MINI PROJECT-II

(2020-21)

CREDIT CARD FRAUD DETECTION

MID-TERM REPORT

INSTITUTE OF ENGINEERING AND TECHNOLOGY

Supervised by:

Mr. VINAY AGRAWAL

(ASSISTANT PROFESSOR)

`

Submitted by:

NITIN KUMAR SINGH

(181500434)

HARSHVARDHAN SINGH

(181500264)

CONTENTS

ABSTRACT

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. Due to the rise and acceleration of E-Commerce, there has been a tremendous use of credit cards for online shopping which led to High amount of frauds related to credit cards. In the era of digitalization, the need to identify credit card frauds is necessary. Fraud detection involves monitoring and analyzing the behaviour of various users to estimate detect or avoid undesirable behaviour. To identify credit card fraud detection effectively, we need to understand the various technologies, algorithms and types involved in detecting credit card frauds. The algorithm can differentiate transactions which are fraudulent or not. Find fraud, they need to passed dataset and knowledge of the fraudulent transaction. They analyze the dataset and classify all transactions. Fraud detection involves monitoring the activities of populations of users to estimate, perceive or avoid objectionable behaviour, which consist of fraud, intrusion, and defaulting. Machine learning algorithms are employed to analyses all the authorized transactions and report the suspicious ones. These reports are investigated by professionals who contact the cardholders to confirm if the transaction was genuine or fraudulent. The investigators provide feedback to the automated system which is used to train and update the algorithm to eventually improve the fraud-detection performance over time

INTRODUCTION

1.1-GENERAL INTRODUCTION OF TOPIC

'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.

1.2- AREAS OF COMPUTER SCIENCE:

1- DATA VISUALIZATION

2- PROGRAMMING LANGUAGES

3- MACHINE LEARNING

1.3 HARDWARE REQUIREMENTS

PROCESSOR USED:- Intel Pentium or above

OPERATING SYSTEM:- Win 7 or above

RAM:- 4GB or above

HARDWARE DEVICES:- Computer or Laptop System

HARD DISK:- 256GB or above

SOFTWARE REQUIREMENTS

TECHNOLOGY USED : Numpy,Pandas,Scikit-learn library

LANGUAGE USED:Python

USER INTERFACE DESIGN-Jupyter Notebook

2-PROBLEM DEFINATION

To recognize fraudulent credit card transactions so that the customers of credit card companies are not charged for items that they did not purchase

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.

3. OBJECTIVE:

1. The model used must be simple and fast enough to detect the anomaly and classify it as a fraudulent

transaction as quickly as possible.

2. Imbalance can be dealt with by properly using some methods which we will talk about in the next paragraph

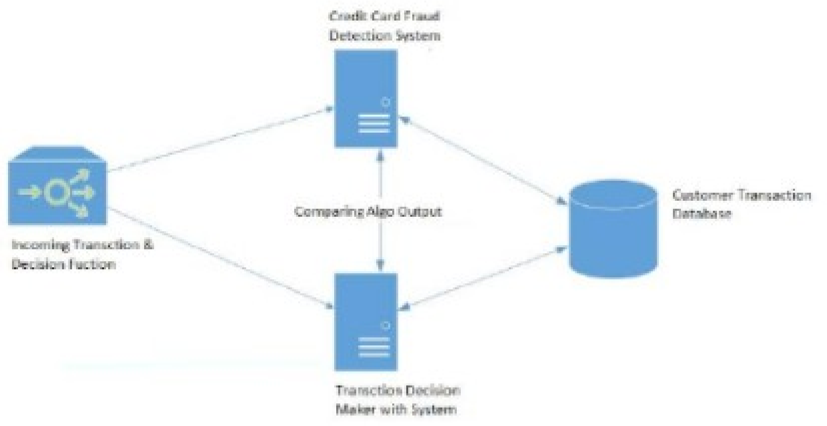
3. For protecting the privacy of the user the dimensionality of the data can be reduced.

4. A more trustworthy source must be taken which double-check the data, at least for training the model.

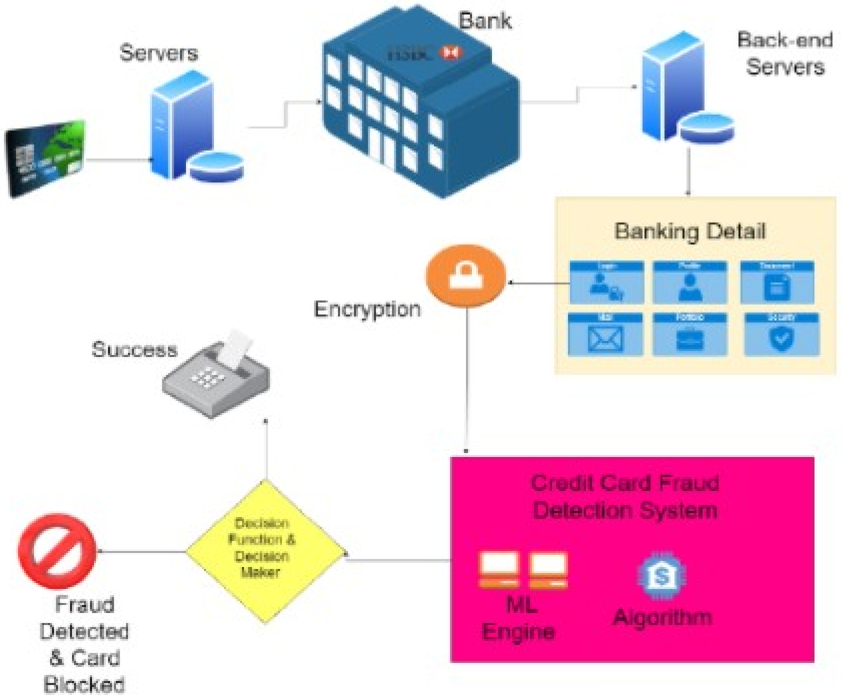
1. The model used must be simple and fast enough to detect the anomaly and classify it as a fraudulent transaction as quickly as possible.
2. Imbalance can be dealt with by properly using some methods which we will talk about in the next paragraph
3. For protecting the privacy of the user the dimensionality of the data can be reduced.
4. A more trustworthy source must be taken which double-check the data, at least for training the model.

III. 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:


Credit Card Fraud Detection System
Incoming Transaction & Decision Fuction
Comparing Algo Output
Customer Transaction Database
Transction Decision Maker with System

When looked at in detail on a larger scale along with real life elements, the full architecture diagram can be represented as follows:

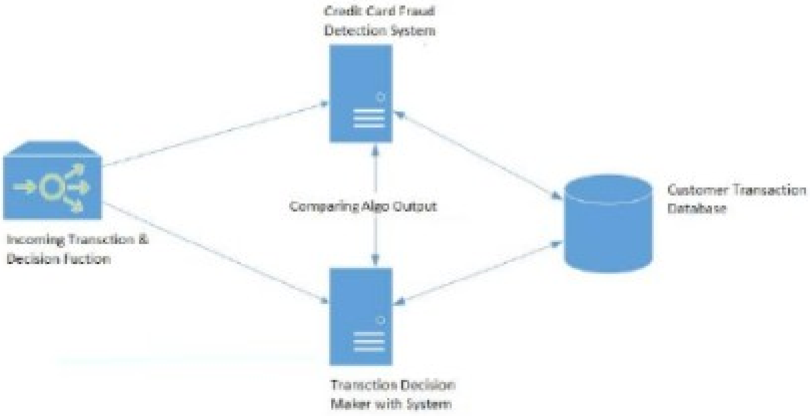

Servers
Bank
Back-end Servers
Banking Detail
Encryption
Success
Credit Card Fraud Detection System
ML Engine
Algorithm
Decision Function & Decision Maker
Fraud Detected & Card Blocked

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.

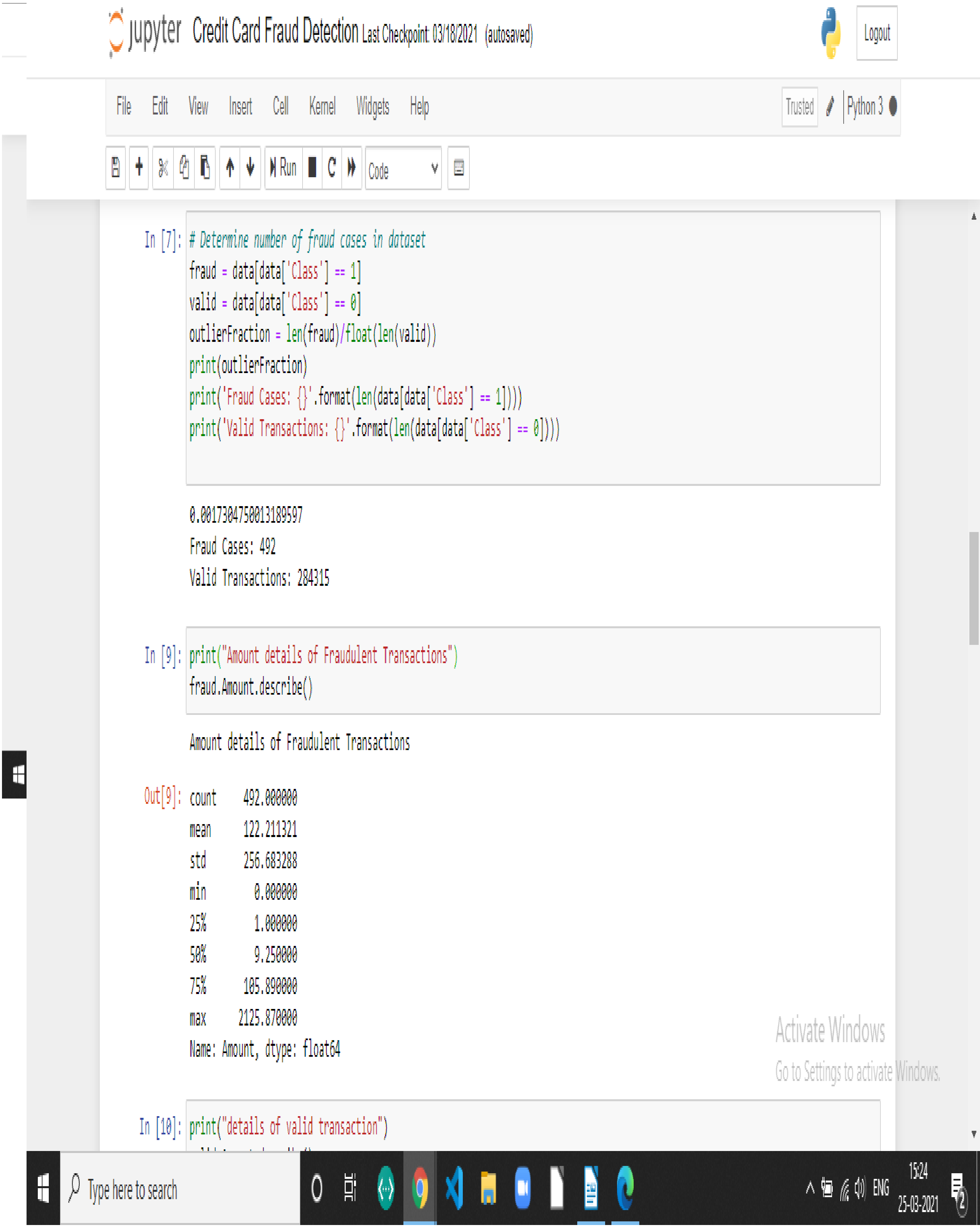
The other columns represent Time, Amount and Class. Time shows the time gap between the first transaction and the following one. Amount is the amount of money transacted. Class 0 represents a valid transaction and 1 represents a fraudulent one.

We plot different graphs to check for inconsistencies in the dataset and to visually comprehend it:


Credit Card Fraud Detection System
Incoming Transction & Decision Fuction
Comparing Algo Output
Customer Transaction Database
Transction Decision Maker with System

IMPLEMENTATION-

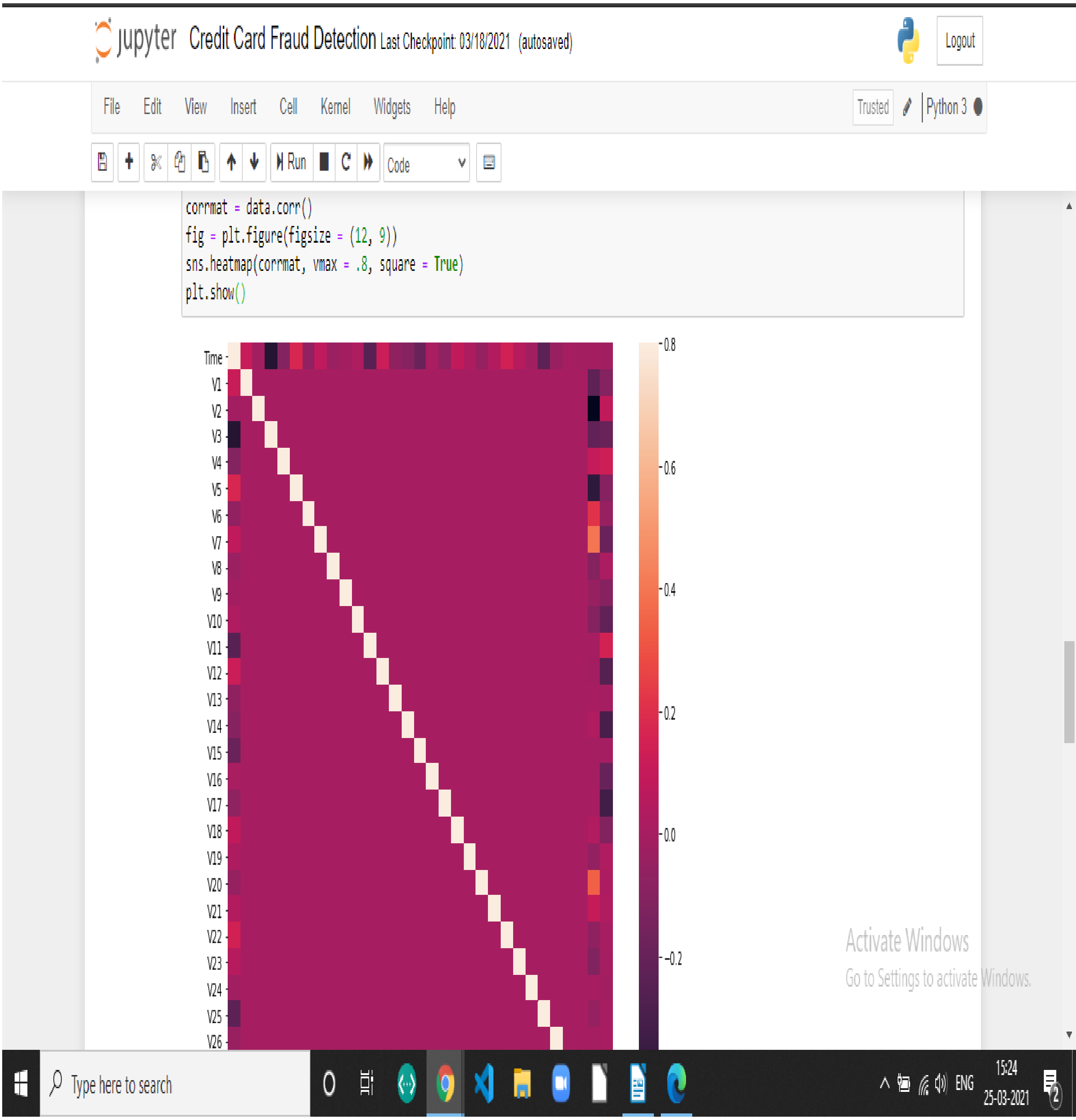
This idea is difficult to implement in real life because it requires the cooperation from banks, which aren't willing to share information due to their market competition, and also due to legal reasons and protection of data of their users. Therefore, we looked up some reference papers which followed 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. queries can be applied to the level 2 list and level 3 list.".



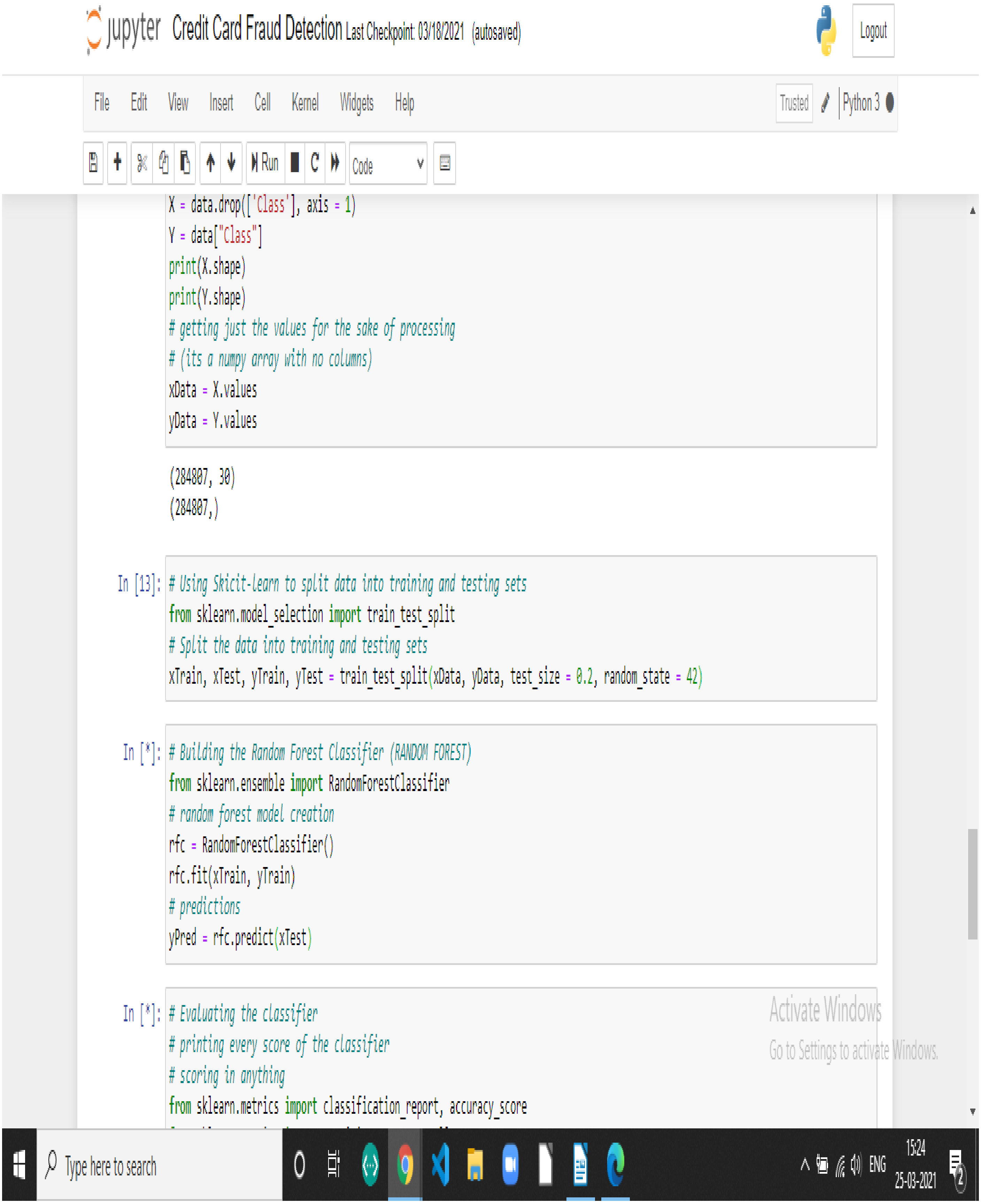
```
In [7]: # Determine number of fraud cases in dataset
        fraud = data[data['Class'] == 1]
        valid = data[data['Class'] == 0]
        outlierFraction = len(fraud)/float(len(valid))
        print(outlierFraction)
        print('Fraud Cases: {}'.format(len(data[data['Class'] == 1])))
        print('Valid Transactions: {}'.format(len(data[data['Class'] == 0])))

        0.0017304750013189597
        Fraud Cases: 492
        Valid Transactions: 284315
```

```
In [9]: print("Amount details of Fraudulent Transactions")
        fraud.Amount.describe()

        Amount details of Fraudulent Transactions

Out[9]: count     492.000000
        mean      122.211321
        std       256.683288
        min         0.000000
        25%         1.000000
        50%         9.250000
        75%       105.890000
        max      2125.870000
        Name: Amount, dtype: float64
```

```
In [10]: print("details of valid transaction")
```



```
corrmat = data.corr()
fig = plt.figure(figsize = (12, 9))
sns.heatmap(corrmat, vmax = .8, square = True)
plt.show()
```


jupyter Credit Card Fraud Detection Last Checkpoint: 03/18/2021 (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code
Activate Windows
Go to Settings to activate Windows.
Type here to search
ENG
15:24
25-03-2021

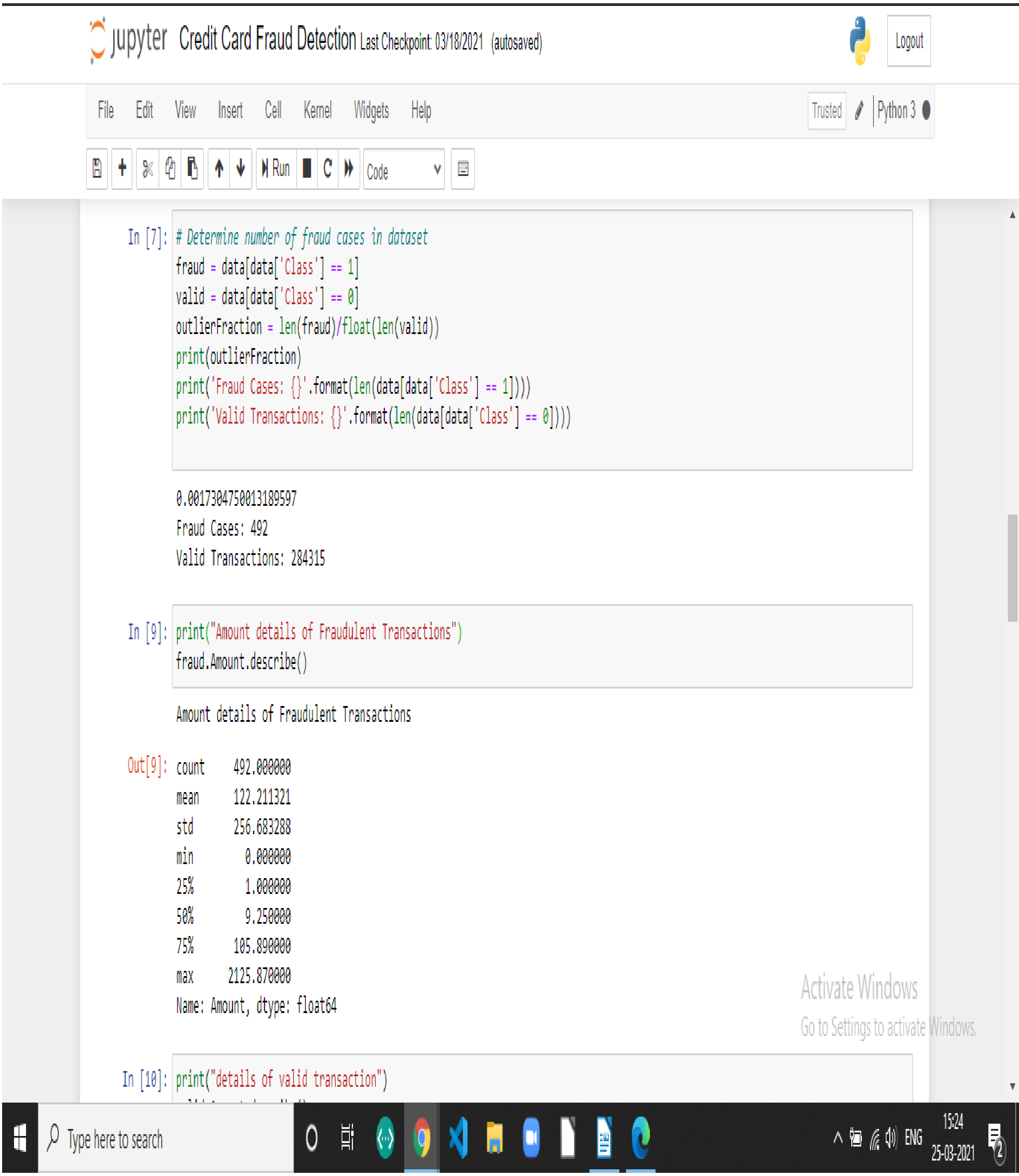
```
X = data.drop(['Class'], axis = 1)
Y = data["Class"]
print(X.shape)
print(Y.shape)
# getting just the values for the sake of processing
# (its a numpy array with no columns)
xData = X.values
yData = Y.values
```

```
(284807, 30)
(284807,)
```

In [13]:
```
# Using Skicit-learn to split data into training and testing sets
from sklearn.model_selection import train_test_split
# Split the data into training and testing sets
xTrain, xTest, yTrain, yTest = train_test_split(xData, yData, test_size = 0.2, random_state = 42)
```

In [*]:
```
# Building the Random Forest Classifier (RANDOM FOREST)
from sklearn.ensemble import RandomForestClassifier
# random forest model creation
rfc = RandomForestClassifier()
rfc.fit(xTrain, yTrain)
# predictions
yPred = rfc.predict(xTest)
```

In [*]:
```
# Evaluating the classifier
# printing every score of the classifier
# scoring in anything
from sklearn.metrics import classification_report, accuracy_score
```



In [7]:

```
# Determine number of fraud cases in dataset
fraud = data[data['Class'] == 1]
valid = data[data['Class'] == 0]
outlierFraction = len(fraud)/float(len(valid))
print(outlierFraction)
print('Fraud Cases: {}'.format(len(data[data['Class'] == 1])))
print('Valid Transactions: {}'.format(len(data[data['Class'] == 0])))
```

```
0.0017304750013189597
Fraud Cases: 492
Valid Transactions: 284315
```

In [9]:

```
print("Amount details of Fraudulent Transactions")
fraud.Amount.describe()
```

```
Amount details of Fraudulent Transactions
```

Out[9]:

```
count     492.000000
mean      122.211321
std       256.683288
min         0.000000
25%         1.000000
50%         9.250000
75%       105.890000
max      2125.870000
Name: Amount, dtype: float64
```

In [10]:

```
print("details of valid transaction")
```

REFERENCES:

www.youtube.com

www.geeeksforgeeks.com

www.analyticvidhya.com

www.tutorialspoint.com

www.learnpython.org

www.simplileaen.com

books-

Hands–On Machine Learning with Scikit–Learn and TensorFlow 2e: Concepts, Tools, and Techniques to Build Intelligent Systems

Machine Learning for Absolute Beginners

FACULTY GUILDELINE:

Mr.VINAY AGARWAL