Credit Card Fraud Detection using Machine

Learning and Data Science

Name:***Mangeshwari R***

Reg.No:***912721104019***

**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. The Credit

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.

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.

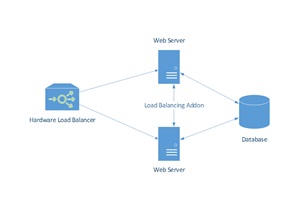
METHODOLOGY

The approach that this paper proposes, uses the latest machine

learning algorithms to detect anomalous activities, called outliers

The basic rough architecture diagram can be represented with

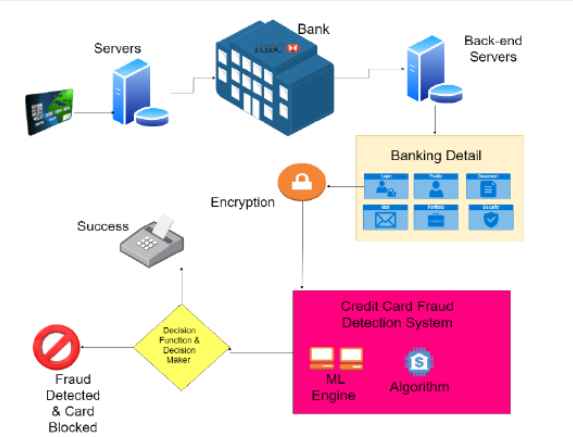
the following figure:



When looked at in detail on a larger scale along with real life

elements, the full architecture diagram can be represented as

follows:



First of all, we obtained our dataset from Kaggle, a data

analysis website which provides datasets.

Inside this dataset, there are 31 columns out of which 28 are

named as v1-v28 to protect sensitive data.

The other columns represent Time, Amount and Class. Time

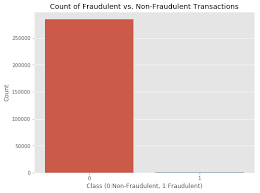
shows the time gap between the first transaction and the

following one. Amount is the amount of money transacted.

Class 0 represents a valid transaction and 1 represents a

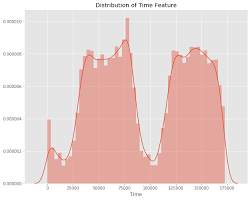
fraudulent one. We plot different graphs to check for inconsistencies in the

dataset and to visually comprehend it



This graph shows that the number of fraudulent transactions is

much lower than the legitimate ones.

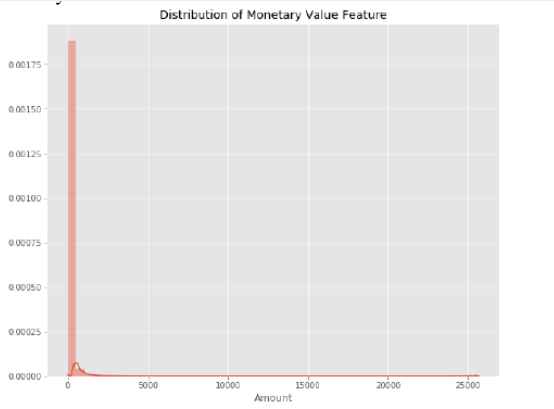


This graph shows the times at which transactions were done

within two days. It can be seen that the least number of

transactions were made during night time and highest during

the days.



This graph represents the amount that was transacted. A

majority of transactions are relatively small and only a handful

of them come close to the maximum transacted amount.

After checking this dataset, we plot a histogram for every

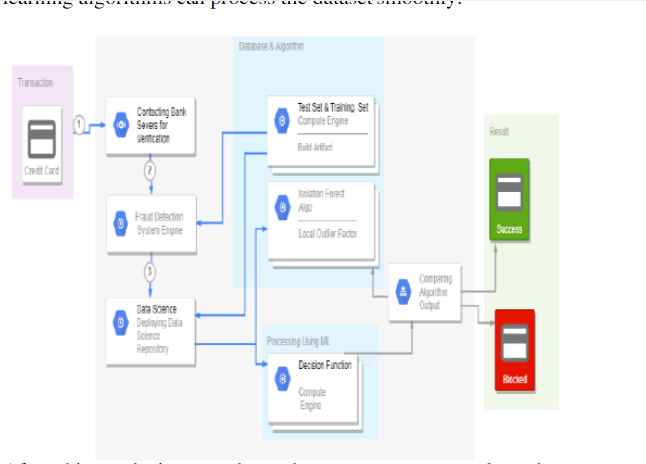
column. This is done to get a graphical representation of the

dataset which can be used to verify that there are no missing

any values in the dataset. This is done to ensure that we don’t

require any missing value imputation and the machine

learning algorithms can process the dataset smoothly.

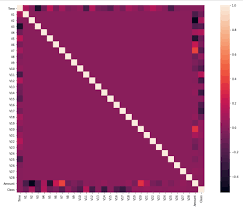


After this analysis, we plot a heatmap to get a coloured

representation of the data and to study the correlation between

out predicting variables and the class variable. This heatmap is

shown below:



The dataset is now formatted and processed. The time and

amount column are standardized and the Class column is

removed to ensure fairness of evaluation. The data is

processed by a set of algorithms from modules.

IMPLEMENTATION

This idea is difficult to implement in real life because it

requires the cooperation from banks, which aren’t willing to

share information due to their market competition, and also

due to legal reasons and protection of data of their users.

Therefore, we looked up some reference papers which

followed similar approaches and gathered results. As stated in

one of these reference papers:

“This technique was applied to a full application data set

supplied by a German bank in 2006. For banking

confidentiality reasons, only a summary of the results obtained

is presented below. After applying this technique, the level 1

list encompasses a few cases but with a high probability of

being fraudsters.

All individuals mentioned in this list had their cards closed to

avoid any risk due to their high-risk profile. The condition is

more complex for the other list. The level 2 list is still

restricted adequately to be checked on a case by case basis.

Credit and collection officers considered that half of the cases

in this list could be considered as suspicious fraudulent

behaviour. For the last list and the largest, the work is

equitably heavy. Less than a third of them are suspicious.

In order to maximize the time efficiency and the overhead

charges, a possibility is to include a new element in the query;

this element can be the five first digits of the phone numbers,

the email address, and the password, for instance, those new

queries can be applied to the level 2 list and level 3 list.”.

RESULTS

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

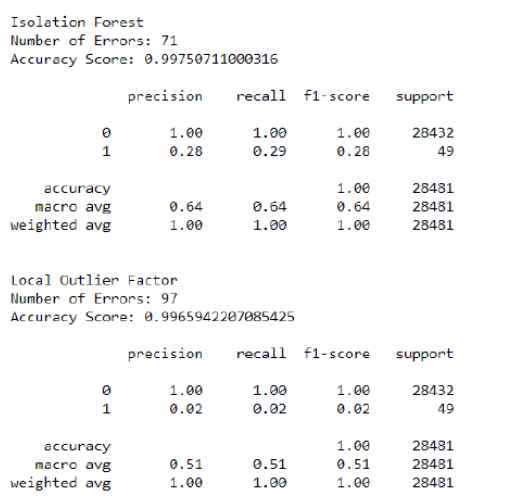
algorithm is given in the output as follows, where class 0

means the transaction was determined to be valid and 1 means

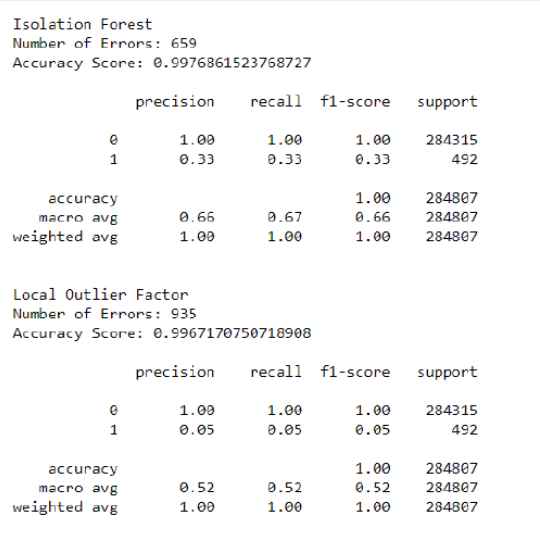
it was determined as a fraud transaction.

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

Results when 10% of the dataset is used:



Results with the complete dataset is used:



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 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

Since the entire dataset consists of only two days’ transaction

records, its 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.

FUTURE ENHANCEMENTS

While we couldn’t reach out 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 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.

“This technique was applied to a ful

queries can be applied to the level 2 list and level 3 lis