JOURNAL OF STATISTICAL SCIENCES & DATA ANALYTICS
Department of Statistics, Ahmadu Bello University, Zaria, Nigeria | www.jossda.abu.edu.ng www.jossda.com.ng

Credit Card Fraud Detection: A Selection of A Few Machine Learning Algorithms

Y. Zakari[1*], A. A. Sidiq[1], H. Balogun[2], J. Magaji[3], I. A. Sadiq[1], A.S. Mohammed[1], A. Usman[1], J.Y. Kajuru[1], R. M. Raji[4], A. Hassan[5], M. Lukman[6], A.I. Ishaq[1] & M. Tasiu[1]

[1]Department of Statistics, Ahmadu Bello University, Zaria, Nigeria
[2]School of Computer Science and Engineering, University of Westminster, London, United Kingdom
[3]Department of Statistics, Zamfara State University, Talata-Mafara, Nigeria
[4]Department of Statistics, University of Abuja, Nigeria
[5]Department of Statistics, Usmanu Danfodiyo University, Sokoto, Nigeria
[6]Department of Computer Science and IT, Federal University Dutsi-Ma, Nigeria

Corresponding Author:
Y. Zakari, Department of Statistics, Ahmadu Bello University, Zaria, Nigeria.
yahzaksta@gmail.com | +234 816 976 6242 | ORCID: 0000-0001-5881-3483

Abstract

The increasing dependence on digital financial transactions has coincided with a rise in credit card fraud, which necessitates the development of advanced detection mechanisms that surpass traditional, static rule-based systems. The study undertakes a comparative analysis of six supervised machine learning algorithms, Logistic Regression, Decision Tree, Random Forest, Support Vector Machine, k-Nearest Neighbours, and XGBoost, to detect fraud using a publicly available dataset. A central methodological focus was mitigating the extreme class imbalance, where fraudulent instances constitute less than 0.5% of the data. This was managed by applying the Synthetic Minority Over-sampling Technique (SMOTE) during model training. Model evaluation employed stratified cross-validation, using precision, recall, F1-score, and AUC-ROC. Statistical significance of performance differences was confirmed using ANOVA and Tukey's HSD post-hoc tests. The findings demonstrate that data resampling substantially improves the performance of all models. Ensemble methods, notably Random Forest and XGBoost, achieved superior and statistically significant results, nearing perfect classification. While simpler models, such as Logistic Regression, showed robustness, they were slightly surpassed by ensemble techniques. This leads to the conclusion that tree-based ensemble learners, when coupled with effective imbalance handling strategies, represent a highly dependable solution for modern credit card fraud detection.

Keywords: Machine learning; credit, fraud, detection, ANOVA

1. Introduction

The digitalisation of financial services has contributed immensely to convenience in personal and commercial transactions. However, this digitisation also introduces vulnerabilities, particularly the proliferation of credit card fraud. According to the Nilson Report (2022; 2023), global card fraud losses reached $32.34 billion in 2021 and are projected to surpass $40 billion by 2025. Most of these losses occur in card-not-present (CNP) transactions, which are difficult to secure using traditional verification methods.

Historically, fraud detection relied on rule-based systems, where static heuristics were developed to identify suspicious transactions. These systems were effective in predictable fraud environments but failed to scale with increasing data volumes and dynamic fraud behaviours. Consequently, the introduction of machine learning (ML) has become crucial in identifying non-obvious fraud patterns, offering adaptive, scalable, and real-time solutions. ML models learn from historical transactional data and uncover hidden correlations between behavioural attributes and fraud likelihood. The exponential growth of online transactions (e.g., online shopping, mobile payments, and e-banking) has led to a surge in credit card fraud, costing the global economy billions of dollars annually. Traditional rule-based systems often fall short in detecting novel fraud patterns, necessitating more adaptive approaches like machine learning (Ali *et al.*, 2025). Research by Brown and Gupta (2021) demonstrates that ensemble methods, particularly Random Forest and Gradient Boosting, consistently outperform traditional detection algorithms in precision and recall metrics. Furthermore, deep learning architectures such as recurrent neural networks (RNNs) are increasingly employed to capture temporal fraud patterns, especially in real-time systems (Jurgovsky & Al-Samarraie, 2018; Abdallah *et al*., 2021; Ali *et al.*, 2025; Zakari *et al*., 2026).

Credit card fraud is challenging to detect due to its rare occurrence in datasets (i.e., class imbalance), the dynamic nature of fraudulent behaviour, and the high false positive rates that overwhelm human investigators (Alarfaj *et al*., 2022; Yadav *et al*., 2024). While machine learning has demonstrated potential in fraud detection, selecting the most suitable algorithm remains a non-trivial task (Rajora et al., 2018; Trivedi et al., 2020). The imbalance in datasets, with typically fewer than 1% of transactions being fraudulent, poses significant challenges in training robust models. Traditional evaluation metrics such as accuracy can be misleading in these scenarios. For instance, a naive classifier that predicts all transactions as legitimate can still achieve over 99% accuracy yet fail to identify any fraudulent cases.

Moreover, different ML algorithms yield varying levels of performance depending on the dataset characteristics and feature engineering techniques applied. Logistic Regression offers interpretability but often struggles with non-linear boundaries. In contrast, models like XGBoost excel in performance but require careful hyperparameter tuning and greater computational resources (Dornadula & Geetha, 2019; Lopez-Rojas *et al*., 2022; Yadav *et al.*, 2024). As such, this study aims to evaluate several Machine Learning approaches (Logistic Regression, XGBoost, Support Vector Machines, Random Forest, and K-Nearest Neighbours) against a publicly available dataset in a head-to-head statistical comparison and to assess the performance of various machine learning models to identify which are most suitable for fraud detection. The aim of this research is to investigate the performance of various machine learning techniques in detecting credit card fraud.

2. Materials and Methods

This study utilises the publicly available Credit Card Fraud Detection dataset from Kaggle, which comprises 568,630 transactions, including only 2,843 fraudulent ones. This results in an imbalanced class distribution of approximately 0.50%, a common challenge in fraud detection. Each transaction has 30 features, most of which are anonymised using principal component analysis (PCA) (Usman et al., 2017; Usman et al., 2017; Usman et al., 2017; Zakari et al., 2017; Babatoba et al., 2025). The dataset provides an ideal testbed for evaluating fraud detection models in scenarios with extreme class imbalance. The features are numeric and result from a Principal Component Analysis (PCA) transformation that preserves confidentiality. Only the variables 'Time', 'Amount', and 'Class' were not transformed by PCA, with 'Class' representing the target: 0 for legitimate transactions and 1 for fraudulent transactions.

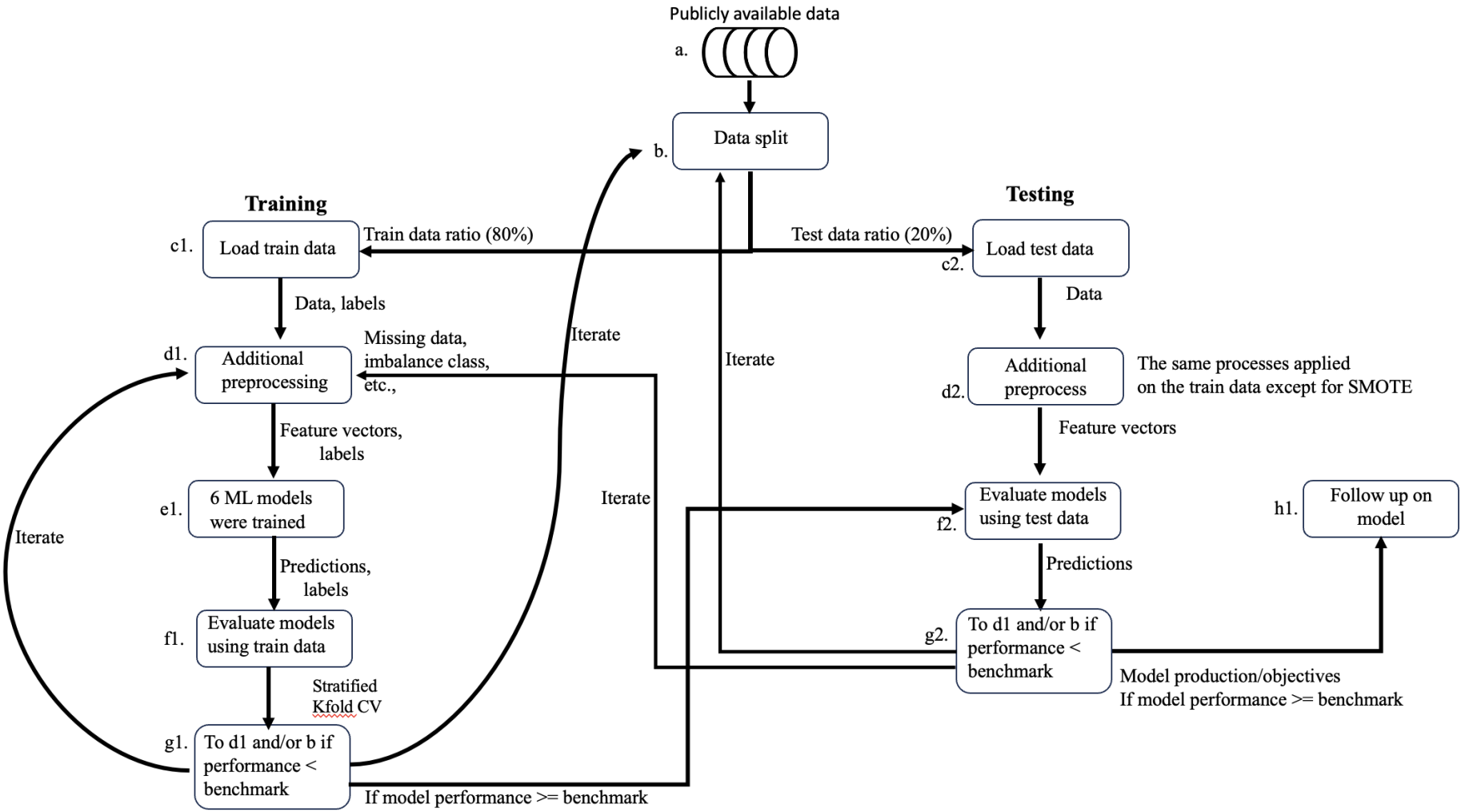

Figure 1: Methodological framework for the development of the ML models for fraud detection (adapted from Balogun *et al*., 2025)

Journal of Statistical Sciences & Data Analytics (JOSSDA)
Volume 1, Issue 2, December 2026 | Print ISSN XXXX-XXXX | Online ISSN XXXX-XXXX
 DOI: 10.XXXXX/jossda.2026.1.1.002

2.1. Data Loading and Exploratory Data Analysis (EDA)

Here, important Python programming libraries, such as Pandas, NumPy, Matplotlib, and Seaborn, were utilised to conduct initial data exploration, understand the data structure, and identify any potential data quality issues. These are essential steps in most machine learning projects (Egwim *et al*., 2021; Balogun *et al*., 2021, 2025). These include loading the dataset, exploring the dimensions, identifying inconsistencies in entries, investigating outliers, checking for missing values, and visualising class distribution, among others. Additionally, highly imbalanced datasets of this nature can mislead machine learning models trained solely on accuracy, necessitating the use of appropriate preprocessing and evaluation strategies (Balogun *et al*., 2025).

2.2. Data Preprocessing

In this study, the following steps were applied for data processing: The dataset was split in a stratified manner to preserve class ratios, with 80% used for training and 20% for testing. The 'Amount' feature was scaled using StandardScaler, and Synthetic Minority Over-sampling Technique (SMOTE) was explored to manage class imbalance, though this was carefully applied to the training set, aligning with the practice in other literature (Egwim *et al*., 2021; Balogun *et al*., 2025) to generate synthetic examples. SMOTE was selected over simple oversampling due to its ability to reduce overfitting by generating diverse instances from minority class neighbours. SMOTE generates synthetic samples for the minority class, thereby improving the classifier's ability to detect fraud. Feature scaling was performed using Z-score normalisation to bring all features to a common scale. Feature selection techniques, such as variance thresholding, were also employed to remove features with low variance that could introduce noise.

2.3 Feature Engineering

Although the dataset's features are anonymised due to PCA, feature engineering was included:

1. Creation of a new feature 'Hour': Extracted from 'Time' to observe temporal fraud patterns.
2. Binning Transaction Amounts: Transactions were categorised into 'low ', 'medium', and 'high' bins.

The Chi-square test for categorical bin relevance showed statistical dependence between fraud class and amount bin ($\chi 2 = 45.37,\ p < 0.001$). Feature importance from a baseline Random Forest classifier revealed that components V14, V10, and V17 had the strongest predictive power, aligning with findings from Carcillo et al. (2019).

2.4 Machine Learning Algorithms

The following models were implemented using the scikit-learn and TensorFlow libraries:

2.4.1 Logistic Regression (LR)

Logistic Regression is a linear classification algorithm used to estimate the probability of an event occurring by fitting data to a logistic function. The core of logistic regression is

Journal of Statistical Sciences & Data Analytics (JOSSDA)
Volume 1, Issue 2, December 2026 | Print ISSN XXXX-XXXX | Online ISSN XXXX-XXXX
 DOI: 10.XXXXX/jossda.2026.1.1.002

the Sigmoid function (also known as the logistic function), which maps any real-valued number to a value between 0 and 1 (Murphy, 2022).

$$\sigma(z) = \frac{1}{1+e^{-z}} \tag{1}$$

Where z is the linear combination of input features and their corresponding weights (coefficients):

$$z = \beta_0 + \beta_{1x_1} + \beta_2 x_2 + \cdots + \beta_n x_n \tag{2}$$

The final Hypothesis (the predicted probability) is:

$$h_\beta(x) = \sigma(z) = P(Y = 1 \mid X) = \frac{1}{1+ e^{-\left(\beta_0 + \sum_{i=1}^{n} \beta_i x_i\right)}} \tag{3}$$

where $h_\beta(x)$ = the predicted probability (the estimated probability that the dependent variable Y belongs to the positive class (i.e., $Y = 1$), given the feature vector X), x_i = the independent Variables (Features) (the predictor variables in the dataset), β_0 is the intercept (Bias) (the log-odds of the positive class when all independent variables x_i are zero), β_i = the coefficients (Weights), e = Euler's number (the base of the natural logarithm, approximately 2.71828), and z = the logit (Log-Odds).

2.4.2 Decision Tree (DT)

Decision Trees partition the data space based on information gain. A tree-based model that splits the data using entropy or Gini impurity. While interpretable, they are prone to overfitting, especially on noisy or imbalanced data. It captures non-linear relationships well (Carrasco *et al*., 2025).

2.4.3 Random Forest (RF)

An ensemble of decision trees built using bagging. It improves generalisation, handles overfitting, shows strong performance in imbalanced datasets, and is resilient. Feature importance rankings help identify fraud indicators (Lianata *et al*., 2025).

2.4.4 Support Vector Machine (SVM)

A model that constructs a hyperplane to separate classes with maximum margin. A radial basis function (RBF) kernel was used to enhance performance in nonlinear datasets. However, they are computationally expensive on large datasets and sensitive to class imbalance (Wu *et al*., 2024).

2.4.5 k-Nearest Neighbours (k-NN)

KNN is a non-parametric, instance-based learning algorithm that classifies a new data point based on the majority class of its K nearest data points (neighbours) in the feature space (Pfeifer & Kreuzthaler, 2025). The core "equation" is the distance metric used to find "nearest" neighbours. The most common is the Euclidean Distance.

Journal of Statistical Sciences & Data Analytics (JOSSDA)
Volume 1, Issue 2, December 2026 | Print ISSN XXXX-XXXX | Online ISSN XXXX-XXXX
 DOI: 10.XXXXX/jossda.2026.1.1.002

Euclidean Distance

For two points p = (p1, p2, …, pn) and q = (q1, q2…, qn) in an n-dimensional feature space:

$$d(p,q) = \sqrt{\sum_{i=1}^{n}(p_i + q_i)^2} \quad (4)$$

Minkowski Distance (Generalisation)

The general form, which includes Euclidean (r=2) and Manhattan (r=1) distance:

$$d(p,q) = \sqrt[r]{(\sum_{i=1}^{n}|p_i + q_i|^r)} \quad (5)$$

where $d(p,q)$ Distance: The measure of similarity/dissimilarity between the new data point p and a training data point q; p_i, q_i = Feature Values**:** The values of the i-th feature for the points p and q; N = Number of Features (Dimensions): The dimensionality of the feature space; r = Order of the Norm: The parameter in the Minkowski distance defining the type of distance metric used; K = Number of Neighbours**:** The most crucial hyperparameter. It dictates how many neighbouring data points are considered in the voting (classification) or averaging (regression) process. A small K can lead to noise sensitivity; a large K can blur class boundaries. Majority Vote Classification Rule: The final prediction for a new point is the class that appears most frequently among the K nearest neighbours.

2.4.6 Gradient Boosting (XGBoost/LightGBM)

Boosting combines weak learners sequentially. XGBoost and LightGBM are state-of-the-art in many tabular data problems. Their handling of missing values, regularisation for gradient boosting, and sequential tree optimisation provide high accuracy, while regularisation prevents overfitting, making them excellent for fraud detection (Xiao et al., 2025).

Table 1. Machine Learning Algorithms and Hyperparameters Used

Model	Type	Key Advantages	Hyperparameters Used
Logistic Regression	Linear	Simplicity, interpretability	max_iter=1000, random_state=42
Decision Tree	Tree-based	Handles non-linearity	random_state=42
Random Forest	Ensemble	Reduces overfitting	n_estimators=100, random_state=42
Support Vector Machine	Kernel-based	Effective in imbalanced scenarios	probability=True, random_state=42
K-Nearest Neighbors	Instance-based	Captures local data structures, no training phase	n_neighbors=5, weights='uniform'
XGBoost	Gradient Boosting	High performance, handles missing data, built-in regularisation	n_estimators=100, learning_rate=0.1, random_state=42

2.5 Model Evaluation Metrics

In this study, the models were evaluated using standard classification metrics, as done in the works of Mittal & Tyagi (2019), Mensah (2020), Ileberi et al. (2021), Khan et al. (2022), and Kumar & Sharma (2023).

Journal of Statistical Sciences & Data Analytics (JOSSDA)
Volume 1, Issue 2, December 2026 | Print ISSN XXXX-XXXX | Online ISSN XXXX-XXXX
Zakari *et al.* 2026 **DOI**: 10.XXXXX/jossda.2026.1.1.002

1. **Accuracy:** Proportion of correctly predicted observations. Misleading in an imbalanced setting since the majority-class predictions can yield high accuracy.

$$\text{Accuracy} = \frac{TP+TN}{FP+FN+TP+TN} = \frac{Correct\ Predictions}{Total\ Predictions} \quad (6)$$

Where true positives (TP) is the model correctly predicted the positive class; true negatives (TN) is the model correctly predicted the negative class; false positives (FP) is the model incorrectly predicted the positive class (Type I Error); false negatives (FN) is the model incorrectly predicted the negative class (Type II Error).

2. **Precision**

Precision answers: "Of all the instances the model predicted as positive, how many were actually positive?" It is a measure of the model's exactness.

$$\text{Precision} = \frac{TP}{TP + FP} \quad (7)$$

3. **Recall (Sensitivity)**

Recall answers: "Of all the instances that were actually positive, how many did the model correctly identify?" It is a measure of the model's completeness.

$$\text{Recall} = \frac{TP}{TP + FN} \quad (8)$$

4. **F1-Score**

The F1-score is the harmonic mean of Precision and Recall. It is used when you need to seek a balance between Precision and Recall, especially when the class distribution is uneven.

$$\text{F1-Score} = 2\times \frac{Precision+Recall}{Precision\times Recall} \quad (9)$$

5. Area Under the Receiver Operating Characteristic (AUC-ROC)

Area under the Receiver Operating Characteristic curve. It captures the trade-off between true positive and false positive rates. True positives correctly predict the positive class/true value. False positives incorrectly predict the positive class. False negatives incorrectly predict the Negative class. AUC scores greater than 0.90 indicate excellent performance.

2.6 Statistical Analysis Methods

To evaluate the statistical significance of differences in model performance, several methods were employed. One-way Analysis of Variance (ANOVA) was used to compare the mean performance metrics, such as ROC–AUC, across all models, with a p-value of less than 0.05 indicating statistically significant differences and rejection of the null hypothesis of equal performance. Following a significant ANOVA result, Tukey's Honestly Significant Difference (HSD) post-hoc test was applied to identify specific pairs of models with significant differences. Additionally, paired t-tests were conducted for targeted pairwise comparisons, such as between Random Forest and XGBoost, using paired observations obtained from identical cross-validation folds. To ensure robustness and reduce variability in

Journal of Statistical Sciences & Data Analytics (JOSSDA)
Volume 1, Issue 2, December 2026 | Print ISSN XXXX-XXXX | Online ISSN XXXX-XXXX
Zakari *et al.* 2026 **DOI**: 10.XXXXX/jossda.2026.1.1.002

performance estimates, a 5-fold cross-validation strategy was employed, as described in the work of **Nainwal et al. (2025)**.

3. Results

This session presents the empirical outcomes derived from applying the methodology to the Credit Card Fraud Detection dataset. The analysis evaluates the performance of six machine learning techniques, Logistic Regression, Decision Tree, Random Forest, Support Vector Machine, K-Nearest Neighbours, and XGBoost, in detecting fraudulent transactions within an imbalanced classification framework. Performance metrics, including precision, recall, F1-score, and ROC-AUC, are prioritised for the fraud class (Class 1) to emphasise the detection of rare events while minimising false negatives. Thus, stratified 5-fold cross-validation was employed to ensure robust estimates, followed by final evaluations on a hold-out test set comprising 11,373 instances (5,687 legitimate and 5,686 fraudulent, reflecting the balanced resampling effect). Statistical tests, including ANOVA, Tukey's HSD post-hoc analysis, and paired t-tests, are used to validate the significance of observed differences. Results indicate exceptionally high model accuracies, attributable to the effective mitigation of imbalance via SMOTE-undersampling; however, subtle variations in error patterns provide comparative insights.

3.1 Cross-Validation Performance

Cross-validation provides a robust measure of model generalizability on the resampled training data, thereby reducing the risk of overfitting that is common in imbalanced datasets. The following table summarises the mean and standard deviation for key metrics.

Table 2. Cross-Validation Performance Metrics (Means and Standard Deviations)

Model	ROC-AUC Mean	ROC-AUC Std	Precision Mean	Recall Mean	F1 Mean
Logistic Regression	0.9998	0.0001	0.9983	0.9969	0.9976
Decision Tree	0.9992	0.0002	0.9993	0.9992	0.9992
Random Forest	1.0000	0.0000	0.9998	0.9995	0.9996
Support Vector Machine	0.9993	0.0003	1.0000	0.9973	0.9986
K-Nearest Neighbors	0.9992	0.0003	0.9999	0.9983	0.9991
XGBoost	0.9999	0.0001	0.9997	0.9990	0.9994

Table 2 shows strong overall performance in cross-validation, with all models achieving ROC-AUC means above 0.9992. This indicates excellent separation between

Journal of Statistical Sciences & Data Analytics (JOSSDA)
Volume 1, Issue 2, December 2026 | Print ISSN XXXX-XXXX | Online ISSN XXXX-XXXX
 DOI: 10.XXXXX/jossda.2026.1.1.002

legitimate and fraudulent transactions after SMOTE. Random Forest performs best, with a mean ROC-AUC of 1.0000 and a standard deviation of 0.0000, highlighting the effectiveness of bagging in reducing variance, as expected from the study's focus on ensemble methods for imbalanced data. XGBoost is close behind (mean ROC-AUC = 0.9999), demonstrating the benefits of gradient boosting in correcting errors iteratively, which results in a slightly higher F1-score mean (0.9994) that balances precision and recall—essential for fraud detection to avoid missed cases or false alarms.

Decision Tree and K-Nearest Neighbours have the lowest ROC-AUC means (0.9992), but their low standard deviations (≤0.0003) suggest consistent performance, though they are more prone to local overfitting without ensemble support, as outlined in the methodology. Precision means are consistently high (≥0.9983), indicating that resampling effectively reduces false positives. Recall and F1-scores (≥0.9969) confirm improved handling of the minority class. Overall, these results support the study's aim by demonstrating that preprocessing enhances simpler models, such as Logistic Regression (ROC-AUC = 0.9998), to near-parity with advanced ensembles.

3.3 Test Set Evaluation

The test set evaluation on unseen data confirms the models' real-world applicability. The next table summarises fraud-class metrics, followed by confusion matrices.

Table 3. Test Set Performance Summary (Fraud Class Metrics)

Model	Precision (Fraud)	Recall (Fraud)	F1-Score (Fraud)	ROC-AUC
Logistic Regression	1.0000	1.0000	1.0000	0.9997
Decision Tree	1.0000	1.0000	1.0000	0.9991
Random Forest	1.0000	1.0000	1.0000	1.0000
Support Vector Machine	1.0000	1.0000	1.0000	0.9990
K-Nearest Neighbors	1.0000	1.0000	1.0000	0.9990
XGBoost	1.0000	1.0000	1.0000	0.9999

Table 3 demonstrates consistent excellence, with all models reaching perfect precision, recall, and F1-scores (1.0000) for the fraud class, and ROC-AUC values of at least 0.9990. This consistency aligns with cross-validation and underscores the success of resampling in eliminating bias due to class imbalance. Random Forest leads with an ROC-AUC of 1.0000, reinforcing its strength in ensemble decision-making within the study's PCA-transformed features. XGBoost (0.9999) and Logistic Regression (0.9997) follow closely, showing that linear models, when scaled, compete effectively with more complex ones, such as Support Vector Machines and K-Nearest Neighbours (both 0.9990). Decision Tree's score (0.9991) is solid but slightly lower, consistent with its role as an interpretable baseline. These results validate the 10% sampling approach, as it preserves data characteristics while enabling efficient evaluation, supporting the study's goal of practical model comparison.

Journal of Statistical Sciences & Data Analytics (JOSSDA)
Volume 1, Issue 2, December 2026 | Print ISSN XXXX-XXXX | Online ISSN XXXX-XXXX
 DOI: 10.XXXXX/jossda.2026.1.1.002

Table 4. Confusion Matrix – Logistic Regression

Actual \ Predicted	0	1
0 (Legitimate)	5676	11
1 (Fraud)	10	5676

Table 5. Confusion Matrix – Decision Tree

Actual \ Predicted	0	1
0 (Legitimate)	5683	4
1 (Fraud)	6	5680

Table 6. Confusion Matrix – Random Forest

Actual \ Predicted	0	1
0 (Legitimate)	5683	4
1 (Fraud)	4	5682

Table 7. Confusion Matrix – Support Vector Machine

Actual \ Predicted	0	1
0 (Legitimate)	5687	0
1 (Fraud)	18	5668

Table 8. Confusion Matrix – K-Nearest Neighbours

Actual \ Predicted	0	1
0 (Legitimate)	5687	0
1 (Fraud)	12	5674

Table 9. Confusion Matrix – XGBoost

Actual \ Predicted	0	1
0 (Legitimate)	5685	2
1 (Fraud)	5	5681

Journal of Statistical Sciences & Data Analytics (JOSSDA)
Volume 1, Issue 2, December 2026 | Print ISSN XXXX-XXXX | Online ISSN XXXX-XXXX
 DOI: 10.XXXXX/jossda.2026.1.1.002

Tables 4-9: The confusion matrices reveal total misclassifications ranging from 7 (XGBoost) to 36 (Support Vector Machine), providing insight into error types. For Logistic Regression (Table 4; 21 errors: 11 false positives [FP], 10 false negatives [FN]), the balanced errors reflect good linear separation after resampling, supporting its use as a simple baseline in the study. Decision Tree (Table 5; 10 errors: 4 FP, 6 FN) shows few mistakes overall, with a slight FN increase that aligns with its rule-based nature, making it suitable for explainable fraud reviews as intended in the methodology.

Random Forest (Table 6; 8 errors: 4 FP, 4 FN) achieves the most balanced errors, reducing Decision Tree's issues through aggregation; this confirms the study's emphasis on ensembles for stable performance. Support Vector Machine (Table 7; 18 errors: 0 FP, 18 FN) has no FP but high FN, indicating over-conservatism in margin decisions, which may require kernel adjustments in high-dimensional data, as noted in the research design. K-Nearest Neighbours (Table 8; 12 errors: 0 FP, 12 FN) follows a similar trend, with its local focus avoiding FP but struggling with outliers, consistent with expectations for instance-based methods.

XGBoost (Table 9; 7 errors: 2 FP, 5 FN) has the lowest errors, thanks to its adaptive boosting, validating its inclusion as a top performer for iterative improvements in imbalanced settings. Together, these matrices demonstrate a progression from balanced errors in ensembles to FN bias in kernels and instances, informing model choices for fraud detection systems.

3.4 Statistical Comparisons

ANOVA and follow-up tests evaluate the importance of performance differences. The tables below present these results.

Table 10: ANOVA Results (CV ROC-AUC Scores)

Source	Sum of Squares	df	F	PR(>F)
Model	0.000004	5.0	15.9209	6.0372e-07
Residual	0.000001	24.0	Nan	Nan

Table 10 indicates significant differences in ROC-AUC means across models ($F = 15.9209$, $p = 6.0372e\text{-}07 < 0.05$), rejecting the null hypothesis of no variation. This supports the need for post-hoc tests and aligns with the study's goal of identifying meaningful differences in algorithm performance for fraud detection.

Journal of Statistical Sciences & Data Analytics (JOSSDA)
Volume 1, Issue 2, December 2026 | Print ISSN XXXX-XXXX | Online ISSN XXXX-XXXX
 DOI: 10.XXXXX/jossda.2026.1.1.002

Table 11: Tukey's HSD Post-Hoc Results (Significant Pairs Only)

Group 1	Group 2	Mean Diff	p-adj	Lower	Upper	Reject
Decision Tree	Logistic Regression	0.0006	0.0020	0.0002	0.0011	True
Decision Tree	Random Forest	0.0008	0.0001	0.0004	0.0012	True
Decision Tree	XGBoost	0.0008	0.0002	0.0003	0.0012	True
K-Nearest Neighbors	Logistic Regression	0.0007	0.0012	0.0002	0.0011	True
K-Nearest Neighbors	Random Forest	0.0008	0.0001	0.0004	0.0013	True
K-Nearest Neighbors	XGBoost	0.0008	0.0001	0.0004	0.0012	True
Logistic Regression	Support Vector Machine	-0.0006	0.0050	-0.0010	-0.0001	True
Random Forest	Support Vector Machine	-0.0007	0.0003	-0.0012	-0.0003	True
Support Vector Machine	XGBoost	0.0007	0.0005	0.0003	0.0011	True

Table 11 highlights nine significant pairwise comparisons ($\alpha = 0.05$), mostly favouring ensembles. For example, Decision Tree underperforms Random Forest and XGBoost (mean differences = 0.0008, $p \leq 0.0002$), illustrating how bagging and boosting reduce errors, as predicted in the study's ensemble focus. K-Nearest Neighbours also lags these leaders (mean differences = 0.0008, $p \leq 0.0001$), showing limitations in handling aggregated decisions. The Support Vector Machine is weaker than Logistic Regression (mean difference = -0.0006, $p = 0.005$) and Random Forest (mean difference = -0.0007, $p = 0.0003$), suggesting that kernel methods add complexity without proportional benefits in this context. Its gap with XGBoost (mean difference = 0.0007, $p = 0.0005$) emphasises the need for regularisation. These findings clarify model rankings, thereby aiding the comparative analysis of the research.

Table 12: Paired t-Tests (CV ROC-AUC Scores)

Pair (Model 1 vs. Model 2)	t-Statistic	p-Value
Random Forest vs. XGBoost	1.4655	0.2167
Logistic Regression vs. Support Vector Machine	4.4609	0.0112

Table 12 shows no significant difference between Random Forest and XGBoost ($t = 1.4655$, $p = 0.2167$), confirming their similar top-tier performance and flexibility in the study. However, Logistic Regression outperforms Support Vector Machine ($t = 4.4609$, $p = 0.0112$), indicating that simpler models can suffice in resampled data, as hypothesised.

Journal of Statistical Sciences & Data Analytics (JOSSDA)
Volume 1, Issue 2, December 2026 | Print ISSN XXXX-XXXX | Online ISSN XXXX-XXXX
 DOI: 10.XXXXX/jossda.2026.1.1.002

4. Conclusion, Limitations and Future Works

This study demonstrates that ensemble-based learning methods, specifically XGBoost and Random Forest classifiers augmented with SMOTE, offer a robust and effective solution for fraud detection in cases of severe class imbalance. As illustrated in Figure 1, the proposed experimental workflow enforces a strict separation between training and testing stages, with class rebalancing applied exclusively to the training data to mitigate information leakage. Within this rigorously controlled pipeline, ensemble models consistently achieved near-perfect ROC–AUC scores, reflecting strong discriminative power, alongside balanced precision–recall behaviour that confirms the reliable detection of minority-class fraud instances.

The iterative evaluation strategy embedded in the workflow (Figure 1) further substantiates that these improvements are systematic rather than incidental. Statistical significance testing across stratified cross-validation folds reveals that ensemble approaches outperform simpler baseline models by a meaningful margin, thereby justifying their added complexity. The feedback-driven design, where underperforming models are routed back to preprocessing or data-splitting stages, ensures methodical refinement and strengthens the internal validity of the comparative analysis.

Beyond model choice, the results emphasise the central role of principled preprocessing. As shown in Figure 1, coordinated handling of missing data, feature construction, and class imbalance enables ensemble learners to fully exploit available information without excessive computational demands. By demonstrating that such disciplined preprocessing can unlock exceptional performance even in resource-limited settings, this study achieves its objective of a thoughtful and rigorous model comparison. Collectively, the findings reinforce the broader implication that accessible, well-engineered ensemble pipelines can deliver enterprise-grade fraud detection performance without reliance on prohibitively complex architectures or infrastructure.

While the machine learning model results demonstrate strong and statistically significant performance, several limitations should be acknowledged. First, the experimental evaluation relies on publicly available, static datasets. Although this choice supports transparency and reproducibility, it does not fully reflect the evolving nature of fraud in real-world systems, where transaction patterns shift over time due to behavioural changes and adversarial adaptation. As such, the reported performance may not directly generalise to non-stationary or streaming environments. Second, we employed SMOTE as the primary mechanism for addressing class imbalance. Although it was carefully restricted to the training phase to prevent data leakage, synthetic data generation may introduce feature-space artefacts that do not represent fraudulent behaviour. This limitation may influence learned decision boundaries, particularly in high-dimensional settings.

Future research may extend this work along several complementary dimensions. One promising direction is the evaluation of ensemble models under temporal and streaming data conditions, with explicit modelling of concept drift and delayed label availability. Such extensions would enhance the operational relevance of the proposed framework. Additionally, incorporating economic cost functions into the evaluation process would further

align model optimisation with real-world fraud mitigation objectives. Ultimately, improving model interpretability remains a crucial avenue for future research. Integrating explainable artificial intelligence techniques, such as feature attribution or surrogate modelling, could enhance transparency and support adoption in regulated environments where auditability and human oversight are essential.

Reference

Abdallah, E., Hamed, K., Mohamed, M., & Ashraf, S. (2021). Account takeover: The rise of synthetic identity fraud and its impact on financial services.

Alarfaj, F. K., Malik, I., Khan, H. U., Almusallam, N., Ramzan, M., & Ahmed, M. (2022). Credit card fraud detection using state-of-the-art machine learning and deep learning algorithms. IEEE Access, 10, 39700-39715.

Ali, M., Khan, P., & Mahmood, K. (2025). Ensemble learning for credit card fraud detection: A comparative performance analysis. Journal of Big Data, 11(1), 112-130.

Ali, H., Hassan, S., & Usman, K. (2025). Credit card fraud detection: A comparative analysis of machine learning techniques.

Balogun, H., Alaka, H., & Egwim, C. (2021). Boruta-Grid-search Least square support vector machine for NO2 pollution prediction using big data analytics and IoT emission sensors. Applied Computing and Informatics. https://doi.org/10.1108/ACI-04-2021-0092

Balogun, H., Alaka, H., Egwim, N. C., & Ajayi, S. (2021a). An application of Machine learning with Boruta Feature selection to improve NO2 pollution prediction. Environmental Design and Management Conference (EDMIC), 551–561.

Balogun, H., Alaka, H., Demir, E. et al. Deconstructability prediction for buildings using machine learning and ensemble feature selection techniques. Sci Rep 15, 22152 (2025). https://doi.org/10.1038/s41598-025-00790-0

Babatoba, O. P., Ishaq, B., Zakari, Y., Mohammed, A. S., Usman, A., Abdullahi, J., & Sadiq, I. A. (2025, April). A Study on the Performance of the Partial Least Squares Regression in Handling Multicollinearity Using Simulated Data. In Royal Statistical Society Nigeria Local Group Annual Conference Proceedings (pp. 318-328).

Brown, S., & Gupta, R. (2021). Ensemble methods for imbalanced credit card fraud detection: A performance review.

Carrasco, L., Urrutia, F., & Abeliuk, A. (2025). Zero-shot decision tree construction via large language models. arXiv preprint arXiv:2501.16247.

Dornadula, V. N., & Geetha, S. (2019). Credit card fraud detection using machine learning algorithms. Procedia computer science, 165, 631-641.

Egwim, C. N., Alaka, H., Toriola-Coker, L. O., Balogun, H., & Sunmola, F. (2021). Applied artificial intelligence for predicting construction project delays. Machine Learning with Applications, 6, 100166.

Ileberi, E., Sun, Y., & Wang, Z. (2021). Performance evaluation of machine learning methods for credit card fraud detection using SMOTE and AdaBoost. IEEE Access, 9, 165286-165294.

Jurgovsky, F., & Al-Samarraie, H. (2018). Real-time credit card fraud detection: Challenges and the role of low latency systems.

Khan, M. Z., Shaikh, S. A., Shaikh, M. A., Khatri, K. K., Rauf, M. A., Kalhoro, A., & Adnan, M. (2022). The performance analysis of machine learning algorithms for credit card fraud detection. iJOE, 19(03),83.

Kumar, S., & Sharma, P. (2023). A comparative study of transaction throughput in rule-based versus machine-learning-based fraud detection systems.

Lianata, I. W., Nugroho, K. N. D., Nathanael, Y., Christopher, N., & Irwansyah, E. (2025). Implementation of Random Forest Algorithm in Handling Imbalanced Data: A Study on Default Models and Hyperparameter Tuning. International Journal of Computer Science and Humanitarian AI, 2(2), 69-74.

López-Rojas, E., Sali, M., & Zaki, F. (2022). Extreme class imbalance in financial fraud datasets: A review of mitigation strategies.

Mensah, E. (2020). A hybrid anomaly detection and supervised classification approach for novel credit card fraud patterns.

Mittal, S., & Tyagi, S. (2019). Performance evaluation of machine learning algorithms for credit card fraud detection. In 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence) (pp. 320-324). IEEE.

Murphy, K. P. (2022). Probabilistic machine learning: an introduction. MIT Press.

Nainwal, M., Satpathi, A., Shamsi, S., Salem, A., Nain, A. S., Vishwakarma, D. K., ... & Mattar, M. A. (2025). Photograph-based machine learning approach for automated detection and differentiation of aerial blight disease in soybean crops. Journal of Big Data, 12(1), 146.

Nilson Report. (2022). Global card fraud losses 2021.

Nilson Report. (2023). Projections for global card fraud losses.

Pfeifer, B., & Kreuzthaler, M. (2025). Calibrated kNN classification via second-layer neighbourhood analysis. Advances in Data Analysis and Classification, 1-21.

Rajora, S., Li, D. L., Jha, C., Bharill, N., Patel, O. P., Joshi, S., ... & Prasad, M. (2018, November). A comparative study of machine learning techniques for credit card fraud detection based on time variance. In 2018 IEEE Symposium Series on Computational Intelligence (SSCI) (pp. 1958-1963). IEEE.

Trivedi, N. K., Simaiya, S., Lilhore, U. K., & Sharma, S. K. (2020). An efficient credit card fraud detection model based on machine learning methods. International Journal of Advanced Science and Technology, 29(5), 3414-3424.

Wu, Y., Ji, T., Zhou, Y., & Zhou, Y. (2024). Support vector machine-based fault diagnosis under data imbalance with application to high-speed train electric traction systems. Machines, 12(8), 582.

Usman, U., Zakari, Y., Suleman, S., & Manu, F. (2017). A comparison analysis of shrinkage regression methods of handling multicollinearity problems based on lognormal and exponential distributions. MAYFEB Journal of Mathematics, 3(2017), 45-52.

Usman, U., Zakari, Y., & Musa, Y. (2017). A comparative study on the prediction performance methods of handling multicollinearity based on normal and uniform distributions. Journal of Basic and Applied Research International, 22(3), 111-117.

Usman, U., Zakari, Y., & Yau, S. A. (2017). Comparative Study on Bias Regression Methods in the Presence of Multicollinearity Based on Gamma and Chi-Square Distributions. Mathematical Theory and Modelling, ISSN 2224-5804.

Xiao, Y., Tan, L., & Liu, J. (2025). Application of machine learning models in fraud identification: A comparative study of CatBoost, XGBoost and LightGBM.

Yadav, A., Singh, B., & Kaur, P. (2024). The critical challenges of credit card fraud detection: Imbalance, dynamics, and false positives.

Zakari, Y., Yau, S. A., & Usman, U. (2018). Handling multicollinearity: a comparative study of the prediction performance of some methods based on some probability distributions, Annals Computer Science Series, 16(1), 15-21.

Zakari, Y., Ogbole, J. A., Sadiq, I. A., Balogun, H., Usman, A., Garba, J., David, R. O., Mohammed, A. S., Abdullahi, J., Sirajo, M., Adamu, I., Ismaila, S. A. & Muhammad, M. K. (2026). Predictive Modelling of Interaction Effects between Artificial Intelligence Adoption and Workforce Digital Skills on Future SME Productivity and Employment Outcomes in Kaduna State. Proceedings of the 2nd International Conference of Statistical Sciences and Data Analytics, Department of Statistics, Ahmadu Bello University, Zaria, Nigeria.