TITLE - CREDIT CARD FRAUD DETECTION

AVIK KUNDU(1828008), CHANDAN MISHRA(1828010), ABHINAV SRIVASTAVA(1828042),

PRINEET KUMAR KUNDU(1828084), SHILADITYA ROY(1828102), SULAGNA CHATTERJEE(1828118)

ABSTRACT:

It is vital that credit card companies are able to identify fraudulent credit card transactions so that customers are not charged for items that they did not purchase. Such problems can be tackled with Data Science and its importance, along with Machine Learning, cannot be overstated. This project intends to illustrate the modelling of a data set using machine learning with Credit Card Fraud Detection.

Card Fraud Detection Problem includes modelling past credit card transactions with the data of the ones that turned out to be fraud. This model is then used to recognize whether a new transaction is fraudulent or not. Our objective here is to detect 100% of the fraudulent transactions while minimizing the incorrect fraud classifications. Credit Card Fraud Detection is a typical sample of classification. In this process, we have focused on

analysing and pre - processing data sets as well as the deployment of multiple anomaly detection algorithms such as

Local Outlier Factor and Isolation Forest algorithm on the PCA transformed Credit Card Transaction data.

1. INTRODUCTION

'Fraud' in credit card transactions is unauthorized and unwanted usage of an account by someone other than the owner of that account. Necessary prevention measures can be taken to stop this abuse and the behaviour of such fraudulent practices can be studied to minimize it and protect against similar occurrences in the future.In other words, Credit Card Fraud can be defined as a case where a person uses someone else's credit card for personal reasons while the owner and the

card issuing authorities are unaware of the fact that the card is being used.

Fraud detection involves monitoring the activities of populations of users in order to estimate, perceive or avoid objectionable behaviour, which consist of fraud, intrusion, and defaulting.

This is a very relevant problem that demands the attention of communities such as machine learning and data science where the solution to this problem can be automated.

This problem is particularly challenging from the perspective of learning, as it is characterized by various factors such as class imbalance. The number of valid transactions far outnumber fraudulent ones. Also, the transaction patterns often change their statistical properties over the course of time.

Fraud detection methods are continuously developed to defend criminals in adapting to their fraudulent strategies. These frauds are classified as:

- Credit Card Frauds: Online and Offline
- Card Theft
- Account Bankruptcy
- Device Intrusion
- Application Fraud
- Counterfeit Card
- Telecommunication Fraud

Some of the currently used approaches to detection of such fraud are:

- Artificial Neural Network
- Fuzzy Logic
- Genetic Algorithm
- Logistic Regression
- Decision tree
- Support Vector Machines
- Bayesian Networks
- Hidden Markov Model
- K-Nearest Neighbour

2. Credit card fraud

Illegal use of credit card or its information without the knowledge of the owner is referred to as credit card fraud.Different credit card fraud tricks belong mainly to two groups of application and behavioral fraud. Application fraud takes place when fraudsters apply for new cards from banks or issuing companies using false or other‟s information. Multiple applications may be submitted by one user with one set of user details (called duplication fraud) or different users with identical details (called identity fraud).

Based on statistical data stated in **2012**, the high risk countries facing credit card fraud threat is illustrated in Fig.1. Ukraine has the most fraud rate with staggering 19%, which is closely followed by Indonesia at 18.3% fraud rate. After these two, Yugoslavia with a rate of **17.8%** is the most risky country. The next highest fraud rate belongs to Malaysia **(5.9%)**, Turkey **(9%)** and finally the United States. Other countries that are prune to credit card fraud with the rate below than 1% are not demonstrated in figure 1.

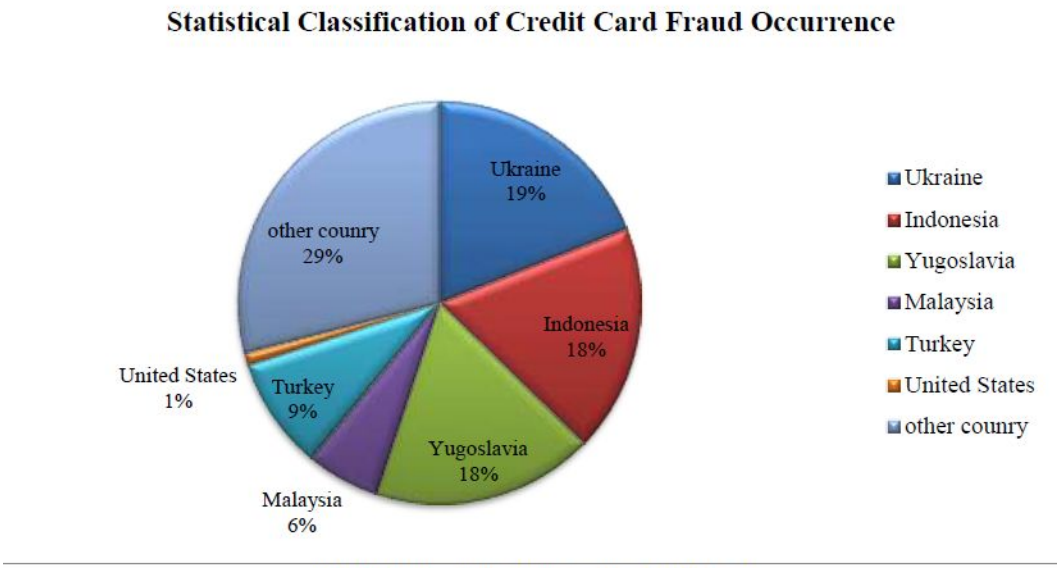


Fig1. High risk countries facing credit card fraud threat

3. ALGORITHMS AND PROCEDURES

- Firstly, we use a clustering method to divide the cardholders into different clusters/groups based on their transaction amount, i.e., high, medium and low using range partitioning.

- Using the Sliding-Window method, we aggregate the transactions into respective groups, i.e., extract some features from the window to find cardholder's behavioural patterns. Features like maximum amount, minimum amount of transaction, followed by the average amount in the window and even the time elapsed.

  Algorithm 1: Algorithm to derive aggregated transaction details and to extract card holder features using sliding

  window technique.

  Input: id of the customer holding a card, a sequence of transactions t and window size w;

  Output: Aggregated transactions details and features of cardholder genuine or fraud;

```
l: length of T
Genuine= [];
Fraud= [];
For i in range 0 to l-w+1:
T: [];
/* sliding window features*/
For j in range i+w-1:
/*Add the transaction to window */
T=T+tj
id;
End
/* features extraction related to amount */
ai1=MAX_AMT(Ti);
```

```
ai2=MIN_AMT(Ti);
ai3=AVG_AMT(Ti);
ai4=AMT(Ti);
For j in range i+w-1:
/* Time elapse */
xi= Time(tj)-Time(tj-1)
End
Xi= (ai1, ai2,ai3,ai4,ai5,);
Y= LABEL(Ti);
/* classifying a transaction into fraud or not */
if Yi=0 then
Genuine =Genuine U Xi;
Else
Fraud =Fraud U Xi;
End
```

- Every time a new transaction is fed to the window the old ones are removed and step-2 is processed for each group of transactions. (Algorithms for Sliding-Window based methods to aggregate are referred from [1]).
- After pre-processing, we train different classifiers on each group using the cardholders behavioural patterns in that group and extract fraud features. Even when we apply classifiers on the dataset, due to imbalance.
- Thus, we perform SMOTE (Synthetic Minority Over-Sampling Technique) operation on the dataset.
- Oversampling does not provide any good results.

- Thus, there are two different ways of dealing with imbalance dataset i.e., consider Matthew Coefficient Correlation of the classifier on the original dataset or we make use of one-class classifiers.
- Finally, the classifier that is used for training the group is applied to each cardholder in that group. The classifier with the highest rating score is considered as the cardholder's recent behavioural pattern.
- Once the rating score [1] is obtained, now we append a feedback system, wherein the current transaction and updated rating score are given back to the system (for further comparison) to solve the problem of concept drift.

  Algorithm 2: Algorithm to update the rating score of the classifier to find the accurate model is.

  Input: id of the cardholder and a previous and a current transaction.

  Output: Rating score of the model after every transaction.

```
1. T: current transaction with w-1 transaction from window.
2. C: represents the classifier
3. Label: true value of the incoming/current transaction.
4. K: total of transactions processed by model.
5. If the predicted value ≠ label and label==0 then,
6. For i in range (0, K):
7. If the predicted value ≠ label then,
8. rsi= rsi-1;
```

```
9.  Else
10. rsi =rsi+1;
11. End
```

4. BENCHMARK DATA SET

Data used for this example is from Kaggle — Credit Card Fraud Detection **(https://www.kaggle.com/mlg-ulb/creditcardfraud)**.
The datasets contain transactions made by credit cards in September 2013 by European cardholders. This dataset presents transactions that occurred in two days, where we have **492** frauds out of **284,807** transactions.
The dataset is highly unbalanced, the positive class (frauds) account for **0.172%** of all transactions.
It contains only numeric input variables which are the result of a PCA transformation. Unfortunately, due to confidentiality issues, original features and more background information about the data was not provided.
Features V1, V2, … V28 are the principal components obtained with PCA, the only features which have not been transformed with PCA are 'Time' and 'Amount'. Feature **'Time'** contains the seconds elapsed between each transaction and the first transaction in the dataset.
The feature **'Amount'** is the transaction Amount, this feature can be used for example-dependant cost-sensitive learning. Feature

‘Class’ is the response variable and it takes value 1 in case of fraud and 0 otherwise.

5. RESULTS ANALYSIS

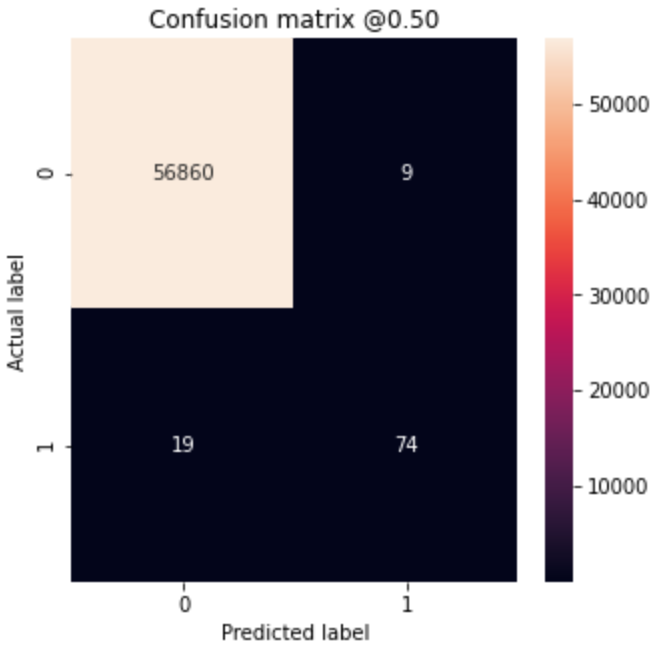
The code prints out the number of false positives it detected and compares it with the actual values. This is used to calculate the accuracy score and precision of the algorithms.

The fraction of data we used for faster testing is 10% of the entire dataset. The complete dataset is also used at the end and both the results are printed.

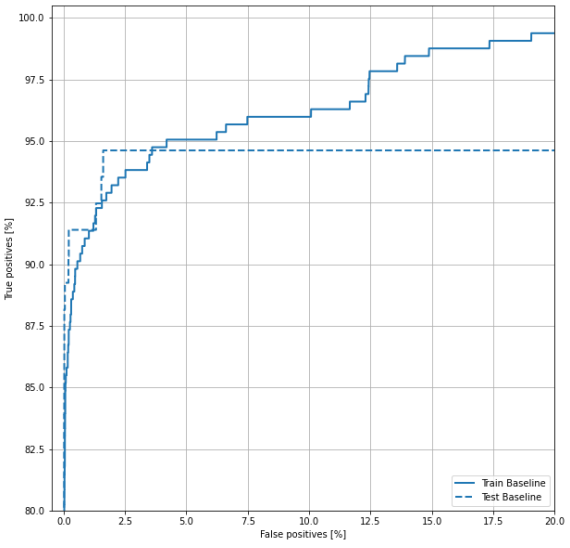
These results along with the classification report for each algorithm is given in the output as follows, where class 0 means the transaction was determined to be valid and 1 means it was determined as a fraud transaction.

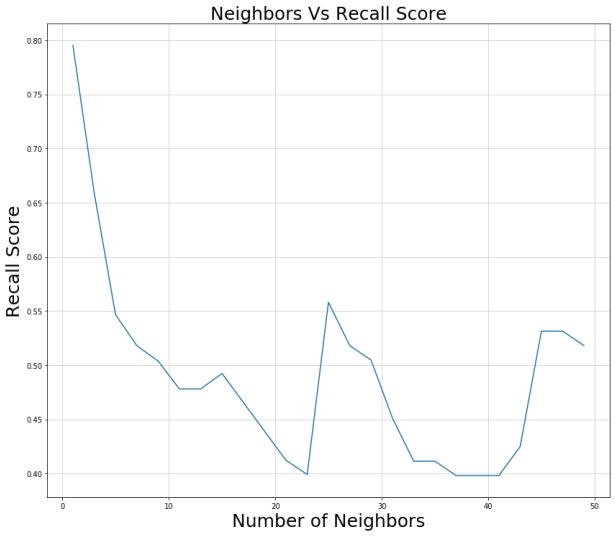
This result matched against the class values to check for false Positives.

5.1. Performance Matrix



5.2. Observations





ROC Curve **Neighbours vs Recall Score**

6. CONCLUSION:

Credit card fraud is without a doubt an act of criminal dishonesty. This article has listed out the most common methods of fraud along with their detection methods and reviewed recent findings in this field. This paper has also explained in detail, how machine learning can be applied to get better results in fraud detection along with the algorithm, pseudocode, explanation of its implementation and experimentation results.

While the algorithm does reach over 99.6% accuracy, its precision remains only at 28% when a tenth of the data set is taken into consideration. However, when the entire dataset is fed into the algorithm, the precision rises to 33%. This high percentage of accuracy is to be expected due to the huge imbalance between the number of valid and number of genuine transactions.

Since the entire dataset consists of only two days' transaction records, it's only a fraction of data that can be made available if this project were to be used on a commercial scale. Being based on machine learning algorithms, the program will only increase its efficiency over time as more data is put into it.

7. FUTURE WORK:

While we couldn't reach our goal of 100% accuracy in fraud detection, we did end up creating a system that can, with

enough time and data, get very close to that goal. As with any such project, there is some room for improvement here.

The very nature of this project allows for multiple algorithms to be integrated together as modules and their results can be combined to increase the accuracy of the final result. This model can further be improved with the addition of more algorithms into it. However, the output of these algorithms needs to be in the same format as the others. Once that condition is satisfied, the modules are easy to add as done in the code. This provides a great degree of modularity and versatility to the project.

More room for improvement can be found in the dataset. As demonstrated before, the precision of the algorithms increases when the size of the dataset is increased. Hence, more data will surely make the model more accurate in detecting frauds and reduce the number of false positives. However, this requires official support from the banks themselves.