

CREDIT CARD FRAUD DETECTION

A PROJECT REPORT

Submitted by:
GONTLA BHARGAVA SAI SATHVIK-18BCI0087

Course Code: CSE3013

Course Title: Artificial Intelligence

Under the guidance of
Dr. GOPICHAND
Associate Professor, SCOPE,
VIT, Vellore.



VIT[®]

Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

JUNE, 2021

INDEX

S.no	Topic	Page no.
1	ABSTRACT	3
2	INTRODUCTION	4
3	LITERATURE SURVEY AND PROBLEM DEFINITION	5-12
4	OVERVIEW OF THE PROJECT	13
5	SYSTEM DESIGN	14-19
6	IMPLEMENTATION	20 - 21
7	OUTPUT AND PERFORMANCE ANALYSIS	22-23
8	CONCLUSION AND FUTURE WORKS	24
9	REFERENCES	24

ABSTRACT

It is a known fact that fraud is a crime and a civil law violation. Many fraud cases involve complicated financial transactions conducted by 'white collar criminals' such as business professionals with specialized knowledge and criminal intent. The people doing fraud can contact their potential victims through various ways, for example face-to-face interaction, by post, phone calls, SMS and/or emails all of this by claiming to be someone else. The reason this works is because of the difficulty of verifying the true identities and legitimacy of individuals and companies, the ease with which fraudsters can divert visitors to dummy sites and steal personal financial information, the international dimensions of the web and ease with which fraudsters can hide their true location, all contribute to making internet fraud the fastest growing area of fraud. Fraud detection is a challenging problem. The fact is that fraudulent transactions are rare; they represent a very small fraction of activity within an organization.

The challenge is that a small percentage of activity can quickly turn into big dollar losses without the right tools and systems in place. Criminals are crafty. As traditional fraud schemes fail to pay off, fraudsters have learned to change their tactics. The good news is that with advances in machine learning, systems can learn, adapt and uncover emerging patterns for preventing fraud. Financial fraud is an issue with far reaching consequences in the finance industry, government, corporate sectors, and for ordinary consumers. Increasing dependence on new technologies such as cloud and mobile computing in recent years has compounded the problem.

Traditional methods of detection involve extensive use of auditing, where a trained individual manually observes reports or transactions in an attempt to discover fraudulent behaviour. This method is not only time consuming, expensive and inaccurate, but in the age of big data it is also impractical. Not surprisingly, financial institutions have turned to automated processes using statistical and computational methods. In this project we are trying to present a comprehensive investigation on financial fraud detection practices using such soft computing methods, with a particular focus on computational intelligence-based techniques. Classification of the practices based on key aspects such as detection algorithm used, fraud type investigated, and success rate have been covered. Issues and challenges associated with the current practices and potential future direction of research have also been identified.

1. INTRODUCTION

Whether internal or external, there are a wide variety of threats posed to enterprises across multiple industries. The most difficult threat to diagnose & address, however, is fraud. Fraudulent activity is a high-cost threat that can compromise the integrity of your company as well as cripple your bottom line. Fraud can take the form of internal activity, such as an employee modifying financial records, or can arise from an external threat, such as customer credit card fraud. In either case, the use of fraud detection analytics using predictive data science methodologies enables companies to discover potentially fraudulent activity before it occurs.

Fraud prevention isn't just about basic regressive analysis. On the contrary, it's about connecting the data points to discover potential fraudulent behaviour before it happens. This starts with finding interactions between products, locations, and devices and then mapping those data points to individual users, customers, and/or employees. This approach effectively connects together vast quantities of knowledge with all of the people who somehow interacted with that knowledge. The wide variety of threat types and varieties pose a significant challenge for fraud detection solutions. Disgruntled employees and criminal elements are continually using more advanced techniques to siphon revenue away from companies; their methods can be straightforward, such as a staged car accident, or more nuanced, such as using accounting irregularities to mask embezzlement schemes. Given the complexity involved, fraud detection techniques used in predictive analytics need to excel at creating connections from raw data and then discovering which interactions convey potential fraudulent behaviour. Creating all of those connections from raw data is the job of Data Science Studio.

2. LITERATURE SURVEY

1	PAPER DESCRIPTION	ALGORITHM	DATA SET	PERFORMANCE	EXISTING METHODS	PROPOSED METHODS
	<p>Title: “Credit Card Fraud Identification Using Artificial Neural Networks”.</p> <p>Author: Chandrabhas Mishra, Dharmendra Lal Gupta, Raghuraj Singh .</p> <p>This paper presents the performance analysis and a comparative study of results of various techniques for credit card fraud identification.</p>	<p>1. BR Algorithm</p> <p>2. GDA algorithm</p> <p>3. LM Algorithm</p>	German (Statlog) credit dataset	<p>(i) Accuracy of BR Algorithm: Percentage Error = $1.3019258 - 1.3 / 1.3 * 100 = 0.1481\%$ Percentage Accuracy = $100 - 0.1481\% = 99.85\%$</p> <p>(ii) Accuracy of GDA algorithm: Percentage Error = $1.243893918 - 1.3 / 1.3 * 100 = 4.3158\%$ Percentage Accuracy = $100 - 4.3158\% = 95.68\%$</p> <p>(iii) Accuracy of LM Algorithm: Percentage Error = $1.221545331 - 1.3 / 1.3 * 100 = 6.0439\%$ Percentage Accuracy = $100 - 6.0439\% = 93.96\%$</p>	To address the credit card fraud issue, institutions currently apply many fraud prevention strategies such as credit card authorization, address verification system and (AVS), rule based detection	This research paper proposes a model for credit card fraud identification using artificial neural network. Accuracy of BR, GDA, and LM techniques for the proposed artificial neural network model applied on two different datasets taken from UCI Repository.

2	PAPER DESCRIPTION	ALGORITHM	DATA SET	PERFORMANCE	EXISTING METHODS	PROPOSED METHODS
	<p>Title: “Credit Card Fraud Detection Using Neural Networks”.</p> <p>Author: Divya Murli, Shailesh Jami, Devika Jog, Sreesha Nath.</p> <p>This paper presents the various parameters that were considered while training and testing the neural network.</p>	Back propagation	Acquired from a data mining blog. It has summary of the transactions of 20000 active credit card holders past six months.	The classification is very accurate and within the limit of maximum error. The results generated can be further optimized by increasing the number of transactions in the training sets and changing the neural network architecture	Credit card fraud have been prevalent for a very long time and several algorithms have being devised are using a variety of classification mechanisms, like Bayesian classification, Hidden Markov model (HMM) etc	In this paper, idea of implementing a mechanism to detect credit card fraud using neural networks using Neuroph IDE is proposed. This paper presents a paradigm to detect credit card frauds using artificial neural networks. The model provides an environment to work with neural networks.

3	PAPER DESCRIPTION	ALGORITHM	DATA SET	PERFORMANCE	EXISTING METHODS	PROPOSED METHODS
	<p>Title: “ Recognition of fraud in online banking by using confirmatory learning in neural network” .</p> <p>Author: Alireza Pouramirarsalani, Majid Khalilian, Alireza Nikravanshalmani.</p> <p>This research introduces new approach for exploring fraud in online banking that has high speed in recognizing and predicting the fraud.</p>	Confirmatory learning	Consideration of fraud in credit cards of one financial institution that in this research, 50 million transactions of 1 million credit cards are used.	The relation of correct transactions than transactions with fraud is 0.05% in this research.	Current technique is regression neural network. The algorithms that are currently used for recognition of fraud are generally performed by consideration of related information of customer like account number and performed transactions.	<p>In this research, they have considered specially the misuses that perform through online banking. The two approaches used are:</p> <ol style="list-style-type: none"> 1. Anomaly detection 2. Misuse detection The approach is of confirmatory learning by the purpose of fraud detection in online banks.

4	PAPER DESCRIPTION	ALGORITHM	DATA SET	PERFORMANCE	EXISTING METHODS	PROPOSED METHODS
	<p>Title: “A Neural Network Based Model for Detecting Irregularities in E-Banking Transactions”.</p> <p>Author: J.A Adeyiga, J.O. Ezike, A. Omotosho & W. Amakulor.</p> <p>This study applies neural network techniques to the bank fraud prediction problem.</p>	Supervised Back propagation training algorithm	Captured with the use of techniques and the CRISP-DM management model.	In this work, the irregularity detection system Model has sought to reduce the risk level of fraudulent transactions that take place in the Nigerian banking industry thereby aiding in the decrement of bank fraud. This will bring about reduced fraudulent transactions if implemented properly	The Cross Industry Standard Process for Data Mining – CRISP-DM is an existing model of a data mining process used to solve similar problems by experts	They have designed a Neural Network-Based Model that employ multi-layered Feed Forward Artificial Neural Network on database system for collecting training data for the Artificial Neural Networks.

5	PAPER DESCRIPTION	ALGORITHM	DATA SET	PERFORMANCE	EXISTING METHODS	PROPOSED METHODS
	<p>Title: “Application of Credit Card Fraud Detection: Based on Bagging Ensemble Classifier”.</p> <p>Author: Masoumeh Zareapoor, Pourya Shamsolmoali.</p> <p>This paper examined the performance of three states of art data mining techniques, with bagging ensemble classifier based on decision tree algorithm which is a novel technique in area of credit card fraud detection system.</p>	K-Nearest Neighbor algorithm	millions of credit card transactions are processed every day.	This is incorporated in the model by creating four sets of dataset (Df1, Df2, Df3, DF4) which the fraud rate in each of them were 20%, 15%, 10%, 3% respectively. Bagging classifier based decision tree algorithm performance is found to be stable gradually during the evaluation.	The most commonly techniques used fraud detection methods are Naïve Bayes (NB), Support Vector Machines (SVM), K-Nearest Neighbor algorithms (KNN). These techniques can be used alone or in collaboration using ensemble or meta-learning techniques to build classifiers.	They have trained various data mining techniques used in credit card fraud detection and evaluate each methodology based on certain design criteria.

6	PAPER DESCRIPTION	ALGORITHM	DATA SET	PERFORMANCE	EXISTING METHODS	PROPOSED METHODS
	<p>Title: “Fraud Detection of Credit Card Payment System by Genetic Algorithm”.</p> <p>Author: K.RamaKalyani, D.UmaDevi.</p> <p>This paper is to propose a credit card fraud detection system using genetic algorithm. Genetic algorithms are evolutionary algorithms which aim at obtaining better solutions as time progresses.</p>	Genetic algorithm	In financial institutions, use the fraud detection which is based on customer behavior variables.	In this study fraud detected and fraud transactions are generated with the given sample data set. If this algorithm is applied into bank credit card fraud detection system, the probability of fraud transactions can be predicted soon after credit card transactions by the banks. And a series of anti-fraud strategies can be adopted to prevent banks from great losses before and reduce risks	<p>1. Based on CC usage location.</p> <p>2. Based on CC over draft</p> <p>3. Based on CC book balance.</p> <p>4. Based on CC usage Frequency.</p>	<p>In this paper, they found the detection of credit card fraud mechanism and examine the result based on the principles of this algorithm.</p> <p>In this paper we saw genetic algorithm that are being used to execute credit card fraud how credit card fraud impact on financial institution as well as merchant and customer, fraud detection technique by genetic algorithm.</p>

7	PAPER DESCRIPTION	ALGORITHM	DATA SET	PERFORMANCE	EXISTING METHODS	PROPOSED METHODS
	<p>Title: “CREDIT CARD FRAUD DETECTION USING SELF ORGANIZING MAPS”.</p> <p>Author: Vladmir ZASLAVSKY and STRIZHAK.</p> <p>This paper presents the various parameters that were considered while training and testing the neural network.</p>	Back propagation	Acquired from a data mining blog. It has summary of the transactions of 20000 active credit card holders past six months.	The classification is very accurate and within the limit of maximum error. The results generated can be further optimized by increasing the number of transactions in the training sets and changing the neural network architecture	Credit card fraud have been prevalent for a very long time and several algorithms have being devised are using a variety of classification mechanisms, like Bayesian classification, Hidden Markov model (HMM) etc	<p>In this paper, idea of implementing a mechanism to detect credit card fraud using neural networks using Neuroph IDE is proposed.</p> <p>This paper presents a paradigm to detect credit card frauds using artificial neural networks. The model provides an environment to work with neural networks.</p>

2.2 PROBLEM DEFINITION

Fraud detection is a challenging problem. The fact is that fraudulent transactions are rare; they represent a very small fraction of activity within an organization. The challenge is that a small percentage of activity can quickly turn into big dollar losses without the right tools and systems in place. Increasing dependence on new technologies such as cloud and mobile computing in recent years has compounded the problem.

3. OVERVIEW OF THE WORK

Traditional methods of detection involve extensive use of auditing, where a trained individual manually observes reports or transactions in an attempt to discover fraudulent behavior. This method is not only time consuming, expensive and inaccurate, but in the age of big data it is also impractical. So, we were trying to find the most efficient way for this process.

Here, we have worked on XGBoost classifier, and tried to compare the accuracy with 4 other algorithms, when trained with the same data-set.

1. RANDOM FOREST ALGORITHM
2. ANN WITH KERAS
3. K NEAREST NEIGHBOR ALGORITHM
4. NAIVE BAYES CLASSIFIER ALGORITHM

3.1. OBJECTIVES OF THE PROJECT

In this project we are trying to present a comprehensive investigation on financial fraud detection practices using such soft computing methods, with a particular focus on computational intelligence-based techniques. Classification of the practices based on key aspects such as detection algorithm used, fraud type investigated, and success rate have been covered. Issues and challenges associated with the current practices and potential future direction of research have also been identified.

3.2. SOFTWARE REQUIREMENTS

1. Operating system: Windows 8/10
2. Coding language: Python 3.6
3. Jupyter Notebook, Anaconda

3.3. HARDWARE REQUIREMENTS

1. Processor: Pentium i3 or higher.
2. RAM: 4 GB or higher.
3. Hard Disk Drive: 20 GB (free).
4. Peripheral Devices: Monitor, Mouse and Keyboard.

4. SYSTEM DESIGN

Here, we have worked on XGBoost Classifier Algorithm, and tried to compare the accuracy with other algorithms, when trained with the same data-set.

4.1. ALGORITHMS

1. RANDOM FOREST ALGORITHM

The Random Forest „isolates“ observations by randomly selecting a feature and then randomly selecting a split value between the maximum and minimum values of the selected feature. Since recursive partitioning can be represented by a tree structure, the number of splitting required to isolate a sample is equivalent to the path length from the root node to the terminating node. This path length, averaged over a forest of such random trees, is a measure of normality and our decision function. Random partitioning produces noticeably shorter paths for anomalies. Hence, when a forest of random trees collectively produces shorter path lengths for particular samples, they are highly likely to be anomalies.

2. ANN WITH KERAS

1. ANNs have the ability to learn and model non-linear and complex relationships, which is really important because in real-life, many of the relationships between inputs and outputs are non-linear as well as complex.

2. ANNs can generalize — After learning from the initial inputs and their relationships, it can infer unseen relationships on unseen data as well, thus making the model generalize and predict on unseen data.

But associated with Keras, makes it **fast at execution of ideas. It has a simple and highly modular interface**, which makes it easier to create even complex neural network models.

3. K NEAREST NEIGHBOR ALGORITHM

The k-nearest neighbours (KNN) algorithm is a simple, easy-to-implement supervised machine learning algorithm that can be used to solve both classification and regression problems.

1. There's no need to build a model, tune several parameters, or make additional assumptions.

2. The algorithm is versatile. It can be used for classification, regression, and search (as we will see in the next section).

4. XG-BOOST CLASSIFIER ALGORITHM

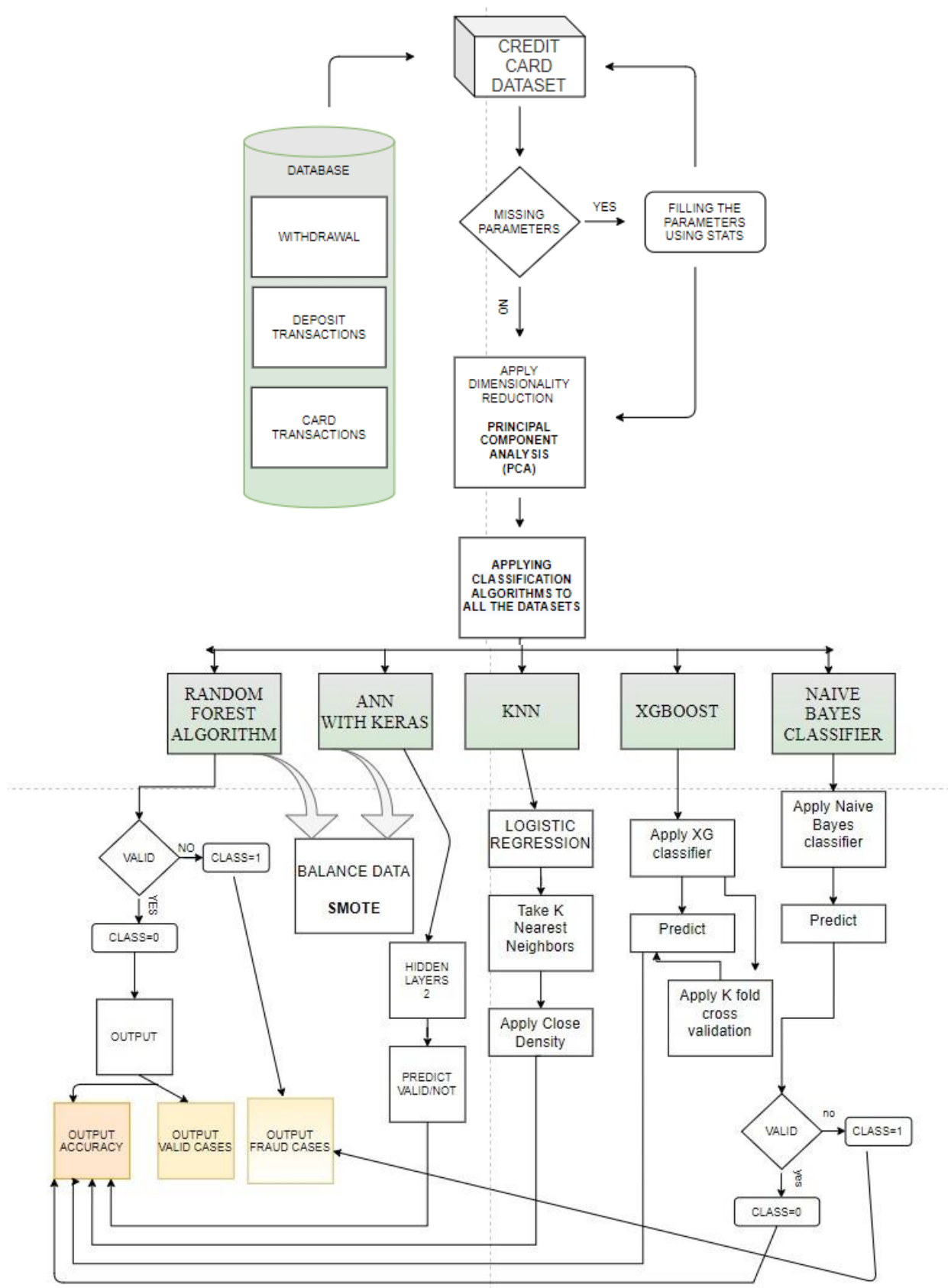
XGBoost is an optimized distributed gradient boosting library designed to be highly efficient, flexible and portable. It implements machine learning algorithms under the Gradient Boosting framework. XGBoost provides a parallel tree boosting (also known as GBDT, GBM) that solve many data science problems in a fast and accurate way.

5. NAIVE BAYES CLASSIFIER ALGORITHM

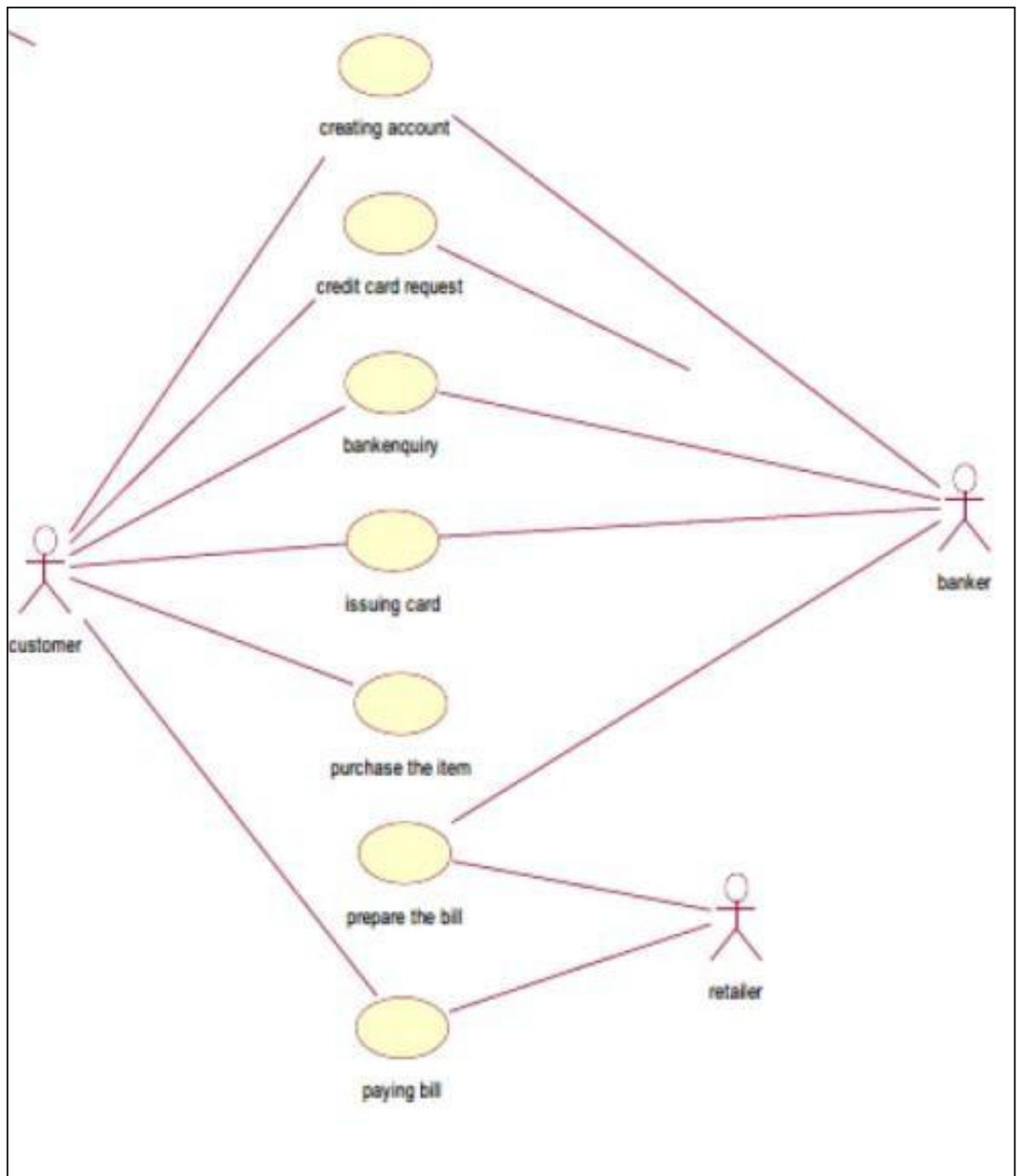
- Naïve Bayes algorithm is a supervised learning algorithm, which is based on **Bayes theorem** and used for solving classification problems.
- It is mainly used in *text classification* that includes a high-dimensional training dataset.
- Naïve Bayes Classifier is one of the simple and most effective Classification algorithms which helps in building the fast machine learning models that can make quick predictions.

It is a probabilistic classifier, which means it predicts on the basis of the probability of an object.

4.2 DETAILED MODEL(ARCHITECTURE) DIAGRAM



4.3. USE CASE DIAGRAM



The Automation system use cases are:

Creating Account: Used to create an account.

Credit card request: Used to send the request to credit card.

Bank Enquiry: Used to get the bank enquiry like pin code to verify your user account.

Issuing card: Used to issuing the card to machine.

Purchase the item: Used to list out the purchase details in shop.

Prepare the bill: Used to issuing the bill for the purchased item.

paying bill: Used to transaction of money to paying the bill.

ACTORS INVOLVED

Customer/user: The person who order for the item.

Banker: The person to check the account details.

Retailer: The person to preparing the bills.

USE-CASE NAME: PURCHASE PRODUCT

Customer purchases items from ecommerce site then proceeds to the site's secure checkout area.

USE-CASE NAME: AUTHORIZATION REQUEST

Credit card processor collects billing information from the customer via a secure connection.

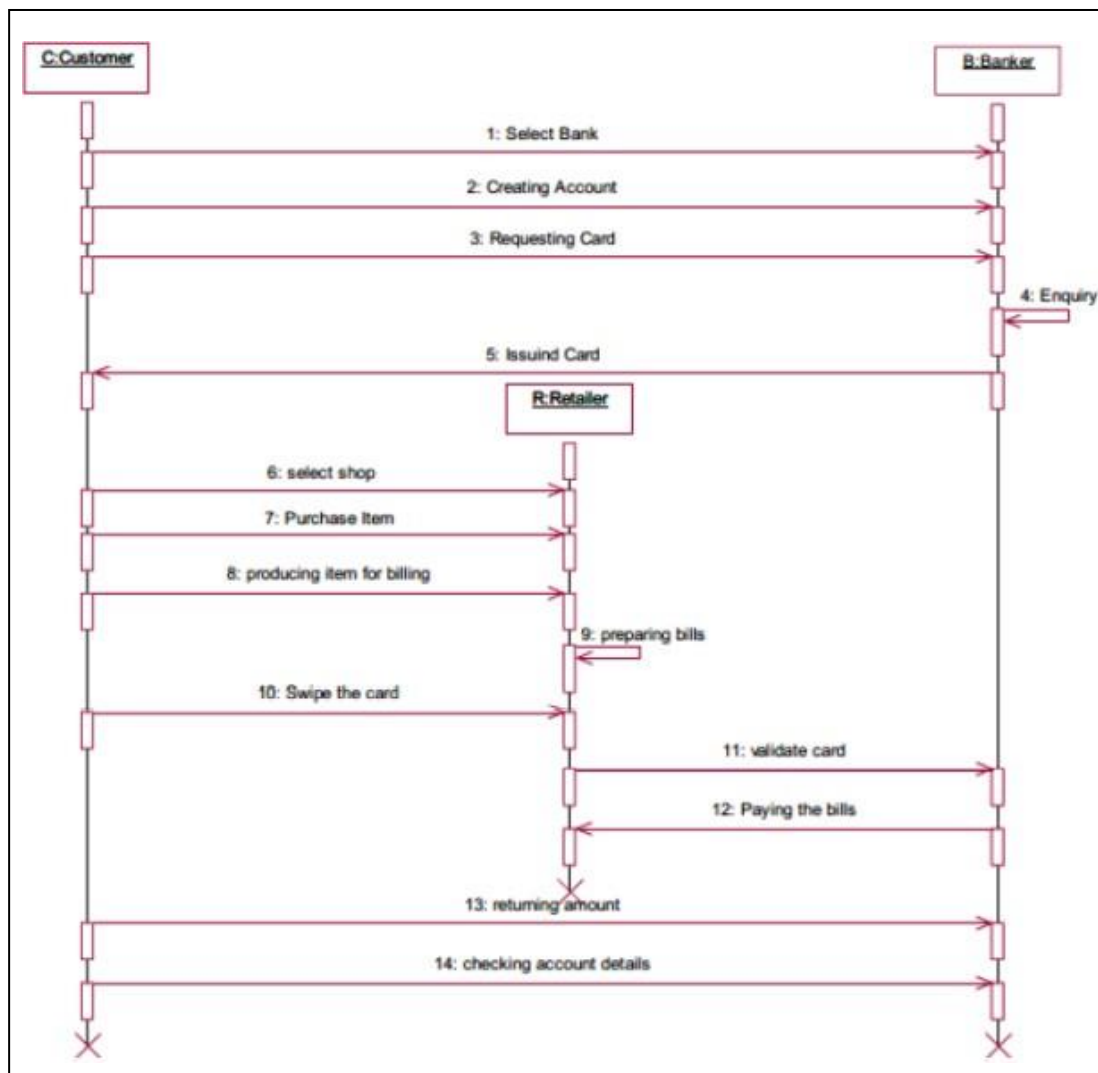
USE-CASE NAME: AUTHORIZATION RESPONSE

Billing information is verified and the transaction is completed by the credit card issuer.

USE-CASE NAME: PAYMENT APPROVAL

The transaction details are recorded by the credit card processor and results are securely relayed to the merchant. Merchant's site receives transaction result and does appropriate actions (e.g. saves the order & shows message).

4.4. INTERACTION DIAGRAM



- A sequence diagram represents the sequence and interactions of a given USE-CASE or scenario. Sequence diagrams can capture most of the information about the system.
- Most object to object interactions and operations are considered events and events include signals, inputs, decisions, interrupts, transitions and actions to or from users or external devices.
- An event also is considered to be any action by an object that sends information.
- The event line represents a message sent from one object to another, in which the “form” object is requesting an operation be performed by the “to” object.
- The “to” object performs the operation using a method that the class contains.
- It is also represented by the order in which things occur and how the objects in the system send message to one another.
- The sequence diagram for each USE-CASE that exists when a user administrator, check status and new registration about passport automation system are given.

5. IMPLEMENTATION

5.1. SOURCE CODE:

```
# XGBoost

# Importing the libraries
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd

# Importing the dataset
dataset = pd.read_csv('creditcard.csv')
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, -1].values

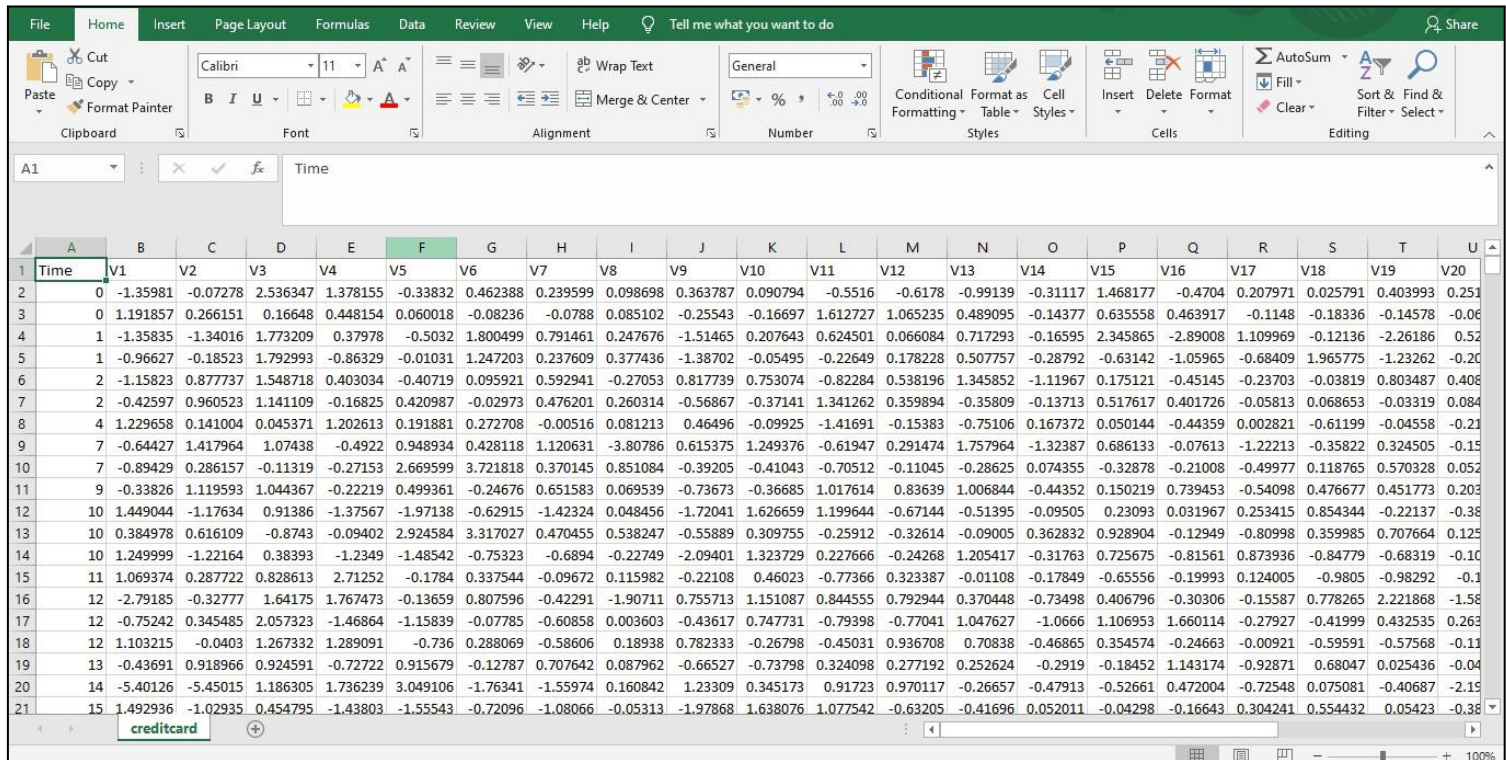
# Splitting the dataset into the Training set and Test set
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state
= 0)

# Training XGBoost on the Training set
from xgboost import XGBClassifier
classifier = XGBClassifier()
classifier.fit(X_train, y_train)

# Making the Confusion Matrix
from sklearn.metrics import confusion_matrix, accuracy_score
y_pred = classifier.predict(X_test)
cm = confusion_matrix(y_test, y_pred)
print(cm)
accuracy_score(y_test, y_pred)

# Applying k-Fold Cross Validation
from sklearn.model_selection import cross_val_score
accuracies = cross_val_score(estimator = classifier, X = X_train, y = y_train, cv =
10)
print("Accuracy: {:.2f} %".format(accuracies.mean()*100))
print("Standard Deviation: {:.2f} %".format(accuracies.std()*100))
```

5.2. TEST CASES:



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
1	Time	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19	V20
2	0	-1.35981	-0.07278	2.536347	1.378155	-0.33832	0.462388	0.239599	0.098698	0.363787	0.090794	-0.5516	-0.6178	-0.99139	-0.31117	1.468177	-0.4704	0.207971	0.025791	0.403993	0.251
3	0	1.191857	0.266151	0.16648	0.448154	0.060018	-0.08236	-0.0788	0.085102	-0.25543	-0.16697	1.612727	1.065235	0.489095	-0.14377	0.635558	0.463917	-0.1148	-0.18336	-0.14578	-0.06
4	1	-1.35835	-1.34016	1.773209	0.37978	-0.5032	1.800499	0.791461	0.247676	-1.51465	0.207643	0.624501	0.066084	0.717293	-0.16595	2.345865	-2.89008	1.109969	-0.12136	-2.26186	0.52
5	1	-0.96627	-0.18523	1.792993	-0.86329	-0.01031	1.247203	0.237609	0.377436	-1.38702	-0.05495	-0.22649	0.178228	0.507757	-0.28792	-0.63142	-1.05965	-0.68409	1.965775	-1.23262	-0.20
6	2	-1.15823	0.877737	1.548718	0.403034	-0.40719	0.095921	0.592941	-0.27053	0.817739	0.753074	-0.82284	0.538196	1.345852	-1.11967	0.175121	-0.45145	-0.23703	-0.03819	0.803487	0.408
7	2	-0.42597	0.960523	1.141109	-0.16825	0.420987	-0.02973	0.476201	0.260314	-0.56867	-0.37141	1.341262	0.359894	-0.35809	-0.13713	0.517617	0.401726	-0.05813	0.068653	-0.03319	0.084
8	4	1.229658	0.141004	0.045371	1.202613	0.191881	0.272708	-0.00516	0.081213	0.46496	-0.09925	-1.41691	-0.15383	-0.75106	0.167372	0.050144	-0.44359	0.002821	-0.61199	-0.04558	-0.21
9	7	-0.64427	1.417964	1.07438	-0.4922	0.948934	0.428118	1.120631	-3.80786	0.615375	1.249376	-0.61947	0.291474	1.757964	-1.32387	0.686133	-0.07613	-1.22213	-0.35822	0.324505	-0.15
10	7	-0.89429	0.286157	-0.11319	-0.27153	2.669599	3.721818	0.370145	0.851084	-0.39205	-0.41043	-0.70512	-0.11045	-0.28625	0.074355	-0.32878	-0.21008	-0.49977	0.118765	0.570328	0.052
11	9	-0.33826	1.119593	1.044367	-0.22219	0.499361	-0.24676	0.651583	0.069539	-0.73673	-0.36685	1.017614	0.83639	1.006844	-0.44352	0.150219	0.739453	-0.54098	0.476677	0.451773	0.203
12	10	1.449044	-1.17634	0.91386	-1.37567	-1.97138	-0.62915	-1.42324	0.048456	-1.72041	1.626659	1.199644	-0.67144	-0.51395	-0.09505	0.23093	0.031967	0.253415	0.854344	-0.22137	-0.38
13	10	0.384978	0.616109	-0.8743	-0.09402	2.924584	3.317027	0.470455	0.538247	-0.55889	0.309755	-0.25912	-0.32614	-0.09005	0.362832	0.928904	-0.12949	-0.80998	0.359985	0.707664	0.125
14	10	1.249999	-1.22164	0.38393	-1.2349	-1.48542	-0.75323	-0.6894	-0.22749	-2.09401	1.323729	0.227666	-0.24268	1.205417	-0.31763	0.725675	-0.81561	0.873936	-0.84779	-0.68319	-0.10
15	11	1.069374	0.287722	0.828613	2.71252	-0.1784	0.337544	-0.09672	0.115982	-0.22108	0.46023	-0.77366	0.323387	-0.01108	-0.17849	-0.65556	-0.19993	0.124005	-0.9805	-0.98292	-0.1
16	12	-2.79185	-0.32777	1.64175	1.767473	-0.13659	0.807596	-0.42291	-1.90711	0.755713	1.151087	0.844555	0.792944	0.370448	-0.73498	0.406796	-0.30306	-0.15587	0.778265	2.221868	-1.58
17	12	-0.75242	0.345485	2.057323	-1.46864	-1.15839	-0.07785	-0.60858	0.003603	-0.43617	0.747731	-0.79398	-0.77041	1.047627	-1.0666	1.106953	1.660114	-0.27927	-0.41999	0.432535	0.263
18	12	1.103215	-0.0403	1.267332	1.289091	-0.736	0.288069	-0.58606	0.18938	0.782333	-0.26798	-0.45031	0.936708	0.70838	-0.46865	0.354574	-0.24663	-0.00921	-0.59591	-0.57568	-0.11
19	13	-0.43691	0.918966	0.924591	-0.72722	0.915679	-0.12787	0.707642	0.087962	-0.66527	-0.73798	0.324098	0.277192	0.252624	-0.2919	-0.18452	1.143174	-0.92871	0.68047	0.025436	-0.04
20	14	-5.40126	-5.45015	1.186305	1.736239	3.049106	-1.76341	-1.55974	0.160842	1.23309	0.345173	0.91723	0.970117	-0.26657	-0.47913	-0.52661	0.472004	-0.72548	0.075081	-0.40687	-2.15
21	15	1.492936	-1.02935	0.454795	-1.43803	-1.55543	-0.72096	-1.08066	-0.05313	-1.97868	1.638076	1.077542	-0.63205	-0.41696	0.052011	-0.04298	-0.16643	0.304241	0.554432	0.05423	-0.38

DATA SET DESCRIPTION

Name: Credit card dataset

It is important that credit card companies are able to recognize fraudulent credit card transactions so that customers are not charged for items that did not purchase.

Attributes (Columns):

Time: Number of seconds elapsed between this transaction and the first transaction in the dataset.

V1 – V28: May be result of a PCA Dimensionality reduction to protect user identities and sensitive features (v1-v28).

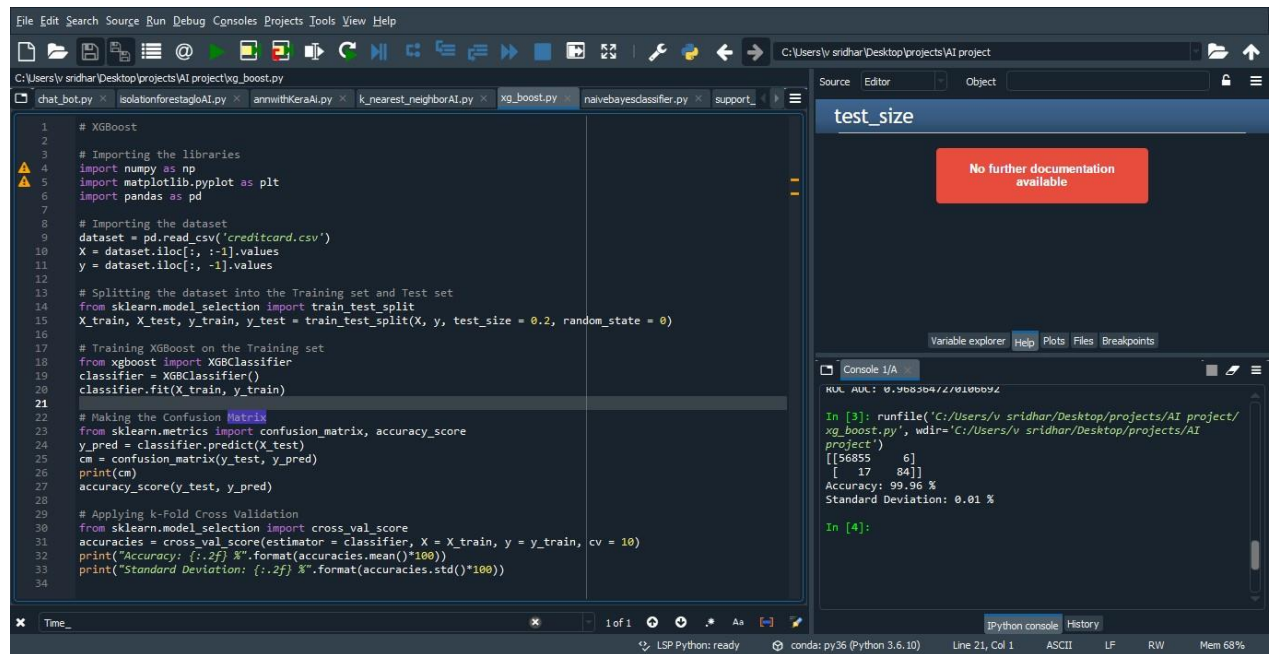
Amount: Transaction amount

Class and Class labels: 1 for fraudulent transactions, 0 otherwise

Dataset Link: <https://www.kaggle.com/mlg-ulb/creditcardfraud>

6. OUTPUT AND PERFORMANCE ANALYSIS

6.1. EXECUTION SNAPSHOT



```
1 # XGBoost
2
3 # Importing the libraries
4 import numpy as np
5 import matplotlib.pyplot as plt
6 import pandas as pd
7
8 # Importing the dataset
9 dataset = pd.read_csv('creditcard.csv')
10 X = dataset.iloc[:, :-1].values
11 y = dataset.iloc[:, -1].values
12
13 # Splitting the dataset into the Training set and Test set
14 from sklearn.model_selection import train_test_split
15 X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2, random_state = 0)
16
17 # Training XGBoost on the Training set
18 from xgboost import XGBClassifier
19 classifier = XGBClassifier()
20 classifier.fit(X_train, y_train)
21
22 # Making the Confusion Matrix
23 from sklearn.metrics import confusion_matrix, accuracy_score
24 y_pred = classifier.predict(X_test)
25 cm = confusion_matrix(y_test, y_pred)
26 print(cm)
27 accuracy_score(y_test, y_pred)
28
29 # Applying k-Fold Cross Validation
30 from sklearn.model_selection import cross_val_score
31 accuracies = cross_val_score(estimator = classifier, X = X_train, y = y_train, cv = 10)
32 print("Accuracy: {:.2f} %".format(accuracies.mean()*100))
33 print("Standard Deviation: {:.2f} %".format(accuracies.std()*100))
34
```

test_size

No further documentation available

Variable explorer | Help | Plots | Files | Breakpoints

Console 1/A

KUL AUL: 0.308304/2/010004

```
In [3]: runfile('C:/Users/v sridhar/Desktop/projects/AI project/
xg_boost.py', wdir='C:/Users/v sridhar/Desktop/projects/AI
project')
[[56855    6]
 [ 17   84]]
Accuracy: 99.96 %
Standard Deviation: 0.01 %

In [4]:
```

Time_

1 of 1

LSP Python: ready

conda: py36 (Python 3.6.10)

Line 21, Col 1

ASCII

LF

RW

Mem 68%

6.2 OUTPUT:

[[56855 6]

[17 84]]

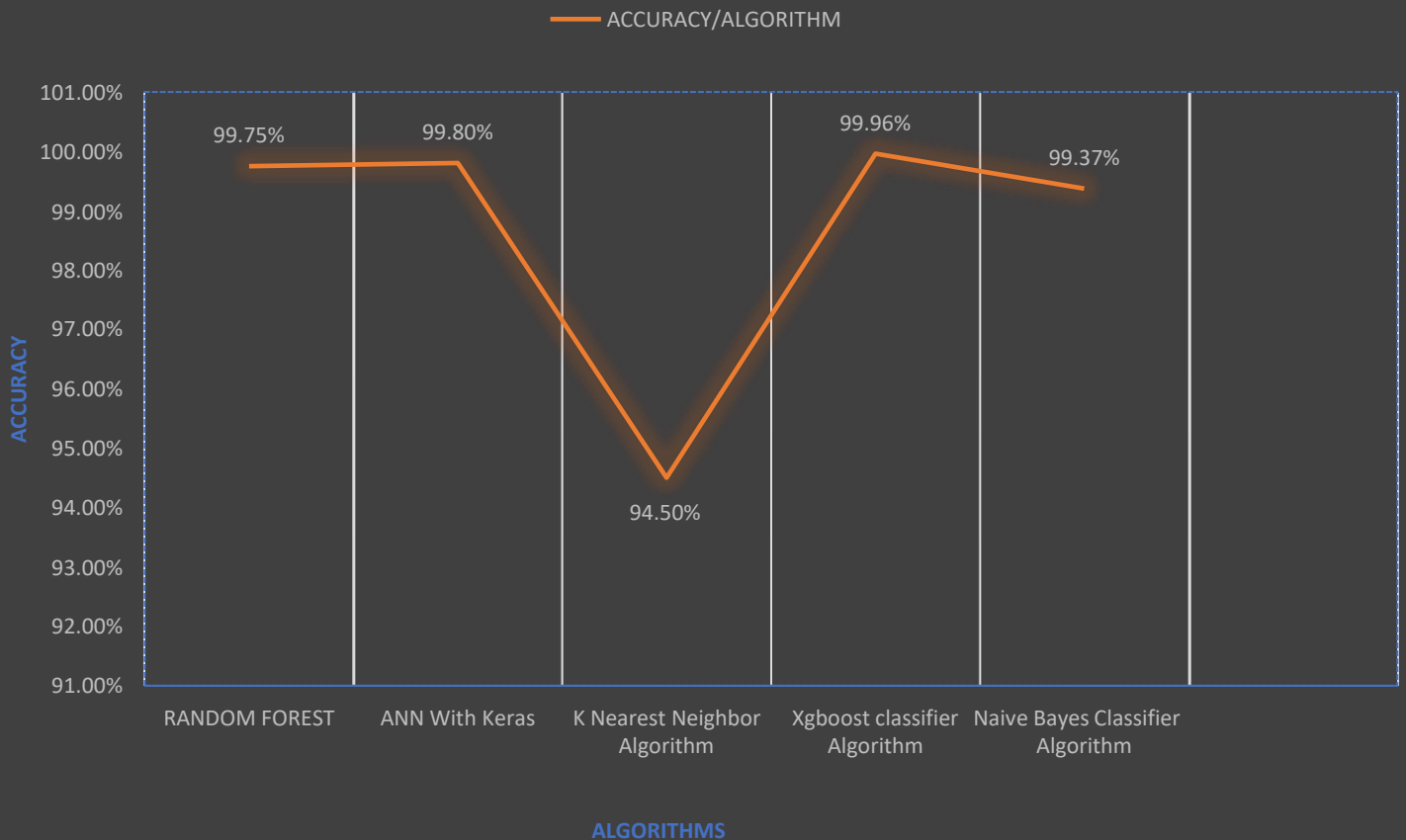
Accuracy: 99.96 %

Standard Deviation: 0.01 %

6.3 ACCURACY COMPARISION

ALGORITHM USED:	ACCURACY:
Random Forest	99.75%
ANN with Keras	99.8%
K Nearest Neighbour Algorithm	94.5%
XGBoost Classifier Algorithm	99.96%
Naïve Bayes Classifier Algorithm	99.37%

ACCURACY COMPARISON ACROSS ALGOITHMS



7. CONCLUSION AND FUTURE DIRECTIONS:

Fraud detection is an important part of the modern finance industry. In this project, we have investigated the current practices in financial fraud detection using intelligent approaches, both statistical and computational. Though their performance differed, each technique was shown to be reasonably capable at detecting various forms of financial fraud. In particular, the ability of CI methods such as neural networks and support vector machines to learn and adapt to new situations is highly effective at defeating the evolving tactics of fraudsters. There are still many aspects of intelligent fraud detection that have not yet been the subject of research. Some types of fraud, as well as some soft computing methods, have been superficially explored but require future study to be completely understood. There is also the opportunity to examine the performance of existing methods by using customization or tuning, as well as the potential to study cost benefit analysis of computational fraud detection. Finally, further research into the differences between each type of financial fraud could lead to a generic framework which would greatly enhance the scope of intelligent detection methods for this problem domain. So, **XGBOOST Classifier Algorithm has got the highest accuracy (99.96%)**, So we recommend to use XGBOOST classifier only.

8. REFERENCES:

- “Credit Card Fraud Identification Using Artificial Neural Networks”, Chandrashekar Mishra, Dharmendra Lal Gupta, Raghuraj Singh (07, July, 2017).
- “CREDIT CARD FRAUD DETECTION USING SELF ORGANIZING MAPS”, Vladimir ZASLAVSKY and STRIZHAK .
- “Fraud Detection of Credit Card Payment System by Genetic Algorithm”, K.RamaKalyani, D.UmaDevi (7, July-2012).
- “Application of Credit Card Fraud Detection: Based on Bagging Ensemble Classifier”, Masoumeh Zareapoor, Pourya Shamsolmoali Year:2015.
- “A Neural Network Based Model for Detecting Irregularities in E-Banking Transactions”, J.A Adeyiga, J.O. Ezike, A. Omotosho & W. Amakulor (December, 2011).
- “Credit Card Fraud Detection Using Neural Networks”, Divya Murli, Shailesh Jami, Devika Jog, Sreesha Nath (March-April 2014)
- “Recognition of fraud in online banking by using confirmatory learning in neural network”, Alireza Pouramirarsalani, Majid Khalilian, Alireza Nikravanshalmani (8, August 2017).
- “A Neural Network Based Model for Detecting Irregularities in E-Banking Transactions”, J.A Adeyiga, J.O. Ezike, A. Omotosho & W. Amakulor (December, 2011).