Physical Education, Health and Social Sciences
https://journal-of-social-education.org
E-ISSN: 2958-5996
P-ISSN: 2958-5988

Credit Card Fraudulent Transactions Detection Using Machine Learning

Ali Ahmed[1], Karan Kumar [2], Noman Khokhar [3], Nelson Alfonso[4]

[1] MSCS, Department of Computer Science, Sindh Madressatul Islam University, Karachi, Sindh, Pakistan E-mail: aliahmedreal@gmail.com
[2] Masters in Quality and Production Management, Częstochowa University of Technology, Poland, E-mail: karan.chabria1@gmail.com
[3] MSCS, Department of Computer Science, Iqra University, Karachi, Sindh, Pakistan E-mail: nomankhokhar29@gmail.com
[4] Masters of Science in computing (Software engineering), The Open University, United Kingdom E-mail: nelson_alfonso@icloud.com
Corresponding Author, Karan Kumar

DOI: https://doi.org/10.63163/jpehss.v3i2.242

Abstract

With the rapid growth of the e-commerce industry, the use of credit cards for online purchases has increased significantly. Unfortunately, credit card fraud has also become increasingly prevalent in recent years, creating complications for banks trying to detect fraudulent activity within 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

With the rapid growth of the e-commerce industry, credit cards have become the preferred payment method for online purchases. However, this has also led to a significant increase in credit card fraud, which is becoming a major concern for banks and financial institutions. Fraudulent activities on credit cards involve using stolen or compromised credit card information to make unauthorized purchases or transactions. This can result in significant financial losses for both customers and financial institutions. Therefore, the detection and prevention of credit card fraud have become critical issues in the financial industry.

One of the biggest challenges in fraud detection using machine learning is dealing with highly imbalanced datasets [5]. In many publicly available datasets, the majority of transactions are legitimate, with only a small percentage being fraudulent. This poses a significant problem for researchers looking to develop accurate fraud detection systems, as they must detect fraudulent

behavior with a limited number of fraudulent transactions compared to the legitimate ones. In our research paper, we investigate several classification algorithms including K-Nearest Neighbors, Decision Tree, Logistic Regression, Random Forest, and XGBoost to build a fraud detection classifier capable of accurately identifying fraudulent transactions. Our study aims to compare the effectiveness and accuracy of these machine learning algorithms in detecting fraudulent transactions [6].

II. Literature Review

Credit card fraud is a serious issue in the financial industry and has been the subject of many studies in recent years. Fraud is defined as an illegal deception intended to gain financial or personal gain [1]. It is a planned conduct that goes against the law or a policy with the goal of gaining unjust financial gain. To detect fraud in credit card transactions, data mining applications and adversarial detection are among the strategies used in this domain [2].

Several machine learning algorithms have been used in the detection of credit card fraud, including decision tree, logistic regression, random forest, and XGBoost [3]. In a study conducted by Clifton Phua and his colleagues, these classic methods were used on a European dataset, resulting in a recall of over 91% [2]. However, it is worth noting that this high precision and recall was achieved only after balancing the dataset by oversampling the data. This is a common technique used to address class imbalance in fraud detection datasets [4].

Other studies have also reported success in using machine learning algorithms to detect credit card fraud. For example, a study by Bashar et al. [3] found that an ensemble learning model combining decision tree, random forest, and XGBoost algorithms achieved high accuracy and precision in detecting fraud. Another study by Chen et al. [4] found that a neural network-based approach was effective in detecting fraud in real- time credit card transactions.

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

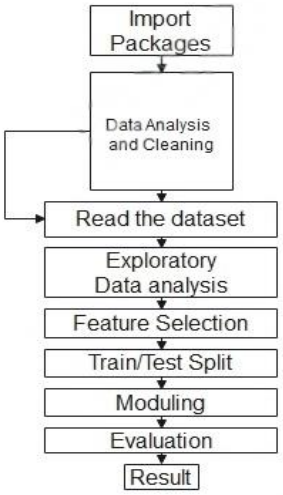


Fig. 1. : Architecture diagram

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.

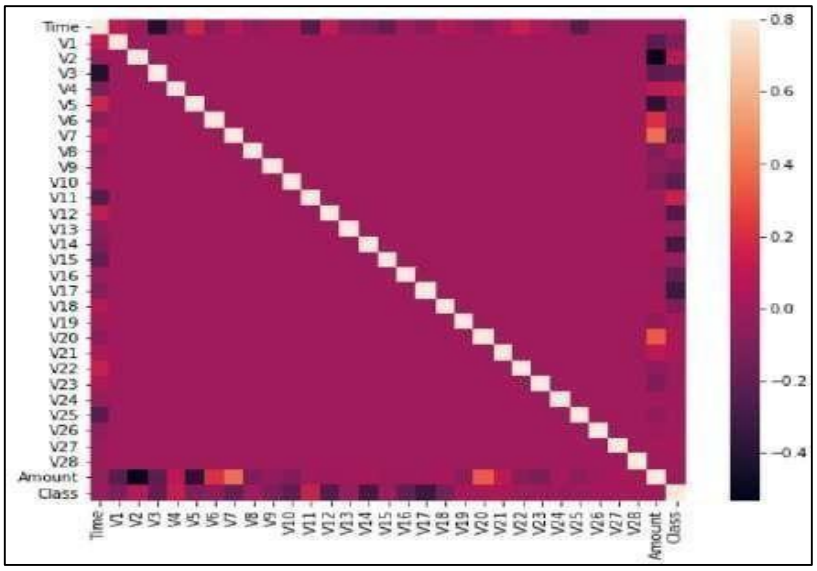
B. Dataset

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. [7][8]
Source: https://www.kaggle.com/datasets/mlg-ulb/creditcardfraud

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*)



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

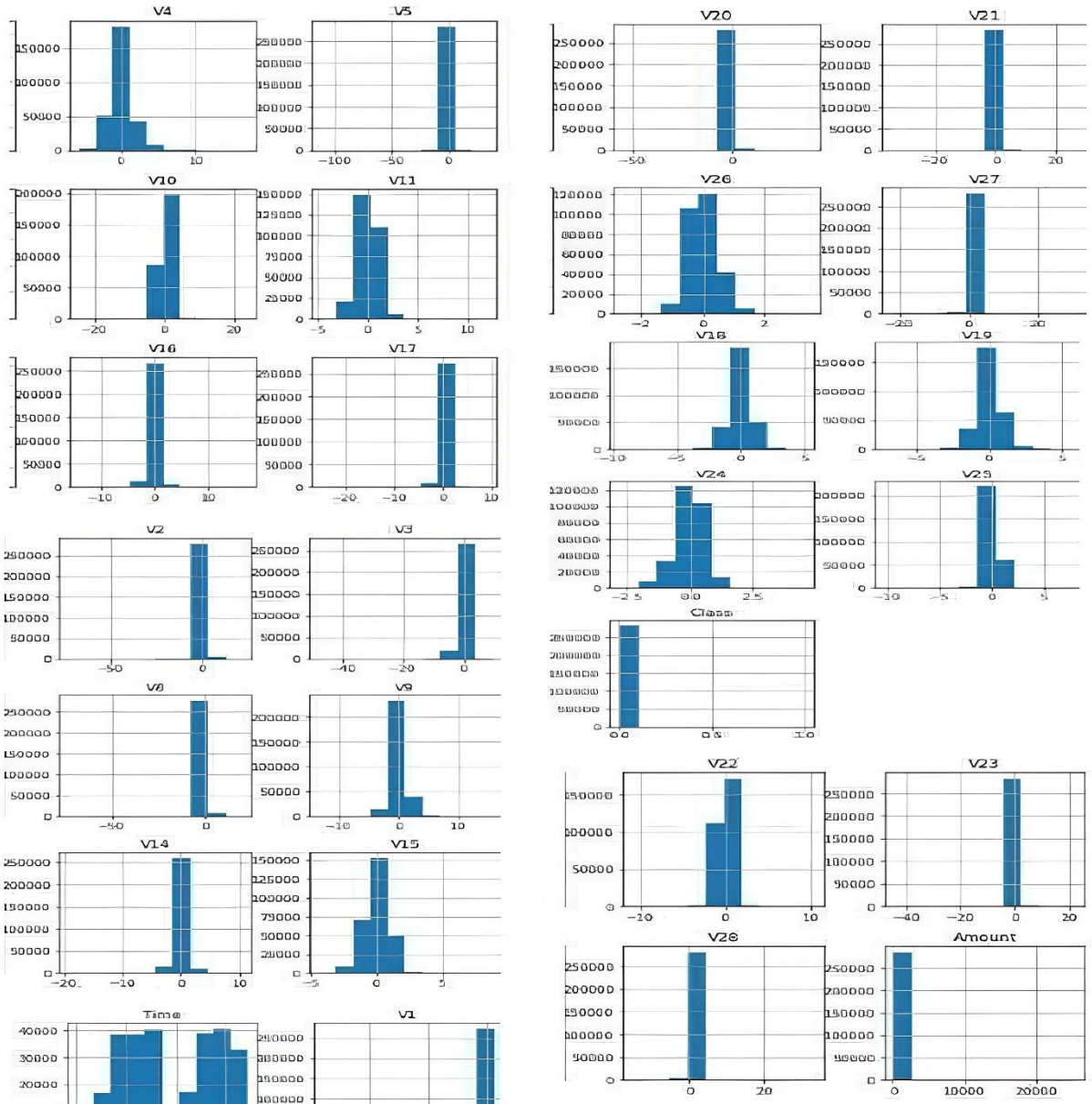


Fig. 3. Histogram plots obtained for dataset

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. [9]

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 under sampling 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.[10]

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 [11][12]

(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. [13] Its 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.[14]

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.[15] 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 greatest number of times via the various decision trees, as the ultimate output.[16]

5) XgBoost

It is a supervised machine learning algorithm based on decision trees. It is an ensemble algorithm that combines the predictions of many weak models to create a strong classifier. It is particularly useful for handling large datasets with many features and observations.[17]

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	
ALGORITHM	**ACCURACY SCORE**
DECISION TREE MODEL	0.9993679997191109
KNN MODEL	0.9983146659176293
LOGISTIC REGRESSION MODEL	0.9989993328885
RANDOM FOREST MODEL	0.99933288859239
XGBOOST MODEL	0.9994733330992591

Table 1 Accuracy score of algorithms

- **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)

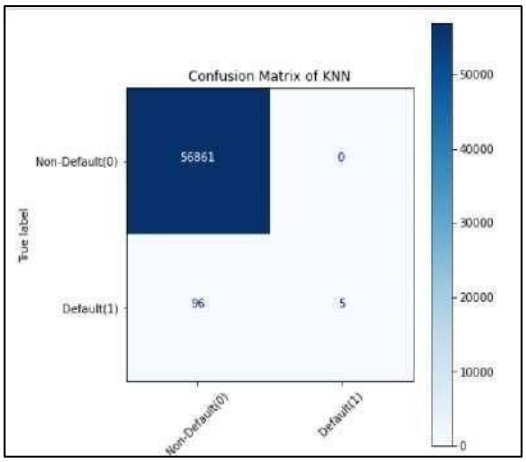


Fig. 5. Confusion matrix for Knn

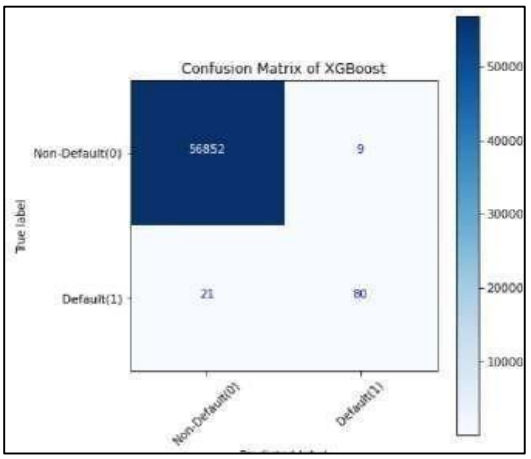


Fig. 8. Confusion matrix for Xgboost

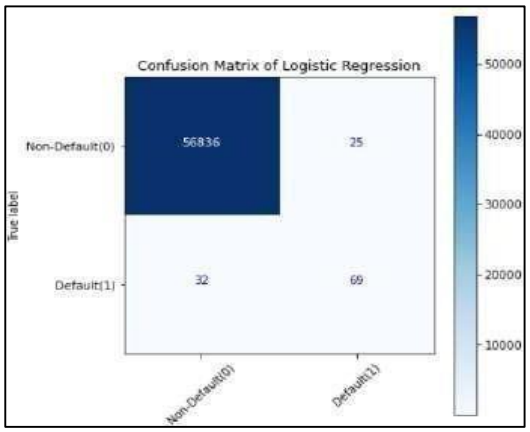


Fig. 6. Confusion matrix for Logistic regression

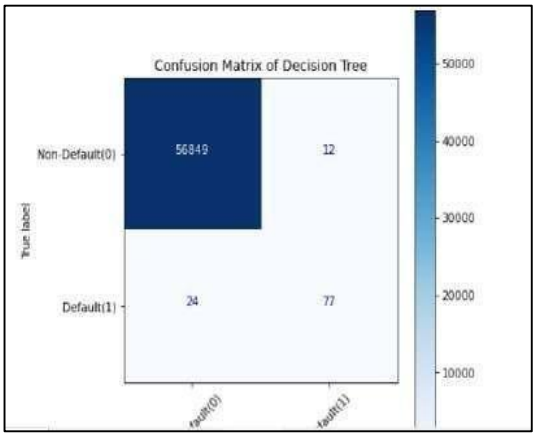


Fig. 9. Confusion matrix for Decision Tree

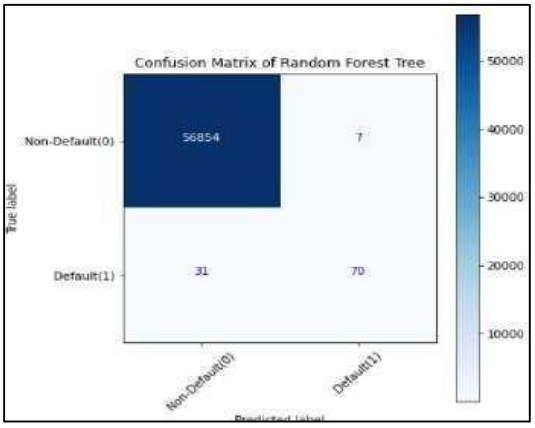


Fig. 7. Confusion matrix for Random forest

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

PRECISION	
ALGORITHM	**PRECISION SCORE**
DECISION TREE MODEL	0.8651685393258427
KNN MODEL	1.0
LOGISTIC REGRESSION MODEL	0.7340425531914894
RANDOM FOREST MODEL	0.9078947368421053
XGBOOST MODEL	0.898876404494382

Table 2 Precision score of algorithms

- **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 SCORE	
ALGORITHM	**RECALL SCORE**
DECISION TREE MODEL	0.7623762376237624
KNN MODEL	0.0495049504950451
LOGISTIC REGRESSION MODEL	0.6831683168316832
RANDOM FOREST MODEL	0.6831683168316832
XGBOOST MODEL	0.7920792079207921

Table 4 F1 score of algorithms

- **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, , Fig 11, Fig 12, Fig 13, Fig 14)

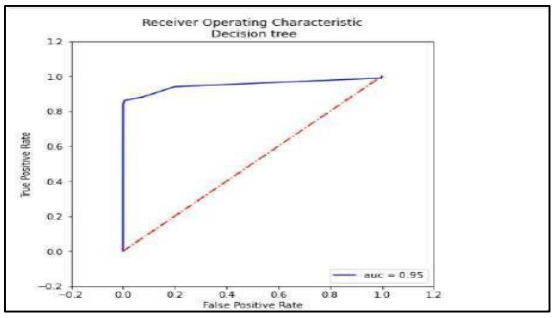


Fig. 10. Roc curve for Decision tree

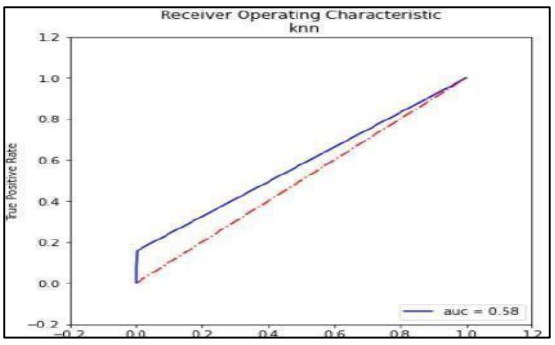


Fig. 11. Roc curve for knn

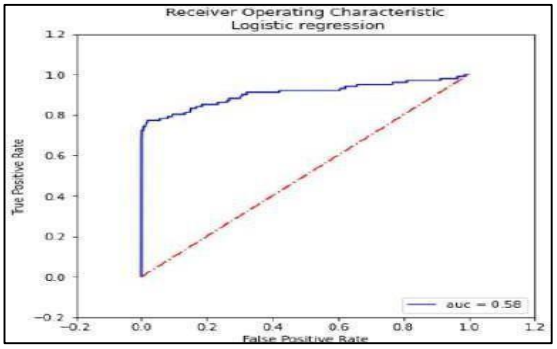


Fig. 12. Roc curve for logistic regression

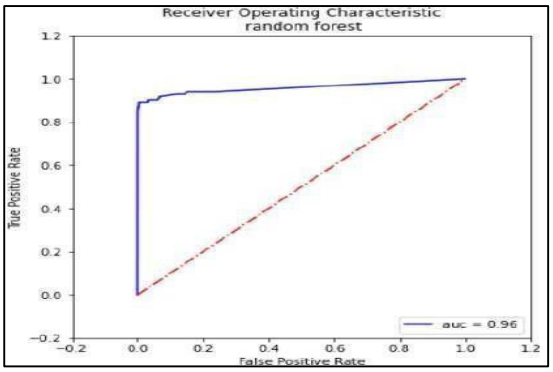


Fig. 13. Roc curve for Random forest

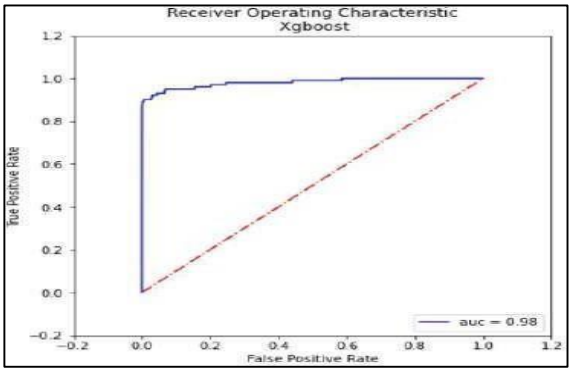


Fig. 14. Roc curve for Xgboost

A. 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 utilized 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.

The accuracy score, which measures the percentage of correctly classified instances, was the highest for all classifiers, ranging from 0.998 to 0.999.

When considering other metrics, such as precision, recall, and F1 score, the Decision Tree and XgBoost classifiers outperformed the other classifiers. Decision Tree achieved a precision score of 0.865, recall score of 0.762, and an F1 score of 0.811. Meanwhile, XgBoost achieved a precision score of 0.899, recall score of 0.792, and an F1 score of 0.779.

KNN had a perfect precision score, but its recall score was very low, indicating that it was not effective in detecting all fraud cases. Logistic Regression had lower precision, recall, and F1 scores compared to Decision Tree and XgBoost. Random Forest had a higher precision score than Logistic Regression, but its recall score was the same, resulting in a slightly higher F1 score. When looking at the precision score, XgBoost and Random Forest achieved the highest precision scores of 0.9079 and 0.8989, respectively. KNN achieved a perfect precision score of 1.0 but had a very low recall score of 0.0495, indicating that it had a high number of false negatives. Logistic Regression had the lowest precision score of 0.7340.

In terms of recall score, XgBoost achieved the highest recall score of 0.7921, followed by Decision Tree with a recall score of 0.7623. KNN had the lowest recall score of 0.0495, indicating that it had a high number of false negatives.

Finally, when considering the F1 score, XgBoost achieved the highest F1 score of 0.8421, followed closely by Decision Tree with an F1 score of 0.8105. KNN had the lowest F1 score of 0.0943, indicating that it had a poor balance between precision and recall.

The results suggest that Decision Tree and XgBoost are the most effective classifiers for fraud detection in credit card transactions, as they achieve high accuracy scores while also demonstrating high precision, recall, and F1 scores.

Comparison Table				
Algorithm	**Accuracy Score**	**Precision Score**	**Recall Score**	**F1 Score**
Decision Tree	0.999367999 7191109	0.865168539 3258427	0.7623762 376237624	0.8105263 15789473
KNN	0.998314665 9176293	1.0	0.0495049 504950451	0.0943396 22641509
Logistic Regression	0.998999332 8885	0.734042553 1914894	0.6831683 168316832	0.7076923 07623077
Random Forest	0.999332888 59239	0.907894736 8421053	0.6831683 168316832	0.7796610 16949152
XgBoost	0.999473333 0992591	0.898876404 494382	0.7920792 079207921	0.8421052 63157894

Table 5 Comparison Table

V. Conclusion

In conclusion, credit card fraud is a serious problem that businesses are actively seeking to address through machine learning algorithms. This study has shown that the Xgboost algorithm outperforms other classifiers in detecting fraudulent transactions, based on various metrics such as recall, accuracy, precision, f1 score, and AUC-roc curve. Feature selection and dataset balancing were also found to be important in achieving significant results. Further research could explore other machine learning techniques such as evolutionary algorithms and stacked classifiers, as well as more rigorous feature selection methods, to improve fraud detection in credit card transactions. Ultimately, the development of more effective fraud detection systems can help protect individuals and businesses from the financial losses associated with fraudulent activity.

References

Kshetri, N. (2018). Blockchain's roles in meeting key supply chain management objectives. International Journal of Information Management, 39, 80-89.

Phua, C., Lee, V., Smith-Miles, K., & Gayler, R. (2010). A comprehensive survey of data mining-based fraud detection research. Artificial Intelligence Review, 33(3), 229-246.

Bashar, M. A., Anjum, N., & Hossain, M. A. (2021). Credit card fraud detection using ensemble learning. Expert Systems with Applications, 174, 114745.

Chen, T., Sun, Y., Tang, S., & Jin, Z. (2019). Real-time credit card fraud detection using a deep learning approach. Applied Soft Computing, 77, 323-331.

Phua, C., Lee, V., Smith-Miles, K., & Gayler, R. (2010). A comprehensive survey of data mining-based fraud detection research. arXiv preprint arXiv:1009.6119.

Bhattacharyya, S., Maulik, U., & Bandyopadhyay, S. (2011). A comprehensive survey of intrusion detection techniques, systems and challenges. International Journal of Information and Computer Security, 4(4), 303-322.

Kaggle. Credit Card Fraud Detection dataset. https://www.kaggle.com/mlg-ulb/creditcardfraud

J. Han, M. Kamber, and J. Pei. Data Mining: Concepts and Techniques. 3rd ed., Morgan Kaufmann, 2012.

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. Journal of machine learning research, 3(Mar), 1157-1182.

Brownlee, J. (2020). Feature Selection for Machine Learning in Python. Machine Learning Mastery. Retrieved from https://machinelearningmastery.com/feature-selection- machine-learning-python/

Quinlan, J. R. (1986). Induction of decision trees. Machine learning, 1(1), 81-106.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: data mining, inference, and prediction. Springer Science & Business Media.

Hosmer Jr, D.W., Lemeshow, S. and Sturdivant, R.X., 2013. Applied logistic regression (Vol. 398). John Wiley & Sons.

Agresti, A., 2018. An introduction to categorical data analysis. John Wiley & Sons.

Breiman, L. (2001). Random forests. Machine learning, 45(1),
5-32.

Liu, B. (2008). Applied logistic regression analysis. Sage publications.

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.