

# Análise da Detecção de Fraudes em Transações Financeiras com Modelos de Aprendizado de Máquina

1<sup>st</sup> Gabriel Lima de Souza  
Departamento de Engenharia de Software  
PUC Minas  
Belo Horizonte, Brasil  
gabriel.souza.1354648@sga.pucminas.br

2<sup>nd</sup> Nikolas Louret  
Departamento de Engenharia de Software  
PUC Minas  
Belo Horizonte, Brasil  
navlouret@sga.pucminas.br

3<sup>rd</sup> Frederico dos Santos  
Departamento de Engenharia de Software  
PUC Minas  
Belo Horizonte, Brasil  
frederico.andrade.1318112@sga.pucminas.br

4<sup>th</sup> Gabriel de Souza  
Departamento de Engenharia de Software  
PUC Minas  
Belo Horizonte, Brasil  
gabriel.souza.1365691@sga.pucminas.br

**Abstract**—A detecção de fraudes financeiras é um desafio crescente na era digital, impulsionado pelo aumento das transações online e pela complexidade das operações financeiras. Este estudo investiga o uso de técnicas de aprendizado de máquina para identificar transações fraudulentas em dados de cartões de crédito, com ênfase no modelo *Random Forest*. O objetivo principal é desenvolver um modelo eficaz para a detecção de fraudes financeiras, utilizando um conjunto de dados público do Kaggle contendo transações reais de cartões de crédito europeus. O modelo foi avaliado com métricas como acurácia, precisão, revocação,  $F_1$ -score e AUC, com resultados indicando que o *Random Forest* superou outros modelos, como *Logistic Regression*, *Naive Bayes*, *Support Vector Machine*, e *K-Nearest Neighbors*, em termos de desempenho geral. Os resultados destacam a eficácia do *Random Forest* em identificar fraudes de forma robusta e eficiente, apresentando uma alta capacidade discriminatória, especialmente em cenários com dados desbalanceados. Este trabalho contribui para a literatura acadêmica ao demonstrar o uso prático de modelos de aprendizado de máquina para a detecção de fraudes financeiras, além de discutir as implicações e desafios dessa tecnologia no setor financeiro.

**Index Terms**—Detecção de fraudes, aprendizado de máquina, *Random Forest*, *Logistic Regression*, *Naive Bayes*, *Support Vector Machine*, *K-Nearest Neighbors*, AUC, *undersampling*, transações financeiras

## I. INTRODUÇÃO

A detecção de fraudes financeiras tem se tornado um desafio crescente na era digital, especialmente com o aumento das transações online e a complexidade das operações financeiras. As fraudes financeiras representam uma ameaça significativa à estabilidade econômica e à confiança dos consumidores, afetando diretamente o desempenho e a reputação das instituições financeiras [1]. Com a crescente dependência de transações digitais, torna-se cada vez mais crucial implementar sistemas eficazes para identificar e prevenir fraudes em tempo real.

Na literatura, os métodos de detecção de fraudes são frequentemente divididos em duas abordagens principais: a detecção de uso indevido e a detecção de anomalias. A primeira utiliza técnicas de classificação baseadas no conhecimento prévio dos padrões de fraudes, avaliando se uma transação é fraudulenta ou não com base em dados históricos. Em contraste, a detecção de anomalias visa identificar transações atípicas que se desviam do comportamento normal do usuário, utilizando métodos que modelam o comportamento típico de transações e sinalizam comportamentos irregulares. A eficácia da detecção de anomalias, no entanto, depende da disponibilidade de dados suficientes para construir um perfil robusto de comportamento normal, o que pode ser um desafio em situações de novos tipos de fraudes [2].

Diversos estudos têm explorado diferentes algoritmos de aprendizado de máquina para resolver esse problema. O estudo de Xuan et al. (2018) testou várias metodologias para identificar fraudes em transações de cartões de crédito e concluiu que o método *Random Forest* (RF) obteve a maior precisão. Nesse contexto, o presente trabalho propõe o uso de cinco algoritmos de aprendizado de máquina — *Random Forest*, *Logistic Regression* (LR), *Naive Bayes* (NB), *Support Vector Machine* (SVM) e *K-Nearest Neighbors* (KNN) — para desenvolver um modelo capaz de identificar transações fraudulentas. O conjunto de dados utilizado neste estudo é proveniente do **Kaggle**<sup>1</sup> e contém informações sobre transações de cartões de crédito realizadas por clientes europeus em setembro de 2013.

O objetivo deste estudo é realizar uma análise comparativa do desempenho de diferentes modelos de aprendizado de máquina em um cenário realista para a detecção de fraudes financeiras. Além disso, busca-se contribuir para a literatura

<sup>1</sup>[Online] Disponível em: <https://www.kaggle.com/>

acadêmica ao detalhar os métodos utilizados, os resultados obtidos e as lições aprendidas, ressaltando os desafios e as oportunidades na aplicação de técnicas de inteligência artificial na detecção de fraudes financeiras.

## II. MÉTODO

O método adotado neste trabalho segue uma abordagem sistemática baseada em técnicas de aprendizado de máquina para a detecção de fraudes financeiras.

A Figura 1 apresenta o fluxo metodológico utilizado neste estudo, destacando as etapas principais do processo de detecção de fraudes financeiras com aprendizado de máquina. O diagrama ilustra desde a configuração do ambiente e carregamento dos dados, passando pelo pré-processamento, balanceamento, divisão do conjunto de dados e treinamento, até a avaliação e visualização dos resultados. Cada etapa foi cuidadosamente planejada para garantir a robustez do modelo e a confiabilidade das análises, com foco na identificação precisa de transações fraudulentas e na mitigação de vieses decorrentes do desbalanceamento dos dados.

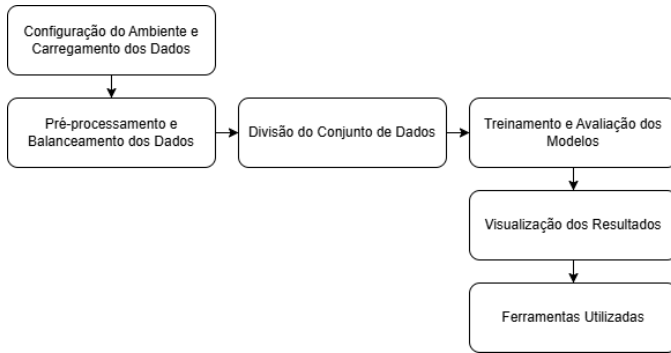


Fig. 1. Fluxo metodológico.

Para alcançar os objetivos propostos, as etapas a seguir foram realizadas:

### A. Configuração do Ambiente e Carregamento dos Dados

A configuração do ambiente de desenvolvimento foi realizada no Google Colab devido à sua capacidade de oferecer recursos computacionais de alto desempenho em nuvem e integração facilitada com diversas bibliotecas de aprendizado de máquina. A API do Kaggle foi configurada para possibilitar o acesso ao conjunto de dados, utilizando credenciais pessoais para autenticação. O **dataset Credit Card Fraud Detection**, amplamente utilizado em estudos relacionados à detecção de fraudes, foi escolhido por conter características anonimizadas e informações específicas sobre transações reais [10].

Após o download do **dataset**, utilizou-se a biblioteca Pandas para carregar os dados em um DataFrame, facilitando a manipulação e análise. A distribuição inicial das classes foi examinada, revelando um alto desbalanceamento entre transações não fraudulentas (99,828%) e fraudulentas (0,172%) [10]. Esse desbalanceamento indicou a necessidade de técnicas de balanceamento para evitar enviesamento dos modelos treinados.

### B. Pré-processamento e Balanceamento dos Dados

A etapa de pré-processamento iniciou-se com a análise estatística dos dados para identificar possíveis valores ausentes, duplicados ou inconsistências. Embora o **dataset** fosse limpo e não apresentasse valores ausentes, foi essencial validar a integridade dos dados antes de prosseguir.

Dada a disparidade na quantidade de exemplos entre as classes, foi aplicado o *undersampling*, uma técnica de balanceamento que reduz a classe majoritária, mantendo o mesmo número de amostras da classe minoritária [9]. A escolha pelo *undersampling* deve-se à sua eficácia em contextos onde os dados são abundantes e onde o foco principal é evitar o aprendizado enviesado. Após o balanceamento, o conjunto resultante continha 984 amostras, sendo metade fraudulentas e metade não fraudulentas.

Essa etapa garantiu que os modelos treinados tivessem condições de aprender características distintivas entre transações fraudulentas e não fraudulentas, sem priorizar excessivamente a classe majoritária.

### C. Divisão do Conjunto de Dados

Com o conjunto de dados balanceado, procedeu-se à divisão em variáveis preditoras ( $X$ ) e variável alvo ( $y$ ). As variáveis preditoras incluem as características anonimizadas do **dataset**, enquanto a variável alvo indica se a transação foi classificada como fraudulenta (1) ou não fraudulenta (0).

Para garantir uma avaliação justa dos modelos, o conjunto foi dividido em dados de treino (80%) e teste (20%) utilizando o método de amostragem estratificada. Esse método assegura que a proporção de classes seja mantida em ambas as divisões, evitando distorções que poderiam impactar o desempenho dos algoritmos [3].

Essa abordagem permitiu treinar os modelos com dados representativos e avaliar seu desempenho em um conjunto não utilizado no treinamento, simulando condições de aplicação real.

### D. Treinamento e Avaliação dos Modelos

Cinco algoritmos de aprendizado de máquina foram selecionados para a tarefa de classificação:

- **RF:** Um modelo baseado em múltiplas árvores de decisão que utiliza o método de *ensemble* para melhorar a precisão e reduzir o risco de *overfitting* [4].
- **SVM:** Um classificador eficiente em problemas de alta dimensionalidade, que busca encontrar o hiperplano ótimo que separa as classes [5].
- **LR:** Um modelo linear simples e interpretável, amplamente utilizado para problemas binários [6].
- **NB:** Um classificador probabilístico baseado no Teorema de Bayes, ideal para dados de alta dimensionalidade e independência entre as variáveis [7].
- **KNN:** Um modelo baseado em instâncias que classifica amostras com base na proximidade de seus vizinhos mais próximos [8].

Os modelos foram treinados no conjunto de treino balanceado e avaliados no conjunto de teste. Foram utilizadas

as métricas de acurácia, matriz de confusão, relatório de classificação (precisão, revocação e  $F_1$ -score) e a AUC da curva ROC. Essas métricas fornecem uma visão abrangente da performance dos modelos em diferentes aspectos, como a capacidade de distinguir entre classes e evitar falsos negativos [3].

#### E. Visualização dos Resultados

A visualização dos resultados foi uma etapa fundamental para interpretar e comparar o desempenho dos modelos de forma intuitiva. Gráficos de barras foram utilizados para comparar a acurácia de cada modelo, enquanto gráficos de pizza representaram a distribuição de previsões (fraudulentas e não fraudulentas) para cada algoritmo.

Mapas de calor das matrizes de confusão destacaram os erros de classificação e os acertos dos modelos, permitindo uma análise detalhada dos tipos de erros cometidos. Além disso, as curvas ROC foram geradas para avaliar a capacidade discriminatória de cada modelo, sendo a AUC uma métrica adicional para determinar a qualidade geral da classificação.

#### F. Ferramentas Utilizadas

As ferramentas empregadas neste trabalho foram selecionadas para atender aos requisitos computacionais e analíticos:

- **Pandas e NumPy:** Para manipulação e análise dos dados.
- **Matplotlib e Seaborn:** Para geração de gráficos e visualizações.
- **scikit-learn:** Para implementação dos modelos de aprendizado de máquina e métricas de avaliação.
- **Google Colab:** Como ambiente de desenvolvimento em nuvem, possibilitando acesso a recursos computacionais avançados.
- **Kaggle:** Repositório utilizado para o download do dataset.

Essas ferramentas proporcionaram um fluxo de trabalho eficiente e reproduzível, permitindo que todas as etapas fossem concluídas de maneira integrada e robusta.

### III. RESULTADOS

Nesta seção, apresentam-se os resultados obtidos nas etapas de pré-processamento, balanceamento, treinamento e avaliação dos modelos de aprendizado de máquina aplicados à detecção de fraudes financeiras.

#### A. Distribuição Inicial dos Dados

O conjunto de dados inicial apresentou um alto desbalanceamento entre as classes: 99,828% das transações eram não fraudulentas e apenas 0,172% eram fraudulentas, conforme ilustrado na Figura 2. Esse desbalanceamento é comum em sistemas de detecção de fraudes [9] e pode causar vieses nos modelos, priorizando a classe majoritária e ignorando fraudes.

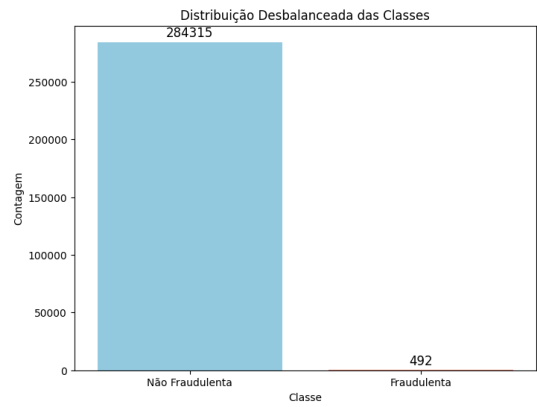


Fig. 2. Distribuição inicial das classes no conjunto de dados.

#### B. Balanceamento dos Dados

Para corrigir o desbalanceamento, foi aplicado o método de *undersampling*, ajustando a classe majoritária para o mesmo número de exemplos da classe minoritária. A Figura 3 ilustra a distribuição balanceada resultante, com 492 amostras em cada classe. Embora essa técnica melhore a representatividade das classes, ela reduz a quantidade total de dados disponíveis, podendo impactar a generalização dos modelos.

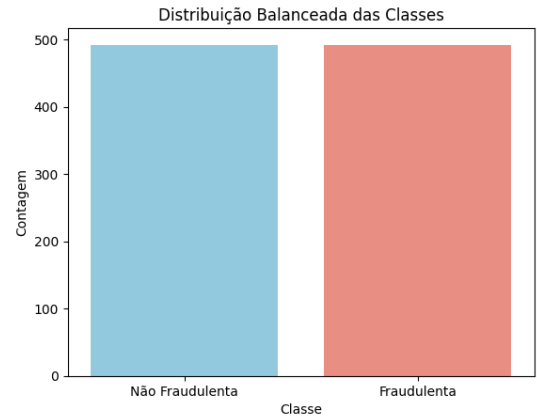


Fig. 3. Distribuição balanceada das classes após *undersampling*.

#### C. Análise dos Resultados

Os cinco modelos avaliados foram comparados utilizando métricas como acurácia, precisão, revocação,  $F_1$ -score e AUC. A Tabela I resume os resultados obtidos.

TABLE I  
RESUMO DAS MÉTRICAS PARA CADA MODELO AVALIADO.

Métrica	RF	LR	NB	SVM	KNN
Acurácia (%)	94,92	93,91	83,76	55,33	58,38
Precisão (%)	92,78	91,58	68,00	51,61	52,00
Revocação (%)	94,62	92,85	92,31	59,18	60,00
$F_1$ -score (%)	93,69	92,21	78,37	55,10	55,74
AUC	0,9793	0,9813	0,9555	0,5920	0,6216

Entre os modelos avaliados, RF apresentou o melhor desempenho geral, com acurácia de 94,92% e AUC de 0,9793,

seguido de perto pelo LR, com acurácia de 93,91% e AUC de 0,9813. Ambos demonstraram alta capacidade discriminatória e consistência na classificação de fraudes e não fraudes.

Por outro lado, NB apresentou uma revocação elevada (92,31%), indicando boa capacidade de identificar transações fraudulentas. No entanto, sua precisão foi baixa (68,00%), sugerindo muitos falsos positivos. Já SVM e KNN tiveram desempenhos insatisfatórios, com acurácias inferiores a 60%, demonstrando limitações significativas no aprendizado de padrões em dados balanceados artificialmente.

O impacto do *undersampling* foi perceptível. Enquanto RF e LR foram pouco afetados pela redução no tamanho do conjunto de dados, SVM e KNN mostraram sensibilidade às alterações na densidade e estrutura dos dados, apresentando dificuldade em separar as classes de forma eficaz.

#### D. Visualização de Resultados por Modelo

As matrizes de confusão destacam o desempenho detalhado dos modelos, revelando acertos e erros em cada classe. A seguir, discutem-se as matrizes de confusão para os principais modelos.

a) *Modelo RF*: A Figura 4 mostra a matriz de confusão para o modelo *RF*. O modelo obteve uma taxa de verdadeiros positivos de 94,62% (fraudes corretamente identificadas) e uma taxa de verdadeiros negativos de 92,78% (não fraudes corretamente classificadas). Esses valores indicam um bom equilíbrio entre a identificação de fraudes e a classificação correta das transações não fraudulentas.

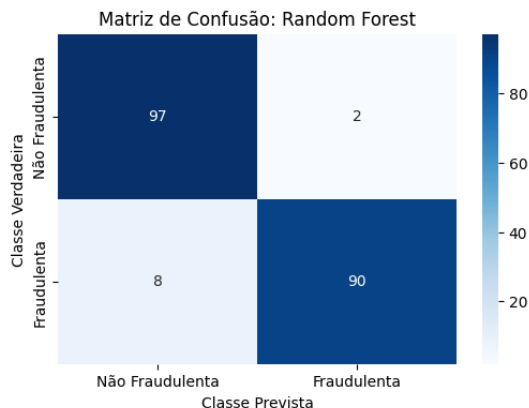


Fig. 4. Matriz de confusão para o modelo RF.

b) *Modelo LR*: A Figura 5 apresenta a matriz de confusão para o modelo *LR*. O modelo teve 92,85% de revocação (fraudes corretamente identificadas) e 91,58% de precisão. No entanto, houve uma proporção de falsos negativos, onde 7,15% das fraudes não foram identificadas, o que resultou em uma leve perda de acurácia, mas sem comprometer significativamente o desempenho geral.

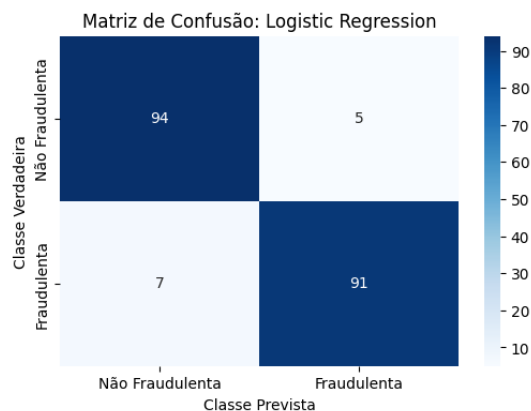


Fig. 5. Matriz de confusão para o modelo LR.

c) *Modelo NB*: A Figura 6 ilustra a matriz de confusão para o *Naive Bayes*. Embora o modelo tenha apresentado uma alta revocação de 92,31% (indicando boa capacidade de identificar fraudes), a precisão foi de apenas 68,00%, sugerindo que 32% das transações classificadas como fraudulentas eram, na verdade, não fraudulentas (falsos positivos).

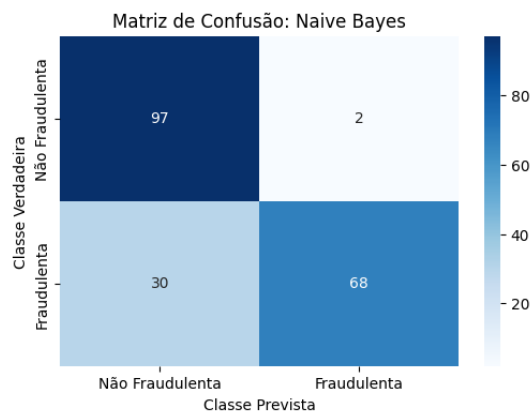


Fig. 6. Matriz de confusão para o modelo NB.

d) *Modelo SVM*: A Figura 7 apresenta a matriz de confusão do modelo *SVM*, que obteve uma revocação de 59,18% e uma precisão de 51,61%. O modelo classificou como não fraudulentas 59,18% das transações fraudulentas (falsos negativos), o que resultou em um desempenho inferior, evidenciado pela AUC de 0,5920. O modelo teve dificuldades em separar adequadamente as classes, mostrando limitações no aprendizado de padrões discriminatórios.

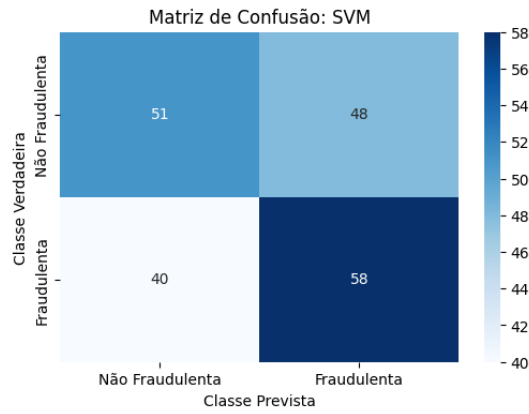


Fig. 7. Matriz de confusão para o modelo SVM.

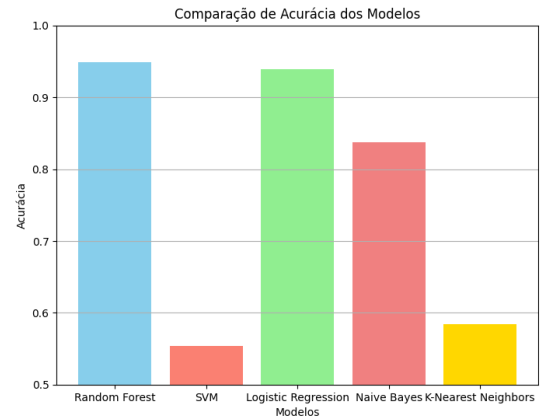


Fig. 9. Comparação da acurácia entre os modelos avaliados.

e) *Modelo KNN*: A Figura 8 apresenta a matriz de confusão para o KNN, que obteve uma revocação de 60,00% e uma precisão de 52,00%. O modelo cometeu falsos negativos, classificando como não fraudulentas 40,00% das transações fraudulentas. Esse comportamento sugere que o modelo teve dificuldade em distinguir entre fraudes e transações legítimas após o balanceamento por *undersampling*, o que impactou sua capacidade de generalização.

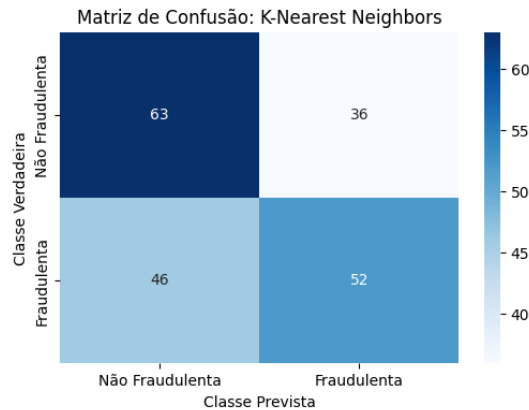


Fig. 8. Matriz de confusão para o modelo KNN.

#### E. Comparação Geral dos Modelos

A Figura 9 mostra a comparação da acurácia dos modelos, destacando o desempenho superior de RF e LR, com valores muito próximos entre si. As curvas ROC, na Figura 10, evidenciam a alta capacidade discriminatória desses modelos, com AUCs próximas de 1,0.

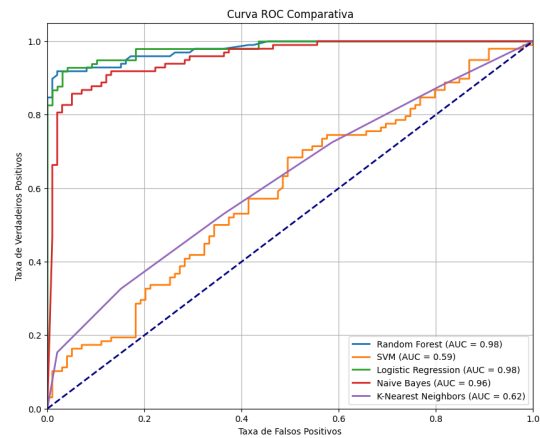


Fig. 10. Curvas ROC para os modelos avaliados.

#### F. Considerações Finais

Os resultados evidenciam que RF e LR são os modelos mais adequados para a detecção de fraudes neste cenário, devido à sua robustez e alta capacidade discriminatória. Em contrapartida, os modelos SVM e KNN apresentaram limitações claras, reforçando a importância de selecionar algoritmos apropriados para dados balanceados artificialmente. Esses achados sugerem que técnicas de balanceamento, como o *undersampling*, devem ser usadas com cautela, pois podem impactar significativamente o desempenho de determinados algoritmos.

#### IV. CONCLUSÃO

Este trabalho apresentou o desenvolvimento e avaliação de modelos de aprendizado de máquina para a detecção de fraudes em transações financeiras. Os modelos RF e LR se destacaram pela alta acurácia, capacidade discriminatória e equilíbrio entre precisão e revocação, alcançando AUCs superiores a 0,97. O modelo RF obteve o melhor desempenho geral, com uma acurácia de 94,92% e um  $F_1$ -score de 93,69%. Em contraste, os modelos SVM e KNN apresentaram acurácias abaixo de 60%, com desempenho insatisfatório na detecção de fraudes.

A análise do impacto do balanceamento dos dados, por meio do *undersampling*, mostrou que os modelos RF e LR foram menos sensíveis à redução da classe majoritária, mantendo boa capacidade de generalização. Por outro lado, os modelos SVM e KNN, mais dependentes da estrutura dos dados, tiveram seu desempenho comprometido pela alteração na distribuição das classes. O NB, apesar de apresentar boa revocação, teve baixa precisão, indicando muitos falsos positivos.

Com base nos resultados, podemos concluir que os modelos RF e LR são adequados para a detecção de fraudes financeiras em contextos desbalanceados. No entanto, é importante considerar a escolha do modelo e as técnicas de balanceamento de dados, uma vez que essas influenciam diretamente a performance. Futuros trabalhos podem explorar o uso de outras técnicas de balanceamento e modelos mais complexos, como redes neurais, para melhorar a acurácia e a capacidade de detecção.

#### REFERENCES

- [1] D. Varmedja, M. Karanovic, S. Sladojevic, M. Arsenovic and A. Anderla, "Credit Card Fraud Detection Machine Learning methods," 2019 18th International Symposium INFOTEH-JAHORINA (INFOTEH), East Sarajevo, Bosnia and Herzegovina, 2019, pp. 1-5, doi: 10.1109/INFOTEH.2019.8717766.
- [2] S. Xuan, G. Liu, Z. Li, L. Zheng, S. Wang and C. Jiang, "Random forest for credit card fraud detection," 2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC), Zhuhai, China, 2018, pp. 1-6, doi: 10.1109/ICNSC.2018.8361343.
- [3] S. Raschka and V. Mirjalili, *Python Machine Learning: Machine Learning and Deep Learning with Python, scikit-learn, and TensorFlow*, 3rd ed., Birmingham, UK: Packt Publishing, 2019.
- [4] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5-32, 2001. doi: 10.1023/A:1010933404324.
- [5] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273-297, 1995. doi: 10.1007/BF00994018.
- [6] D. W. Hosmer, S. Lemeshow, and R. X. Sturdivant, *Applied Logistic Regression*, 3rd ed., Hoboken, NJ, USA: John Wiley & Sons, 2013. doi: 10.1002/9781118548387.
- [7] H. Zhang, "The optimality of Naive Bayes," *AAAI Conference on Artificial Intelligence*, pp. 562-567, 2004.
- [8] N. S. Altman, "An introduction to kernel and nearest-neighbor non-parametric regression," *The American Statistician*, vol. 46, no. 3, pp. 175-185, 1992. doi: 10.1080/00031305.1992.10475879.
- [9] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 21, no. 9, pp. 1263-1284, Sept. 2009. doi: 10.1109/TKDE.2008.239.
- [10] A. Pozzolo, "Credit Card Fraud Detection," Kaggle, 2016. [Online]. Available: <https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud>. [Accessed: Oct. 30, 2024].