogisticRegression(max_iter=1000),
    'Random Forest': RandomForestClassifier(),
    'SVM': SVC(probability=True),
    'Neural Network': MLPClassifier(max_iter=1000)
}

# Validação cruzada
results = {}
for model_name, model in models.items():
    scores = cross_val_score(model, X, y, cv=10)
    results[model_name] = {'Acurácia Média': scores.mean(), 'Desvio Padrão':
scores.std()}

# Exibir resultados
results_df = pd.DataFrame(results).T
print(results_df)
```

APÊNDICE 19 – PYTHON: AJUSTE DE HIPERPARÂMETROS

```
# Ajuste de hiperparâmetros (Exemplo com Random Forest)
param_grid_simplified = {
    'n_estimators': [50, 100],
    'max_features': ['sqrt', 'log2'],
    'max_depth': [10, None],
    'min_samples_split': [2, 5],
    'min_samples_leaf': [1, 2]
}

# GridSearchCV para Random Forest
grid_search_simplified = GridSearchCV(RandomForestClassifier(),
param_grid_simplified, cv=3)
grid_search_simplified.fit(X_train, y_train)

best_params_simplified = grid_search_simplified.best_params_

# Treinar modelo Random Forest com melhores hiperparâmetros
best_rf_simplified = grid_search_simplified.best_estimator_
best_rf_simplified.fit(X_train, y_train)

print(best_params_simplified)
```

APÊNDICE 20 – PYTHON: VALIDAÇÃO CRUZADA COM SMOTE

```
import matplotlib.pyplot as plt

# Modelos ajustados
models_adjusted = {
    'Logistic Regression': LogisticRegression(max_iter=2000),
    'Random Forest': RandomForestClassifier(),
    'SVM': SVC(probability=True),
    'Neural Network': MLPClassifier(max_iter=2000)
}

# Validação cruzada com reamostragem e escalagem dos dados
results_resampled = {}
for model_name, model in models_adjusted.items():
    scores = cross_val_score(model, X_resampled_scaled, y_resampled, cv=10)
    results_resampled[model_name] = {'Acurácia Média': scores.mean(), 'Desvio
Padrão': scores.std()}

results_resampled_df = pd.DataFrame(results_resampled).T

# Plotar resultados da validação cruzada
plt.figure(figsize=(10, 6))
plt.errorbar(results_resampled_df.index, results_resampled_df['Acurácia Média'],
yerr=results_resampled_df['Desvio Padrão'], fmt='o', capsize=5, capthick=2)
plt.title('Validação Cruzada com SMOTE')
plt.xlabel('Modelos')
plt.ylabel('Acurácia Média')
plt.grid(True)
plt.show()

results_resampled_df
```

APÊNDICE 21 – PYTHON: AVALIAÇÃO DAS VARIÁVEIS

```
# Importância das características com Random Forest
importances = best_rf_simplified.feature_importances_
features = X.columns
feature_importance_df = pd.DataFrame({'Característica': features, 'Importância':
importances})
feature_importance_df = feature_importance_df.sort_values(by='Importância',
ascending=False)

print(feature_importance_df)

# Seleção de características com ANOVA
selector = SelectKBest(f_classif, k='all')
selector.fit(X_train, y_train)
anova_scores = selector.scores_

anova_df = pd.DataFrame({'Característica': features, 'Pontuação': anova_scores})
anova_df = anova_df.sort_values(by='Pontuação', ascending=False)

print(anova_df)

# RFE com Regressão Logística
logreg = LogisticRegression(max_iter=1000)
rfe = RFE(logreg, n_features_to_select=1)
rfe.fit(X_train, y_train)
rfe_ranking = rfe.ranking_

rfe_df = pd.DataFrame({'Característica': features, 'Rank': rfe_ranking})
rfe_df = rfe_df.sort_values(by='Rank')

print(rfe_df)
```

APÊNDICE 22 – PYTHON: AVALIAÇÃO E COLETA DAS MÉTRICAS DOS MODELOS

```
# Avaliação de ROC e métricas adicionais
model_metrics = {}
for model_name, model in models.items():
    model.fit(X_train, y_train)
    y_pred = model.predict(X_test)
    y_pred_prob = model.predict_proba(X_test)[:, 1] if hasattr(model, "predict_proba")
else None

    accuracy = accuracy_score(y_test, y_pred)

    # Detectar classes presentes no conjunto de dados
    lb = LabelBinarizer()
    lb.fit(y)
    y_test_binarized = lb.transform(y_test)
    y_pred_binarized = lb.transform(y_pred)

    precision, recall, f1_score, _ = precision_recall_fscore_support(y_test_binarized,
y_pred_binarized, average='macro', zero_division=1)

    # Calcular AUC apenas se ambas as classes estiverem presentes
    if y_pred_prob is not None and len(np.unique(y_test)) == 2:
        auc = roc_auc_score(y_test_binarized, y_pred_prob, average='macro',
multi_class='ovr')
    else:
        auc = None

    model_metrics[model_name] = {
        'Acurácia': accuracy,
        'Precisão': precision,
        'Recall': recall,
        'F1-score': f1_score,
        'AUC': auc,
        'Matriz de Confusão': confusion_matrix(y_test, y_pred),
        'Relatório de Classificação': classification_report(y_test, y_pred,
zero_division=1)
    }

print(pd.DataFrame(model_metrics).T)
```

APÊNDICE 23 – PYTHON: GRÁFICO DE CURVAS DE PRECISÃO-RECALL

```
# Transformar os rótulos de classe para serem binários (0 e 1)
y_test_binary = (y_test == 2).astype(int)

# Função para plotar a curva de precisão-recall
def plot_precision_recall_curve(y_test_binary, y_pred_prob, model_name):
    precision, recall, _ = precision_recall_curve(y_test_binary, y_pred_prob)
    plt.plot(recall, precision, marker='.', label=model_name)

# Inicializar a plotagem
plt.figure(figsize=(10, 8))

# Plotar curvas de precisão-recall para cada modelo
for model_name, model in models.items():
    model.fit(X_train, y_train)
    if hasattr(model, "predict_proba"):
        y_pred_prob = model.predict_proba(X_test)[:, 1]
        plot_precision_recall_curve(y_test_binary, y_pred_prob, model_name)

# Configurar a plotagem
plt.xlabel('Recall')
plt.ylabel('Precision')
plt.title('Precision-Recall Curve')
plt.legend()
plt.grid(True)
plt.show()
```