Scientific Computing, Modeling & Simulation
Savitribai Phule Pune University

Master of Technology (M.Tech.)
Programme in Modeling and Simulation

Internship Project Report

Credit Card Fraud Detection

Ashwini Awari, Aishwarya Bhosale
MT2103,MT2107

Academic Year 2022-23

Scientific Computing, Modeling & Simulation
Savitribai Phule Pune University

Certificate

This is certify that this report titled

Credit Card Fraud Detection

authored by

Ashwini Awari, Aishwarya Bhosale (MT2103,MT2107)

describes the project work carried out by the author under our supervision during the period from September 2022 to November 2022. This work represents the project component of the Master of Technology (M.Tech.) Programme in Modeling and Simulation at the Department of Scientific Computing, Modeling & Simulation, Savitribai Phule Pune University.

Mihir Arjunwadkar, Faculty
SCMS-SPPU, Pune, India

Dr. Arun G. Banpurkar, Head
SCMS-SPPU, Pune, India

Scientific Computing, Modeling & Simulation
Savitribai Phule Pune University

Author's Declaration

This document titled

Credit Card Fraud Detection

authored by me is an authentic report of the project work carried out by me as part of the Master of Technology (M.Tech.) Programme in Modeling and Simulation at the Department of Scientific Computing, Modeling & Simulation, Savitribai Phule Pune University. In writing this report, I have taken reasonable and adequate care to ensure that material borrowed from sources such as books, research papers, internet, etc., is acknowledged as per accepted academic norms and practices in this regard. I have read and understood the University's policy on plagiarism (`http://unipune.ac.in/administration_files/pdf/Plagiarism_Policy_University_14-5-12.pdf`).

Ashwini Awari, Aishwarya Bhosale
MT2103,MT2107

Abstract

Fraud is one of the major ethical issues in the credit card industry. Credit card frauds are easy and friendly targets. E-commerce and many other online sites have increased the online payment modes, increasing the risk for online frauds. It is vital that credit card companies are able to identify fraudulent credit card transactions so that customers are not charged for items that they did not purchase. Such problems can be tackled with Data Science and its importance, along with Machine Learning, cannot be overstated. This project intends to illustrate the modelling of a data set using machine learning with Credit Card Fraud Detection.

The Credit Card Fraud Detection Problem includes modelling past credit card transactions with the data of the ones that turned out to be fraud. This model is then used to recognize whether a new transaction is fraudulent or not. Our objective here is to detect 100 percent fraudulent transactions while minimizing the incorrect fraud classifications. Credit Card Fraud Detection is a typical example classification and recognize whether a new transaction is fraudulent or not.

Acknowledgements

We would like to sincerely thank Kaggle platform which provided free dataset for our project . This dataset is collected and analysed during a research collaboration with Worldline and the Machine Learning group of Univrsite Libre de bruxelles(USB) on big data mining and fraud detection who collects credit card transaction data for providing with sample reference content as well as advice . Thank you Mr. Mihir Arjunwadkar sir ,The Faculty who provided their unlimited support and guidance .

Contents

Chapter 1

Introduction

1.1 Credit card fraud

Fraud is a crime that the finance industry is committed to tackling, but it is also one that requires the combined efforts of both public and private sector to overcome. But fraud is a threat that the finance industry cannot tackle alone. According to "FRAUD THE FACTS 2019 — THE DEFINITIVE OVERVIEW OF PAYMENT INDUSTRY FRAUD", data breaches at third parties continue to be a major contribution to fraud losses. There has been a number of high-profile incidents in 2018, many targeting well-known brands, where customer data was stolen. Whether it's at a retailer, utility company, transport provider or elsewhere, the theft of personal and financial data can both directly lead to fraud losses or be used by criminals as part of their scams. The data can be used for months and even years after the breach takes place. Unauthorised financial fraud losses across payment cards, remote banking and cheques totalled £844.8 million in 2018, an increase of 16 per cent compared to 2017. Banks and card companies prevented £1.66 billion in unauthorised fraud in 2018. This represents incidents that were detected and prevented by firms and is equivalent to £2 in every £3 of attempted fraud being stopped. In addition to this, in 2018 UK Finance members reported 84,624 incidents of authorised push payment scams with gross losses of £354.3 million.

1.1.1 Types of fraud

In order to avoid these overheads and depending on the type of fraud committed, diverse solutions can be implemented.

Credit card fraud is unauthorized use of a payment card such as a credit card or debit card or card information to make transactions related to purchases. Removing funds from the cardholder's account also comes under fraud. credit card fraud is an act of criminal dishonesty. Credit card fraud is an important issue and has considerable cost for banks and card issuer companies. Financial organizations try to prevent account misuse using different security solutions. The more complex the security solutions are, the more sophisticated fraudsters get i.e. fraudsters change their methods over time. Therefore it is crucial to improve fraud detection methods along with security modules which try to prevent fraud.

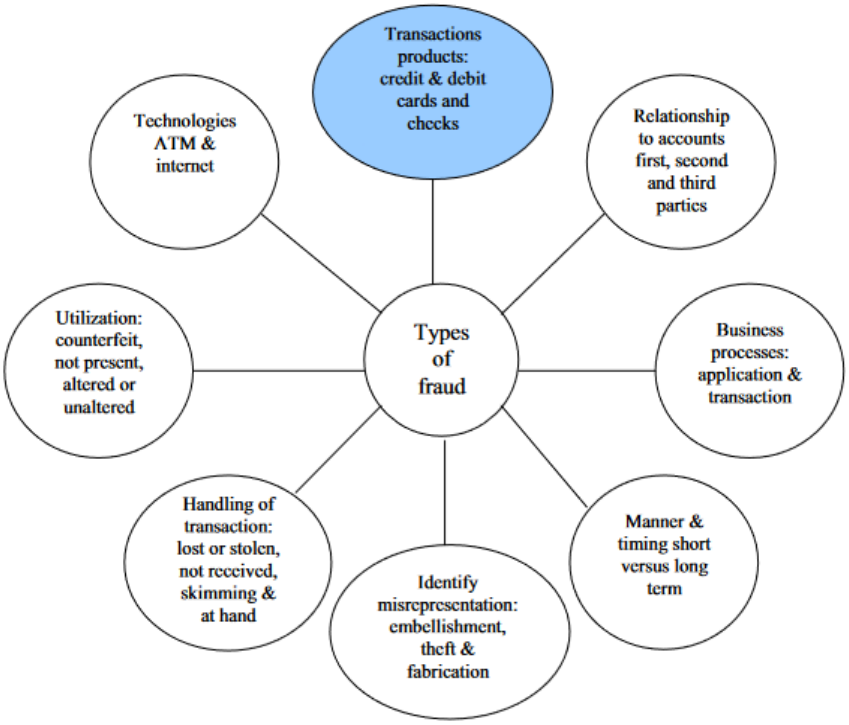

Figure 1.1: Types of Fraud

1.1.2 How does credit card fraud occur

Credit card fraud happens

- When consumers give their credit card number to unfamiliar individuals, when cards are lost or stolen.
- When mail is diverted from the intended recipient and taken by criminals.
- When employees of a business copy the cards or card numbers of a cardholder.

1.1.3 Types of Credit card fraud:

1.1.4 Biggest credit card frauds in history:

Two of the biggest credit card frauds.

1. £17 million stolen - the biggest UK credit card fraud:

In the mid 2000's, a gang of international fraudsters managed to steal the details of over 32,000 credit cards. They used this information to create clone credit cards and scam at least £17million over a period of several years. The gang members lived a life of luxury on the money: purchasing English mansions, Spanish villas, and a portfolio of other London properties; enjoying first class travel and five-star holidays; and spending a small fortune on designer clothes and shoes. The scam was masterminded by Russian and eastern European criminals operating out of London with a sophisticated money laundering system, shifting money from the UK to Poland to Estonia, Russia, the United States and the Virgin Islands.

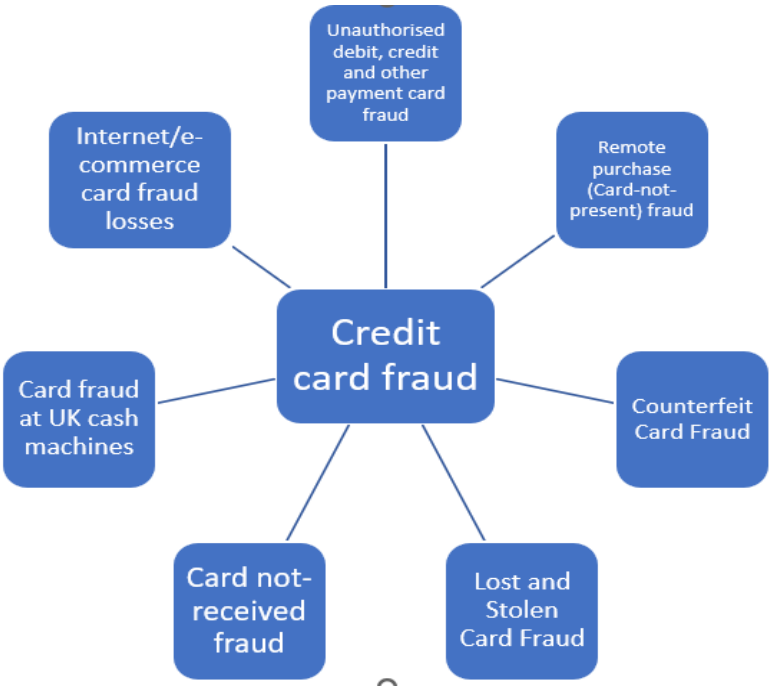

Figure 1.2: Credit card fraud

It began to unravel when a gang member was caught, by chance, during a routine anti-terror check by a transport police officer who grew suspicious when they found forty mobile phone top-up cards. This began an investigation that finally put to a stop to the scam in 2007. Five men were jailed with sentences up to five and half years.

2. The biggest credit card scam ever - 200 million Dollar

America likes to do things big, and their credit card scams are no exception. The biggest card fraud to date was committed by a gang of eighteen criminals from New York, who managed to steal 200 million Dollar before being stopped.

The story wouldn't be out place in a Hollywood movie, with the fraudsters using their ill-gotten gains to live the high life: buying luxury cars; holidays; and millions of dollars worth of gold.

A three-step scam

This scam was more elaborate than the simple card cloning techniques used in the British and Australian frauds.

Instead, the American fraudsters created thousands of false identities with addresses across the US and in eight countries around the world.

- The card fraudsters created all the information and documents needed to make false profiles with America's major credit agencies. Black market businesses were employed to provide fake credit histories for these false profiles.
- With perfect credit scores for their fake identities, they would apply for large loans and credit cards with high limits.
- The money from these cards was spent in a network of sham companies and businesses in on the scam, which laundered the cash. Tens of millions of dollars were wired overseas to Pakistan, India, the United Arab Emirates, Canada, Romania, China and Japan.

The FBI shut the scam down in 2013 and 18 people were jailed on charges of defrauding both banks and the United States. The longest sentence handed down was 30 years.

1.1.5 Prevention of payment card fraud

Card information is stored in a number of formats. Card numbers – formally the Primary Account Number (PAN) – are often embossed or imprinted on the card, and a magnetic stripe on the back contains the data in a machine-readable format. Fields can vary, but the most common include the Name of the cardholder; Card number; Expiration date; and Verification CVV code.

Several technologies have been used to prevent fraud from happening, such as the Address Verification System (AVS), Chip and Pin verification and Card Verification Code (CVV). However, even these advanced systems are prune to fail. The development of fraud detection methods is thus of crucial importance.

1.1.6 How to detect credit card fraud using technology

- Artificial and Computational intelligence
- Machine learning

Fraud detection technique:

Credit-card fraud leads to billions of dollars in losses for online merchants. With the development of machine learning algorithms, researchers have been finding increasingly sophisticated ways to detect fraud. The different types of fraud, such as bankruptcy fraud, counterfeit fraud, theft fraud, application fraud and behavioural fraud can be detect using measures such as pairwise matching, decision trees, clustering techniques, neural networks, and genetic algorithms.

- Decision tree

  The idea of a similarity tree using decision tree logic has been developed. A similarity tree is defined recursively: nodes are labelled with attribute names, edges are labelled with values of attributes that satisfy some condition and 'leaves' that contain an intensity factor which is defined as the ratio of the number of transactions that satisfy these condition(s) over the total number of legitimate transaction in the behaviour (Kokkinaki, 1997). The advantage of the method that is suggested is that it is easy to implement, to understand and to display. However, a disadvantage of this system is the requirements to check each transaction one by one.

- Genetic algorithms and other algorithm

  Algorithms are often recommended as predictive methods as a means of detecting fraud. One algorithm that has been suggested by Bentley et al. (2000) is based on genetic programming in order to establish logic rules capable of classifying credit card transactions into suspicious and non-suspicious classes. Basically, this method follows the scoring process. In the experiment described in their study, the database was made of 4,000 transactions with 62 fields. As for the similarity tree, training and testing samples were employed. Different types of rules were tested with the different fields. The best rule is the one with the highest predictability. Their method has proven results for real home insurance data and could be one efficient method against credit card fraud.

  Chan et al. (1999) also developed an algorithm to predict suspect behaviour. The originality of their research is that the model is evaluated and rated by a cost model, whereas

other studies use evaluation based on their prediction rate/the true positive rate and the error rate/the false negative rate.

Wheeler Aitken (2000) developed the idea of combining algorithms to maximize the power of prediction. In their article, they present different algorithms: diagnostic algorithms, diagnostic resolution strategies, probabilistic curve algorithms, best match algorithms, negative selection algorithms, and density selection algorithms. They conclude from their investigation that neighbourhood-based and probabilistic algorithms have been shown to be appropriate techniques for classification, and may be further enhanced using additional diagnostic algorithms for decision-making in borderlines cases, and for calculating confidence and relative risk measures.

- Clustering techniques Bolton Hand (2002) suggest two clustering techniques for behavioural fraud. The peer group analysis is a system that allows identifying accounts that are behaving differently from others at one moment in time whereas they were behaving the same previously. Those accounts are then flagged as suspicious. Fraud analysts have then to investigate those cases.

  The hypothesis of the peer group analysis is that if accounts behave the same for a certain period of time and then one account is behaving significantly differently, this account has to be notified. Break-point analysis uses a different approach. The hypothesis is that if a change of card usage is notified on an individual basis, the account has to be investigated. In other words, based on the transactions of a single card, the break-point analysis can identify suspicious behaviour. Signals of suspicious behaviour are a sudden transaction for a high amount, and a high frequency of usage.

- Neural network Neural networks are also often recommended for fraud detection. Dorronsoro et al. (1997) developed a technically accessible on- line fraud detection system, based on a neural classifier. However, the main constraint is that data need to be clustered by type of account. Similar concepts are: Card watch (Aleskerov et al., 1997); Back-propagation of error signals (Maes et al., 2002); FDS (Ghosh Reilly, 1994); SOM (Quah and Sriganesh, 2008; Zaslavsky Strizkak, 2006); improving detection efficiency "mis-detections" (Kim Kim, 2002). Data mining tools, such as 'Clementine' allow the use of neural network technologies, which have been used in credit card fraud (Brause et al., 1999a; Brause et al., 1999b).

  Bayesian networks are also one technique to detect fraud, and have been applied to detect fraud in the telecommunications industry (Ezawa Norton,1996) and also in the credit card industry (Maes et al., 2002). Results from this technique are optimistic. However, the time constraint is one main disadvantage of such a technique, especially compared with neural networks (Maes et al., 2002). Furthermore, expert systems have also been used in credit card fraud using a rule-based expert system (Leonard, 1995).

- AIS

  AIS simulates human body immune system functionality. Human body detects non-self cells, which might be viruses, pathogens, germs, etc., by creating detector cells named lymphocytes. As this functionality is similar to what a typical fraud detection system does, AIS is used for fraud detection in some researches. AIS detects non-self cells using two basic functions in human body which generate and mature lymphocytes: Negative Selection and Clonal Selection. Detector cells (lymphocytes) are generated through random composition of protein patterns. Then they will be able to prevent any potential threat by covering many protein pat- terns randomly. In order to have self-tolerant detectors which do not react to self cells, the system declines those which do. It means that any randomly

generated detector which detects a self cell dies immediately. Right after generation, detectors are presented to self cells and only those which do not react to self cells survive. This process is called Negative Selection. After this process the detectors enter the system and in their short life-time they are expected to face any potential non-self cell and detect it. If any detector detects a non-self cell, it can live longer in order to make body vaccinated against that non-self; the process is called Clonal Selection. When a detector comes across a non-self cell and detects it, the detector is cloned through mutation. One of the clones having the highest affinity with the non-self cell is selected as memory cell and lives longer in human body. If that specific type of non-self cell enters human body again, the system will detect it using memory cells.

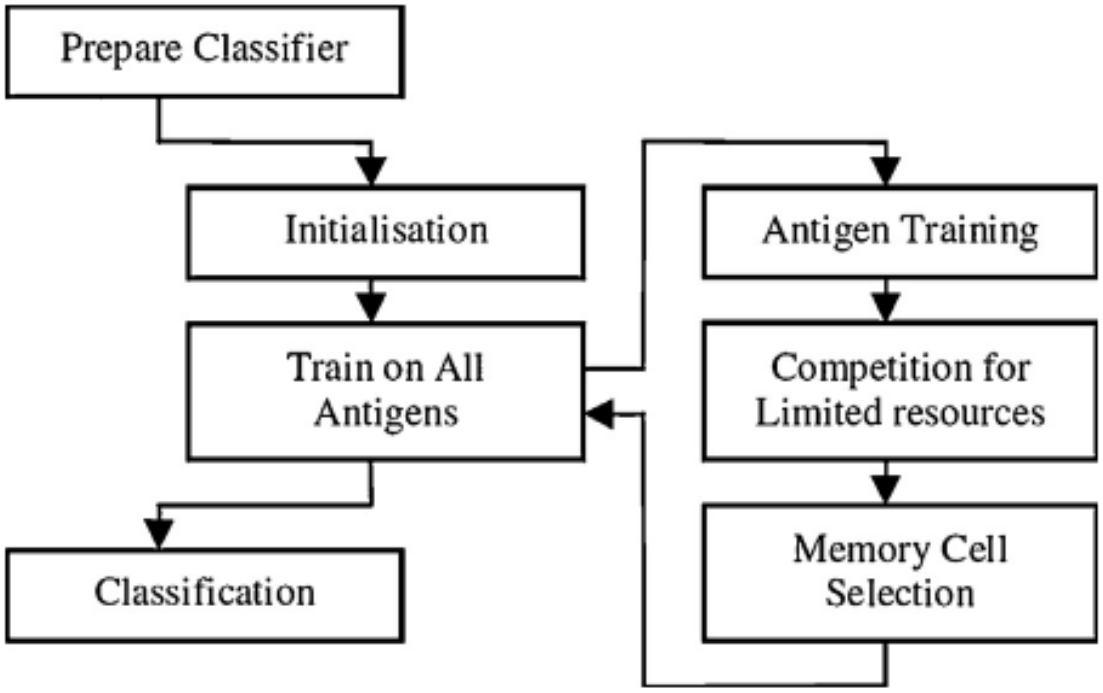

Figure 1.3: Lifecycle overview of AIS algorithm

1.2 Credit card fraud detection

1.2.1 Scope of Project:

- Modelling past credit card transactions.
- Recognize whether a new transaction is fraudulent or not.

1.2.2 Dataset:

The dataset used contains transactions made by credit cards in September 2013 by European cardholders. This dataset presents transactions that occurred in two days, where we have 492 frauds out of 284,807 transactions. The dataset is highly imbalanced; the positive class (frauds)

account for 492 and Non-fraud(0): 284315 0.9982 percentage of all transactions. The dataset has been collected and analyzed during a research collaboration of Worldline and the Machine Learning Group (http://mlg.ulb.ac.be) of ULB (Universite Libre de Bruxelles) on big data mining and fraud detection.

As the dataset was created using the PCA method, preprocessing of data has little scope. The imbalance between classes is compensated using oversampling and undersampling. The logistic regression, random forest, support vector machine, k-means are used within a cross-validation framework. Lastly, Recall and Accuracy are chosen as metrics while deducing the best classifier.

We're given features V1, V2, ... V28, that are the principal components obtained with PCA. The only features which have not been transformed with PCA are 'Time' and 'Amount'. Feature 'Time' contains the seconds elapsed between each transaction and the first transaction in the dataset. The feature 'Amount' is the transaction Amount, this feature can be used for example-dependent cost-sensitive learning. Feature 'Class' is a binary variable and it takes value 1 in case of fraudulent transaction and 0 otherwise.

1.2.3 Difficulty:

- Imbalanced data: Non-fraud(0): 284315 (0.9982 percentage) Fraud(1): 492 (0.172 percentage)

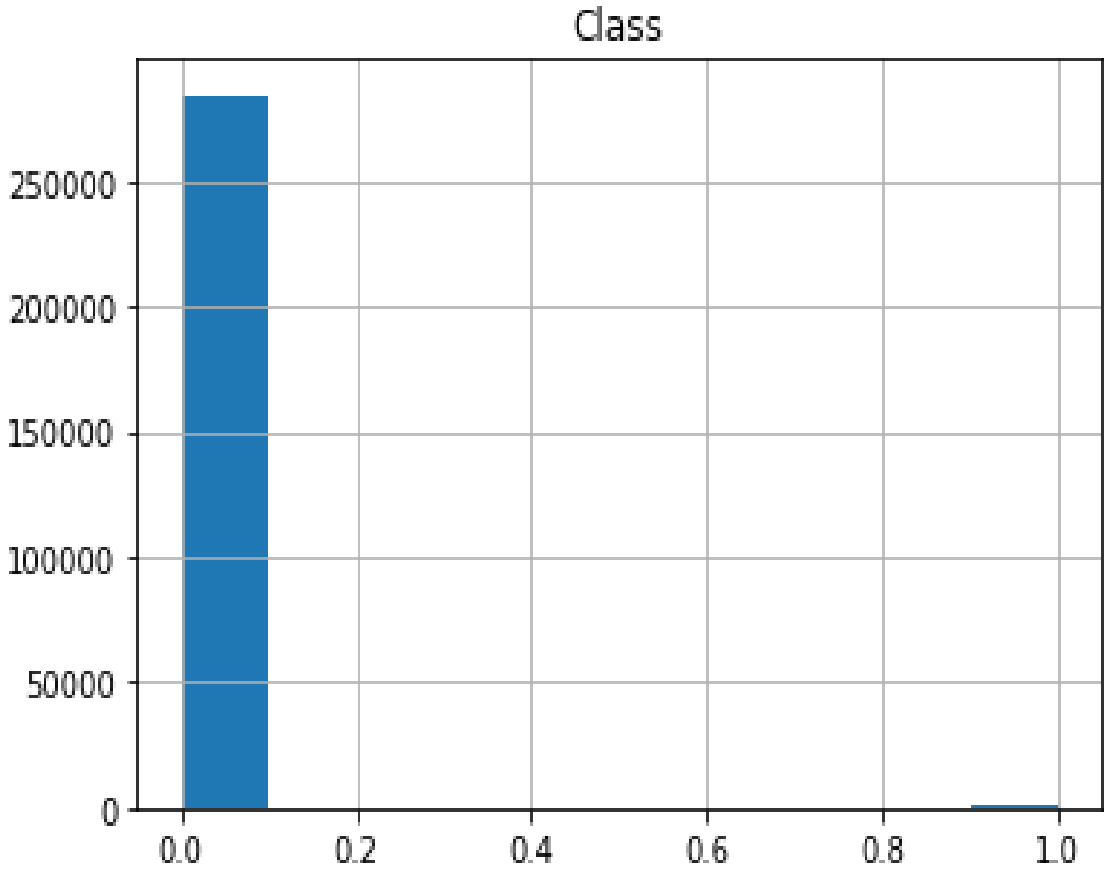

Figure 1.4: Imbalanced data

- Feature Scaling: Here 'amount' and other feature variables ranges are different Scale 'amount' by Standardization techniques

In [4]: df

Out[4]:

	V4	V5	V6	V7	V8	V9	...	V21	V22	V23	V24	V25	V26	V27	V28	Amount	Class
55	-0.338321	0.462388	0.239599	0.098698	0.363787	...	-0.018307	0.277838	-0.110474	0.066928	0.128539	-0.189115	0.133558	-0.021053	149.62	0
54	0.060018	-0.082361	-0.078803	0.085102	-0.255425	...	-0.225775	-0.638672	0.101288	-0.339846	0.167170	0.125895	-0.008983	0.014724	2.69	0
80	-0.503198	1.800499	0.791461	0.247676	-1.514654	...	0.247998	0.771679	0.909412	-0.689281	-0.327642	-0.139097	-0.055353	-0.059752	378.66	0
91	-0.010309	1.247203	0.237609	0.377436	-1.387024	...	-0.108300	0.005274	-0.190321	-1.175575	0.647376	-0.221929	0.062723	0.061458	123.50	0
34	-0.407193	0.095921	0.592941	-0.270533	0.817739	...	-0.009431	0.798278	-0.137458	0.141267	-0.206010	0.502292	0.219422	0.215153	69.99	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
56	-5.364473	-2.606837	-4.918215	7.305334	1.914428	...	0.213454	0.111864	1.014480	-0.509348	1.436807	0.250034	0.943651	0.823731	0.77	0
89	0.868229	1.058415	0.024330	0.294869	0.584800	...	0.214205	0.924384	0.012463	-1.016226	-0.606624	-0.395255	0.068472	-0.053527	24.79	0
28	2.630515	3.031260	-0.296827	0.708417	0.432454	...	0.232045	0.578229	-0.037501	0.640134	0.265745	-0.087371	0.004455	-0.026561	67.88	0
99	-0.377961	0.623708	-0.686180	0.679145	0.392087	...	0.265245	0.800049	-0.163298	0.123205	-0.569159	0.546668	0.108821	0.104533	10.00	0
71	-0.012546	-0.649617	1.577006	-0.414650	0.486180	...	0.261057	0.643078	0.376777	0.008797	-0.473649	-0.818267	-0.002415	0.013649	217.00	0

Figure 1.5: Dataframe

1.2.4 Why machine learning required?

Machine learning algorithms do not assume the logic that differentiates fraudulent transactions from non-fraudulent ones. Rather, they take maximum advantages of the transactions details and customers information. These algorithms are best suited to reveal the hidden patterns in the dataset and are therefore becoming a popular choice for solving problems like detecting credit card frauds.

1.2.5 Which are the best algorithms for credit card fraud detection?

The problem of credit card fraud detection is an example of a binary classification problem that can be solved using classification algorithms like Random Forests, Logistic Regression, Support Vector Machines, K-Nearest Neighbour, etc. You can analyse the performance of these algorithms using metrics like Recall, Precision, Accuracy, Confusion Matrix, ROC Curve, etc. and deduce which works best for your dataset.

Chapter 2

Algorithms For Credit Card Fraud Detection

2.1 Algorithms

The processing steps to detect the best algorithm are given below-

2.1.1 Algorithm steps

- Step 1: Firstly read the data set
- Step 2: Under Sampling is done on the dataset to make unbalanced data as balanced
- Step 3: following we divided the data set into two parts i.e., training dataset and testing data set
- Step 4: Then we applied feature selection for proposed models
- Step 5: Accuracy and performance metrics has been calculated to know the efficiency for different algorithms
- Step6: After that we retrieve the best algorithm based on its efficiency for data set

Logistic Regression

It is a supervised classification algorithm. In a classification problem, the target variable or output 'y' is dependent variable , which can take only discrete values for a given set of or inputs ' X' i.e. independent variable. The model builds a regression model to predict the probability that a given data entry belongs to the category numbered as '1' or the data entry belongs to the category numbered as '0'.

If resulting variable is categorical then the log of odds are used for dependent variable and also it predicts the probability of occurrence of an event by fitting data to a logistic function. Such as

$$y = e^(b0 + b1 * x)/(1 + e^(b0 + b1 * x)) \tag{2.1}$$

Where, y = predicted output b0 = intercept b1 = the coefficient for the single input value (x). Each column in the input data has an associated coefficient (a constant real value) that must be learned from the training data.

$$y = e^(b0 + b1 * x)/(1 + e^(b0 + b1 * x)) \tag{2.2}$$

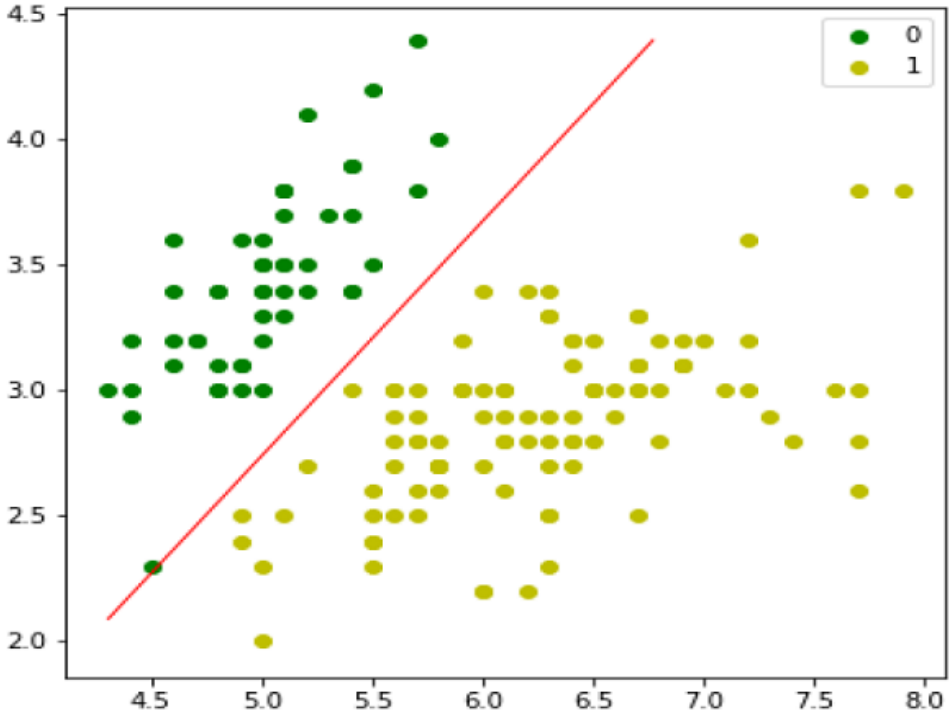

Figure 2.1: Logistic Regression classification algorithm

Decision Tree:

Decision Tree Decision tree algorithm are used to solve both regression and classification problems. Decision tree is nothing but tree like structure to solve the problem in which each leaf node corresponds to a class label and attributes are represented on the internal node of the tree. We can represent any boolean function on discrete attributes using the decision tree.

Types of decision Tree:

- 1. Categorical Variable Decision Tree: Decision Tree which has categorical target variable then it called as categorical variable decision tree.
- 2. Continuous Variable Decision Tree: Decision Tree has continuous target variable then it is called as Continuous Variable Decision Tree

In Decision Tree the major challenge is to identification of the attribute for the root node in each level. This process is known as attribute selection. We have two popular attribute selection measures:

- Information Gain When we use a node in a decision tree to partition the training instances into smaller subsets the entropy changes. Information gain is a measure of this change in entropy.
- Gini Index 1. Gini Index is a metric to measure how often a randomly chosen element would be incorrectly identified. 2. It means an attribute with lower Gini index should be preferred.

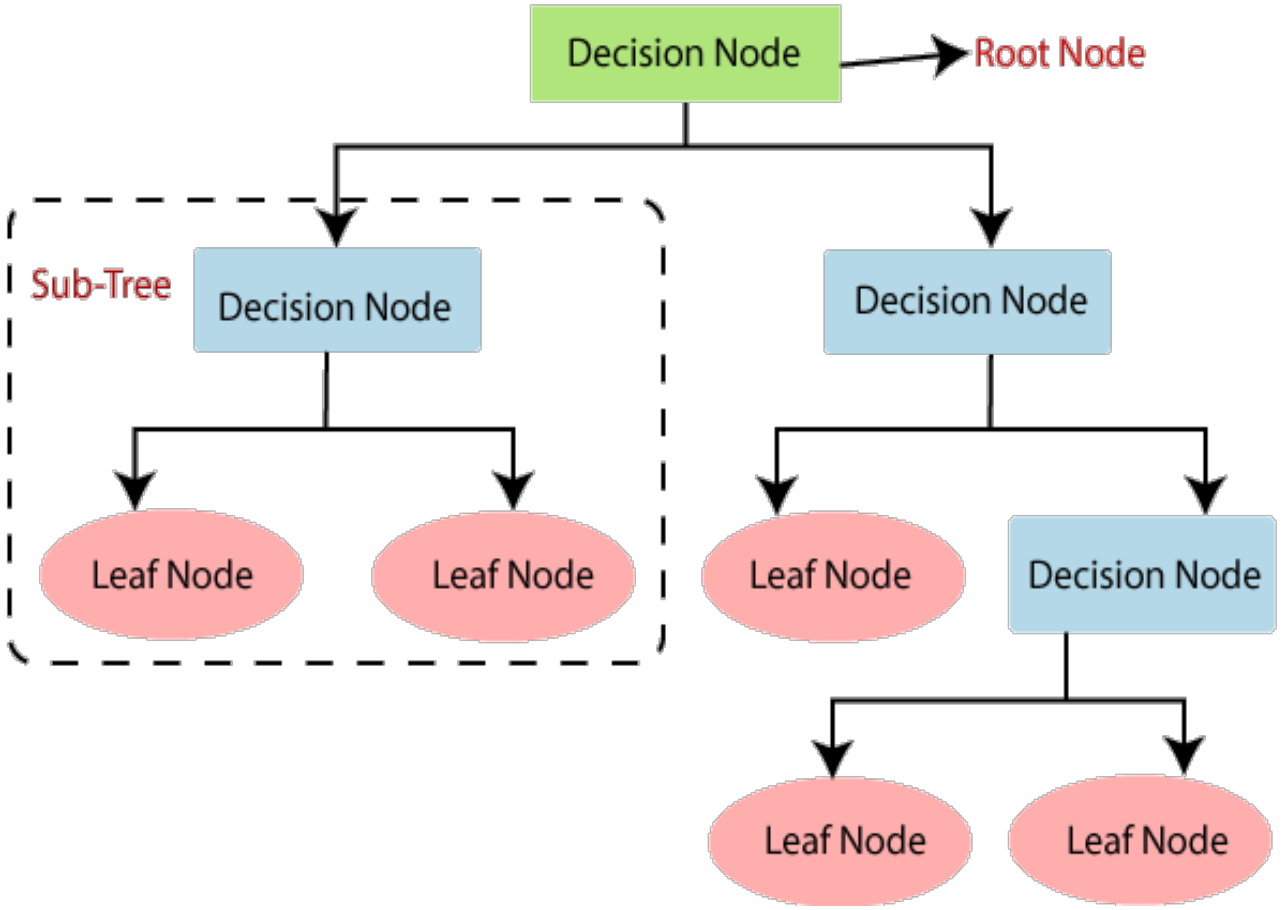

Figure 2.2: Decision tree classification algorithm

1. Information Gain When we use a node in a decision tree to partition the training instances into smaller subsets the entropy changes. Information gain is a measure of this change in entropy. 2. Gini Index • Gini Index is a metric to measure how often a randomly chosen element would be incorrectly identified. • It means an attribute with lower Gini index should be preferred.

Random forest:

It is supervised machine learning algorithm which is based on ensemble learning . Ensemble learning is nothing but an algorithm where the predictions are derived by bagging or assembling different or similar model multiple times. In a same way random forest algorithm works . It is named as "Random Forest" because it uses multiple algorithm or multiple decision trees which looks like forest of trees. The random forest algorithm can be used for both classification and regression tasks.

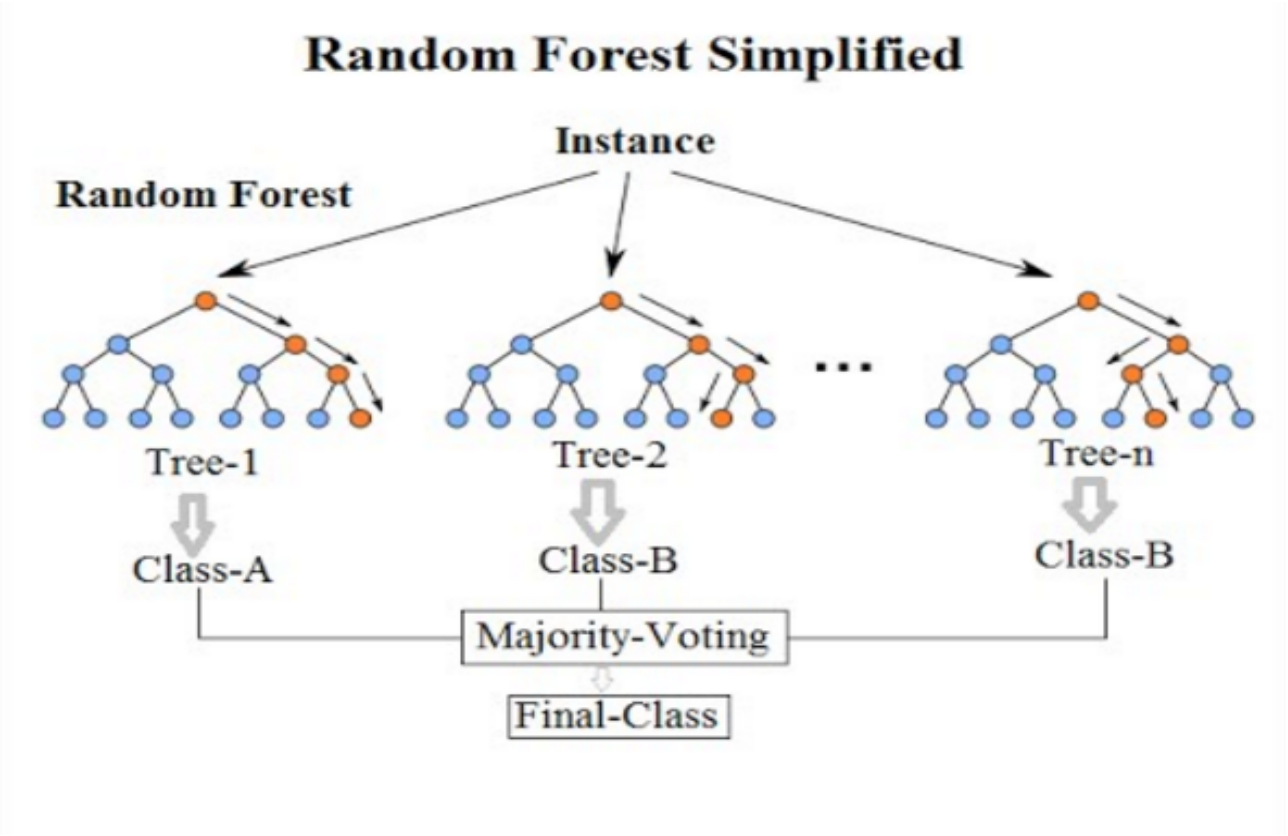

Figure 2.3: Random forest classification algorithm

Steps involved in random forest algorithm:

- Step 1: In Random forest we are taking 'n' number of random records from the data set which is having ' k' number of records.
- Step 2: for each sample we are constructing individual decision trees.
- Step 3: In that each decision tree will generate an output.
- Step 4: On the basis of Majority Voting or Averaging final output is considered for regression and classification respectively.

Support Vector Machine:

Support Vector Machine (SVM) is used for Classification(mostly) and Regression problems.SVM algorithm's goal is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future which is called as "Hyper plane". SVM chooses the extreme points/vectors that help in creating the hyper plane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.

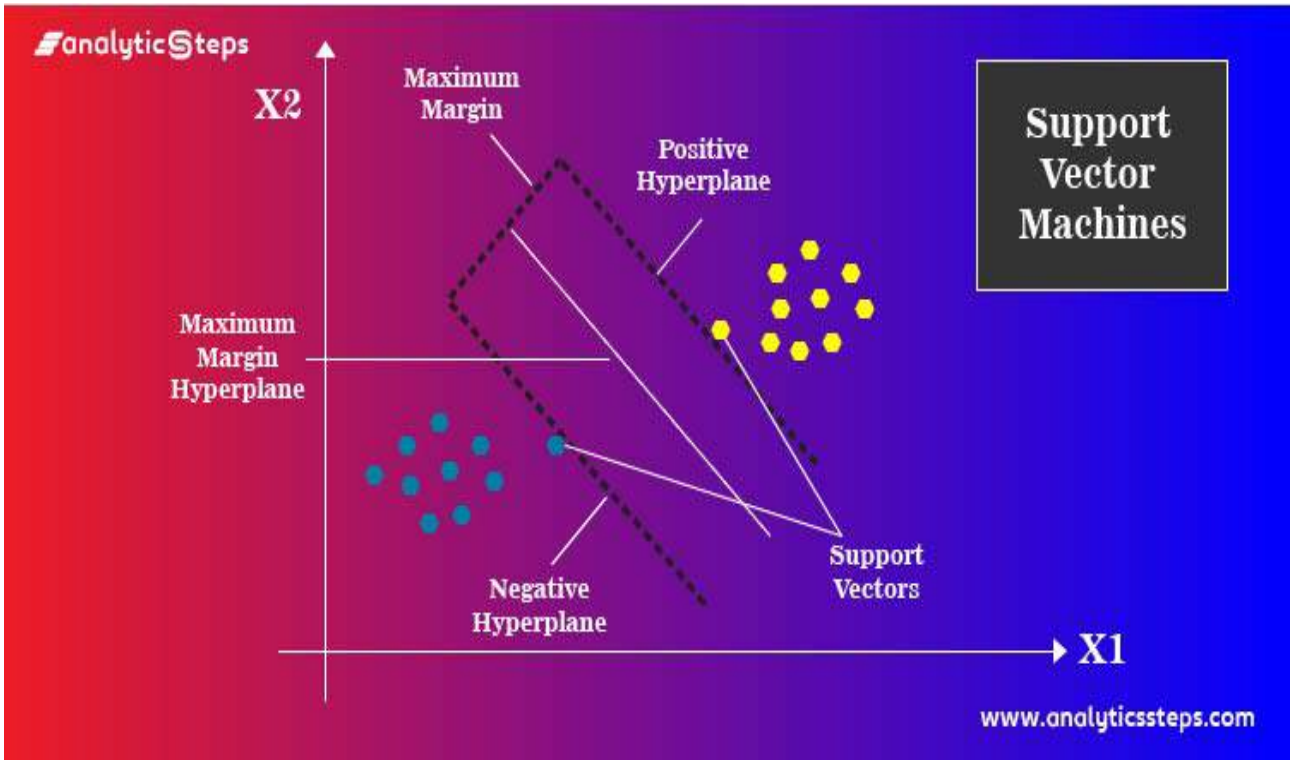

Figure 2.4: Support vector machine classifier

Steps involved in SVM Algorithm:

- Step 1: In SVM algorithm we predict the class first , one class as 0 and one class as 1.
- Step 2:Initializing SVM classifier model ,in that loss/cost function is used and tweaked to find the maximum margin
- Step 3: Here we found trade-off between maximizing margin and the loss generated if the margin is maximized to a very large extent. To overcome this problem, regularization parameter is added.
- Step 4: By using partial derivatives ,weights are optimized
- Step 5:In classification problem when there is no error gradients will be updated using regularization parameter

KNN Algorithm

K-Nearest Neighbor(KNN) Algorithm

K-NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems. K-NN algorithm assumes the similarity between the new case/data and available cases and put the new case into the category that is most similar to the available categories.K-NN algorithm stores all the available data and classifies a new data point based on the similarity. This means when new data appears then it can be easily classified into a well suite category by using K-NN algorithm. How does K-NN work?

> The K-NN working can be explained on the basis of the below algorithm:Step-1: Select the number K of the neighbors Step-2: Calculate the Euclidean distance of K number of neighbors Step-3: Take the K nearest neighbors as per the calculated Euclidean distance. Step-4: Among these k neighbors, count the number of the data points in each category. Step-5: Assign the new data points to that category for which the number of the neighbor is maximum.

XGboost Algorithm

XGboost The most widely used machine learning algorithm, independent of whether the challenge is among classification or regression, is XGboost. When compared with all other machine learning algorithms, it is known for its solid performance. Bagging Even though decision trees are among the models that are easiest to understand, their performance is extremely unpredictable. Think about a single training dataset that was randomly divided into two sections. Let's now use each component to train a decision tree to create two models. These two models would produce various outcomes when we fit them both. It is because of this characteristic that decision trees are said to be high variance. The variance in any learner can be decreased by bagging or boosting aggregation. The fundamental learners of the bagging technique are a number of parallel-generated decision trees. These learners are trained using data that has been sampled with replacement. The sum of all learner outputs, or the final prediction, is calculated. Boosting In boosting, the trees are constructed in a systematic fashion with each one aiming to minimize the faults of the one before it. Each tree updates the residual errors as a result of what its predecessors have learned. As a result, the tree that develops next in the succession will learn things from an updated set of residuals. In boosting, the weak learners with large bias and slightly improved predictive power over random guessing are the base learners. The boosting strategy efficiently integrates these weak learners to build a strong learner by leveraging the valuable information that each of these weak learners provides for prediction. The last adept learner reduces both the bias and the variance. Following are the steps involved in creating a Decision Tree using similarity score:

- Step 1 : Create a single leaf tree.
- Step 2:Calculate the target variable's average as a prediction for the first tree then use the specified loss function to compute the residuals. The residuals for subsequent trees are based on predictions from the prior tree.
- Step 3: Apply the following equation to obtain the similarity score: Hessian = number of residuals; Gradient2 = square root of residuals; and = regularization hyper parameter.
- Step 4 :We choose the right node based on similarity score. More homogeneity is observed with higher similarity scores.

Gaussian Naive Bayes Algorithm

Gaussian Naive Bayes A class of machine learning supervised classification methods built on the Bayes theorem are known as naive bayes. Although it is a simple classification method, it is

highly functional. When the inputs are highly dimensional, they were useful. The Naive Bayes Classifier can be used to solve complex classification issues as well. The assumption that the constant values associated with each class are distributed according to a normal (or Gaussian) distribution is constantly made when working with continuous data.

Assuming that the data is characterised by a Gaussian distribution with no covariance (independent dimensions) across dimensions is one method for building a straightforward model. Finding the standard deviation of the points within each label, which is all that is required to establish such a distribution, will enable this model to be fit. The Gaussian Naive Bayes (GNB) classification is illustrated in the image above. Every data point's z-score distance from each class mean, which is the length from the class mean divided by the class's standard deviation, is produced.

- Step 1: Now we import the data.
- Step 2: Now we do the step of splitting data into test and train sets.
- Step 3: Now, heading into Feature Scaling.
- Step 4: Applying the classifier model.
- Step 5: try to find the confusion matrix.
- Step 6: We can compute the accuracy test from the confusion matrix

Chapter 3

Results and Discussion

3.1 Exploratory data analysis

Exploratory Data Analysis is a process of examining or understanding the data and extracting insights or main characteristics of the data. EDA is generally classified into two methods, i.e. graphical analysis and non-graphical analysis.

EDA is very essential because it is a good practice to first understand the problem statement and the various relationships between the data features before getting your hands dirty.

3.1.1 The primary motive of EDA is to

- Examine the data distribution
- Handling missing values of the dataset(a most common issue with every dataset)
- Handling the outliers
- Removing duplicate data
- Encoding the categorical variables
- Normalizing and Scaling

3.1.2 Steps involved in EDA:

- Import libraries such as NumPy, Pandas and Seaborn.
- Read the dataset using Pandas.
- We first check the shape of our dataframe. It contains 284807 rows or observations and 31 columns.
- Now we drop duplicates of dataframe. It has 283726 rows or observations and 31 columns.
- Check for missing values. Our dataset do not contain any missing values.
- Statistics Summary:

  The information gives a quick and simple description of the data. It includes Count, Mean, Standard Deviation, median, mode, minimum value, maximum value, range, standard deviation, etc.

```
In [7]: df.describe()
```

Out[7]:

	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	..
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	..
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	..
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	..
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	..
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	..
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	..
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	..

8 rows × 31 columns

Figure 3.1: Statistical Summary

Statistics summary gives a high-level idea to identify whether the data has any outliers, data entry error, distribution of data such as the data is normally distributed or left/right skewed.

- Transaction time distribution:

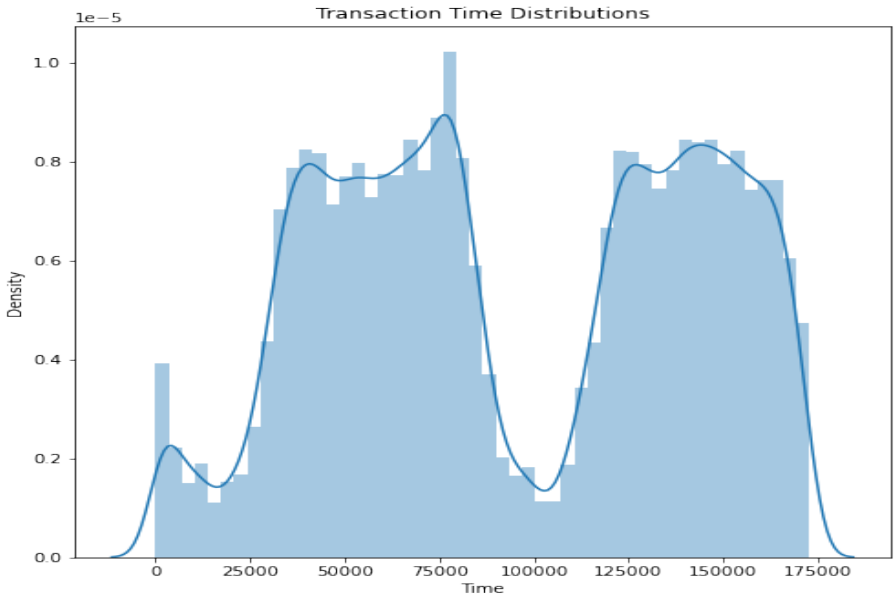

Figure 3.2: Transaction time distribution

There are two peaks in the graph. This dataset is for 2 days. We can relate this as two peaks corresponds to the two times in each day where maximum number of transactions that are happening(and depth corresponds to the night time where people are not doing any transactions).

- Check for categories of dependent variable i.e. class.

  It contains two classes represented by "0" and "1".

- Check count of categories of class variable. To check whether the data is balanced or not.

3.2 Data Visualization and Data Processing

3.2.1 Data Visualization

Data visualization is essential; we must decide what charts to plot to better understand the data. In our project, we visualize our data using Matplotlib and Seaborn libraries.

- Matplotlib Matplotlib is a Python 2D plotting library used to draw basic charts we use Matplotlib.
- Seaborn Seaborn is also a python library built on top of Matplotlib that uses short lines of code to create and style statistical plots from Pandas and Numpy.

Univariate analysis can be done for both Categorical and Numerical variables. Categorical variables can be visualized using a Count plot, Bar Chart, Pie Plot, etc. Numerical Variables can be visualized using Histogram, Box Plot, Density Plot, etc.

In our example, we have done a Univariate analysis using Histogram and Box Plot for continuous Variables. A histogram and box plot is used to show the pattern of the variables, as some variables have skewness and outliers.

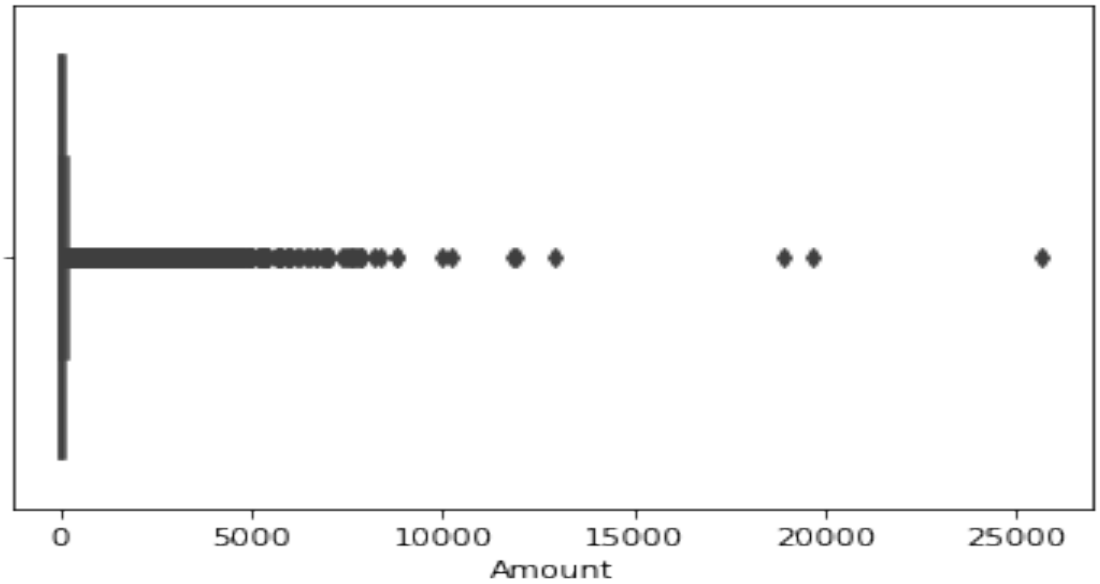

Figure 3.3: Boxplot of "Amount"

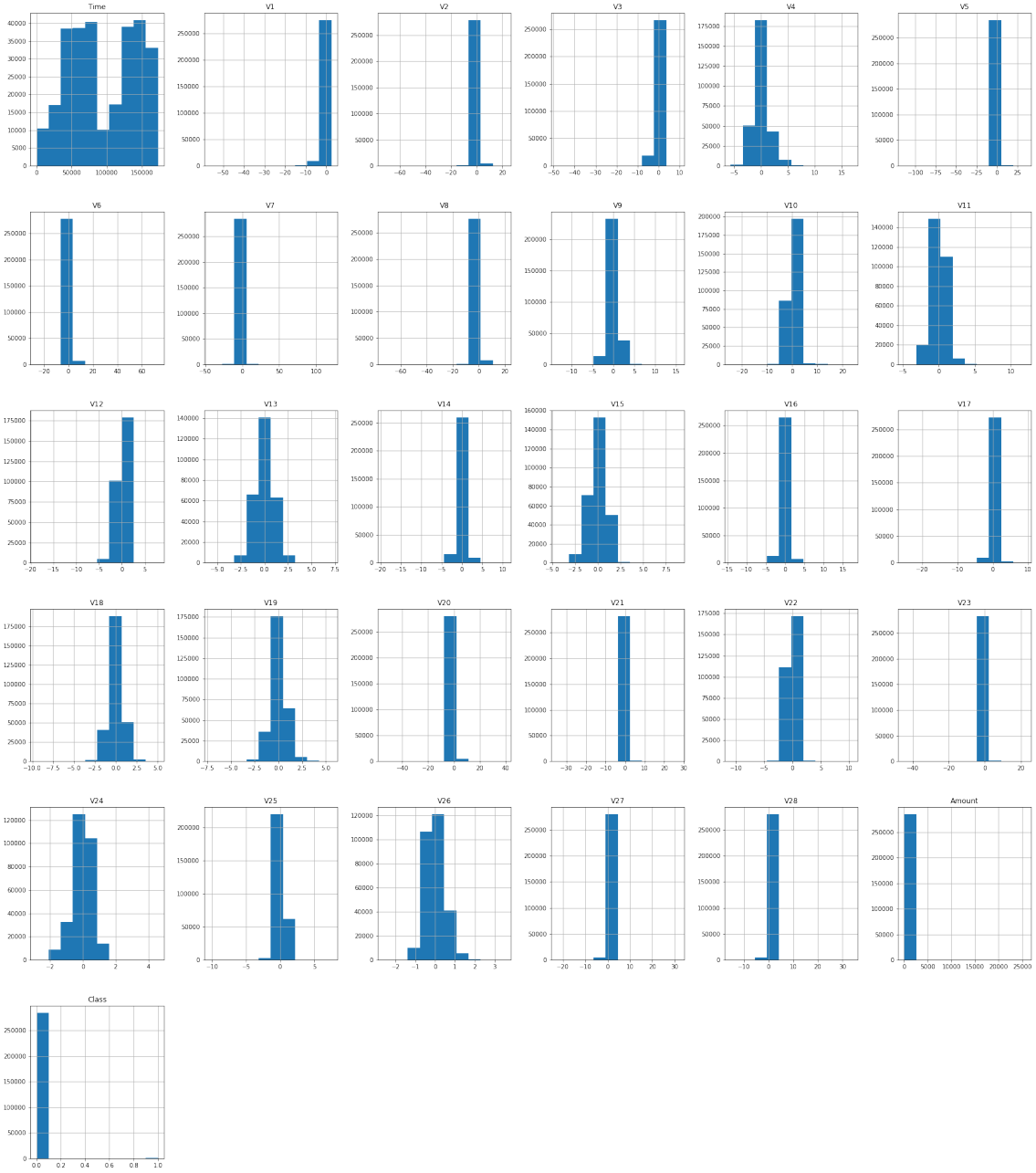

Figure 3.4: Histogram

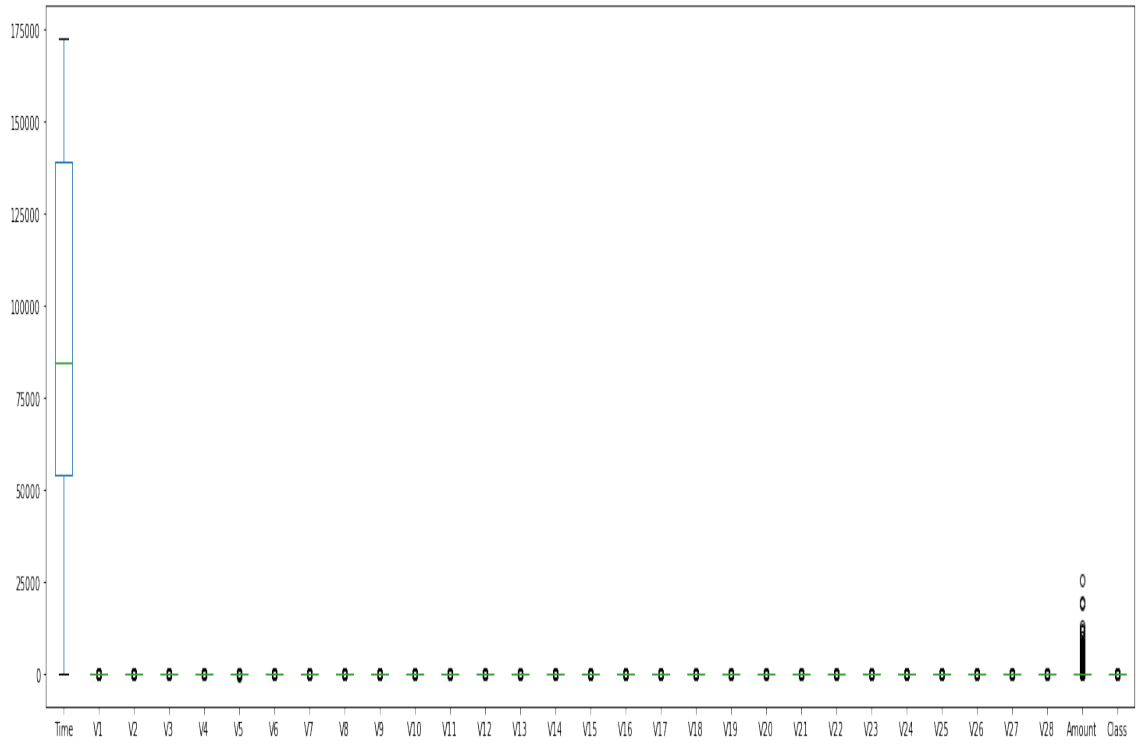

Figure 3.5: Boxplot of all feature variables

3.2.2 Data processing

Feature scaling:

From above boxplot, 'amount' and other feature variables ranges are different so we need to standardize the 'amount'. In our dataset, the 'amount' is in hundreds but the other columns would be under 10. This would lead the amount column to dominate the feature prediction even though it might be less significant. For this reason, different types of Scaling is used.

Different types of Feature Scaling:

- Log
- Standardization
- Normalization

Log is a scaling technique which is done when the variables span several orders of magnitude.

Standardization is a scaling technique are the ones where the values are centered around the mean with a unit standard deviation. This means that the mean of the attribute becomes zero and the resultant distribution has a unit standard deviation.

Normalization (Min-Max Scaling) is a scaling technique in which values are shifted and are then rescaled so that they end up ranging between 0 and 1.

In our project, we have used standardization method for feature scaling of feature 'amount'. We have used Sklearn library. Now, the value of 'amount' feature will lie between -3 to +3.

3.2.3 Split the data

Split the dataset into feature variables and response variable.

Why is Class Imbalance a Problem?

When a statistical classifier is trained on a highly imbalanced dataset, it has a tendency to pick the patterns in the most popular class and ignore the rest.

For example, in this dataset, 99.9 percentage of the data are labelled as 'Not Fraud' and rest are 'Fraud'. So, even if a model classifies everything that it sees as 'Not Fraud', the accuracy is going to be 99.9 percentage which seems excellent.

But the model is not good. Because it is not classifying any of the transaction as 'Fraud'. So, even if the model has an accuracy of 99.9 percentage, it is completely useless.

We need some strategies to work with in such a dataset or we need to use some other metrics(except for accuracy) in such scenarios.

Dealing with Imbalance Class

- Under Sample majority class
- Over Sample minority class
- Synthetic Minority Over Sampling Technique(SMOTE)
- Adaptive Synthetic Samples(ADASYN)

In under-sampling, the number of samples in majority class are down sampled(by eliminating them randomly)to align them to the number of samples in minority class. This can lead to Data Inefficiency as loss of useful data can make the decision boundary between minority and majority samples harder to learn for rule-based classifiers. This technique is only effective when minority class has sufficient data despite of being affected by severe imbalance.

Here, we'll use under sample majority class techniques to deal with the imbalance class. After under sampling there will be 662 observations are considered in both 'Fraud' and 'Non-Fraud' cases before it was 199364 observations each.

3.2.4 Split the data

We split the dataset into training data and test data in ratio of 80:20. We have done under sampling of training data. so we will also create the training data variables of x and y as well as will the test data variables of x and y.

3.3 Model fitting

3.3.1 Evaluation metrics

In case of machine learning or deep learning it is always the best practice to test the model. By evaluating the model we can measure the quality of our model and can see how well can our model do with respect to our use case.

- Accuracy: The accuracy of a classifier is calculated as the ratio of the total number of correctly predicted samples by the total number of samples. It can be used to evaluate the classifier when the data is balanced data set. If it is used to evaluate model accuracy for balanced data then it will give more accuracy for the majority class.

  Accuracy = (TP + TN) / (TP + FP + TN + FN)

$$Accuracy = \frac{Total\ number\ of\ correctly\ predicted\ samples}{Total\ number\ of\ samples}$$

Figure 3.6: Accuracy

- Precision: It is positive predictive value. It is the ratio of true positives (TP) and the sum of true positives (TP) and false positives (FP). Precision tells us out of total predicted positive values how many were actual positive values in case of binary classes. It is used based on use case. In case of spam detection we have to check whether the mail is spam and if 'spam' then its value will be '1' and for 'not spam' its value will be '0'. If it detects the important mail as spam which is not actually spam here actual value is '0' and predicted value is '1' then it is a false positive (FP) case. Here the false positive cases should be reduced. Hence, here precision will be used as a metric to measure the quality of our classifier.

$$Precision = \frac{True\ Positives\ (TP)}{True\ Positives\ (TP) + False\ Positives\ (FP)}$$

Figure 3.7: Precision

- Recall: It is ratio of true positives (TP) by the sum of true positives (TP) and false negatives (FN). Recall tells us out of total actual negative values how many did our classifier predict negatively in case binary classifier. It is used based on use case. In case of cancer detection we have to check whether the patient has cancer or not and if 'Yes' then its value will be '1' and for 'No' its value will be '0'. If it detects that patient doesn't have cancer where in actual the patient has cancer here actual value is '1' and predicted value is '0' then it is a false negative (FN) case. Here the false negative cases should be reduced. Hence, here recall will be used as a metric to measure the quality of our classifier.

$$Recall = \frac{True\ Positives\ (TP)}{True\ Positives\ (TP) + False\ Negatives\ (FN)}$$

Figure 3.8: Recall

- F1 Score: F1 score should be used when both precision and recall are important for the use case. F1 score is the harmonic mean of precision and recall. It lies between [0,1]. It is used when both False Positives (FP) and False Negatives (FN) are important. F1 score is derived from F Beta Score. F Beta score is the weighted harmonic mean of precision and recall.

$$F1\ Score = \frac{2 * Precision * Recall}{Precision + Recall}$$

Figure 3.9: F1-score

- AUC-ROC Curve: AUC-ROC Curve is a performance metric that is used to measure the performance for the classification model at different threshold values. ROC is Receiver Operating Characteristic Curve and AUC is Area Under Curve. The higher the value of AUC (Area under the curve), the better is our classifier in predicting the classes. AUC-ROC is mostly used in binary classification problems. The ROC curve is plotted between True Positive Rate (TPR) and False Positive Rate (FPR) i.e. TPR on the y-axis and FPR on the x-axis. AUC is the area under the ROC curve. An excellent classifier has an AUC value near 1, whereas a poor-performing classifier has an AOC value near 0. A classifier with an AOC score of 0.5 doesn't have any class separation capacity.

$$TPR = \frac{TP}{TP + FN}$$

Figure 3.10: True positive rate

$$FPR = \frac{FP}{FP + TN}$$

Figure 3.11: False positive rate

- Confusion Matrix: A confusion matrix is n dimensional square matrix where n represents the total number of target classes. Confusion matrix can be used to evaluate a classifier whenever the data set is imbalanced.

  There are four terms in confusion matrix: 1. True Positives (TP): The no. of times when actual value is '1' and predicted value is also '1'. 2. True Negatives (TN): The no. of times when actual value is '0' and predicted value is also '0'. 3. False Positive (FP): The no. of times when actual value is '0' and predicted value is '1'. 4. False Negative (FN): The no. of times when actual value is '1' and predicted value is '0'.

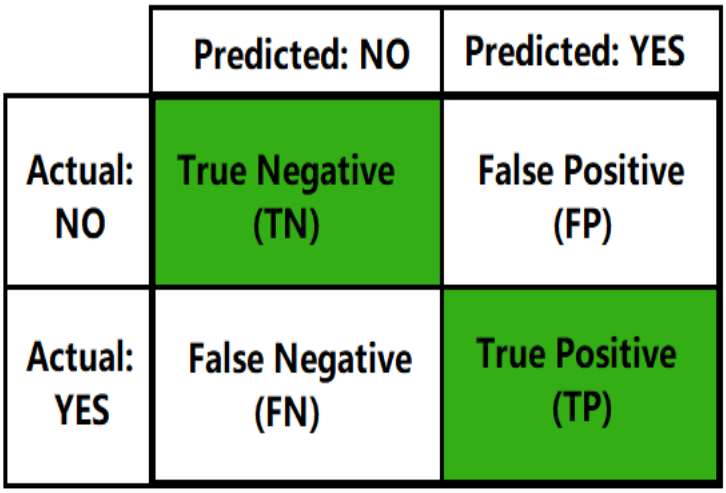

Figure 3.12: Confusion rate

3.3.2 Classification algorithms:

Classification is a supervised learning technique which involves predicting the class label for the given input data. In a classification problem, we understand the problem, explore the data, process the data and then build a classification model using machine learning algorithms or a deep learning technique.

In our project, we have fitted models on various classification algorithms such as

- Logistic regression classifier: We have imported the classifier from Sci-kit learn. In logistic regressor we have applied the logistic regression model on imbalanced data as well as balanced data. Then we applied the performance metrics on the model.
- Random Forest classifier: We have imported the classifier from Sci-kit learn. In Random Forest classifier we have fitted the model on imbalanced data as well as balanced data. Then we applied the performance metrics on the model.
- Gaussian Naïve Bayes Classifier: We have imported the classifier from Sci-kit learn. In Gaussian Naïve Bayes classifier we have fitted the model on imbalanced data as well as balanced data. Then we applied the performance metrics on the model.
- Decision tree classifier: We have imported the classifier from Sci-kit learn. In Decision tree classifier we have fitted the model on imbalanced data as well as balanced data. Then we applied the performance metrics on the model.
- K-Nearest Neighbours: We have imported the classifier from Sci-kit learn. In KNN classifier we have fitted the model on imbalanced data as well as balanced data. Then we applied the performance metrics on the model.
- XG Boost classifier: We have imported the 'XGBClassifier' from xgboost. In XG Boost classifier we have fitted the model on imbalanced data as well as balanced data. Then we applied the performance metrics on the model.

3.3.3 Comparison of various models:

We obtain the comparison of different models based on the F1 score value of respective models. We can see that the F1 score of XGBoost classifier is maximum of imbalanced data followed by Random forest classifier of imbalanced data and then decision tree classifier of imbalanced data. Whereas F1 score of logistic regression is zero.

	Model	Accuracy	AUC	Precision Score	Recall Score	F1 Score
10	XGBoost Imbalanced	0.999520	0.900568	0.934783	0.801242	0.862876
2	RF IMABALANCED	0.999520	0.897469	0.941176	0.795031	0.861953
6	DT Imbalanced	0.998982	0.878601	0.717647	0.757764	0.737160
4	NB Imbalanced	0.992919	0.847666	0.168657	0.701863	0.271961
5	NB Undersample	0.991152	0.890177	0.149588	0.788820	0.251485
8	KNN Imbalanced	0.998303	0.555889	0.900000	0.111801	0.198895
3	RF Undersample	0.976686	0.948025	0.069582	0.919255	0.129371
11	XGBoost Undersample	0.973772	0.952764	0.063025	0.931677	0.118064
7	DT Undersample	0.896914	0.901864	0.016333	0.906832	0.032088
9	KNN Undersample	0.614796	0.639649	0.003246	0.664596	0.006460
0	LRImbalanced	0.998116	0.500000	0.000000	0.000000	0.000000
1	LRImbalanced	0.998116	0.500000	0.000000	0.000000	0.000000

Figure 3.13: Comparison of various model

Chapter 4

Summary and Conclusion

Our project is about studying credit card fraud detection models based on different machine learning algorithms. Training and testing is goal of our project to find the way for processing dataset. For this credit card fraud detection data set we found the beat machine learning algorithm . Furthermore , for achieving this goal we selected five different classifiers. By making classifier combinations with each other to evaluate their predicted performance to get better results for our credit card fraud detection dataset. We used undersampling to deal with a unbalanced credit card transaction dataset in the confusion matrix ended up with the same results as we expected. Logistic regression is a simple algorithms which is having advantages in targeting data processing .Logistic regression also achieves high accuracy results. Random forest algorithm will provide better performance with many testing data , but speed during testing and application will still suffer. Our future work will try to represent this into a software application and providing solution to the new technologies like artificial intelligence and deep learning.

Chapter 5

References

- Google for understanding
- https://www.ijcsmc.com/docs/papers/April2021/V10I4202112.pdf
- https://pats.cs.cf.ac.uk/@archive$_file?p = 1859n = finalf = 1-report.pdfSIG = 04e2353a2ff38fd$c
  //www.freeprojectz.com/project - report/26873

- https://www.ripublication.com/ijaer18/ijaerv13n24$_1$8.*pdfhttps* : *//www.researchgate.net/publication/33*
- https://apps.scms.unipune.ac.in/moodle/pluginfile.php/5200/mod$_r$*esource/content*/1/*an$_i$ntroduction$_t$o$_s$t*
  *james$_e$tal-springer-*2021.*pdfhttps* : *//www.kaggle.com/datasets/mlg-ulb/creditcardfraud*
- https://www.analyticsvidhya.com/blog/2022/03/exploratory-data-analysis-eda-credit-card-fraud-detection-case-study/
- "Credit Card Fraud - Consumer Action" (PDF). Consumer Action. Retrieved 28 November 2017.
-

Bibliography