Credit Card Fraud Detection using Machine Learning

Sanmati Marabad
Infosys Limited,
Bangalore, India
sanmatimorbad@gmail.com

Abstract **Due to the rapid growth of the E-Commerce industry, the use of credit cards for online purchases has increased dramatically. In recent years, credit card fraud is becoming a major complication for banks as it has become very difficult for detecting fraud in the credit card system. To overcome this hardship Machine learning plays an eminent role in detecting the credit card fraud in the transactions. Modeling prior credit card transactions with data from ones that turned out to be fraudulent is part of the Card Fraud Detection Problem. In Machine learning the machine is trained at first to predict the output so, to predict the various bank transactions various machine learning algorithms are used. The SMOTE approach was employed to oversample the dataset because it was severely unbalanced. This paper the examines and overview the performance of K-nearest neighbors, Decision Tree, Logistic regression and Random forest, XGBoost for credit card fraud detection. The assignment is implemented in Python and uses five distinct machine learning classification techniques. The performance of the algorithm is evaluated by accuracy score, confusion matrix, f1-score, precision and recall score and auc-roc curve as well.**

Key Words: Fraud Detection, Machine Learning, Logistic regression, KNN, Decision tree, random forest, SMOTE,XGboost.

I. INTRODUCTION

In credit card transactions, fraud is defined as the unlawful and unwanted use of an account by someone who is not the account's authorised user. This misuse, as well as the behaviour of such fraudulent operations, can be investigated in order to reduce it and prevent such occurrences in the future. In simple terms, credit card fraud occurs when a person uses another person's credit card for personal gain while the owner and card-issuing authorities are unaware of the transaction. It is currently one of the most serious risks to enterprises. However, to fight the fraud completely, it is essential to first understand the structure of executing a fraud. Credit card fraudsters opt many numbers of ways to commit fraud. Card fraud occurs when a physical card is stolen or when critical account information is stolen, such as the card account number or other information that must be available to conduct a transaction.

A major challenge in applying Machine Learning to fraud detection is the presence of highly imbalanced dataset. In many publicly available databases, the vast majority of transactions are lawful, with only a small percentage of them being fraudulent. Researchers face a big issue in designing an accurate fraud detection system with fewer fraud transactions compared to legal transactions, allowing them to detect fraudulent behaviour successfully. In our paper, we apply multiple classification approaches such as KNN, Decision Tree, Logistic regression and Random forest, XGBoost. Our aim is to build a classifier which will be able to separate fraud transactions from non-fraud ones. We will be comparing the accuracy and effectiveness of these applied algorithms in detecting fraud transactions.

II. LITERATURE REVIEW

Fraud is defined as an illegal deception intended to gain financial or personal gain. It's a planned conduct that goes against the law or a policy with the goal of gaining unjust financial gain. Data mining applications and adversarial detection are among the strategies used in this domain, according to a comprehensive survey undertaken by Clifton Phua and his colleagues. On a European dataset, classic methods such as Decision Tree, XGboost, random forest, and a mixture of particular classifiers were utilised, resulting in a recall of over 91 percent. Only after balancing the dataset by oversampling the data was high precision and recall achieved.

III. METHODOLOGY

A. Proposed Method

The proposed techniques emphasizes on detecting Credit Card Fraudulent transactions whether it is a genuine/nonfraud or a fraud transaction and the approaches used to separate fraud and non-fraud are KNN, Decision Tree, Logistic regression, XGBoost, Random forest and Finally we will observe which approach is best for detecting credit card frauds.

The system architecture has following steps:

- Import of Necessary Packages
- Read the Dataset
- Exploratory Data Analysis i.e. finding null values, duplicate values etc.
- Selecting Features (X) and the Target (y) columns
- Train Test Split will split the whole dataset into train and test data
- Build the model i.e. Training the model
- Test the model i.e. Model prediction
- Evaluation of the system i.e. Accuracy score, F1- score etc.

The figure(Fig-1) below shows the system architecture diagram.

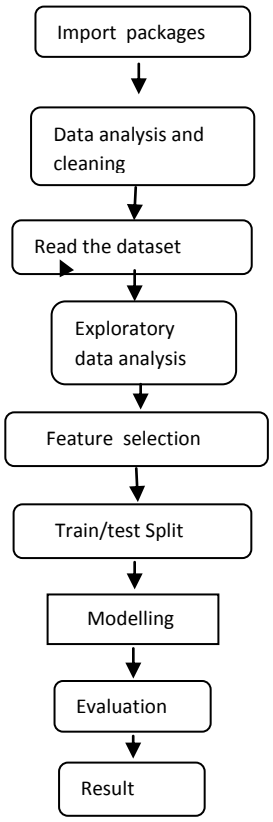


Fig. 1. : Architecture diagram

Machine learning: It is a set of strategies for identifying patterns in data on the fly and then using those patterns to predict future outcomes. Also, provides several algorithms that allow machines to perceive current events and make appropriate judgments based on that perception. It is self-contained and makes its own decisions. Unsupervised learning and supervised learning are the two main types of machine learning.

Supervised Learning: In this technique, both the input and output are known ahead of time. This is known as supervised learning because it learns from a training data set and builds a model from it, which then predicts results when applied to new data. Supervised learning techniques include Decision Trees, Nave Bayes, and others.

Unsupervised Learning: When we have only input data and no corresponding output variable, we call it unsupervised learning. Unsupervised learning's main task is to automatically create class labels. The association between the data can be discovered using unsupervised learning methods to see if they can be grouped together. Clusters are the name for this type of group. Cluster analyses is another term for unsupervised learning. Unsupervised learning techniques include K Means Clustering, KNN, and others.

B. Dataset

In this work, Kaggle's Credit Card Fraud Detection dataset was employed. The transactions in this dataset were made by European cardholders over the period of two days in September 2013. The dataset has 31 numerical features. The PCA transformation of these input variables was performed to keep these data anonymous due to privacy concerns and some of the input variables contained financial information. Three of the listed characteristics were not altered. The "Time" feature shows the amount of time that has passed between the first and subsequent transactions in the dataset. The "Amount" function shows the total amount of credit card transactions. The "Class" feature displays the label and only allows two values: 1 for fraudulent transactions and 0 for all regular transactions. The dataset included 284,807 transactions, 492 of which were fraudulent and the rest were legitimate. When we look at the numbers, we can see that the dataset is severely skewed, with only 0.173 percent of transactions being classified as fraudulent. Preprocessing the data is critical since the distribution ratio of classes plays such an important role in model accuracy and precision. As a result, it is critical to balance the data, which is accomplished using sampling procedures. The Smote technique was used.

Understanding the dataset

Histogram plots and correlation matrix are being used to understand the dataset. Correlation matrix depicts if there is very little or no correlation between individual features and the targeted column. It gives an idea of how features correlate with each other and can help in predicting what features are more relevant for our prediction. We can see that time and amount are correlated features in our data.(*Fig-2*)

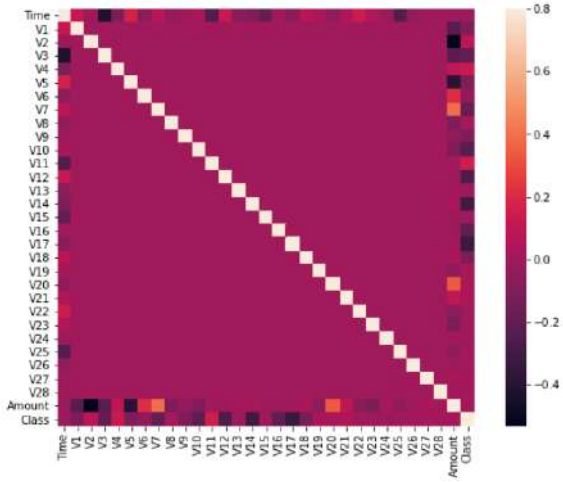
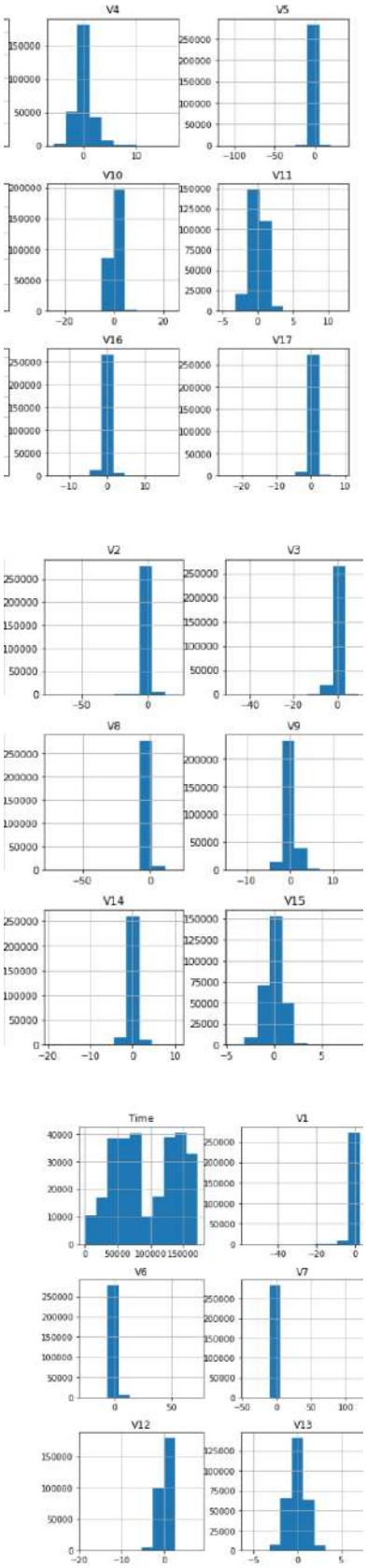


Fig. 2. Correlation matrix of dataset

The Heatmap (Fig-2) clearly shows that the majority of the features do not correlate with one another, but there are a few that have a negative or positive association with one another. The features "V2" and "V5", for example, are substantially negatively linked with the feature "Amount." We can also notice a link between "V20" and "Amount." This allows us to gain a better comprehension of the information. The histogram display allows us to see and understand the frequency distribution of a set of continuous data. It enables data inspection for underlying distribution, outliers, and skewness.

Histogram plot obtained from our dataset are displayed below*(Fig 3)*

We can clearly notice that Time amount and class are relevant features for modelling the dataset



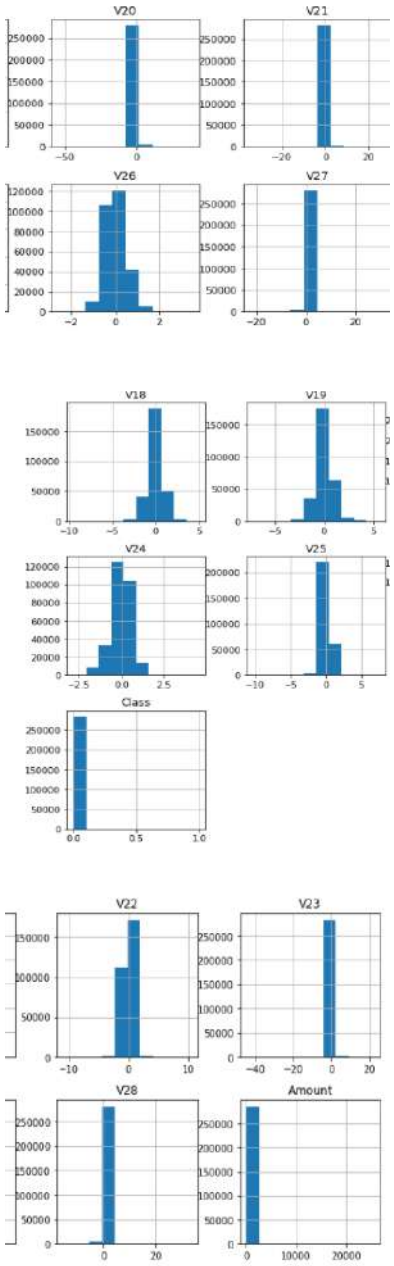


Fig. 3. Histogram plots obtained for dataset

C. Preprocessing

Feature selection is a key strategy for determining which variables in a dataset are the most important. Overfitting can be reduced, accuracy can be improved, and training time can be reduced by carefully selecting useful features and deleting the less critical ones. Techniques like visualisation can help with this.

When working with data that is highly uneven, some type of balancing is essential in order for the model to be trained efficiently. Changing the class distribution involves undersampling the dominant class, oversampling the minority class, or a mix of the two. SMOTE (Synthetic Minority Oversampling Technique) is a well-known oversampling

technique that has been proved to work with unbalanced datasets.

D. Selected algorithm for implementing

1) KNN Algorithm

Various anomaly detection algorithms have exploited the concept of nearest neighbour analysis. Three primary elements influence the performance of the KNN algorithm:

- The distance metric used to locate the nearest neighbors.
- The distance rule that is used to classify k nearest neighbours.
- The fresh sample was classified based on the number of neighbours it had.

2) Decision Tree

The training set is divided into nodes, each of which can contain all or most of one data category. Decision Tree is built by using recursive partitioning to classify the data. Firstly, an attribute is selected and its being the best attribute to split the data. It is split by minimizing the impurity at each step. Impurity of a node is calculated by the entropy of data in the node. Entropy is a measure of uncertainty, in simple words, Entropy of the node is how much random data is in that node. The lower the entropy the purer the node

(a) **Root Node**: It depicts the maximum population of the dataset and this is then split into two or more homogeneous groups.

(b) **Splitting**: It is the splitting or distribution of a node into two or more sub-nodes..

(c) **Decision Node**: The decision node is defined as a sub-node that splits into other sub-nodes.

(d) **Leaf/Terminal Node**: Leaf and Terminal nodes are nodes that do not split.

(e) **Pruning**: The process of eliminating sub-nodes from a decision node is referred to as pruning. Splitting is the polar opposite of pruning.

(f) **Branch / Sub-Tree**: The term "branch" or "sub-tree" refers to a portion of the entire tree.

(g) **Parent and Child Node**: A parent node is referred to as the parent node of sub-nodes, whilst sub-nodes are referred to as the child of a parent nod.

3) Logistic regression

In machine learning, logistic regression is one of the most often used classification techniques. The link between continuous, binary, and categorical predictors is expressed using the logistic regression model. It's also feasible to have binary dependent variables. Based on some forecasts, we can anticipate if something will occur or not. We calculate the probability of belonging to each group for each set of predictors.

4) Random Forest

Random forest is a tree-based technique that involves constructing numerous trees and connecting them with the output to reinforce the model's abilities. It's a supervised learning algorithm as well. The phrase "forest" refers to a collection of decision trees. Simply said, a random forest is a collection of decision trees that helps to solve the problem of overfitting in decision trees. These decision trees are generated at random by selecting random features from a dataset. The random forest arrives at a call decision or forecast that receives the most votes from the decision trees. The random forest considers the end result, which is the result that appears the most number of times via the various decision trees, as the ultimate output.

5) XgBoost

This is decision tree based machine learning algorithm, a supervised learning method. It is an ensemble algorithm which focuses on creating a strong classifier based on weak classifiers. It is used when we have huge number of observations.

IV. EXPERIMENTAL RESULTS

A. Evaluation criteria

To evaluate the results of the classification algorithms there are various parameter such as Accuracy score, classification report, F1-score, confusion matrix etc.

Some important definitions

- True positive(TP)- It is an outcome in which the model accurately predicts the positive class.
- False positive(FP)- It occurs when the positive class is predicted wrongly by the model.
- True negative(TN)- It is an outcome in which the model accurately predicts the negative class.
- False negative(FN)- It is an outcome in which the model predicts the negative class inaccurately.

- **Accuracy**- The number of correct predictions divided by the total number of input samples is known as accuracy.

Accuracy=TP+TN/TP+FN+FN+TN

```
ACCURACY SCORE
----------------------------------------------------------------
Accuracy score of the Decision Tree model is 0.9993679997191109
----------------------------------------------------------------
Accuracy score of the KNN model is 0.9983146659176293
----------------------------------------------------------------
Accuracy score of the Logistic Regression model is 0.9989993328885
----------------------------------------------------------------
Accuracy score of the Random Forest Tree model is 0.99933288859239
----------------------------------------------------------------
Accuracy score of the XGBoost model is 0.9994733330992591
----------------------------------------------------------------
```

- **Confusion Matrix** - It is a table that shows how well a classification model (or "classifier") performs on a set of test data for which the true values are known.

		Actual Values	
		Positive(1)	Negative(0)
Predicted values	Positive(1)	TP	FP
	Negative(0)	FN	TN

Fig. 4. Confusion matrix

- Obtained Confusion matrix for KNN, Logistic regression,Random forest, Xgboost, Decision tree respectively. (Fig.5, fig 6, fig 7, fig 8 fig 9)

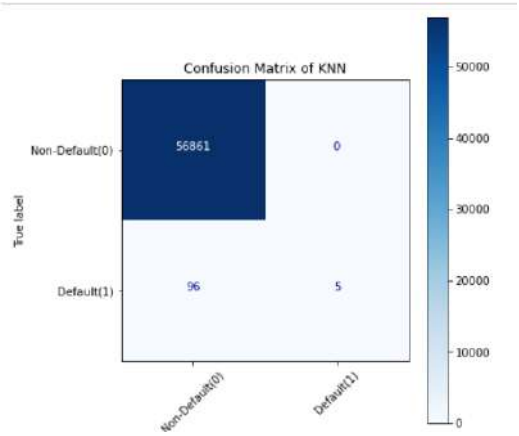


Fig. 5. Confusion matrix for Knn

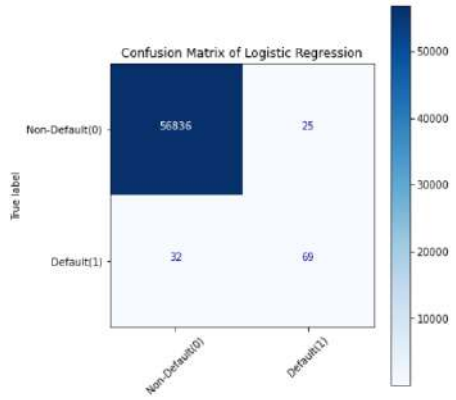


Fig. 6. Confusion matrix for Logistic regression

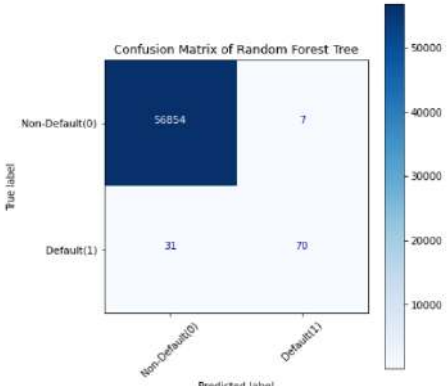


Fig. 7. Confusion matrix for Random forest

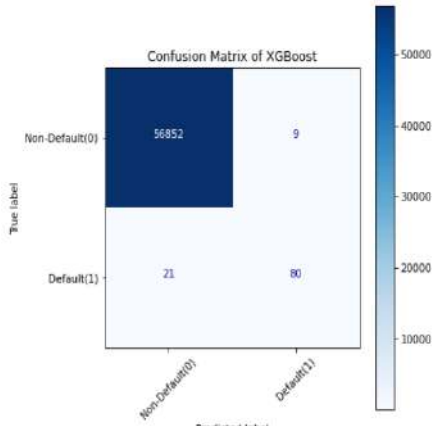


Fig. 8. Confusion matrix for Xgboost

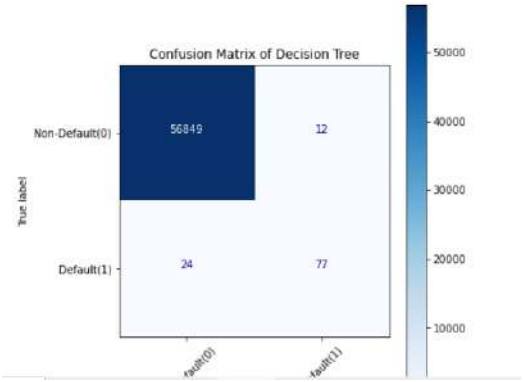


Fig. 9. Confusion matrix for Decision Tree

- **Precision (Specificity**)- It's the number of correct positive outcomes divided by the classifier's projected number of positive finding.

Precision=TP/TP+FP

```
PRECISION
-----------------------------------------------------------------------
Precision score of the Decision Tree model is 0.8651685393258427
-----------------------------------------------------------------------
Precision score of the KNN model is 1.0
-----------------------------------------------------------------------
Precision score of the Logistic Regression model is 0.7340425531914894
-----------------------------------------------------------------------
Precision score of the Random Forest Tree model is 0.9078947368421053
-----------------------------------------------------------------------
Precision score of the XGBoost model is 0.898876404494382
-----------------------------------------------------------------------
```

- **Recall (Sensitivity**) - It's calculated by dividing the number of correct positive results by the total number of relevant samples (all samples that should have been identified as positive).

Recall=TP/TP+FN

```
RECALL
--------------------------------------------------------------------------
Recall score of the Decision Tree model is 0.7623762376237624
--------------------------------------------------------------------------
Recall score of the KNN model is 0.04950495049504951
--------------------------------------------------------------------------
Recall score of the Logistic Regression model is 0.6831683168316832
--------------------------------------------------------------------------
Recall score of the Random Forest Tree model is 0.6831683168316832
--------------------------------------------------------------------------
Recall score of the XGBoost model is 0.7920792079207921
--------------------------------------------------------------------------
```

- **F1- score** - F1 Score is the Harmonic Mean between precision and recall. The range for F1 Score is [0, 1].

F1 Score= 2*(Recall*Precision)/(Recall+ Precision)

```
F1 SCORE
--------------------------------------------------------------------------
F1 score of the Decision Tree model is 0.8105263157894738
--------------------------------------------------------------------------
F1 score of the KNN model is 0.09433962264150944
--------------------------------------------------------------------------
F1 score of the Logistic Regression model is 0.7076923076923077
--------------------------------------------------------------------------
F1 score of the Random Forest Tree model is 0.7796610169491525
--------------------------------------------------------------------------
F1 score of the XGBoost model is 0.8421052631578948
--------------------------------------------------------------------------
```

- **ROC-AUC Curve-** It is a performance metric for classifying issues at various thresholds. It's a probability curve, and the AUC stands for the degree of separation. It expresses the model's capacity to distinguish across classes. The AUC indicates how well the model predicts 0s as 0s and 1s as 1s. TPR is plotted against FPR, with TPR on the y-axis and FPR on the x-axis.
- Terms in ROC-AUC curve
- TPR(true positive rate/recall or sensitivity)

TP/TP+FN

- Specificity

TN/TN+FP

- FPR

1-specificity=FP/TN+FP

- **Obtained Roc curve for Decision tree, knn, logistic regression, random forest, Xgboost.(**Fig 1, Fig 11, Fig 12, Fig 13, Fig 14)

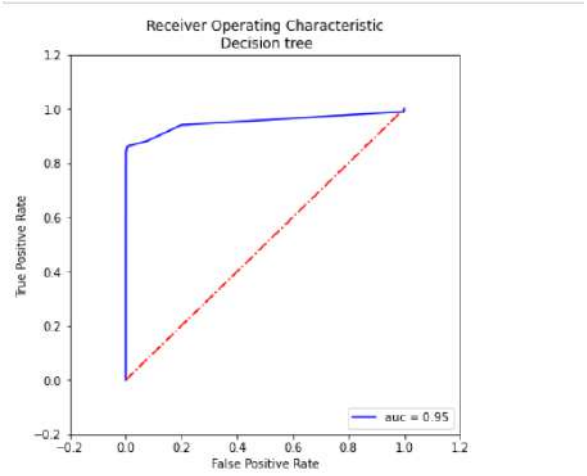


Fig. 10. Roc curve for Decision tree

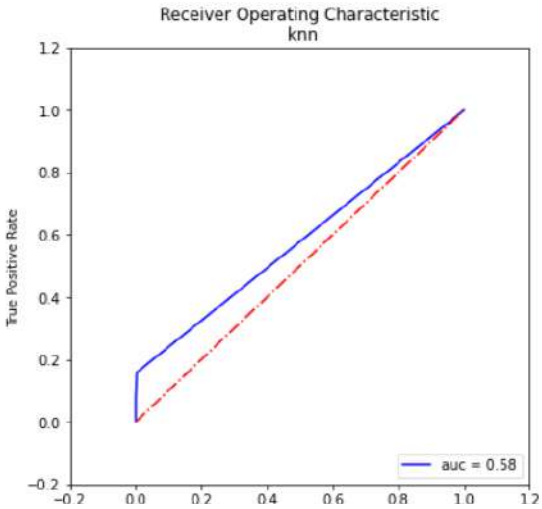


Fig. 11. Roc curve for knn

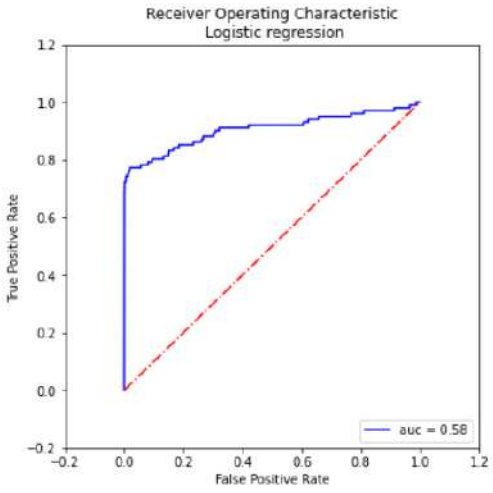


Fig. 12. Roc curve for logistic regression

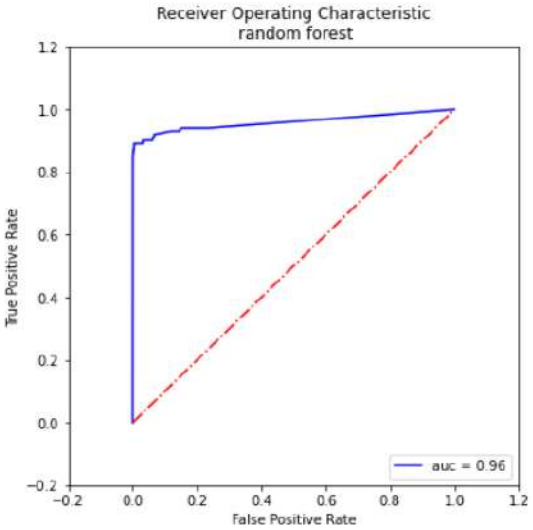


Fig. 13. Roc curve for Random forest

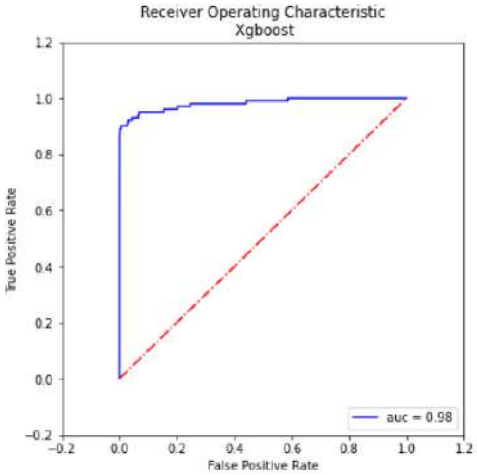


Fig. 14. Roc curve for Xgboost

B. Results

Five machine learning methods were employed to detect fraud in the credit card system in this article. Data from 80% of the training dataset and 20% of the testing dataset were utilised to evaluate the algorithms. Accuracy, F1-score, precision, and recall score are used to analyze the performance these four approaches. As shown in the observations of accuracy outcomes .The accuracy score for KNN, Decision tree, Logistic Regression and Random forest ,KNN. Xg-boost each algorithm is great. But as we look to the other 3 criteria, we can clearly see that the Xgboost and decision tree classifiers outruns all the above classifier and predicts the fraudulent transaction with impressive F1 score, precision and recall score.

V. CONCLUSION

Credit card fraud is a significant commercial issue. These types of scams can result in significant personal and business losses. As a result, businesses are investing an increasing amount of money in creating new concepts and methods for detecting and limiting fraud. The major purpose of this article was to look at a variety of machine learning algorithms for detecting fraudulent transactions. As a consequence of the comparison, it was discovered that the Xgboost algorithm produces the best results, i.e. best classifies whether transactions are fraudulent or not. This was determined using a variety of metrics, including recall, accuracy, and precision, the f1 score, and the AUC-roc curve. For this type of situation, having a high recall value is crucial. It has been established that feature selection and dataset balancing are critical in achieving significant results. Other machine learning techniques, such as evolutionary algorithms and various forms of stacked classifiers, as well as rigorous feature selection, should be studied further to improve results.

REFERENCES

[1] Machine Learning For Credit Card Fraud Detection System, Lakshmi S V S , Selvani Deepthi Kavila, november 2018.

[2] Credit Card Fraud Detection using Data science and Machine learning, S P Maniraj, Aditya Saini , Shadab Ahmed, Swarna Deep Sarkar, September 2019.

[3] A. Mishra, C. Ghorpade, “Credit Card Fraud Detection on the Skewed Data Using Various Classification and Ensemble Techniques” 2018 IEEE International Students' Conference on Electronics ,Electrical and Computer Science (SCEECS) pp. 1-5. IEEE.

[4] S. V. S. S. Lakshmi, S. D. Kavilla “Machine Learning For Credit Card Fraud Detection System”, unpublished [7] N. Malini, Dr. M. Pushpa, “Analysis on Credit Card Fraud Identification Techniques on the basis of KNN and Outlier Detection“, Advances in Electrical, Electronics, Information, Communication and BioInformatics (AEEICB), 2017 Third International Conference on pp. 255- 258. IEEE.

[5] C. Wang, Y. Wang, Z. Ye, L. Yan, W. Cai, S. Pan, “Credit card fraud detection on basis of whale algorithm optimized BP neural network”, 2018 13th International Conference on Computer Science & Education (ICCSE) pp. 1-4. IEEE.