http://journal.unoeste.br/index.php/ce/index
DOI: 10.5747/ce.2026.v18.e443
ISSN on-line 2178-8332

Colloquium Exactarum

Submetido: 30/06/2025 Revisado: 14/11/2025 Aceito: 20/03/2026

PREVENÇÃO DE CHURN: COMPARAÇÃO DE MODELOS PREDITIVOS

Churn Prevention: Comparison of Predictive Models

Daniel Figueirinha Silveira[1]; Helton Molina Sapia[1]

[1]Universidade do Oeste Paulista – Unoeste, Faculdade de Informática de Presidente Prudente – FIPP, SP, Brasil
E-mail: d.f.silveira@unoeste.edu.br, helton@unoeste.br

RESUMO – A evasão de clientes (*churn*) compromete a receita recorrente e a sustentabilidade de operadoras de telecomunicações. Este artigo compara três algoritmos supervisionados — *K-Nearest Neighbors* (KNN), *Decision Tree* e *Multilayer Perceptron* (MLP) — na previsão de *churn* usando o *Iranian Churn Dataset* (3.150 registros; 15,7% de *churns*), um cenário típico de classes desbalanceadas. O pipeline inclui checagem de duplicidades e de valores ausentes, transformação de variáveis, normalização (*StandardScaler*), divisão estratificada em treino e avaliação (75/25) e correção do desbalanceamento no treino com *SMOTE*. Para robustez, aplicou-se *Grid Search* com *cross-validation* 5-fold, otimizando *F1-score* e reportando acurácia, precisão, *recall* e *AUC-ROC*. O MLP apresentou o melhor equilíbrio entre detecção e precisão (*F1-score* = 0,8402; *AUC-ROC* = 0,9862), seguido por KNN (0,8033) e Árvore de Decisão (0,8000). As análises confirmam a superioridade do MLP e, na prática, indicam que o modelo identifica mais clientes em risco com menos alarmes falsos, favorecendo ações de retenção mais eficientes. Conclui-se que modelos de aprendizado de máquina, especialmente o MLP, apoiam ações preventivas e a otimização de recursos em telecomunicações.
Palavras-chave: *churn*; aprendizado de máquina; telecomunicações; MLP; KNN; árvore de decisão.

ABSTRACT – Customer churn undermines recurring revenue and sustainability in telecommunications. This paper compares three supervised algorithms — K-Nearest Neighbors (KNN), Decision Tree, and Multilayer Perceptron (MLP) — for churn prediction using the Iranian Churn Dataset (3,150 records; 15.7% churn), a typical imbalanced-class scenario. The pipeline included duplicate and missing-value checks, variable transformation, numerical standardization (StandardScaler), a stratified 75/25 train–evaluation split, and training imbalance correction with SMOTE. For robustness, we applied Grid Search with 5-fold cross-validation, optimizing F1-score and reporting accuracy, precision, recall, and AUC-ROC. MLP achieved the best balance between detection and precision (F1-score = 0.8402; AUC-ROC = 0.9862), followed by KNN (0.8033) and Decision Tree (0.8000). The analyses confirm MLP's superiority and, in practice, indicate that it identifies more at-risk customers with fewer false alerts, enabling more efficient retention actions. We conclude that machine learning models, especially MLP, support proactive retention and resource optimization in telecommunications.
Keywords: churn; machine learning; telecommunications; MLP; KNN; decision tree.

1. INTRODUÇÃO

A evasão de clientes, conhecida como *churn*, representa uma ameaça significativa à sustentabilidade financeira das empresas de telecomunicações, impactando diretamente nas receitas, estabilidade da base de clientes e custos de aquisição (Geiler; Affeldt; Nadif, 2022). Em mercados competitivos, o custo para conquistar um novo cliente pode ser até seis vezes maior do que o de reter um já existente, tornando a retenção uma prioridade estratégica (Barsotti *et al.*, 2024). Pequenas elevações nas taxas de *churn* podem comprometer o crescimento e a lucratividade das operadoras, sendo a retenção de clientes um fator crítico em negócios baseados em assinatura (Lalwani *et al.*, 2022).

Antecipar o risco de evasão é essencial para evitar perdas de receita e direcionar ações de retenção eficazes (Manzoor *et al.*, 2024). No entanto, prever *churn* com precisão envolve desafios técnicos e conceituais: bases de dados frequentemente desbalanceadas, múltiplos fatores de influência (como tempo de assinatura, reclamações, engajamento e valor de consumo), além da necessidade de modelos interpretáveis e métricas adequadas (Wang *et al.*, 2016; Ahn *et al.*, 2020). A acurácia isolada, nestes contextos, pode ser enganosa, já que modelos tendem a privilegiar a classe majoritária. Por isso, métricas como F1-score são recomendadas para avaliação do desempenho preditivo em cenários reais de *churn* (Tang; Xia, 2020).

Nos últimos anos, técnicas de aprendizado de máquina têm se destacado como ferramentas eficazes para predição de *churn*, permitindo a identificação antecipada de clientes com maior risco de cancelamento (Ahn *et al.*, 2020; Barsotti *et al.*, 2024). Algoritmos como *K-Nearest Neighbors* (KNN), Árvores de Decisão e *Multilayer Perceptron* (MLP) são amplamente estudados por sua capacidade de capturar padrões complexos e apoiar estratégias de retenção de clientes (Mouli *et al.*, 2024). O KNN é valorizado por sua simplicidade e clareza interpretativa; as Árvores de Decisão se destacam pela explicabilidade; e o MLP, por sua capacidade de modelar relações não lineares (Przybyła-Kasperek; Marfo; Sulikowski, 2024). Entretanto, não existe um modelo universalmente superior: cada abordagem apresenta vantagens e limitações que dependem fortemente das características dos dados e do contexto de aplicação (Geiler; Affeldt; Nadif, 2022). Diante desse contexto, este artigo compara o desempenho dos algoritmos KNN, Árvore de Decisão e MLP na predição de *churn* em telecomunicações, utilizando o *Iranian Churn Dataset*. O estudo adota um pipeline robusto de pré-processamento, balanceamento de classes e otimização de hiperparâmetros, avaliando os modelos por métricas adequadas a dados desbalanceados; os resultados visam apoiar a escolha de métodos eficazes para retenção de clientes, contribuindo para a sustentabilidade e a competitividade do setor.

As demais seções deste trabalho estão organizadas da seguinte forma: a Seção 2 apresenta a revisão da literatura (fundamentos, algoritmos, desbalanceamento, métricas e trabalhos relacionados). A Seção 3 descreve a metodologia (ambiente computacional, base de dados, pré-processamento e treinamento/otimização). A Seção 4 traz os experimentos e resultados (tabelas, figuras e análises). Por fim, a Seção 5 apresenta as considerações finais e sugestões de trabalhos futuros.

2. REVISÃO DA LITERATURA

2.1 Conceito e impacto do churn

O fenômeno de evasão de clientes, conhecido como *churn*, é particularmente crítico em serviços recorrentes e modelos por assinatura, pois afeta diretamente a receita, o crescimento e a sustentabilidade das organizações (Geiler; Affeldt; Nadif, 2022; Lalwani *et al.*, 2022). Em ambientes competitivos, a retenção costuma ser mais eficiente que a aquisição de novos clientes, com impactos positivos no *Customer Lifetime Value* (CLV) (Bojei *et al.*, 2013; Barsotti *et al.*, 2024). Para além do aspecto econômico, a evasão é multidimensional e também reflete experiência e percepção de valor do usuário (Soliman; Rinta-Kahila, 2020). Nesse contexto, antecipar o risco por meio de modelos preditivos permite intervenções proativas de retenção e melhor alocação de recursos (Ahn *et al.*, 2020).

2.2 Predição de churn

A literatura especializada em telecomunicações converge para um pipeline que combina pré-processamento, tratamento de desbalanceamento, padronização quando pertinente, particionamento estratificado em treino e avaliação, e otimização de hiperparâmetros com validação robusta (Barsotti *et al.*, 2024; Chang *et al.*, 2024). Dois pontos são recorrentes: o desbalanceamento entre ***churn***s (minoria) e não *churns* (maioria), que pode enviesar classificadores se não tratado (Van Hulse *et al.*, 2009; Wang *et al.*,

2016), e a necessidade de métricas sensíveis à classe positiva, pois acurácia isolada pode ser enganosa em eventos raros (King; Zeng, 2001; Chang *et al.*, 2024). Em linha com essas recomendações, este estudo adota divisão estratificada 75/25, aplica *SMOTE* exclusivamente ao conjunto de treino, padroniza variáveis numéricas com *StandardScaler* (relevante para métodos baseados em distância) e realiza ajuste via *Grid Search* com *cross-validation* 5-fold, otimizando *F1-score*.

2.3 Algoritmos analisados neste estudo

2.3.1 *K-Nearest Neighbors* (KNN)

O *K-Nearest Neighbors* (KNN) classifica por proximidade em um espaço métrico e não exige treinamento paramétrico. Sua simplicidade e transparência como *baseline* são vantagens; contudo, é sensível à escala dos atributos, à dimensionalidade e ao desbalanceamento (Dubey; Pudi, 2013; Van Hulse *et al.*, 2009). Pesquisas comparativas em *churn* mostram que, em dados reais e desbalanceados, o KNN tende a perder precisão, embora possa alcançar bons níveis de *recall* dependendo do k, da métrica e da ponderação dos vizinhos (Hassonah *et al.*, 2019; Dubey; Pudi, 2013). A padronização (p. ex., *StandardScaler*) e a escolha adequada de k e da métrica (Manhattan/Euclidiana/Chebyshev) são determinantes para o desempenho.

2.3.2 Árvore de Decisão

Árvores de Decisão constroem regras hierárquicas explicáveis e não requerem padronização de escala. Para evitar sobreajuste, controlam-se profundidade, tamanho mínimo de folhas e critérios de divisão (Gini/entropia) (Lalwani *et al.*, 2022). No domínio de telecomunicações, árvores são valorizadas pela interpretabilidade e por oferecerem importâncias de variáveis (redução de impureza), útil para orientar ações de negócio (Chang *et al.*, 2024). Trabalhos empíricos indicam desempenho competitivo e, por vezes, superior ao KNN em bases reais (Hassonah *et al.*, 2019), especialmente quando os atributos possuem relações não lineares simples e interações moderadas. Em paralelo, a literatura registra o uso da terminologia em inglês como *Decision Tree* para referência técnica.

2.3.3 *Multilayer Perceptron* (MLP)

O *Multilayer Perceptron* (MLP) modela relações não lineares e interações complexas, com ganhos relevantes quando há ajuste cuidadoso de arquitetura, regularização e critérios de parada (Przybyła-Kasperek; Marfo; Sulikowski, 2024). Em bases desbalanceadas, sem tratamento, redes tendem a privilegiar a classe majoritária; por isso, o uso de reamostragem e validação robusta é essencial (Wang *et al.*, 2016). Evidências recentes em telecom mostram MLP superando métodos tradicionais quando bem configurado, embora com menor interpretabilidade (Przybyła-Kasperek; Marfo; Sulikowski, 2024; Saha *et al.*, 2024). A escolha de funções de ativação, camadas/neurônios e regularização (p. ex., alpha) impacta diretamente o desempenho.

2.4 Desbalanceamento de classes e *SMOTE*

Bases reais de *churn* são tipicamente desbalanceadas, como no Iranian *Churn Dataset*, e, sem tratamento, o ajuste dos modelos minimiza o erro global às custas da sensibilidade à classe minoritária (Van Hulse *et al.*, 2009). A literatura recomenda intervenções no conjunto de treino, em particular a superamostragem via *SMOTE*, que gera exemplos sintéticos da minoria e costuma elevar *recall* e *F1-score* quando aplicada sem vazamento ao conjunto de avaliação; alternativas incluem subamostragem e ponderação de classes, especialmente útil em redes neurais (Wang *et al.*, 2016). Nesta pesquisa, adotou-se *SMOTE* exclusivamente no treino, com divisão estratificada e avaliação preservada para garantir mensuração realista.

2.5 Métricas para avaliação em classe desbalanceada

Em cenários com classe positiva rara, como a predição de *churn*, a acurácia isolada pode ocultar limitações relevantes do modelo, pois classificadores tendem a privilegiar a classe majoritária e ainda assim registrar altas taxas globais de acerto (King; Zeng, 2001; Wang *et al.*, 2016). Diante disso, a literatura recomenda o uso de métricas sensíveis à classe positiva para uma avaliação mais informativa (Chang *et al.*, 2024). As principais métricas adotadas em problemas de classificação desbalanceada são:

- Acurácia (*Accuracy*): proporção global de instâncias corretamente classificadas em relação ao total de exemplos. Embora intuitiva, pode ser enganosa em bases desbalanceadas, já que um modelo que ignora completamente a classe minoritária pode alcançar altos valores desta métrica.
- Precisão (*Precision*): proporção de instâncias previstas como *churn* que são efetivamente *churns* reais. Reflete a capacidade do modelo de evitar falsos positivos, ou seja, alarmes de evasão disparados erroneamente para clientes que não cancelariam.
- Sensibilidade (*Recall*): proporção de *churns* reais que foram corretamente identificados pelo modelo dentre todos os casos de evasão existentes. Métrica crítica em contextos onde o custo de um *churn* não detectado (falso negativo) é elevado.
- F1-score: média harmônica entre Precisão e *Recall*, expressa pela fórmula F1 = 2 × (Precisão × *Recall*) / (Precisão + *Recall*). Por penalizar igualmente desequilíbrios em qualquer uma das duas componentes, é amplamente recomendada como métrica principal em problemas de classes desbalanceadas (Chang *et al.*, 2024; Tang; Xia, 2020).
- AUC-ROC (*Area Under the ROC Curve*): resume a capacidade de separação do classificador entre as classes ao longo de diferentes limiares de decisão. Valores próximos a 1 indicam excelente discriminação; valor 0,5 equivale a um classificador aleatório. Em dados desbalanceados, as curvas *Precision-Recall* e a *Average Precision* (AP) complementam a análise ao concentrar a avaliação na classe positiva (King; Zeng, 2001; Chang *et al.*, 2024).

A escolha dessas métricas é orientada pela necessidade de avaliar adequadamente o desempenho dos modelos diante do desbalanceamento de classes, sendo o *F1-score* adotado como critério principal de comparação e otimização neste estudo, por refletir de forma equilibrada o compromisso entre identificar *churns* e evitar falsos alarmes.

2.6 Trabalhos relacionados em telecom (KNN, Árvores e MLP)

Levantamentos recentes documentam o uso consistente de árvores, regressão e redes neurais na predição de *churn* em telecomunicações, com crescente rigor em tratamento de desbalanceamento e validação (Barsotti *et al.*, 2024; Lalwani *et al.*, 2022). Comparativos empíricos mostram Árvores de Decisão superando o KNN em bases reais, evidenciando as limitações do classificador baseado em distância quando há desbalanceamento e maior dimensionalidade (Hassonah *et al.*, 2019; Dubey; Pudi, 2013; Van Hulse *et al.*, 2009). Em redes, resultados recentes apontam o MLP com bom equilíbrio entre acurácia e *recall* quando ajustado e regularizado de forma adequada (Przybyła-Kasperek; Marfo; Sulikowski, 2024; Saha *et al.*, 2024). No pré-processamento e avaliação, estudos em telecom confirmam ganhos consistentes com superamostragem no treino e uso de métricas sensíveis à classe positiva, como *F1-score* e *Precision-Recall* (Chang *et al.*, 2024).

2.7 Síntese crítica e justificativa das escolhas deste estudo

A literatura converge para a inexistência de um "modelo universalmente superior" para *churn*; a eficácia depende do domínio, dos dados e dos objetivos de negócio (Barsotti *et al.*, 2024; Lalwani *et al.*, 2022). Diante disso, a comparação entre três paradigmas complementares — distância (KNN), regras interpretáveis (Árvore de Decisão) e redes neurais (MLP) —, sob um pipeline robusto (padronização quando pertinente, *SMOTE* no treino, *Grid Search* com *cross-validation* 5-fold e otimização por *F1-score*), aplicada a um conjunto real e desbalanceado, fornece base metodológica sólida e alinhada às melhores práticas para avaliar o desempenho e os compromissos entre detecção e precisão, preservando interpretabilidade operacional via importância de variáveis da Árvore de Decisão.

3. METODOLOGIA

Este estudo foi conduzido a partir de uma abordagem quantitativa experimental, visando a comparação do desempenho de diferentes algoritmos de aprendizado supervisionado na predição de *churn* em telecomunicações. O desenvolvimento seguiu um pipeline completo, desde o pré-processamento dos dados até a avaliação dos resultados, com rigorosa padronização para garantir reprodutibilidade e imparcialidade.

3.1. Ambiente computacional e ferramentas

Os experimentos foram realizados em ambiente local, utilizando um MacBook Pro (chip Apple M2, 8GB de RAM), sistema operacional macOS, ambiente virtual Python 3.11 e controle de versões via Git. O código-fonte foi modularizado para facilitar a manutenção e reprodutibilidade.

As bibliotecas e ferramentas utilizadas foram:

- pandas e numpy: manipulação e análise de dados;
- scikit-learn: implementação de modelos, pré-processamento e métricas;
- imblearn: técnicas de balanceamento (*SMOTE*);
- matplotlib e seaborn: visualização de dados;
- joblib: serialização de modelos;
- os e time: operações do sistema e análise de desempenho.

3.2. Base de dados

Foi utilizado o *Iranian Churn Dataset*, disponível publicamente no repositório UCI Machine Learning Repository (DOI: https://doi.org/10.24432/C5JW3Z). O conjunto contempla 3.150 registros de clientes de uma operadora de telecomunicações iraniana, coletados durante 12 meses, com 13 atributos preditivos numéricos e um atributo alvo binário (*Churn*: 0 = não *churn*, 1 = *churn*). Não há valores ausentes no *dataset* e aproximadamente 15,7% dos clientes são *churns*, o que caracteriza um desbalanceamento típico para este problema. Os atributos (exceto o rótulo churn) são agregações dos primeiros 9 meses; os rótulos de churn refletem o estado ao final do mês 12, considerando uma janela de planejamento de 3 meses.

Atributos do *Iranian Churn Dataset*:

- Call Failure: número de falhas de chamadas realizadas pelo cliente;
- Subscription Length: tempo de assinatura do cliente;
- Charge Amount: valor total cobrado do cliente no período analisado;
- Seconds of Use: tempo total de uso dos serviços, em segundos;
- Frequency of use: frequência de utilização dos serviços;
- Frequency of SMS: frequência de envio de SMS pelo cliente;
- Distinct Called Numbers: quantidade de números distintos contatados;
- Customer Value: valor do cliente para a empresa, conforme critérios internos;
- Age: idade do cliente;
- Age Group: faixa etária do cliente, categorizada;
- Tariff Plan: forma de pagamento ou tipo de plano tarifário contratado;
- Status: status cadastral do cliente;
- Complains: presença de reclamações registradas pelo cliente.

3.3. Pré-processamento dos dados

Foram verificados e removidos registros duplicados e validada a ausência de valores ausentes. As variáveis categóricas “Tariff Plan” e “Status” foram mapeadas para valores binários, preservando a codificação ordinal das demais variáveis numéricas.

As variáveis numéricas foram normalizadas utilizando o *StandardScaler*, ajustado no conjunto de treino e aplicado no conjunto de avaliação. Os dados foram divididos de forma estratificada, com 75% para treino e 25% para avaliação, mantendo a proporção original de *churns* em ambos os conjuntos.

Considerando o desbalanceamento entre as classes (15,7% de *churns*), foi aplicada a técnica *SMOTE* (Synthetic Minority Over-sampling Technique) exclusivamente ao conjunto de treino, gerando amostras sintéticas da classe minoritária. O conjunto de avaliação foi mantido inalterado para garantir uma avaliação realista. Esta abordagem segue recomendações da literatura (Przybyła-Kasperek; Marfo; Sulikowski, 2024).

3.4. Treinamento e otimização dos modelos

Foram avaliados três algoritmos de aprendizado supervisionado: K-Nearest Neighbors (KNN), Árvore de Decisão e *Multilayer Perceptron* (MLP). Todos os modelos foram treinados e validados sob as mesmas condições de dados e particionamento, assegurando uma comparação justa e imparcial quanto ao desempenho.

A escolha dos hiperparâmetros a serem otimizados foi fundamentada em ampla revisão da literatura (Hassonah *et al.*, 2019; Przybyła-Kasperek; Marfo; Sulikowski, 2024; Tang; Xia, 2020), bem como em diretrizes das principais bibliotecas de machine learning (scikit-learn). Foram priorizados aqueles

parâmetros que, segundo estudos prévios, exercem maior impacto no desempenho preditivo e na capacidade de generalização dos modelos, especialmente em cenários de dados desbalanceados como o de *churn*.

A otimização dos hiperparâmetros foi realizada por meio de busca exaustiva em grade (*Grid Search*), combinada à validação cruzada estratificada (5-fold), utilizando o F1-score como métrica principal. Essa abordagem busca garantir que o ajuste dos parâmetros maximize o equilíbrio entre precisão e *recall*, superando limitações da acurácia simples em bases desbalanceadas.

3.4.1. K-Nearest Neighbors (KNN)

- n_neighbors: [3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23]
- weights: ['uniform', 'distance']
- metric: ['euclidean', 'manhattan', 'chebyshev']
- algorithm: ['auto', 'ball_tree', 'kd_tree', 'brute']

A escolha desses parâmetros se baseia em sua influência direta na suavização das decisões, sensibilidade a ruídos e desempenho computacional. Valores ímpares de n_neighbors impedem empates, enquanto diferentes métricas de distância e algoritmos de busca permitem avaliar o comportamento do modelo em diferentes contextos de dados (Dubey; Pudi, 2013).

3.4.2. Árvore de Decisão (*Decision Tree*)

- criterion: ['gini', 'entropy']
- max_depth: [None, 5, 10, 15, 20, 25, 50]
- min_samples_split: [2, 3, 4, 5, 10]
- class_weight: [None]
- min_samples_leaf: [1, 2, 4, 8]
- max_features: [None, 'sqrt', 'log2']

A seleção desses hiperparâmetros visa controlar a complexidade da árvore, prevenindo sobreajuste e promovendo generalização. Critérios de divisão, profundidade máxima, limites mínimos para splits e folhas, e variedade no número de atributos por divisão estão alinhados às recomendações de Lalwani *et al.* (2022).

3.4.3. *Multilayer Perceptron* (MLP)

- hidden_layer_sizes: [(8,), (13,), (8, 4), (13, 6), (8, 4, 2), (13, 6, 3), (50,), (100,), (50, 25), (100, 50), (50, 25, 12), (100, 50, 25)]
- activation: ['relu', 'tanh', 'logistic']
- alpha: [0.0001, 0.001, 0.01]
- learning_rate: ['constant', 'adaptive']
- max_iter: [2000]
- early_stopping: [True]

Com relação à configuração da MLP, explorou-se uma variedade de arquiteturas de camadas ocultas, funções de ativação e estratégias de regularização, conforme sugerido por Przybyła-Kasperek, Marfo e Sulikowski (2024), visando balancear a capacidade de modelar relações não lineares sem incorrer em sobreajuste. A utilização de *early stopping* e limiares para parada antecipada busca garantir treinamento eficiente e robusto.

4. EXPERIMENTOS E RESULTADOS

Os experimentos foram conduzidos utilizando o I*ranian Churn Dataset*, composto por 3.150 registros de clientes de telecomunicações, caracterizando um cenário típico de desbalanceamento: aproximadamente 15,7% de *churns* e 84,3% de clientes ativos. O pipeline experimental seguiu as etapas metodológicas descritas anteriormente, incluindo o pré-processamento com normalização via *StandardScaler*, balanceamento de classes pelo método *SMOTE* e divisão estratificada (75% treino, 25% avaliação). Todos os modelos foram avaliados no mesmo conjunto de avaliação, garantindo comparabilidade justa dos resultados.

A otimização dos hiperparâmetros foi realizada por busca em grade (*Grid Search*) com validação cruzada 5-fold, adotando o F1-score como métrica principal. As configurações finais selecionadas para cada modelo foram:

- KNN: k=3, weights='distance', metric='manhattan', algorithm='auto';
- Árvore de Decisão: criterion='entropy', max_depth=None, min_samples_split=3, min_samples_leaf=1;
- MLP: hidden_layer_sizes=(100, 50, 25), activation='relu', solver='adam', alpha=0.001, max_iter=2000.

4.1. Desempenho Geral dos Modelos

Nesta subseção é realizada a comparação do desempenho dos modelos no conjunto de avaliação. A Tabela 1 apresenta acurácia, precisão, recall e F1-score de cada algoritmo.

Tabela 1. Desempenho Geral dos Modelos.

Modelo	Acurácia	Precisão	Recall	F1-score
MLP	0,9509	0,8598	0,8214	0,8402
KNN	0,9327	0,7424	0,8750	0,8033
Árvore de Decisão	0,9355	0,7797	0,8214	0,8000

Fonte: Os autores.

O MLP apresentou o melhor desempenho geral, com acurácia de 95,09% e F1-score de 0,8402. Sua precisão de 85,98% indica que a maioria das suas predições de *churn* corresponde a casos reais, minimizando falsos alarmes. O *recall* de 82,14% demonstra boa capacidade de detecção de *churns*.

O KNN alcançou o maior *recall* (87,50%), identificando mais *churns* que os outros modelos, embora com menor precisão (74,24%), o que resultou em mais falsos positivos. Esse perfil pode ser interessante em estratégias que priorizam não perder nenhum cliente potencialmente em risco.

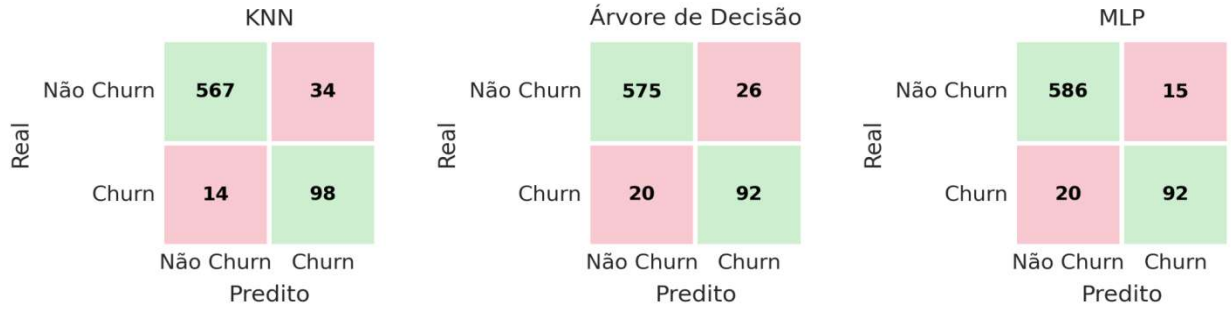
A Árvore de Decisão apresentou desempenho intermediário, com métricas equilibradas, porém ligeiramente inferiores aos demais algoritmos. Sua principal vantagem reside na interpretação das regras de decisão geradas.

4.2. Análise da Matriz de Confusão

Nas matrizes de confusão, TP/TN representam acertos nas classes positiva/negativa; FP são casos previstos como churn que não são (custos de retenção desnecessários); FN são *churns* não identificados (perda de receita). O equilíbrio entre FP e FN define o desempenho da previsão. As matrizes de confusão dos três modelos, apresentadas na Figura 1, revelam padrões distintos de erro:

- MLP: Menor número de falsos positivos (15), indicando maior precisão.
- KNN: Menor número de falsos negativos (14), maximizando a detecção de *churns*. Maior número de falsos positivos (34).
- Árvore de Decisão: Desempenho intermediário em ambos os tipos de erro.

Figura 1. Matrizes de confusão para MLP, KNN e Árvore de Decisão.



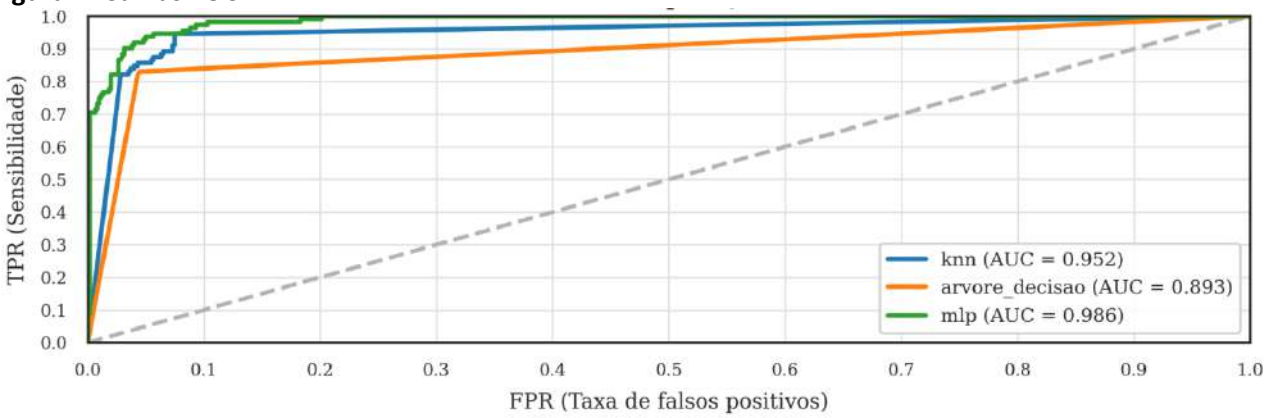
Fonte: Os autores.

Esses resultados evidenciam o trade-off entre precisão e *recall* em cada abordagem e reforçam que a escolha do modelo deve considerar as prioridades do negócio (minimizar perdas ou evitar desperdício de recursos).

4.3. Análise das Curvas ROC e Precision-Recall

A Figura 2 apresenta as curvas *ROC*, que relacionam a taxa de verdadeiros positivos (TPR) e a taxa de falsos positivos (FPR) ao variar o limiar de decisão. A linha diagonal representa um classificador aleatório (*AUC* = 0,5). Quanto mais a curva se aproxima do canto superior esquerdo, melhor a separação entre classes. Observa-se *AUC-ROC* superior do *MLP* (≈ 0,9862), seguida de *KNN* (≈ 0,9515) e Árvore (≈ 0,8932).

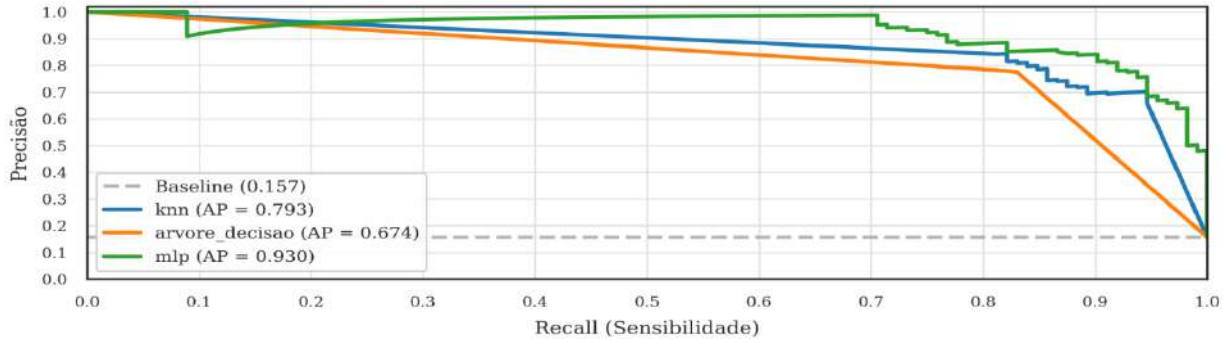
Figura 2. Curvas *ROC*.



Fonte: Os autores.

As curvas *Precision-Recall*, apresentadas na Figura 3, mostram a variação da precisão em função do recall à medida que o limiar de decisão muda. A *Average Precision* (AP) resume a área sob a curva e é particularmente informativa em bases desbalanceadas por focar na classe positiva. O *MLP* manteve maior precisão em altos níveis de *recall* (AP ≈ 0,930), seguido de *KNN* (≈ 0,793) e Árvore (≈ 0,674).

Figura 3. Curvas *Precision-Recall*.



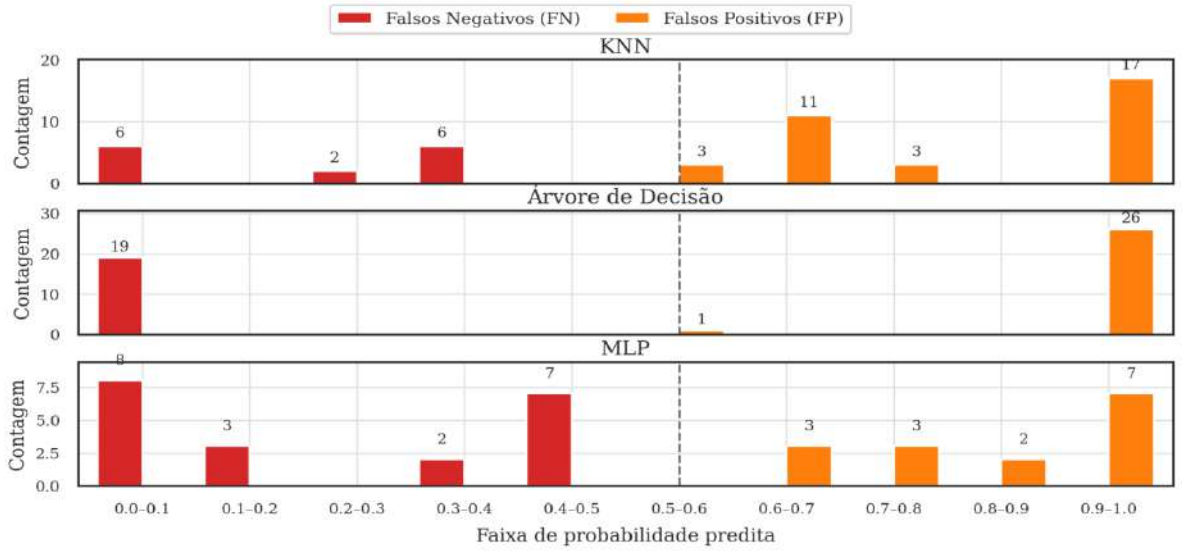
Fonte: Os autores.

4.4. Análise de Erros

Na Figura 4, as barras representam a contagem de erros por faixa de probabilidade predita, e os números sobre as barras indicam a quantidade de casos em cada faixa.

- KNN: Falsos negativos ocorreram principalmente em casos de baixa probabilidade (0,3–0,4), indicando incerteza do modelo. Falsos positivos foram frequentes em probabilidades altas, sugerindo "superconfiança".
- Árvore de Decisão: Falsos negativos distribuídos próximos ao limiar de decisão (0,4–0,5), refletindo incerteza. Falsos positivos concentrados ligeiramente acima de 0,5, indicando decisões borderline.
- MLP: Falsos negativos concentrados em probabilidades baixas (0,2–0,3). Falsos positivos distribuídos entre 0,6–0,8, sem extremos de confiança, sugerindo melhor calibração de probabilidades.

Figura 4. Análise de Erros.



Fonte: Os autores.

O equilíbrio entre falsos positivos e falsos negativos tem impacto direto sobre as estratégias de retenção:

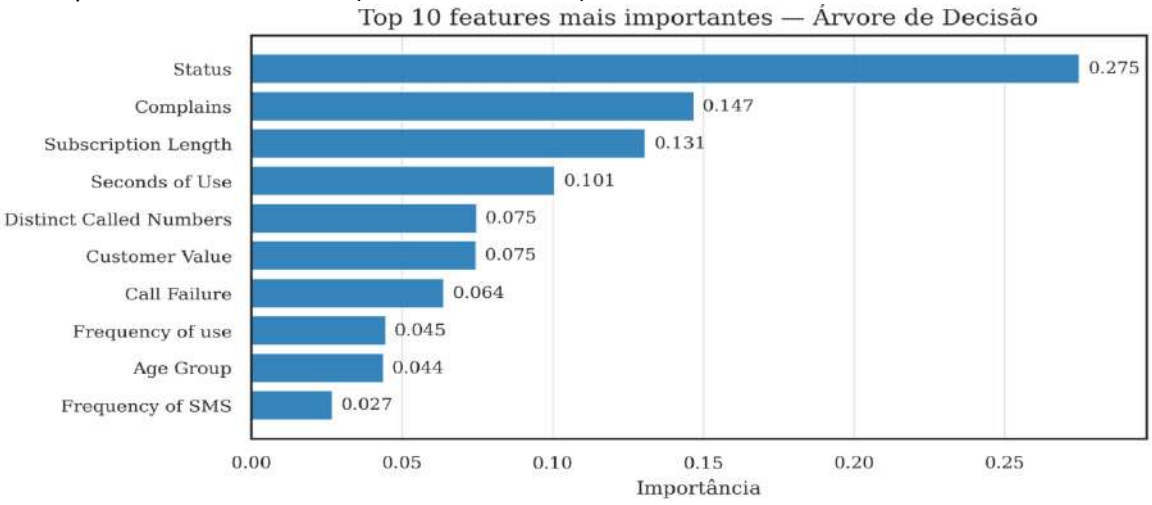
- Falsos negativos representam *churns* não identificados, trazendo perdas potenciais.
- Falsos positivos implicam em recursos gastos desnecessariamente em clientes propensos a permanecer.

O MLP apresentou o melhor compromisso, com poucos falsos positivos e moderados falsos negativos, além de maior calibração das probabilidades, permitindo ajustes finos conforme a estratégia empresarial.

4.5. Importância das Variáveis e Interpretabilidade

A análise de importância das variáveis, baseada na Árvore de Decisão (Figura 5), indica que os principais preditores de *churn* foram: status cadastral do cliente (27,5%), número de reclamações registradas (14,7%) e tempo de assinatura (13,1%). Esses fatores sugerem que situações contratuais críticas, insatisfação manifesta e relacionamento recente elevam o risco de evasão. A Figura 5 apresenta as importâncias relativas das variáveis (redução média de impureza — Gini) em porcentagem; as barras e os números indicam a contribuição percentual de cada atributo para as decisões da Árvore.

Figura 5. Importância das variáveis (Árvore de Decisão).



Fonte: Os autores.

Esses resultados reforçam que a propensão ao *churn* está ligada tanto a aspectos contratuais e de satisfação quanto a padrões de uso e valor do cliente, oferecendo subsídios para ações de retenção mais direcionadas.

5. CONSIDERAÇÕES FINAIS

Este trabalho apresenta uma comparação sistemática de modelos de aprendizado de máquina aplicados à predição de *churn* em telecomunicações, utilizando o *Iranian Churn* Dataset com 3.150 registros reais. Foram avaliados os algoritmos *K-Nearest Neighbors (KNN)*, Árvore de Decisão e *Multilayer Perceptron* (*MLP*), todos otimizados via *Grid Search* e avaliados pelas métricas de acurácia, precisão, *recall* e *F1-score*, conforme recomendações da literatura para problemas de classificação desbalanceada.

Os resultados mostram que o MLP apresentou o melhor desempenho global, com acurácia de 95,09%, *F1-score* de 0,8402 e precisão de 85,98%, sendo a opção mais equilibrada para aplicações que visam otimizar recursos de retenção. O *KNN* destacou-se pelo maior *recall* (87,50%), sendo indicado para cenários onde a prioridade é não perder nenhum churn, mesmo ao custo de mais falsos positivos. A Árvore de Decisão obteve desempenho intermediário, mas se destacou pela interpretabilidade, permitindo a identificação de status, reclamações e tempo de assinatura como principais fatores de risco.

Entre as contribuições do estudo destacam-se o desenvolvimento de um pipeline de avaliação comparativa, a demonstração da viabilidade de modelos preditivos de *churn* em dados reais e o estabelecimento de benchmarks de desempenho para futuras implementações. Como limitações, ressalta-se o uso de dados de uma única operadora e período, além da análise restrita a modelos individuais.

Como trabalhos futuros, recomenda-se a investigação de técnicas ensemble, como Random Forest e Gradient Boosting, que podem potencialmente elevar o desempenho preditivo. Sugere-se ainda a aplicação de métodos de interpretabilidade em modelos mais complexos, como o *MLP*, bem como a replicação da metodologia em diferentes contextos e a inclusão de novas variáveis, visando ampliar a robustez e a aplicabilidade dos resultados.

Conclui-se que modelos de aprendizado de máquina podem ser aplicados de forma eficaz para predição de *churn* em telecomunicações, fornecendo subsídios valiosos tanto para decisões estratégicas quanto para pesquisas futuras na área.

REFERÊNCIAS

AHN, J.; HWANG, J.; KIM, D.; CHOI, H.; KANG, S. A Survey on Churn Analysis in Various Business Domains. **IEEE Access**, v. 8, p. 220816–220839, 2020. DOI: https://doi.org/10.1109/ACCESS.2020.3042657.

SAHA, S.; SAHA, C.; HAQUE, M.; ALAM, G. R.; TALUKDER, A. ChurnNet: Deep Learning Enhanced Customer Churn Prediction in Telecommunication Industry. **IEEE Access**, v. 12, p. 4471–4484, 2024. DOI: https://doi.org/10.1109/ACCESS.2024.3349950

BARSOTTI, A.; GIANINI, G.; MIO, C.; LIN, J.; BABBAR, H.; SINGH, A.; TAHER, F.; DAMIANI, E. A Decade of Churn Prediction Techniques in the TelCo Domain: A Survey. SN **Computer Science**, v. 5, art. 404, 2024. DOI: https://doi.org/10.1007/s42979-024-02722-7

BOJEI, J.; JULIAN, C. C.; CHE WEL, C. A. B.; AHMED, Z. U. The empirical link between relationship marketing tools and consumer retention in retail marketing. **Journal of Consumer Behaviour**, v. 12, n. 1, p. 171–181, 2013. DOI: https://doi.org/10.1002/cb.1408

CHANG, V.; HALL, K.; XU, Q. A.; AMAO, F. O.; GANATRA, M. A.; BENSON, V. Prediction of Customer Churn Behavior in the Telecommunication Industry Using Machine Learning Models. **Algorithms**, v. 17, n. 6, 231, 2024. DOI: https://doi.org/10.3390/a17060231

DUBEY, H.; PUDI, V. Class Based Weighted K-Nearest Neighbor over Imbalance Dataset. *In*: PAKDD 2013. **Lecture Notes in Computer Science**, Springer, 2013. v. 7819. DOI: https://doi.org/10.1007/978-3-642-37456-2_26

GEILER, L.; AFFELDT, S.; NADIF, M. A survey on machine learning methods for churn prediction. International **Journal of Data Science and Analytics**, v. 14, n. 3, p. 217–242, 2022. DOI: https://doi.org/10.1007/s41060-022-00312-5

HASSONAH, M. A.; RODAN, A.; AL-TAMIMI, A-K; ALSAKRAN, J. Churn Prediction: A Comparative Study Using KNN and Decision Trees. In: 2019 Sixth HCT Information Technology Trends (ITT). **IEEE**, 2019. p. 182–186. DOI: https://doi.org/10.1109/ITT48889.2019.9075077

IRANIAN CHURN. **UCI Machine Learning Repository**, 2020. Disponível em: https://doi.org/10.24432/C5JW3Z. Acesso em: 15 fev. 2025.

KING, G.; ZENG, L. Logistic Regression in Rare Events Data. **Political Analysis**, v. 9, n. 2, p. 137–163, 2001. DOI: https://doi.org/10.1093/oxfordjournals.pan.a004868

LALWANI, P.; MISHRA, M. K.; CHADHA, J. S.; SETHI, P. Customer churn prediction system: a machine learning approach. **Computing**, v. 104, n. 2, p. 271–294, 2022. DOI: https://doi.org/10.1007/s00607-021-00908-y

MANZOOR, A.; QURESHI, M. A.; KIDNEY, E.; LONGO, L. A Review on Machine Learning Methods for Customer Churn Prediction and Recommendations for Business Practitioners. **IEEE Access**, v. 12, p. 70434–70457, 2024. DOI: https://doi.org/10.1109/ACCESS.2024.3402092

MOULI, K. C.; RAGHAVENDRAN, C. V.; BHARADWAJ, V. Y.; VYBHAVI, G. Y.; SRAVANI, C.; VAFAEVA, K.M.; DEORARI, R.; HUSSEIN, L. An analysis on classification models for customer churn prediction. **Cogent Engineering**, v. 11, n. 1, 2378877, 2024. DOI: https://doi.org/10.1080/23311916.2024.2378877.

PRZYBYŁA-KASPEREK, M.; MARFO, K.F.; SULIKOWSKI, P. Multi-Layer Perceptron and Radial Basis Function Networks in Predictive Modeling of Churn for Mobile Telecommunications Based on Usage Patterns. **Applied Sciences**, v. 14, n. 20, p. 9226, 2024. DOI: https://doi.org/10.3390/app14209226.

SOLIMAN, W.; RINTA-KAHILA, T. Toward a refined conceptualization of IS discontinuance: Reflection on the past and a way forward. **Information & Management**, v. 57, n. 2, 2020. DOI: https://doi.org/10.1016/j.im.2019.05.002

TANG, L.; XIA, Y. Using XGBoost and MLP for Predicting Customer Churn. Procedia **Computer Science,** v. 174, p. 314–323, 2020. DOI: https://doi.org/10.1016/j.procs.2020.06.081

VAN HULSE, J.; KHOSHGOFTAAR, T. M.; NAPOLITANO, A.; WALD, R. Feature Selection with High-Dimensional Imbalanced Data. *In:* IEEE ICDM WORKSHOPS, Miami, 2009 **Anais** [...]. Miami, 2009. p. 507–514. DOI: https://doi.org/10.1109/ICDMW.2009.35

WANG, S.; LIU, W.; WU, J.; CAO, L.; MENG, Q.; KENNEDY, P. J. Training deep neural networks on imbalanced data sets. *In*: INTERNATIONAL JOINT CONFERENCE ON NEURAL NETWORKS (IJCNN), 2016. **Anais** [...]. 2016. p. 4368–4374. DOI: https://doi.org/10.1109/IJCNN.2016.7727770