

Detecção de Fraudes em Transações Financeiras: Uma Abordagem de Machine Learning para Ligia

Guilherme José

Universidade Federal de Pernambuco (UFPE)

Recife, PE, Brasil

Resumo—A detecção de fraudes em transações financeiras representa um desafio crítico devido à natureza altamente desbalanceada dos dados e à constante evolução das táticas fraudulentas. Este relatório descreve a jornada de pesquisa e desenvolvimento de uma solução de Machine Learning para o desafio técnico da Liga de IA da UFPE (Ligia). O estudo abrange desde a exploração inicial e tratamento de dados anônimos até a engenharia de atributos temporais e a avaliação de múltiplos algoritmos. O objetivo principal é maximizar a métrica ROC-AUC, garantindo alta detecção da classe minoritária sem comprometer a precisão global do sistema.

Index Terms—Detecção de Fraude, Machine Learning, Dados Desbalanceados, Análise Exploratória, Redes Neurais.

I. INTRODUÇÃO

O crescimento das transações financeiras digitais trouxe consigo um aumento proporcional nas tentativas de fraude, exigindo sistemas de segurança cada vez mais sofisticados. No contexto de cartões de crédito e pagamentos eletrônicos, o problema de detecção de fraudes é classicamente modelado como uma tarefa de classificação binária, onde o principal obstáculo é o extremo desbalanceamento das classes: transações legítimas ocorrem em volume massivamente superior às fraudulentas.

Este trabalho documenta a solução desenvolvida para o desafio de Machine Learning da Ligia. A relevância da tarefa reside na necessidade de criar um modelo capaz de identificar padrões anômalos sutis sem gerar um excesso de falsos positivos, o que prejudicaria a experiência do usuário com bloqueios indevidos de cartão. A abordagem foca na extração de características temporais relevantes e no treinamento de um modelo robusto, avaliado pela métrica ROC-AUC.

II. METODOLOGIA

A estratégia de desenvolvimento dividiu-se em análise exploratória, engenharia de características e modelagem preditiva. O conjunto de dados utilizado compreende variáveis contínuas anonimizadas via Análise de Componentes Principais (V1-V28), *Time* (segundos), *Amount* (valor) e *Class* (variável alvo).

A. Inspeção Inicial e Tratamento de Outliers

A inspeção inicial confirmou a integridade da base, sem dados faltantes ou duplicados. Para a detecção de valores atípicos, empregou-se o método estatístico do Intervalo Interquartil (IQR). A análise revelou uma quantidade substancial de *outliers* nas componentes principais e no valor financeiro

(*Amount*). Contudo, uma investigação aprofundada ao cruzar estes pontos fora da curva com a variável alvo revelou um padrão fundamental: a proporção de transações fraudulentas aninhadas nesses *outliers* era significativamente superior à proporção encontrada na distribuição normal da base.

Como fraudes são inherentemente anomalias comportamentais (frequentemente manifestadas como valores atípicos em horários incomuns), a remoção tradicional de *outliers* excluiria instâncias vitais da classe minoritária, destruindo o sinal preditivo. Assim, estabeleceu-se a premissa de manter 100% dos registros atípicos, utilizando algoritmos de escalonamento robustos posteriormente para mitigar distorções geométricas.

B. Análise Multivariada

Para avaliar a interação entre as características, calculou-se a matriz de correlação de Pearson. Notou-se forte associação entre as variáveis anonimizadas e o valor da transação (*Amount*), permitindo isolar as *features* com maior correlação com a classe alvo (com destaque para V17, V14, V12, V11 e V4). Adicionalmente, a análise bivariada evidenciou que as fraudes evitam valores financeiros extremos, configurando uma tática clara para burlar os sistemas de alerta de limite.

Em relação à variável temporal, a conversão momentânea da variável contínua *Time* em faixas de horas do dia permitiu uma investigação mais aprofundada. A distribuição foi analisada por meio da Estimativa de Densidade de Kernel (KDE) (ver Figura 1). Essa análise expôs um padrão comportamental claro: durante a madrugada, as fraudes representam uma proporção significativamente maior em relação às transações legítimas. Além disso, observou-se que durante o dia, embora o volume absoluto de fraudes cresça, a quantidade de operações normais aumenta exponencialmente. Esse fenômeno cria um intenso "ruído" estatístico que camufla as ações fraudulentas em meio ao tráfego legítimo, justificando a importância de extrair sinais precisos dessa variável.

Em seguida, a projeção de redução de dimensionalidade via algoritmo t-SNE (ver Figura 2) — executada sobre uma subamostra representativa para viabilizar o alto custo computacional — revelou que uma parcela significativa das fraudes pode ser separada em *clusters* bem definidos. No entanto, a projeção também expôs densas áreas de sobreposição onde as anomalias se misturam perfeitamente com transações normais, justificando o uso posterior de modelos não-lineares.

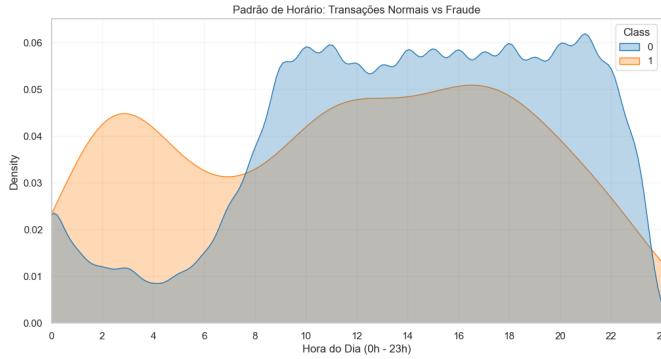


Figura 1. Estimativa de Densidade de Kernel (KDE) ilustrando a proporção relativa de fraudes superior durante a madrugada.

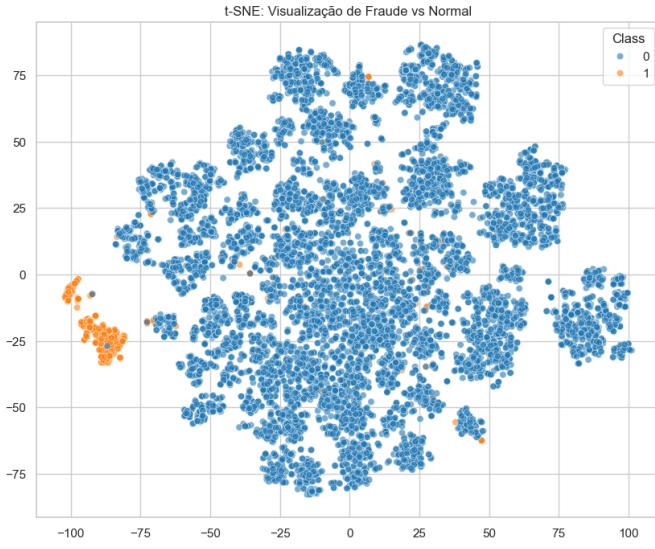


Figura 2. Projeção t-SNE: clusters de fraude formam-se em meio a uma forte sobreposição com transações legítimas.

C. Engenharia de Características (Feature Engineering)

A variável linear *Time* foi convertida em hora do dia (*Hour*). Para preservar a adjacência cronológica entre as extremidades do dia (23h e 00h), aplicaram-se transformações trigonométricas:

$$Hour_Sin = \sin\left(\frac{2\pi \times Hour}{24}\right) \quad (1)$$

$$Hour_Cos = \cos\left(\frac{2\pi \times Hour}{24}\right) \quad (2)$$

A representação linear (*Hour*) foi mantida junto às cíclicas para fornecer a progressão contínua e a representação bidimensional do ciclo.

D. Preparação de Dados e Seleção de Atributos

Para prevenir o vazamento de dados (*data leakage*), a divisão estratificada da base em Treino (80%) e Validação (20%) precedeu qualquer etapa analítica de seleção. O *RobustScaler* foi ajustado exclusivamente no subconjunto de treino para

normalizar as variáveis contínuas, mitigando o impacto das distorções geradas pelos *outliers* mantidos.

Visando identificar tanto relações preditivas diretas quanto interações matemáticas complexas com a classe alvo, adotou-se uma estratégia híbrida de seleção de atributos. Foram descartadas apenas as variáveis que demonstraram irrelevância simultânea em critérios lineares (Correlação de Pearson normalizada < 0,04) e não-lineares (Importância de atributos via *Random Forest* < 0,015).

Essa filtragem dupla assegurou a remoção de ruído puro, culminando na exclusão das variáveis *Time*, *id* e da categorização manual *Amount_Risk_Level*. Por outro lado, embora a variável *Hour_sin* tenha pontuado abaixo da nota de corte linear, ela foi forçosamente mantida por demonstrar relevância no modelo baseado em árvores e para preservar a integridade espacial do par vetorial temporal.

E. Modelagem Preditiva

Seis algoritmos (Regressão Logística, Random Forest, XGBoost, LightGBM, CatBoost e MLPClassifier) foram treinados via Validação Cruzada Estratificada ($k=5$) e otimizados via RandomizedSearchCV visando maximizar o ROC-AUC. Testaram-se hiperparâmetros diversos (taxas de aprendizado, profundidade de árvores, regularização ElasticNet e arquiteturas neurais).

III. RESULTADOS E DISCUSSÃO

A. Comparação entre Modelos e Baselines

Para contextualizar e mensurar os ganhos de desempenho dos 6 algoritmos otimizados, avaliou-se inicialmente três modelos de referência (*baselines*): **Mais Frequente** (prediz sempre a classe majoritária), **Estratificado** (preserva a proporção das classes) e **Aleatório** (chute uniforme).

Tabela I
MODELOS TESTADOS VS BASELINES (VALIDAÇÃO)

Modelo	ROC-AUC	Acc.	Prec.	Recall	F1
Base: Aleatório	0.5000	0.5001	0.0017	0.4810	0.0033
Base: Estratif.	0.5055	0.9967	0.0133	0.0127	0.0130
Base: Frequente	0.5000	0.9983	0.0000	0.0000	0.0000
Rede Neural	0.9762	0.9993	0.8333	0.7595	0.7947
LightGBM	0.9750	0.9995	0.9242	0.7722	0.8414
CatBoost	0.9746	0.9993	0.8267	0.7848	0.8052
XGBoost	0.9742	0.9991	0.7273	0.8101	0.7665
Random Forest	0.9732	0.9991	0.6989	0.8228	0.7558
Reg. Logística	0.9725	0.9812	0.0735	0.8481	0.1354

A Tabela I avalia detalhadamente as previsões na validação inédita e reforça a falácia da Acurácia em dados desbalanceados: o *baseline* Mais Frequente atinge quase 100% de acerto geral, mas tem *Recall* nulo para fraudes. Em contrapartida, os modelos de Machine Learning fornecem um salto absoluto. A **Rede Neural** obteve o maior ROC-AUC (0.9762), cumprindo o objetivo primário de ordenação probabilística global. Sob a ótica de negócios, o **LightGBM** destacou-se com 0.9242 de Precisão, errando pouquíssimos Falsos Positivos, minimizando bloqueios indevidos de clientes legítimos. Observa-se claramente o *trade-off* entre minimizar falsos alarmes e

capturar o maior volume de fraudes (maior *Recall*, liderado pela Regressão Logística e Random Forest).

B. Desempenho e Matriz de Decisão

A avaliação das Curvas ROC (ver Figura 3) consolidou a superioridade de separação probabilística dos algoritmos não-lineares avaliados na Tabela I. Adicionalmente, as Matrizes de Confusão (ver Figura 4) evidenciam o fracasso da borda linear: apesar do alto *Recall*, a Regressão Logística gerou milhares de falsos alarmes (Precisão de apenas ~7%), sendo comercialmente inviável. Modelos não-lineares, por outro lado, equilibraram eficientemente essa relação.

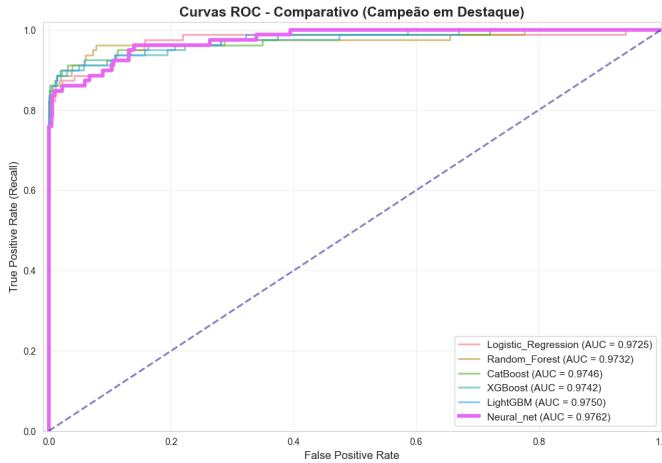


Figura 3. Curvas ROC indicando a convergência probabilística de excelência dos algoritmos avaliados.

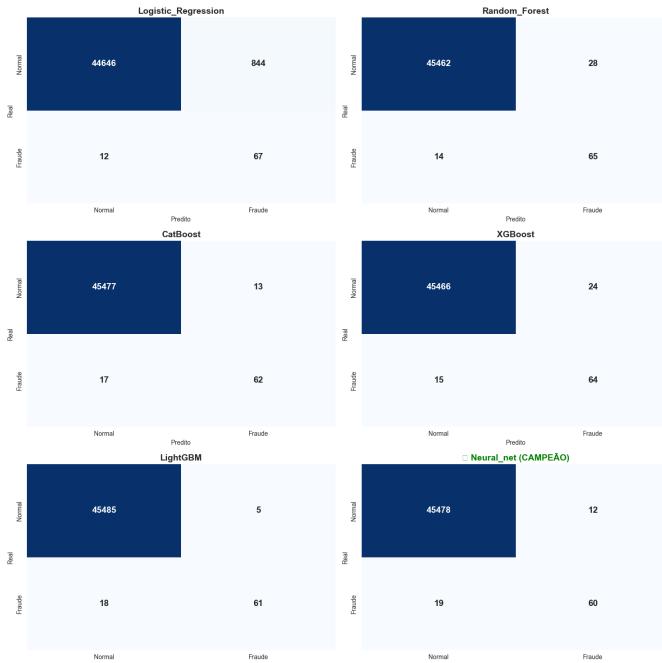


Figura 4. Matrizes de Confusão destacando a taxa massiva de Falsos Positivos no modelo linear de Regressão Logística.

C. Avaliação no Kaggle (Teste Público)

As probabilidades geradas pelos melhores modelos foram enviadas ao **leaderboard** oficial (*test.csv*). Surpreendentemente, o desempenho cego foi superior à validação interna, provando a ausência de **overfitting** e a robustez do escalonamento utilizado.

Tabela II
DESEMPENHO (ROC-AUC) NO TESTE PÚBLICO (KAGGLE)

Modelo	ROC-AUC Público
Neural Net (MLP)	0.99208
XGBoost	0.99103
LightGBM	0.98615
CatBoost	0.98232

A **Rede Neural** liderou as submissões alcançando um **ROC-AUC Público de 0.99208**, seguida pelo **XGBoost** (0.99103). Estes dois algoritmos foram selecionados para a análise de explicabilidade.

IV. INTERPRETABILIDADE

Para garantir que a Rede Neural (campeã) e o XGBoost não atuassem como "caixas-pretas", aplicou-se a biblioteca SHAP (*SHapley Additive exPlanations*). O método baseia-se na teoria dos jogos cooperativos para calcular a contribuição marginal exata de cada variável na predição final de uma transação, permitindo desvendar a lógica interna dos algoritmos.

A. Impacto de Negócio e Eficiência Operacional

No conjunto de validação, a Rede Neural bloqueou com sucesso 60 transações fraudulentas reais. O seu grande diferencial competitivo comprovou-se na sua Precisão: o modelo gerou apenas 12 bloqueios indevidos. Em um cenário bancário real, isso garante um atrito mínimo na experiência dos clientes, poupando a instituição de altos custos operacionais com equipes de atendimento (*call center*) e diminuindo severamente o risco de insatisfação.

B. Análise Global de Atributos e Direcionalidade (SHAP)

O *Summary Plot* ilustrando a topologia de decisão da Rede Neural (ver Figura 5) revela que o valor financeiro (*Amount*) e as componentes *V5* e *V17* encabeçam a hierarquia de importância global. Além disso, o gráfico consolida o triunfo empírico da variável *Hour_sin* ao figurar no influente "top 10" do modelo. Isso fornece a prova visual de que a rede neural aprendeu ativamente a utilizar a trigonometria temporal para mapear anomalias da madrugada. Analisando o espectro de cores e a direcionalidade, nota-se um padrão fundamental: para essas principais variáveis (como *Amount*, *V17* e *Hour_sin*), são os **valores mais baixos (representados em azul)** que possuem SHAP positivo, empurrando a probabilidade ativamente a favor da fraude, enquanto valores extremos de alta magnitude (vermelhos) protegem a transação, classificando-a como normal.

O modelo XGBoost (ver Figura 6) corrobora a relevância desse mesmo ecossistema de variáveis-chave, embora priorize

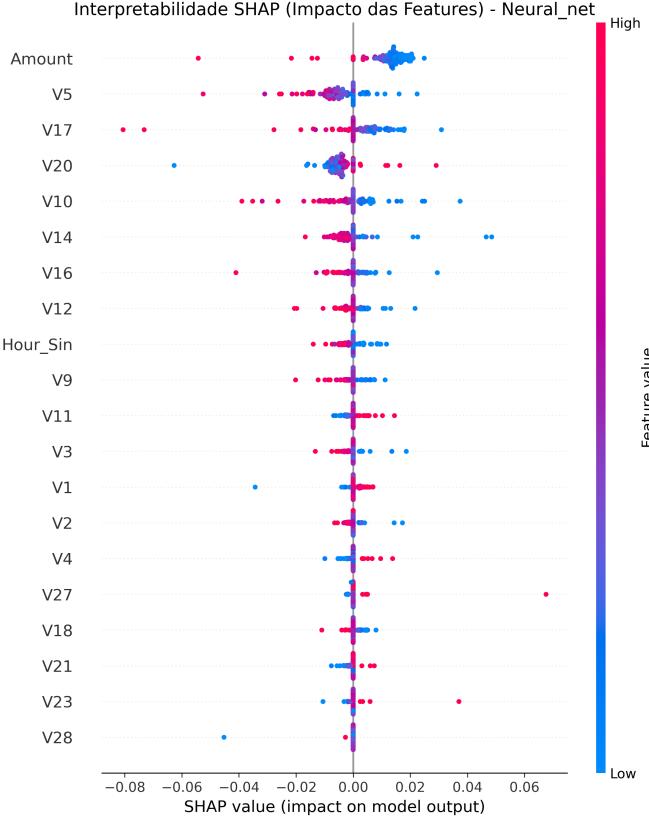


Figura 5. SHAP Summary Plot (Rede Neural). Cores quentes indicam valores altos da característica; o eixo X reflete o impacto na predição.

componentes distintas no topo absoluto, notavelmente *V14* e *V4*. A grande diferença metodológica fica evidente na transição visual da predição: enquanto as árvores do XGBoost agrupam as decisões em cortes ortogonais (formando blocos densos e estratificados de cores), a Rede Neural cria transições de gradiente muito mais suaves, contínuas e distribuídas. Essa suavidade topológica justifica a ligeira vantagem de abstração matemática da rede neural para generalizar padrões complexos de forma mais orgânica no Kaggle.

V. LIMITAÇÕES E TRABALHOS FUTUROS

Apesar da altíssima acurácia, a abordagem possui limitações de desenvolvimento:

- **Falsos Negativos:** A evasão de 19 fraudes na validação (*Recall* de ~76%) exige atenção. Trabalhos futuros devem focar na geração sintética de dados (SMOTE/ADASYN) exclusivamente para as fraudes atípicas não detectadas.
- **Custo Computacional:** A explicabilidade por *KernelExplainer* é lenta para inferência bancária em tempo real, exigindo otimizações de arquitetura.
- **Modelagem em Grafos:** Como passo evolutivo, a utilização de Redes Neurais em Grafo (GNNs) permitiria mapear a teia de conexões entre contas, atuando contra as transferências dinâmicas de quadrilhas.

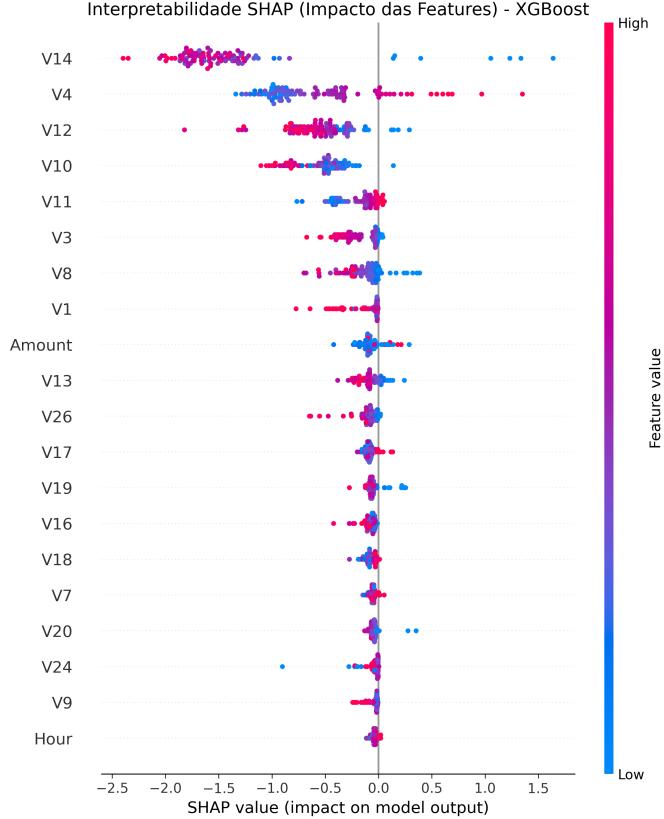


Figura 6. SHAP Summary Plot (XGBoost). Nota-se a formação de cortes mais secos e blocos densos, característicos de modelos baseados em árvores.

VI. CONCLUSÃO

O estudo documentou uma solução altamente eficaz contra o severo desbalanceamento de fraudes financeiras do desafio Ligia. A vitória da arquitetura de Rede Neural sobressai-se aos algoritmos *Gradient Boosting*, não apenas pela capacidade do modelo, mas devido ao rigor na preparação dos dados: a manutenção justificada dos *outliers*, o uso do *RobustScaler* e a modelagem temporal cíclica. Alcançando a marca de 0.992 no teste cego com explicabilidade validada via matrizes de Shapley, o sistema desenvolvido prova-se comercialmente seguro, acurado e perfeitamente aplicável ao ecossistema financeiro.

REFERÊNCIAS

- [1] V. D. Oliveira, “Métodos de machine learning na detecção de fraude em cartão de crédito: Um estudo comparado,” *Revista Científica Acerite*, vol. 5, no. 9, p. e59265, 2025. DOI: 10.63026/acerite.v5i9.265.
- [2] Kaggle, “Credit Card Fraud Detection Dataset / Desafio Ligia,” Kaggle Inc., 2024. [Online]. Disponível: <https://www.kaggle.com/competitions/lidia-machine-learning/overview>
- [3] S. M. Lundberg and S.-I. Lee, “A Unified Approach to Interpreting Model Predictions,” in *Advances in Neural Information Processing Systems 30*, Curran Associates, Inc., 2017, pp. 4765–4774.
- [4] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785–794.
- [5] F. Pedregosa *et al.*, “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.