PROJECT: FRAUD DETECTION IN FINANCIAL TRANSACTIONS

Akshay Shinde

Context:

The ability to detect fraudulent credit card transactions is crucial for credit card companies to protect their customers and prevent unauthorized charges. This problem is a common challenge in machine learning due to the highly imbalanced nature of the data, where the majority of transactions are legitimate, and a very small minority is fraudulent.

Objective

Develop a machine learning model capable of accurately identifying fraudulent transactions, minimizing false positives (legitimate transactions classified as fraudulent) and false negatives (fraudulent transactions not detected).

Dataset Description

The dataset contains credit card transactions made in September 2013 by European cardholders. It presents transactions that occurred over two days, containing 284,807 transactions, of which 492 are fraudulent.

Dataset Features:

- **Time**: Number of seconds elapsed between this transaction and the first transaction in the dataset.
- **V1 - V28**: Principal components obtained through PCA (Principal Component Analysis) to protect user identities and sensitive information.
- **Amount**: Transaction amount.
- **Class**: Response variable, where 1 indicates a fraudulent transaction and 0 indicates a legitimate transaction.

Challenges:

1. **Data Imbalance**: The fraud class represents only 0.172% of all transactions.
2. **Interpretation of Principal Components**: Features V1 to V28 are results of a PCA transformation, making direct interpretation difficult.
3. **Performance Evaluation**: Due to the imbalance, confusion matrix accuracy is not adequate. It is recommended to use the Area Under Curve (AUC) to measure model performance.

Proposed Methodology:

1. Preprocessing:
    - Handle missing values (if any).
    - Removing outliers
2. Exploratory Data Analysis (EDA):

- Understand the distribution of 'Amount' variables.
- Check the correlation among the principal components.
- Visualize the distribution of fraudulent and legitimate transactions.

3. Modeling:
    - Test different machine learning algorithms, such as:
        - Logistic Regression
        - Decision Tree

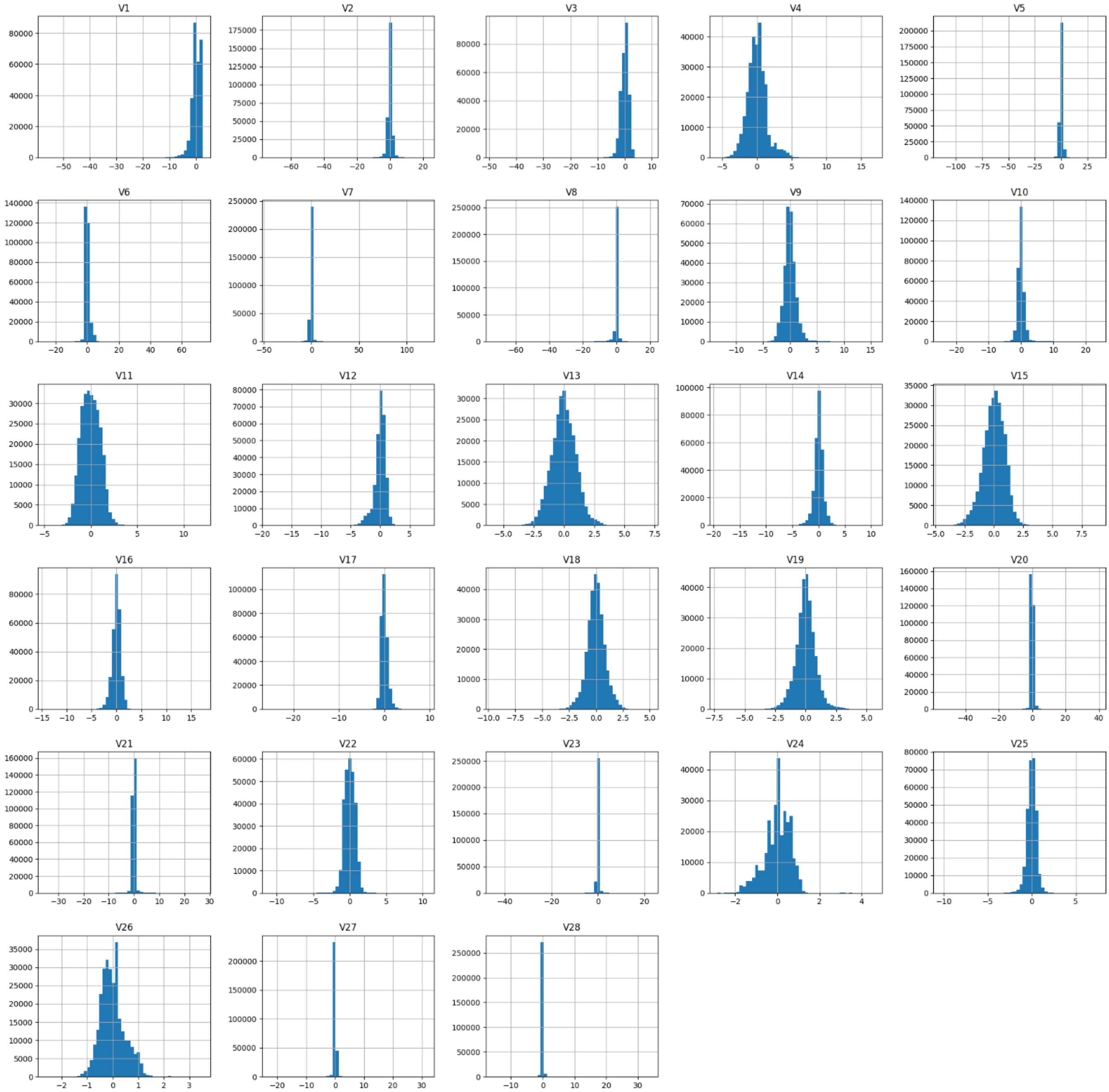
Evaluation:

- Use appropriate parameters like precision, recall, ROC-AUC Curve and F1-score to evaluate performance.
- Analyse the confusion matrix to identify error patterns.

Importing and data preprocessing:

I chose Kaggle dataset for the task, after choosing the most suited dataset the preparation phase begins, the preparation of the dataset includes selecting the wanted attributes or variables, cleaning it by excluding Null rows, deleting duplicated variables, treating outlier if necessary, in addition to transforming data types to the wanted type, data merging can be performed as well where two or more attributes get merged. All those alterations lead to the wanted result which is to make the data ready to be modeled.

The dataset chosen for this project didn't need to go through all of the alterations mentioned earlier, as there were no missing nor there was merging needed as well, there was some duplicates present in the data which we removed. After removing the duplicates with the help of quartiles we remove the outliers for better prediction. The data set include 473 Fraud Cases and 283253 Valid Transactions. The following graph shows the distribution of each variable V1-V28

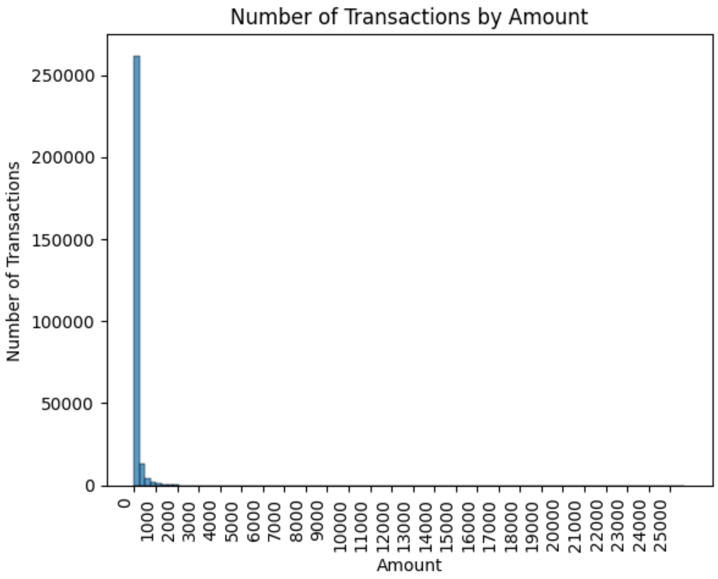


Exploratory Data Analysis (EDA):

The basic statistics for the data is obtained which include mean, standard deviation, quartiles, minimum and maximum for each variable for fraud and valid transactions. Following inferences can be drawn with the help of the statistics:

1. Mean amount for a fraud transaction is 123.872 and maximum is 2125.870
2. Mean amount for a legitimate transaction is 88.414 and maximum is 25691.160

To understand the distribution of amount variable plot the histogram.

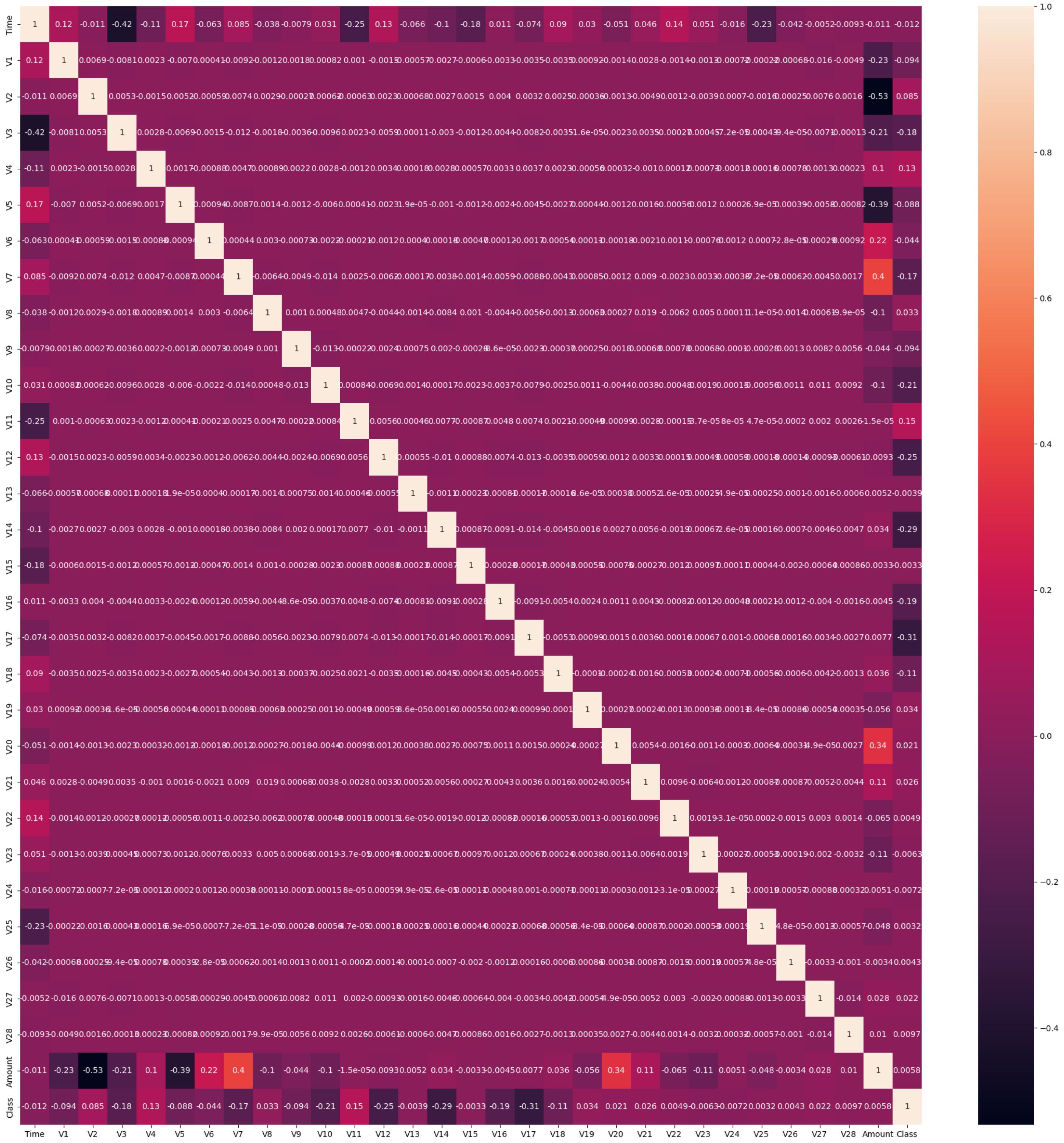


The graph shows that the variable amount is right skewed, following information is drawn from the variable amount.

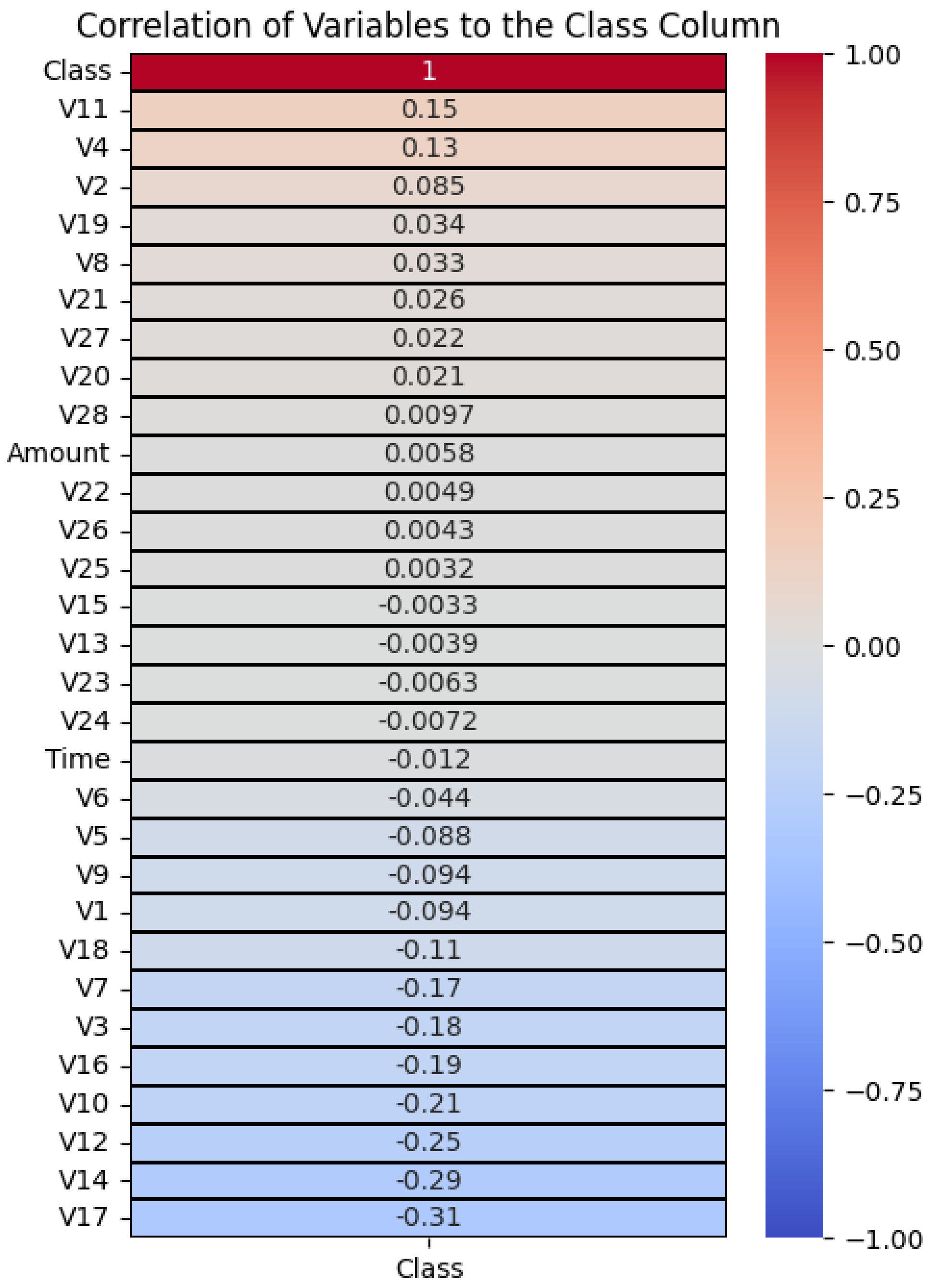
Class	0	1
count	283253.000	473.000
mean	88.414	123.872
std	250.379	260.211
min	0.000	0.000
25%	5.670	1.000
50%	22.000	9.820
75%	77.460	105.890
max	25691.160	2125.870

Checking the correlation among the principle components using heat map.

Heatmap:- A heatmap is a graphical representation of data where values are depicted by colour. It helps visualize complex information in a way that’s easy to comprehend. They’re particularly useful for identifying patterns and trends within data, making them valuable. From the heat map it is evident that the correlation among the principle components is very low.


Time
V1
V2
V3
V4
V5
V6
V7
V8
V9
V10
V11
V12
V13
V14
V15
V16
V17
V18
V19
V20
V21
V22
V23
V24
V25
V26
V27
V28
Amount
Class
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4

To check the correlation between principle components and target variable plot correlation heatmap for target variable class.



It is evident that the correlation between the principle components and target variable class is very low.

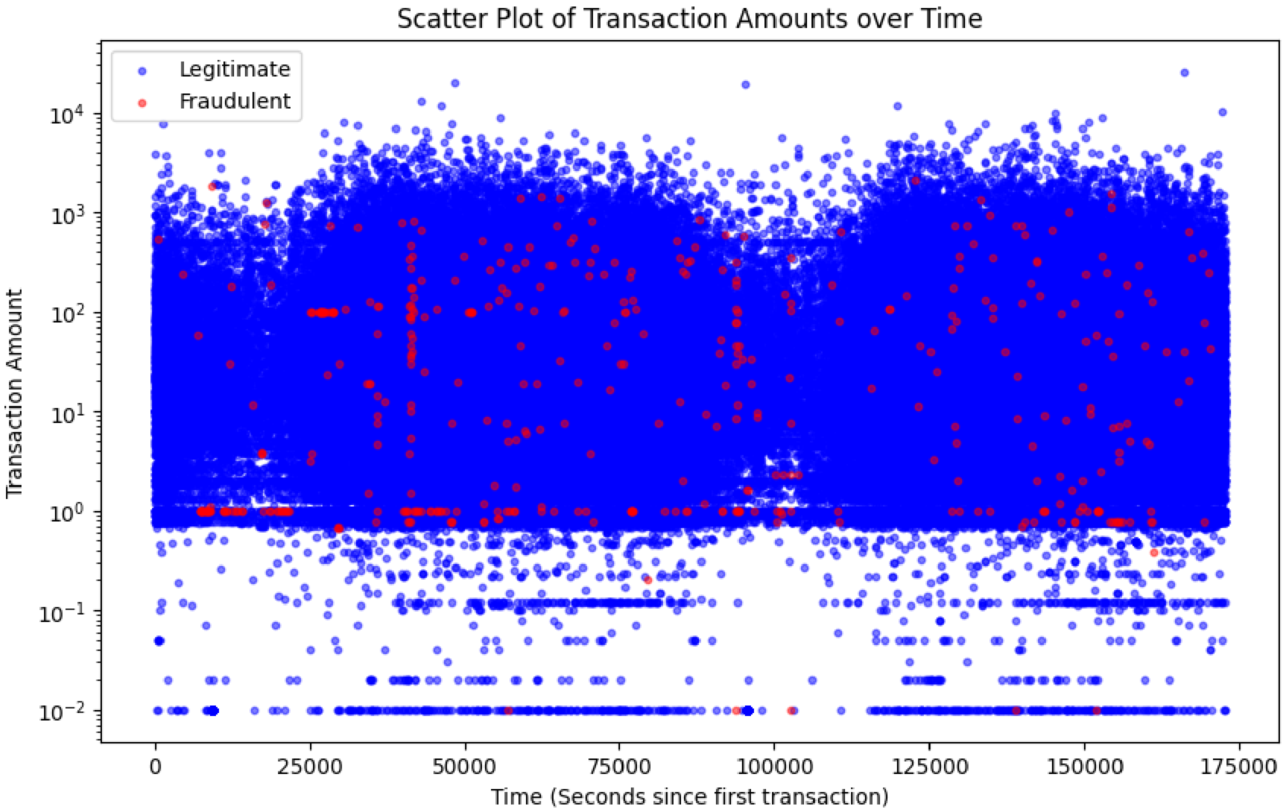
Visual Representation:-

Using scatter plot visualize the distribution of fraudulent and legitimate transactions.

Scatter Plot:

A scatter plot, also known as a scattergram or scatter chart, is a type of graph that visually displays the relationship between two continuous variables.

Here transaction amount and time is used as two continuous variables.



Modeling:-

Two machine learning models were created in the modeling phase, Logistic Regression and Decision Tree. A comparison of the results will be presented later in the paper to know which technique is most suited in the credit card fraudulent transactions detection. The dataset is sectioned into a ratio of 80:20, the training set will be the 80% and remaining set will be the testing set which is the 20%. The two models were created using libraries in python sklearn

Logistic Regression Model:

Logistic Regression model is statical model where evaluations are formed of the connection among dependent qualitative variable (binary or binomial logistic regression) or variable with three values or higher (multinomial logistic regression) and one independent explanatory variable or higher whether qualitative or quantitative.

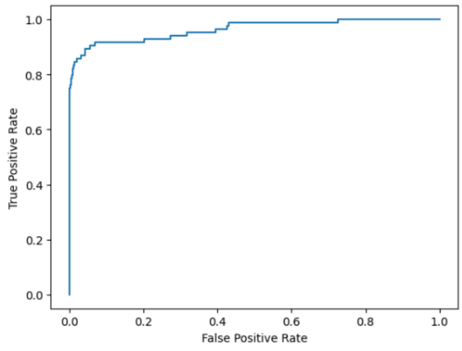
The model created using Python is Logistic Regression, the model managed to score 99.89% and accuracy of 99.90% in Python with 52 misclassified instances.

```
Classifcation report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     50325
           1       0.82      0.49      0.61        84

    accuracy                           1.00     50409
   macro avg       0.91      0.74      0.81     50409
weighted avg       1.00      1.00      1.00     50409

Confusion matrix :
 [[50316     9]
 [   43    41]]

Accuracy: 99.90%
```

Conclusion:-

1. Model has 100% precision for valid transactions and 82% precision for fraud transactions. This implies that model has high accuracy in correctly identifying positive outcomes.
2. Model has 100% recall for valid transactions and 49% recall for fraud transactions. Model has high effectiveness in identifying all positive instances for valid transactions.
3. f1-score for valid transactions is 1 and 0.61 for fraud transactions this implies model has excellent balance between precision and recall.

ROC-AUC Curve:



ROC AUC score is a measure of how well a classifier distinguishes positive and negative classes, and can take values from 0 to 1.
AUC for the model is 0.9630, this implies model has performed well.

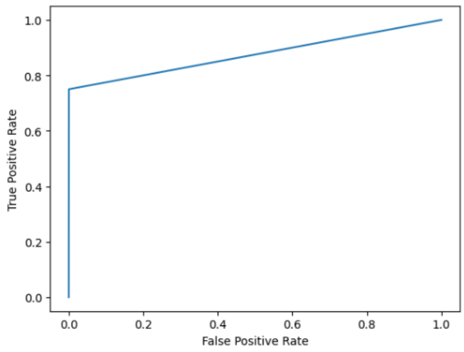
<u>Decision Tree</u>: -

A decision tree is a hierarchical model used in decision support that depicts decisions and their potential outcomes. It is a non-parametric, supervised learning algorithm that is useful for both classification and regression tasks. The tree structure in the decision model helps in drawing a conclusion for any problem which is more complex in nature. The decision rules are generally in the form of if-then-else statements. Decision trees are transparent, efficient, and flexible.
The model created using Python is Decision Tree, the model managed to score 99.92% and accuracy of 99.92% in Python with 40 misclassified instances.

```
Classifcation report:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00     50325
           1       0.77      0.75      0.76        84

    accuracy                           1.00     50409
   macro avg       0.88      0.87      0.88     50409
weighted avg       1.00      1.00      1.00     50409

Confusion matrix :
 [[50306    19]
 [   21    63]]

Accuracy: 99.92%
```

<u>Conclusion:-</u>
1. Model has 100% precision for valid transactions and 77% precision for fraud transactions. This implies that model has high accuracy in correctly identifying positive outcomes.
2. Model has 100% recall for valid transactions and 75% recall for fraud transactions. Model has high effectiveness in identifying all positive instances for valid transactions.
3. f1-score for valid transactions is 1 and 0.76 for fraud transactions this implies model has excellent balance between precision and recall.

<u>ROC-AUC Curve:</u>



ROC AUC score is a measure of how well a classifier distinguishes positive and negative classes, and can take values from 0 to 1.
AUC for the model is 0.8748, this implies model has performed well.

Conclusion: -

In conclusion, the main objective of this project was to find the most suited model in credit card fraud detection in terms of the machine learning techniques chosen for the project, and it was met by building the two models and finding the accuracies of them all, the best model in terms of accuracies is Decision Tree which scored 99.92% with only 40 misclassified instances. I believe that using the model will help in decreasing the amount of credit card fraud and increase the customers satisfaction as it will provide them with better experience in addition to feeling secure.