Credit Card Fraud Detection Using Supervised Machine Learning

A PROJECT REPORT

Submitted by:

Jatin Kumar Saini (20BC4446)
Saksham Bhatia (20BCS4441)
Mriganka Das (20BCS4457)
Jatin Choudhary (20BCS4494)

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING IN

COMPUTER SCIENCE & ENGINEERING

Chandigarh University

NOVEMBER 2023

BONAFIDE CERTIFICATE

Certified that this project report "**Credit Card Fraud Detection Using Supervised Machine Learning**" is the bonafide work of **Jatin Kumar Saini, Saksham Bhatia , Mriganka Das, Jatin Choudhary"** who carried out the project work under my/our supervision.

SIGNATURE	**SIGNATURE**
Aman Khausik	Mansi Kajal
HEAD OF DEPARTMENT	**SUPREVISOR**

Submitted for the project viva-voce examination held____/__/2023

INTERNAL EXAMINER	**EXTERNAL EXAMINER**

DECLARATION

I, 'Saksham Bhatia, Jatin Choudhary, Jatin Saini, Mriganka Das', student of **'Bachelor of Engineering in CSE (Hons.) with Specialization in Big Data Analytics', session: 2020-2024,** Department of Computer Science and Engineering, Apex Institute of Technology, Chandigarh University, Punjab, hereby declare that the work presented in this Project Work entitled **'Credit Card Fraud Detection using supervised machine learning'** is the outcome of our own bona fide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. It contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Saksham Bhatia - 20BCS4441
Jatin Choudhary – 20BCS4494
Jatin Saini – 20BCS4446
Mriganka Das – 20BCS4457

Date: 01/10/2023
Place: Chandigarh University

ACKNOWLEDGEMENT

We are grateful to our respectable teacher, whose insightful leadership and knowledge benefited us to complete this project successfully. Thank you so much for your continuous support and presence whenever needed.
We would also like to thank for his advice and contribution to the project and the preparation of this report.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

This Project is focused on credit card fraud detection in real world scenarios. Nowadays credit card frauds are drastically increasing in number as compared to earlier times. Criminals are using fake identity and various technologies to trap the users and get the money out of them. Therefore, it is very essential to find a solution to these types of frauds. In this proposed project we designed a model to detect the fraud activity in credit card transactions. This system can provide most of the important features required to detect illegal and illicit transactions. As technology changes constantly, it is becoming difficult to track the behaviour and pattern of criminal transactions. To come up with the solution one can make use of technologies with the increase of machine learning, artificial intelligence and other relevant fields of information technology; it becomes feasible to automate this process and to save some of the intensive amounts of labour that is put into detecting credit card fraud. Initially, we will collect the credit card usage data-set by users and classify it as trained and testing dataset using a random forest algorithm and decision trees. Using this feasible algorithm, we can analyze the larger data-set and user provided current data-set. Then augment the accuracy of the result data. Proceeded with the application of processing of some of the attributes provided which can find affected fraud detection in viewing the graphical model of data visualization. The performance of the techniques is gauged based on accuracy, sensitivity, and specificity, precision. The results is indicated concerning the best accuracy for Random Forest are unit 98.6% respectively.

CHAPTER 1- INTRODUCTION:

1.1 Overview

Credit card is the most popular mode of payment. As the number of credit card users is rising world-wide, the identity theft is increased, and frauds are also increasing. In the virtual card purchase, only the card information is required such as card number, expiration date, secure code, etc. Such purchases are normally done on the Internet or over telephone. To commit fraud in these types of purchases, a person simply needs to know the card details. The mode of payment for online purchase is mostly done by credit card. The details of credit card should be kept private. To secure credit card privacy, the details should not be leaked. Different ways to steal credit card details are phishing websites, steal/lost credit cards, counterfeit credit cards, theft of card details, intercepted cards etc. For security purpose, the above things should be avoided. In online fraud, the transaction is made remotely and only the card's details are needed. The simple way to detect this type of fraud is to analyze the spending patterns on every card and to figure out any variation to the "usual" spending patterns. Fraud detection by analyzing the existing data purchase of cardholder is the best way to reduce the rate of successful credit card frauds. As the data sets are not available and also the results are not disclosed to the public. The fraud cases should be detected from the available data sets known as the logged data and user behavior. At present, fraud detection has been implemented by a number of methods such as data mining, statistics, and artificial intelligence.

1.2 Problem Statement

The card holder faced a lot of trouble before the investigation finish. And also, as all the transaction is maintained in a log, we need to maintain huge data, and also now a day's lotof online purchase are made so we don't know the person how is using the card online, we just capture the ip address for verification purpose. So there need a help from the cyber- crime to investigate the fraud.

1.3 Significance and Relevance of Work

Relevance of work includes consideration of all the possible ways to provide a solution to given problem. The proposed solution should satisfy all the user requirements and should be flexible enough so that future changes can easily done based on the future upcoming requirements like Machine learning techniques.

There are two important categories of machine learning techniques to identify the frauds in credit card transactions: supervised and unsupervised learning model. In supervised approach, early transactions of credit card are labelled as genuine or frauds. Then, the scheme identifies the fraud transaction with credit card data.

1.4 Objectives

Features Extractions from recognized facial information then data will be normalized for extracting features of good Objective of the project is to predict the fraud and fraud less transaction with respect to the time and amount of the transaction using classification machine learning algorithms such as SVM, Random Forest, Decision tree and confusion matrix in building of the complex machine learning models.

1.5 Methodology

First the Dataset is read. Exploratory Data Analysis is performed on the dataset to clearly understand the statistics of the data, Feature selection is used, A machine learning model is developed. Train and test the model and analysis the performance of the model using certain evaluation techniques such as accuracy, confusion matrix, precision etc.

1.6 Organization of the report

Chapter 1

1.6.1 **Overview:** the overview provides the basic layout and the insight about the work proposed. It briefs the entire need of the currently proposed work.

1.6.2 **Problem statement:** A problem statement is a concise description of an issue tobe addressed or a condition to be improved upon. We have identifiedthe gap between addressed or a condition to be improved upon.

1.6.3 **Significance and Relevance of Work:** We have mentioned about the contribution of our work to the society.

1.6.4 **Objectives:** A project objective describes the desired results of the work. We have mentioned about the work we are trying to accomplish in this section.

1.6.5 **Methodology:** A methodology is a collection of methods, practices, processes and techniques. We have explained in this section about theworking of the project in a brief way.

Chapter 2

1. **Literature Survey:** the purpose of a literature review is to gain an understanding of the existing resources to a particular topic or area of study. We have referred to many research papers relevant to our work in a better way.

Chapter 3

1. **System Requirements and Specifications:** System Requirements and Specifications is a document that describes the nature of a project, software or application. This section contains the brief knowledge about the functional and non – functional that are needed to implement the project.

Chapter 4

1. **System Analysis:** System Analysis is a document that describes about the existing system and proposed system in the project. And also describes about advantages and disadvantages in the project.

Chapter 5

1. **System design:** System design is a document that describes about the project modules, Activity diagram, Use Case Diagram, Data Flow Diagram, and Sequence Diagram detailed in the project.

Chapter 6

1. **Implementation:** Implementation is a document that describes about the detailed concepts of the projt. Also describes about the algorithm with the

Chapter 7

1. Testing: Testing is a document that describes about the
   a. **Methods of testing:** This contains the information about Unit testing, Validation testing, Functional testing, Integration testing, User Acceptance testing.
   b. **Test Cases:** In Test Cases we contain the detailed description about program Testcases.

Chapter 8

1. **Performance Analysis:** Performance Analysis is a document that describes about the study system in detailed.

Chapter 9

1. **Conclusion and Future Enhancement:** Conclusion and Future Enhancementis a document that describes about the brief summary of the project and undetermined events that will occur in that time.

Chapter 2: Literature survey

2.1 Credit Card Fraud Detection Techniques : Data and Technique OrientedPerspective

Authors: SamanehSorournejad, Zahra Zojaji, Amir Hassan Monadjemi.

In this paper, after investigating difficulties of credit card fraud detection, we seek to reviewthe state ofthe art in credit card fraud detection techniques, datasets and evaluation criteria.

Disadvantages

- Lack of standard metrics

2.2 Detection of credit card fraud: State of art

Authors: Imane Sadgali, Nawal Sael, Faouzia Benabbau

In this paper, we propose a state of the art on various techniques of credit card fraud detection. The purpose of this study is to give a review of implemented techniques for creditcard fraud detection, analyses their incomes and limitless, and synthesize the finding in order to identify the techniques and methods that give the best results so far.

Disadvantages

- Lack of adaptability

2.3 Credit card fraud detection using machine learning algorithm Authors: Vaishnavi Nath Dornadulaa, Geetha S.

The main aim of the paper is to design and develop a novel fraud detection method for Streaming Transaction Data, with an objective, to analyze the past transaction details of the customers and extract the behavioral patterns.

Disadvantages

- Imbalanced Data

2.4 Fraudulent Transaction Detection in Credit Card by Applying Ensemble Machine Learning techniques Authors: Debachudamani Prusti, Santanu Kumar Rath

In this study, the application of various classification models is proposed by implementing machinelearning techniques to find out the accuracy and other performance parameters to identify the fraudulent transaction.

Disadvantages

- **Overlapping data.**

2.5 Detection of Credit Card Fraud Transactions using Machine Learning Algorithms andNeural Networks

Authors: Deepti Dighe, Sneha Patil, Shrikant Kokate

Credit card fraud resulting from misuse of the system is defined as theft or misuse of one's credit card information which is used for personal gains without the permission of the card holder. To detect such frauds, it is important to check the usage patterns of a user over the past transactions. Comparing the usage pattern and current transaction, we can classify it as either fraud or a legitimate transaction.

Disadvantages

- Different misclassification importance

2.6 Credit card fraud detection using machine learning algorithms andcyber security Authors: JiatongShen

As they have the same accuracy the time factor is considered to choose the best algorithm. Byconsideringthe time factor they concluded that the Adaboost algorithm works well to detect credit card fraud.

Disadvantages

- Accuracy is not getting perfectly

CHAPTER-3

SYSTEM REQUIREMENTS AND SPECIFICATION

3.1 System Requirement Specification:

System Requirement Specification (SRS) is a fundamental document, which forms the foundation of the software development process. The System Requirements Specification (SRS) document describes all data, functional and behavioral requirements of the software under production or development. An SRS is basically an organization's understanding (in writing) of a customer or potential client's system requirements and dependencies at a particular point in time (usually) prior to any actual design or development work. It's a two- way insurance policy that assures that both the client and the organization understand the other's requirements from that perspective at a given point in time. The SRS also functionsas a blueprint for completing a project with as little cost growth as possible. The SRS is often referred to as the "parent" document because all subsequent project management documents, such as design specifications, statements of work, software architecture specifications, testing and validation plans, and documentation plans, are related to it. It is important to note that an SRS contains functional and non-functional requirements only. It doesn't offer design suggestions, possible solutions to technology or business issues, or any other information other than what the development team understands the customer's system requirements.

3.2 Hardware specification

- RAM: 4GB and Higher
- Processor: intel i3 and above
- Hard Disk: 500GB: Minimum

3.3 Software specification

- OS: Windows or Linux
- Python IDE: python 2.7.x and above
- Jupyter Notebook
- Language: Python

3.4 Functional Requirements:

Functional Requirement defines a function of a software system and how the system must behave when presented with specific inputs or conditions. These may include calculations, data manipulation and processing and other specific functionality. In this system following are the functional requirements:

- Collect the Datasets.
- Train the Model.
- Predict the results

3.5 Non-Functional Requirements

- The system should be easy to maintain.
- The system should be compatible with different platforms.
- The system should be fast as customers always need speed.
- The system should be accessible to online users.
- The system should be easy to learn by both sophisticated and novice users.
- The system should provide easy, navigable and user-friendly interfaces.
- The system should produce reports in different forms such as tables and graphs foreasy visualization by management.
- The system should have a standard graphical user interface that allows for the online

3.6 Performance Requirement:

Performance is measured in terms of the output provided by the application. Requirement specification plays an important part in the analysis of a system. Only when the requirement specifications are properly given, it is possible to design a system, which will fit into required environment. It rests largely with the users of the existing system to give the requirement specifications because they are the people who finally use the system. This is because the requirements have to be known during the initial stages so that the system can be designed according to those requirements. It is very difficult to change the system onceit has been designed and on the other hand designing a system, which does not cater to the requirements of the user, is of no use

CHAPTER-4 SYSTEM ANALYSIS

Systems analysis is the process by which an individual studies a system such that an information system can be analyzed, modeled, and a logical alternative can be chosen. Systems analysis projects are initiated for three reasons: problems, opportunities, and directives

4.1 Existing System

- Since the credit card fraud detection system is a highly researched field, there are many different algorithms and techniques for performing the credit card fraud detection system.
- One of the earliest systems is CCFD system using Markov model. Some other various existing algorithms used in the credit cards fraud detection system includes Cost sensitivedecision tree (CSDT).
- credit card fraud detection (CCFD) is also proposed by using neural networks. The existing credit card fraud detection system using neural network follows the whale swarmoptimization algorithm to obtain an incentive value. • It the uses BP network to rectify the values which are found error.

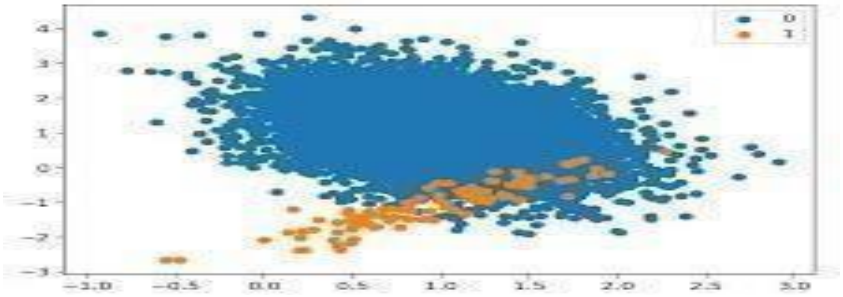


Figure 4.1.1 fraud and Non Fraud Representation

4.1.1 Limitations

- If the time interval is too short, then Markov models are inappropriate because the individual displacements are not random, but rather are deterministically related in time. This example suggests that Markov models are generally inappropriate over sufficiently short time intervals.

4.2 Proposed System

Support Vector Machine:

SVM works by mapping data to a high-dimensional feature space so that data points can be categorized, even when the data are not otherwise linearly separable. A separator between the categories is found, then the data are transformed in such a way that the separator could be drawn as a hyperplane Training regression model and finding out the best one.

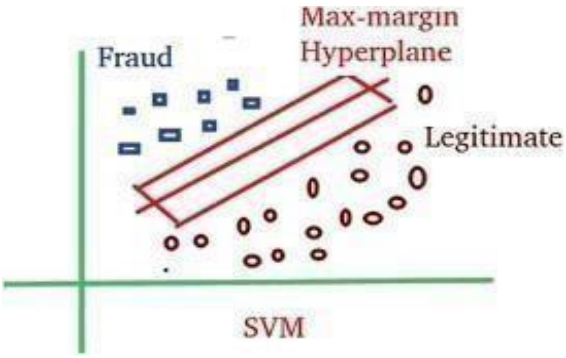


Fig 4.2.1 SVM Representation

Random Forest Classifier

Features are cheekbone to jaw width, width to upper facial height ratio, perimeter to area ratio, eye size, lower face to face height ratio, face width to lower face heightratio and mean of eyebrow height. The extracted features are normalized and finally subjected to support regression.

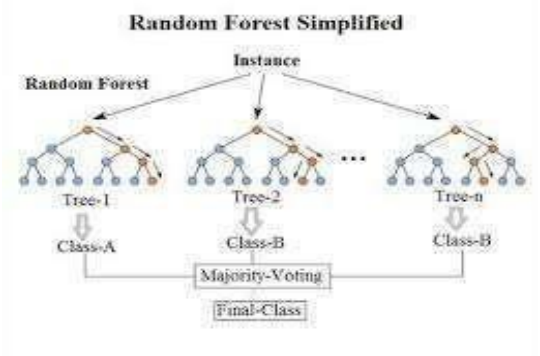
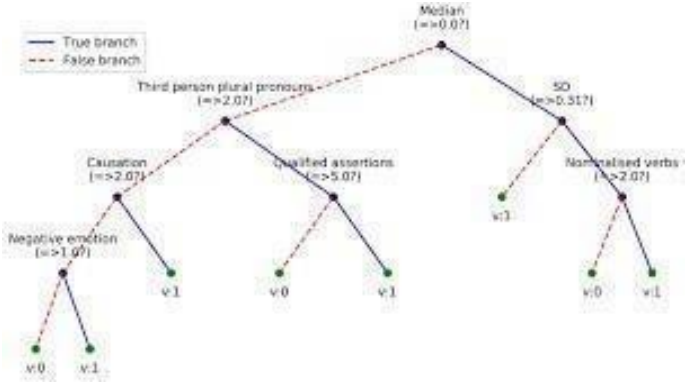


Fig 4.2.2 Simplified Random Forest algorithm

Decision Tree

A decision tree is a type of supervised machine learning used to categorize or make predictions based on how a previous set of questions were answered. The model is a formof supervised learning, meaning that the model is trained and tested on a set of data that contains the desired categorization.



4.2.3 Decision tree Algorithm

4.2.1 Advantages

- Support vector machine works comparably well when there is an understandablemargin of dissociation between classes.
- SVM is effective in instances where the number of dimensions is larger than thenumber of specimens.
- Simple to understand and to interpret.
- Requires little data preparation.
- The cost of using the tree (i.e., predicting data) is logarithmic in the number of datapoints used to train the tree.
- Able to handle both numerical and categorical data.
- Random forest classifier can be used to solve for regression or classification problems.
- The random forest algorithm is made up of a collection of decision trees, and each tree in the ensemble is comprised of a data sample drawn from a training set with replacement, called the bootstrap sample.

Chapter 5: Design flow/Process

Increase in online transactions using payment methods like credit card has also increased the fraudulent activities. Every year, a large amount of financial losses are caused by these illegal credit card transactions. No system is 100% secure and there is always a loophole in them. Therefore there is need to solve the issues of detecting fraud in transactions done by credit cards. To overcome this problem the proposed system for fraud identification in credit card transactions is designed using Random Forest algorithm. This algorithm uses combination of Decision Tree to solve the problem. Each tree is trained using dataset and based on this training each tree gives probability of transaction been fraud or legal. After that model predicts the result.

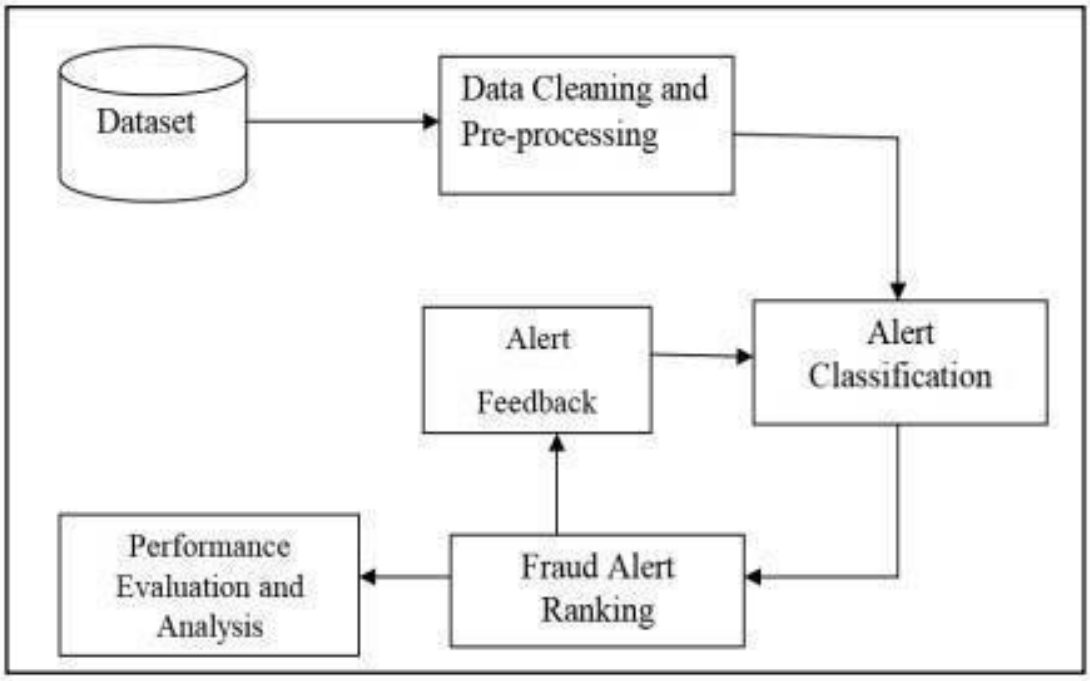


***Figure 1** Proposed System Block Diagram*

Modules

As shown in the block diagram, following are the modules of system: a.

Data Cleaning and Preprocessing b. Alerts Classification

c. Alert Ranking

d. Performance Analysis

Data Cleaning and Preprocessing

The model accuracy depends on amount of data on which it is trained. The more amounts of data better will be the performance of model. In this first module the selected data is cleaned and preprocessed as follow:

a. Cleaning: Fixing of missing data or removal of duplicate data from dataset is called as cleaning. The dataset may contain record which may be duplicate, incomplete or may have null values. Such records need to remove by cleaning.

b. Sampling: As number of frauds in dataset is less than overall transaction, class distribution is unbalanced in credit card transaction. Hence sampling method is used to solve this issue.

Alert Classification

Here machine learning model is used that trains the model based on features associated with transactions like location from where transaction is made, zip code, IP address, time and identity of customer. All this dataset is fed as input to the classifier and classifier splits them into multiple decision trees. The sub-trees check this input for an authorized transaction and give probabilities of transaction to be fraud or legal. Combining the results of all sub-trees, the model will alert the fraudulent transaction.

Alert Ranking

This module ranks each alert using learning to rank algorithm. The algorithm ranks each alert identified by the model using likelihood. If it is found that alert has greater rank then a security question is generated. If the individual answers the security question correctly then the transaction is allowed otherwise it is blocked. The IP address and location of fraudster is then tracked by the system. This security questions will be created every time whenever the transaction is identified to be suspicious and rank of alert is highest. This makes the FDS user friendly and helps to launch complaint against fraudster. Also the number of alert generated by the system is reduced as compared to rule based approach system.

Algorithm of Proposed Strategy

User inputs v1,v2,…vn

Dataset D

Step1: Initiate User Input

Step2: for each transaction x do deploy pattern P compute probability PD generate alerts A

Step3: Implement RankNet Model to generate rank return rank

Step 4: if rank>=n then Display Security Question SQ verify SQ

Step 5: if SQ verified then allow transaction else block transaction and track fraudster location [end if]

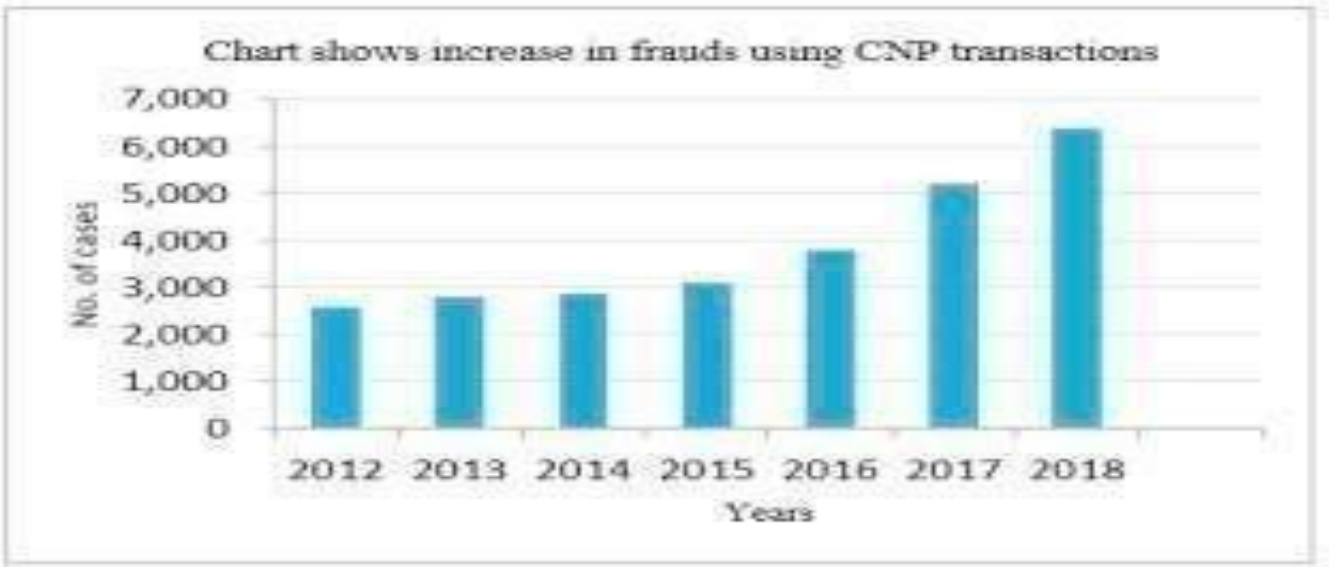


Fig. 2: Frauds Using Card Not Present Transaction

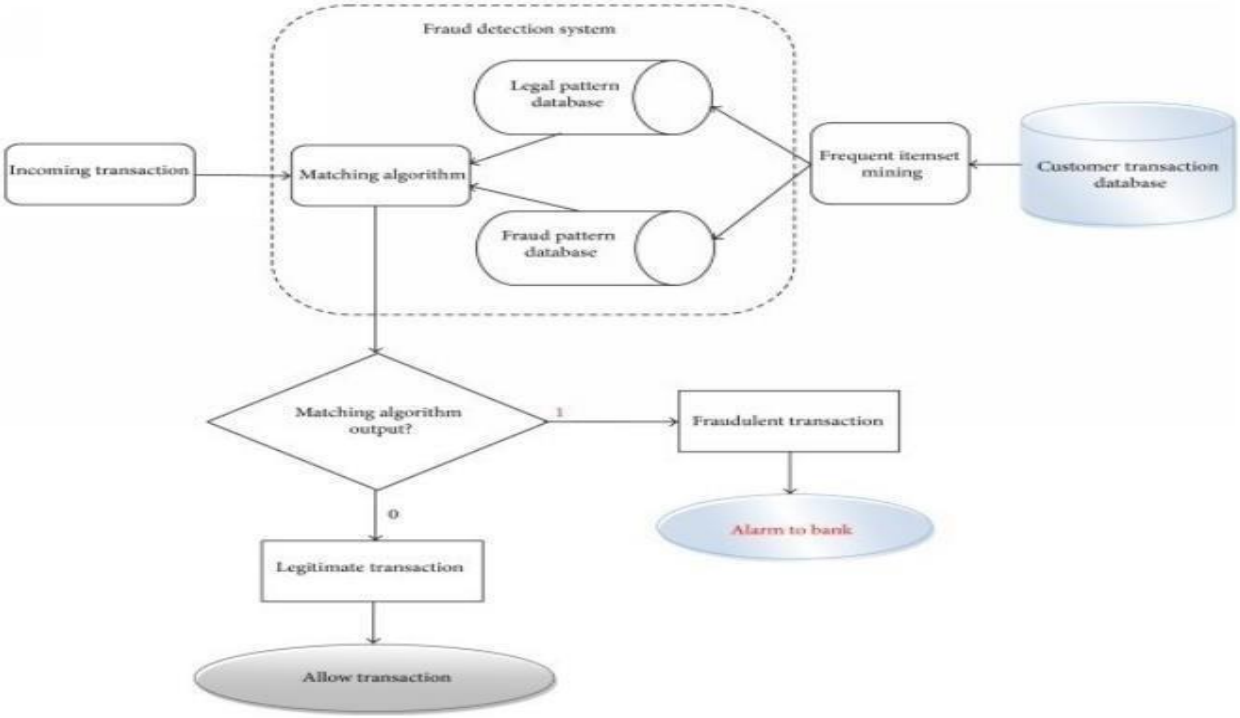


Fig. 3: flow chart

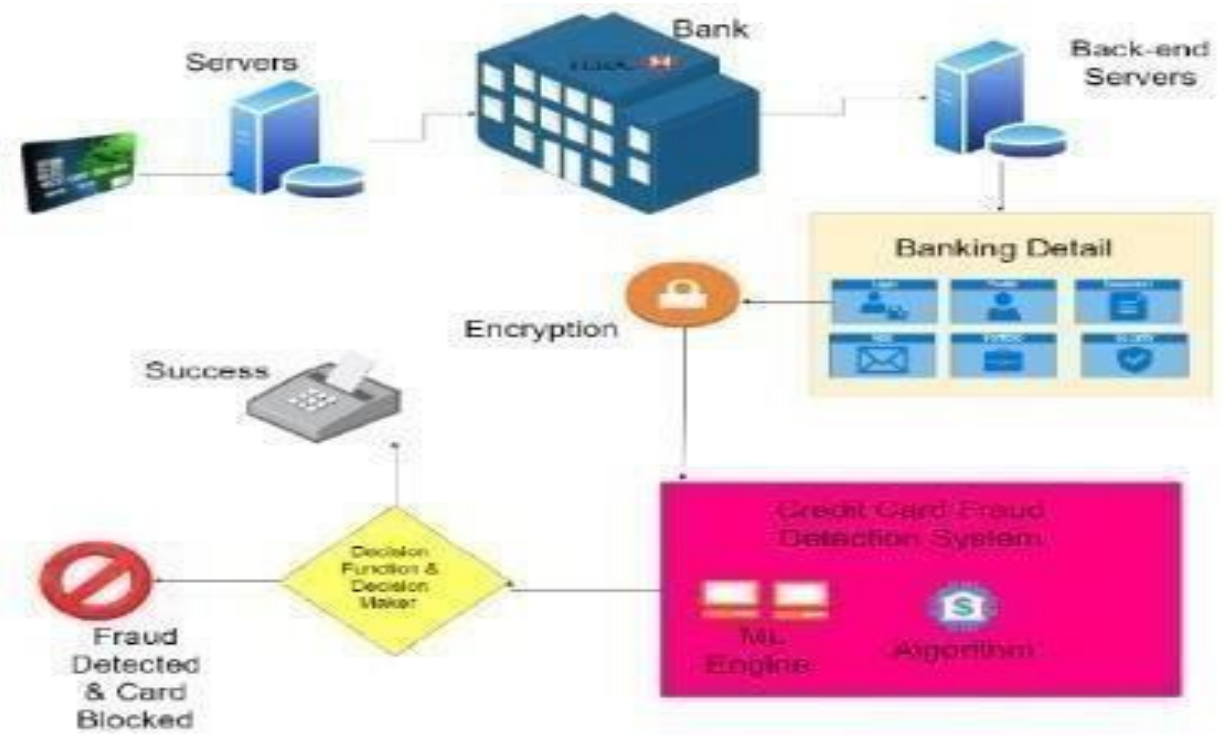


Fig. 4: Architecture diagram

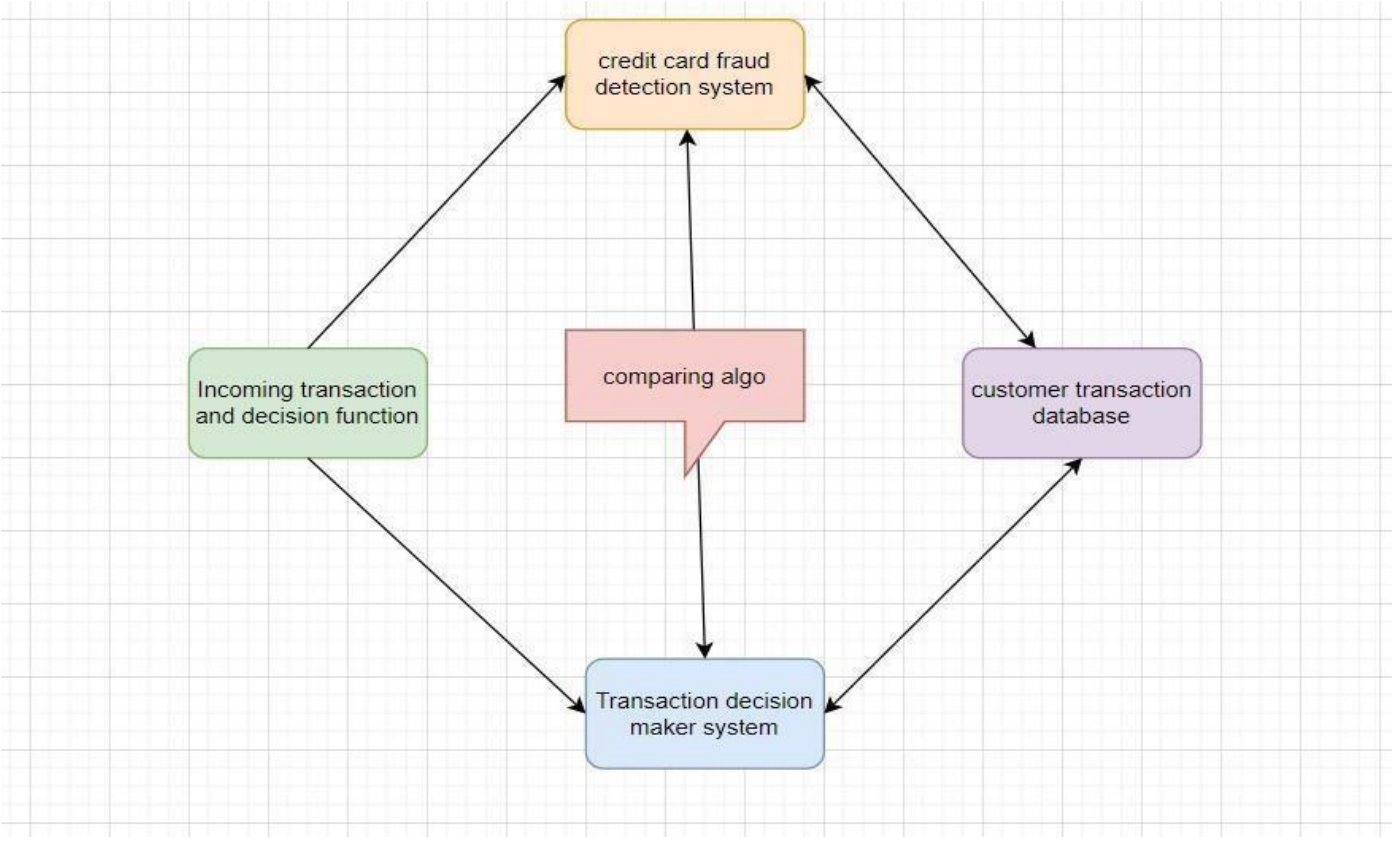


Fig. 5: Basic concept of research or architecture diagram

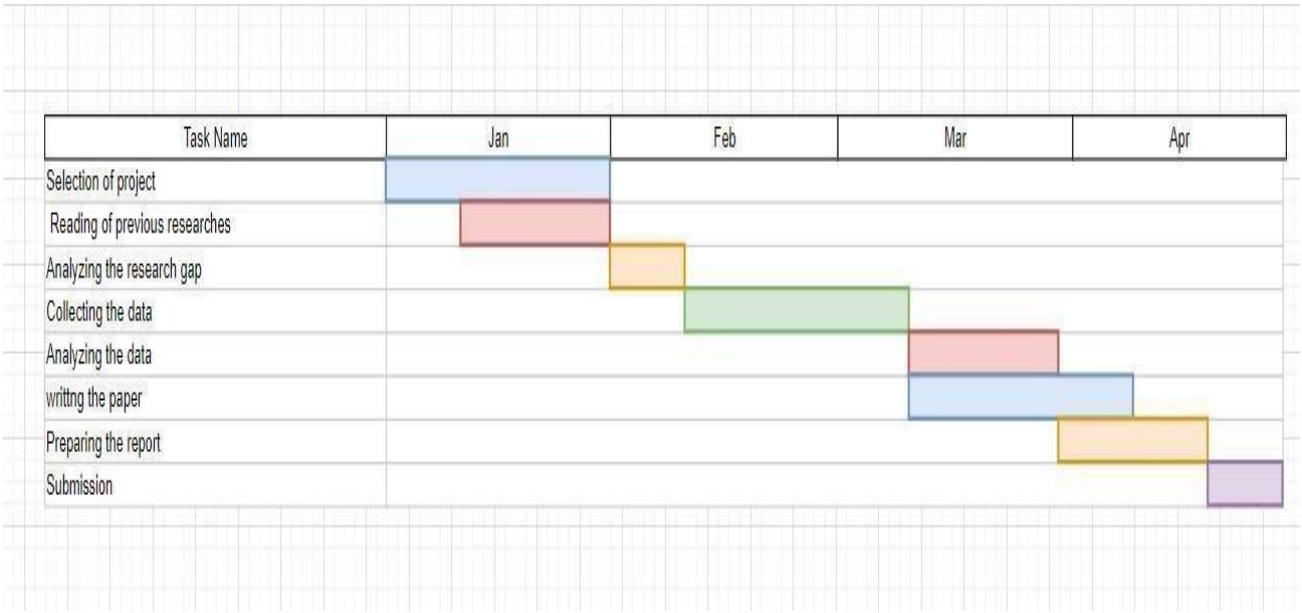


Fig. 6: Chart

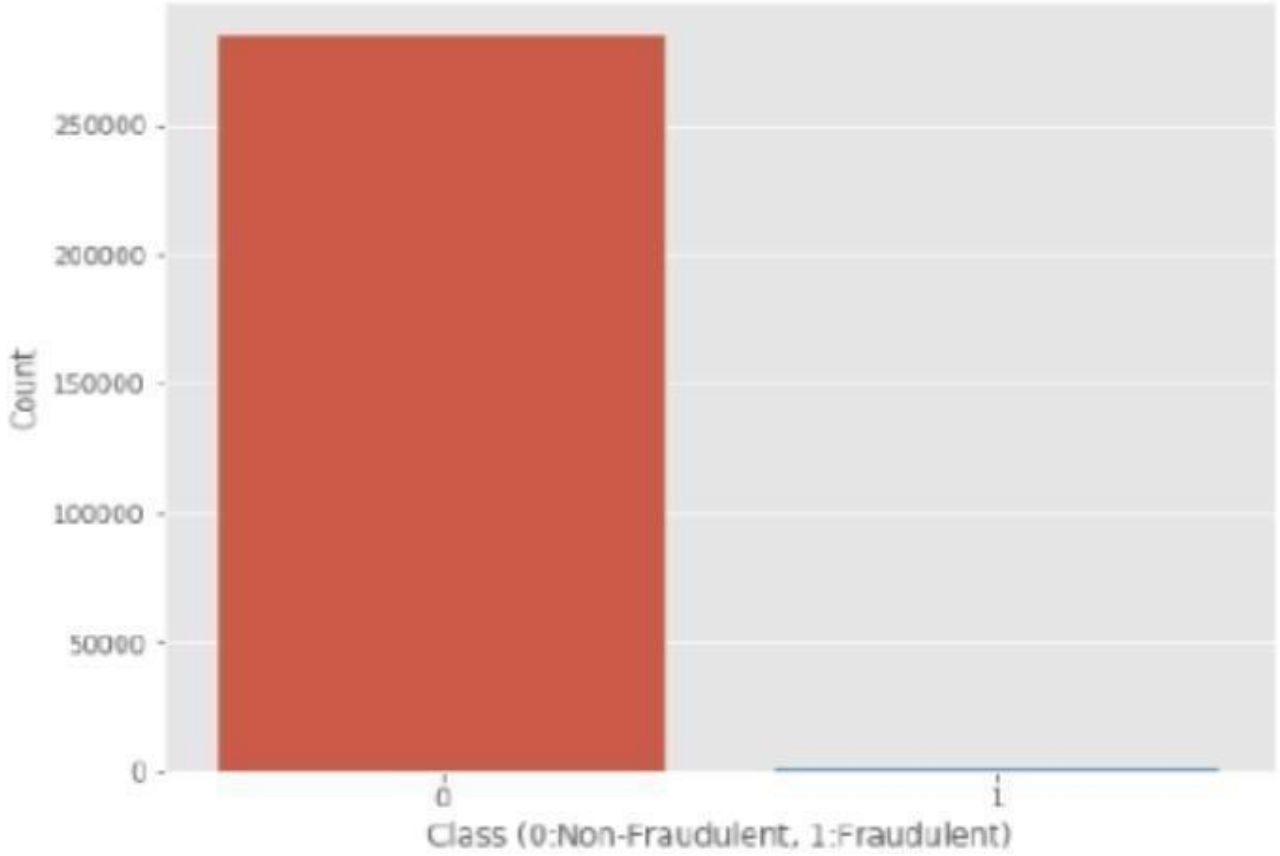


Fig. 7: Count of Fraudulent vs Non-Fraudulent Transactions

Chapter 6 Results analysis and validation

1. Jupyter Notebook

Computation notebooks have been used as electronic lab notebooks to document procedures, data, calculations, and findings. Jupyter notebooks provide an interactive computational environment for developing data science applications.

Jupyter notebooks combine software code, computational output, explanatory text, and rich content in a single document. Notebooks allow in-browser editing and execution of code and display computation results. A notebook is saved with an .ipynb extension. The Jupyter Notebook project supports dozens of programming languages, its name reflecting support for Julia (Ju), Python (Py), and R.

The following are some of the features of Jupyter notebooks that makes it one of the best components of Python ML ecosystem −

- Jupyter notebooks can illustrate the analysis process step by step by arranging the stuff like code, images, text, output etc. in a step by step manner.
- It helps a data scientist to document the thought process while developing the analysis process.
- One can also capture the result as the part of the notebook. • With the help of jupyter notebooks, we can share our work with a peer also.

2. DataSets

- **NumPy**

It is another useful component that makes Python as one of the favorite languages for Data Science. It basically stands for Numerical Python and consists of multidimensional array objects. By using NumPy, we can perform the following important operations −

- Mathematical and logical operations on arrays.
- Fourier transformation • Operations associated with linear algebra.

We can also see NumPy as the replacement of MatLab because NumPy is mostly used along with Scipy (Scientific Python) and Mat-plotlib (plotting library).

Execution :

import numpy as np

- **Pandas**

It is another useful Python library that makes Python one of the favorite languages for Data Science. Pandas is basically used for data manipulation, wrangling and analysis. It was developed by Wes McKinney in 2008. With the help of Pandas, in data processing we can accomplish the following five steps −

- Load
- Prepare
- Manipulate
- Model
- Analyze

Data representation in Pandas

The entire representation of data in Pandas is done with the help of following three data structures −

Series − It is basically a one-dimensional ndarray with an axis label which means it is like a simple array with homogeneous data. For example, the following series is a collection of integers 1,5,10,15,24,25...

1	5	10	15	24	25	28	36	40	89

Data frame − It is the most useful data structure and used for almost all kind of data representation and manipulation in pandas. It is basically a two-dimensional data structure which can contain heterogeneous data. Generally, tabular data is represented by using data frames. For example, the following table shows the data of students having their names and roll numbers, age and gender.

Name	Roll number	Age	Gender

Aarav	1	15	Male
Harshit	2	14	Male
Kanika	3	16	Female
Mayank	4	15	Male

Table: 1

Panel – It is a 3-dimensional data structure containing heterogeneous data. It is very difficult to represent the panel in graphical representation, but it can be illustrated as a container of DataFrame.

The following table gives us the dimension and description about above mentioned data structures used in Pandas –

Data Structure	Dimension	Description
Series	1-D	Size immutable, 1-D homogeneous data
DataFrames	2-D	Size Mutable, Heterogeneous data in tabular form
Panel	3-D	Size-mutable array, container of DataFrame.

Table:

- **Scikit-learn**

Supervised learning algorithms: Think of any supervised machine learning algorithm you might have heard about and there is a very high chance that it is part of scikit-learn. Starting from Generalized linear models (e.g Linear Regression), Support Vector Machines (SVM), Decision Trees to Bayesian methods – all of them are part of scikit-learn toolbox. The spread of machine learning algorithms is one of the big reasons for the high usage of scikit-learn. I started using scikit to solve supervised learning problems and would recommend that to people new to scikit / machine learning as well.

Another useful and most important python library for Data Science and machine learning in Python is Scikit-learn. The following are some features of *Scikitlearn* that makes it so useful −

- It is built on NumPy, SciPy, and Matplotlib.
- It is an open source and can be reused under BSD license.
- It is accessible to everybody and can be reused in various contexts.
- Wide range of machine learning algorithms covering major areas of ML like classification, clustering, regression, dimensionality reduction, model selection etc. can be implemented with the help of it.

Execution:

```
from sklearn.model_selection import train_test_split
```

RESULTS

The figure 8 shows the user interface for test and train the data. Train and Test buttons are given to the user where using train the algorithms are trained and then o predict the fraud by clicking predict button it will take to another window where the input is given and output is seen as fraud or non fraud.

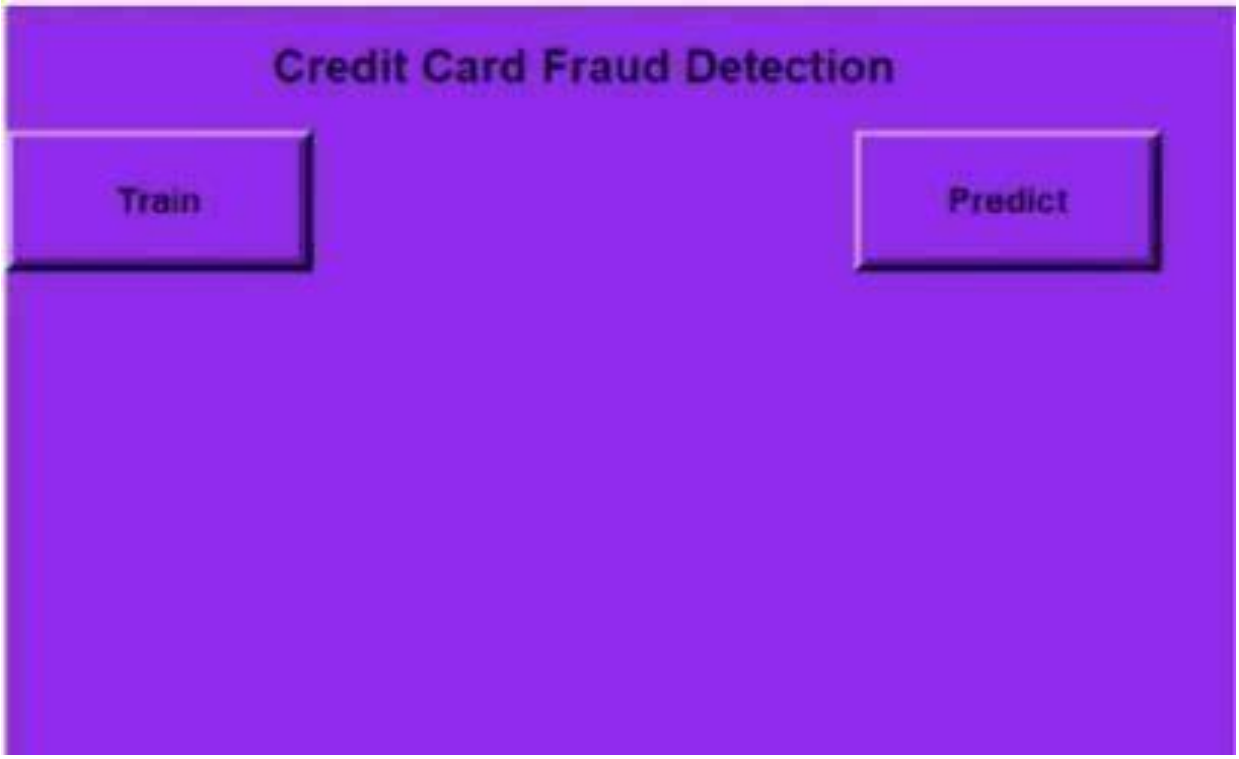


Fig. 8: User interface for train and test data

The figure 9 shows detection of fraud or nonfraud transaction. when predict button is clicked it will take to another window where it asks for data which is input to the machine learning algorithms and in the predict it will give output as fraud or nonfraud. comma separated 30 values are given including amount and time. Predicted result is displayed as fraud after providing the data. These results along with the as follows, where class 0 means the transaction was determined to be valid and 1 means it was determined as a fraud transaction.

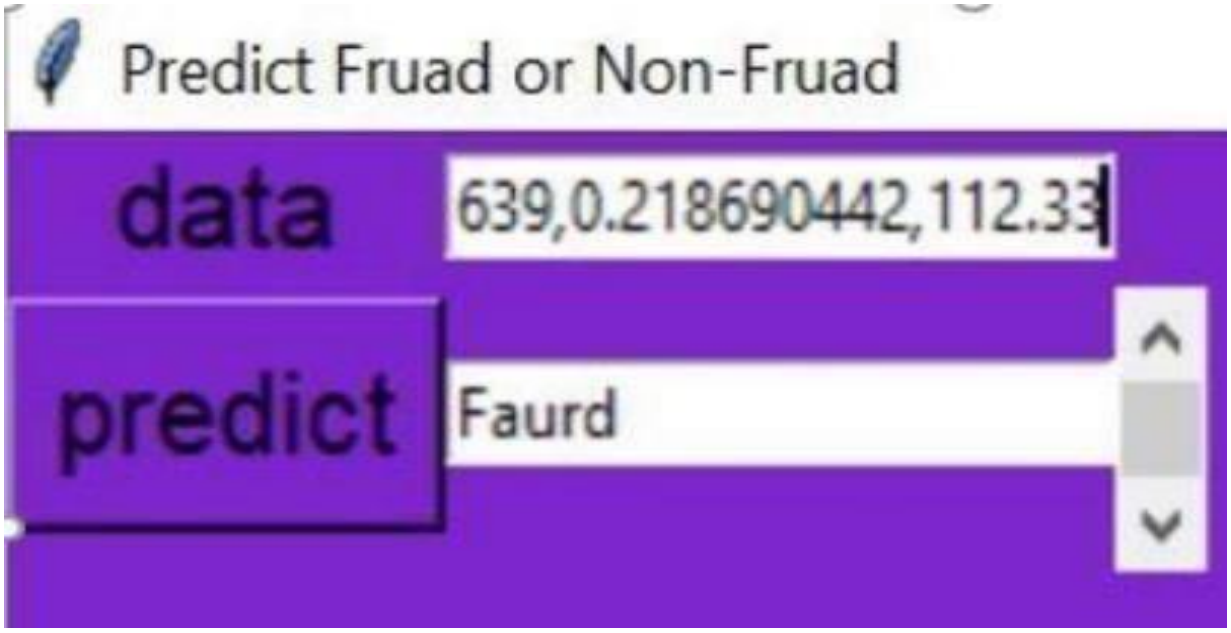


Fig. 9: Detection of fraud or normal transaction 1)Confusion matrix for Logistic regression Algorithm:

Fig 10 represents confusion matrix for Logistic regression algorithm. Which contains True Positive, True Negative, False Positive, False Negative. False positive value is lesser which shows fraud not detected cases are low. For logistic regression algorithm accuracy, recall, precision achieved are 94.84, 92.00, 97.58 respectively.

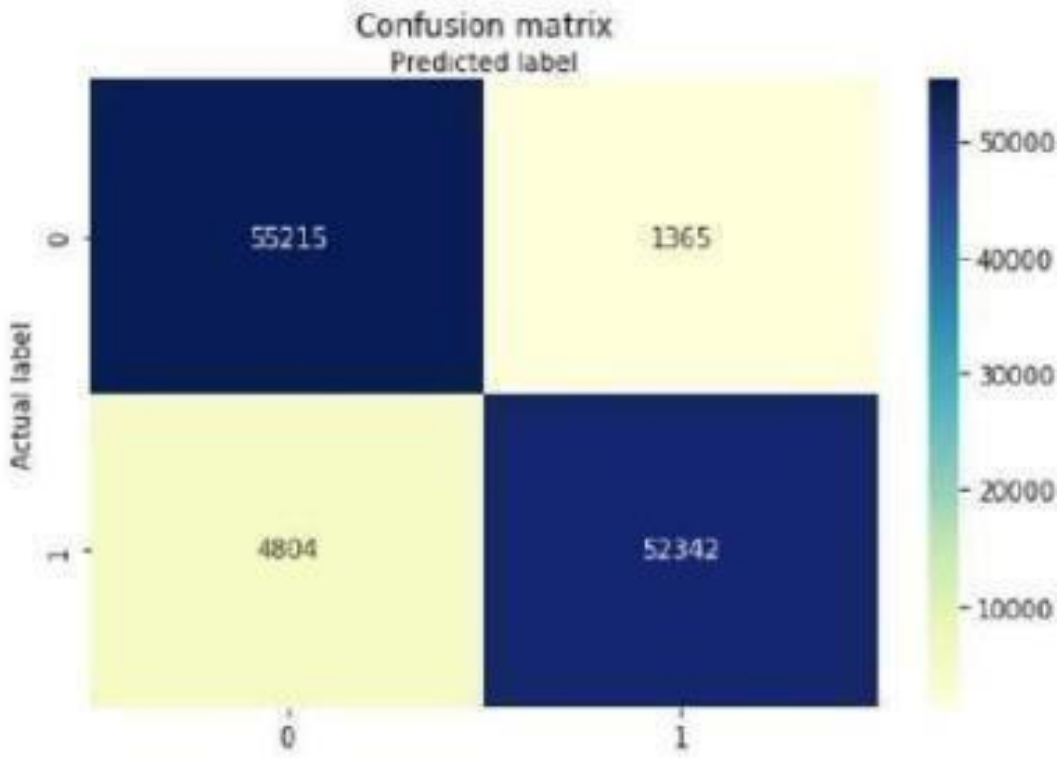


Fig. 10: Confusion matrix for Logistic regression

2) Confusion matrix for Naive Bayes Algorithm:

Fig 11 represents confusion matrix for Naive Bayes algorithm. Which contains True Positive, True Negative, False Positive, False Negative. False positive value is lesser which shows fraud not detected cases are low. For Naive Bayes algorithm accuracy, recall, precision achieved are 91.62, 84.82, 97.09 respectively.

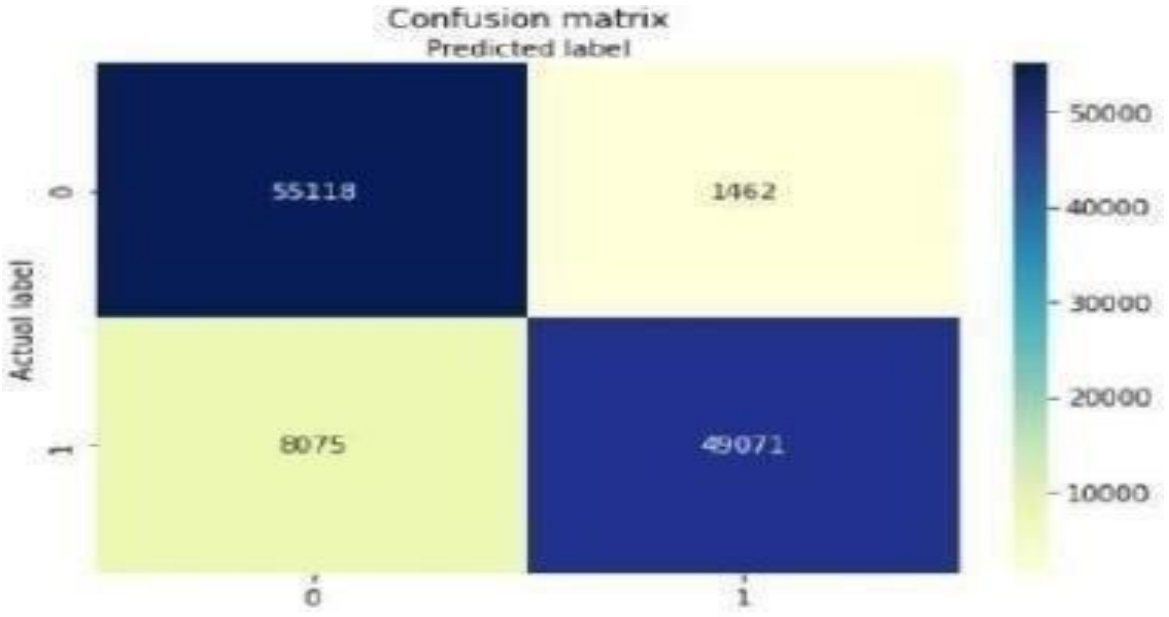


Fig. 11: Confusion matrix for Naive Bayes

3)Confusion matrix for Decision Tree Algorithm:

Fig 12 represents confusion matrix for Decision Tree algorithm. Which contains True Positive, True Negative, False Positive, False Negative. False positive value is lesser which shows fraud not detected cases are low. For Decision Tree algorithm accuracy, recall, precision achieved are 92.88, 98.98, 99.48 respectively.

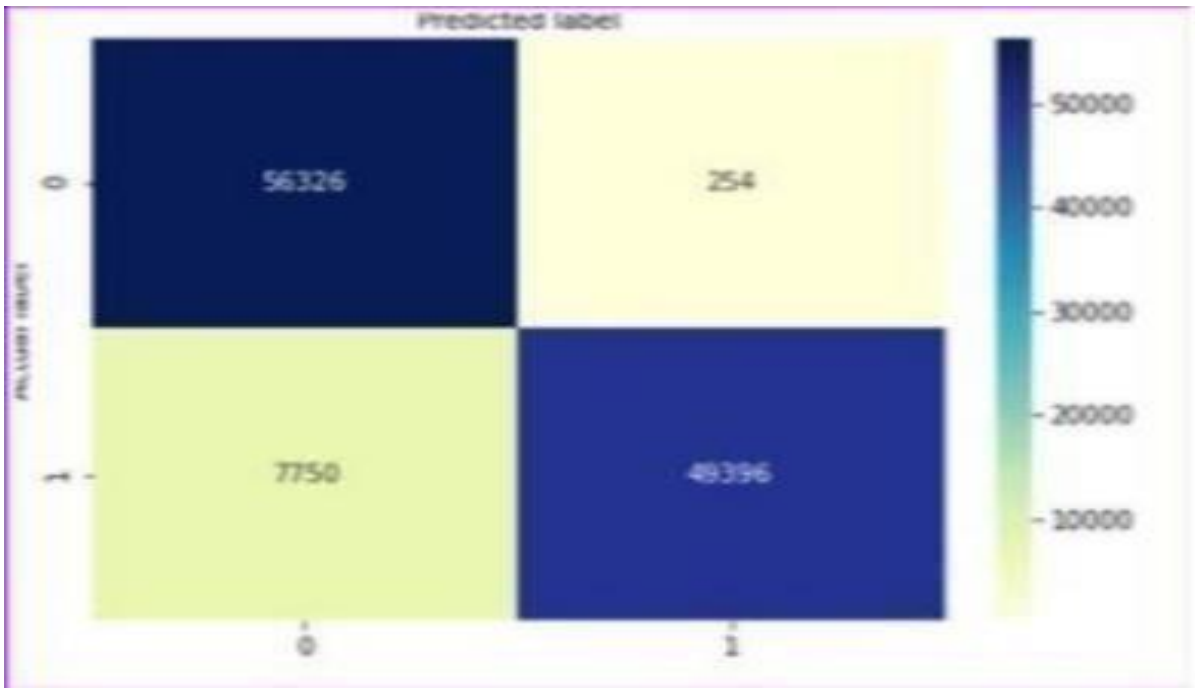


Fig. 12: Confusion matrix for Decision Tree 4)Confusion matrix for ANN model:

Fig 13 represents confusion matrix for ANN model. Which contains True Positive, True Negative, False Positive, False Negative. False positive value is lesser which shows fraud not detected cases are low. For ANN model algorithm accuracy, recall, precision achieved are 98.69, 98.98, 98.41 respectively.

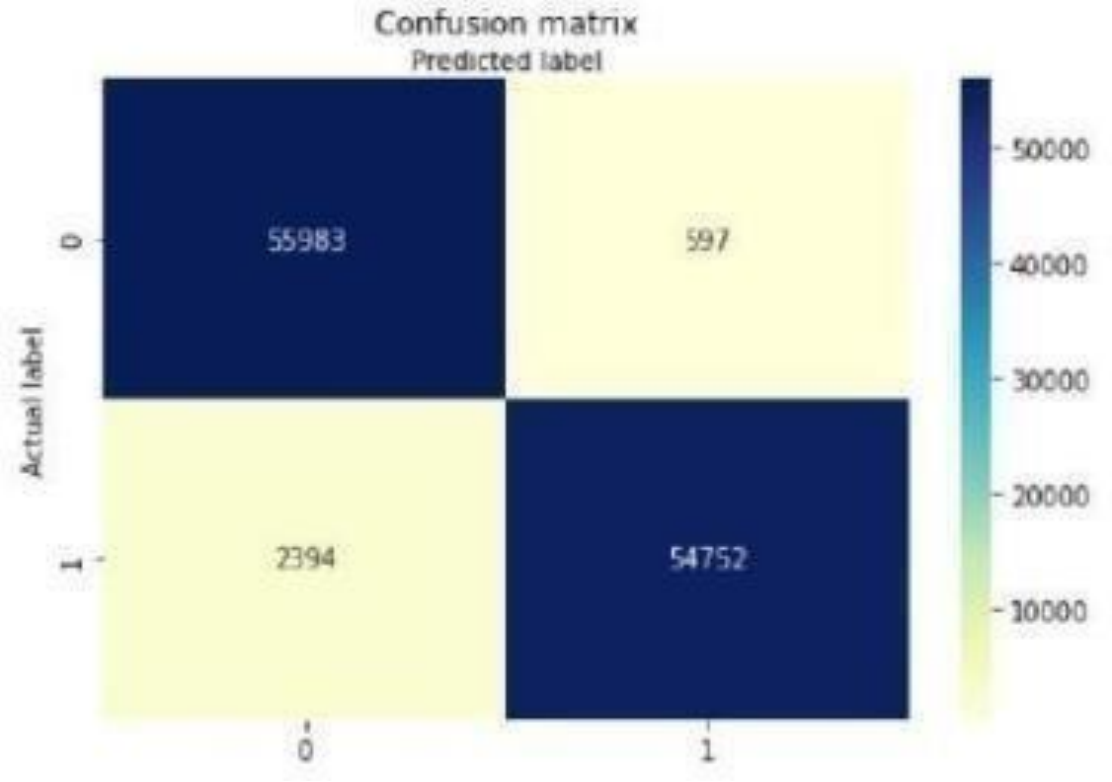


Fig. 13: Confusion matrix for ANN

Comparison of algorithms:

Table 3 represents the comparison table made using results obtained using simulation. Factors compared are accuracy, precision, recall. From table we can conclude that ANN model as best accuracy, precision and recall.

Achievement of accuracy is done using different algorithms and Ann model gives the best accuracy. confusion matrix gives visualization of results in the form table and minimum false positive rate is seen in all algorithms which is required results to achieve the objective. finally by providing the numerical data fraud or nonfraud detected using basic user interface design.

[Grab your reader's attention with a great quote from the document or use this space to emphasize a key point. To place this text box anywhere on the page, just drag it.]

	Accuracy	Precision	Recall
Logistic Regression	94.84	97.58	92.00
Naive Bayes	91.62	97.09	84.82
Decision Tree	92.88	99.48	86.34
ANN model	98.69	98.41	98.98

Table 3: Accuracy, precision, recall comparison table for different ML algorithms

SYSTEM ANALYSIS

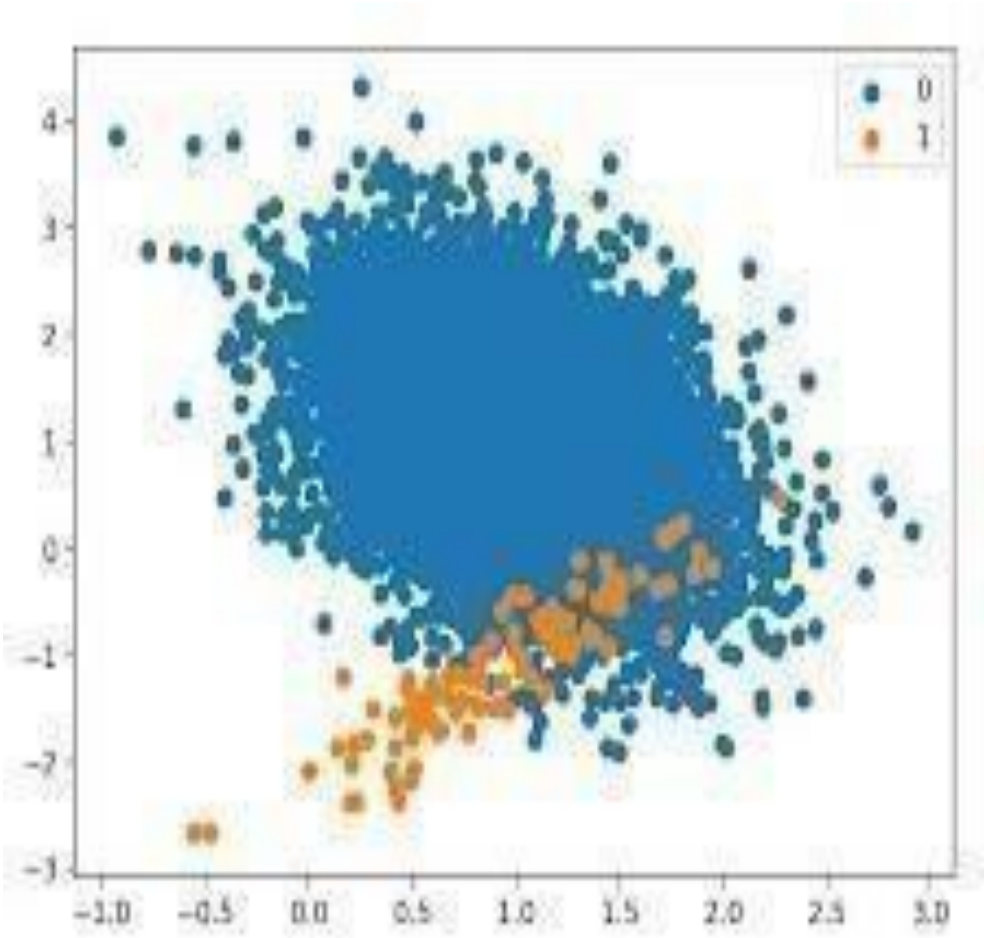
- Since the credit card fraud detection system is a highly researched field, there are many different algorithms and techniques for performing the credit card fraud detection system.

- One of the earliest systems is CCFD system using Markov model. Some other various existing algorithms used in the credit cards fraud detection system includes Cost sensitive decision tree (CSDT).

- Credit card fraud detection (CCFD) is also proposed by using neural networks. The existing credit card fraud detection system using neural network follows the whale swarmoptimization algorithm to obtain an

incentive value. • It the uses BP network to rectify the values which are found error.

Figure.. Fraud and Non Fraud Representation



Proposed System

Support Vector Machine:

SVM works by mapping data to a high-dimensional feature space so that data points can be categorized, even when the data are not otherwise linearly separable. A separator between the categories is found, then the data are transformed in such a way that the separator could be drawn as a hyperplane Training regression model and finding out the best one.

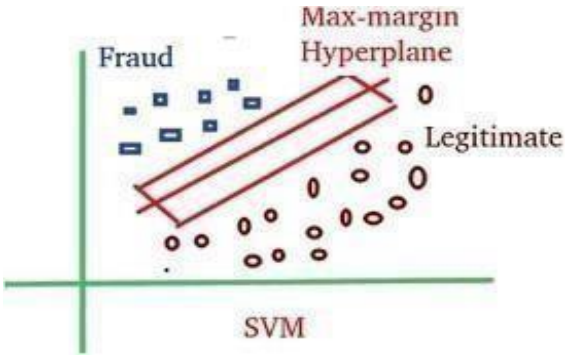
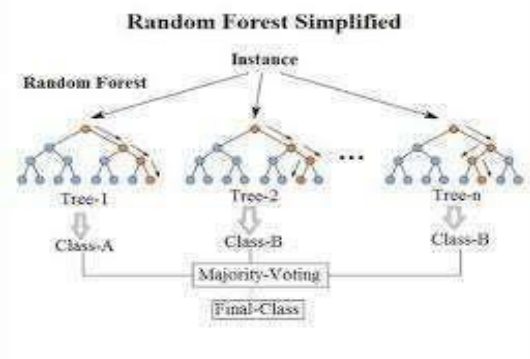


Fig SVM Representation

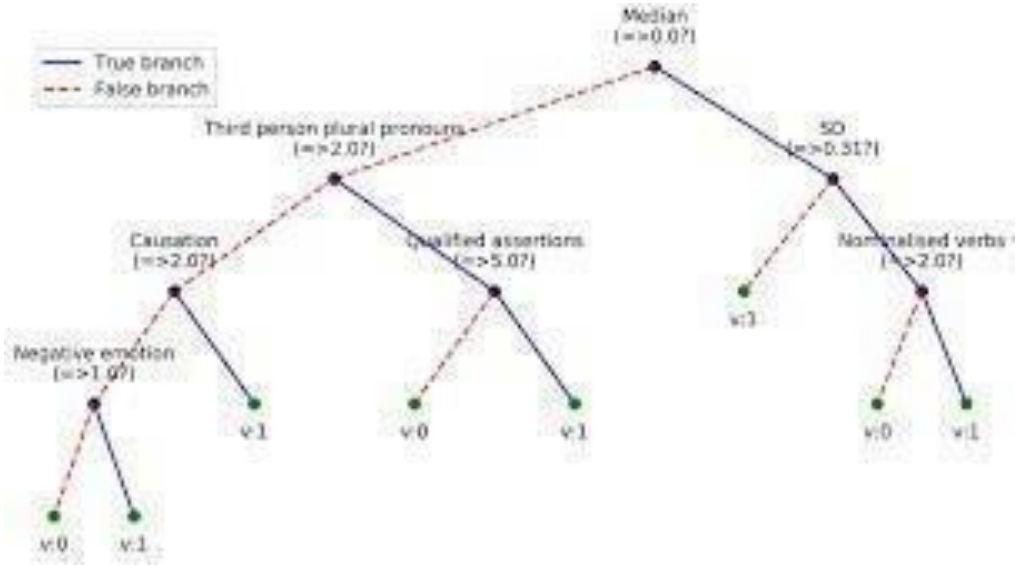
Random Forest Classifier

Features are cheekbone to jaw width, width to upper facial height ratio, perimeter to area ratio, eye size, lower face to face height ratio, face width to lower face height ratio and mean of eyebrow height. The extracted features are normalized and finally subjected to support regression.



Decision Tree

A decision tree is a type of supervised machine learning used to categorize or make predictions based on how a previous set of questions were answered. The model is a form of supervised learning, meaning that the model is trained and tested on a set of data that contains the desired categorization.



4 Decision tree Algorithm

4 Advantages

- Support vector machine works comparably well when there is an understandable margin of dissociation between classes.
- SVM is effective in instances where the number of dimensions is larger than the number of specimens.
- Simple to understand and to interpret.
- Requires little data preparation.
- The cost of using the tree (i.e., predicting data) is logarithmic in the number of data points used to train the tree.
- Able to handle both numerical and categorical data.
- Random forest classifier can be used to solve for regression or classification problems.

- The random forest algorithm is made up of a collection of decision trees, and each tree in the ensemble is

comprised of a data sample drawn from a training set with replacement, called the bootstrap sample.

SYSTEM DESIGN

CHAPTER-7 SYSTEM DESIGN

5.1 Project Modules

Entire project is divided into 3 modules as follows:

Data Gathering and pre processing

Training the model using following Machine Learning algorithms

i. SVM

ii. Random Forest Classifier iii. Decision Tree

Module 1: Data Gathering and Data Pre processing

a. A proper dataset is searched among various available ones and finalized with the dataset.

b. The dataset must be preprocessed to train the model.

c. In the preprocessing phase, the dataset is cleaned and any redundant values, noisy dataandnull values are removed.

d. The Preprocessed data is provided as input to the module.

Module 2: Training the model

a. The Preprocessed data is split into training and testing datasets in the 80:20 ratio to avoidthe problems of over-fitting and under-fitting.

b. A model is trained using the training dataset with the following algorithms SVM, Random Forest Classifier and Decision Tree

c. The trained models are trained with the testing data and results are visualized using bar graphs, scatter plots.

d. The accuracy rates of each algorithm are calculated using different params like F1 score, Precision, Recall. The results are then displayed using various data visualization tools for analysis purpose.
e. The algorithm which has provided the better accuracy rate compared to remaining algorithms is taken as final prediction model.

Module 3: Final Prediction model integrated with front end

a. The algorithm which has provided better accuracy rate has considered as the finalprediction model.
b. The model thus made is integrated with front end.
c. Database is connected to the front end to store the user information who are using it.

SYSTEM ARCHITECTURE

Our Project main purpose is to making Credit Card Fraud Detection awaring to people from credit card online frauds. the main point of credit card fraud detection system is necessaryto safe our transactions & security. With this system, fraudsters don't have the chance to make multiple transactions on a stolen or counterfeit card before the cardholder is aware of the fraudulent activity. This model is then used to identify whether a new transaction is fraudulent or not. Our aim here is to detect 100% of the fraudulent transactions while minimizing the incorrect fraud classifications.

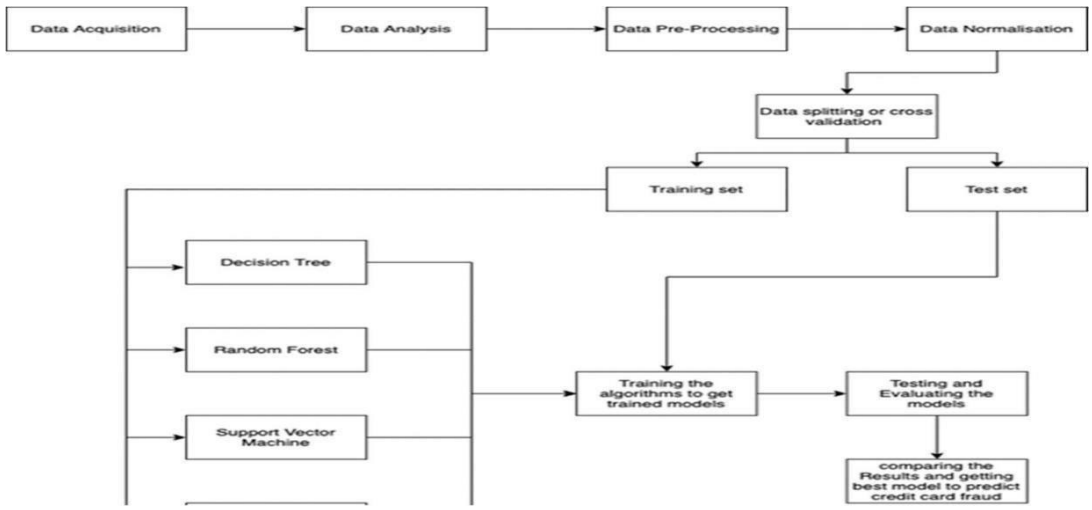


Fig 5.1 System Architecture

Activity diagram

Activity diagram is an important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity toanother activity. The activity can be described as an operation of the system. The control flow is drawnfrom one operation to another. This flow can be sequential, branched, or concurrent. Activity diagramsdeal with all type of flow control by using different elements such as fork, join, etc. The basic purposesof activity diagram are it captures the dynamic behavior of the system. Activity diagram is used to showmessage flow from one activity to another Activity is a particular operation of the system. Activitydiagrams are not only used for visualizing the dynamic nature of a system, but they are also used to construct the executable system by using forward and reverse engineering techniques.The only missing thing in the activity diagram is the message part.

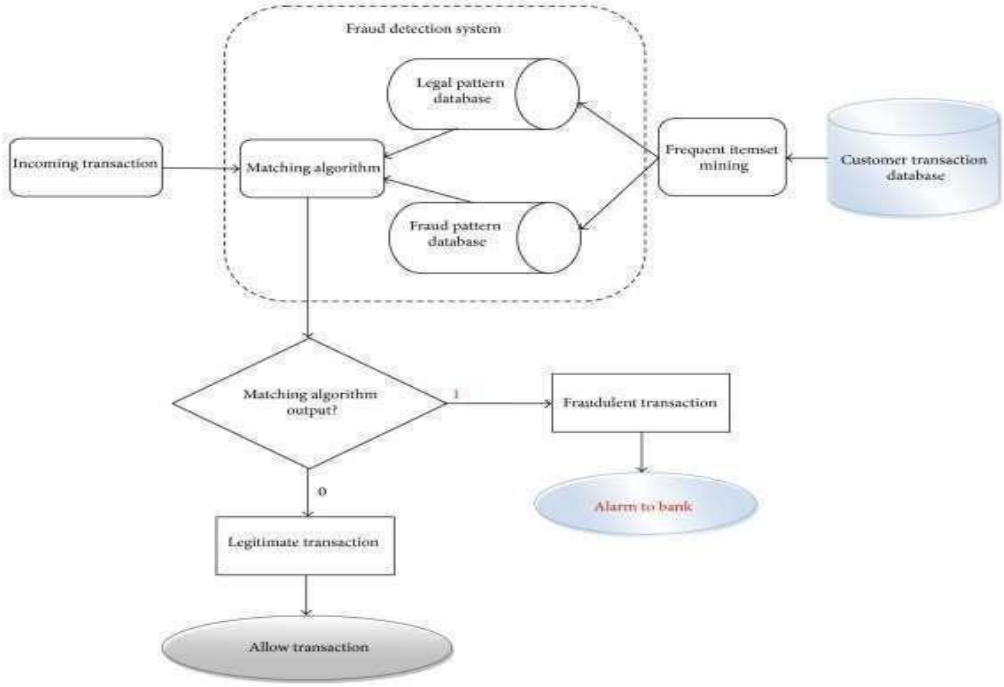


Fig 5.2 Activity Diagram

Use case diagram

In UML, use-case diagrams model the behavior of a system and help to capture the requirements of the system. Use-case diagrams describe the high-level functions and scope of a system. These diagrams also identify the interactions between the system and its actors. The use cases and actors in use-case diagrams describe what the system does and how the actors use it, but not how the system operates internally. Use-case diagrams illustrate and define the context and requirements of either an entire system or the important parts of the system. You can model a complex system with a single use-case diagram, or create many use-case diagrams to model the components of the system. You would typically develop use- case diagrams in the early phases of a project and refer to them throughout the development process.

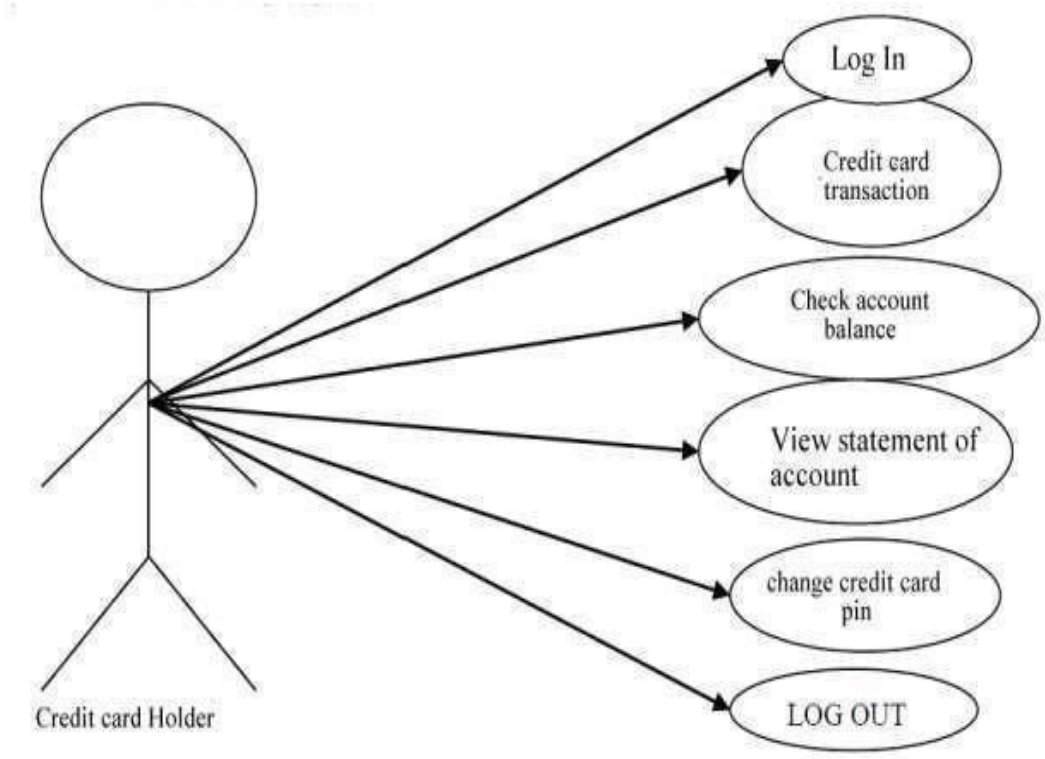


Fig 5.3 Use case Diagram

Sequence Diagram

The sequence diagram represents the flow of messages in the system and is also termed as an event diagram. It helps in envisioning several dynamic scenarios. It portrays the communication between any two lifelines as a time-ordered sequence of events, such that these lifelines took part at the run time. In UML, the lifeline is represented by a vertical bar, whereas the message flow is represented by a vertical dotted line that extends across the bottom of the page. It incorporates the iterations as well as branching.

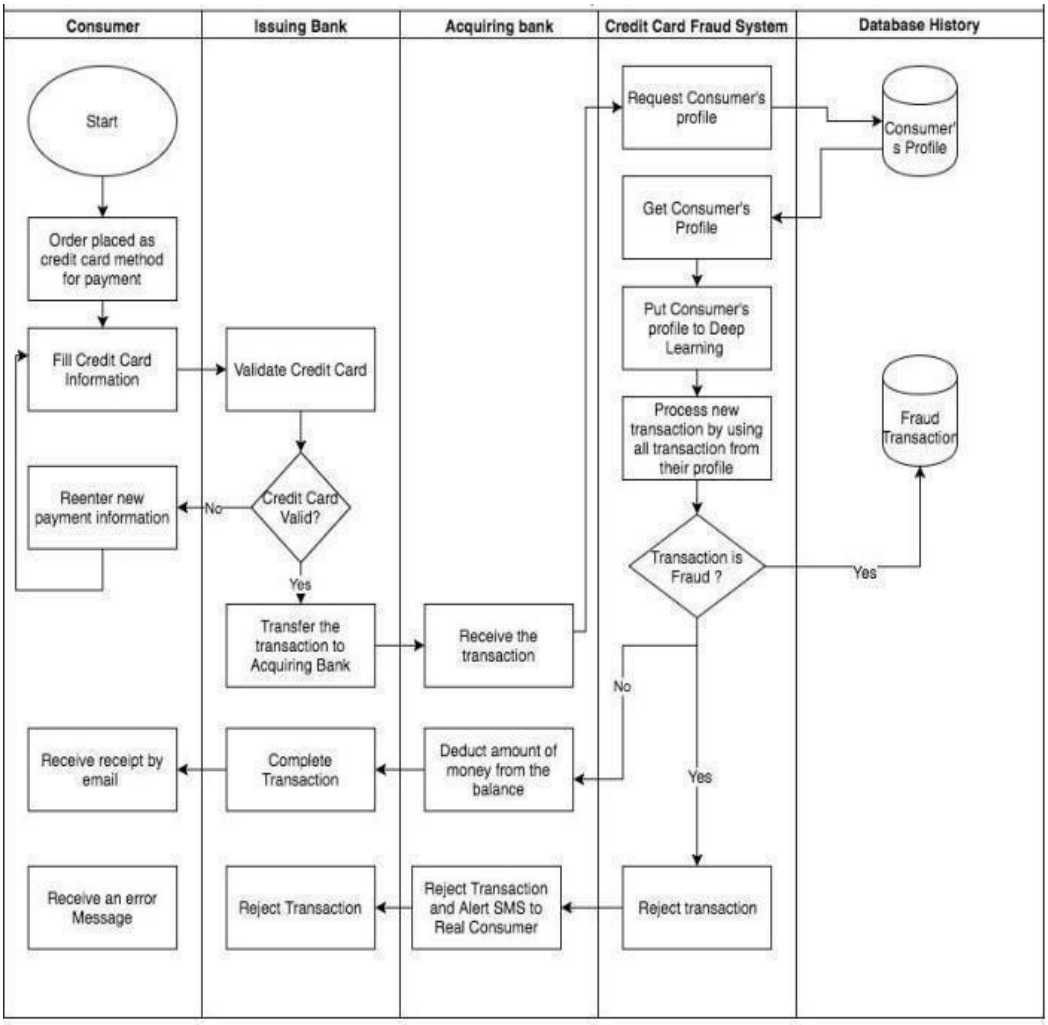


Fig 5.4 Sequence diagram

Data Flow Diagram

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirementgraphically. It can be manual, automated, or a combination of both. It shows how data entersand leaves the system, what changes the information, and where data is stored. The objectiveof a DFD is to show the scope and boundaries of a system as a whole. It may be used as a communication tool between a system analyst and any person who plays a part in the order that acts as a starting point for redesigning a system. The DFD is also called as a data flow graph or bubble chart.

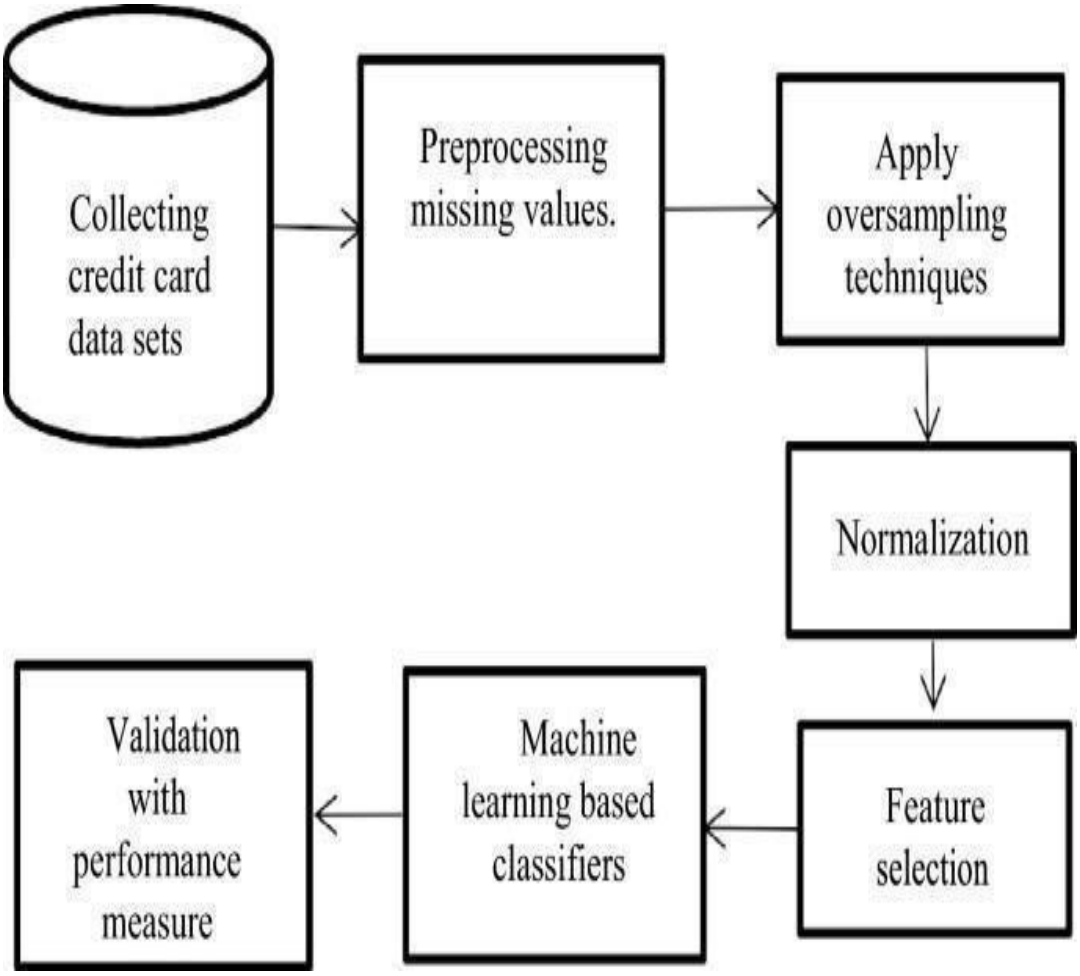
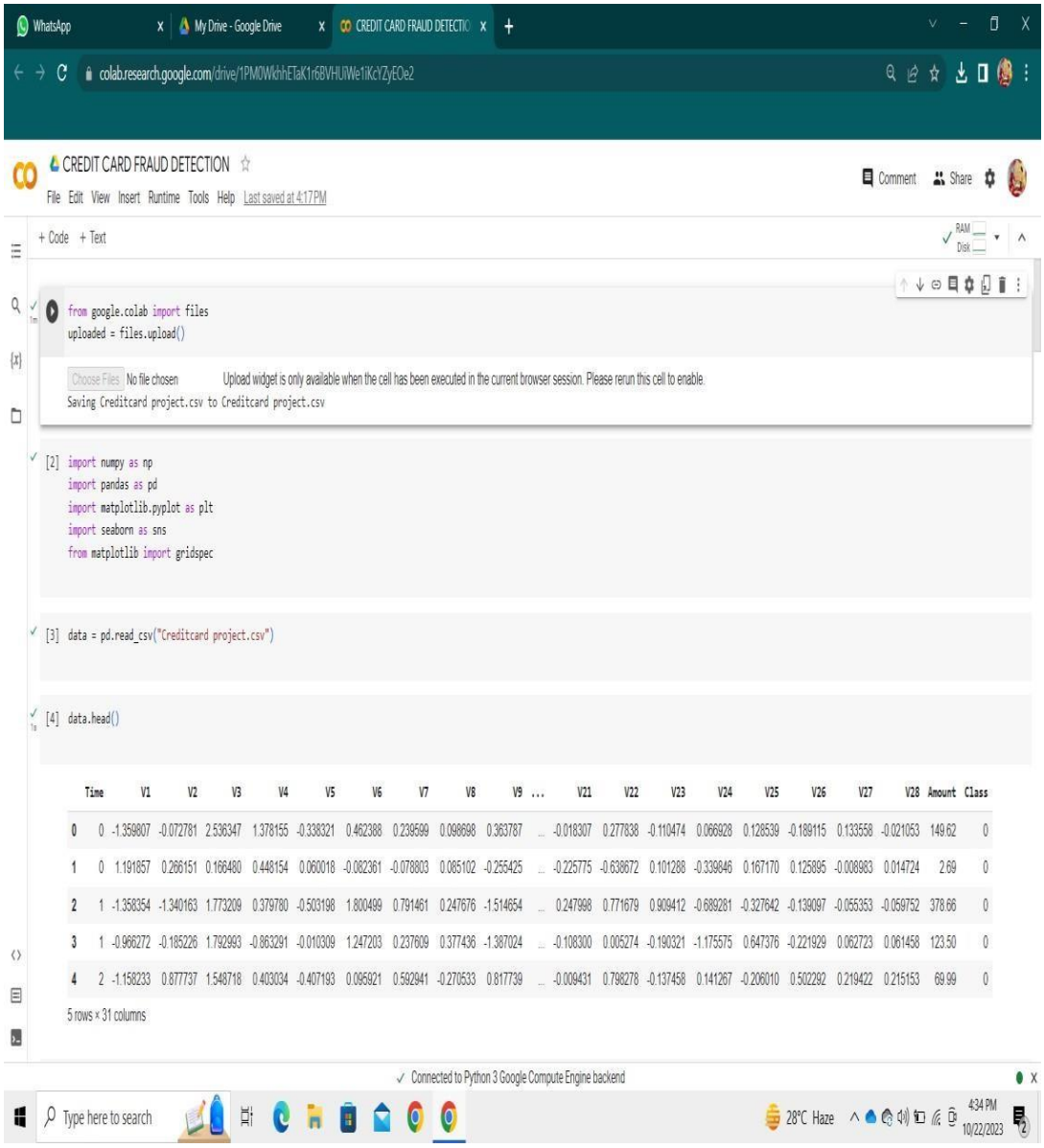
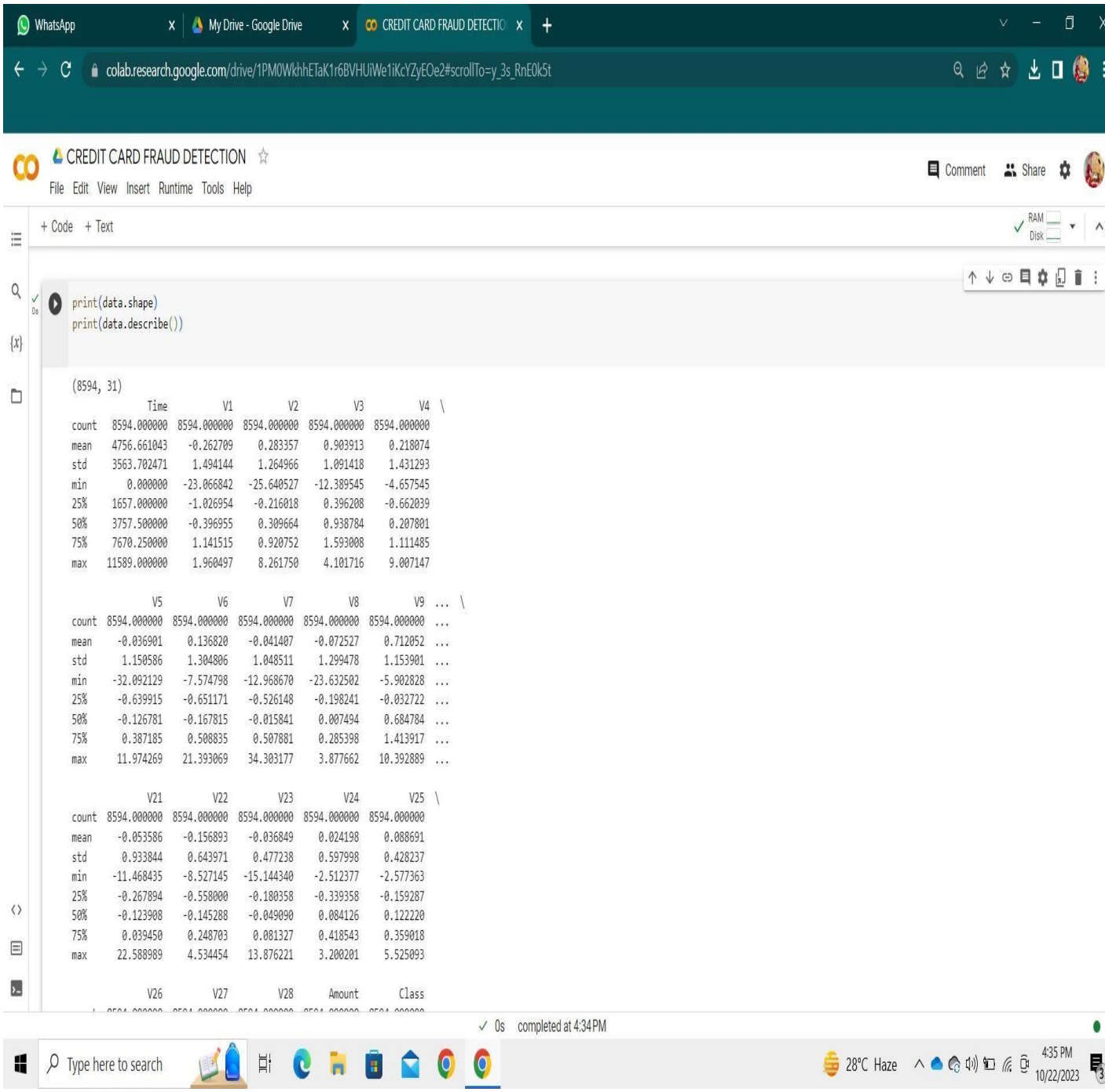
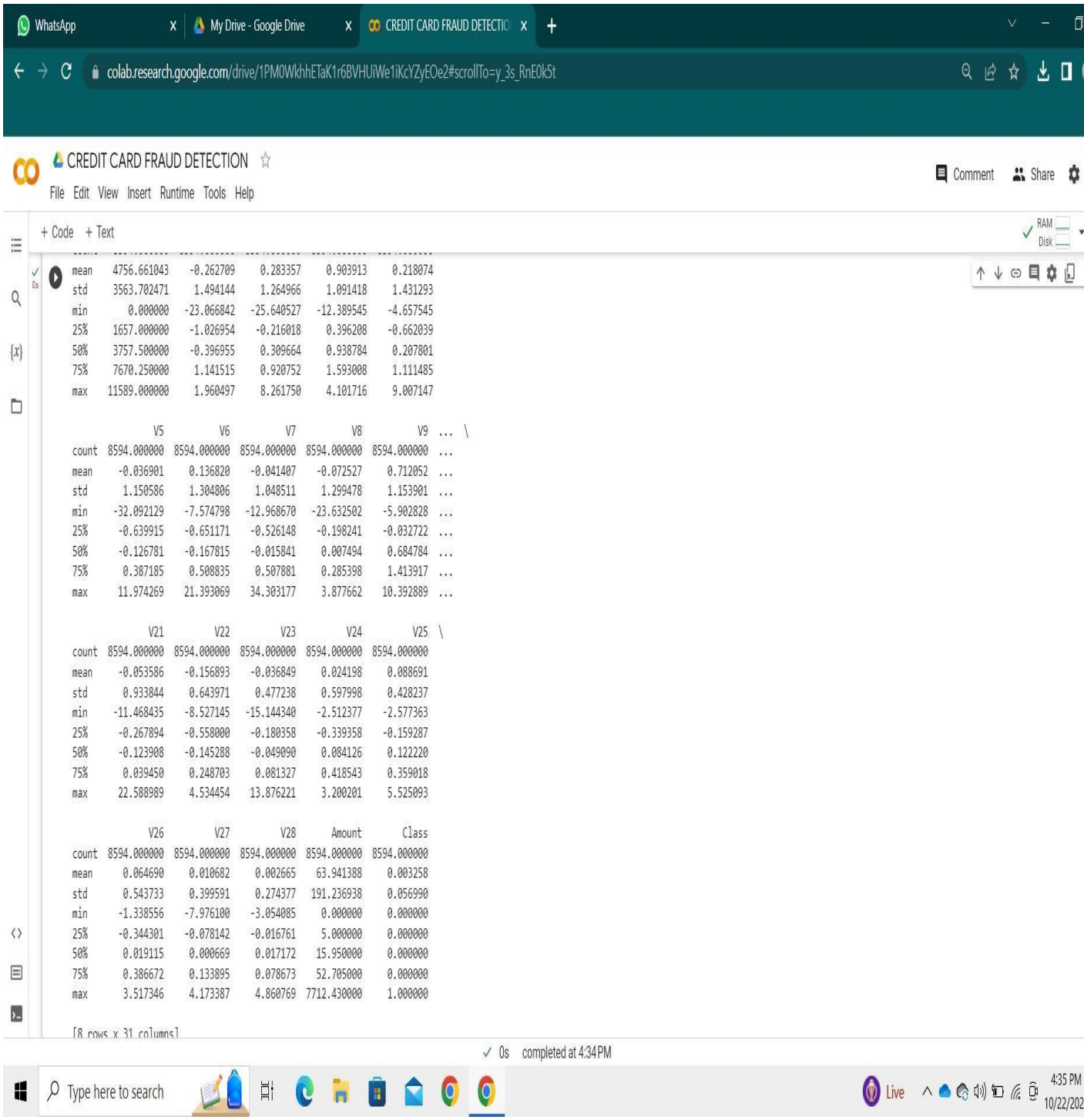


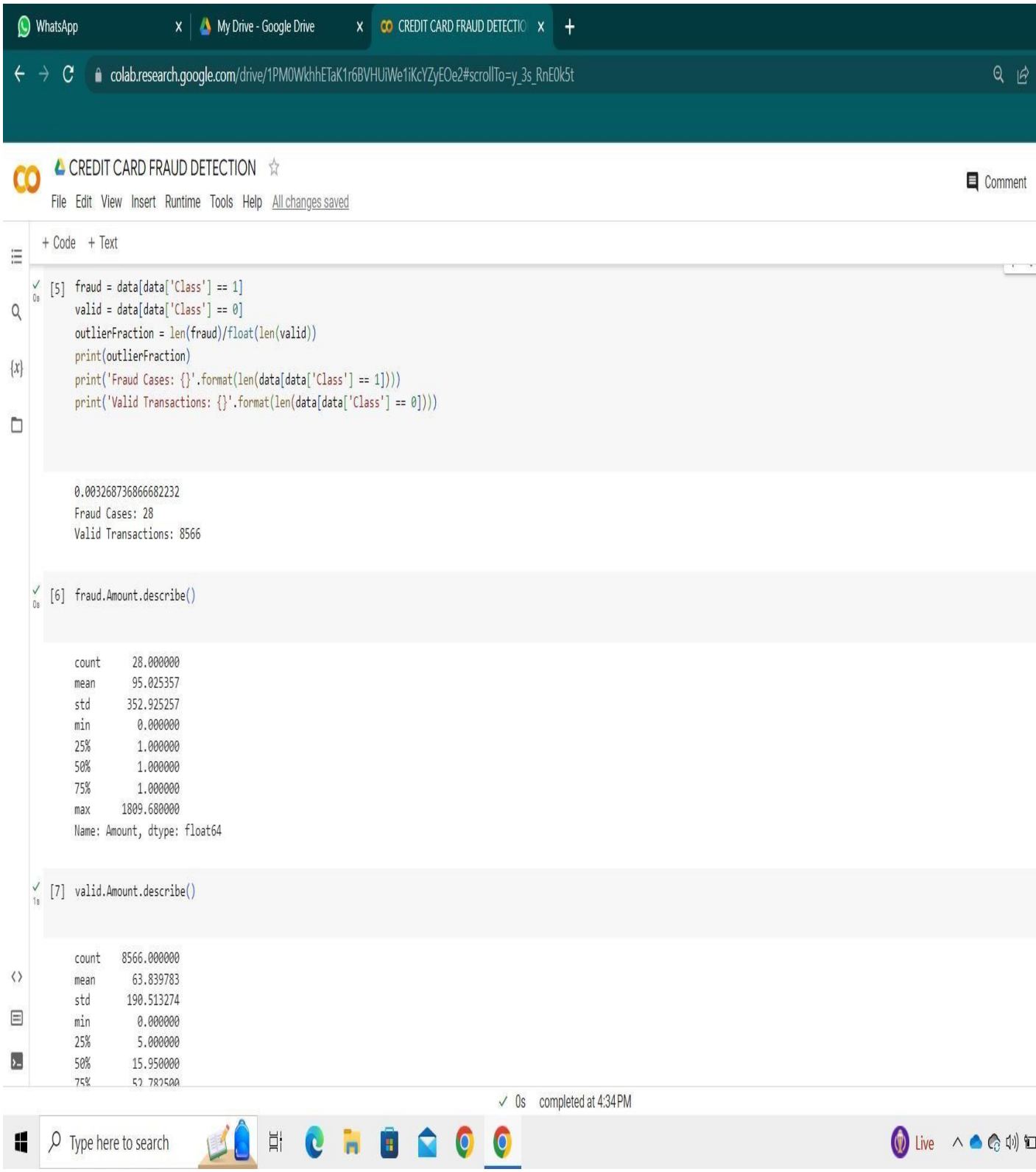
Fig 5.5 Data Flow diagram

Code


WhatsApp
My Drive - Google Drive
CREDIT CARD FRAUD DETECTIO
colab.research.google.com/drive/1PM0WkhhETaK1r6BVHUIWe1iKcYZyEOe2
CREDIT CARD FRAUD DETECTION
File Edit View Insert Runtime Tools Help Last saved at 4:17 PM
Comment
Share
+ Code + Text
RAM
Disk
from google.colab import files
uploaded = files.upload()
Choose Files No file chosen
Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.
Saving Creditcard project.csv to Creditcard project.csv
[2] import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from matplotlib import gridspec
[3] data = pd.read_csv("Creditcard project.csv")
[4] data.head()
Time V1 V2 V3 V4 V5 V6 V7 V8 V9 ... V21 V22 V23 V24 V25 V26 V27 V28 Amount Class
0 0 -1.359807 -0.072781 2.536347 1.378155 -0.338321 0.462388 0.239599 0.098698 0.363787 ... -0.018307 0.277838 -0.110474 0.066928 0.128539 -0.189115 0.133558 -0.021053 149.62 0
1 0 1.191857 0.266151 0.166480 0.448154 0.060018 -0.082361 -0.078803 0.085102 -0.255425 ... -0.225775 -0.638672 0.101288 -0.339846 0.167170 0.125895 -0.008983 0.014724 2.69 0
2 1 -1.358354 -1.340163 1.773209 0.379780 -0.503198 1.800499 0.791461 0.247676 -1.514654 ... 0.247998 0.771679 0.909412 -0.689281 -0.327642 -0.139097 -0.055353 -0.059752 378.66 0
3 1 -0.966272 -0.185226 1.792993 -0.863291 -0.010309 1.247203 0.237609 0.377436 -1.387024 ... -0.108300 0.005274 -0.190321 -1.175575 0.647376 -0.221929 0.062723 0.061458 123.50 0
4 2 -1.158233 0.877737 1.548718 0.403034 -0.407193 0.095921 0.592941 -0.270533 0.817739 ... -0.009431 0.798278 -0.137458 0.141267 -0.206010 0.502292 0.219422 0.215153 69.99 0
5 rows × 31 columns
Connected to Python 3 Google Compute Engine backend
Type here to search
28°C Haze
4:34 PM
10/22/2023


WhatsApp
My Drive - Google Drive
CREDIT CARD FRAUD DETECTIO
colab.research.google.com/drive/1PM0WkhhETaK1r6BVHUiWe1iKcYZyEOe2#scrollTo=y_3s_RnE0k5t
CREDIT CARD FRAUD DETECTION
File Edit View Insert Runtime Tools Help
Comment
Share
+ Code + Text
RAM
Disk
print(data.shape)
print(data.describe())
(8594, 31)
Time V1 V2 V3 V4 \
count 8594.000000 8594.000000 8594.000000 8594.000000 8594.000000
mean 4756.661043 -0.262709 0.283357 0.903913 0.218074
std 3563.702471 1.494144 1.264966 1.091418 1.431293
min 0.000000 -23.066842 -25.640527 -12.389545 -4.657545
25% 1657.000000 -1.026954 -0.216018 0.396208 -0.662039
50% 3757.500000 -0.396955 0.309664 0.938784 0.207801
75% 7670.250000 1.141515 0.920752 1.593008 1.111485
max 11589.000000 1.960497 8.261750 4.101716 9.007147
V5 V6 V7 V8 V9 ... \
count 8594.000000 8594.000000 8594.000000 8594.000000 8594.000000 ...
mean -0.036901 0.136820 -0.041407 -0.072527 0.712052 ...
std 1.150586 1.304806 1.048511 1.299478 1.153901 ...
min -32.092129 -7.574798 -12.968670 -23.632502 -5.902828 ...
25% -0.639915 -0.651171 -0.526148 -0.198241 -0.032722 ...
50% -0.126781 -0.167815 -0.015841 0.007494 0.684784 ...
75% 0.387185 0.508835 0.507881 0.285398 1.413917 ...
max 11.974269 21.393069 34.303177 3.877662 10.392889 ...
V21 V22 V23 V24 V25 \
count 8594.000000 8594.000000 8594.000000 8594.000000 8594.000000
mean -0.053586 -0.156893 -0.036849 0.024198 0.088691
std 0.933844 0.643971 0.477238 0.597998 0.428237
min -11.468435 -8.527145 -15.144340 -2.512377 -2.577363
25% -0.267894 -0.558000 -0.180358 -0.339358 -0.159287
50% -0.123908 -0.145288 -0.049090 0.084126 0.122220
75% 0.039450 0.248703 0.081327 0.418543 0.359018
max 22.588989 4.534454 13.876221 3.200201 5.525093
V26 V27 V28 Amount Class
0s completed at 4:34PM
Type here to search
28°C Haze
4:35 PM
10/22/2023


WhatsApp
My Drive - Google Drive
CREDIT CARD FRAUD DETECTIO
colab.research.google.com/drive/1PM0WkhhETaK1r6BVHUiWe1iKcYZyEOe2#scrollTo=y_3s_RnE0k5t
CREDIT CARD FRAUD DETECTION
File Edit View Insert Runtime Tools Help
Comment
Share
+ Code + Text
RAM
Disk
mean 4756.661043 -0.262709 0.283357 0.903913 0.218074
std 3563.702471 1.494144 1.264966 1.091418 1.431293
min 0.000000 -23.066842 -25.640527 -12.389545 -4.657545
25% 1657.000000 -1.026954 -0.216018 0.396208 -0.662039
50% 3757.500000 -0.396955 0.309664 0.938784 0.207801
75% 7670.250000 1.141515 0.920752 1.593008 1.111485
max 11589.000000 1.960497 8.261750 4.101716 9.007147
V5 V6 V7 V8 V9 ... \
count 8594.000000 8594.000000 8594.000000 8594.000000 8594.000000 ...
mean -0.036901 0.136820 -0.041407 -0.072527 0.712052 ...
std 1.150586 1.304806 1.048511 1.299478 1.153901 ...
min -32.092129 -7.574798 -12.968670 -23.632502 -5.902828 ...
25% -0.639915 -0.651171 -0.526148 -0.198241 -0.032722 ...
50% -0.126781 -0.167815 -0.015841 0.007494 0.684784 ...
75% 0.387185 0.508835 0.507881 0.285398 1.413917 ...
max 11.974269 21.393069 34.303177 3.877662 10.392889 ...
V21 V22 V23 V24 V25 \
count 8594.000000 8594.000000 8594.000000 8594.000000 8594.000000
mean -0.053586 -0.156893 -0.036849 0.024198 0.088691
std 0.933844 0.643971 0.477238 0.597998 0.428237
min -11.468435 -8.527145 -15.144340 -2.512377 -2.577363
25% -0.267894 -0.558000 -0.180358 -0.339358 -0.159287
50% -0.123908 -0.145288 -0.049090 0.084126 0.122220
75% 0.039450 0.248703 0.081327 0.418543 0.359018
max 22.588989 4.534454 13.876221 3.200201 5.525093
V26 V27 V28 Amount Class
count 8594.000000 8594.000000 8594.000000 8594.000000 8594.000000
mean 0.064690 0.010682 0.002665 63.941388 0.003258
std 0.543733 0.399591 0.274377 191.236938 0.056990
min -1.338556 -7.976100 -3.054085 0.000000 0.000000
25% -0.344301 -0.078142 -0.016761 5.000000 0.000000
50% 0.019115 0.000669 0.017172 15.950000 0.000000
75% 0.386672 0.133895 0.078673 52.705000 0.000000
max 3.517346 4.173387 4.860769 7712.430000 1.000000
[8 rows x 31 columns]
✓ 0s completed at 4:34PM
Type here to search
Live
4:35 PM
10/22/202


WhatsApp
My Drive - Google Drive
CREDIT CARD FRAUD DETECTIO
colab.research.google.com/drive/1PM0WkhhETaK1r6BVHUiWe1iKcYZyEOe2#scrollTo=y_3s_RnE0k5t
CREDIT CARD FRAUD DETECTION
File Edit View Insert Runtime Tools Help All changes saved
Comment
+ Code + Text
[5] 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])))
0.003268736866682232
Fraud Cases: 28
Valid Transactions: 8566
[6] fraud.Amount.describe()
count 28.000000
mean 95.025357
std 352.925257
min 0.000000
25% 1.000000
50% 1.000000
75% 1.000000
max 1809.680000
Name: Amount, dtype: float64
[7] valid.Amount.describe()
count 8566.000000
mean 63.839783
std 190.513274
min 0.000000
25% 5.000000
50% 15.950000
75% 52.782500
0s completed at 4:34 PM
Type here to search
Live

WhatsApp | My Drive - Google Drive | CREDIT CARD FRAUD DETECTIO

colab.research.google.com/drive/1PM0WkhhETaK1r6BVHUiWe1iKcYZyEOe2#scrollTo=y_3s_RnE0k5t

CREDIT CARD FRAUD DETECTION

File Edit View Insert Runtime Tools Help All changes saved

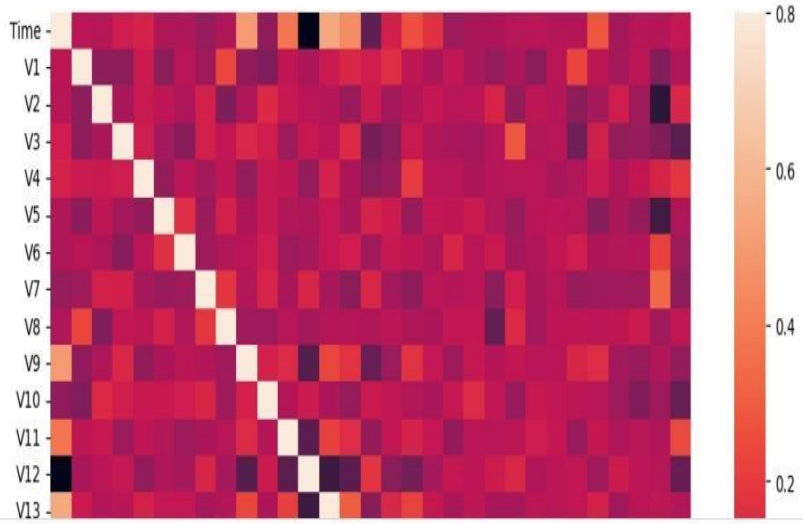
Comment

+ Code + Text

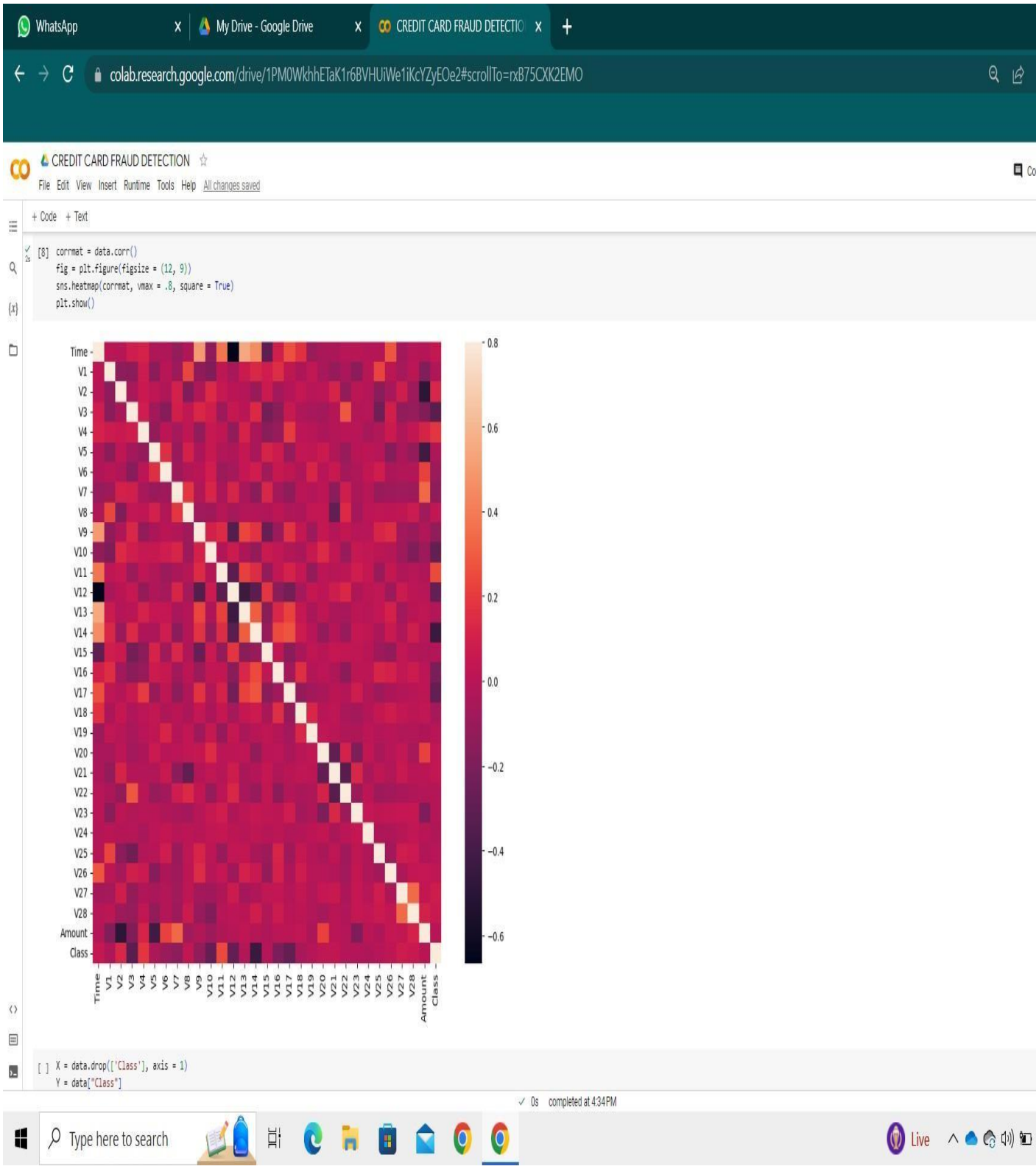
[7] valid.Amount.describe()

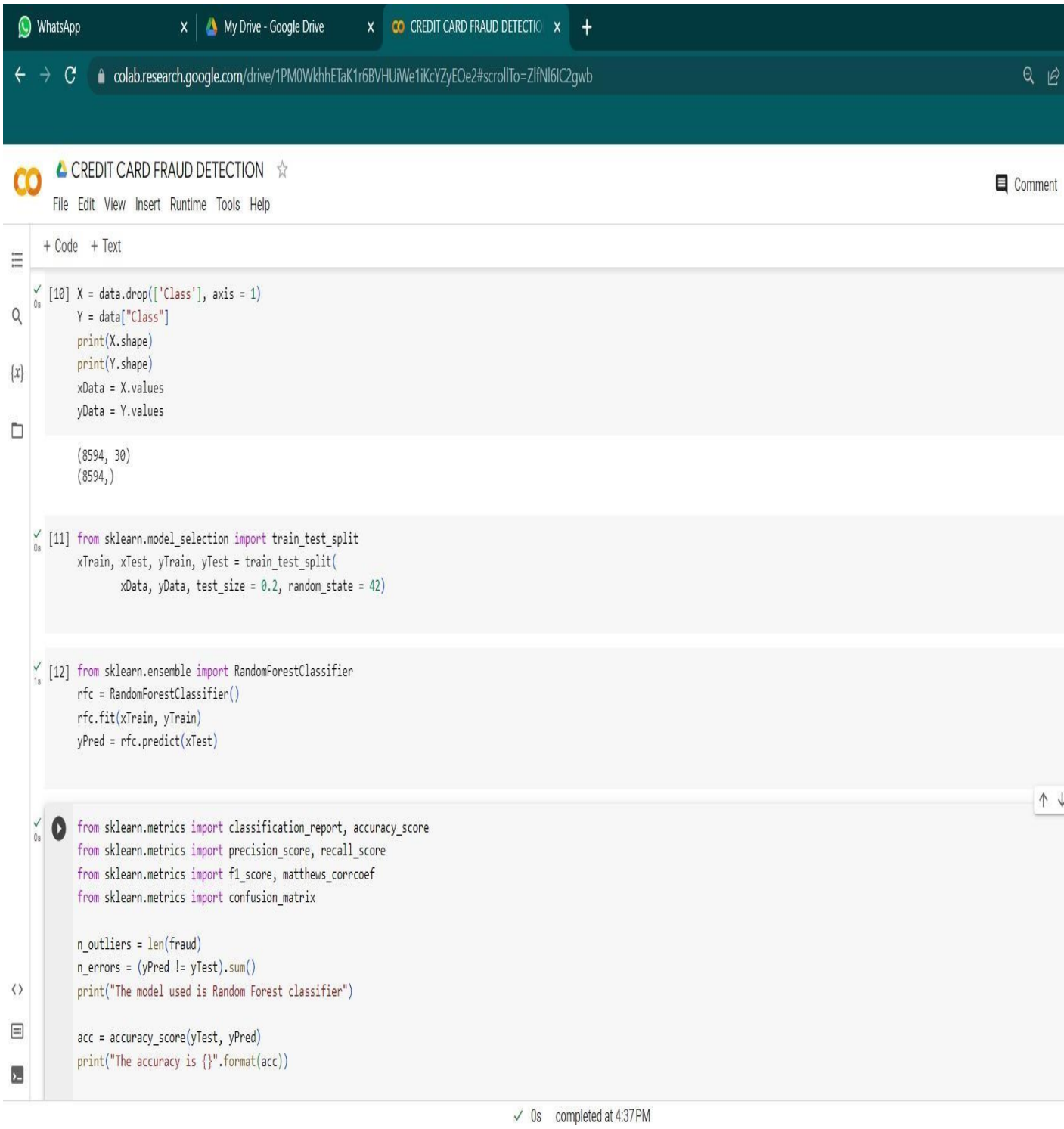
```
count    8566.000000
mean       63.839783
std       190.513274
min         0.000000
25%         5.000000
50%        15.950000
75%        52.782500
max      7712.430000
Name: Amount, dtype: float64
```

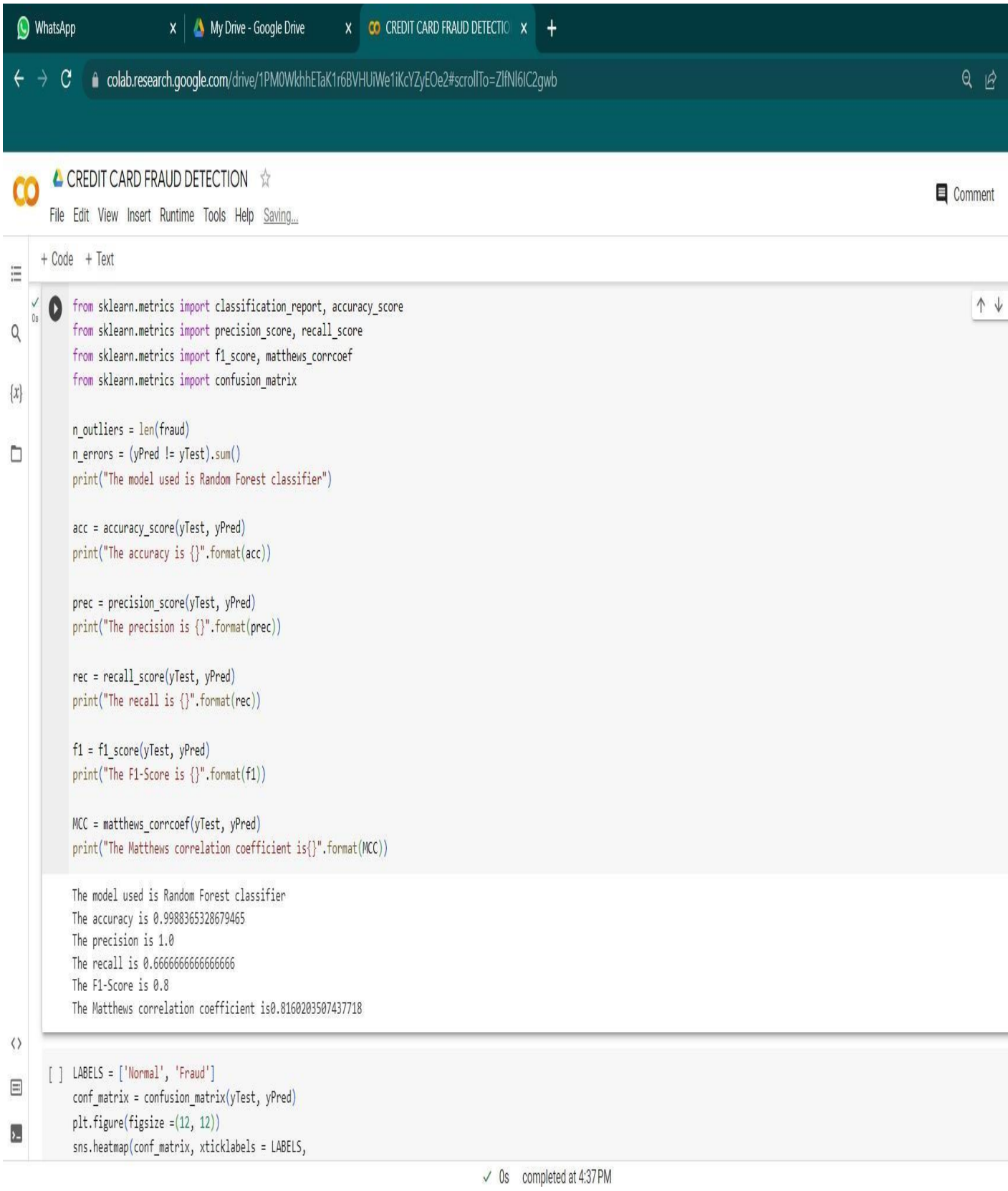
[8]
```
corrmat = data.corr()
fig = plt.figure(figsize = (12, 9))
sns.heatmap(corrmat, vmax = .8, square = True)
plt.show()
```

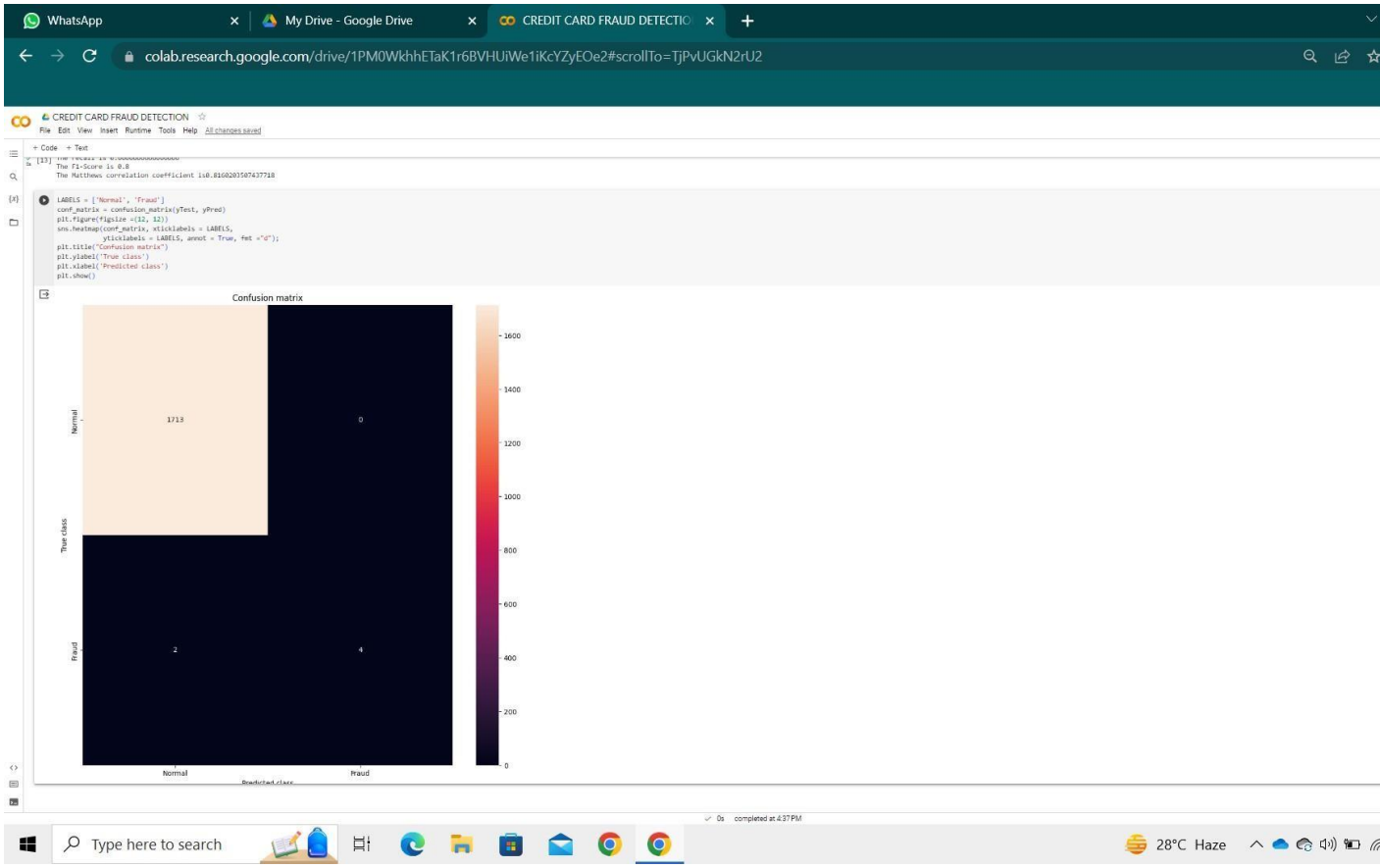


✓ 0s completed at 4:34PM


WhatsApp
My Drive - Google Drive
CREDIT CARD FRAUD DETECTIO
colab.research.google.com/drive/1PM0WkhhETaK1r6BVHUiWe1iKcYZyEOe2#scrollTo=rxB75CXK2EMO
CREDIT CARD FRAUD DETECTION
File Edit View Insert Runtime Tools Help All changes saved
+ Code + Text
[8] corrmat = data.corr()
fig = plt.figure(figsize = (12, 9))
sns.heatmap(corrmat, vmax = .8, square = True)
plt.show()
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
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4
−0.6
[ ] X = data.drop(['Class'], axis = 1)
Y = data["Class"]
0s completed at 4:34PM
Type here to search
Live


WhatsApp
My Drive - Google Drive
CREDIT CARD FRAUD DETECTIO
colab.research.google.com/drive/1PM0WkhhETaK1r6BVHUiWe1iKcYZyEOe2#scrollTo=ZlfNl6lC2gwb
CREDIT CARD FRAUD DETECTION
File Edit View Insert Runtime Tools Help
Comment
+ Code + Text
[10] X = data.drop(['Class'], axis = 1)
Y = data["Class"]
print(X.shape)
print(Y.shape)
xData = X.values
yData = Y.values
(8594, 30)
(8594,)
[11] from sklearn.model_selection import train_test_split
xTrain, xTest, yTrain, yTest = train_test_split(
xData, yData, test_size = 0.2, random_state = 42)
[12] from sklearn.ensemble import RandomForestClassifier
rfc = RandomForestClassifier()
rfc.fit(xTrain, yTrain)
yPred = rfc.predict(xTest)
from sklearn.metrics import classification_report, accuracy_score
from sklearn.metrics import precision_score, recall_score
from sklearn.metrics import f1_score, matthews_corrcoef
from sklearn.metrics import confusion_matrix
n_outliers = len(fraud)
n_errors = (yPred != yTest).sum()
print("The model used is Random Forest classifier")
acc = accuracy_score(yTest, yPred)
print("The accuracy is {}".format(acc))
0s completed at 4:37 PM


WhatsApp
My Drive - Google Drive
CREDIT CARD FRAUD DETECTIO
colab.research.google.com/drive/1PM0WkhhETaK1r6BVHUiWe1iKcYZyEOe2#scrollTo=ZlfNl6lC2gwb
CREDIT CARD FRAUD DETECTION
File Edit View Insert Runtime Tools Help Saving...
Comment
+ Code + Text
from sklearn.metrics import classification_report, accuracy_score
from sklearn.metrics import precision_score, recall_score
from sklearn.metrics import f1_score, matthews_corrcoef
from sklearn.metrics import confusion_matrix
n_outliers = len(fraud)
n_errors = (yPred != yTest).sum()
print("The model used is Random Forest classifier")
acc = accuracy_score(yTest, yPred)
print("The accuracy is {}".format(acc))
prec = precision_score(yTest, yPred)
print("The precision is {}".format(prec))
rec = recall_score(yTest, yPred)
print("The recall is {}".format(rec))
f1 = f1_score(yTest, yPred)
print("The F1-Score is {}".format(f1))
MCC = matthews_corrcoef(yTest, yPred)
print("The Matthews correlation coefficient is{}".format(MCC))
The model used is Random Forest classifier
The accuracy is 0.9988365328679465
The precision is 1.0
The recall is 0.6666666666666666
The F1-Score is 0.8
The Matthews correlation coefficient is0.8160203507437718
[ ] LABELS = ['Normal', 'Fraud']
conf_matrix = confusion_matrix(yTest, yPred)
plt.figure(figsize =(12, 12))
sns.heatmap(conf_matrix, xticklabels = LABELS,
✓ 0s completed at 4:37PM



CHAPTER-8 PERFORMANCE ANALYSIS

4.1 Performance metrics:

The basic performance measures derived from the confusion matrix. The confusion matrix is a 2 by 2 matrix table contains four outcomes produced by the binary classifier. Various measures such as sensitivity, specificity, accuracy and error rate are derived from theconfusion matrix.

Accuracy: Accuracy is calculated as the total number of two correct predictions(A+B) divided by thetotal number of the dataset(C+D). It is calculated as (1-error rate).

Accuracy=A+B/C+D

Whereas,

A=True Positive B=True Negative

C=Positive D=Negative

Error rate:

Error rate is calculated as the total number of two incorrect predictions(F+E) divided by thetotal number of the dataset(C+D).

Error rate=F+E/C+D

Whereas,

E=False Positive

F=False Negative

C=Positive

D=Negative

Sensitivity:

Sensitivity is calculated as the number of correct positive predictions(A) divided by thetotal number of positives(C).

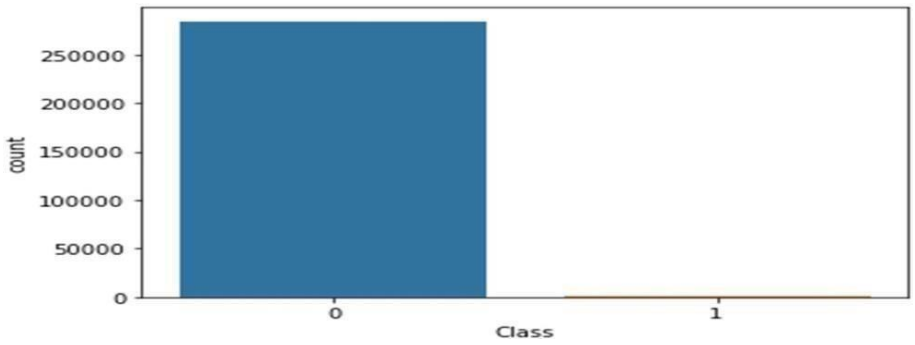
Sensitivity=A/C

Specificity: Specificity is calculated as the number of correct negative predictions(B) divided by the total number of negatives(D).

Specificity=B/D.

25

DATA ANALYSIS



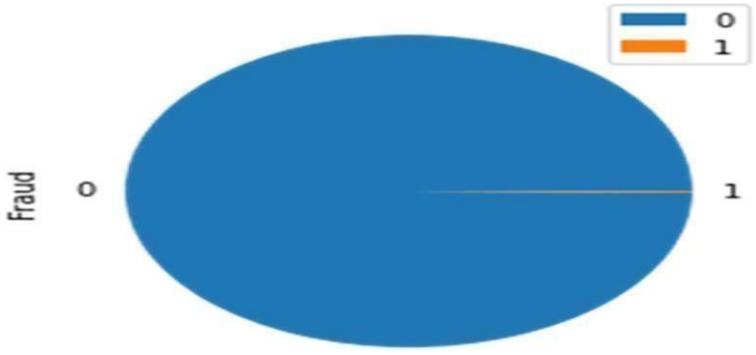


Fig 8.1 Dataset analysis

SUPPORT VECTOR MACHINE

Accuracy: 0.9994557775359011

Precision: 0.6781609195402298

Recall: 0.9516129032258065

F1-Score: 0.7919463087248322

AUC score: 0.975560405918703

	precision	recall	f1-score	support
Normal	1.00	1.00	1.00	56900
Fraud	0.68	0.95	0.79	62
accuracy			1.00	56962
macro avg	0.84	0.98	0.90	56962
weighted avg	1.00	1.00	1.00	56962

RANDOM FOREST

Accuracy: 0.9995611109160493

Precision: 0.7701149425287356

Recall: 0.9305555555555556

F1-Score: 0.8427672955974842

AUC score: 0.9651019999609383

	precision	recall	f1-score	support
Normal	1.00	1.00	1.00	56890
Fraud	0.77	0.93	0.84	72
accuracy			1.00	56962
macro avg	0.89	0.97	0.92	56962
weighted avg	1.00	1.00	1.00	56962

DECISION TREE

Accuracy: 0.9992802219023208

Precision: 0.7241379310344828

Recall: 0.7875

F1-Score: 0.7544910179640718

AUC score: 0.8935390369536936

precision	recall		f1-score	support
Normal	1.00	1.00	1.00	56882
Fraud 0.72	0.79		0.75	80
accuracy			1.00	56962
macro avg	0.86	0.89	0.88	56962
weighted avg	1.00	1.00	1.00	56962

Chapter 9: Conclusion and future work

Nowadays, in the global computing environment, online payments are important, because online payments use only the credential information from the credit card to fulfill an application and then deduct money. Due to this reason, it is important to find the best solution to detect the maximum number of frauds in online systems.

Accuracy, Error-rate, Sensitivity and Specificity are used to report the performance of the system to detect the fraud in the credit card. In this paper, three machine learning algorithmsare developed to detect the fraud in credit card system. To evaluate the algorithms, 80% of the dataset is used for training and 20% is used for testing and validation. Accuracy, error rate, sensitivity and specificity are used to evaluate for different variables for three algorithms. The accuracy result is shown for SVM; Decision tree and random forest classifier are 99.94, 99.92, and 99.95 respectively. The comparative results show that the Random Forest performs better than the SVM and decision tree techniques.

Future Enhancement

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 asmodules 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.

BIBLIOGRAPHY

[1] B.Meena, I.S.L.Sarwani, S.V.S.S.Lakshmi," Web Service mining and its techniques in Web Mining" IJAEGT,Volume 2,Issue 1 , Page No.385-389.

[2] F. N. Ogwueleka, "Data Mining Application in Credit Card Fraud Detection System", Journal of Engineering Science and Technology, vol. 6, no. 3, pp. 311-322, 2019.

[3] G. Singh, R. Gupta, A. Rastogi, M. D. S. Chandel, A. Riyaz, "A Machine Learning Approach for Detection of Fraud based on SVM", International Journal of Scientific Engineering and Technology, vol. 1, no. 3, pp. 194-198, 2019, ISSN ISSN: 2277-1581.

[4] K. Chaudhary, B. Mallick, "Credit Card Fraud: The study of its impact and detection techniques", International Journal of Computer Science and Network (IJCSN), vol. 1, no. 4, pp. 31-35, 2019, ISSN ISSN: 2277-5420.

[5] M. J. Islam, Q. M. J. Wu, M. Ahmadi, M. A. Sid- Ahmed, "Investigating the Performance of Naive-Bayes Classifiers and KNearestNeighbor Classifiers", IEEE International Conference on Convergence Information Technology, pp. 1541-1546, 2017.

[6] R. Wheeler, S. Aitken, "Multiple algorithms for fraud detection" in Knowledge-Based Systems, Elsevier, vol. 13, no. 2, pp. 93-99, 2018.

[7] S. Patil, H. Somavanshi, J. Gaikwad, A. Deshmane, R. Badgujar, "Credit Card Fraud Detection Using Decision Tree Induction Algorithm", International Journal of Computer Science and Mobile Computing (IJCSMC), vol. 4, no. 4, pp. 92-95, 2020, ISSN ISSN: 2320-088X.

[8] S. Maes, K. Tuyls, B. Vanschoenwinkel, B. Manderick,"Credit card fraud detection using Bayesian and neural networks", Proceedings of the 1st international naiso congress on neuro fuzzy technologies, pp. 261-270, 2017.

[9] S. Bhattacharyya, S. Jha, K. Tharakunnel, J. C.Westland, "Data mining for credit card fraud: A comparative study", Decision Support Systems, vol. 50, no. 3, pp. 602-613, 2019.

[10] Y. Sahin, E. Duman, "Detecting credit card fraud by ANN and logistic regression", Innovations in Intelligent Systems and Applications (INISTA) 2018 International Symposium, pp. 315-319, 2018.

APPENDIX

Appendix A: Screen Shots

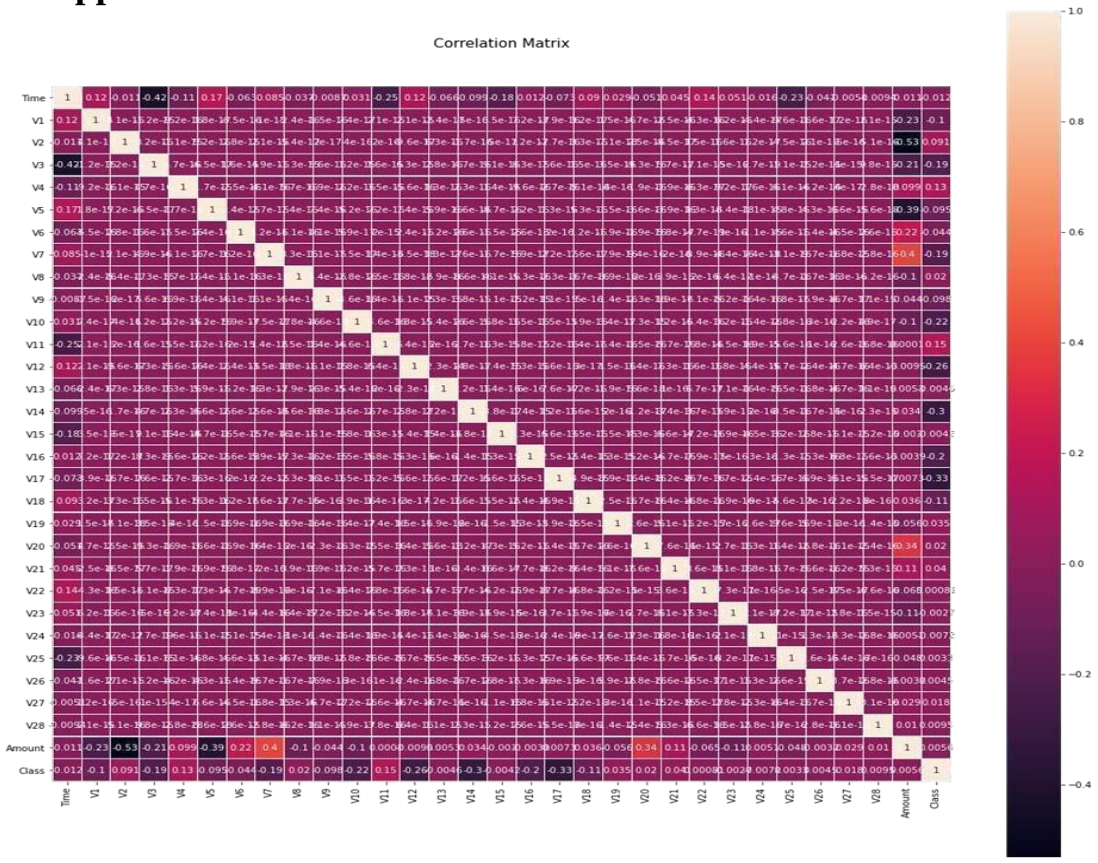


Fig 1 Correlation Matrix

```
In [11]:
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import sklearn
import scipy
import seaborn as sns
from sklearn.metrics import classification_report,accuracy_score
from sklearn.ensemble import IsolationForest
from pylab import rcParams
rcParams['figure.figsize'] = 14,8
RANDOM_SEED = 42
LABELS = ["Normal", "Fraud"]

In [12]:
data = pd.read_csv('creditcard.csv', sep=',')
data.head()
```

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V
0	0.0	-1.359807	-0.072781	2.536347	1.378155	-0.338321	0.462388	0.239599	0.098698	0.363787	...	-0.018307	0.277838	-0.1104
1	0.0	1.191857	0.266151	0.166480	0.448154	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...	-0.225775	-0.638672	0.1012
2	1.0	-1.358354	-1.340163	1.773209	0.379780	-0.503198	1.800499	0.791461	0.247676	-1.514654	...	0.247998	0.771679	0.9094
3	1.0	-0.966272	-0.185226	1.792993	-0.863291	-0.010309	1.247203	0.237609	0.377436	-1.387024	...	-0.108300	0.005274	-0.1903
4	2.0	-1.158233	0.877737	1.548718	0.403034	-0.407193	0.095921	0.592941	-0.270533	0.817739	...	-0.009431	0.798278	-0.1374

5 rows × 31 columns

Fig 2 Dataset

```
df = _
data = df.df

fraud = data[data['Class'] == 1]
valid = data[data['Class'] == 0]
outlierFraction = len(fraud)/float(len(valid))
print(outlierFraction)
fraud.Amount.describe()
```

```
Last executed at 2021-01-21 14:36:13 in 166ms
```

```
0.0017304750013189597
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
```

Fig 3 Data set reading code

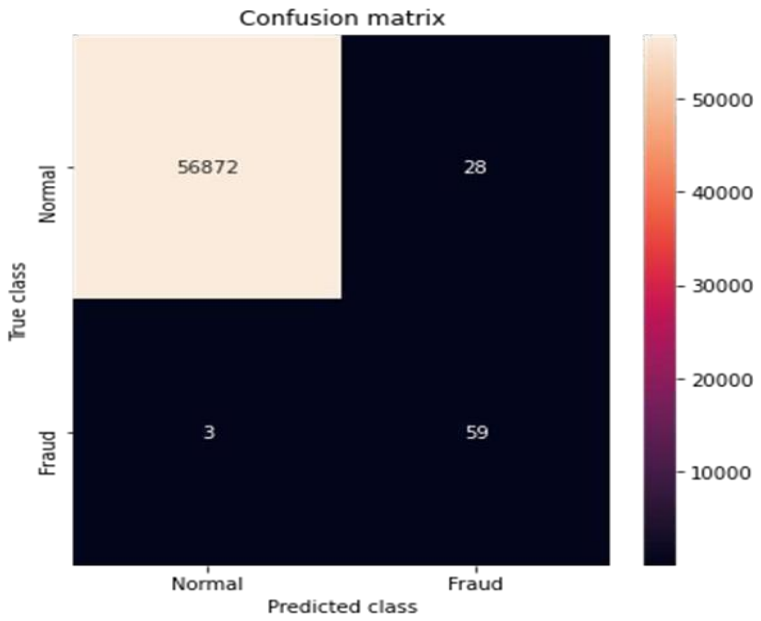


Fig 4 Confusion Matrix

Appendix B: Abbreviations

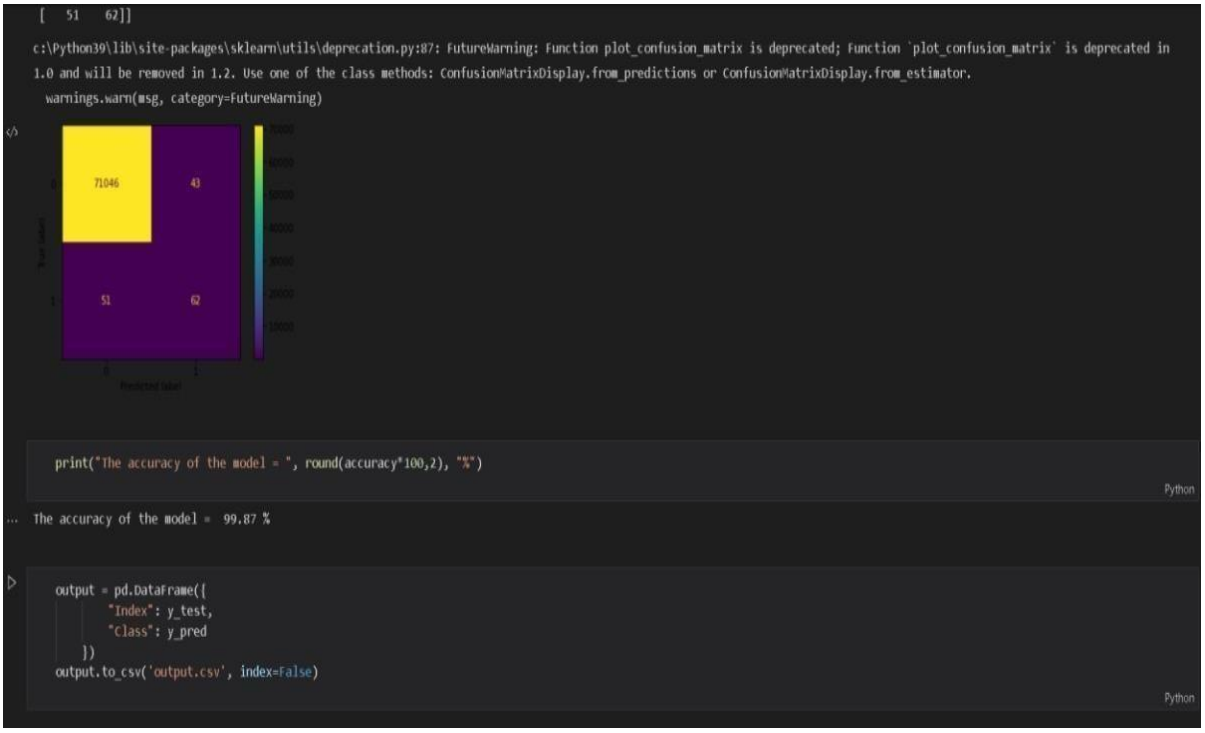
CCFD – Credit Card Fraud Detection

CSDT – Cost Sensitive Decision Tree

ML – Machine Learning

SVM – Support Vector Machine

URL – Uniform Resource



REFERENCES

1. Maniraj SP, Saini A, Ahmed S, Sarkar D. Credit card fraud detection using machine learning and data science. Int J Eng Res 2019; 8(09).
2. Dornadula VN, Geetha S. Credit card fraud detection using machine learning algorithms. Proc Comput Sci. 2019;165:631–41.
3. Thennakoon, Anuruddha, et al. Real-time credit card fraud detection using machine learning. In: 2019 9th international conference on cloud computing, data science & engineering (Confluence). IEEE; 2019.
4. Robles-Velasco A, Cortés P, Muñuzuri J, Onieva L. Prediction of pipe failures in water supply networks using logistic regression and support vector classification. Reliab Eng Syst Saf. 2020;196:106754.
5. Liang J, Qin Z, Xiao S, Ou L, Lin X. Efficient and secure decision tree classification for cloud-assisted online diagnosis services. IEEE Trans Dependable Secure Comput. 2019;18(4):1632–44.
6. Ghiasi MM, Zendehboudi S. Application of decision tree-based ensemble learning in the classification of breast cancer. Comput in Biology and Medicine. 2021;128:104089.
7. Lingjun H, Levine RA, Fan J, Beemer J, Stronach J. Random forest as a predictive analytics alternative to regression in institutional research. Pract Assess Res Eval. 2020;23(1):1.
8. Breiman L. Random forests. Mach Learn. 2001;45(1):5–32.
9. Ning B, Junwei W, Feng H. Spam message classification based on the Naive Bayes classification algorithm. IAENG Int J Comput Sci. 2019;46(1):46–53.
10. Katare D, El-Sharkawy M. Embedded system enabled vehicle collision detection: an ANN classifier. In: 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC); 2019. p. 0284–0289.

11. Campus K. Credit card fraud detection using machine learning models and collating machine learning models. Int J Pure Appl Math. 2018;118(20):825–38.
12. Varmedja D, Karanovic M, Sladojevic S, Arsenovic M, Anderla A. Credit card fraud detection-machine learning methods. In: 18th international symposium INFOTEHJAHORINA (INFOTEH); 2019. p. 1-5.
13. Khatri S, Arora A, Agrawal AP. Supervised machine learning algorithms for credit card fraud detection: a comparison. In: 10th international conference on cloud computing, data science & engineering (Confluence); 2020. p. 680-683.
14. Awoyemi JO, Adetunmbi AO, Oluwadare SA. Credit card fraud detection using machine learning techniques: a comparative analysis. In: International conference on computer networks and Information (ICCNI); 2017. p. 1-9.
15. Seera M, Lim CP, Kumar A, Dhamotharan L, Tan KH. An intelligent payment card fraud detection system. Ann Oper Res 2021;1–23.
16. Guo S, Liu Y, Chen R, Sun X, Wang X. X, Improved SMOTE algorithm to deal with imbalanced activity classes in smart homes. Neural Process Lett. 2019;50(2):1503–26.
17. The Credit card fraud [Online]. https://www.kaggle.com/mlg-ulb/creditcardfraud
18. Kasongo SM. An advanced intrusion detection system for IIoT based on GA and tree based algorithms. IEEE Access. 2021;9:113199–212.
19. Mienye ID, Sun Y. Improved heart disease prediction using particle swarm optimization based stacked sparse autoencoder. Electronics. 2021;10(19):2347.
20. Hemavathi D, Srimathi H. Effective feature selection technique in an integrated environment using enhanced principal component analysis. J Ambient Intell Hum Comput. 2021;12(3):3679–88.
21. Pouramirarsalani A, Khalilian M, Nikravanshalmani A. Fraud detection in E-banking by using the hybrid feature selection and evolutionary algorithms. Int J Comput Sci Netw Secur. 2017;17(8):271–9.

22. Saheed YK, Hambali MA, Arowolo MO, Olasupo YA. Application of GA feature selection on Naive Bayes, random forest and SVM for credit card fraud detection. In: 2020 international conference on decision aid sciences and application (DASA); 2020. p. 1091– 1097.

23. Davis L. Handbook of genetic algorithms; 1991.

24. Li Y, Jia M, Han X, Bai XS. Towards a comprehensive optimization of engine efficiency and emissions by coupling artificial neural network (ANN) with genetic algorithm (GA). Energy. 2021;225:120331.

25. Khalilia M, Chakraborty S, Popescu M. Predicting disease risks from highly imbalanced data using random forest. BMC Med Inf Decis Mak. 2011;11(1):1–13.

26. Abhishek L. Optical character recognition using ensemble of SVM, MLP and extra trees classifier. In: International conference for emerging technology (INCET) IEEE; 2020. p. 1–

4.

27. Chen T, He T, Benesty M, Khotilovich V, Tang Y, Cho H. Xgboost: extreme gradient boosting. R package version 04-2. 2015;1(4):1–4.

28. Harik GR, Lobo FG, Goldberg DE. The compact genetic algorithm. IEEE Trans Evol Comput. 1999;3(4):287–97.

29. Jain A, Nandakumar K, Ross A. Score normalization in multimodal biometric systems. Pattern Recognit. 2005;38(12):2270–85.

30. Kasongo SM, Sun Y. A deep long short-term memory based classifier for wireless intrusion detection system. ICT Express. 2020;6(2):98–103.

31. Norton M, Uryasev S. Maximization of auc and buffered auc in binary classification. Math Program. 2019;174(1):575–612.

33.Google Colab [Online]. Available: https://colab.research.google.com/

Scikit-learn : machine learning in Python [Online].

Altman ER. Synthesizing credit card transactions. 2019. arXiv preprint