CREDIT CARD FRAUD DETECTION

Major project report submitted
in partial fulfillment of the requirement for award of the degree of

Bachelor of Technology
in
Computer Science & Engineering

By

P.SAI SESHA VARMA	(19UECS0780)	**(13297)**
K.LAHARI SAI SREE	(19UECS0520)	**(13085)**
D.SAI KRUPA	(19UECS0252)	**(15769)**

Under the guidance of
Dr.M.GURU VIMAL KUMAR, B.Tech,M.E,Ph.D..,
ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
SCHOOL OF COMPUTING

VEL TECH RANGARAJAN DR. SAGUNTHALA R&D INSTITUTE OF SCIENCE & TECHNOLOGY

(Deemed to be University Estd u/s 3 of UGC Act, 1956)
Accredited by NAAC with A++ Grade
CHENNAI 600 062, TAMILNADU, INDIA

April, 2023

CREDIT CARD FRAUD DETECTION

Major project report submitted
in partial fulfillment of the requirement for award of the degree of

Bachelor of Technology
in
Computer Science & Engineering

By

P.SAI SESHA VARMA	(19UECS0780)	**(13297)**
K.LAHARI SAI SREE	(19UECS0520)	**(13085)**
D.SAI KRUPA	(19UECS0252)	**(15769)**

Under the guidance of
Dr.M.GURU VIMAL KUMAR, B.Tech,M.E,Ph.D.,
ASSISTANT PROFESSOR

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
SCHOOL OF COMPUTING

VEL TECH RANGARAJAN DR. SAGUNTHALA R&D INSTITUTE OF SCIENCE & TECHNOLOGY

(Deemed to be University Estd u/s 3 of UGC Act, 1956)
Accredited by NAAC with A++ Grade
CHENNAI 600 062, TAMILNADU, INDIA

April, 2023

CERTIFICATE

It is certified that the work contained in the project report titled ”CREDIT CARD FRAUD DETECTION” by ”P.SAI SESHA VARMA (19UECS0780), K.LAHARI SAI SREE (19UECS0520), D.SAI KRUPA (19UECS0252)” has been carried out under my supervision and that this work has not been submitted elsewhere for a degree.

Signature of Supervisor
Dr.M.GURU VIMAL KUMAR
ASSISTANT PROFESSOR
Computer Science & Engineering
School of Computing
Vel Tech Rangarajan Dr.Sagunthala R&D
Institute of Science & Technology
April, 2023

Signature of Head of the Department
Computer Science & Engineering
School of Computing
Vel Tech Rangarajan Dr. Sagunthala R&D
Institute of Science & Technology
April, 2023

Signature of the Dean
Dr. V. Srinivasa Rao
Professor & Dean
Computer Science & Engineering
School of Computing
Vel Tech Rangarajan Dr. Sagunthala R&D
Institute of Science & Technology
April, 2023

DECLARATION

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

(Signature)

P.SAI SESHA VARMA

Date: / /

(Signature)

K.LAHARI SAI SREE

Date: / /

(Signature)

D.SAI KRUPA

Date: / /

APPROVAL SHEET

This project report entitled ”CREDIT CARD FRAUD DETECTION” by ”P.SAI SESHA VARMA (19UECS0780), K.LAHARI SAI SREE (19UECS0520), D.SAI KRUPA (19UECS0252)” is approved for the degree of B.Tech in Computer Science & Engineering.

Examiners **Supervisor**

Dr.M.GURU VIMAL KUMAR, B.TECH,M.E,Ph.D.,

Date: / /

Place:

ACKNOWLEDGEMENT

We express our deepest gratitude to our respected **Founder Chancellor and President Col. Prof. Dr. R. RANGARAJAN B.E. (EEE), B.E. (MECH), M.S (AUTO),D.Sc., Foundress President Dr. R. SAGUNTHALA RANGARAJAN M.B.B.S.** Chairperson Managing Trustee and Vice President.

We are very much grateful to our beloved **Vice Chancellor Prof. S. SALIVAHANAN,** for providing us with an environment to complete our project successfully.

We record indebtedness to our **Professor & Dean, Department of Computer Science & Engineering, School of Computing, Dr. V. SRINIVASA RAO, M.Tech., Ph.D.,** for immense care and encouragement towards us throughout the course of this project.

We are thankful to our **Head, Department of Computer Science & Engineering,Dr.M.S. MURALI DHAR, M.E., Ph.D.,** for providing immense support in all our endeavors.

We also take this opportunity to express a deep sense of gratitude to our Internal Supervisor **Dr.M.GURU VIMAL KUMAR, B.TECH,M.E,Ph.D.,** for his cordial support, valuable information and guidance, he helped us in completing this project through various stages.

A special thanks to our **Project Coordinators Mr. V. ASHOK KUMAR, M.Tech., Ms. C. SHYAMALA KUMARI, M.E.,** for their valuable guidance and support throughout the course of the project.

We thank our department faculty, supporting staff and friends for their help and guidance to complete this project.

P.SAI SESHA VARMA (19UECS0780)
K.LAHARI SAI SREE (19UECS0520)
D.SAI KRUPA (19UECS0252)

ABSTRACT

Falsification of the credit card can be defined as the unapproved use of a customer's card data to create purchases or to dismiss funds from the cardholder's record. The misconduct extortion starts from the credit card when somebody incorrectly acquires the number printed on card or the essential records for the card to be operated. The owner of the card, the agent by whom card is issued and even guarantor of a card might not be informed of the fraud until the record is used to create purchases. As shopping through internet-based applications and paying bills online has been come into practice, there is no longer requirement of a physical card to create purchases. Frauds can be categorized in three ways: financial frauds, communication frauds and online marketing frauds. Credit card frauds come under financial frauds. These frauds must be prevented and detected in time. In this direction, many researches are carried out by various researchers to devise the effective and efficient technique. Hackers and intruders are trying different new approaches to breach the security. Therefore, there should always be a safety alert against such frauds. Several achine learning based algorithms have been proposed in this direction. A learning based technique is proposed for detecting the credit card frauds.The online shopping growing day to day. Credit cards are used for purchasing goods and services with the help of virtual card and physical card where as virtual card for online transaction and physical card for offline transaction. In a physical-card based purchase, the cardholder presents his card physically to a merchant for making a payment. To carry out fraudulent transactions in this kind of purchase, an attacker has to steal the credit card. If the cardholder does not realize the loss of card, it can lead to a substantial financial loss to the credit card company. In online payment mode, attackers need only little information for doing fraudulent transaction (secure code, card number, expiration date etc.). In this purchase method, mainly transactions will be done through Internet or telephone. To commit fraud in these types of purchases, a fraudster simply needs to know the card details.

Keywords:LOGISTIC REGRESSION,FRAUD DETECTION,K-MEANS CLUSTERING,DATASET.

LIST OF FIGURES

Chapter 1

INTRODUCTION

1.1 Introduction

Credit card fraud stands as major problem for word wide financial institutions. Annual lost due to it scales to billions of dollars. We can observe this from many financial reports. Such as 10th annual online fraud report by Cyber Source shows that estimated loss due to online fraud is in billions for 2008 which is eleven percent increase than other billions loss in 2007 and in 2006, fraud in United Kingdom alone was estimated to be £535 million in 2007 and now costing around 13.9 billion a year.From 2006 to 2008, UK alone has lost £427.0 million to £609.90 million due to credit and debit card fraud (Woolsey Schulz, 2011). Although, there is some decrease in such losses after implementation of detection and prevention systems by government and bank, card-not-present fraud losses are increasing at higher rate due to online transactions. Worst thing is it is still increasing un-protective and un-detective way.To commit fraud in these types of purchases, a fraudster simply needs to know the card details. Most of the time, the genuine cardholder is not aware that someone else has seen or stolen his card information. The only way to detect this kind of fraud is to analyse the spending patterns on every card and to figure out any inconsistency with respect to the "usual" spending patterns. Fraud detection based on the analysis of existing purchase data of cardholder is a promising way to reduce the rate of successful credit card frauds. Since humans tend to exhibit specific behavioristic profiles, every cardholder can be represented by a set of patterns containing information about the typical purchase category, the time since the last purchase, the amount of money spent, etc. Deviation from such patterns is a potential threat to the system. In the existing system we try to use primitive classification algorithms and clustering algorithms in order to find out the fraud present in credit card transaction. Initially they used k-means Algorithm, in which this will have only 2 features to train the model. By using that k-means they try to cluster the dataset into 2 clusters : 0 being non-fraud and 1 as Fraud parameters, but they cant able to classify clearly all the fields.

1.2 Aim of the project

The aim of the project is to identify the type of credit card fraud through an external interference and analyzing the situation using logistic regression model in order to classify the type of credit card transaction happened.

1.3 Project Domain

Machine learning is a large field of study that overlaps with and inherits ideas from many related fields such as artificial intelligence.The focus of the field is learning, that is, acquiring skills or knowledge from experience.As such, there are many different types of learning that you may encounter as a practitioner in the field of machine learning: from whole fields of study to specific techniques.

1. Supervised Learning
2. Unsupervised Learning
3. Reinforcement Learning

Logistic regression

The Logistic Regression (LR) classifier, sometimes referred to as the Logit classifier, is a supervised ML method that is generally used for binary classification tasks [6]. LR is a special type of linear regression whereby a linear function is fed to the logit function. (y=0+1X1+2X2++nXn q=11+ey) where the value of q will be between 0 and 1. q is the probability that determines the prediction of a given class. The closer q is to 1, the more accurately it predicts a particular class.

1.4 Scope of the Project

In the existing system we try to use primitive classification algorithms and clustering algorithms in order to find out the fraud present in credit card transaction. Initially they used k-means Algorithm, in which this will have only 2 features to train the model. By using that k-means they try to cluster the dataset into 2 clusters : 0 being non-fraud and 1 as Fraud parameters, but they cant able to classify clearly all the fields. If there are less dimensions the k-means can easily able to cluster the data and find the fraud activities but if the same dataset contains more dimensions this may not generate the accurate results.

Chapter 2

LITERATURE REVIEW

S.P.Maniraj, Aditya Saini, Swarna Deep Sarkar developed a Credit Card Fraud Detection using Machine Learning Data Science Methodology. This paper defines "fraud" in credit card transactions as the unauthorised un welcomed use of a credit card account by someone other than the account owner. The abuse can be stopped with the use of necessary preventative measures, the behaviour of such fraudulent acts can be researched to lessen it and safeguard against recurrence. In other terms, credit card fraud is the use of another person's credit card for personal gain when neither the cardholder nor the organisation responsible for providing the card are aware that the card is being used. Monitoring user populations behaviour is a key component of fraud detection since it helps identify,detect,prevent undesirable behaviours including fraud, intrusion, defaulting. This is a really pertinent issue that has to be addressed by communities like machine learning data science, where an automated solution is possible. Due to its varied characteristics, including class imbalance, this issue is particularly difficult to solve from the standpoint of learning. There are significantly more legitimate transactions than fraudulent ones. Additionally, the statistical characteristics of the transaction patterns frequently vary over time. Unquestionably, using a credit card fraudulently is a criminal act of dishonesty. The most popular fraud schemes, as well as how to spot them, are listed in this article, which also reviews recent research in the area. Along with the method, pseudocode, explanation of its implementation, experimentation findings, this work has also provided a detailed explanation of how machine learning might be used to improve fraud detection.

Aisha Mohammad Fayyomi, Derar Eleyan Amina Eleyan published an International Journal of Scientific Technology Research introducing A Survey on Credit Card Fraud Detection Techniques. Their poll indicates that credit card fraud has grown to be a major global threat. Globally, fraud causes enormous financial losses. This pushed credit card firms to spend money developing creating methods to detect lessen fraud. The main objective of this study is to establish acceptable algorithms

that credit card issuers can use to more quickly cheaply identify fraudulent transactions. The survey compares various machine learning methods, such as Nearest Neighbours, Decision Trees, Random Forest, Artificial Neural Networks, Logistic Regression, K-means clustering. A scenario-based algorithm can be used to choose the scenario that is the greatest fit for a given scenario because no 2 scenarios are the same. In this survey article,all these fraud detection methods are covered. Emmanuel Ileberi, Yanxia Sun Zenghui Wang introduced A Machine Learning Based Credit Card Fraud Detection Using the GA Algorithm for Feature Selection [4]. Using this strategy,it was discovered that the Internet has grown exponentially over the past 10 years. As a result, services like e-commerce, tap-and-pay systems,online bill payment systems, etc. have proliferated become more widely used. As a result, credit card thieves are now more active than ever in their attacks on transactions. Credit card data encryption tokenization are just 2 of the safeguards in place to protect credit card transactions. Although most of the time these techniques work, they don't completely safeguard credit card transactions from fraud.

John Richard D. Kho Larry A. Vea developed a Credit Card Fraud Detection Based on Transaction Behavior. According to this concept,credit card fraud occurs when someone uses another person's credit card for their own gain, often in complete secrecy or anonymity, even the issuing banks are unaware that the card is being used. In addition, the offender has no connection to the cardholder or issuer has no intention of telling the owner of the card about the missing card or making good on the transactions done. The top 5 methods used by fraudsters in cases involving credit cards over the past 2 decades are as follows: Counterfeit credit cards,Lost or stolen,No-card fraud (e.g., giving card information to non-legitimate telemarketer),Stolen cards during mailing fraud,Identity-theft fraud. These fraud cases account for 81 of all recognised fraud categories in the credit card sector. Banks businesses are still a target of these attacks even if they may seem extremely common to them. The current state of the credit card industry with regard to fraud issues has been described in this report. Although there are new technologies that can be used to decrease or even eliminate the effects of credit card fraud, banks merchants throughout the world are starting to question its conception application. This study advises creating a model based on cardholder spending patterns utilising it to spot unusual transactions. Due to a non-disclosure agreement (NDA) between the participating bank the proponent, this study [5] didn't go into depth about the built model.

But this study was able to demonstrate the methods procedures used to create the model. The author of this paper expects that some banks or people may use it as a guide when implementing fraud detection systems in the financial industry in the near future. Benefits of putting in place such a detection system include lowering the expenses borne by banks for phone SMS service; instead of sending SMS transaction notifications to all clients, message will be sent to those with detected anomalous transaction. The Random Tree outperformed J48 in the evaluation of classifiers conducted during the model's development. J48 produced a tight constraint with respect to its variance in accuracy values, according to further examination of the 2 classifiers that involved introducing randomness into the dataset.

Dejan Varmedja, Mirjana Karanovic, Srdjan Sladojevic, Marko Arsenovic, and Andras Anderla.They made a survey on credit card fraud detection, considering the major areas of credit card fraud detection that are bank fraud, corporate fraud, Insurance fraud. With these they have focused on the two ways of credit card transactions Virtually (card, not present) ii) With Card or physically present. They had focused on the techniques which are Regression, classification, Logistic regression, Support vector machine, Neural network, Artificial Immune system, K-nearest Neighbor, Naïve Bayes, Genetic Algorithm, Data mining, Decision Tree, Fuzzy logic-based system, etc. In which, they have explained six data mining approaches as theoretical background that are classification, clustering, prediction, outlier detection, Regression, and visualization. Then have explained about existing techniques based on statistical and computation which is Artificial Immune system (AIS), Bayesian Belief Network, Neural Network, Logistic Regression, Support Vector Machine, Tree, Selforganizing map,Hybrid Methods, As a result, they had concluded that all the present machine learning techniques mentioned above can provide high accuracy for the detection rate and industries are looking forward to finding new methods to increase their profit and reduce the cost. Machine learning can be a good choice for it.

Chapter 3

PROJECT DESCRIPTION

3.1 Existing System

In the existing system there are lot of machine learning approaches implemented for detecting credit card fraud detection which are implemented based on AE, IF, LOF and K- Means which are giving 70,73 and 71 accuracy respectively. In this project we are trying to improve the accuracy by refining the data in a better way for efficient detection of credit card fraud. . By using that k-means they try to cluster the dataset into 2 clusters : 0 being non-fraud and 1 as Fraud parameters, but they cant able to classify clearly all the fields. If there are less dimensions the k-means can easily able to cluster the data and find the fraud activities but if the same dataset contains more dimensions this may not generate the accurate results.

3.2 Proposed System

Here, the unsupervised learning approaches are used for fraud detections. In this project we collected credit card fraud data from kaggle below I am specifying description regarding the data set which was taken from kaggle directly. Our goal is to implement machine learning model in order to classify, to the highest possible degree of accuracy, credit card fraud from a dataset gathered from Kaggle. After initial data exploration, we knew we would implement a logistic regression model for best accuracy reports. For that we try to use Gaussian Mixture, Isolation Forest and K-Means algorithm and classify each and every record which is present in that dataset. As per the observation we can see Gaussian Mixture gives best accuracy.

3.3 Feasibility Study

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system

analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential. Three key considerations involved in the feasibility analysis are

3.4 Economic Feasibility

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

3.5 Technical Feasibility

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

3.6 Social Feasibility

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

3.7 System Specification

3.8 Functional Requirement

In software engineering, a functional requirement defines a function of a software system or its component. A function is described as a set of inputs, the behavior, and outputs (see also software). Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish. Behavioral requirements describing all the cases where the system uses the functional requirements are captured in use cases. Generally, functional requirements are expressed in the form "system shall do requirement". The plan for implementing functional requirements is detailed in the system design. In requirements engineering, functional requirements specify particular results of a system. Functional requirements drive the application architecture of a system. A requirements analyst generates use cases after gathering and validating a set of functional requirements. The hierarchy of functional requirements is: user/stakeholder,request feature,use case,business rule. Functional requirements drive the application architecture of a system. A requirements analyst generates use cases after gathering and validating a set of functional requirements. Functional requirements may be technical details, data manipulation and other specific functionality of the project is to provide the information to the user. The following are the functional requirements of our system:

1.We are providing all the information related to credit card transactions to the system.

2.The user can give his own transaction data and can predict if there is any credit card transaction failed or not.

3.We are having multiple algorithms for classification.

4.The proposed application can accurately identify the credit card fraud by analyzing all the parameters present in the transaction.

3.9 Non Functional Requirements

Non-functional requirements define the overall qualities or attributes of the resulting System Non-functional requirements place restrictions on the product being developed, the development process, and specify external constraints that the product

must meet. Examples of NFR include safety, security, usability, reliability and performance Requirements. Project management issues (costs, time, and schedule) are often considered as non-functional requirements.

3.10 Performance requirements

Requirements about resources required, response time, transaction rates, throughput, benchmark specifications or anything else having to do with performance. In this project, the user will try to gather all the information from KAGGLE website and then try to train the system with that dataset.

3.11 Modifiability

Requirements about the effort required to make changes in the software. Often, the measurement is personnel effort (person- months).

3.12 Portability

The effort required to move the software to a different target platform. The measurement is most commonly person-months or modules that need changing.

3.13 Reliability

Requirements about how often the software fails. The measurement is often expressed in MTBF (mean time between failures). The definition of a failure must be clear. Also, don't confuse reliability with availability which is quite a different kind of requirement. Be sure to specify the consequences of software failure, how to protect from failure, a strategy for error detection, and a strategy for correction.

3.14 Security

One or more requirements about protection of your system and its data. The measurement can be expressed in a variety of ways (effort, skill level, time) to break into the system. Do not discuss solutions (e.g. passwords) in a requirements document.

3.15 Usability

Requirements about how difficult it will be to learn and operate the system. The requirements are often expressed in learning time or similar metrics.

3.16 Legal

There may be legal issues involving privacy of information, intellectual property rights, export of restricted technologies, etc.

3.17 Hardware Specification

Processor : Core i7
RAM : 16 GB
Hard Disk : 256 GB SSD/1 TB HDD
Generation : 12th

3.18 Software Specification

Operating system : Windows11(HOME)
Coding Language : Python
Front-End : Google Collab
Dataset : Dataset From Kaggle

3.19 Standards and Policies

Google Colab:- Colaboratory (”Colab” for short) is a data analysis and machine learning tool that allows you to combine executable Python code and rich text along with charts, images, HTML, LaTeX and more into a single document stored in Google Drive**Standard Used: ISO/IEC 27001**

Chapter 4

METHODOLOGY

4.0.1 LOAD DATASET MODULE

In this module we try to load the dataset which is collected from Kaggle website and then try to give that excel file information as input to the next module. Dataset URL: https://www.kaggle.com/mlg-ulb/creditcardfraud To provide privacy to users transaction data kaggles peoples have converted transaction data to numerical format using PCA Algorithm. Below are some example from dataset.

4.0.2 GENERATE TEST AND TRAIN MODULE

Here we try to divide the data into test and train datasets and we used 70: 30 percent ratio for dividing the whole dataset into multiple parts.Here 70 percent data records are used for training the system and 30 percent data is used for testing the model.

4.0.3 RUN SEVERAL ALGORITHMS MODULE

Here we try to run the Several algorithm on the train dataset and try to check the probability of each and every attribute which is present in that record. Once all the records are processed now we try to find out which records are having fraud activity and which are having normal activities. Once we use Kmeans, Gaussian mixture, Isolation forest on training dataset ,we can get accuracy of each and every algorithm and finally we can tell Gaussian mixture is accurate more compared with all algorithms.

4.0.4 DETECT FRAUD MODULE

Here we try to apply several algorithms and check the model on test data. Once the test data is given as input we can see the data can be categorized into 2 categories where how many records are found fraud activity and how many are having normal activities.Here we can identify the fraud from the given set of transactions..

4.0.5 COMPARATIVE ANALYSIS OF ALGORITHMS ACCURACY

In this current application we tested the dataset on Kmeans, Gaussian mixture, Isolation forest on training dataset ,we can get accuracy of each and every algorithm and finally we can tell Gaussian mixture is accurate more compared with all algorithms.

4.1 General Architecture

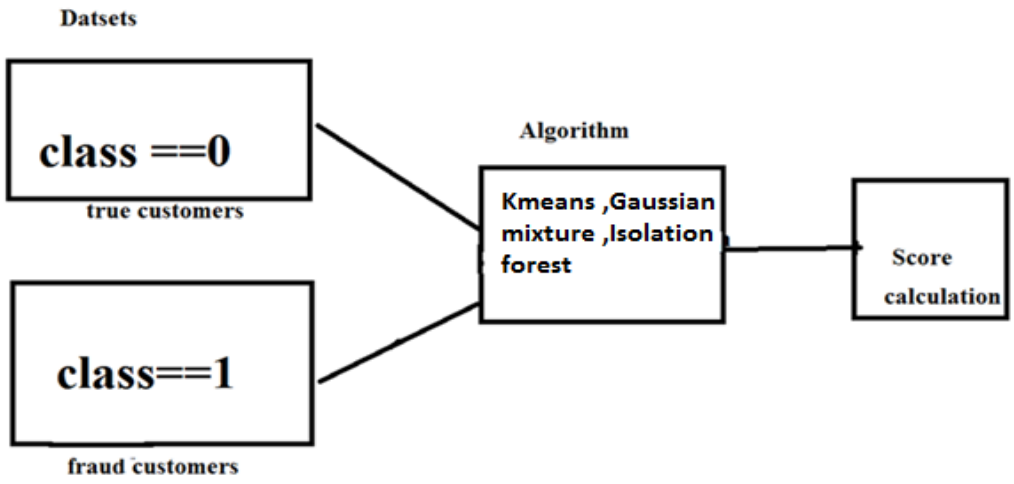

Figure 4.1: **GENERAL ARCHITECTURE FOR CREDIT CARD FRAUD DETECTION**

Description
Implementation is the stage where the theoretical design is converted into programmatically manner. In this stage we will divide the application into a number of modules and then coded for deployment. The front end of the application takes Google Collaboratory and as a Back-End Data base we took UCI Heart Patients Records as dataset. Here we are using Python as Programming Language to Implement the current application.The application is divided mainly into following 5 modules.

4.2 Design Phase

4.2.1 Data Flow Diagram

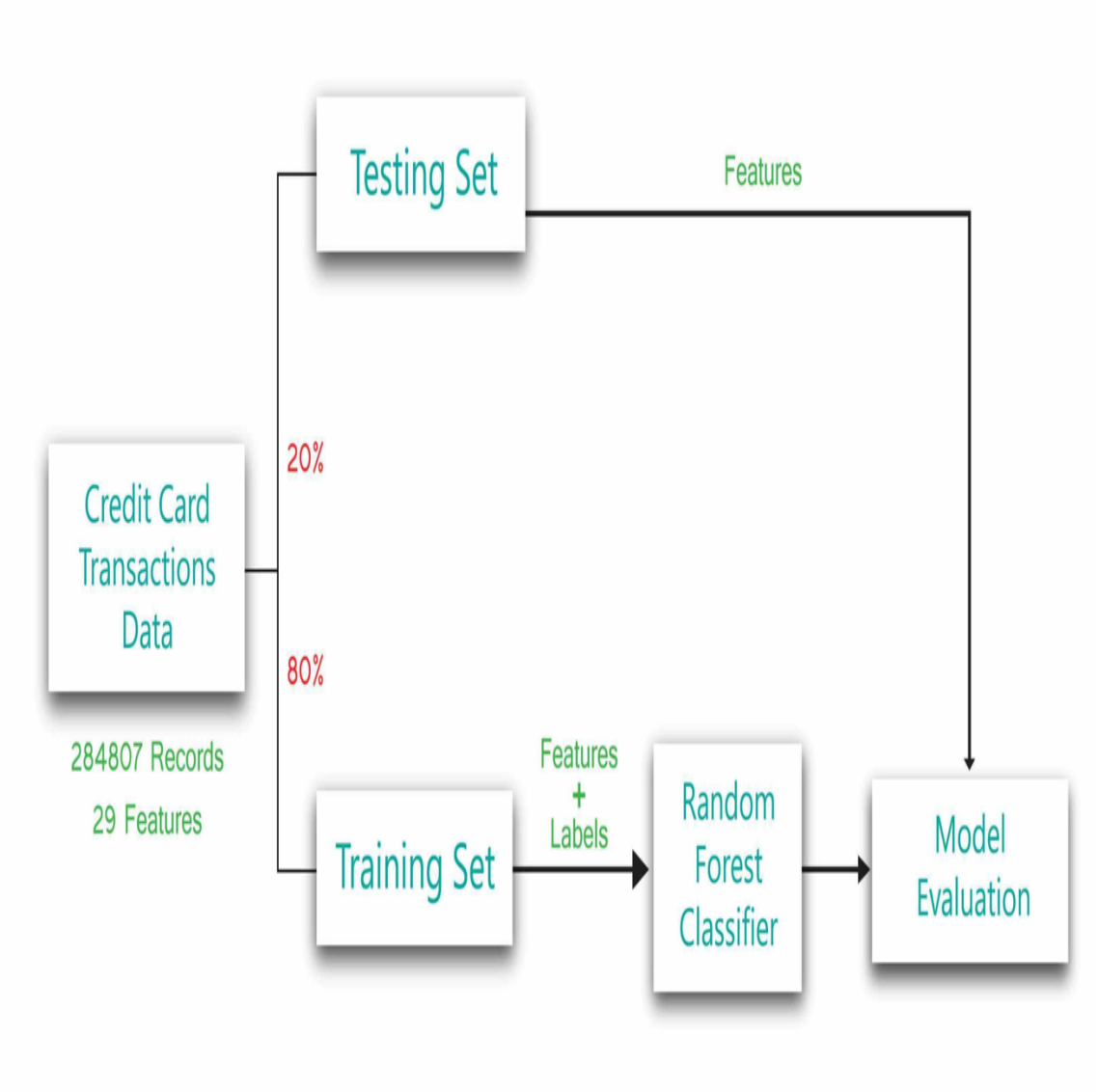

Figure 4.2: **DATA FLOW DIAGRAM TO DEFINE THE PROCESS FOR CREDIT CARD FRAUD**

Description

1.Load Dataset Module

2.Generate Test and Train Data

3.Run Several Algorithms

4.Detect Fraud from Test Dataset

5.Fraud Transaction Detection Graph

6.Comparative Analysis of Algorithms Accuracy

A software system can be said to have two distinct characteristics: a structural,

”static” part and a behavioral, ”dynamic” part. In addition to these two characteristics, an additional characteristic that a software system possesses is related to implementation. Before we categorize UML diagrams into each of these three characteristics, let us take a quick look at exactly what these characteristics are.

4.2.2 Use Case Diagram

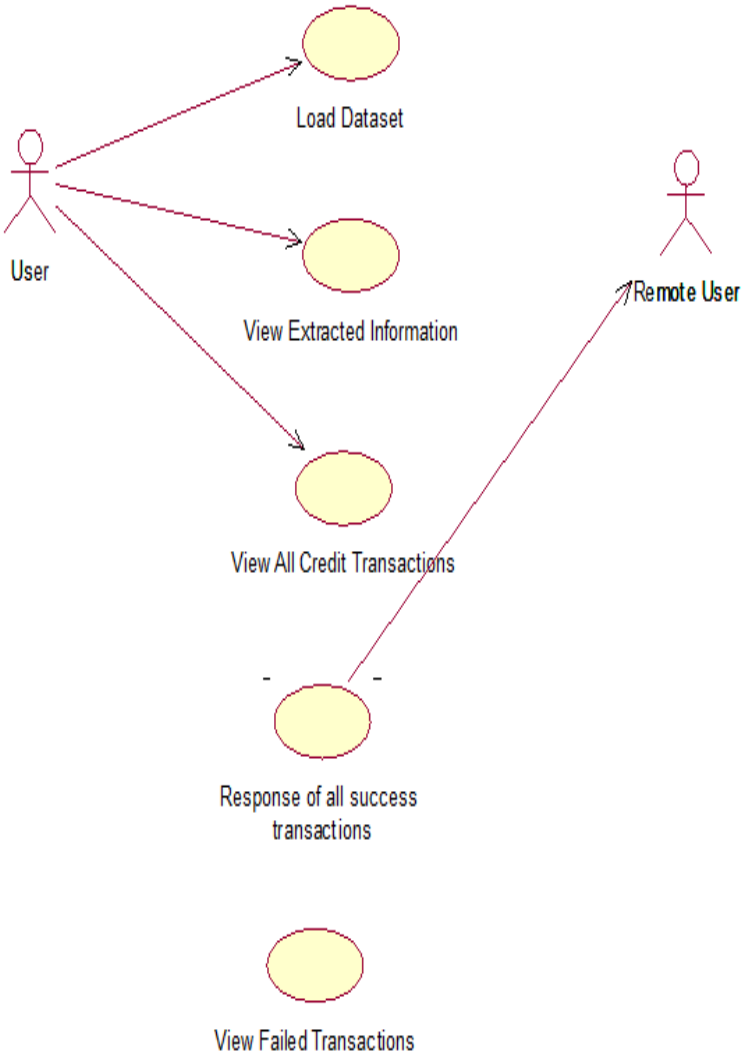

Figure 4.3: **USE CASE DIAGRAM TO DEMONSTRATE THE FRAUDULENT ACTIVITY**

Description

The use case diagram is used to identify the primary elements and processes that form the System. The primary elements are termed as actors and the processes are called use cases. The use case diagram shows which actors interact with each use case.

Use cases:- A use case describes a sequence of actions that provide something of measurable value to an actor and is drawn as a horizontal ellipse.

Actors:- An actor is a person, organization, or external system that plays a role in one or more interactions with the system.

4.2.3 Class Diagram

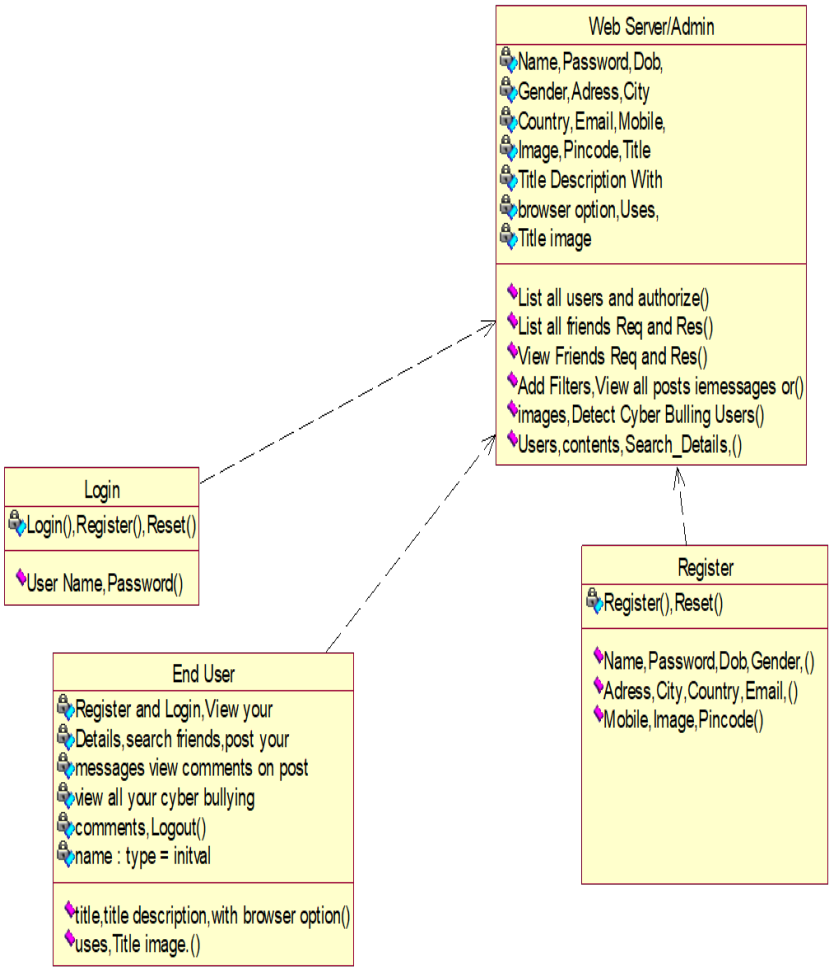

Figure 4.4: **CLASS DIAGRAM TO IDENTIFY THE TYPE OF FRUAD FROM USER**

Description

The class diagram is used to refine the use case diagram and define a detailed design of the System. The class diagram classifies the actors defined in the use case diagram into a set of interrelated classes. The relationship or association between the classes

can be either an ”is-a” Or ”has-a” relationship.

4.2.4 Sequence Diagram

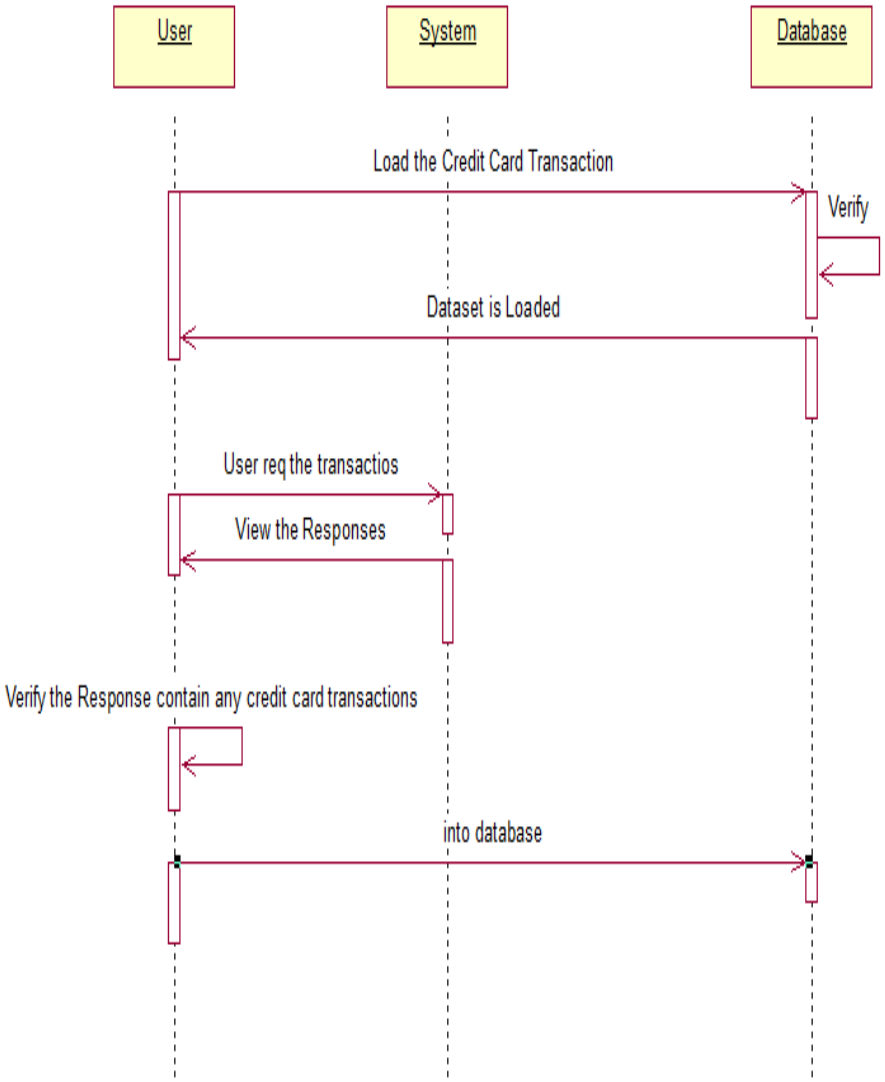

Figure 4.5: **SEQUENCE DIAGRAM TO ESTABLISH THE CONNECTION TO TYPE OF SOURCE AND FRAUD**

Description

A sequence diagram represents the interaction between different objects in the system. The Important aspect of a sequence diagram is that it is time-ordered. Different objects In the sequence diagram interact with each other by passing ”messages”.

4.2.5 Collaboration diagram

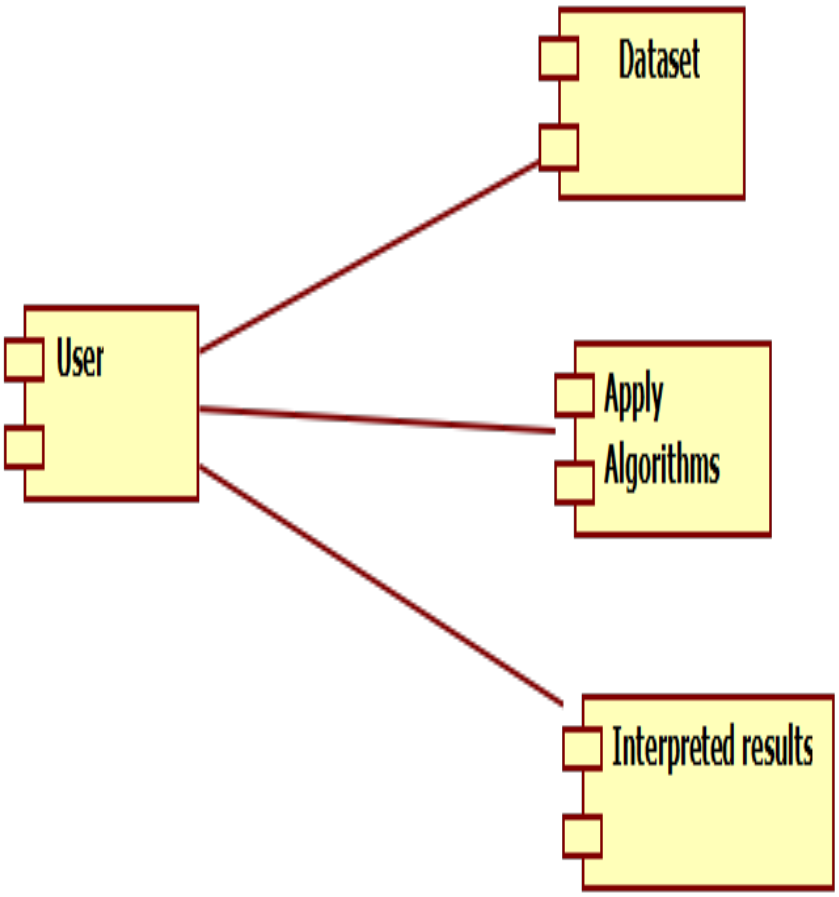

Figure 4.6: **COLLABORATION DIAGRAM TO COMPARE AND IDENTIFY THE TYPE OF FRUADS**

Description

The collaboration diagram captures the configuration of the runtime elements of the Application. This diagram is by far most useful when a system is built and ready to be Deployed. The name Deployment itself describes the purpose of the diagram. Deployment diagrams are used for describing the hardware components where software components are deployed. Component diagrams and deployment diagrams are closely related. The purpose of deployment diagrams can be described as:

1.Visualize hardware topology of a system.

2.Describe the hardware components used to deploy software components.

3.Describe runtime processing nodes.

4.2.6 Activity Diagram

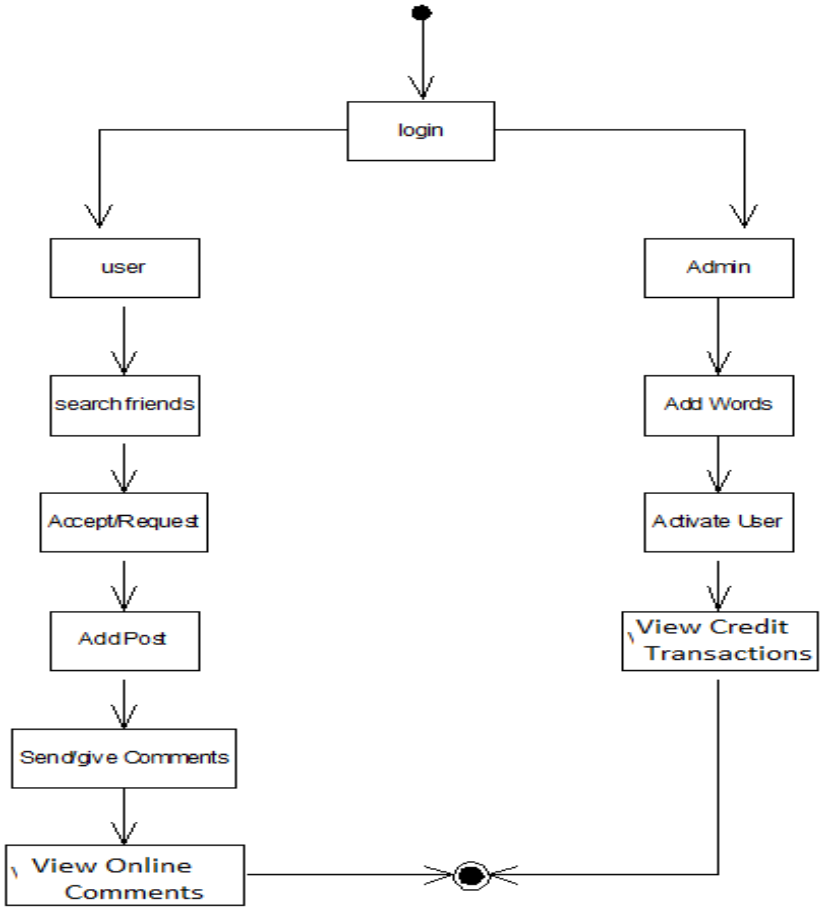

Figure 4.7: **ACTIVITY DIAGRAM TO KNOW WHEN THE FRAUD HAPPENED FROM USER INFORMATION**

Description

The process flows in the system are captured in the activity diagram. Similar to a state Diagram, an activity diagram also consists of activities, actions, transitions, initial and final States, and guard conditions.

4.3 Algorithm & Pseudo Code

4.4 Logistic Regression

1.Logistic regression is a Machine Learning classification algorithm that is used to predict the probability of certain classes based on some dependent variables. In short, the logistic regression model computes a sum of the input features (in most cases, there is a bias term), and calculates the logistic of the result.The output of logistic regression is always between (0, and 1), which is suitable for a binary classification task. The higher the value, the higher the probability that the current sample is classified as class=1, and vice versa.

2.The objective of this project is to create a simple Logistic Regression model capable of detecting fraud in credit card operations, thus seeking to minimize the risk and loss of the business. The biggest challenge is to create a model that is very sensitive to fraud, since most transactions are legitimate, making detection difficult.The input variables are numeric, the result of a PCA transformation. Due to confidentiality issues, the original data and other complementary information were not made available.The only variables that have not been transformed with the PCA are 'Time' and 'Value'. The variable 'Time' contains the seconds between each transaction and the first transaction in the data set. The 'Amount' variable refers to the amount of the transaction.

3.Not all algorithms fit cleanly into this simple dichotomy, though, and logistic regression is a notable example. Logistic regression is part of the regression family as it involves predicting outcomes based on quantitative relationships between variables. However, unlike linear regression, it accepts both continuous and discrete variables as input and its output is qualitative. In addition, it predicts a discrete class such as "Yes/No" or "Customer/Non-customer".

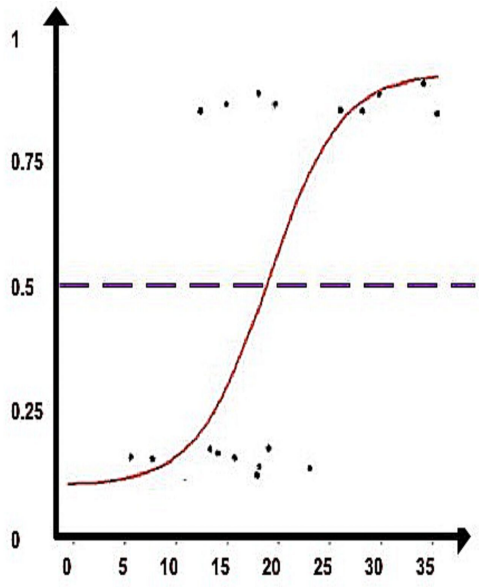

Figure 4.8: **GRAPH REPRESENTING TWO CLASS VARIABLES SEPERATED BASED ON LOGISTIC REGRESSION**

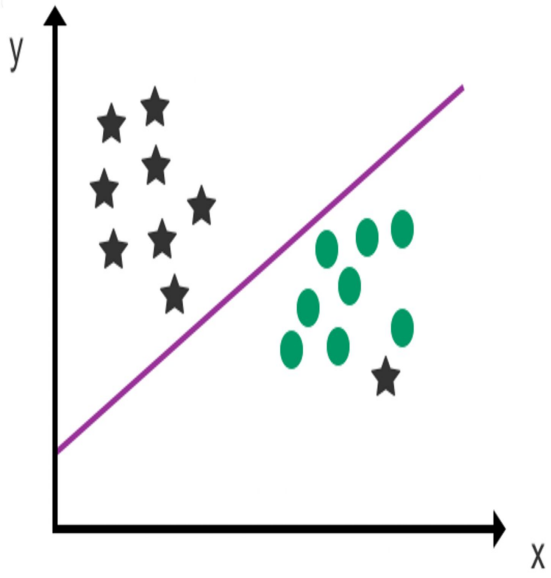

Figure 4.9: **FIGURE REPRESENTING TWO VARIABLES SEPERATED ON REGRESSION TECHNIQUE**

4.The application of methods for data balancing, such as undersampling and oversampling techniques are widely used in these cases. Changing the sampling makes the algorithm more ”sensitive” to fraudulent transactions.

Undersampling is the technique of removing major class records from the sample. In this case, it is necessary to remove random records from the legitimate class (No fraud), in order to obtain a number of records close to the amount of the minority class (fraud) in order to train the model.

Oversampling is exactly the opposite: it means adding minority class records (fraud) to our training sample, thus increasing the overall proportion of fraud records. There are methods to generate samples from the minority class, either by duplicating existing records or artificially generating others

4.5 Pseudo Code

1: **Input:** Training data
2: **Begin**
3: For i = 1 to k
4: For each training data instance d_i.
5: Set the target value for the regression to $z_i = \frac{y_i - P(1|d_j)}{[P(1|d_j)(1-P(1|d_j))]}$
6: Initialize the weight of instance d_j to $[P(1|d_j)(1 - P(1|d_j))]$
7: Finalize a $f(j)$ to the data with class value (Z_j) and weight (w_j)
8: **Classical label decision**
9: Assign (class label: 1) if $P_{id} > 0.5$, otherwise (class label: 2)
10: **End**

4.6 Module Description

4.6.1 FRONT END IMPLEMENTATION

Python is a general purpose, dynamic, high level, and interpreted programming language. It supports Object Oriented programming approach to develop applications. It is simple and easy to learn and provides lots of high-level data structures. Python is easy to learn yet powerful and versatile scripting language, which makes it attractive for Application Development. Python's syntax and dynamic typing with its interpreted nature make it an ideal language for scripting and rapid application development. It supports multiple programming pattern, including object-oriented, imperative, and functional or procedural programming styles.

4.7 Python Applications

Python is known for its general purpose nature that makes it applicable in almost each domain of software development. Python as a whole can be used in any sphere of development. Here, we are specifying applications areas where python can be applied. Web Applications: We can use Python to develop web applications. It provides libraries to handle internet protocols such as HTML and XML, JSON, Email processing, request, beautifulSoup, Feedparser etc. It also provides Frameworks such as Django, Pyramid, Flask etc to design and delelop web based applications. Some important developments are: PythonWikiEngines, Pocoo, PythonBlogSoftware etc. The useful library and package are SciPy, Pandas, IPython etc. SciPy is group of packages of engineering. Desktop GUI Applications: Python provides Tk GUI library to develop user interface in python based application. Some other useful toolkits wxWidgets, Kivy, pyqt that are useable on several platforms. The Kivy is popular for writing multitouch applications.
1.Software Development: Python is helpful for software development process. It works as a support language and can be used for build control and management, testing etc.
2.Scientific and Numeric: Python is popular and widely used in scientific and numeric computing. Some useful library and package are SciPy, Pandas, IPython etc. SciPy is group of packages of engineering, science and mathematics.
3.Business Application: Python is used to build Bussiness applications like ERP and e-commerce systems. Tryton is a high level application platform.
4.Console Based Application: We can use Python to develop console based applications. For example: IPython.Audio or Video based Applications: Python is awesome to perform multiple tasks and can be used to develop multimedia applications. Some of real applications are: TimPlayer, cplay etc.
5.3D CAD Applications: To create CAD application Fandango is a real application which provides full features of CAD.
6.Enterprise Applications: Python can be used to create applications which can be used within an Enterprise or an Organization. Some real time applications are: OpenErp, Tryton, Picalo etc.
7.Applications for Images: Using Python several application can be developed for image. Applications developed are: VPython, Gogh, imgSeek etc. Python's syntax and dynamic typing with its interpreted nature make it an ideal language for script-

ing and rapid application development. It supports multiple programming pattern, including object-oriented, imperative, and functional or procedural programming styles.

4.8 Back End Implementation

4.9 Machine Learning

Machine learning is a subfield of artificial intelligence (AI). The goal of machine learning generally is to understand the structure of data and fit that data into models that can be understood and utilized by people. . In traditional computing, algorithms are sets of explicitly programmed instructions used by computers to calculate or problem solve. Machine learning algorithms instead allow for computers to train on data inputs and use statistical analysis in order to output values that fall within a specific range.
Machine learning is the scientific field dealing with the ways in which machines learn from experience. For many scientists, the term “machine learning” is identical to the term “artificial intelligence”, given that the possibility of learning is the main characteristic of an entity called intelligent in the broadest sense of the word. The purpose of machine learning is the construction of computer systems that can adapt and learn from their experience. There are two types.

1.Supervised Learning In supervised learning, the system must “learn” inductively a function called target function, which is an expression of a model describing the data. The objective function is used to predict the value of a variable, called dependent variable from a set of variables, called independent variables or input variables or characteristics or features. The set of possible input values of the function, i.e. its domain, are called instances. Each case is described by a set of characteristics. A subset of all cases, for which the output variable value is known, is called training data or examples. In order to infer the best target function, the learning system, given a training set, takes into consideration alternative functions, called hypothesis and denoted by h. In supervised learning, there are two kinds of learning tasks: classification and regression.

2.Unsupervised Learning In unsupervised learning, the system tries to discover

the hidden structure of data or associations between variables. In that case, training data consists of instances without any corresponding labels. Association Rule Mining appeared much later than machine learning and is subject to greater influence from the research area of databases.

4.10 K-Means Clustering Algorithm

The k-means clustering method is an unsupervised machine learning technique used to identify clusters of data objects in a dataset. There are many different types of clustering methods, but k-means is one of the oldest and most approachable. These traits make implementing k-means clustering in Python reasonably straightforward, even for novice programmers and data scientists.If you're interested in learning how and when to implement k-means clustering in Python, then this is the right place. You'll walk through an end-to-end example of k-means clustering using Python, from preprocessing the data to evaluating results.

4.11 Gaussian Mixture Models

In the world of Machine Learning, we can distinguish two main areas: Supervised and unsupervised learning. The main difference between both lies in the nature of the data as well as the approaches used to deal with it. Clustering is an unsupervised learning problem where we intend to find clusters of points in our dataset that share some common characteristics. Let's suppose we have a dataset that looks like this:

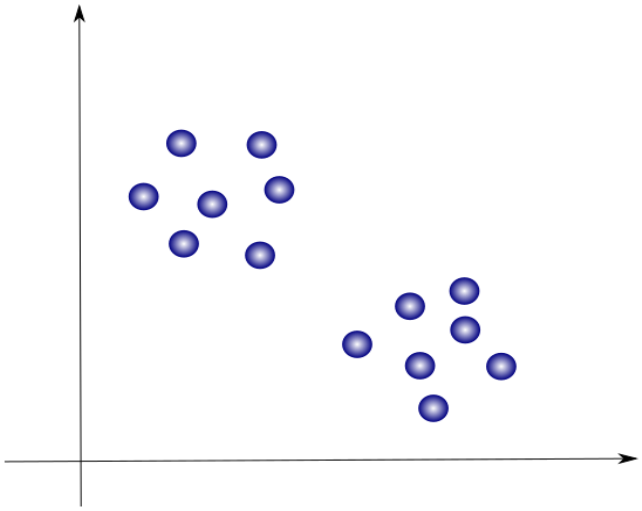

Figure 4.10: **FIGURE TO REPRESENT TWO VARIABLES SEPERATED BY CLASS**

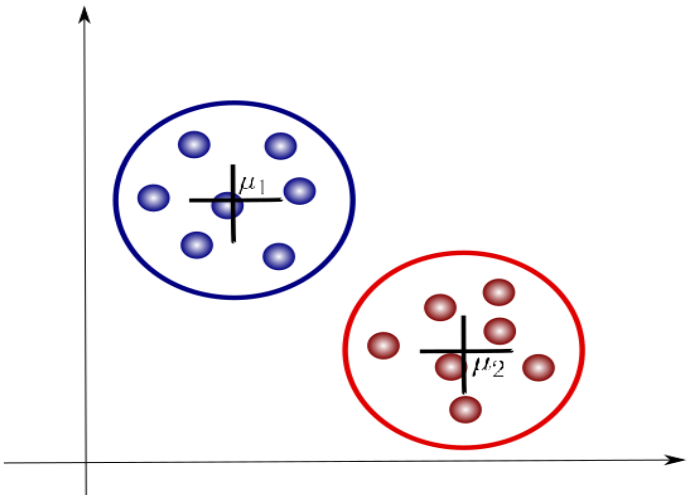

Figure 4.11: **FIGURE REPRESENTING CLASS VARIABLES DEFINED INTO TWO PARTS**

One important characteristic of K-means is that it is a hard clustering method, which means that it will associate each point to one and only one cluster. A limitation to this approach is that there is no uncertainty measure or probability that tells us how much a data point is associated with a specific cluster. So what about using a soft clustering instead of a hard one? This is exactly what Gaussian Mixture Models, or simply GMMs, attempt to do. Let's now discuss this method further.

Chapter 5

IMPLEMENTATION AND TESTING

5.1 Input

5.1.1 Input Design

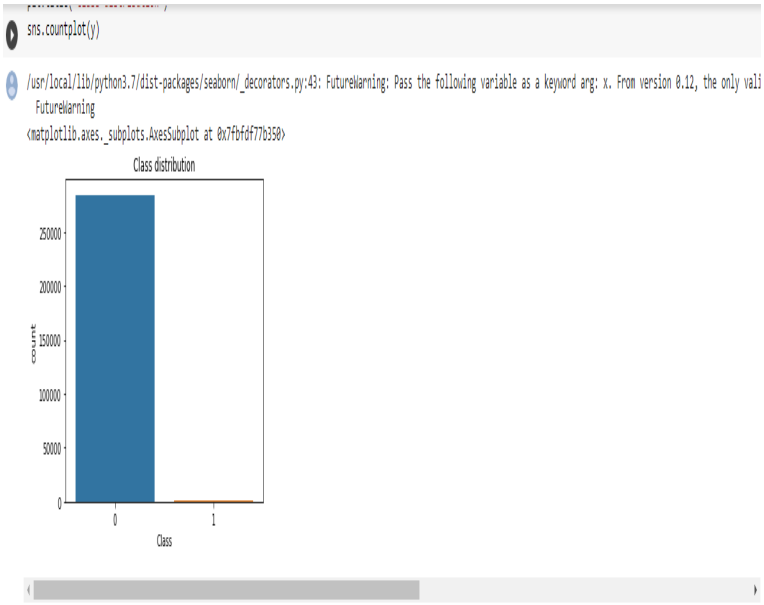

Figure 5.1: **INPUT 1 REPRESENTING THE CODE**

Description:-
The figure represents the code and the execution background(google collab).In this we can identify the code segment used for this project.

5.1.2 Input Design

```
sns.heatmap(df_cm, annot=True, annot_kws={"size": 16},fmt='.1f') # font size

plt.show()
```

```
res={'Gaussian Mixture':[gm_acc],'KMeans':[kmeans_acc],'Isolation Forest':[is_acc]}
results=pd.DataFrame(data=res)
# print(results.head())
sns.barplot(data=results)
plt.title('Comparision of model accuracies')
plt.xlabel('Models')
plt.ylabel('Accuracy')
plt.show()
```

Figure 5.2: **INPUT 2 REPRESENTING THE CODE**

5.2 Output

5.2.1 Output Design

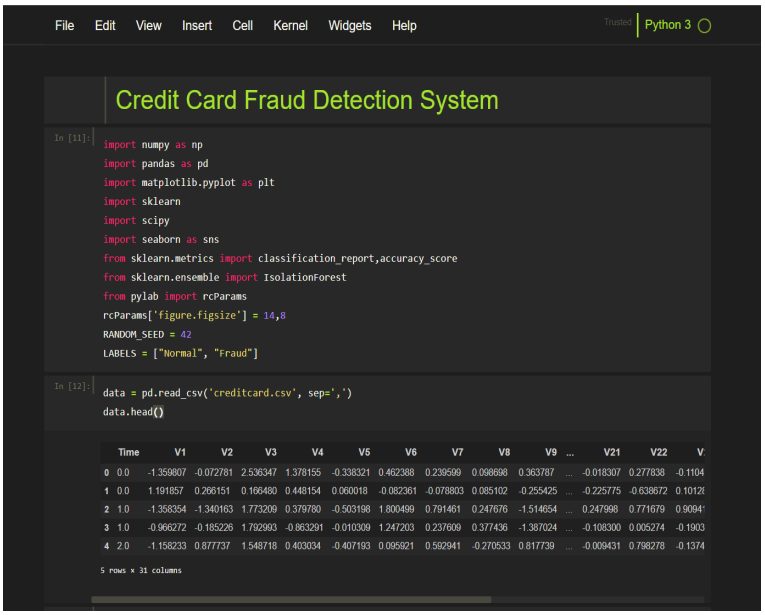

Figure 5.3: **OUTPUT 1 FOR THE GIVEN INPUT**

5.2.2 Output Design

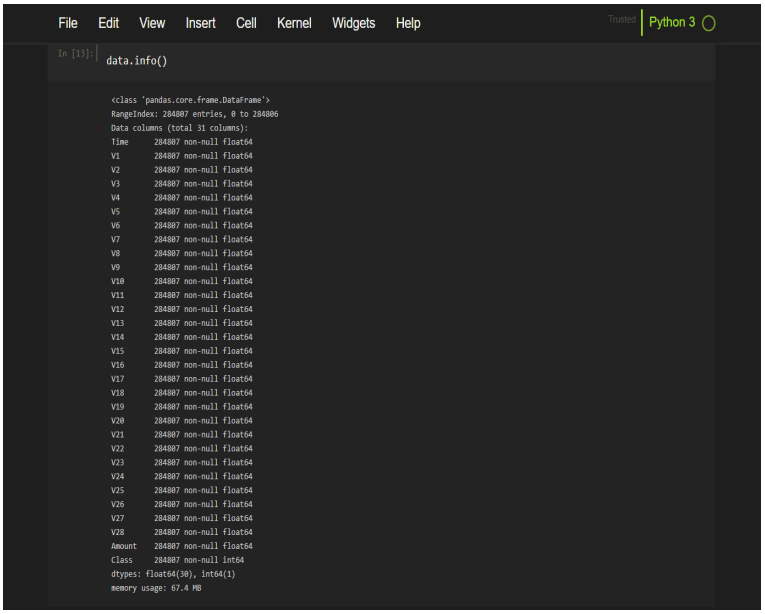

Figure 5.4: **OUTPUT 2 FOR THE GIVEN INPUT**

Description:-
In this figure the result is achieved from the given input or from the code and this particular segment represents a the tail of dataset which we have considered for the project.

5.3 Testing

5.4 Types of Testing

5.5 Unit testing

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the

documented specifications and contains clearly defined inputs and expected results.

5.6 Integration testing

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

5.7 System testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

5.8 Functional Testing

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.
Functional testing is centered on the following items:
Valid Input : identified classes of valid input must be accepted.
Invalid Input : identified classes of invalid input must be rejected.
Functions : identified functions must be exercised.
Output : identified classes of application outputs must be exercised.
Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive pro-

cesses must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

5.9 Test Result

5.10 Negavtive Test Cases

Test Case 1: Dataset Selection	Priority(H.L):High
Test Objective: to check Dataset Selection success or fail	
22252Test Description: In this HOME screen, when the user selects the credit card transactions datataset as inputl	
Requirement Verified: Yes	
Test Environment: System connected with the dataset.	
Actions	Expected Results
• when the user selects the dataset by clicking upload button.	• Uploading dataset fail • Select valid Dataset
Pass: no **Condition Pass**: No **Fail:** Yes	
Problems/Issues: Yes	
Notes: Selection is fail	

Figure 5.5: **TABULATION FOR NEGATIVE TEST CASE1**

Description:-
For this project we have performed a negative test case in order to check for a legit or fraud transaction.The transaction history and data is collected from the source and testing is done in order to classify the activity by considering legit=class1 and

fraud=class0.

Test Case 2: Execute Several Algorithms	Priority(H.L):High
Test Objective: has to check whether algorithms are working	
52Test Description: In this home page user dataset and after that he will check best ML algorithm which can find the credit card fraud detection	
Requirement Verified: No	
Test Environment: System connected with the dataset.	
Actions	Expected Results
• Run several algorithm by clicking Run button	• Comparing fail • Select valid dataset and algorithms
Pass: no **Condition Pass**: No **Fail:** Yes	
Problems/Issues: Yes	

Figure 5.6: **TABULATION FOR NEGATIVE TEST CASE2**

Description:-

For this project we have performed another negative test case in order to check for a legit or fraud transaction.The transaction history and data is collected from the source and testing is done in order to classify the activity by considering legit=class1 and fraud=class0.

5.11 Positive Test Cases

Test Case 1: Dataset Selection	Priority(H.L):High
Test Objective: to check dataset Selection success or fail	
22252Test Description: In this HOME screen, when the user selects the input dataset it display path label	
Requirement Verified: Yes	
Test Environment: System connected with the dataset.	
Actions	Expected Results
• when the user selects the input by clicking upload button.	• Dataset path Label can be shown
Pass: yes **Condition Pass**: No **Fail:** No	
Problems/Issues: No	
Notes: Dataset Selection successfully completed	

Figure 5.7: **TABULATION FOR POSITIVE TEST CASE1**

Description:-

For this project we have performed a positive test case in order to check for a legit or fraud transaction.The transaction history and data is collected from the source and testing is done in order to classify the activity by considering legit=class1 and fraud=class0.

Test Case 2: Running Several Algorithm	Priority(H.L):High
Test Objective:has to check ML algorithms related to trained Dataset	
22252Test Description: In this home page user will choose best ML algorithms for classification	
Requirement Verified: No	
Test Environment: System connected with the dataset.	
Actions	Expected Results
• Run ML algorithm by clicking Run button	• Display Features of fraud messages if the text contain fraud ids.
Pass: yes **Condition Pass**: No **Fail:** Yes	
Problems/Issues: No	
Notes: Ml Classification Algorithm successful.	

Figure 5.8: **TABULATION FOR POSITIVE TEST CASE2**

Description:-

For this project we have performed another positive test case in order to check for a legit or fraud transaction.The transaction history and data is collected from the source and testing is done in order to classify the activity by considering legit=class1 and fraud=class0.

Chapter 6

RESULTS AND DISCUSSIONS

6.1 Efficiency of the Proposed System

Credit card fraud becomes a serious concern to the world. Fraud brings huge financial losses to the world. This urged Credit card companies have been invested money to create and develop techniques to reveal and reduce fraud. The prime goal of this study is to define algorithms that confer the appropriate, and can be adapted by credit card companies for identifying fraudulent transactions more accurately, in less time and cost. Different machine learning algorithms are compared, including Logistic Regression, Decision Trees, Random Forest, Artificial Neural Networks, Logistic Regression, K-Nearest Neighbors, and K-means clustering. Because not all scenarios are the same, a scenario-based algorithm can be used to determine which scenario is the best fit for that scenario.The researchers use different performance measures employed (techniques) and algorithms to predict and show transactions fraudulent. Studies are refreshed and encouraged to improve the fraud detection basis to determine the weight that is suitable with cost factors, the tested accuracy, and detection accuracy. Surveys of such kind will allow the researchers to build a hybrid approach most accurate for fraudulent credit card transaction detection.

6.2 Comparison of Existing and Proposed System

Existing system:
In the existing system there are lot of machine learning approaches implemented for detecting credit card fraud detection which are implemented based on AE, IF, LOF and K- Means which are giving accuracy respectively. In this project we are trying to improve the accuracy by refining the data in a better way for efficient detection of credit card fraud. By using that k-means they try to cluster the dataset into 2 clusters.0 being non-fraud and 1 as Fraud parameters, but they cant able to classify clearly all the fields. If there are less dimensions the k-means can easily able to

cluster the data and find the fraud activities but if the same dataset contains more dimensions this may not generate the accurate results.

6.2.1 LIMITATION OF EXISTING SYSTEM

1.The Clustering produce the less accuracy when compared to Regression methods in scenarios like credit card fraud detection.
2.Comparatively with other algorithms k-means produce less accurate scores in prediction in this kind of scenarios.
3.This is accurate if we use for less dimensions
4.This is not accurate for large dimensional dataset.

Proposed system:

Here, the unsupervised learning approaches are used for fraud detections. In this project we collected credit card fraud data from kaggle below I am specifying description regarding the data set which was taken from kaggle directly. Our goal is to implement machine learning model in order to classify, to the highest possible degree of accuracy, credit card fraud from a dataset gathered from Kaggle. After initial data exploration, we knew we would implement a logistic regression model for best accuracy reports. For that we try to use Gaussian Mixture, Isolation Forest and K-Means algorithm and classify each and every record which is present in that dataset. As per the observation we can see Gaussian Mixture gives best accuracy.

6.2.2 ADVANTAGES OF PROPOSED SYSTEM

1.The results obtained by the GM Algorithm is best compared to any other Algorithms.
2.The Accuracy obtained was almost equal to 92.7 percent which proves using of GM gives best results.
3.The plots that were plotted according to the proper data that is processed during the implementation

6.3 Sample Code

```
#importing the modules
import numpy as np
```

```
#import sklearn python machine learning  modules
importsklearn as sk
#import pandasdataframes
import pandas as pd
#import matplotlib for plotting
importmatplotlib.pyplot as plt
#import datasets  andlinear_model from sklearn module
fromsklearn import datasets, linear_model
#import Polynomial features from sklearn module
fromsklearn.preprocessing import PolynomialFeatures
#import train_test_split data classification
fromsklearn.model_selection import train_test_split
#import ConfusionMatrix from pandas_ml
frompandas_ml import ConfusionMatrix
#reading the csvfile from C:/Python27
dataframe = pd.read_csv('C:/Python27/creditcard.csv', low_memory=False)
#dataframe.sample Returns a random sample of items from an axis of object.
#Thefrac keyword argument specifies the fraction of rows to return in the random sample, so frac=1
    means return all rows (in random order).
# If you wish to shuffle your dataframe in-place and reset the index
dataframe = dataframe.sample(frac=1).reset_index(drop=True)
#dataframe.head(n) returns a DataFrame holding the first n rows of dataframe.
dataframe.head()
printdataframe
```

6.4 Output

```
[ ]
Archive:  creditcardfraud.zip
  inflating: creditcard.csv
```

```
[ ] import pandas as pd
    import numpy as np
    import seaborn as sns
    import matplotlib.pyplot as plt
```

```
df=pd.read_csv('creditcard.csv')
df.head()
```

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	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	-0.110474	0.066928	0.1285
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	0.101288	-0.339846	0.1671
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	0.909412	-0.689281	-0.3276
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	-0.190321	-1.175575	0.6473
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	-0.137458	0.141267	-0.2060

Figure 6.1: **OUTPUT FOR THE GIVEN SAMPLE CODE 1**

```
df.describe()
```

	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	...	
count	284807.000000	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	2.848070e+05	...	2.84807
mean	94813.859575	3.918649e-15	5.682686e-16	-8.761736e-15	2.811118e-15	-1.552103e-15	2.040130e-15	-1.698953e-15	-1.893285e-16	-3.147640e-15	...	1.47312
std	47488.145955	1.958696e+00	1.651309e+00	1.516255e+00	1.415869e+00	1.380247e+00	1.332271e+00	1.237094e+00	1.194353e+00	1.098632e+00	...	7.34524
min	0.000000	-5.640751e+01	-7.271573e+01	-4.832559e+01	-5.683171e+00	-1.137433e+02	-2.616051e+01	-4.355724e+01	-7.321672e+01	-1.343407e+01	...	-3.48303
25%	54201.500000	-9.203734e-01	-5.985499e-01	-8.903648e-01	-8.486401e-01	-6.915971e-01	-7.682956e-01	-5.540759e-01	-2.086297e-01	-6.430976e-01	...	-2.28394
50%	84692.000000	1.810880e-02	6.548556e-02	1.798463e-01	-1.984653e-02	-5.433583e-02	-2.741871e-01	4.010308e-02	2.235804e-02	-5.142873e-02	...	-2.94501
75%	139320.500000	1.315642e+00	8.037239e-01	1.027196e+00	7.433413e-01	6.119264e-01	3.985649e-01	5.704361e-01	3.273459e-01	5.971390e-01	...	1.86377
max	172792.000000	2.454930e+00	2.205773e+01	9.382558e+00	1.687534e+01	3.480167e+01	7.330163e+01	1.205895e+02	2.000721e+01	1.559499e+01	...	2.72028

8 rows × 31 columns

```
df.info()
```

Figure 6.2: **OUTPUT FOR THE GIVEN SAMPLE CODE 2**

Chapter 7

CONCLUSION AND FUTURE ENHANCEMENTS

7.1 Conclusion

Credit card fraud detection dataset is used with machine learning algorithms in this experiment to identify which algorithm works better with Credit card fraud detection.The fraud transaction detection is the major issue of prediction due to a frequent and large number of transactions. The fraud transaction prediction has the two phases which are feature extraction and classification. In the first phase, the feature extraction technique is applied and in the second phase, classification is applied for the fraud transaction detection. In this various techniques of credit card fraud detection are reviewed. In future hybrid approach will be designed for the credit card fraud detection. In total five algorithms such as SVM, Naïve Bayes, Logistic Regression, KNN, and Random Forest. In which the best score result is given by Random forest and then KNN. As the MCC is used to measure the performance of an algorithm, the best score of MCC is 1 and its values lie between -1 and 1.Credit card fraud has become more and more rampant in recent years. Fraud detection methods are continuously developed to defend criminals in adapting to their strategies. In Fraud detection, identifying Fraud as quickly as possible once it has been done through fraud detection techniques, is now becoming easier and faster. The techniques which were studied here, through which credit card fraud can be detected quickly and fast and the crime can be stopped.

7.2 Future Enhancements

7.3 Faster detection

A machine learning model can quickly identify any drifts from regular transactions and user behaviours in real time. By recognising anomalies, such as a sudden increase in transactional amount or location change, ML algorithms can minimise the risk of fraud and ensure more secure transactions.

7.4 Higher accuracy

Conventional fraud detection techniques cause errors at the payment gateways that sometimes result in genuine customers being blocked. With sufficient training data and insights, ML models can achieve higher accuracy and precision, reducing these errors along with the time required to be spent on performing manual analysis.

7.5 Improved efficiency with larger data

Once an algorithm picks up different transactional patterns and behaviours, it can efficiently work with large datasets to separate authentic payments from fraudulent ones. The models can analyse huge amounts of data in seconds while offering real-time insights for improved decision-making capabilities.

Chapter 8

INDUSTRY DETAILS

8.1 OJAS INNOVATIVE TECHNOLOGIES PVT.LTD

8.2 Duration of Internship (10/01/2021-10/08/2023)

8.3 6-MONTHS

8.4 GACHIBOWLI,HYDERABAD.,

8.5 Internship offer letter

OJAS INNOVATIVE TECHNOLOGIES PVT LTD

Dear P. Sai Sesha Varma,

We are pleased to offer you an internship at our company in the Computer Engineering department at our **OJAS Innovative Technologies Pvt Ltd.** Your internship shall commence on **10th January 2023** and shall end on **11th August 2023**. The terms and conditions of your internship with the Company are set forth below:

1. Subject to your acceptance of the terms and conditions contained herein, your project and responsibilities during the Term will be determined by the supervisor assigned to you for the duration of the internship.
2. Your timings will be from **10.00am to 6.00pm,** Monday to Friday. Please be sure to bring Required documents with you on your first day to complete your profile.
3. You will sign a confidentiality agreement with the company before you commence your internship.
4. The internship cannot be construed as an employment or an offer of employment with **OJAS Innovative Technologies Pvt Ltd.**

Please confirm your acceptance of the terms of this offer by **09th January 2023** failing which, we have the right to cancel the internship. We look forward to having you on our team! If you have any questions, please feel free to reach out to us.

Thanks & Regards

HARISH RAO

Hr-Manager

Ojas Innovative Technologies Pvt Ltd

Address: PLOT NO. 504. MANIKAR BUILDINGS. GACHIBOWLI. HYDERABAD.

E-Mail : info@ojasinnovativetechnologies.in
Contact Details: 040-6517760, 7815970380

8.6 Project Commencement Form

Vel Tech
Rangarajan Dr. Sagunthala
R&D Institute of Science and Technology
(Deemed to be University Estd. u/s 3 of UGC Act, 1956)

Project Commencement Form

Name of the Industry: OJAS INNOVATIVE TECHNOLOGIES

Address: GACHIBOWLI, HYDERBAD

Team Details:

S.No	ID No	Student Name	Degree & Branch
1.	VTU13085	KURALI LAHARI SAISREE	B. TECH /CSE
2.	VTU13297	P.SAI SESHA VARMA	
3.	VTU15769	D.SAI KRUPA	

Date of reporting for project work: 21-01-2023

Name of the Industry Supervisor : P. HARISH

Department : COMPUTER SCIENCE AND TECHNOLOGY

Designation : SOFTRWARE DEVELOPER

Contact Number : 9640212290

Email ID : h9640212290@gmail.com

Name of the Internal Supervisor : DR.M. GURU VIMAL KUMAR

Contact No. : 9791842009

Email ID : drguruvimalkumarm@veltech.edu.in

Tentative Project Title / Project domain: CREDIT CARD FRAUD DETECTION/ MACHINE LEARNING

Brief Project/task description: It is important for credit card companies to know see fraudulent credit card sales for customers they are not charged for things they did not buy. Such problems can be dealt with Data Science and its importance, and Mechanical

8.7 Internship Completion certificate

OJAS INNOVATIVE TECHNOLOGIES PVT LTD

Internship Certificate

TO WHOM IT MAY CONCERN

This is to certify that Mr. **P SAI SESHA VARMA** bearing hall ticket number **19UECS0780** from **VELTECH University** has successfully completed an internship with OJAS Innovative Technologies Pvt. Ltd. as an junior trainee Intern on the **Machine Learning Domain** from 05th January 2023 to 22nd April 2023.

They have worked on the **Credit Card Fraud Detection** Project under the supervision and guidance of **Mr. Veera Reddy**. During the internship, he has gained several learning's and developed considerable skills.

Besides showing high comprehension capacity, managing assignments with the utmost expertise, and exhibiting maximal efficiency, he has also maintained an outstanding professional demeanor and showcased excellent moral character throughout the internship period.

I hereby certify his overall work as excellent to the best of my knowledge.

Wishing him the best of luck in his future endeavors.

For **OJAS Innovative Technologies Pvt. Ltd.**

Authorized Signatory

Address: PLOT NO. 504, MANIKAR BUILDINGS, GACHIBOWLI, HYDERABAD.

E-Mail : info@ojasinnovativetechnologies.in
Contact Details: 040-6517760, 7815970380

Chapter 9

PLAGIARISM REPORT

StudyMoose
Free essays

ABOUT US OUR SERVICES ▼ TOOLS ▼ LOG IN/SIGN UP HIRE WRITER

The uniqueness of the text: 88.1%

See all the sources that contain text like yours

I NEED PLAGIARISM-FREE CONTENT

credit card fraud detection major project report submitted in partial fulfillment of the requirement for award of the degree of bachelor of technology in computer science engineering by psai sesha varma 19uecs0780 13297 klahari sai sree 19uecs0520 13085 dsai krupa 19uecs0252 15769 under the guidance of drmguru vimal kumar btechmephd assistant professor department of computer science engineering school of computing vel tech rangarajan dr sagunthala rd institute of science technology deemed to be university estd us 3 of ugc act 1956 accredited by naac with a grade chennai 600 062 tamilnadu india april 2023 credit card fraud detection major project report submitted in partial fulfillment of the requirement for award of the degree of bachelor of technology in computer science engineering by psai sesha varma 19uecs0780 13297 klahari sai sree 19uecs0520 13085 dsai krupa 19uecs0252 15769 under the guidance of drmguru vimal kumar btechmephd assistant professor department of computer science engineering school of computing vel tech rangarajan dr

The length of the text: 16117 (No spaces: 13532)

GET NEW REPORT

Sources:	Similarity index:	View in the text:
https://www.academia.edu/32890340/Project_Proposal_Project_Title_Credit_Card_Fraud_Detection_Using_Neural_Network	10.4	Show
https://medium.com/diko-hary-adhanto-portfolio/card-fraud-detection-based-on-data-analytic-perspective-dd742e60ad3a	5.5	Show

Chapter 10

SOURCE CODE & POSTER PRESENTATION

10.1 Source Code

```
    MAIN WINDOW

fromtkinter import messagebox
fromtkinter import *
fromtkinter import simpledialog
importtkinter
fromtkinter import filedialog
importmatplotlib.pyplot as plt
import numpy as np
fromtkinter.filedialog import askopenfilename
import numpy as np
import pandas as pd
fromsklearn import *
fromsklearn.model_selection import train_test_split
fromsklearn.metrics import accuracy_score
fromsklearn.metrics import classification_report
fromsklearn.ensemble import RandomForestClassifier
#from sklearn.tree import export_graphviz
#from IPython import display


main = tkinter.Tk()
main.title("Credit Card Fraud Detection") #designing main screen
main.geometry("1300x1200")

global filename
globalcls
global X, Y, X_train, X_test, y_train, y_test
globalrandom_acc # all global variables names define in above lines
global clean
global attack
global total


deftraintest(train):     #method to generate test and train data from dataset
```

```
    X = train.values[:, 0:29]
    Y = train.values[:, 30]
print(X)
print(Y)
X_train, X_test, y_train, y_test = train_test_split(
    X, Y, test_size = 0.3, random_state = 0)
return X, Y, X_train, X_test, y_train, y_test

defgenerateModel(): #method to read dataset values which contains all five features data
global X, Y, X_train, X_test, y_train, y_test
train = pd.read_csv(filename)
    X, Y, X_train, X_test, y_train, y_test = traintest(train)
text.insert(END,"Train& Test Model Generated\n\n")
text.insert(END,"Total Dataset Size : "+str(len(train))+"\n")
text.insert(END,"Split Training Size : "+str(len(X_train))+"\n")
text.insert(END,"Split Test Size : "+str(len(X_test))+"\n")



def upload(): #function to upload tweeter profile
global filename
filename = filedialog.askopenfilename(initialdir="dataset")
text.delete('1.0', END)
text.insert(END,filename+" loaded\n");



def prediction(X_test, cls):  #prediction done here
y_pred = cls.predict(X_test)
for i in range(50):
print("X=%s, Predicted=%s" % (X_test[i], y_pred[i]))
returny_pred

# Function to calculate accuracy
defcal_accuracy(y_test, y_pred, details):
accuracy = accuracy_score(y_test,y_pred)*100
text.insert(END,details+"\n\n")
text.insert(END,"Accuracy : "+str(accuracy)+"\n\n")
return accuracy


defrunRandomForest():
    headers = ["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"]
globalrandom_acc
globalcls
global X, Y, X_train, X_test, y_train, y_test
cls = RandomForestClassifier(n_estimators=50,max_depth=2,random_state=0,class_weight='balanced')
cls.fit(X_train, y_train)
```

```
text.insert(END,"Prediction Results\n\n")
prediction_data = prediction(X_test, cls)
random_acc = cal_accuracy(y_test, prediction_data,'Random Forest Accuracy')
    #str_tree = export_graphviz(cls, out_file=None, feature_names=headers,filled=True,
        special_characters=True, rotate=True, precision=0.6)
    #display.display(str_tree)



def predicts():
global clean
global attack
global total
clean = 0;
attack = 0;
text.delete('1.0', END)
filename = filedialog.askopenfilename(initialdir="dataset")
test = pd.read_csv(filename)
test = test.values[:, 0:29]
total = len(test)
text.insert(END,filename+" test file loaded\n");
y_pred = cls.predict(test)
for i in range(len(test)):
ifstr(y_pred[i]) == '1.0':
attack = attack + 1
text.insert(END,"X=%s, Predicted = %s" % (test[i], 'Contains Fraud Transaction Signature')+"\n\n")
else:
clean = clean + 1
text.insert(END,"X=%s, Predicted = %s" % (test[i], 'Transaction Contains Cleaned Signatures')+"\n\n"
    )


def graph():
height = [total,clean,attack]
bars = ('Total Transactions','NormalTransaction','Fraud Transaction')
y_pos = np.arange(len(bars))
plt.bar(y_pos, height)
plt.xticks(y_pos, bars)
plt.show()

font = ('times', 16, 'bold')
title = Label(main, text='Credit Card Fraud Detection Using Random Forest Tree Based Classifier')
title.config(bg='greenyellow', fg='dodger blue')
title.config(font=font)
title.config(height=3, width=120)
title.place(x=0,y=5)

font1 = ('times', 12, 'bold')
text=Text(main,height=20,width=150)
scroll=Scrollbar(text)
```

```
text.configure(yscrollcommand=scroll.set)
text.place(x=50,y=120)
text.config(font=font1)


font1 = ('times', 14, 'bold')
uploadButton = Button(main, text="Upload Credit Card Dataset", command=upload)
uploadButton.place(x=50,y=550)
uploadButton.config(font=font1)

modelButton = Button(main, text="Generate Train & Test Model", command=generateModel)
modelButton.place(x=350,y=550)
modelButton.config(font=font1)

runrandomButton = Button(main, text="Run Random Forest Algorithm", command=runRandomForest)
runrandomButton.place(x=650,y=550)
runrandomButton.config(font=font1)

predictButton = Button(main, text="Detect Fraud From Test Data", command=predicts)
predictButton.place(x=50,y=600)
predictButton.config(font=font1)

graphButton = Button(main, text="Clean & Fraud Transaction Detection Graph", command=graph)
graphButton.place(x=350,y=600)
graphButton.config(font=font1)

exitButton = Button(main, text="Exit", command=exit)
exitButton.place(x=770,y=600)
exitButton.config(font=font1)

main.config(bg='LightSkyBlue')
main.mainloop()
```

10.2 Poster Presentation

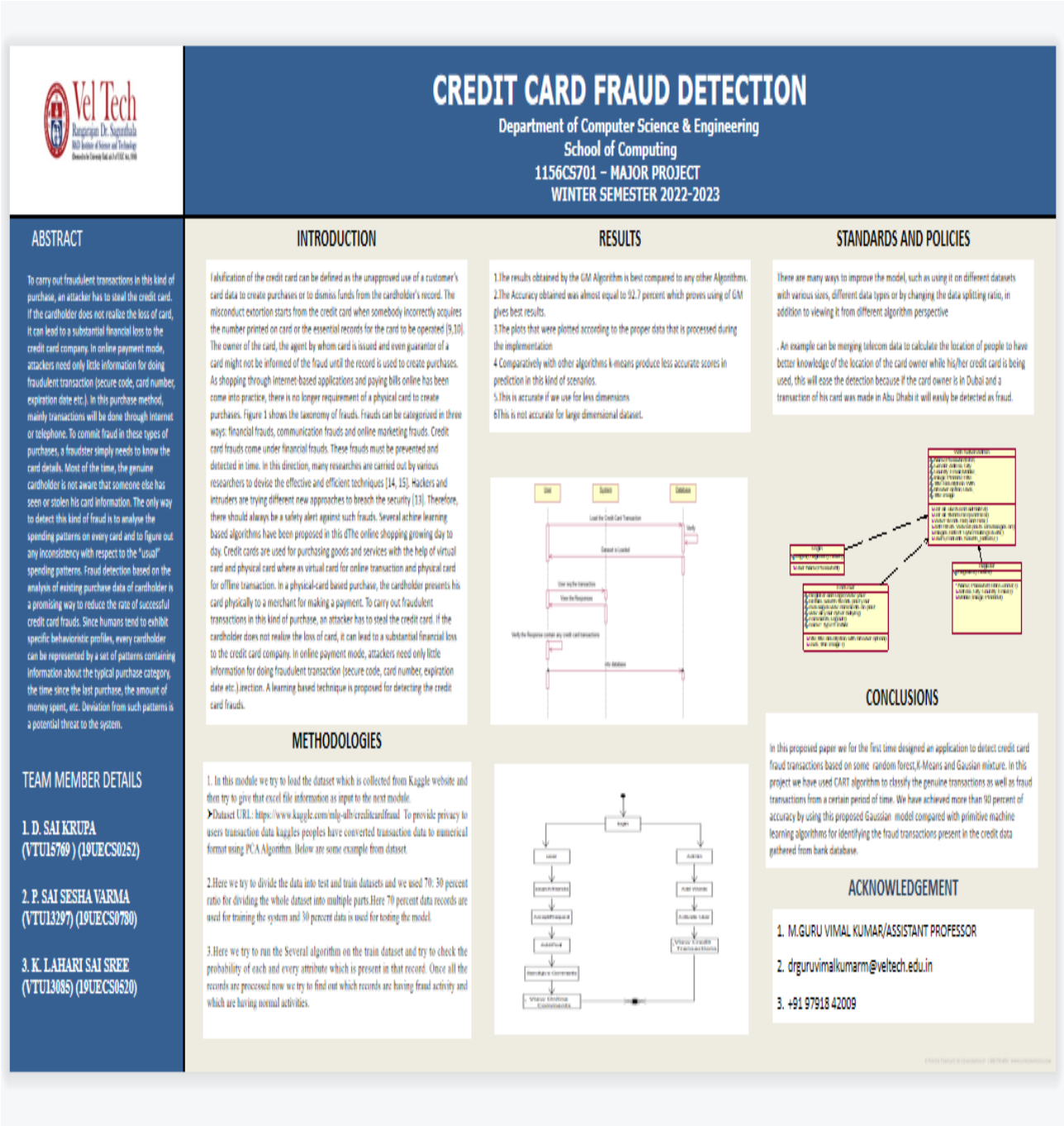

Figure 10.1: **POSTER**

References

[1] “Credit Card Fraud Detection using Machine Learning and Deep Learning Techniques – by Mohammed Azhan, Shazli Meraj” Published by Proceedings of the Third International Conference on Intelligent Sustainable Systems [ICISS 2020] IEEE Xplore Part Number: CFP20M19-ART; ISBN: 978-1-7281-7089-3

[2] “A Survey Paper on Credit Card Fraud Detection Techniques – by Aisha Mohammad Fayyomi, Derar Eleyan, Amina Eleyan” Published by International Journal of Scientific Technology Research Volume 10, Issue 09, September 2021 ISSN 2277-8616

[3] “A Machine Learning Based Credit Card Fraud Detection Using the GA Algorithm for Feature Selection – by Emmanuel Ileberi, Yanxia Sun, Zenghui Wang” Published by Ileberi et al. Journal of Big Data (2022) https://doi.org/10.1186/s40537-022-00573-8

[4] S. H. Projects and W. Lovo, —JMU Scholarly Commons Detecting credit card fraud: An analysis of fraud detection techniques,‖ 2020.

[5] A. H. Alhazmi and N. Aljehane, —A Survey of Credit Card Fraud Detection Use Machine Learning,‖ 2020 Int. Conf. Comput. Inf. Technol. ICCIT 2020, pp. 10–15, 2020, doi: 10.1109/ICCIT-144147971.2020.9213809

[6] B. Wickramanayake, D. K. Geeganage, C. Ouyang, and Y. Xu, —A survey of online card payment fraud detection using data mining-based methods,‖ arXiv, 2020

[7] A. Agarwal, —Survey of Various Techniques used for Credit Card Fraud Detection,‖ Int. J. Res. Appl. Sci. Eng. Technol., vol. 8, no. 7, pp. 1642–1646, 2020, doi: 10.22214/ijraset.2020.30614.

[8] C. Reviews, —a Comparative Study: Credit Card Fraud,‖ vol. 7, no. 19, pp. 998–1011, 2020.

[9] R. Sailusha, V. Gnaneswar, R. Ramesh, and G. Ramakoteswara Rao, —Credit Card Fraud Detection Using Machine Learning,‖ Proc. Int. Conf. Intell. Comput. Control Syst. ICICCS 2020, no. Iciccs, pp. 1264–1270, 2020, doi: 10.1109/ICICCS48265.2020.9121114.

[10] A. RB and S. K. KR, —Credit Card Fraud Detection Using Artificial Neural Network,‖ Glob. Transitions Proc., pp. 0–8, 2021, doi: 10.1016/j.gltp.2021.01.006.

[11] S. Bagga, A. Goyal, N. Gupta, and A. Goyal, —Credit Card Fraud Detection using Pipeling and Ensemble Learning,‖ Procedia Comput. Sci., vol. 173, pp. 104–112, 2020, doi: 10.1016/j.procs.2020.06.014.

[12] R. San Miguel Carrasco and M.-A. Sicilia-Urban, —Evaluation of Deep Neural Networks for Reduction of Credit Card Fraud Alerts,‖ IEEE Access, vol. 8, pp. 186421–186432, 2020, doi: 10.1109/access.2020.3026222.

[13] G. Kibria and M. Sevkli, —Application of Deep Learning for Credit Card Approval: A Comparison with Application of Deep Learning for Credit Card Approval: A Comparison with Two Machine Learning Techniques,‖ no. January, pp. 0–5, 2021, doi: 10.18178/ijmlc.2021.11.4.1049.

[14] —K-Nearest Neighbor(KNN) Algorithm for Machine Learning - Javatpoint.‖ https://www.javatpoint.com/knearest-neighbor-algorithm-for-machine-learning (accessed Apr. 03, 2021).

[15] D. D. Borse, P. S. H. Patil, and S. Dhotre, —Credit Card Fraud Detection Using Naïve Bayes and C4,‖ vol. 10, no. 1, pp. 423–429, 2021.

General Instructions

- Cover Page should be printed as per the color template and the next page also should be printed in color as per the template
- **Wherever Figures applicable in Report , that page should be printed in color**
- Dont include general content , write more technical content
- Each chapter should minimum contain 3 pages
- Draw the notation of diagrams properly
- Every paragraph should be started with one tab space
- Literature review should be properly cited and described with content related to project
- All the diagrams should be properly described and dont include general information of any diagram
- Example Use case diagram - describe according to your project flow
- All diagrams,figures should be numbered according to the chapter number
- Test cases should be written with test input and test output
- All the references should be cited in the report
- **Internship Offer letter and neccessary documents should be attached**
- **Strictly dont change font style or font size of the template, and dont customize the latex code of report**
- **Report should be prepared according to the template only**
- **Any deviations from the report template,will be summarily rejected**
- **Number of Project Soft Binded copy for each and every batch is (n+4) copies as given in the table below**
- **Attach the CD in last Cover page of the Project Report with CD cover and details of batch like Title,Members name and VTU No ,Batch No,Project category (Inhouse/Internship)should be written in Marker pen**

- For **Standards and Policies** refer the below link
  https://law.resource.org/pub/in/manifest.in.html
- Plagiarism should be less than 15%
- **Journal/Conference Publication proofs should be attached in the last page of Project report after the references section**

General Instructions