# Machine Learning Algorithms for Financial Fraud Detection

[1] Ayesha Khalid, [2] Muhammad Hamza Siddiqui, [3] Sana Iftikhar

[1] Department of Computer Science, National University of Computer & Emerging Sciences (FAST-NUCES), Lahore, Pakistan
[2,] School of Electrical Engineering & Computer Science, National University of Sciences & Technology (NUST), Islamabad, Pakistan
[3,]Department of Information Systems, University of the Punjab, Lahore, Pakistan
***Corresponding Author:** ayesha.khalid@nu.edu.pk*

***Abstract:*** *Financial institutions face a persistent arms race against increasingly sophisticated fraud, from card-not-present (CNP) transactions and account takeovers to synthetic identities and mule networks. This article surveys and systematizes machine-learning (ML) approaches for fraud detection across supervised, semi-supervised, and graph-based paradigms, with a focus on severe class imbalance, concept drift, and real-time constraints. We review data pipelines, feature engineering for transactional and device telemetry, and algorithmic choices (tree ensembles, margin-based models, deep learning, and graph neural networks). We discuss cost-sensitive learning, imbalanced evaluation (PR-AUC, MCC), explainability (SHAP/LIME), and privacy-preserving deployment (federated learning, differential privacy). A comparative illustration highlights the practical trade-offs among commonly used algorithms. We conclude with a roadmap that integrates MLOps, human-in-the-loop review, and regulatory compliance to reduce false positives while capturing evolving fraud patterns.*

***Keywords:*** *Blockchain, Blockchain Networks, Frameworks in blockchain, Improvement in blockchain, Performance, Optimization in performance, Throughput, Performance, Number of transactions.*

## Introduction

Digital payments and instant rails have compressed decision windows for risk engines from minutes to milliseconds, while adversaries rapidly re-tool via automation and compromised credentials. Traditional rules engines—although interpretable—struggle with latency, coverage, and maintenance. Machine learning augments or replaces static rules by learning non-linear boundaries from behavioral, network, and device-level signals. However, fraud events are rare (often <0.5%), labels are delayed, distributions drift with new campaigns, and business costs are asymmetric: false negatives (missed fraud) are expensive, but false positives erode customer experience and revenue. Effective fraud systems therefore combine: (i) high-recall models (tree ensembles, GNNs) with calibrated thresholds; (ii) continual learning and drift monitoring; (iii) cost-sensitive optimization aligned to chargeback economics; (iv) explainability for adverse-action notice and model governance; and (v) privacy-preserving data flows compatible with PCI-DSS and data-protection regimes.

## Data Sources and Labeling

Financial fraud detection relies heavily on the diversity, granularity, and reliability of data sources. The foundation of modern fraud detection systems lies in **transactional data**, which includes

variables such as transaction amount, merchant category codes (MCC), merchant risk ratings, transaction timestamps, and channel identifiers (POS, mobile, or web). These data provide the immediate behavioral signature of a payment event. In addition, **customer and account metadata**—such as age of the account, account type, spending patterns, and linked payment instruments—offer contextual depth that helps distinguish legitimate deviations from fraudulent anomalies.

Device-level indicators also play a critical role. **Device and browser fingerprints**, operating system identifiers, and screen resolution data capture subtle changes that often accompany fraudulent access attempts. **IP geolocation** and proxy detection further contribute by revealing geographic inconsistencies, such as improbable location jumps within short time intervals. **Velocity features**, derived from the rate of transactions across accounts, devices, or IPs, identify bursts of activity typical in bot-driven or coordinated fraud.

Labeling fraud data remains one of the most challenging tasks. True labels are often derived from **chargeback outcomes** or confirmed customer disputes, which may appear weeks or months after the transaction, introducing **label lag**. To counter this delay, researchers and practitioners employ **delayed feedback learning** mechanisms that adjust models as verified labels accumulate. Furthermore, due to the rarity of confirmed fraud cases, **positive-unlabeled (PU) learning** strategies are used—treating known fraudulent transactions as positive examples and assuming the rest as unlabeled rather than genuine.

Another critical data stream emerges from **consortium and network-level signals**, where multiple financial institutions share anonymized risk intelligence to detect patterns spanning different banks or merchants. When properly aggregated, these shared datasets improve recall and help capture previously unseen fraud typologies. Together, these diverse data sources—transactional, behavioral, device-based, and consortium-level—form a multi-dimensional view of financial activity, allowing machine learning algorithms to recognize subtle, evolving, and context-dependent patterns indicative of fraudulent behavior.

**Feature Engineering for Behavior and Devices**

Feature engineering serves as the intellectual core of fraud detection systems, transforming raw, high-volume transactional data into meaningful behavioral indicators. Because financial fraud often manifests as subtle deviations from a customer's habitual patterns rather than isolated anomalies, capturing temporal and contextual nuances is essential.

A fundamental approach involves constructing **rolling aggregates**—statistical summaries calculated per entity (e.g., card, merchant, bank identification number, or IP address) over sliding time windows. Examples include the number of transactions, average spending, and transaction variance over the past hour, day, or week. These features encode both recent activity intensity and temporal clustering that frequently precede fraud bursts.

Complementing these are **Recency, Frequency, and Monetary (RFM)** variables, long utilized in customer analytics. Recency measures the time elapsed since the last transaction, frequency captures how often a customer transacts, and monetary quantifies cumulative spending. Deviations from a user's historical RFM profile may signal compromised credentials or automated attacks.

Temporal representation is refined through **cyclic encodings of time-of-day and day-of-week**, enabling models to capture periodic behavioral rhythms—such as nighttime inactivity for genuine users or uniform temporal spacing typical of bots.

An important personalization layer is achieved by quantifying the **deviation from personal baselines**—for instance, comparing the current transaction's amount or merchant risk to an individual's historical median. Such relative measures enhance model sensitivity to idiosyncratic habits.

Device-related features reinforce behavioral profiling. **Device change scores** track how frequently a customer's device fingerprint or browser configuration shifts across sessions; abrupt or frequent changes often indicate credential stuffing or device emulation. Similarly, **cross-feature interactions**—for example, linking IP risk with transaction amount or combining device type and merchant region—allow the model to recognize complex fraud signatures invisible to univariate analyses.

Finally, certain **proxies for social trust**, such as account tenure, linkage to verified phone numbers, or history of successful high-value transactions, function as stabilizing features that mitigate false positives. When integrated through ensemble models or neural embeddings, these engineered features collectively represent a multidimensional behavioral fingerprint, enhancing the discriminative power of machine learning algorithms while preserving interpretability for human analysts.

**Handling Extreme Class Imbalance**

In financial fraud detection, class imbalance is among the most severe and persistent challenges. Genuine transactions overwhelmingly dominate the dataset, often comprising more than 99.5% of total observations, while confirmed fraudulent instances represent only a minute fraction. This disproportionate distribution causes conventional learning algorithms to be biased toward the majority class, yielding deceptively high accuracy yet poor recall on fraudulent cases—the very instances of greatest business importance.

To mitigate this, practitioners employ **calibrated class weights**, assigning higher misclassification penalties to minority (fraudulent) instances during training. Weighted loss functions compel the model to focus on minority examples without artificially altering data distribution. Another advanced technique, **focal loss**, dynamically scales the loss contribution of well-classified samples, concentrating learning on harder, misclassified cases. This approach is especially effective for gradient boosting and deep learning architectures used in real-time fraud scoring.

Beyond adjusting the learning objective, **threshold tuning** aligns model decision boundaries with business costs rather than raw prediction probabilities. Fraud detection is inherently asymmetric—false negatives (missed fraud) incur direct financial losses, while false positives (incorrectly blocked genuine users) harm customer trust and revenue. Therefore, thresholds are optimized using cost-sensitive metrics derived from expected chargeback value, investigation costs, and customer lifetime value.

Data-level solutions complement these algorithmic methods. **Synthetic oversampling techniques**, such as SMOTE (Synthetic Minority Over-sampling Technique) and its variants

(Borderline-SMOTE, ADASYN), generate synthetic minority examples to balance class proportions. However, to avoid information leakage, oversampling must be restricted to **training folds only**, ensuring test data remains untouched and realistic.

Model performance in such settings cannot be assessed through accuracy alone, which is misleading under heavy imbalance. Instead, specialized evaluation metrics are employed: **Precision-Recall AUC** provides insight into the trade-off between fraud detection rate and false alarms; **F2/Fβ scores** prioritize recall for cost-sensitive detection; **Matthews Correlation Coefficient (MCC)** offers a balanced metric reflecting all confusion matrix components; and **cost curves** visualize the business impact of classification thresholds.

Together, these methods enable fraud detection models to maintain sensitivity to rare fraudulent behavior without sacrificing stability or inflating false positives, ensuring both statistical robustness and operational viability in production environments.

**Temporal Dynamics and Concept Drift**

Fraud detection environments are inherently **non-stationary**, meaning that the statistical properties of the data distribution change over time due to evolving criminal strategies, regulatory interventions, and customer behavior shifts. This phenomenon, known as **concept drift**, undermines the long-term reliability of static models that assume a fixed input–output relationship. In the financial domain, such drifts manifest through new attack vectors (e.g., phishing-induced takeovers), changes in transaction channels, or seasonal spending behaviors that alter baseline statistics. Recognizing and adapting to these temporal dynamics is therefore critical to maintaining detection accuracy and business relevance.

A primary strategy to manage drift is **sliding-window training**, in which models are periodically retrained on the most recent data, effectively "forgetting" outdated patterns. The window size is carefully chosen to balance responsiveness and stability—too short a window increases variance, while too long a window risks incorporating obsolete patterns. Complementary to this approach, **time-aware cross-validation** ensures that training and validation sets respect chronological order, preventing information leakage from future transactions into past predictions. This temporal discipline mirrors real-world deployment, where future labels are inherently unavailable at prediction time.

Quantitative monitoring tools further enhance drift management. The **Population Stability Index (PSI)** measures changes in feature distributions between training and live-scoring populations, flagging shifts that may degrade performance. Similarly, **drift detectors** such as **ADaptive WINdowing (ADWIN)** and **Kolmogorov–Smirnov (K–S) tests** detect significant deviations in data streams or prediction distributions, providing early warnings of concept or data drift.

Operational resilience is strengthened through **shadow models**, which are deployed in parallel with production systems to simulate performance on live data without affecting real decisions. These models help assess emerging trends, evaluate potential retraining triggers, and validate new architectures before full rollout. When performance degradation is detected, **rapid rollback mechanisms** supported by **model registries** enable institutions to revert to previously validated versions, ensuring business continuity and regulatory compliance.

By integrating these temporal adaptation strategies—continuous retraining, drift diagnostics, and shadow deployment—fraud detection systems can evolve dynamically with changing fraud landscapes, preserving both predictive accuracy and operational stability in real-time financial ecosystems.

**Explainability and Model Governance**

In modern financial ecosystems, **explainability** and **model governance** have become integral pillars of responsible artificial intelligence, particularly in high-stakes domains such as fraud detection where algorithmic decisions directly influence customers' access to financial services. Regulators, including those enforcing GDPR's "right to explanation" and banking supervision frameworks (e.g., Basel II/III), increasingly demand transparency, accountability, and interpretability from automated decision systems. Consequently, fraud detection models must not only be accurate but also **explainable, auditable, and aligned with regulatory standards**.

**Explainability operates at two complementary levels—global and local.**

At the **global level**, techniques such as **feature importance analysis** (based on gain, split count, or permutation impact) provide insights into the overarching drivers of model behavior. These metrics help identify which variables—such as transaction frequency, device change rate, or IP risk—most influence fraud predictions. Global interpretability facilitates high-level audits, supports governance committees, and enables non-technical stakeholders to understand the system's rationale.

At the **local level**, explainability focuses on individual predictions. Tools like **SHAP (SHapley Additive exPlanations)** and **LIME (Local Interpretable Model-agnostic Explanations)** attribute prediction outcomes to specific features, enabling analysts to determine why a transaction was flagged as suspicious. **Counterfactual explanations** further enhance interpretability by illustrating minimal input changes required to alter the model's decision—for instance, showing that lowering transaction velocity or matching historical device signatures would have produced a "non-fraudulent" label.

Operationally, these explanations translate into **reason codes** presented to human fraud analysts and case reviewers. Reason codes help prioritize alerts, guide manual verification, and build trust in automated scoring systems. Moreover, financial institutions commonly adopt **challenger–champion frameworks**, wherein a "champion" model serves as the current production standard, while "challenger" models are tested concurrently to compare performance, interpretability, and bias metrics before deployment. This iterative governance mechanism ensures model evolution under controlled conditions.

Finally, robust governance requires comprehensive **documentation and auditability**. Every model must maintain a clear lineage—from data preprocessing to feature selection, hyperparameter tuning, and validation metrics—stored in **model registries** that support reproducibility and rollback. Regular internal audits and external regulatory reviews rely on this documentation to confirm compliance, fairness, and risk transparency.

In essence, explainability and governance transform machine learning models from opaque black boxes into accountable decision-support systems. By embedding interpretability, reason codes, and documented control processes throughout the model lifecycle, financial institutions can achieve both **regulatory trustworthiness and operational transparency**, ensuring that fraud detection systems remain not only powerful but also ethically and legally defensible.

**Cost-Sensitive Decisioning and Thresholding**

In financial fraud detection, model optimization must extend beyond statistical accuracy to incorporate **economic rationality**. Unlike conventional classification tasks, the cost of misclassification in fraud detection is highly asymmetric: a false negative (missed fraud) directly results in monetary loss, whereas a false positive (incorrectly declined legitimate transactions) damages customer trust, increases attrition, and incurs opportunity costs. Therefore, the **threshold setting** process—determining the score boundary between fraud and non-fraud—must be guided by **expected loss modeling** rather than default probability cutoffs.

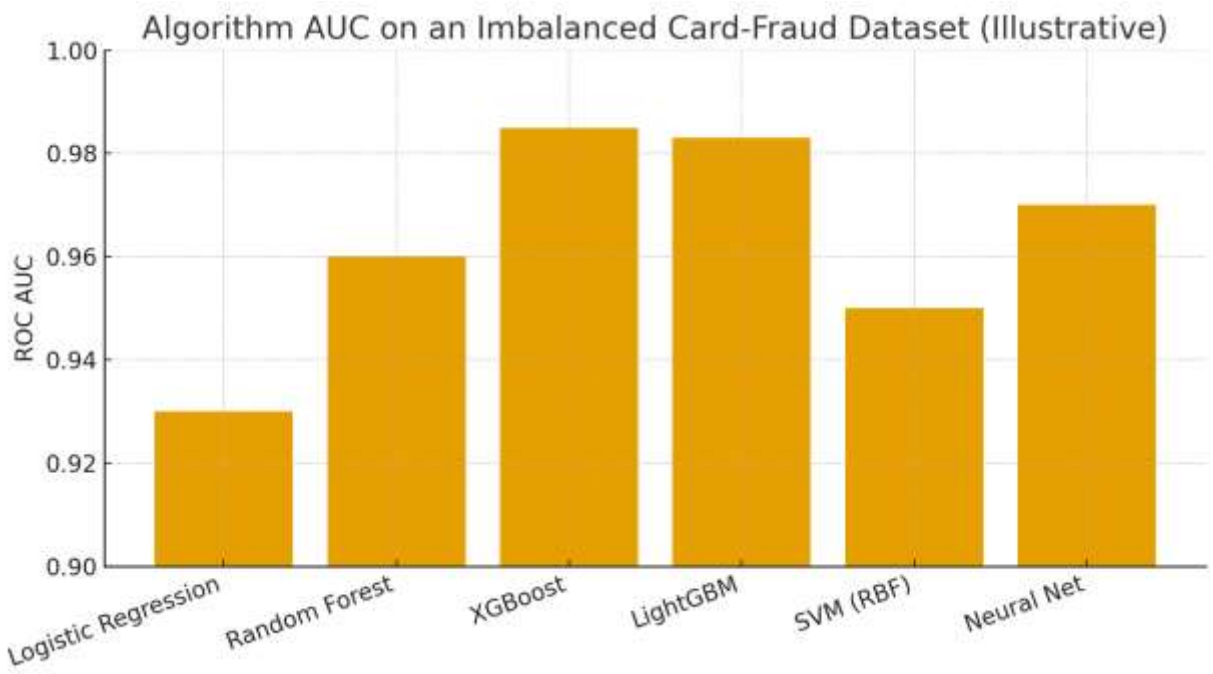
Cost-sensitive learning frameworks address this imbalance by **aligning decision thresholds with the financial impact of outcomes**. Each transaction type and customer segment is assigned an expected cost function encompassing variables such as **interchange fees**, **chargeback liabilities**, **manual review expenses**, and potential **customer churn** from unnecessary declines. The objective is to minimize the total expected loss across all transactions, rather than to simply maximize recall or precision.

institutions implement **reject-option classification**, introducing an intermediate "gray zone" where the model's confidence is low. Instead of issuing an immediate decision, these transactions are routed to manual or semi-automated review queues. This triage strategy enhances both efficiency and customer satisfaction: high-confidence legitimate transactions are automatically approved, clearly fraudulent ones are instantly declined, and only ambiguous cases undergo additional scrutiny.

To further optimize operational throughput, **triage bands**—commonly labeled as *approve*, *review*, and *decline*—are dynamically adjusted based on real-time fraud prevalence and resource availability. Advanced organizations integrate **queue optimization algorithms** that allocate review capacity according to predicted case utility, ensuring that manual analysts focus on the most economically significant or high-risk alerts.

Furthermore, ongoing **threshold calibration** plays a crucial role in adapting to evolving business goals. during high-volume shopping seasons, institutions may prefer to slightly increase approval rates to preserve customer experience, accepting a marginally higher fraud risk that is offset by increased transaction revenue. Conversely, in periods of heightened fraud activity, thresholds can be tightened to reduce losses.

cost-sensitive decisioning transforms fraud detection from a purely technical exercise into a **strategic business optimization problem**, balancing financial exposure, regulatory obligations, and customer satisfaction. By coupling predictive analytics with economic reasoning and adaptive thresholding, organizations can achieve a sustainable equilibrium between risk mitigation and operational efficiency.




## Summary

Machine learning has transformed financial fraud detection from static, rule-based screening to adaptive, data-driven risk decisioning. High-performing solutions combine gradient-boosted trees or graph-based models with cost-sensitive thresholds, robust temporal validation, and human-interpretable explanations. Operational excellence—feature stores, streaming inference, and vigilant monitoring—is as crucial as algorithm choice. Looking ahead, cross-bank collaboration via privacy-preserving learning, standardized governance artifacts, and hybrid rules-plus-ML systems will be central to staying ahead of adversarial innovation while preserving customer experience.

## References

Bolton, R. J., & Hand, D. J. (2002). Statistical fraud detection: A review. *Statistical Science*.

Dal Pozzolo, A., et al. (2015). Calibrating probability with undersampling for unbalanced classification. *IEEE Symposium on CIDM*.

Carcillo, F., et al. (2019). Combining unsupervised and supervised learning in credit card fraud detection. *Information Sciences*.

Jurgovsky, J., et al. (2018). Sequence classification for credit-card fraud detection. *Expert Systems with Applications*.

Bahnsen, A. C., et al. (2016). Cost-sensitive credit card fraud detection using Bayes minimum risk. *IEEE ICDM Workshops*.

Van Vlasselaer, V., et al. (2015). APATE: A novel approach for automated credit card transaction fraud detection using network-based extensions. *Decision Support Systems*.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *KDD*.

Ke, G., et al. (2017). LightGBM: A highly efficient gradient boosting decision tree. *NeurIPS*.

Breiman, L. (2001). Random forests. *Machine Learning*.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*.

He, H., & Garcia, E. (2009). Learning from imbalanced data. *IEEE TKDE*.

Chawla, N. V., et al. (2002). SMOTE: Synthetic minority over-sampling technique. *JAIR*.

Ribeiro, M. T., et al. (2016). "Why Should I Trust You?": Explaining the predictions of any classifier. *KDD*.

Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. *NeurIPS*.

Ghasemi, A., et al. (2020). A survey on credit card fraud detection using ML. *Computer Science Review*.

Aggarwal, C. C. (2015). *Outlier Analysis*. Springer.

Zareapoor, M., & Shamsolmoali, P. (2015). Application of credit card fraud detection: Based on bagging ensemble classifier. *Procedia CS*.

McMahan, H. B., et al. (2017). Communication-efficient learning of deep networks from decentralized data (FedAvg). *AISTATS*.

Sokol, K., & Flach, P. (2020). Explainability fact sheets. *FAccT*.

Ba, J. L., & Kingma, D. P. (2015). Adam: A method for stochastic optimization. *ICLR*.