**Financial Fraud Detection Using Machine Learning Algorithms**

**Abstract**

Financial fraud can cause large financial losses and is a serious threat to the stability of financial systems. This research focuses on the use of machine learning (ML) classification methods for the identification of financial fraud in transactional data. The main goal is to create a reliable and accurate binary classification system that can distinguish between transactions that are fraudulent and those that are real.

Various machine learning algorithms, such as logistic regression and random forests are used in this research. The selection of these algorithms enables a thorough assessment of how well they identify fraudulent activity. To improve the performance of the models and extract relevant information from the transactional data, feature engineering and selection techniques will be used.

The dataset consists of historical transactional data, annotated with information on whether or not each transaction is fraudulent. The project's success will be determined by how well the machine learning algorithms are able to identify transactions while reducing the number of false positives and false negatives. By giving financial institutions a useful tool to recognize and reduce the risks connected with fraudulent transactions, the developed system hopes to improve the general safety and reliability of financial networks.

**Keywords**

Machine Learning, Logistic Regression, Random Forest.

**Introduction**

India has lost at least Rs. 100 crore every day due to bank frauds or scams over the past seven years. A fraudulent transaction is an unauthorized or illegal activity involving the use of payment instruments or financial systems, typically for the purpose of obtaining money, goods, or services without proper consent or authorization from the account holder[1]. Credit card frauds are easy and friendly targets.

E-commerce and many other online sites have increased the online payment modes, increasing the risk for online frauds. Increase in fraud rates, researchers started using different machine learning methods to detect and analyse frauds in online transactions [2]. With different frauds mostly credit card frauds, often in the news for the past few years, frauds are in the top of mind for most the world’s population. Fraud detection dataset is highly imbalanced because there will be more legitimate transaction when compared with a fraudulent one[2].

Machine learning algorithms such as Logistic Regression, Decision Tree Classifiers, Random Forest Classifiers and Support Vector Machines are widely used for classification of fraudulent transactions. This study aims to compare three classification and prediction techniques such as Logistic Regression and Random Forest in classifying financial transactions as either fraudulent or not fraudulent.

**Literature Review**

One approach used to predict a binary outcome variable is logistic regression. This method does not require that the explanatory variables be correlated or have a normal distribution. In this model the targe variable is qualitative. Numbers or categories can be used as examples of explanatory variables. Many researchers have used logistic regression to identify bankruptcy in the banking sector. However, the datasets used were highly imbalanced resulting in low precision and recall scores [3].

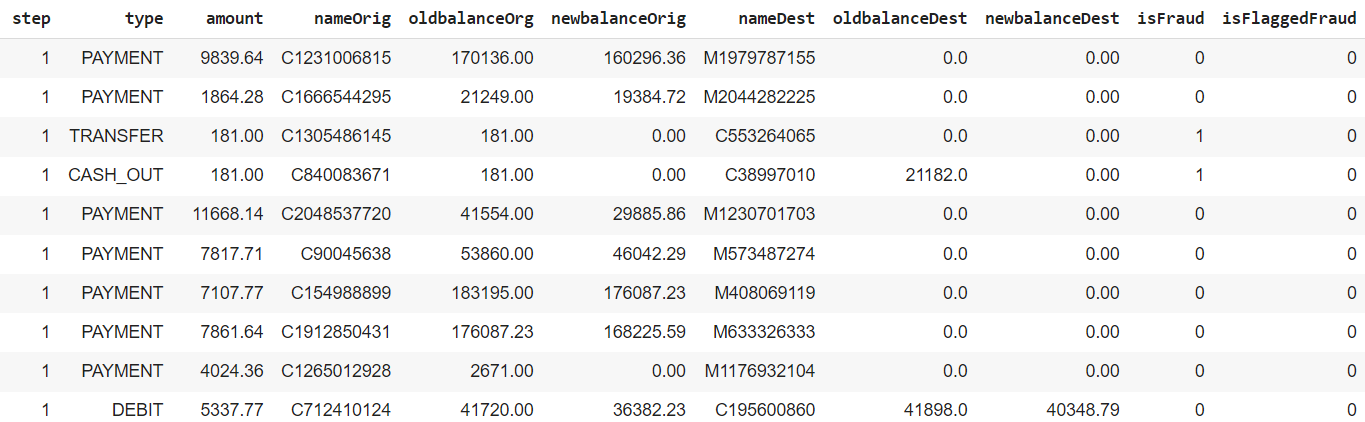
In a different study, a novel approach to fraud detection was applied, grouping customers according to their transactions and using behavioural patterns to create a profile for each cardholder. Following the application of various classifiers to three distinct groups, rating scores for each kind of classifier were produced. The system is able to promptly adjust to new cardholder transaction behaviors due to these dynamic parameter changes [2].

In another approach, in order to detect fraud, a variety of supervised and semi-supervised machine learning techniques were employed. However, the primary goals of the dataset related to card frauds were to address three main issues: a significant class imbalance, the inclusion of labelled and unlabelled samples, and the need to process a higher volume of transactions [4].

**Data and methods**

**Data**

The dataset comprises 30 days of hourly transactions, categorized into types such as cash-in, cash-out, debit, payment, and transfer, each with corresponding amounts in local currency. Transactions involve two parties, the initiator and recipient, with initial and new balances. The 'isFraud' label flags transactions by fraudulent agents attempting to take control of accounts and deplete funds. Additionally, the 'isFlaggedFraud' flag identifies attempts to transfer over 200,000 in a single transaction, as part of the model's effort to prevent large-scale illegal transfers. The dataset offers insights into financial interactions and facilitates the analysis of fraudulent activities in the simulated environment.



**Variable description**

* step - maps a unit of time in the real world. In this case 1 step is 1 hour of time. Total steps 744 (30 days simulation).
* type - CASH-IN, CASH-OUT, DEBIT, PAYMENT and TRANSFER.
* amount - amount of the transaction in local currency.
* nameOrig - customer who started the transaction
* oldbalanceOrg - initial balance before the transaction
* newbalanceOrig - new balance after the transaction
* nameDest - customer who is the recipient of the transaction
* oldbalanceDest - initial balance recipient before the transaction. Note that there is not information for customers that start with M (Merchants).
* newbalanceDest - new balance recipient after the transaction. Note that there is not information for customers that start with M (Merchants).
* isFraud - This is the transactions made by the fraudulent agents inside the simulation. In this specific dataset the fraudulent behavior of the agents aims to profit by taking control or customers accounts and try to empty the funds by transferring to another account and then cashing out of the system.
* isFlaggedFraud - The business model aims to control massive transfers from one account to another and flags illegal attempts. An illegal attempt in this dataset is an attempt to transfer more than 200.000 in a single transaction.

**Methods**

**1. Logistic Regression**

Logistic Regression is a supervised machine learning algorithm primarily used for binary classification problems, although it can be extended to handle multiclass classification. Logistic Regression is a linear model that estimates the probability of an instance belonging to a particular class.

**Key Characteristics**

* Linear Model:

Logistic Regression models the relationship between the input features and the probability of belonging to a specific class through a linear combination of the features. It applies the logistic function to this linear combination to obtain probabilities.

* Sigmoid Function:

The logistic function (sigmoid) is employed to map the linear combination of features to a range between 0 and 1. This output represents the probability of the instance belonging to the positive class.

Where,

e is the base of natural logarithms,

x is the numerical value one wishes to transform.

**Advantages**

* Interpretability: Logistic Regression provides interpretable results, allowing for a clear understanding of the impact of each feature on the predicted probability.
* Efficiency: The algorithm is computationally efficient and performs well on relatively simple datasets.
* Probabilistic Output: Logistic Regression outputs probabilities, making it suitable for tasks where understanding the confidence of predictions is important.

**Limitations**

* Linearity: Logistic Regression assumes a linear relationship between features and the log-odds of the output, which may limit its ability to capture complex patterns.
* Binary Classification: While logistic regression is designed for binary classification, techniques like one-vs-all can be used for multiclass problems.

**2. Random Forest Classifier**

Random Forest is a supervised ensemble learning algorithm that operates in both classification and regression domains. It is designed to enhance predictive accuracy and robustness by aggregating the predictions of multiple decision trees. The algorithm was introduced by Leo Breiman and Adele Cutler, and it is widely used for its versatility and effectiveness in handling complex datasets.

**Key Characteristics**

* Ensemble Learning: Random Forest is based on the ensemble learning principle, which involves combining the predictions of multiple models to improve overall performance. In the case of Random Forest, the individual models are decision trees.
* Decision Trees: The fundamental building blocks of Random Forest are decision trees. Decision trees are constructed recursively by splitting the dataset based on feature values, resulting in a tree-like structure that represents decision rules.
* Random Subsets (Bootstrapping): Random Forest introduces an element of randomness by training each decision tree on a random subset of the original dataset. This process, known as bootstrapping, involves sampling with replacement, ensuring diversity among the trees.
* Feature Randomness: In addition to sampling data points, Random Forest further introduces randomness by considering only a random subset of features at each split when constructing individual trees. This helps in reducing correlation between the trees.

**Advantages**

* Robustness: Random Forest is less prone to overfitting, thanks to the ensemble of diverse trees and the randomness introduced during both bootstrapping and feature selection.
* High Predictive Accuracy: By aggregating predictions from multiple trees, Random Forest often achieves high predictive accuracy and generalizes well to new, unseen data.
* Versatility: The algorithm can handle both classification and regression tasks, making it suitable for a wide range of machine learning problems.

**Evaluation metrics**

An evaluation metric is a quantitative measure used to assess the performance of a machine learning model, algorithm, or system in solving a particular task or problem. These metrics provide a numerical way to gauge how well the model is performing and help in comparing different models or tuning hyperparameters. The choice of evaluation metrics depends on the nature of the task, whether it is a classification, regression, clustering, or another type of problem.

Evaluation metrics for classification are as given below.

**Accuracy**

Accuracy is a commonly used metric for evaluating the overall performance of a classification model. It measures the proportion of correctly classified instances out of the total number of instances. The formula for accuracy is as follows:

**Confusion matrix**

A confusion matrix is a table that is often used to describe the performance of a classification model on a set of data for which the true values are known. It provides a detailed breakdown of the model's predictions, highlighting the instances of correct and incorrect classifications. The confusion matrix is particularly useful for evaluating the performance of a classification algorithm, especially in situations where there are multiple classes. It is of the form:

|  |  |
| --- | --- |
| TP | FP |
| FN | TN |

True Positive (TP): Instances where the model correctly predicts the positive class.

False Positive (FP): Instances where the model incorrectly predicts the positive class (false alarm or Type I error).

False Negative (FN): Instances where the model incorrectly predicts the negative class (miss or Type II error).

True Negative (TN): Instances where the model correctly predicts the negative class.

From the confusion matrix we can derive the following:

**Classification Report**

A classification report is a summary of the performance of a classification model, providing a comprehensive overview of various evaluation metrics for each class in the dataset. It is particularly useful for multi-class classification problems, where the goal is to classify instances into more than two classes. The classification report typically includes metrics such as precision, recall, F1 score, and support for each class.

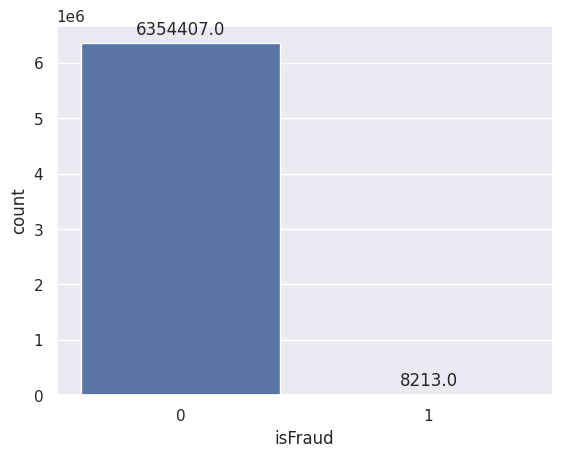
Support is the number of actual occurrences of the class in the specified dataset. It represents the number of instances belonging to each class.

**Experimental Results**

1. Data Preprocessing:

During the initial stages of the experiment, comprehensive data preprocessing was conducted to ensure the quality of the dataset. This involved meticulous cleaning, including handling missing values and removing outliers. Exploratory Data Analysis (EDA) was performed to gain valuable insights into the underlying patterns within the data.

Below is the countplot showing class imbalance in the target variable.



Additionally, feature engineering was undertaken, with careful consideration given to addressing multicollinearity using the Variance Inflation Factor (VIF).

Variable VIF

0 step 1.191094

1 amount 4.078952

2 oldbalanceOrg 544.053103

3 newbalanceOrig 546.346008

4 oldbalanceDest 72.895522

5 newbalanceDest 84.584619

6 isFraud 1.188386

7 isFlaggedFraud 1.002546

2. Logistic Regression Model:

The logistic regression model was constructed following the preprocessing steps. While the model demonstrated an impressive overall accuracy of 97%, closer inspection revealed significant challenges in terms of precision and recall, particularly for instances belonging to label 1. This highlighted a limitation in the model's ability to correctly identify positive instances.

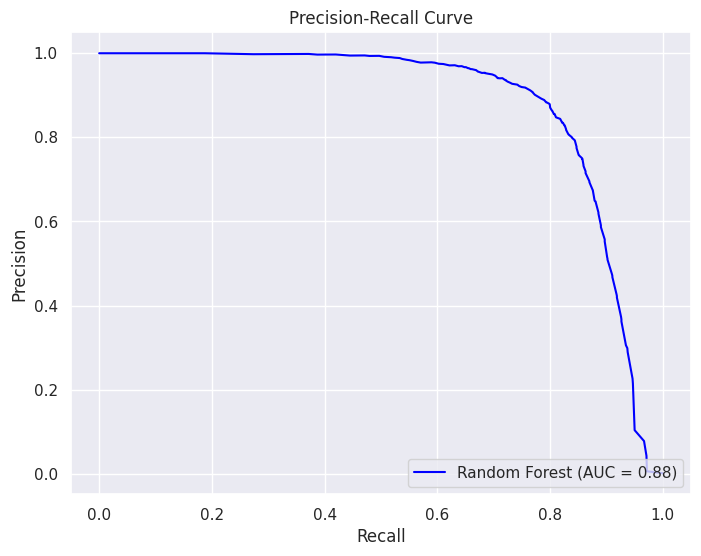
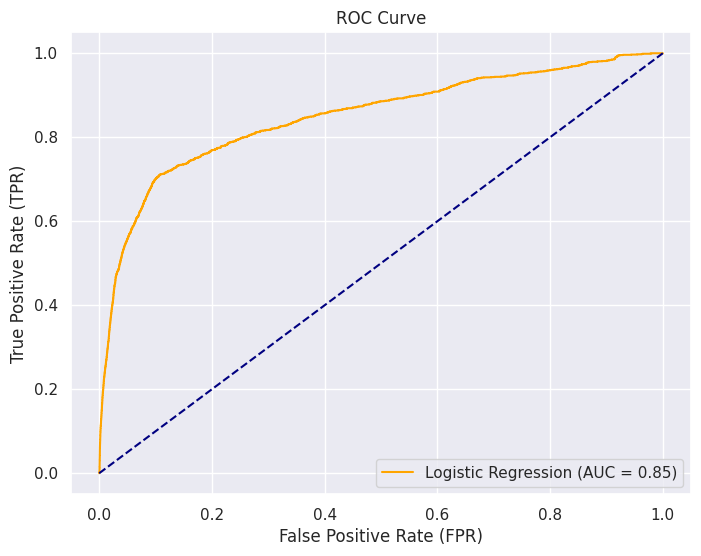
3. Improvement Attempts:

In an effort to address the identified challenges, Synthetic Minority Over-sampling Technique (SMOTE) was applied to mitigate class imbalance. Cross-validation was employed to ensure robustness and assess the model's generalization performance. Despite these improvement attempts, the precision and recall for label 1 did not show substantial improvement.

4. Random Forest Model:

Subsequently, a Random Forest model was introduced, and its performance was notably superior to the logistic regression model. The accuracy, precision, and recall metrics exhibited significant improvements. This suggested that the ensemble nature of the Random Forest, with its ability to capture complex relationships and handle imbalances, was better suited for the given dataset.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Precision for class 0** | **Recall for class 0** | **Precision for class 1** | **Recall for class 1** |
| Logistic Regression | 97% | 1 | 0.97 | 0.04 | 0.90 |
| Random Forest Classifier | 99.96% | 1 | 1 | 0.94 | 0.71 |

**Discussion**

The comparison between the logistic regression and Random Forest models underscored the importance of selecting an appropriate algorithm for the specific characteristics of the data. While logistic regression excelled in accuracy, the Random Forest model demonstrated superior precision and recall, particularly for the challenging instances of label 1. The decision to favor Random Forest over logistic regression was motivated by the necessity to prioritize both correctness and the ability to capture positive instances effectively.

**Limitations and Future Work**

It is important to acknowledge the limitations of this study, including potential biases, assumptions, and constraints. Future work could involve further exploration of feature engineering techniques, the application of advanced ensemble methods, or a deeper investigation into the nature of the data to enhance model performance.

**References**

[1] Fraudulent transactions 101: What causes them and how businesses can act.

[2] Credit Card Fraud Detection using Machine Learning Algorithms - Vaishnavi Nath Dornadula, S Geetha.

[3] A supervised machine learning algorithm for detecting and predicting fraud in credit card transactions - Jonathan Kwaku Afriyie, Kassim Tawiah.

[4] Fraud Detection in Big Data Using Supervised and Semi-Supervised Learning Techniques, 2017 IEEE Colombian Conference on Communications and Computing.

[5] Science Direct: Logistic Regression