International Journal
of
Research in Finance and Management

P-ISSN: 2617-5754
E-ISSN: 2617-5762
Impact Factor (RJIF): 5.32
IJRFM 2025; 8(2): 526-533
www.allfinancejournal.com
Received: 15-08-2025
Accepted: 17-09-2025

Dr. Prashant Wadkar
MCA, Assistant Professor, International Institute of Management Science, Chinchwad, Pune, Maharashtra, India

Dr. Shivaji Mundhe
BCA, Directore, International Institute of Management Science, Chinchwad, Pune, Maharashtra, India

Correspondence Author:
Dr. Prashant Wadkar
MCA, Assistant Professor, International Institute of Management Science, Chinchwad, Pune, Maharashtra, India

Comparative study of different machine learning algorithms used for credit card fraud detection

Prashant Wadkar and Shivaji Mundhe

DOI: https://www.doi.org/10.33545/26175754.2025.v8.i2f.579

Abstract
With credit card fraud becoming a major concern, advanced fraud detection systems are required to protect financial transactions due to the exponential growth in credit card transactions. It appears that manually identifying the questionable transaction is very challenging and time-consuming. These issues are resolved by machine learning because of its accuracy and speed. This study has demonstrated the accuracy with which machine learning algorithms identify fraudulent transactions.

Robust models were constructed using algorithms such as Logistic Regression, Decision Tree, Random Forest, LightGBM, XGBoost, Adaboost, and CatBoost using a standardized dataset that contains both authentic and fraudulent transactions. Comprehensive analyses were conducted using a variety of classification criteria, including F1 score, recall, accuracy, and precision. The effectiveness, drawbacks, and advantages and downsides of various algorithms were examined. All of the developed models have been proven to perform better; however, in comparison, models constructed using the Random Forest, XGBoost, Decision Tree, and LightGBM algorithms are more accurate, while CatBoost has produced the lowest accuracy.

Keyword: Machine learning, credit card fraud detection, fraud prevention, logistic regression, random forest, LightGBM, decision tree, XGBoost, AdaBoost, CatBoost

1. Introduction

The amount of credit card fraud is rising daily. Fraudsters are using new technologies to create new schemes and methods of committing fraud. Earlier identifying credit card fraud before a transaction is made is one of the main obstacles. By examining transaction patterns and detecting anomalies, machine learning can be used to find these kinds of frauds and suspicious activity. Financial losses can be greatly decreased. And also the consumer and financial institution security can be improved by implementing real-time fraud detection systems. Through the use of powerful algorithms and regular data updates, these systems are able to adjust to changing fraud strategies, providing more durable and trusted defense against fraudulent activities.

2. Literature Review

The paper [2] proposes a machine learning-based credit card fraud detection engine using genetic algorithms for feature selection. The engine uses various ML classifiers, including Decision Tree, Random Forest, Logistic Regression, Artificial Neural Network and Naïve Bayes. The performance of the engine was evaluated using a European cardholder dataset, showing it outperforms existing systems. The GA-RF achieved an overall optimal accuracy of 99.98%, while GA-DT achieved a remarkable accuracy of 99.92%. The results were superior to existing methods. The proposed framework was validated on a synthetic credit card fraud dataset, with GA-DT achieving an AUC of 1 and 100% accuracy. The GA-ANN achieved an AUC of 0.94 and 100% accuracy.

During research [3] researchers have designed and developed a fraud detection method for Streaming Transaction Data. Their objective for doing so was to analyze the past transaction details of the customers and extract the behavioral patterns, by which the cardholders are clustered into different groups based on their transaction amount. Then they created models separately and did the comparison of the models created. They have used the Matthews Correlation Coefficient (MCC) for performance measure, SMOTE

(Synthetic Minority Oversampling Technique) to handle imbalance dataset and found SMOTE is the best to handle the imbalanced dataset. They have also used one-class SVM (Support Vector Machine) to handle the imbalance dataset. Models built with Logistic regression, Decision Tree and Random Forest produced the better results.

In paper [12] the authors have worked on the Credit Card dataset and received the accuracy for Naïve Bayes, K-nearest neighbor and Logistic Regression as 97.92%, 97.69% and 54.86% respectively. They found that the K-nearest neighbor algorithm performs better than that of Naïve Bayes and Logistic Regression Algorithm.

In a research paper [13] researchers have used Naïve Bayes and Support Vector Machine algorithms. They did the Evaluations using individual (standard) models and hybrid models that combine majority voting and AdaBoost techniques. As a performance statistic, the MCC (Matthews Correlation Coefficient) metric was used.

The above research papers have provided important knowledge about the different machine learning techniques and algorithms which can be used while building the robust credit card fraud detection models.

Below Table 1 shows the characteristics of different algorithms such as their effectiveness, strengths and weaknesses.

Table 1: Comparison of Machine Learning Algorithms based on their characteristics and properties.

Sr. No.	ML Algorithms	Effectiveness	Strengths	Weaknesses
1	Logistic Regression	Good for binary classification problems, interpretable, and computationally efficient.	Simple and easy to implement. Provides probabilities for outcomes. Low risk of overfitting.	Limited expressiveness for complex relationships. Assumes linear decision boundaries
2	Random Forest	Strong performer for both classification and regression tasks. Robust to overfitting.	Robust to outliers and noise. Can handle a large number of features. Handles non-linearities well.	Lack of interpretability compared to simpler models. Can be computationally expensive.
3	LightGBM	Efficient gradient boosting framework, particularly suited for large datasets.	High performance and efficiency. Can handle categorical features without preprocessing. Scalable to large datasets.	May require tuning to achieve optimal performance. Prone to overfitting with smaller datasets.
4	Decision Tree	Simple and interpretable model for classification and regression tasks.	Easily interpretable. Handles non-linearity. No need for feature scaling.	Prone to overfitting, especially with deep trees. Sensitive to small variations in the data. Limited expressiveness for complex relationships.
5	XGBoost	Powerful gradient boosting algorithm with high predictive performance.	Regularization techniques to prevent overfitting. High accuracy and speed. Handles missing values well.	Requires careful tuning of hyperparameters. Can be computationally intensive.
6	Adaboost	Ensemble method that combines weak learners to create a strong classifier.	Can adapt to complex decision boundaries. Less prone to overfitting. Works well with a variety of base learners.	Sensitive to noisy data and outliers. Training can be slow.
7	CatBoost	Gradient boosting algorithm designed for categorical features.	Handles categorical features without preprocessing. Robust to overfitting. Good out-of-the-box performance.	May require tuning for optimal performance. Can be computationally intensive.

References: [4, 5, 6, 7, 8, 9, 10, 11], The comparison as shown in Table 1 is self-explanatory.

3. Objective of the study: The main objectives of this research study are as mentioned below,

1. To carry out exploratory data analysis in order to understand the characteristics of features related to the transactions as well as the structure of the dataset.
2. To create various models by using different machine learning algorithms for detecting credit card frauds.
3. To perform a comparative analysis of all the build models on the basis of different evaluation metrics.
4. To find out the most accurate model for detecting credit card fraud transactions.

4. Technology Used

While developing the different machine learning based algorithms for credit card fraud detection following technologies are used,

Table 2: Technology Used

Technology	Name
Editor	Google Colaboratory
Programming Language	Python
Python Libraries	pandas, numpy, seaborn, matplotlib, sklearn

5. Research Methodology Adopted: While developing multiple models for detecting credit card frauds standard research methodology was followed, The required dataset was collected from the Kaggle website and steps like statistical data analysis, exploratory data analysis, data preprocessing, model building and model evaluation were performed carefully.

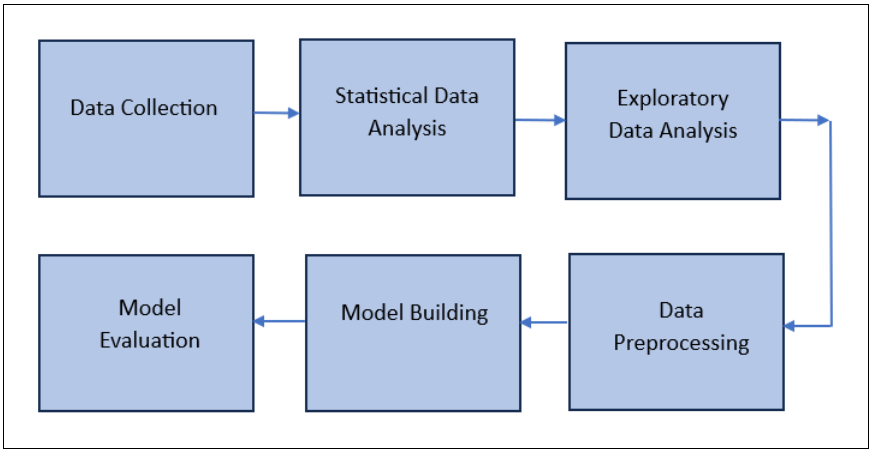


Fig 1: Model Building Lifecycle

VI. Model Development Lifecycle

A. Dataset Information: Secondary data has been used during the research, and it has been taken from the genuine and renowned Kaggle website. The dataset contained a total of 2,84,807 transactions with 31 different features, providing information related to the transactions. The PCA was already performed on multiple features in the dataset; those were named V1, V2, V3, V4, V5... V28. There were two features, i.e., 'Time' and 'Amount', on which PCA was not performed, so they were present in their original format. The 'Time', which was in seconds, is the time elapsed between each transaction and the first transaction in the dataset. The feature 'Amount' was representing the transaction amount. The feature 'Class' is the target variable, representing a value of 0 if the transaction is legitimate or normal and 1 if it is fraudulent or suspicious.

B. Data Analysis

After data collection, statistical and exploratory data analysis was performed on the entire dataset to understand underlying trends and patterns present in the dataset. It has been found that except for the "Class" column, all remaining columns were in the float data type. Target column "Class" was in the integer format and comprises two different values such as 1 and 0, representing if the respective transaction is genuine or legitimate. No missing values were present in the dataset. Dataset was imbalanced in nature, out of a total 2,84,807 number of transactions, 284,315 were legitimate/normal and 492 were fraudulent/suspicious transactions. The graphical representation of the imbalance dataset is as shown in Figure 2.

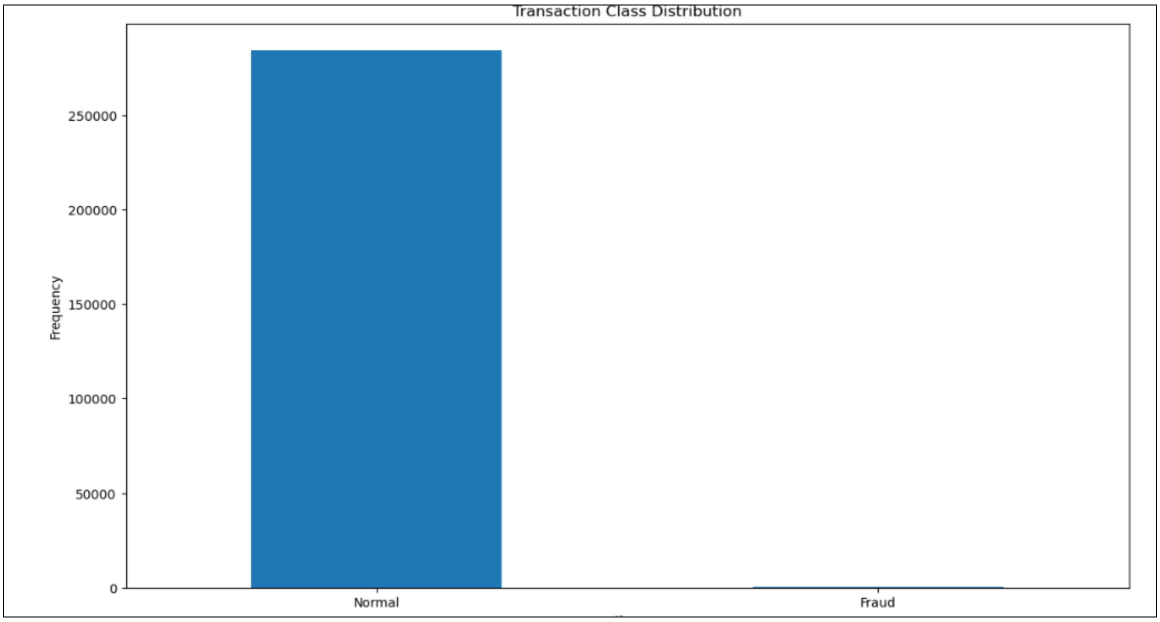
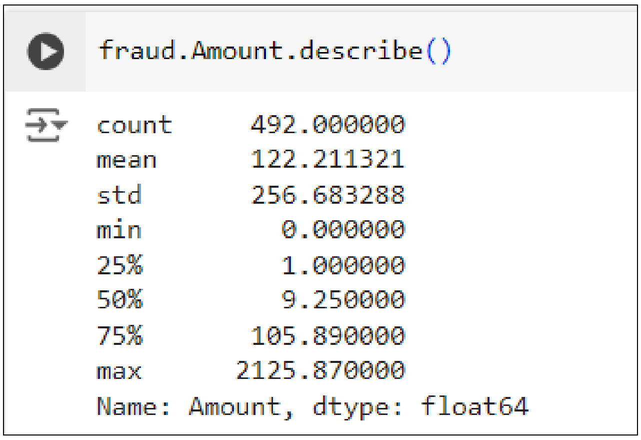


Fig 2: Transaction Class Distribution (Imbalanced Dataset)

The column "Amount" was representing the transaction amounts, their patterns in genuine and fraudulent transactions were observed as below:

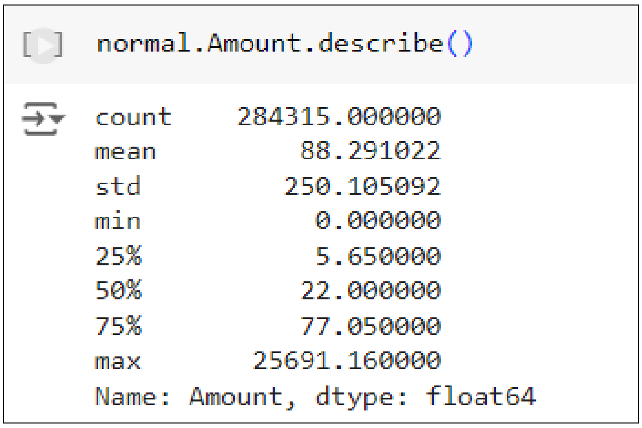
The Fraud transactions amount were ranging in between $0 to $2125.87, while legitimate transactions amount were ranging in between $0 to $25691.16. Average amount of fraudulent and genuine transactions were $122.21 and $88.29 respectively.



```
fraud.Amount.describe()
```

```
count     492.000000
mean      122.211321
std       256.683288
min         0.000000
25%         1.000000
50%         9.250000
75%       105.890000
max      2125.870000
Name: Amount, dtype: float64
```

Fig 3: Amount Details of Fraudulent Transactions



```
normal.Amount.describe()
```

```
count    284315.000000
mean         88.291022
std         250.105092
min           0.000000
25%           5.650000
50%          22.000000
75%          77.050000
max       25691.160000
Name: Amount, dtype: float64
```

Fig 4: Amount Details of Legitimate Transactions

Distribution of amounts were examined by plotting histograms as in Figure 5. Below. The data is right skewed in nature.

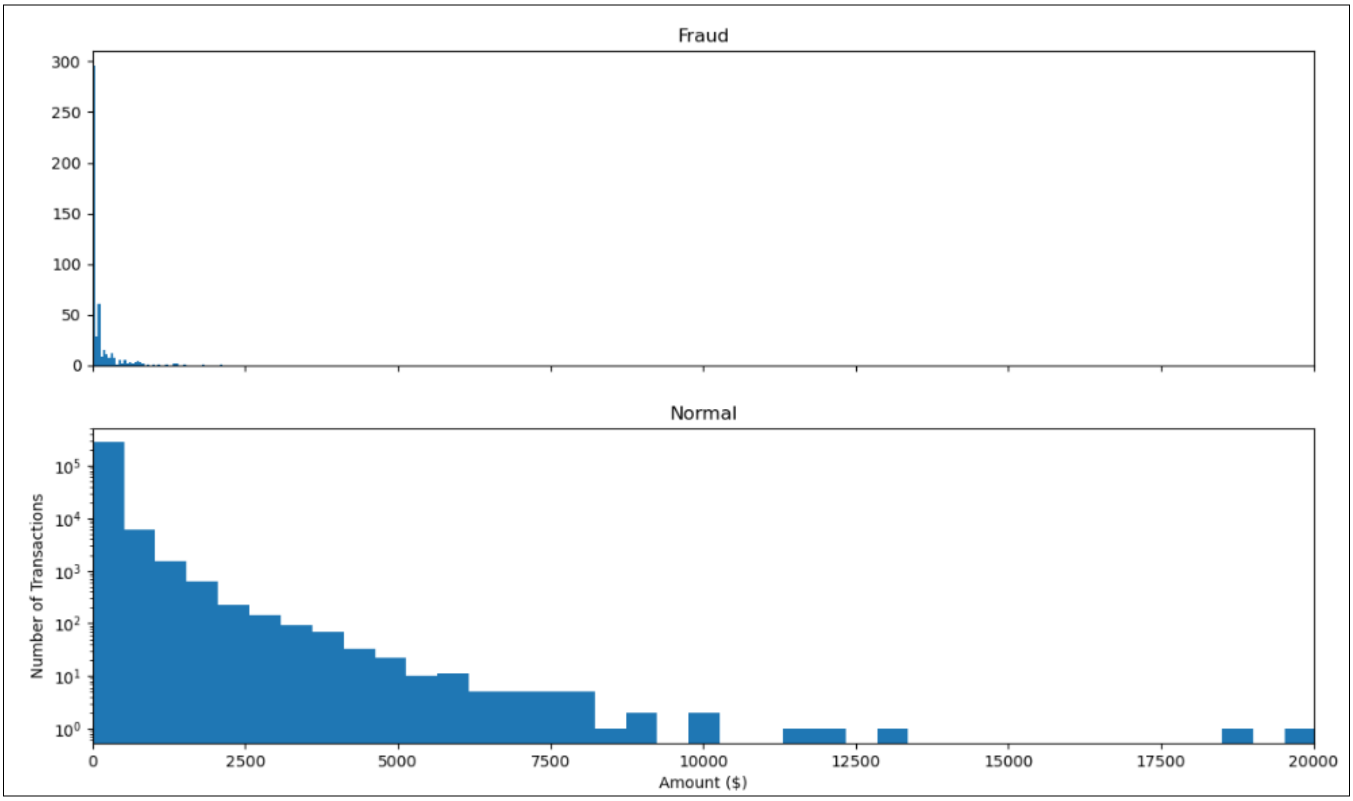


Fig 5: Transaction class (normal and fraudulent) distribution

Transactional amounts with respect to the time in seconds were observed through scatter plots, below are the plots as in Figure 6.

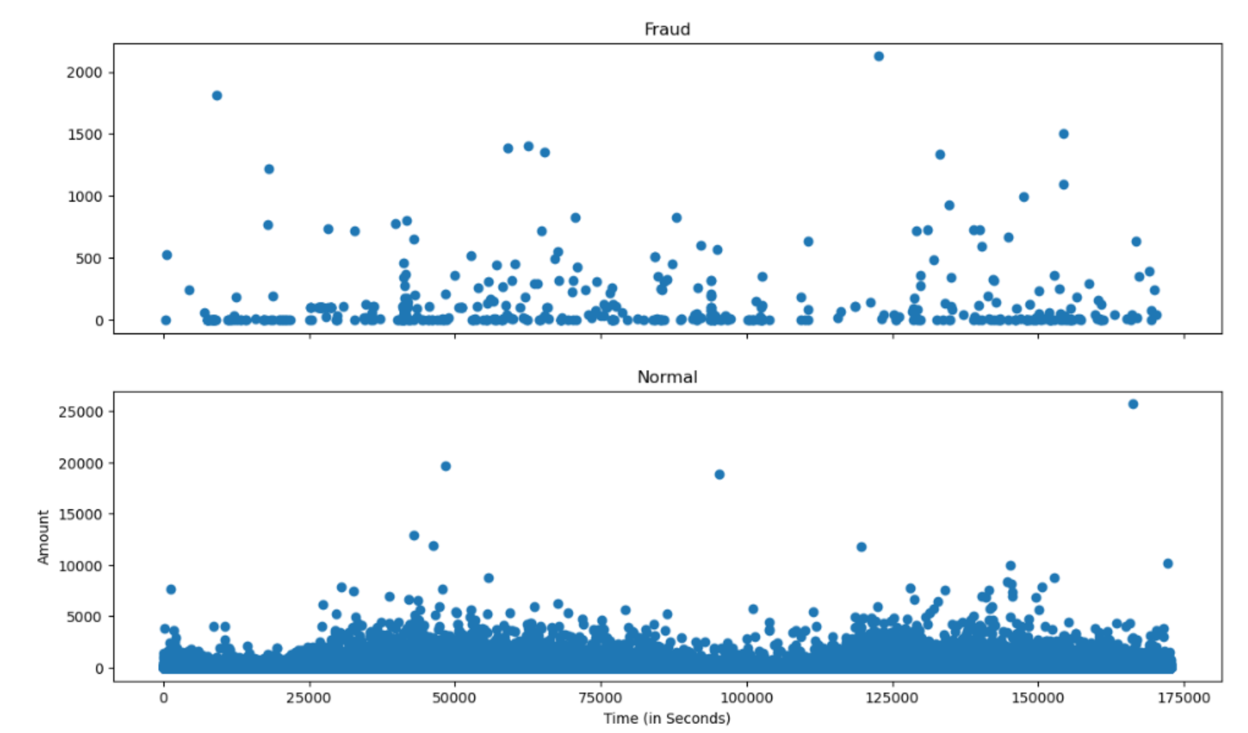


Fig 6: Transaction class (normal and fraudulent) distribution as per time (in seconds)

To observe the correlation between different features the heatmap also was observed.

C. Data Preprocessing: After analyzing the dataset, it was transformed into a suitable format before being passed to the algorithms to build the models. Less correlated columns were dropped to reduce the further complexity. As the dataset was imbalanced in nature, the random oversampling method was applied to balance it, by which the transactional dataset got increased to 568630. Both normal and fraudulent transactions now became the same in count. The below bar plot represents it in better way

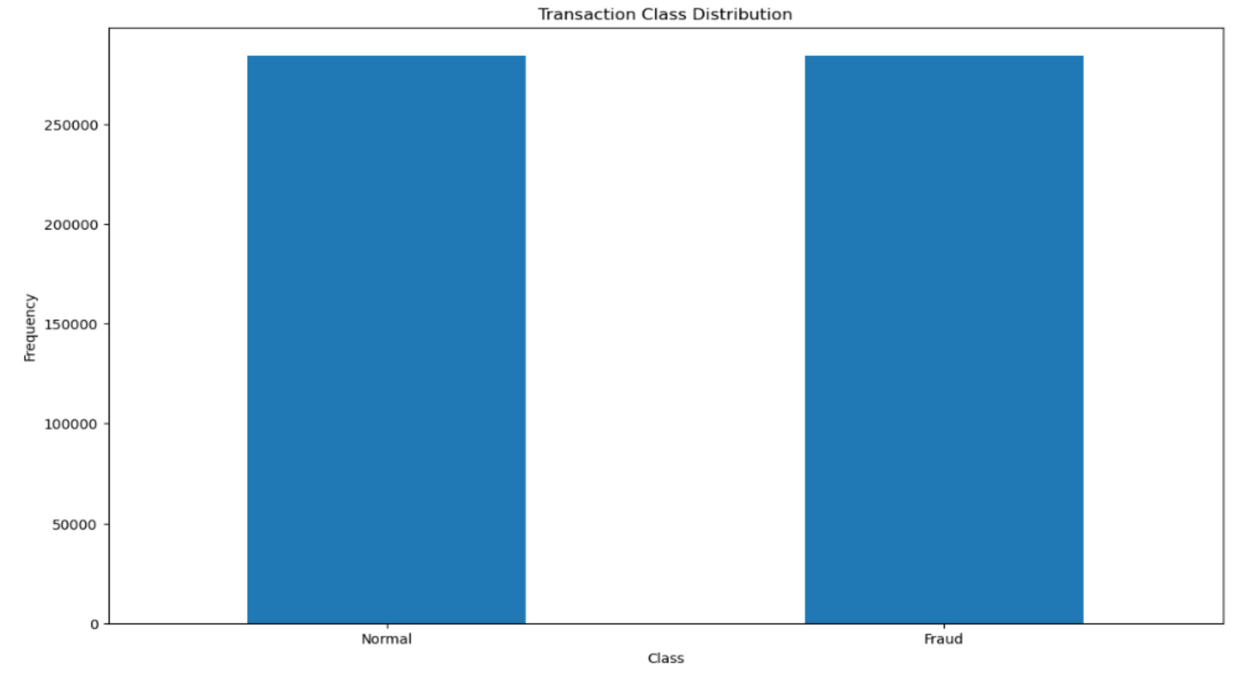


Fig 7: Transaction class (normal and fraudulent) distribution after performing oversampling.

As all columns were on different scales, to bring them on the same scale, the Standard Scaler method was applied. After that, the data was splitted into the train and test sets and provided for the model training and testing purpose.

D. Algorithms

Machine learning based algorithms such as Logistic Regression, Decision Tree, Random Forest, XGBoost, LightGBM, Adaboost and CatBoost were utilized while building the models for credit card fraud detection.

With its straightforward implementation and probabilistic interpretations, logistic regression is a powerful yet simple algorithm. Decision Trees can capture nonlinear relationships between features, making them versatile for a wide range of classification problems. Several trees are combined in Random Forest to provide robustness against noise and overfitting. Because tree boosting is optimized, XGBoost performs exceptionally well and quickly. LightGBM offers even faster training through effective algorithms based on histograms. Adaboost focuses on enhancing weak classifiers into a strong ensemble through iterative improvement. CatBoost's efficient implementation ensures competitive performance even with large datasets, making it a reliable choice for real-world applications where accuracy and speed are crucial.

E. Results

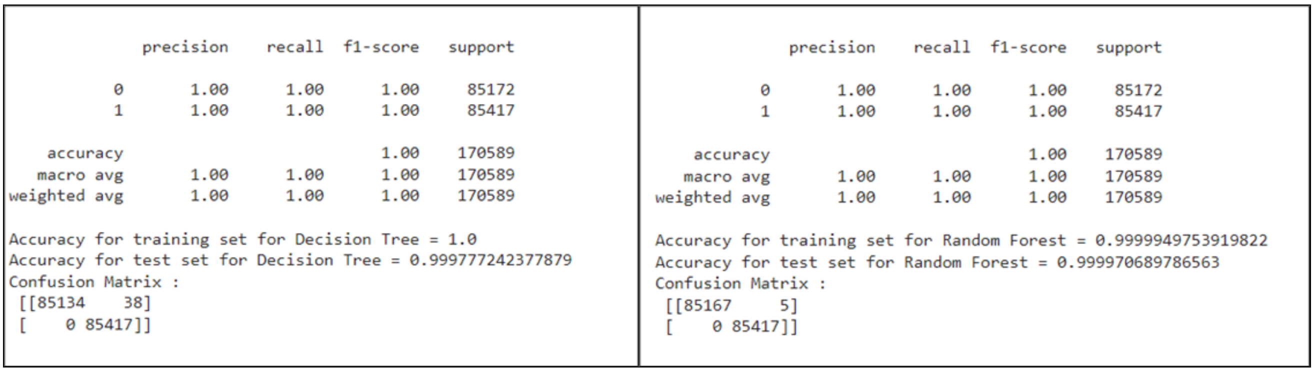
After compilation of the training process all the models were tested on the basis of different classification evaluation metrics which are mentioned below.

- Training and Testing Accuracy
- Precision
- Recall
- F1 Score

Table 3: Comparison of Model's Accuracy and the Ranks Obtained

Sr. No.	Machine learning based built models	Train Accuracy	Test Accuracy	Ranks based on the Performance
1	Logistic Regression	0.94986	0.95101	6
2	Decision Tree	1.0	0.99984	3
3	Random Forest	0.99999	0.99996	1
4	XGBoost	1.0	0.99992	2
5	LightGBM	0.99959	0.99950	4
6	Adaboost	0.96410	0.96496	5
7	CatBoost	0.92133	0.92204	7

Table 3. Represents the comparison of the training and testing accuracy of all the models built on the basis of various machine learning algorithms. Actually, all the models are performing excellently. But comparatively, the Decision Tree, Random Forest, XGBoost and LightGBM models are more accurate, as shown in the figure, with a train and test accuracy of them nearly equal to 1.00. CatBoost has received the lowest test accuracy of 0.92204.



```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     85172
           1       1.00      1.00      1.00     85417

    accuracy                           1.00    170589
   macro avg       1.00      1.00      1.00    170589
weighted avg       1.00      1.00      1.00    170589

Accuracy for training set for Decision Tree = 1.0
Accuracy for test set for Decision Tree = 0.999777242377879
Confusion Matrix :
 [[85134    38]
 [    0 85417]]
```

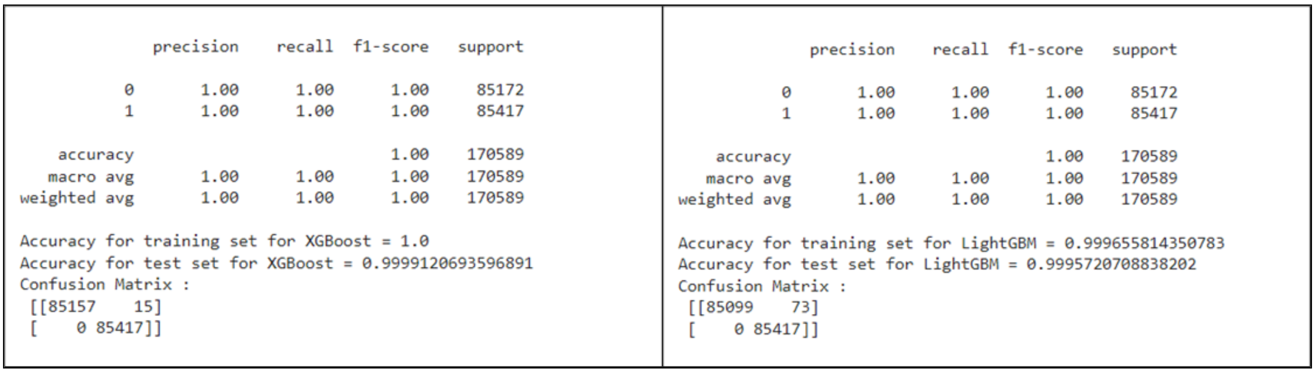
```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     85172
           1       1.00      1.00      1.00     85417

    accuracy                           1.00    170589
   macro avg       1.00      1.00      1.00    170589
weighted avg       1.00      1.00      1.00    170589

Accuracy for training set for Random Forest = 0.9999949753919822
Accuracy for test set for Random Forest = 0.999970689786563
Confusion Matrix :
 [[85167     5]
 [    0 85417]]
```

Fig 8: Evaluation Metric results of Decision tree and Random Forest Model



```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     85172
           1       1.00      1.00      1.00     85417

    accuracy                           1.00    170589
   macro avg       1.00      1.00      1.00    170589
weighted avg       1.00      1.00      1.00    170589

Accuracy for training set for XGBoost = 1.0
Accuracy for test set for XGBoost = 0.9999120693596891
Confusion Matrix :
 [[85157    15]
 [    0 85417]]
```

```
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     85172
           1       1.00      1.00      1.00     85417

    accuracy                           1.00    170589
   macro avg       1.00      1.00      1.00    170589
weighted avg       1.00      1.00      1.00    170589

Accuracy for training set for LightGBM = 0.999655814350783
Accuracy for test set for LightGBM = 0.9995720708838202
Confusion Matrix :
 [[85099    73]
 [    0 85417]]
```

Fig 9: Evaluation Metric results of XGBoost and LightGBM Model

Table 4: Comparison of Model's with Precision, Recall and F1 Score

Model	Evaluation Metric	0 (legitimate)	1 (Fraudulent)
Logistic Regression	Precision	0.92	0.98
	Recall	0.98	0.92
	F1 Score	0.95	0.95
Decision Tree	Precision	1.00	1.00
	Recall	1.00	1.00
	F1 Score	1.00	1.00
Random Forest	Precision	1.00	1.00
	Recall	1.00	1.00
	F1 Score	1.00	1.00
XGBoost	Precision	1.00	1.00
	Recall	1.00	1.00
	F1 Score	1.00	1.00
LightGBM	Precision	1.00	1.00
	Recall	1.00	1.00
	F1 Score	1.00	1.00
Adaboost	Precision	0.96	0.98
	Recall	0.98	0.95
	F1 Score	0.97	0.97
CatBoost	Precision	0.88	0.97
	Recall	0.97	0.87
	F1 Score	0.92	0.92

Above table 4 compares precision, recall and f1 score values of all the models for class 0 and 1. By examining all the values it can be concluded that there is a good balance between precision and recall values of all the models. All models are good at detecting fraudulent transactions correctly. Comparatively tree based algorithms such as Decision Tree, Random Forest, XGBoost and LightGBM have received the highest evaluation metric results. The CatBoost model is less accurate than others.

7. Conclusion and future scope

Conclusion

It has been found that machine learning has played a vital role in finding suspicious transactions. The Decision Tree, Random Forest, XGBoost and LightGBM algorithms are proven best for detecting fraudulent credit card transactions due to highest accuracy among the rest of all algorithms such as Logistic Regression, Adaboost and CatBoost utilized during this research. During this research, it has been observed that python and machine learning libraries played vital roles in Data Analysis, Visualization, Preprocessing and Model Development phases.

Future Scope

The other oversampling and scaling techniques available can be utilized on the same dataset and then it can be used with these algorithms to train and examine how they will perform. On the other hand, the other algorithms can also be applied by considering their pros and cons and to assess their performance on the same dataset.

These highly accurate models can be used to create end-to-end applications by deploying them on the cloud, allowing anyone to use them in real-time. To achieve this, datasets with different features and from various sources can be ingested batchwise to train the models. In this research, the dataset used had already undergone PCA on multiple columns. Therefore, a concern in a production environment is the format in which users will provide the inputs.

References

1. Preda G. Credit card fraud detection predictive models [Internet]. Kaggle; 2024 [cited 2025 Oct 4]. Available from: https://www.kaggle.com/code/gpreda/credit-card-fraud-detection-predictive-models
2. Ileberi E, Sun Y, Wang Z. A machine learning based credit card fraud detection using the GA algorithm for feature selection. J Big Data. 2022;9:24. https://doi.org/10.1186/s40537-022-00573-8
3. Dornadula VN, Geetha S. Credit Card Fraud Detection using Machine Learning Algorithms. Procedia Comput Sci. 2019;165:631–41. https://doi.org/10.1016/j.procs.2020.01.057
4. Scikit-learn. LogisticRegression [Internet]. [cited 2025 Oct 4]. Available from: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html
5. Scikit-learn. RandomForestClassifier [Internet]. [cited 2025 Oct 4]. Available from: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
6. Scikit-learn. Decision Trees [Internet]. [cited 2025 Oct 4]. Available from: https://scikit-learn.org/stable/modules/tree.html
7. Chen T, Guestrin C. XGBoost: A scalable tree boosting system [Internet]. [cited 2025 Oct 4]. Available from: https://xgboost.readthedocs.io/en/stable/
8. XGBoost GitHub repository [Internet]. [cited 2025 Oct 4]. Available from: https://github.com/dmlc/xgboost
9. LightGBM Documentation [Internet]. [cited 2025 Oct 4]. Available from: https://lightgbm.readthedocs.io/en/stable/
10. Scikit-learn. AdaBoostClassifier [Internet]. [cited 2025 Oct 4]. Available from: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html
11. CatBoost Documentation [Internet]. [cited 2025 Oct 4]. Available from: https://catboost.ai/en/docs/
12. Awoyemi JO, Adetunmbi AO, Oluwadare SA. Credit

card fraud detection using machine learning techniques: A comparative analysis. In: 2017 International Conference on Computing Networking and Informatics (ICCNI); 2017 Oct; [place unknown]. IEEE; 2017. p. 1–9.

13. Randhawa K, Loo CK, Seera M, Lim CP, Nandi AK. Credit card fraud detection using AdaBoost and majority voting. IEEE Access. 2018;6:14277–14284.
14. Wadkar PN, Misal S, Mundhe S, Yashomanthan. Analysis of breast cancer dataset and its prediction using machine learning. Int Inst Manag Sci J. 2022;Vol/Issue: [pages unknown]. ISSN: 2347-8039.
15. Shinde S, Patil Y, Wadkar P. Evolution of cybersecurity standards in financial sectors. Bharti Humanit Soc Sci UGC Care Group I J. 2023;84(29):[pages unknown]. ISSN: 0974-0066.
16. Wadkar P, Mundhe S, Misal S, Shinde S, Patil Y. Evolution of cybersecurity standards in financial sectors. Published by Dr. Harisingh Gour University. [Internet]. 2023.
17. Wadkar P, Mundhe S, Misal S. Comparative analysis of different machine learning algorithms used in breast cancer prediction. Educ Soc (Shikshan Aashan Samaj). 2023;47(1):10. ISSN: 2278-6864.
18. Kulkarni AR, Mundhe SD. Data Mining Technique: An Implementation of Association Rule Mining in Healthcare. Int Adv Res J Sci Eng Technol. 2017;4(7):[pages unknown]. ISSN (Online): 2393-8021, ISSN (Print): 2394-1588.
19. Wadkar P, Mundhe S. Exploring the effectiveness of different machine learning algorithms in credit card fraud detection: A comparative study. Sustainable Smart Technol Bus Glob Econ. 2024 Dec. Routledge.