INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

ORIGINAL ARTICLE

Benchmarking Machine Learning Techniques for Credit Card Fraud Detection

Dhwanir Shah[1]*, **Lokesh Kumar Sharma**[2]

1 Department of Computer Science, Gujarat University, Navrangpura, Ahmedabad–9, Gujarat, India
2 National Institute of Occupational Health, Ahmedabad, Gujarat, India

OPEN ACCESS

Received: 14/05/2025
Accepted: 20/05/2025
Published: 10/06/2025

Citation: Shah D, Sharma LK (2025) Benchmarking Machine Learning Techniques for Credit Card Fraud Detection. Indian Journal of Science and Technology 18(21): 1681-1695. https://doi.org/10.17485/IJST/v18i21.903

* **Corresponding author**.

dhwanir12may1982@gmail.com

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

ISSN
Print: 0974-6846
Electronic: 0974-5645

Abstract

Objective: An analysis aimed at measuring the success of five machine learning algorithms—Logistic Regression, Support Vector Machine, Decision Tree, Random Forest, and XGBoost—in recognizing credit card fraud, based on a Kaggle dataset offered by Mr. Anurag Verma for credit card transactions. **Methods:** The dataset was segmented into an 80% training set and a 20% testing set. To counteract class imbalance, the Synthetic Minority Over-sampling Technique was applied. The models' performance was assessed using Accuracy, Precision, Recall, F1-score, ROC-AUC, and PRUAC, with validation executed via K-fold cross-validation for K = 3, 5,10. **Findings:** XGBoost exhibits a commendable equilibrium in its performance metrics, achieving a precision of 84.44%, a recall of 73.57%, and an F1-score of 78.63%. Notably, it excels with the highest PRAUC of 88.65%, demonstrating exceptional performance in contexts with potentially unbalanced classes. Other algorithms may present marginally superior values in specific metrics, the amalgamation of XGBoost's competitive precision, recall, and F1-score, alongside the leading PRAUC, indicates a strong and dependable model. For example, even if an alternative algorithm boasts a higher recall, XGBoost sustains a more favourable balance between precision and recall, as evidenced by its F1-score and PRAUC. Its impressive accuracy of 97.12% and ROCAUC of 98.77% further validate its efficacy. **Novelty:** This research outlines several essential methodological considerations. We implement k-fold cross-validation, varying k at 3, 5, and 10, to assess the robustness and generalization performance of the model. GridSearchCV is utilized for comprehensive hyperparameter tuning to optimize the model's performance. A data split of 80:20 for training and testing is adopted. We incorporate PRAUC as one of the evaluation metrics, with a particular emphasis on the performance of the positive class. Additionally, we apply both Interquartile Range and Z-score methods for effective outlier detection, and we employ a Random Forest method for feature selection.

Keywords: OneHotEncoding; PRAUC; SMOTE; IQR; Z-Score; Logistic Regression; Support Vector Machine; Decision Tree; Random Forest; XGBoost; K-Fold cross validation

1 Introduction

In the time of the pandemic, online shopping has revealed its benefits for consumers of every age, as it allows them to buy products from the comfort of their own homes. The online payment systems are uncomplicated, convenient, and easy to navigate. A credit card, typically a small rectangular piece of plastic or metal, is issued to consumers to facilitate payments, allowing them to borrow funds and repay them without incurring interest within a designated timeframe. The cardholder must subsequently repay the borrowed amount. In today's world, fraudsters exploit credit cards for fraudulent activities. Credit card fraud involves the unauthorized theft of money, items, or services. It transpires when an individual employs someone else's credit card for personal use without the cardholder's awareness or alerting the card issuer.

The rate of credit card transactions in Indonesia increased to 28,360 in May 2022, in contrast to 23,452 in May of the previous year.[(1)] The Federal Trade Commission (FTC) reported approximately 1,579 data breaches, which compromised around 179 million data records, with credit card fraud being the most frequently occurring type of fraud.[(2)] Again In 2023, the Federal Trade Commission (FTC) disclosed that credit card fraud leads to yearly losses totalling tens of billions of dollars.[(3)] This highlights the need for a reliable fraud detection system to ensure that cardholders can use their credit cards safely. Fraud detection can be viewed as a classification problem, where after analysing a large number of transactions, the system can identify and classify them as either fraudulent or legitimate.

Many existing research miss important preprocessing steps such as normalization and standardization and also overlook outlier detection. It has been observed that K-fold cross-validation, proper hyperparameter tuning and feature selection techniques are also missing. As data balancing technique is rarely address in some research by using methods like SMOTE, gives biased results. It is noted that many research focus only on accuracy, with insufficient use of precision, recall, f1-score and PRAUC techniques which are really important for imbalanced classification problem. Such deficiencies considerably impair the real-world performance of fraud detection systems. In detail the limitations of the various studies are provided in Table 1. The primary goal of the proposed model is to develop efficient credit card fraud detection system.

To address these challenges, this study implements a thorough and systematic methodology. The Synthetic Minority Over-Sampling Technique (SMOTE) is used to rectify class imbalance, thereby preventing the model from favouring the majority class. K-fold cross-validation with 3, 5, and 10 folds has been used to ensure the model's stability across various data subsets. For the optimization of hyperparameters, GridSearchCV is applied to enhance model performance. Features with significant outliers are meticulously examined and managed to avoid any distortion during model training. Feature scaling is also conducted to standardize input variables and enhance the efficiency of the algorithms.

In this analysis, the performance of the model is assessed using accuracy, precision, recall, F1-score, ROC-AUC, and PR-AUC. Notably, for performance evaluation the focus is more on precision, recall, f1-score and PRAUC. Precision works to reduce false positives, recall focuses on capturing the majority of fraudulent cases, and PR-AUC delivers a more precise evaluation of performance amidst imbalanced class distributions.

The concurrent use of an 80:20 train-test split with K-fold cross-validation, SMOTE for balancing, GridsearchCV for hyperparameter tuning, explicit feature engineering, a broad set of evaluation metrics including specifically PRAUC, and Random Forest for feature selection might be a novel combination not found in the exact same way in the referenced studies.

Table 1 presents the methodological overview of earlier studies.

Table 1. Summarizing the Methodology

Reference	Dataset Used	Key Contribution	Limitation
[(3)]	IEEE-CIS fraud detection competition	• Data preprocessing and Feature selection have been used. • The results of the experiment show that the integrated Stacking model excels in the detection of credit card fraud, recording an AUC value of 0.960. • Normalization and standardization concepts have been used. • The dataset is distributed in the 80%:20% ratio. • SMOTE+ TOMEK is used to deal with imbalance problem. • Feature selection methods have been used.	• Even though as per the specification precision is one of the evaluation metrics but there is no specification for precision again model performance evaluation focus is only and only on accuracy and AUC values. • No proper justification for Cross validation technique is given. • No justification for why different parameters tuning methods have been used.

Continued on next page

Table 1 continued

(4)	Kaggle	• Challenge: To identifying unusual transactions • Out of the three models evaluated, the Random Forest model achieved near-perfect scores, close to 1.00, across all metrics.	• Normalization technique is missing • Cross validation technique is missing • No specification for hyper parameter tuning approach • Data Balancing technique is missing. • Feature selection method is missing.
(5)	European cardholders dataset	• The goal of this research is to identify legitimate and fraudulent transactions and to compare the performance of different machine learning algorithms. • The research shows that the Random Forest (RF) algorithm achieves the highest accuracy, precision, recall, f1-Score, and Matthews correlation in comparison of other models. • It is concluded that random forest is best among four.	• No specification for which oversampling technique has been used. • Lack of hyper parameter tuning • Lack of Cross validation technique • Lack of Data distribution information. • Lack of Feature selection method • Lack of Train Test split ratio information.
(6)	European bankdataset	• In this paper the authors have compared the algorithms on various parameters and have found that ANN algorithm is giving highest accuracy in comparison of SVM and KNN algorithms. • The end result is evaluated based on the confusion matrix and precision, recall and accuracy is calculated. • Normalization is done. • Undersampling technique has been used.	• The algorithm performance is evaluated using accuracy, precision and recall only. • Hyper parameter tuning is missing. • Lack of Cross validation technique • Only Recall, precision and accuracy metrics and confusion matrix are used for performance evaluation. • Lack of Train Test split ratio information.
(7)	ULB Dataset	• Recommendations such as cost-sensitive loss functions, oversampling, and undersampling are proposed to achieve dataset balance. • A well-balanced dataset enhances the comprehension of the fraud issue. • The study examines different forms of credit card fraud and illustrates how machine learning and neural networks can identify fraudulent activities by scrutinizing historical behaviour. • It also points out that technologies like EMV and 3D Secure have mitigated internal fraud. • Among the classifiers analysed, KNN demonstrated the highest performance.	• Same amount of training and test data has been taken. • No proper specification for data balancing concept for other algorithms except NN. • Cross validation technique is missing. • Hyperparameter tuning is missing. • Normalization and standardization is missing. • Feature selection information is missing. • Final evaluation is done only on the basis of precision and f1-score.
(8)	Kaggle Dataset	• Accuracy: LR -99.88%, K-Means -54.27% and CNN 99.61% • So, LR is the best model among the three classifiers • Main focus is on accuracy and precision to evaluate the performance.	• Hyper parameter tuning is missing. • Precision score values are missing. • Scaling is applied only in K-Means. • No specification for data balancing concept. • Lack of Cross validation technique

Continued on next page

Table 1 continued

[9]	Online Dataset	• According to author analysis KNN and LR are not capable of handling fraud detection at the time of transaction. • All the four algorithms are compared on the basis of accuracy and time factors. • AdaBoost is the best algorithm among the four.	• No specification for Normalization and standardization of data • Parameter tuning to improve the performance is also missing • Lack of cross validation technique • Time duration is specified but without specification of what does it indicate? • Performance evaluation is done only on the basis of Accuracy and time duration only.
[10]	Generated dataset	• Each technique performance was compared on the basis of MSE and accuracy. • It is concluded that SVM is best among the other three for fraud detection.	• Not shown exact scaling process • Lack of Hyper parameter tuning • Lack of parameter description • cross validation technique implementation is missing • Lack of Data Distribution information.
[11]	UCI Machine Learning Data Repository	• The distribution of frauds changes over time due to seasonal factors. • The primary objective of this paper is to enhance fraud detection through improved performance and accuracy. • Random Forest demonstrated a higher level of accuracy at 81.79%, outperforming Logistic Regression and Support Vector Machines, which achieved accuracies of 77.97% and 65.16%, respectively.	• Cross validation technique is missing. • No specification for data balancing technique. • Hyper parameter tuning is missing. • There arises confusion as in paper at 3 places different data distribution is given one is 24000:6000, 70%:20% and 75%:25% and justification for the same is missing. • The performance evaluation decision is solely taken only on the basis of accuracy only.

This paper discusses challenges such as the absence of real-life data, data imbalance, overlapping datasets, lengthy execution times with large data, feature selection, detection costs, and adaptability. To address the issue of imbalance, SMOTE was utilized for oversampling, while CNN and RUS were used for undersampling. The proposed system supports real-time fraud detection by promptly alerting banks, allowing teams to respond quickly. The accuracies of the models reported include 74% for Logistic Regression, 83% for Naive Bayes, 72% for another Logistic Regression model, and 91% for SVM. [12]

The performance of the algorithm is evaluated through fine-tuning techniques such as random oversampling, adjustment of class weights, and the application of both SMOTE and Tomek links, along with smoke oversampling. It has been noted that the model demonstrates high levels of accuracy, precision, recall, and F1-score, while effectively minimizing both false positive and false negative rates. The study indicates that the combination of SMOTE and Tomek yields robust overall performance, whereas the Class Weights method provides a superior balance between precision and recall. Random Oversampling achieved the highest recall and excelled in accurately classifying transactions, as illustrated by the confusion matrix analysis. [13]

This paper discusses a system that continuously tracks transaction data, analysing aspects such as transaction amounts, locations, and user behaviour to pinpoint potentially fraudulent activities. When suspicious transactions are recognized, the system rapidly notifies both the merchant and the cardholder, facilitating immediate measures to prevent possible losses. Data is sourced from a variety of platforms, including data.world, data.gov.in, and Kaggle. The Apriori algorithm is employed in the detection of credit card fraud to analyse and compare purchasing patterns. [14]

The paper discusses about Trojans, phishing kits, keyloggers, and fraudulent base stations, alongside internet vulnerabilities and defective platforms, which exacerbate credit card fraud. Frequency encoding is often a suitable choice for nominal categorical data, as it replaces each category with the frequency or proportion of its occurrences in the dataset. To mitigate the data imbalance issue, under-sampling, over-sampling, and SMOTE techniques are applied. The results demonstrate that XGBoost achieved superior F1-score, accuracy, precision, and recall compared to earlier methods, making it the optimal solution for real-time fraud detection. [15]

Autoencoders are useful in fraud detection as they can reconstruct normal transaction, but it faces struggle with fraudulent transaction and make it useful for real time fraud detection. By evaluating reconstruction errors, transactions that exceed a

specified threshold can be marked as potential fraud. The system recorded an accuracy of 98.5%, precision of 92.3%, recall of 90.1%, and an F1-score of 91.2%.[16]

The proposed framework, based on Deep Convolutional Neural Networks (DCNN), secures 99% accuracy in just 45 seconds by adeptly learning long-term patterns in real-time credit card transactions. It preprocesses the data to eliminate noise and assesses new transactions against historical data to identify fraudulent behaviour. In contrast to Random Forest (RF) and SVM, DCNN provides quicker execution and enhanced accuracy, while Logistic Regression (LR) also achieves 99% accuracy but in a swift 20 seconds, positioning both DCNN and LR as more effective for real-time fraud detection.[17]

This study presents a real-time fraud detection framework for online transactions, utilizing a hybrid strategy that merges Autoencoders (AE) with Extended Isolation Forests (EIF). The model is designed to function without synthetic balancing, naturally accommodating imbalanced datasets. Autoencoders function to reconstruct transactions and identify those with elevated reconstruction errors as suspicious, which are further analysed by the EIF to assign a fraud score. The AE-EIF hybrid approach surpasses standalone models by enhancing detection accuracy and decreasing the rates of false positives and false negatives.[18]

This study introduces an algorithm that combines K-Nearest Neighbor (KNN), Linear Discriminant Analysis (LDA), and Linear Regression (LR) for the purpose of credit card fraud detection. Utilized on four large datasets, including a real-world dataset, the model is designed to maximize recall, thus ensuring the detection of the majority of fraudulent transactions. The method proposed achieved recall scores as high as 1.0 and accuracy rates between 96.64% and 99.89%, reflecting strong performance across all datasets.[19]

This research seeks to create an enhanced deep learning model for identifying fraudulent activities in online interbank money transfers. The authors assessed three models—CNN, LSTM, and CNN-LSTM—utilizing data that conforms to the ISO 8583:1987 standard, which is frequently employed in ATM and POS transactions. The CNN model demonstrated best performance, achieving an accuracy of 99.88%, precision of 87.07%, recall of 100%, an F1-score of 93.09%, and an AUC of 99.94%.[20]

Two algorithms Neural Network and Auto-encoder performance have been shown through confusion matrix. NN has detected fewer false positives in comparison of AE, so it gives the highest precision. The result shows that NN performs better than AE. The algorithms have been compared on the basis of accuracy, precision, recall and F1-score.[21]

This paper has proposed a system to classify alerts in fraud detection system using Random Forest to classify alert as fraudulent or non-fraudulent. The authors have used learning to rank approach to rank the fraudulent alert generated by classifier based on priority. The performances of all these techniques are examined based on precision, F1-score, recall and accuracy. Along with RF Decision tree algorithm is also used for the classification of a transaction.[22]

The algorithms' performance has been evaluated by using Accuracy, F1-score, Precision and Recall, ROCAUC and Confusion metric. The research employed appropriate preprocessing and feature engineering methods, including standardization and parameter optimization. The effectiveness of fraud detection was comprehensively assessed through feature selection, confusion matrices, ROC curves, and evaluation graphs for tuning. The grid search method was used for hyperparameter tuning. The parameter tuning has been applied only on the 4 parameters.[23]

1.1 Research Gap

1. Inconsistent Division of Training and Testing Data:
   - Multiple studies (References[5–7,10]) fail to specify or elucidate their train-test ratio.
   - Other studies like[4,8], and[9] show inconsistency in using conventional splits.
   - Our study uses a transparent and conventional 80:20 division, guaranteeing both consistency and impartiality.
2. Limited Utilization of Cross-Validation:
   - The majority of studies did not implement cross-validation or did not provide adequate justification for its application. (References[4–11]).
   - Only our study and Reference[3] discuss this matter, yet Reference[3] does not offer sufficient justification.
   - Utilizing k-fold cross-validation with k values of 3, 5, and 10 increases the dependability of model evaluation.
3. Limited Balancing Methods:
   - A number of studies either fail to use any balancing techniques or only briefly mention the use of partial or undersampling methods (e.g.,[4,5],[6],[7]).
   - We use SMOTE, a widely accepted method for resolving class imbalance issues.

- The only use of advanced balancing (SMOTE+TOMEK) is found in Reference[3], but it also lacks deeper implementation clarity.

4. Insufficient Hyperparameter Adjustment:

- Various research either neglect or inadequately address hyperparameter tuning (References[4–11]).
- Our research uses a systematic GridSearchCV approach, while Reference[3] incorporates Bayesian optimization.
- This points to a significant shortcoming in the optimal fine-tuning of models in existing studies.

5. Overlooking Feature Engineering:

- A number of studies either fail to conduct feature engineering or only implement it partially References ([4–9,11]).
- This study, together with sources[3] and[10], incorporates thorough feature engineering, a vital component for optimizing model performance.

6. Narrow Measurement Metrics:

- A number of studies rely primarily on simple metrics such as accuracy (References[7–10]), which may fail to depict the actual scenario for imbalanced datasets.
- Our study uses a thorough set of metrics: Accuracy, Precision, Recall, F1-Score, ROC AUC, and PR AUC.
- This investigation uniquely features PRAUC (Precision-Recall AUC), providing a stronger evaluation framework for fraud detection tasks in which the positive (fraud) class is rare.

7. Scarcity of Feature Selection Methods:

- Only our investigation and Reference[3] address the topic of feature selection techniques.
- Numerous studies ignore the importance of feature selection, which can lead to overfitting or the presence of irrelevant input features.
- Utilizing the Random Forest method for selecting features improves the model's explanatory power and overall performance.

8. Overlooking Outlier Detection in Key Features:

- A comprehensive analysis of the selected research shows that most do not address specific techniques for identifying or dealing with outliers in continuous variables such as age.
- Outlier values can notably disrupt model accuracy and alter the dependability of predictive forecasting.
- Conversely, our investigation distinctly uses Z-Score and IQR techniques to identify and address outliers in the age attribute during preprocessing, which guarantees a more accurate and trustworthy dataset for training the model.

2 Methodology

2.1 Analytical Review of Data

The data has been sourced from the Kaggle platform.[?] The dataset contains around 100,000 credit card transactions were captured on the 13th and 14th of October in the year 2020. It contains both genuine and fraud credit card transactions. As it can be seen from the Figure 1 that it contains 7.2% fraud transactions in comparison of 92.8% genuine transactions. In starting the dataset was containing roughly 16 features, including the target feature. Approximately four attributes had missing values, which were dropped using the dropna () method. One-Hot Encoding was utilized for nominal categorical attributes. Suitable scaling techniques were employed to normalize and standardize the numerical attributes. Analysis disclosed that only the 'Age' feature exhibited outliers, as demonstrated in Figure 2. Outliers present in the age feature were systematically detected utilizing both Z-Score and Interquartile Range (IQR) techniques. This two-pronged strategy allowed for the identification and optional exclusion or modification of extreme values that could potentially bias the model. These steps made the data more similar and assisted the model to perform well on new and unseen data.

The bar chart (in Figure 3(a)) explains the frequency of fraudulent transactions over a 24-hour period. The peak activity is noted in the early morning hours, specifically from midnight to 6 AM, with a maximum of 536 incidents recorded at 2 AM. A secondary increase is observed at 1 PM, with 329 cases. It shows that fraudulent activities occur more during off-peak hours.

As shown in the Figure 3(b) merchant wise fraud study. This pattern depicts that fraudsters might be paying attention on high-demand and high-value categories, particularly those involving children's products, electronics, and fashion.

Dataset with Uneven Distribution

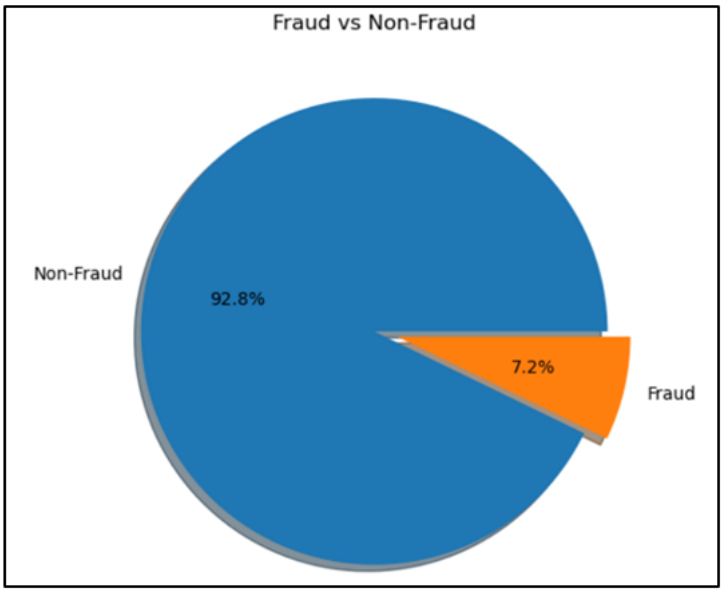


Fig 1. Bias in class distribution

Outlier Detection

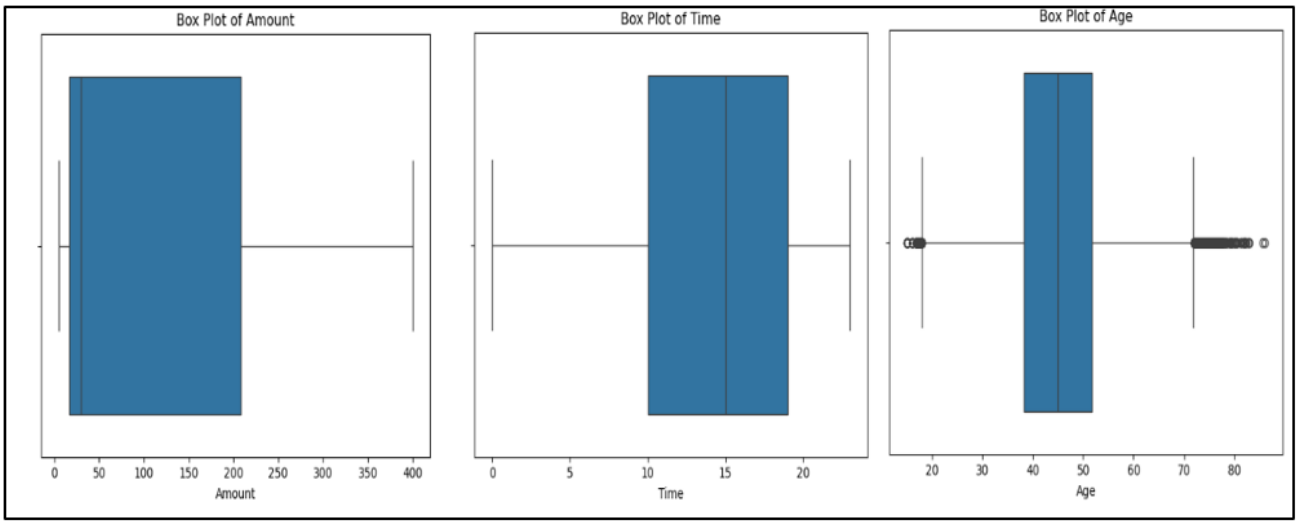


Fig 2. Anomaly Identification

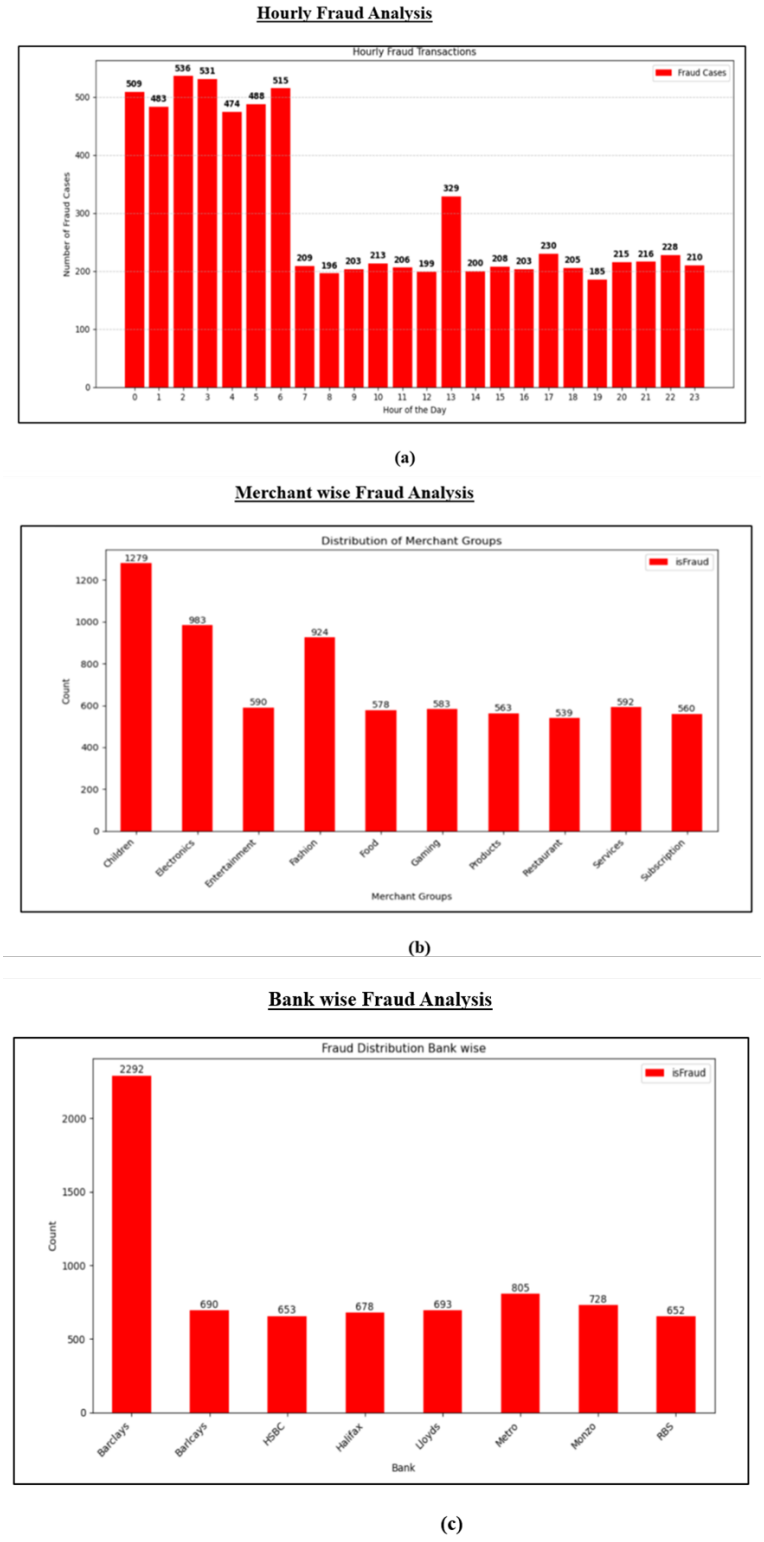


Fig 3. (a) Fraudulent Activities on an Hourly Basis (b) Fraud Analysis by Merchant (c) Fraud Analysis by Bank

As shown in the Figure 3(c), the bar chart indicates that Barclays has the highest incidence of fraudulent transactions, totalling 2,292.

In Figure 4(a), the distribution of fraudulent transactions is presented across different age segments. The age range of 41-50 records the highest number of fraudulent transactions, totalling 2714. The 31-40 and 51-60 age groups follow with significant counts of 1836 and 1704 fraudulent transactions respectively.

The frequency of fraudulent transactions across various transaction amount ranges is shown in Figure 4(b). The majority of these transactions, totalling 5737, occur within the 0 to 50 range. As the transaction amount increases, there is a notable decrease in the number of fraudulent transactions. It indicates a strong correlation between smaller transaction amounts and the likelihood of fraudulent activity in this dataset.

Features such as 'Day', 'Month', and 'Year' were introduced to enhance the model's learning process. SMOTE has been utilized to tackle the significant class imbalance between fraudulent and non-fraudulent transactions, as illustrated in Figure 1. A method for determining feature importance using Random Forest has been implemented to preserve only the most significant features that influence fraud prediction.

Logistic Regression is an extensively utilized statistical method for binary classification tasks, which will separate fraudulent transactions from non-fraudulent transaction. It decides the probability that a particular input X belongs to a designated class by employing the sigmoid function on a linear combination of the input features. The model generates a probability score that can be used to classify the observation into one of two categories by using a threshold.

The supervised learning technique known as Support Vector Machine (SVM) is employed in binary and multi-class classification problems. It seeks to identify the ideal hyperplane that maximizes the distance between the data points of various classes in order to effectively separate them. SVM is resistant to overfitting, particularly when the number of features surpasses the number of observations. To achieve linear separation in the transformed space, SVM uses kernel functions such as polynomial or radial basis function.

A Decision Tree serves as a supervised learning algorithm which is used for classification and regression tasks. It works by recursively dividing the dataset into subsets based on the values of features, resulting in a tree-like structure where each internal node denotes a decision rule, and each leaf node represents an output class or value. The model chooses splits according to criteria like Gini Impurity, Information Gain, or Entropy, with the objective of maximizing the purity of each subset.

Random Forest is an ensemble learning technique that integrates several decision trees to enhance performance in classification or regression tasks. Each tree is constructed from a random sample of the training data, and a random selection of features is chosen at each split, creating varieties among the tress. The overall prediction is given by combining the outputs of all individual trees—using majority voting for classification. This methodology improves accuracy, mitigates overfitting, and effectively manages high-dimensional data. Additionally, Random Forest offers feature importance scores, which are beneficial for feature selection.

XGBoost, an abbreviation for Extreme Gradient Boosting, is a formidable ensemble learning algorithm that is based on gradient boosting decision trees. It constructs models in a sequential order, with each new tree aimed at correcting the errors of the previous trees by optimizing a regularized objective function. XGBoost is common for its fast, accurate, and scalable models using tree pruning, advanced gradient methods, and regularization to prevent overfitting.

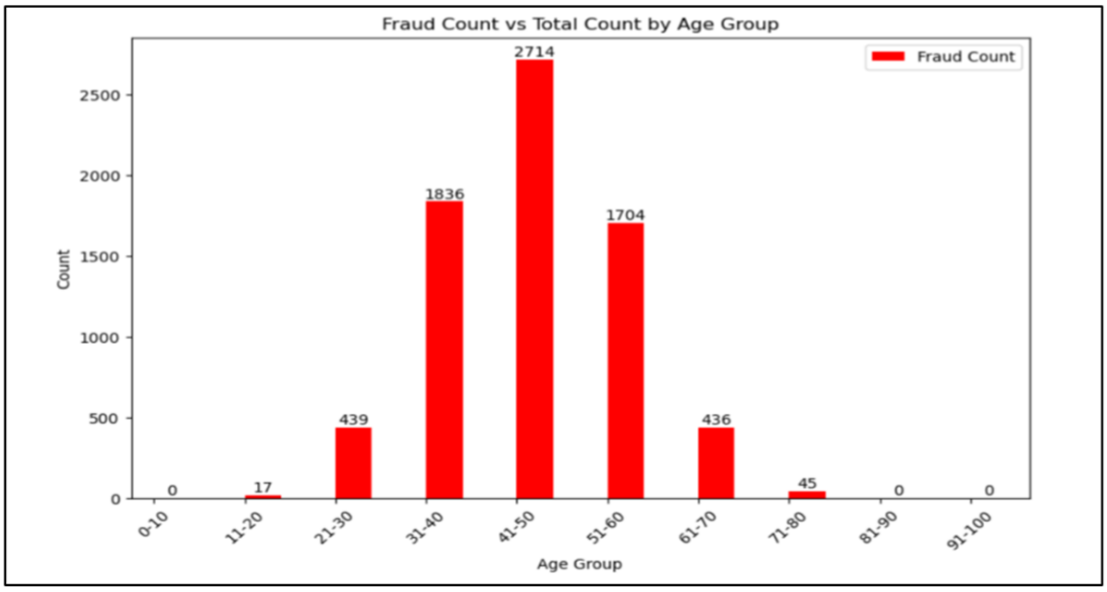
3 Result and Discussion

The main rule of k-fold cross validation is to divide the dataset into 'k' subsets or folds. The model is then trained and evaluated 'k' times, with a different fold acting as the test set during each iteration, while the other folds are employed for training. We have applied the k-fold cross validation method to validate the results following parameter tuning, assuming values of K equal to 3, 5, and 10.

Every classifier will achieve complete accuracy due to the characteristics of the dataset. Since accuracy is not a suitable measure for imbalanced datasets, we have utilized alternative metrics to evaluate the model's performance. We have employed several metrics, including accuracy, precision, recall, F1-score, and ROC-AUC, to evaluate the model's performance. The current study incorporates the Precision-Recall AUC (PRAUC) as one of the evaluation metrics, following the framework established in Reference [24], which demonstrates its effectiveness in evaluating models on imbalanced datasets, especially in fraud detection scenarios.

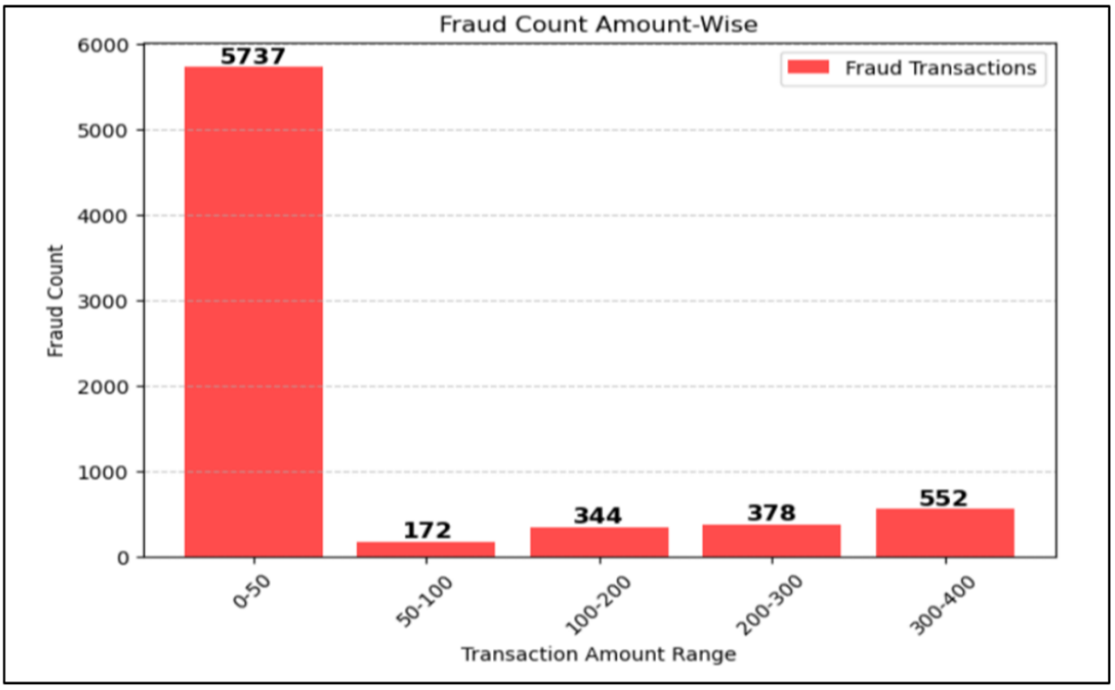
The dataset has been partitioned into 80% for training purposes and 20% for testing. Tables 2(a) and 2(b) present the performance assessment of all models based on various metrics for this data distribution. XGBoost consistently achieves the highest performance in both Train: Test splits and K-fold cross-validation, demonstrates that it is strong enough, reliable and can adjust as per the situations. Although K-fold validation helps in stabilizing performance across all algorithms by minimizing bias and variance, XGBoost remains superior to the other models.

Fraud Incidence Compared to Total Incidence by Age Group



(a)

Distribution of Fraud by Amount wise



(b)

Fig 4. (a) Comparison of Fraud Cases to Total Cases by Age Group (b) Amount-wise Distribution of Fraud

The following analysis can be outlined from the Table 2(a):

XGBoost exceeds all other models across 5 out of 6 metrics, particularly demonstrating superior performance in Precision, F1-Score, ROC AUC, and PR AUC, establishing it as the most balanced and efficient model for fraud detection. The SVM model shows the highest Recall, indicating superior capability in detecting fraud cases; however, this comes with a notable decrease in precision (66.91%), potentially leading to an increase in false positives. While Logistic Regression (LR) exhibits high accuracy and satisfactory precision, it does not perform as well as XGBoost regarding recall and F1-score. With a ROC AUC of 86.22%, the Decision Tree (DT) demonstrates the least effective classification ability when compared to ensemble methods. The Random Forest (RF) model demonstrates a commendable balance in various metrics, particularly with a PRAUC of 88.36%. However, its Accuracy, Precision, F1-score, and ROCAUC metrics fall short when compared to XGBoost.

In summary, it can be concluded that XGBoost outperformed all other algorithms by gaining the highest scores in five critical metrics: Accuracy (97.12%), Precision (84.44%), F1-score (78.63%), ROCAUC (98.77%), and PRAUC (88.65%). These results show that XGBoost is really good at finding fraud accurately and at the same time keep balance between precision and recall. Figure 5 depicts various PRAUC for all the algorithms which have been used in the current study.

As shown in the Figure 5, the subplots depict how precision fluctuates with recall, focus light on the performance of the models when dealing with imbalanced data. XGBoost leads with a PRAUC of 88.66%, indicating the most effective precision-recall balance.

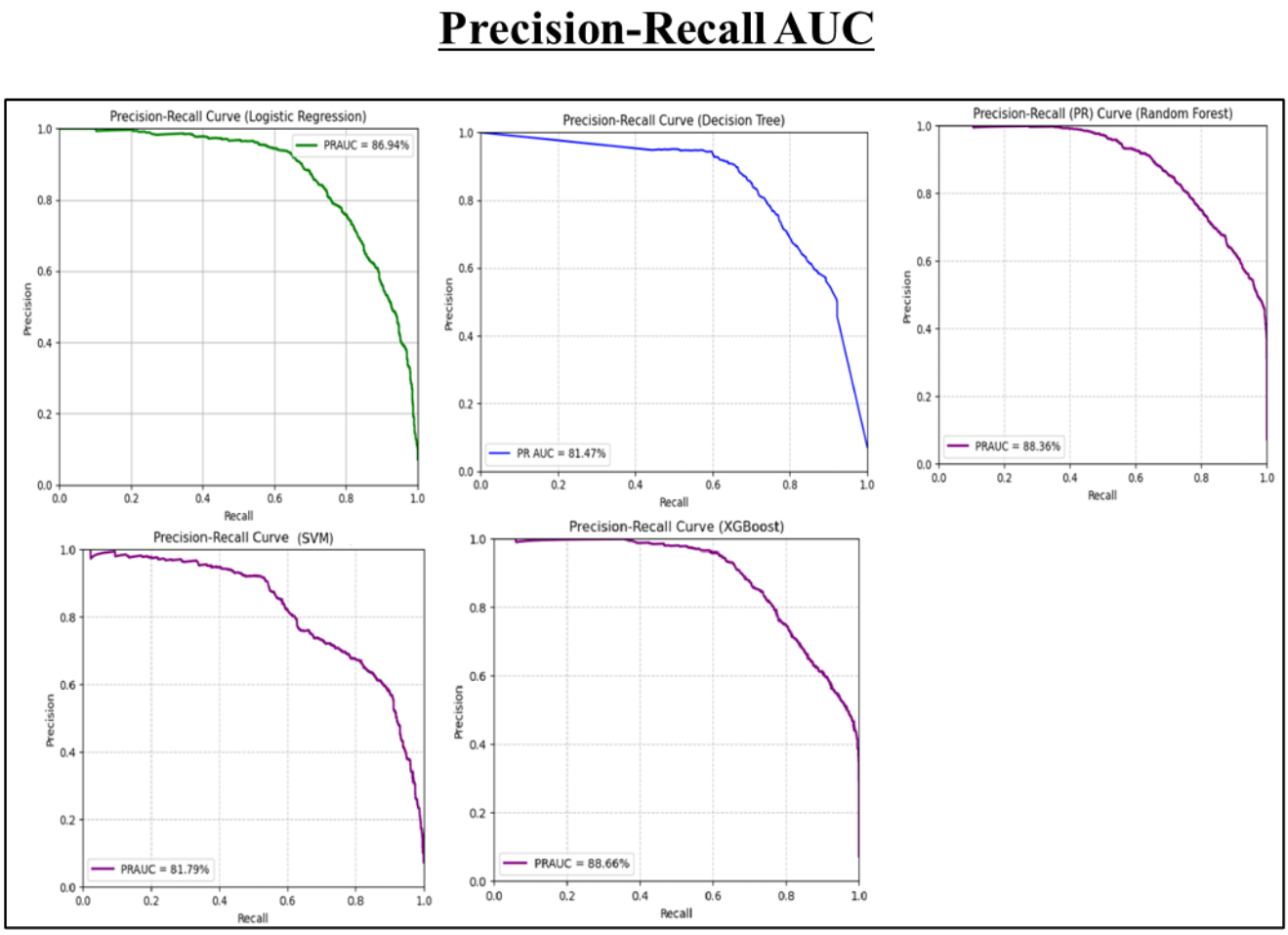


Fig 5. Area Under the Precision-Recall Curve

Random Forest is a close second with a PRAUC of 88.36%, demonstrating solid performance. Logistic Regression shows a moderate PRAUC of 86.94%. Meanwhile, Decision Tree and SVM present lower PRAUC values of 81.47% and 81.79%, respectively, indicating poorer precision-recall trade-offs.

The following summary can be outlined from Table 2(b):

The results obtained from K-Fold cross-validation indicate that the XGBoost algorithm displayed outstanding and consistent performance in five out of six critical evaluation metrics. It notably achieved the highest scores in Accuracy (97.12%), Precision (84.44%), F1-score (78.63%), ROCAUC (98.77%), and PRAUC (88.65%), exceeding the performance of other classifiers in these metrics.

Despite the Random Forest classifier achieving a slightly raised ROCAUC score of 98.79%, XGBoost presented a more balanced and strong performance overall. These results suggest that XGBoost provides better performance by keeping a good balance between finding fraud and catching most fraud cases and which makes it a more reliable choice for fraud detection.

Table 2. Algorithm Performance Evaluation by using 80%:20%. (a) after Parameter Tuning, (b) K-fold Cross Validation Technique (K=3,5,10)

(a) Algorithm Performance Evaluation						
	Accuracy	**Precision**	**Recall**	**F1-score**	**ROCAUC**	**PRAUC**
LR	97.05%	82.98%	74.27%	78.39%	97.92%	86.93%
DT	96.75%	79.51%	73.92%	76.61%	86.22%	81.47%
RF	96.90%	80.00%	75.94%	77.92%	87.23%	88.36%
XGBoost	97.12%	84.44%	73.57%	78.63%	98.77%	88.65%
SVM	95.77%	66.91%	81.43%	73.46%	89.16%	81.79%
(b) K-Fold Performance Evaluation						
	Accuracy	**Precision**	**Recall**	**F1-score**	**ROCAUC**	**PRAUC**
LR	97.05%	82.98%	74.27%	78.39%	97.92%	86.93%
DT	96.75%	79.51%	73.92%	76.61%	94.91%	81.47%
RF	96.90%	80.00%	75.94%	77.92%	98.79%	88.36%
XGBoost	97.12%	84.44%	73.57%	78.63%	98.77%	88.65%
SVM	95.77%	66.91%	81.43%	73.46%	97.50%	81.79%

Table 3. Best Hyper Parameters after Parameter Tuning

Algorithm	**Hyper Parameters**
Logistic Regression	{C=100, class_weight=None, dual=False, fit_intercept=True,max_iter=100, penalty='l2', solver='liblinear', tol=0.01}
SVM	{C=10, class_weight=None, gamma='scale', kernel='rbf', probability=True,random_state=42}
Decision Tree	{ccp_alpha=0.0, class_weight='balanced', criterion='entropy', max_depth=40, min_samples_leaf=3, min_samples_split=5, splitter="best",random_state=42}
Random Forest	{criterion='entropy', max_depth=20, max_features='sqrt', min_samples_leaf=30, min_samples_split=50, n_estimators=300, class_weight='balanced', ccp_alpha=0.0, n_jobs=-1, random_state=42}
XGBoost	{colsample_bytree=1.0, gamma=0.9, learning_rate=0.01, max_depth=5, n_estimators=1500, reg_alpha=0.1, reg_lambda=0.3, subsample=0.9,objective='binary: logistic', random_state=42, n_jobs=-1}

Table 3 presents the optimal hyperparameters obtained following the parameter tuning process across different algorithms. As shown in the Table 4, Its analysis concentrates on critical evaluation metrics—Precision, Recall, F1-Score, and PR-AUC—that are particularly relevant for imbalanced classification issues such as fraud detection. While Accuracy and ROC-AUC are mentioned for completeness, they are less useful in situations where the positive class (fraudulent transactions) is rare. Accuracy can be misleading due to class imbalance, and ROC-AUC measures the model's ability to differentiate between classes but assigns equal importance to both positive and negative classes. In contrast, PR-AUC focuses on the performance of the minority class, making it a more significant metric for evaluating fraud detection systems. Metrics like Matthews Correlation Coefficient (MCC) and Mean Squared Error (MSE) were omitted, as they do not provide additional insights beyond those captured by PR-AUC and F1-score in this context. References [4] and [9] indicate a perfect accuracy, precision, recall, and F1-score, suggesting potential overfitting or inadequate validation methods. There is a lack of information regarding PR-AUC, ROC, or the validation techniques employed. References [5,6,8], and [10] indicate an accuracy greater than 99%, yet they frequently fail to provide metrics such as ROC, PR-AUC, or other significant indicators like PRAUC. This lack of reporting undermines the evaluation of practical

relevance. Reference [3] shows 86% accuracy and only 59% recall, which shows very poor detection for fraud. Reference [11] shows 81.79% accuracy with low precision (65%) and recall (37%) which is not suitable for fraud detection.

Table 4. Metrics Evaluation and Comparison

Result Comparison Table									
Reference	**Algorithms Used**	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**ROCAUC**	**PR-AUC**	**MSE**	**MCC**
[3]	Logistic regression, random forest, LightGBM, XGBoost, andStacking models	86%	N/A	59%	67%	85%	N/A	N/A	N/A
[4]	Logistic Regression, Random Forest and SVM	100%	100%	100%	100%	N/A	N/A	N/A	N/A
[5]	Support Vector Machine, Decision Tree, Naïve Bayes, and Random Forest	99.96%	97.43%	77.56%	86.36%	86.36	N/A	N/A	86.36%
[6]	Artificial Neural Network, Support vector machineand k-Nearest Neighbor	99.92%	81.15%	76.19%	N/A	N/A	N/A	N/A	N/A
[7]	Multiple Linear Regression, Logistic Regression, Random Forest, Neural Network, K Nearest Neighbours and Naive Bayes	N/A	78.00%	81.00%	75.00%	N/A	N/A	N/A	N/A
[8]	Logistic Regression, K-Means and Convolution Neural networks	99.88%	N/A	N/A	N/A	N/A	N/A	N/A	N/A
[9]	Naive Bayes, J48 and AdaBoost	100.00%	N/A	N/A	N/A	N/A	N/A	N/A	N/A
[10]	Decision Tree, Support Vector Machine, Random Forest, K-Nearest Neighbor	99.70%	N/A	N/A	N/A	N/A	N/A	0.0024	N/A
[11]	Logistic Regression, Random Forest, and Support Vector Machines	81.79%	65.00%	37.00%	47.00%	N/A	N/A	N/A	N/A
Present Work	Logistic Regression, Decision Tree, Random Forest, Support Vector Machine, and XGBoost	97.12%	84.44%	73.57%	78.63%	98.77%	88.65%	N/A	N/A

As shown in the Table 5, this study presents a thorough and methodologically robust approach in comparison to the examined references. Most studies ([3,4,8,9,11]) utilize the Train-Test Split technique, yet a few fail to mention this aspect. In our research, Cross-Validation is accurately defined through the application of K-fold cross-validation with K values of 3, 5, and 10. Reference [3] acknowledges this but lacks justification; other sources omit it entirely. The SMOTE+TOMEK approach is used in Reference [3], and Reference [6] applies undersampling balancing techniques, whereas other references either omit this or apply it partially. Our research uniquely and accurately employs SMOTE. The GridSearchCV method for hyperparameter tuning is solely employed by our research and reference [3], which also integrates Bayesian methods. Reference [5] partially adopts this technique. Feature engineering is comprehensively addressed in our Research and References [3,10], whereas [5,6], and [8] only utilize it to a limited extent. Our research employs the most comprehensive metrics set, including Accuracy, Precision, Recall, F1, ROCAUC, and PRAUC. In contrast, other studies utilize only partial sets; specifically, references [3] and [5] incorporate AUC and Matthews correlation, respectively. Reference [10] relies on MSE and accuracy, which are infrequently found in other literature. Only our Research and [3] implement feature selection techniques, while others do not utilize any such methods.

It is important to note that outlier detection methods such as IQR and Z-score, essential for enhancing model robustness, were not discussed in any of the cited literature, yet they have been incorporated into our study.

4 Conclusion

In order to address the previously mentioned limitations, the proposed work incorporates proper preprocessing, data balancing, feature selection, hyperparameter tuning, robust validation, and comprehensive evaluation metrics. Compared to the previous studies which frequently lacked proper validation or relied on limited metrics, this study provides a more detailed and solid evaluation framework. This study examined five machine learning algorithms—Logistic Regression, SVM, Decision Tree, Random Forest, and XGBoost—to detect credit card fraud using a balanced and feature-engineered dataset. Outliers were managed through IQR and Z-score techniques, while significant features were identified using the Random Forest method. The final model employed an 80:20 train-test split, K-fold (K=3,5,10) cross-validation, SMOTE for dataset balancing, and GridSearchCV for optimizing parameters. Among the algorithms assessed, XGBoost yielded the highest performance, achieving an accuracy of 97.12%, precision of 84.44%, recall of 73.57%, ROC-AUC of 98.77%, and PR-AUC of 88.65%. However,

Table 5. Comparative Analysis Table

Analysis Summary Table							
Source	**Train-Test Ratio (in %)**	**Cross-Validation**	**Balancing Technique**	**Hyper parameter Tuning**	**Feature Engineering**	**Evaluation Metrics**	**Feature Selection Method**
This Study	80:20	K-fold cross validation	SMOTE	Gridsearch Method	Done	Accuracy, Precision, Recall, F1- Score, ROCAUC, PRAUC	Random Forest Method
(3)	80:20	No Proper Justification	SMOTE + TOMEK	Gridsearch and Bayesian optimization methods	Done	Accuracy, precision, recall, F1 score, and AUC value	Univariate selection method and recursive elimination method
(4)	70:30	N/A	N/A	N/A	N/A	Accuracy, recall, F1-score, and a confusion matrix	N/A
(5)	N/A	N/A	N/A	Partially	Partially	Accuracy, precision, recall, f1-Score, and Matthews correlation.	N /A
(6)	N/A	N/A	Under-sampling	N/A	Partially	Recall, precision and accuracy metrics and confusion matrix	N/A
(7)	same:same	N/A	Partially	N/A	N/A	Precision, Recall, F1-score	N/A
(8)	75:25	N/A	N/A	N/A	Partially	Accuracy	N/A
(9)	70;30	N/A	N/A	N/A	N/A	Accuracy	N/A
(10)	N/A	N/A	N/A	N/A	Yes done	MSE and accuracy	N/A
(11)	75::25	N/A	N/A	N/A	N/A	Accuracy, Precision, Recall, F1-Score, Support	N/A

a notable limitation is its reliance on synthetic data, which may not completely represent the complexities of actual fraud scenarios. Future efforts should concentrate on testing with real transaction datasets, exploring hybrid approaches, and creating cost-sensitive, real-time fraud detection systems that can evolve with changing attack patterns.

5 Acknowledgements

Authors Submit or Dhwanir Shah and Dr. Lokesh Sharma Submit sincere appreciation to Mr. Anurag Verma for his outstanding effort in providing a user-friendly dataset of fraudulent transactions. His contribution has been critical to facilitating successful experimentation and analysis in this study.

References

1) Wijaya S, Wesly W, Ginting K, Dharma A. Analysis of Credit Card Fraud Detection Performance Using Random Forest Classifier & Neural Networks Model. *Engineering and Technology Journal*. 2024;9(2). https://doi.org/10.47191/etj/v9i02.11.
2) S PM, Swetha BN, Patil M. Credit card fraud detection using Machine Learning. *International Journal of Advanced Research (IJAR)*. https://dx.doi.org/10.21474/IJAR01/16824.
3) Identification and Analysis of Credit Card Fraud Based on Machine Learning Methods. 2024. https://doi.org/10.54254/2754-1169/94/2024OX0182.
4) Efficient Credit Card Fraud Detection Based on Binary Logistic Regression. 2025. https://doi.org/10.54254/2755-2721/2025.18483.
5) Sarker A, Yasmin MA, Rahman MA, Rashid MHO, Roy BR. Credit Card Fraud Detection Using Machine Learning Techniques. *Journal of Computer and Communications*. 2024;12:1–11. https://doi.org/10.4236/jcc.2024.126001.
6) Asha RB, Kumar KRS. Credit card fraud detection using artificial neural network. *Global Transitions Proceedings*. 2021;2:35–41. https://doi.org/10.1016/j.gltp.2021.01.006.
7) Credit Card Fraud Detection using Machine Learning and Deep Learning Techniques. 2020. https://doi.org/10.1109/ICISS49785.2020.9316002.
8) Reddy BS, Mohan UA, Karishma S. Credit Card Fraud Detection using Classification, Unsupervised, Neural Networks Models. *International Journal of Engineering Research & Technology (IJERT)*. 2020;9(4). https://doi.org/10.17577/IJERTV9IS040749.

9) Naik H, Kanikar P. Credit card Fraud Detection based on Machine Learning Algorithms. *International Journal of Computer Applications*. 2019;182(44). https://doi.org/10.5120/ijca2019918521.
10) Fraud detection in credit card transaction using machine learning techniques. 2019. https://doi.org/10.1109/ICSSD47982.2019.9002674.
11) kumar Y, Saini S, Payal R. Comparative Analysis for Fraud Detection Using Logistic Regression, Random Forest and Support Vector Machine. *IJRAR*. 2020;7(4). https://doi.org/10.2139/ssrn.3751339.
12) Real-time Credit Card Fraud Detection Using Machine Learning. 2019. https://doi.org/10.1109/CONFLUENCE.2019.8776942.
13) Adesina MT, Howe L. Credit Card Fraud Detection Using Machine Learning and Artificial Intelligence (Imbalanced Dataset). *International Journal of Science and Research Archive*. 2024. https://doi.org/10.30574/ijsra.2024.12.2.1430.
14) Kumar R, Yaing TH. Dynamic E-commerce Solution for Real-Time Detection of Credit Card Fraud. *Computer Networks*. 2023. Available from: https://www.researchgate.net/publication/371469915_Dynamic_Ecommerce_Solution_for_Real-Time_Detection_of_Credit_Card_Fraud.
15) Kumar BS, Yadav PP, Reddy MR. An intelligent approach to detect and predict online fraud transaction using XGBoost algorithm. *Indonesian Journal of Electrical Engineering and Computer Science*. 2024;35(3):1491–1498. https://doi.org/10.11591/ijeecs.v35.i3.pp1491-1498.
16) Real-Time Fraud Detection in Financial Transactions Using Autoencoders. 2024. Available from: https://www.researchgate.net/publication/387271311_Real-Time_Fraud_Detection_in_Financial_Transactions_Using_Autoencoders.
17) Chen JIZ, Lai KL. Deep Convolution Neural Network Model for Credit-Card Fraud Detection and Alert. *Journal of Artificial Intelligence and Capsule Networks*. 2021;3(2):101–112. https://doi.org/10.36548/jaicn.2021.2.003.
18) Abbassi H, Mendili SE, Gahi Y. Real-Time Online Banking Fraud Detection Model by Unsupervised Learning Fusion. *High Tech and Innovation Journal*. 2024;5(1). https://doi.org/10.28991/HIJ-2024-05-01-014.
19) Chung J, Lee K. Credit Card Fraud Detection: An Improved Strategy for High Recall Using KNN, LDA, and Linear Regression. *Sensors*. 2023;23:7788. https://doi.org/10.3390/s23187788.
20) Hasugian LS, Suharjito. FRAUD DETECTION FOR ONLINE INTERBANK TRANSACTION USING DEEP LEARNING. *Syntax Literate: Jurnal Ilmiah Indonesia*. 2023;8(6). http://dx.doi.org/10.36418/syntax-literate.v6i6.
21) Sharma P, Pote S. Credit Card Fraud Detection using Deep Learning based on Neural Network and Auto-encoder. *International Journal of Engineering and Advanced Technology (IJEAT)*. 2020;9(5). https://doi.org/10.35940/ijeat.E9934.069520.
22) More RS, Awati CJ, Shirgave DSK, Deshmukh DRJ, Patil SS. Credit Card Fraud Detection Using Supervised Learning Approach. *INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH*. 2020;9(10). Available from: https://www.ijstr.org/final-print/oct2020/Credit-Card-Fraud-Detection-Using-Supervised-Learning-Approach.pdf.
23) Bogireddy SR, Murari H. A Hyperparameters Tunned ML Algorithm for Fraud Identification in Banking and Financial Transactions. *International Journal of Innovative Science and Research Technology*. 2024;9(8). https://doi.org/10.38124/ijisrt/IJISRT24AUG458.
24) Sofaer HR, Hoeting JA, Jarnevich CS. The area under the precision-recall curve as a performance metric for rare binary events. *Methods in Ecology and Evolution*. 2019;10:565–577. https://doi.org/10.1111/2041-210X.13140.