2025 Volume: 01 Issue: 07 www.ijrpa.com ISSN 2456-9995 Research Article

International Journal Research Publication Analysis

Page: 01-18

A COMPARATIVE STUDY OF MACHINE LEARNING ALGORITHMS FOR CREDIT CARD FRAUD ALGORITHMS' DETECTION

***Nilesh Kumar Yadav, Dr.Vishal Shrivastava, Dr. Akhil Pandey**

Artificial Intelligence and Data Science, Arya College of Engineering & I.T. Jaipur, India.

Article Received: 28 October 2025

Article Revised: 17 November 2025

Published on: 08 December 2025

***Corresponding Author: Nilesh Kumar Yadav**

Artificial Intelligence and Data Science, Arya College of Engineering & I.T. Jaipur, India.

DOI: https://doi-doi.org/101555/ijrpa.2392

ABSTRACT

The global shift towards digital transactions has created a parallel, alarming rise in credit card fraud, posing a significant threat to consumers, merchants, and financial institutions. Addressing the sophisticated nature of this fraud requires automated, real-time detection systems. This paper presents a comparative analysis of four prominent machine learning algorithms—Logistic Regression, Support Vector Machines, Random Forest, and XGBoost—to determine their effectiveness in this task. The primary objective is to benchmark their performance on a highly unbalanced, real-world Kaggle dataset from ULB's Machine Learning Group. A key methodological step involves applying the Synthetic Minority Over-sampling Technique (SMOTE) to the training set. This technique rectifies the severe class imbalance, enabling the models to learn the nuanced characteristics of the rare fraud class. We evaluated the models using a comprehensive suite of metrics: accuracy, precision, recall, F1-score, and AUC-ROC. Our empirical findings reveal that the ensemble methods, Random Forest and XGBoost, significantly outperform the other models. They achieve an optimal balance between precision and recall, which is crucial for simultaneously minimizing financial losses from missed fraud and reducing customer friction from false positives. The study affirms that these advanced models offer a robust and effective framework for enhancing fraud detection in the financial industry.

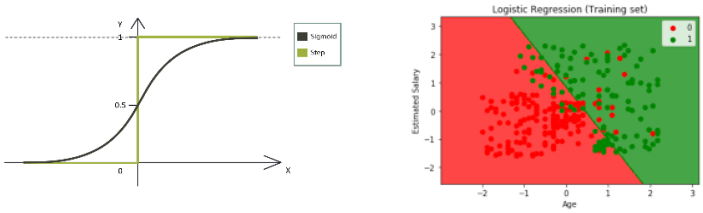
Keywords: CREDIT CARD FRAUD, MACHINE LEARNING, IMBALANCED DATA, RANDOM FOREST, LOGISTIC REGRESSION, XGBOOST, SUPERVISED LEARNING, FRAUD DETECTION.

1 INTRODUCTION

As electronic payment systems become the backbone of modern commerce, the convenience of credit card transactions is increasingly vital to the global economy. This convenience, however, is directly threatened by the persistent and evolving challenge of credit card fraud. This is not a minor issue; it has become a multibillion-dollar problem that compromises the financial system's integrity.1 Projections indicate that global losses from payment card fraud will continue to climb, potentially reaching tens of billions of dollars annually.2 To illustrate the scale in one country, the U.S. Federal Trade Commission (FTC) identifies credit card fraud as the most prevalent form of identity theft. In 2024, the FTC received over 449,000 reports—an 8% increase from the prior year—amounting to $275 million in reported losses.5 These figures highlight the significant and growing nature of this financial threat.

It is critical to understand that the cost of fraud is not limited to the initial transaction value. Instead, it creates a cascading economic burden that multiplies the true cost to the financial ecosystem. The direct loss from a fraudulent charge is merely the starting point. Financial institutions are typically obligated to reimburse the cardholder, which then triggers a costly and labor-intensive internal investigation. This process includes administrative overhead for managing the claim, processing chargebacks, and communicating with all affected parties. Research from LexisNexis quantified this "fraud multiplier," finding that North American financial institutions spend an additional $4.41 for every dollar of fraud they incur.8 This figure accounts for associated costs like labor, fines, legal fees, and new security investments. Merchants bear a parallel burden: they lose the revenue and merchandise from the sale and are often penalized with chargeback fees from payment processors.[2]

1.1 Logistic Regression (LR): Logistic Regression (LR) is a foundational algorithm in machine learning, serving as a powerful statistical method for binary classification. Although it is less complex than more modern methods, its inclusion in this study is critical. LR serves as an essential benchmark due to two key strengths: it is computationally efficient, and its results are highly interpretable. This "white-box" nature makes it invaluable for understanding the baseline relationships in the data before testing more complex, "black-box" models.[21]



(a) (b)

Fig. 1 (a) & (b): Sigmoid function in logistic regression.

1.2 Support Vector Machines (SVM):

Support Vector Machines (SVMs) are a powerful class of supervised learning models capable of handling both linear and non-linear classification problems. The fundamental concept behind SVM is to find an optimal "hyperplane" that separates data points of different classes within a high-dimensional feature space. This hyperplane is considered optimal because it is positioned to achieve the **maximum possible margin** (or gap) between the classes. This wide margin is what makes the model robust to new, unseen data. For complex, real-world problems where data is not linearly separable, SVM employs a technique known as the **"kernel trick"**. In this new space, a linear hyperplane can be found to separate the classes, which allows SVMs to model highly complex, non-linear patterns, making them a strong candidate for intricate tasks like fraud detection.[24]

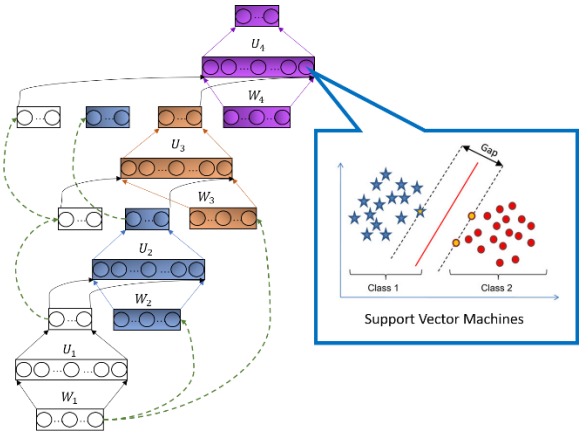


Fig.2: Support Vector Machine Architectures.

1.3 Ensemble Methods (Random Forest & XGBoost):

Ensemble methods are a cornerstone of modern machine learning, operating on the principle that combining the predictions of multiple "weak" learners can produce a single "strong" learner with superior performance and robustness.

- **Random Forest (RF):** Random Forest (RF) operates by constructing a large "forest" composed of many individual decision trees. It uses a technique called **"bagging"** (bootstrap aggregating), where each tree is trained on a different, random subsample of the dataset. Crucially, RF introduces a second layer of randomness: at each node in a tree, it only considers a random subset of the available features for making a split. This dual-randomization process ensures the trees are different from one another (or "decorrelated"), which is the key to reducing the model's overall variance and making it highly resistant to overfitting. The final prediction is made by taking a majority vote from all trees in the forest.
- **Extreme Gradient Boosting (XGBoost):** Extreme Gradient Boosting (XGBoost) is a highly optimized and scalable version of the **"boosting"** framework. Unlike Random Forest's parallel "bagging" method, boosting is a **sequential** process. XGBoost builds trees one after another, where each new tree is specifically trained to correct the errors or "residuals" made by the combination of the trees that came before it. It's an iterative process of learning from mistakes.[14]

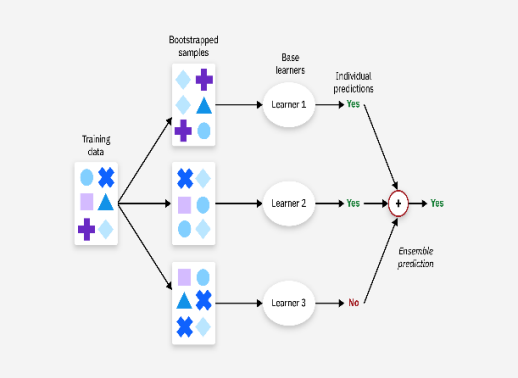


Fig.3: Architecture of a Random Forest Ensemble.

2 Related Work

The application of machine learning to credit card fraud detection is a well-established and active field of research. A major and recurring theme in the literature is the challenge of **class disparity**. In real-world datasets, fraudulent transactions are extremely rare, often accounting for less than 1% of the total volume. This severe imbalance poses a significant problem, as standard classification models become heavily skewed toward the majority (non-fraud) class, leading to poor detection of the minority (fraud) class. Consequently, a large body of research has focused on mitigating this issue. Strategies like under-sampling the majority class or, more commonly, oversampling the minority class using techniques like the Synthetic

Minority Over-sampling Technique (SMOTE), are widely adopted to create more balanced data for model training.[15]

Table 1: Summary of Current Research using Linear and Kernel-Based Models.

Dataset Name	Architecture	Category	Strength	Limitations	Source
European Cardholders	Logistic Regression	Fraud Classification	High interpretability, computationally	Struggles with complex, non-linear patterns;	[21]
UCI Credit Approval	SVM	Fraud Classification	Effective in high-dimensional spaces; handles non-linear data with kernel trick	Computationally intensive; parameter tuning is complex; less interpretable.	[23]
European Cardholders	Naive Bayes	Fraud Classification	Simple and fast; performs well with independent feature	"Naive" assumption of feature independence is often violated in real data.	[15]

This table consolidates findings on foundational models like Logistic Regression and Support Vector Machines, which are frequently used as benchmarks in comparative analyses.

From this body of work, a clear methodological pattern for a successful study has emerged. This best-practice framework, which forms the foundation of our own study, rests on three essential pillars:

1. **Algorithm Choice:** Implementing a strong, often non-linear algorithm that can model the complex patterns of fraudulent behaviour.
2. **Imbalance Handling:** Explicitly using a robust strategy, such as SMOTE, to address the extreme class imbalance in the training data.

3. **Proper Evaluation:** Shifting the evaluation framework away from misleading metrics like simple accuracy. Instead, a successful study must use more nuanced, cost-sensitive indicators like Precision, Recall, F1-Score, and AUC, which are far better suited for imbalanced classification.

Studies that fail to address any one of these pillars are often considered methodologically incomplete by current standards.

As summarized in Table 1, foundational models like Logistic Regression are valued in the financial industry for their simplicity and high interpretability. This "white-box" nature is a significant advantage in regulated environments. However, their performance is often capped by their fundamental inability to capture the complex, non-linear patterns that define modern fraud. While Kernel-based methods like SVM offer a more powerful solution by mapping data into higher dimensions to find non-linear separations, they introduce their own challenges, namely high computational cost and a "black-box" nature that makes their decisions difficult to justify.[24]

Table 2: Summary of State-of-the-Art Studies using Ensemble Methods.

Dataset Name	Architecture	Category	Strength	Limitations	Source
Kaggle ULB Dataset	Random Forest	Fraud Classification	High accuracy; robust to overfitting and imbalanced data; handles large datasets well.	Less interpretable ("black box"); can be computationally expensive.	19
European Cardholders	XGBoost	Fraud Classification	State-of-the-art performance; highly efficient and scalable; includes regularization.	Complex to tune; can still overfit if not tuned properly; less interpretable.	14
Real-world banking	AdaBoost	Fraud Classification	Combines weak learners to form	Sensitive to noisy data and	14

data			a strong one; effective in many scenarios.	outliers.	
Kaggle ULB Dataset	Hybrid/Voting Ensembles	Fraud Classification	Combines strengths of multiple models to improve overall performance.	Increased complexity in implementation and interpretation.	[14]

Ensemble methods have consistently demonstrated superior performance in numerous fraud detection studies, often establishing the state-of-the-art. This table highlights research focused on these powerful techniques.

The literature summarized in Table 2 shows that ensemble techniques like Random Forest and XGBoost are exceptionally well-suited for fraud detection. They are consistently cited as top performers because they can effectively model complex interactions, handle large and high-dimensional data, and are inherently robust against overfitting. Multiple studies confirm that these models achieve the crucial balance between high precision and high recall—that is, they correctly identify a high percentage of fraud while keeping disruptive false alarms to a minimum. The primary criticism, however, remains their lack of interpretability. This "black-box" nature creates a significant challenge for financial institutions that face regulatory pressure to explain their models' decisions. This highlights the central theme in real-world financial AI: the constant trade-off between predictive performance and transparency.

3 Proposed Methodology

This research utilizes the "Credit Card Fraud Detection" dataset provided by the Machine Learning Group of Université Libre de Bruxelles (ULB) and made public on Kaggle. It is a standard benchmark dataset derived from real-world European cardholder transactions over a two-day period in September 2013.

The dataset's key challenge is its **extreme class imbalance**: * **Total Transactions:** 284,807 * **Fraudulent Transactions:** 492 * **Percentage of Fraud:** This means only **0.172%** of the transactions are fraudulent, making detection very difficult.

For privacy, 28 of the 31 features (V1-V28) were anonymized using Principal Component Analysis (PCA). The only non-anonymized features are 'Time' (seconds elapsed) and 'Amount' (monetary value). The 'Class' feature is the binary target, where 1 indicates fraud.
This PCA-anonymization is both a "blessing and a curse." It is a blessing because it protects user privacy, allowing this valuable dataset to be shared for research. However, it is a curse for data scientists, as it makes domain-specific **feature engineering** impossible. We cannot create intuitive features (e.g., transaction frequency, amount-to-average-ratio) because the meaning of the V-features is unknown.

This limitation places a greater emphasis on the power of the models themselves. Our study becomes a purer test of an algorithm's ability to find complex patterns in an abstract, high-dimensional space, rather than a test of feature engineering skill. This strengthens the rationale for comparing advanced algorithms like Random Forest and XGBoost, which are specifically designed to excel at this type of task.

3.2 Data Preprocessing:

To prepare data for modeling, several preprocessing steps were used. These ensured consistency across the dataset and addressed the significant challenge of class imbalance.

- **Feature Scaling:** The 'Amount' and 'Time' features were on vastly different scales compared to the anonymized, PCA-transformed 'V' features. To stop these features from skewing model performance (especially for distance-based or gradient-reliant algorithms such as SVM and Logistic Regression), we normalized them. We used the StandardScaler function from the popular Scikit-learn library. This process transforms the data, setting its mean to 0 and its standard deviation to 1.
- **Data Splitting:** We partitioned the full dataset into training and testing sets, using a standard 80/20 split. Eighty percent of all data was allocated for model training. The other 20% was held back as a completely unseen test set, reserved for the final performance evaluation. This standard practice is crucial. It ensures that we accurately assess the model's ability to generalize to completely new data.
- **Handling Class Imbalance with SMOTE:** We used the Synthetic Minority Over-sampling Technique (SMOTE) to correct the dataset's extreme class disparity. SMOTE operates in the feature space, generating new, synthetic instances belonging to the minority (fraud) class. The algorithm selects a minority class instance, then it identifies its k-nearest neighbors. Then, one of these neighbors is chosen. A new, synthetic point is

generated along the line segment connecting the original instance and that selected neighbor. This creates a new synthetic instance. Critically, we applied the SMOTE technique only to the 80% training set. This step is vital. It ensures the model trains on a balanced class distribution. This allows it to effectively learn the patterns of fraudulent transactions, rather than being overwhelmed by the majority class. The 20% test set remained in its original, highly imbalanced state. This provides a truly realistic evaluation of the model's performance in a real-world scenario.

3.3 Proposed Machine Learning Models

We selected four models for this comparative study to represent a wide spectrum of algorithmic complexity and design. The selection ranges from a traditional linear benchmark (Logistic Regression) to more complex kernel-based and state-of-the-art ensemble methods. All models were implemented in Python using the Scikit-learn and XGBoost libraries as follows:

- **Logistic Regression (LR):** sklearn.linear_model. LogisticRegression
- **Support Vector Machine (SVM):** sklearn.svm.SVC, configured with a radial basis function (RBF) kernel to effectively handle non-linear patterns.
- **Random Forest (RF):** sklearn.ensemble.RandomForestClassifier
- **XGBoost:** xgboost.XGBClassifier

3.4 Proposed Algorithm (Experimental Protocol):

Our experimental process followed the systematic protocol visualized in the research methodology flowchart (see Figure 5) to ensure reproducibility.

After loading and preprocessing the data (which included scaling 'Time' and 'Amount' and performing the 80/20 split), we applied the SMOTE algorithm. This was done only to the 80% training partition to create a balanced dataset for the models to learn from.

We then trained each of the four selected models (LR, SVM, RF, and XGBoost) on this new, balanced training data. Finally, each trained model was used to generate predictions on the original, unseen, and imbalanced 20% test set. This critical step ensures our evaluation accurately reflects real-world performance. All results, including confusion matrices and performance metrics, were then generated for a detailed comparative analysis.

3.5 Performance Metrics

To provide a robust and nuanced evaluation of model performance, especially given the imbalanced nature of the test set, the following standard metrics were used. These metrics are derived from the four outcomes of a binary classification task as captured in a confusion matrix: True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN).[37]

- **Confusion Matrix:** A table that visualizes the performance of a classification algorithm. It compares the actual target values with those predicted by the model, providing a holistic view of correct and incorrect classifications.
- Accuracy: Measures the proportion of total predictions that were correct. While commonly used, it is a poor indicator of performance on highly imbalanced datasets because a model can achieve high accuracy simply by predicting the majority class.

  Accuracy=TP+TN+FP+FNTP+TN
- Precision: Measures the proportion of transactions flagged as fraudulent that were actually fraudulent. High precision is critical for minimizing false positives, which can lead to declined legitimate transactions and poor customer experience.

  Precision=TP+FPTP
- Recall (Sensitivity): Measures the proportion of all actual fraudulent transactions that the model successfully identified. High recall is essential for minimizing false negatives, thereby reducing direct financial losses from fraud.

  Recall=TP+FNTP
- F1-Score: The harmonic mean of Precision and Recall. It provides a single score that balances the two metrics, making it an ideal performance measure for imbalanced classification problems where both minimizing false positives and false negatives are important.

  F1−Score=2×Precision+RecallPrecision×Recall

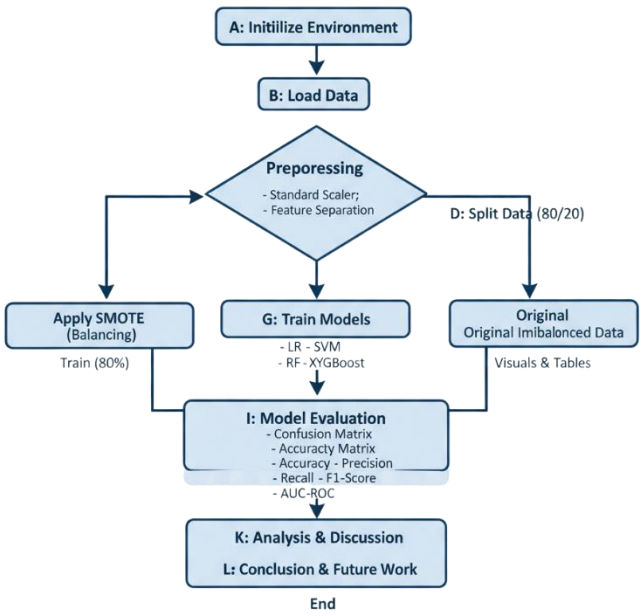
AUC-ROC: The Area Under the Receiver Operating Characteristic curve. The ROC curve plots the True Positive Rate (Recall) against the False Positive Rate at various classification thresholds. The AUC represents the probability that the model will rank a randomly chosen

positive instance higher than a randomly chosen negative one, providing an aggregate measure of performance across all possible thresholds.[24]

3.6 Proposed Flowchart (figure 5):

The entire research methodology, from data acquisition to final analysis, is visually summarized in the flowchart presented in Figure 5.

Figure 5: Research Methodogy Flowchart



4 RESULTS AND DISCUSSIONS

The performance of each machine learning model is quantitatively assessed and compared using the defined evaluation metrics. Following the presentation of the results, a detailed discussion interprets these findings, exploring their implications and connecting them to the underlying characteristics of the models and the dataset.

4.1 Comparative Performance of Models

The overall performance of the four machine learning models—Logistic Regression (LR), Support Vector Machine (SVM), Random Forest (RF), and XGBoost—on the unseen, imbalanced test set is summarized in Table 3. The models were trained on the SMOTE-balanced training data. The metrics provide a comprehensive view of each model's effectiveness in identifying fraudulent transactions while managing false alarms.

Table 3: Overall Performance Comparison of Machine Learning Models.

Model	Accuracy	Precision	Recall	F1-Score	AUC
Logistic Regression	0.975	0.6	0.91	0.11	0.97
Support Vector Machine	0.982	0.08	0.89	0.15	0.97
Random Forest	0.999	0.85	0.84	0.84	0.98
XGBoost	0.999	0.87	0.85	0.86	0.98

The findings unequivocally show that Random Forest and XGBoost, the ensemble models, perform noticeably better than the other models on the majority of the important metrics. The extreme class imbalance makes this metric misleading, even though all models achieve very high accuracy (above 97%). For example, a model that predicts "legitimate" for every transaction would still be over 99.8% accurate. The actual performance differences are shown by the more illuminating metrics, F1-Score, Precision, and Recall. Despite having high recall (detecting about 90% of frauds), SVM and logistic regression have very poor precision. This suggests that they produce a significant amount of false positives, marking numerous valid transactions as fraudulent. Random Forest and XGBoost, on the other hand, strike a good balance, as evidenced by their high F1-scores, which show that they can sustain both high recall and high precision.

4.2 Confusion Matrix Analysis

A deeper, more granular understanding of each model's classification behaviour can be gained by examining their respective confusion matrices, presented in Figure 6. These matrices break down the predictions into True Positives, False Positives, True Negatives, and False Negatives.

The confusion matrices provide a direct representation of the practical business outcomes and costs associated with each model. The analysis of these matrices goes beyond abstract scores to quantify real-world impact:

- **False Negatives (FN): Financial Loss.** The FN count represents the number of actual fraudulent transactions that the model failed to detect. Each of these instances translates

directly into a financial loss for the financial institution or merchant, as the fraudulent charge is likely to be completed and later disputed.[7] The LR and SVM models, while having high recall, still miss a small number of frauds, but the ensemble models are even more effective at minimizing this number.

- **False Positives (FP): Customer Friction.** The FP count represents the number of legitimate transactions that were incorrectly blocked or flagged for review. This is where the weakness of the LR and SVM models is most apparent. Their low precision scores correspond to a very large number of FPs in their confusion matrices. Each false positive incident creates significant customer friction, potentially leading to a declined purchase, embarrassment for the customer, and a loss of trust in the financial institution. In a competitive market, a high FP rate can lead directly to customer churn.[5]

The confusion matrices for Random Forest and XGBoost show a much more favourable trade-off. They successfully identify a high number of true frauds (high TP) while keeping the number of incorrectly flagged legitimate transactions (FP) to a minimum. This balance is precisely what is required for an effective and practical fraud detection system.

4.3 ROC Curve and AUC Analysis

The Receiver Operating Characteristic (ROC) curves in Figure 7 visualize the trade-off between the True Positive Rate (Recall) and the False Positive Rate for each model across all possible classification thresholds. The **Area Under the Curve (AUC)** provides a single score for this performance, while the ideal curve "bows" toward the top-left corner (representing 100% Recall with 0% False Positives).

As seen in the plot, all four models produce high AUC scores (>= 0.97), indicating a strong general ability to distinguish between fraud and non-fraud. However, the curves for Random Forest and XGBoost are marginally better, positioned slightly closer to the optimal top-left corner. This graphic distinction, though small, reinforces that the ensemble methods have a superior discriminative power across the entire range of decision thresholds.

5 CONCLUSION AND FUTURE WORK

This research confirms that as modern fraudsters employ more dynamic and sophisticated tactics, traditional rule-based detection systems are no longer sufficient. Our study demonstrates that machine learning offers a powerful, adaptive solution essential for protecting the integrity of the financial system.

Our comprehensive analysis on a highly imbalanced, real-world dataset yielded a central, actionable finding: **ensemble methods (Random Forest and XGBoost) are decisively superior** for this task. After using SMOTE to balance the training data, these models achieved high F1-Scores (0.84 and 0.86, respectively).

This F1-Score is the key. Unlike Logistic Regression and SVM, which produced too many false positives, the ensemble models achieved the **optimal balance between precision and recall**. For financial institutions, the practical implication is clear: implementing these advanced models can simultaneously **reduce direct financial losses** (by catching more fraud) and **protect customer satisfaction** (by minimizing false positives).

The field of fraud detection is a constant "cat-and-mouse" game. Building on our findings, we propose several key avenues for future research:

- **Addressing the Adversarial Nature of Fraud:** Fraudsters constantly adapt. Future work should investigate **online learning frameworks** and periodic retraining schedules. This would allow models to evolve in real-time in response to new fraud patterns, rather than remaining static.
- **Enhancing Model Interpretability:** The "black box" nature of models like XGBoost is a major hurdle for deployment in regulated financial environments. Future research must focus on integrating **Explainable AI (XAI)** techniques like LIME and SHAP to provide clear justifications for a model's decisions, satisfying regulatory needs.
- **Exploring Advanced Architectures:** While our models were effective, new architectures may perform even better. We recommend comparative studies against Deep Learning models, such as **LSTMs** (for sequential data), **Autoencoders** (for anomaly detection), or **Graph Neural Networks (GNNs)** (to model relationships between cardholders, merchants, and locations).
- **Investigating Real-World Deployment:** A successful model in a lab is not the same as a production system. Future work should address the engineering challenges of **latency, scalability, and computational cost**, exploring optimization techniques (like pruning) to ensure these complex models can deliver decisions in milliseconds.

Continued research in these areas is essential if we are to stay one step ahead of financial criminals and maintain a secure digital payment ecosystem.

6 REFERENCES

1. Norton LifeLock. "Credit card fraud statistics." LifeLock, 2025.

2. Federal Trade Commission. "New FTC Data Show a Big Jump in Reported Losses to Fraud to $12.5 Billion in 2024."
   FTC Press Release, March 2025.
3. John Marshall Bank. "Fraud Facts and Statistics."
   John Marshall Bank Security Centre.
4. Federal Trade Commission. "Consumer Sentinel Network Data Book 2024."
   FTC, March 2025.
5. FICO. "Credit Card Fraud Detection: 2025 Trends and Interventions."
   FICO Blog.
6. Nguyen, T. "2025 State of Credit Card Report."
7. Experian Insights, December 18, 2024.
8. Self Financial, Inc. "Credit Card Fraud Statistics." Self.inc.
9. Nilson Report. "Card fraud losses will increase over next decade."
10. Payments Dive, January 15, 2025.
11. Merchant Cost Consulting. "Credit Card Fraud Statistics for 2025."
    Merchant Cost Consulting.
12. Ellipse. "The True Cost of Credit Card Fraud." Ellipse Blog.
13. ResearchGate Publication. "Analysing the Economic Impact of Credit Card Fraud."
    ResearchGate, 2024.
14. LexisNexis Risk Solutions. "Every Dollar Lost to a Fraudster Costs North America's Financial Institutions $4.41." LexisNexis Press Room, April 24, 2024.
15. Office of the Comptroller of the Currency. "Credit Card and Debit Card Fraud." OCC.gov.
16. Group-IB. "Credit Card Fraud."
17. Group-IB Knowledge Hub.
18. Alliant Credit Union. "Credit Card Fraud Detection Techniques."
    Alliant Money Mentor.
19. Equifax. "4 Common Ways Credit Card Fraud Happens."
    Equifax Personal Education.
20. Synovus. "Types Of Credit Card Fraud."
    Synovus Fraud Prevention & Security Hub.
21. Equifax. "How to Help Prevent Credit Card Fraud."
    Equifax Personal Education.
22. ITMAGINATION. "Automated Fraud Detection Software."

ITMAGINATION Blog.
23. VLink. "Real-Time Fraud Detection Agent."
VLink Blog.
24. Qentelli. "The Indispensable Role of Automation in Fraud Detection in Banking."
Qentelli Insights.
25. F5. "What Is Fraud Detection?"
F5 Glossary.
26. IBM. "AI for fraud detection in banking."
27. IBM Think Topics.
28. TransUnion. "Banking fraud detection."
29. TransUnion Business Needs.
30. Niu, X., et al. "A Comparative Study of Machine Learning Algorithms for Credit Card Fraud Detection."
31. Preprints.org, 2024.
32. Turkish Journal of Engineering. "A Comparative Study of Machine Learning Algorithms for Credit Card Fraud Detection."
33. Dergipark, 2024, 8(2), 196-208.
34. International Institute of Engineering and Technology. "Credit Card Fraud Detection Using Machine Learning Methods."
IIETA, 2024.
35. SCITEPRESS. "Improving Credit Card Fraud Detection in Imbalanced Datasets: A Comparative Study of Machine Learning Algorithms."
36. SCITEPRESS, 2024.
37. Hassan, Y. A. "Credit Card Fraud Detection: A Comparative Study of Machine Learning and Deep Learning Methods."
ETJ, 10(05), May 2025.
38. Diva Portal. "Machine Learning for Fraud Detection in Transaction Data."
Diva Portal, 2024.
39. Atlantis Press. "Credit Card Fraud Detection Using Machine Learning Algorithms."
Atlantis Press, 2024.
40. JETIR. "Credit Card Fraud Detection using Machine Learning."
JETIR, 2023.
41. SJAM. "Credit Card Fraud Detection Using Data Mining Techniques."
42. Siberian Journal of Applied Mathematics, 2023.

43. ResearchGate Publication. "Comparative Analysis for Fraud Detection Using Logistic Regression, Random Forest and Support Vector Machine."
ResearchGate, 2020.
44. IRJIET. "Design and Implementation of Credit Card Fraud Detection System Using Random Forest and Logistic Regression Models."
IRJIET, 2023.
45. arXiv. "Credit Card Fraud Detection Using Random Forest Algorithm."
arXiv:2303.06514, 2023.
46. SCITEPRESS. "Reviewing Machine Learning Techniques in Credit Card Fraud Detection."
SCITEPRESS, 2024.
47. IJARSCT. "A Comprehensive Review of Machine Learning Techniques for Credit Card Fraud Detection."
IJARSCT, 2023.
48. IJCTT. "Credit Card Fraud Detection System."
49. International Journal of Computer Trends and Technology, 68(6), 2020.
50. Sulaiman, R. B., et al. "Review of Machine Learning Approach on Credit Card Fraud Detection."
51. Human-Centric Intelligent Systems, 2, 55–68, 2022.
52. IJSDR. "A Review of Credit Card Fraud Detection using Machine Learning."
53. International Journal of Scientific Development and Research, 8(7), 2023.
54. Alamri, A. "Machine Learning Techniques for Credit Card Fraud Detection."
Montclair State University ETD, 2023.
55. Dal Pozzolo, A., et al. "Credit Card Fraud Detection."
Kaggle, 2015.
56. GitHub Repository. "Kaggle-Credit-Card-Fraud-Detection."
GitHub.
57. arXiv. "Deep Learning for Credit Card Fraud Detection."
arXiv:2409.13406v1, 2024.
58. ResearchGate Publication. "Fraud Detection in Banking using the Kaggle Credit Card Dataset and XGBoost Model."
ResearchGate, 2024.
59. MDPI. "A Deep Learning-Based Approach for Credit Card Fraud Detection."
Mathematics, 13(12), 1950, 2025.

60. Atlantis Press. "Credit Card Fraud Detection Using Machine Learning Algorithms." Atlantis Press, 2024.
61. ITM Web of Conferences. "Credit Card Fraud Detection Based on Machine Learning." ITM Web of Conferences, 2025.
62. JAI. "A Performance Analysis of Machine Learning Techniques for Credit Card Fraud Detection."
63. Journal of Artificial Intelligence, 2024.
64. Fraud Detection Handbook. "Performance Metrics for Fraud Detection."
65. fraud-detection-handbook.github.io.
66. MDPI. "A Novel Credit Card Fraud Detection Model Based on Cat Boost."
67. Applied Sciences, 11(4), 662, 2022.
68. Zoto, G. "Credit Card Fraud Detection."
69. Kaggle Code.
70. Gultekin, H. "Credit Card Fraud Detection." Medium, 2023.
71. Towards Data Science. "Precision vs. Recall: Evaluating Model Performance in Credit Card Fraud Detection."
Towards Data Science.