**CREDIT CARD FRAUD DETECTION**

**A MINI PROJECT REPORT**

***Submitted by***

**ANUPRIYA K (952020104005)**

**JAYAKAVITHA S (952020104009)**

***in partial fulfillment for the award of the degree***

***of***

**BACHELOR OF ENGINEERING**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

****

**P.S.R.R COLLEGE OF ENGINEERING**

**SIVAKASI-626140**

**ANNA UNIVERSITY: CHENNAI-600025**

**MAY:2023**

**ANNA UNIVERSITY: CHENNAI-600025**

**BONAFIDE CERITIFICATE**

Certified that this mini project title "**CREDIT CARD FRAUD DETECTION USING EMERGENCY CALL AND BLOCK SYSTEM"** is the bonafide work of **JAYAKAVITHA S(952020104009) ANUPRIYA K(952020104005),** who carried out the project work under my supervision.

**SIGNATURE SIGNATURE**

**Mrs.K.RAJASATHIYA M.Tech Mrs.K.RAJASATHIYAM.Tech**

**HEAD OF THE DEPARTMENT ASSISTANT PROFESSOR**

DEPARTMENT OF CSE DEPARTMENT OF CSE

P.S.R.R COLLEGE OF ENGINEERING P.S.R.R COLLEGE OF ENIGINEERING

SIVAKASI-6262140 SIVAKASI-6262140

**Submitted by theVIVA-VOICE examination held on** .........................

**Internal examiner External examiner**

**ABSTRACT**

Credit card fraud detection is presently the most frequently occurring problem in the present world. This is due to the rise in both online transactions and e-commerce platforms. Credit card fraud generally happens when the card was stolen for any of the unauthorized purposes or even when the fraudster uses the credit card information for his use. In the present world, we are facing a lot of credit card problems. To detect the fraudulent activities the credit card fraud detection system was introduced. This project aims to focus mainly on machine learning algorithms. The algorithms used are random forest algorithm, linear regression, Support vector classifier, Linear Discriminant Analysis algorithm. In the instance of any unusual activity, the system will not only raise emergency calls, but it will also block the user after three invalid attempts. 'Fraud' in credit card transactions is unauthorized and unwanted usage of an account by someone other than the owner of that account. The performance of fraud detecting in credit card transactions is greatly affected by the sampling approach on data-set, selection of variables and detection techniques used. This paper investigates the performance of logistic regression, decision tree and random forest for credit card fraud detection. Dataset of credit card transactions is collected from kaggle and it contains a total of 2,84,808 credit card transactions of a European bank data set. It considers fraud transactions as the “positive class” and genuine ones as the “negative class”. The data set is highly imbalanced, it has about 0.172% of fraud transactions and the rest are genuine transactions. The author has been done oversampling to balance the data set, which resulted in 60% of fraud transactions and 40% genuine ones. The performance of the techniques is evaluated for different variables based on sensitivity, specificity, accuracy and error rate. The result shows of accuracy for logistic regression,

Decision tree and random forest classifier are 90.0, 94.3, 95.5 respectively. The comparative results show that the Random forest performs better than the logistic regression and decision tree techniques. - This Project is focused on credit card fraud detection in real world scenarios. Nowadays credit card frauds are drastically increasing in number as compared to earlier times. Criminals are using fake identity and various technologies to trap the users and get the money out of them. Therefore, it is very essential to find a solution to these types of frauds. In this proposed project we designed a model to detect the fraud activity in credit card transactions. This system can provide most of the important features required to detect illegal and illicit transactions. As technology changes constantly, it is becoming difficult to track the behavior and pattern of criminal transactions. To come up with the solution one can make use of technologies with the increase ofmachine learning, artificial intelligence and other relevant fields of information technology; it becomes feasible to automate this process and to save some of the intensive amounts of labor that is put into detecting credit card fraud. Initially, we will collect the credit card usage data-set by users and classify it as trained and testing dataset using a random forest algorithm and decision trees. The performance of the techniques is gauged based on accuracy, sensitivity, and specificity, precision. The results is indicated concerning the best accuracy for Random Forest are unit 98.6% respectively.

**ACKNOWLEDGEMENT**

First of all I thank **LORD ALMIGHTY** for his grace and countless blessing in making this work a great success

I sincerely thank Thiru. **R. SOLAISAMY** our respected Correspondent and **Er.S.VIGNESHWARI ARUNKUMAR B. Tech.** Our respected Director and **Dr.C.BALASUBRAMANIAN M.E., Ph.D.,** for providing sample facilities and necessary infrastructure made available during the course of project.

I wish to express my thanks and sincere gratitude to our Head of the Department, **Mrs. K. RAJASATHIYA M.Tech,** for her valuable advice and guidance to complete this work successfully and also for her enthusiastic encouragement.

With deep some of the gratitude, I would like to thank our project Coordinator **Mrs.P.PACKIYALAKSHMI M.E** for her encouragement and help throughout our project.

I am almost grateful to my internal guide **Mrs. K. RAJASATHIYA M.Tech,** who guided me and encouraged me by giving constructive criticism during the entire work. Her vision and wisdom helped me to realize the importance of this work. She let me develop my potential at my own peace, and shepherded me in her special way.

Also I think all the teaching and non-teaching faculties of all department of P.S.R.R College of Engineering for their support.

|  |  |  |
| --- | --- | --- |
| **TABLE OF CONTENTS** | | |
| **CHAPTER NO** | **TITLE** | **PAGENO** |
|  | **ABSTRACT** | iii |
|  | **ACKNOWLEDGEMENT** | v |
|  | **LIST OF FIGURES** | viii |
|  | **LIST OF ABBREVIATIONS** | ix |
| **1** | **INTRODUCTION** | 1 |
|  | 1.1 OVERVIEW |  |
|  | 1.2BACKGROUND OF STUDY |  |
|  | 1.3SIGNIFICANCE OF STUDY |  |
|  | 1.4 OBJECTIVE OF PROJECT |  |
|  | 1.5 PROJECT JUSTIFICATION |  |
|  | 1.6 SCOPE OF THE PROJECT |  |
| **2** | **LITERATURE REVIEW** |  |
| **3** | **SYSTEM ANALYSIS** |  |
|  | 3.1 EXISTING SYSTEM |  |
|  | 3.2 PROPOSED SYSTEM |  |
|  | 3.3 ARCHITECTURE |  |
| **4** | **SYSTEM DESIGN** |  |
|  | 4.1 USECASE DIAGRAM |  |
|  | 4.2 ACTIVITY DIAGRAM |  |
|  | 4.3 SEQUENCE DIAGRAM |  |
|  | 4.4 ER DIAGRAM |  |
| 5 | **REQUIREMENTS SPECIFICATION** |  |
|  | 5.1 SYSTEM REQUIREMENTS |  |
|  | 5.1.1 SOFTWARE REQUIREMENTS |  |
|  | 5.1.2 HARDWARE REQUIREMENTS |  |
|  | 5.2 FLOW DIAGRAM |  |
| 6 | **SYSTEM IMPLEMENTATION** |  |
|  | 6.1 MODULES |  |
|  | 6.2 MODULES DESCRIPTION |  |
|  | 6.3 IMPLEMENTATION OF SYSTEM |  |
|  | 6.4 FORM INPUT & REPORT DESIGN |  |
|  | 6.5 THE LOGIN FORM |  |
| **7** | **SYSTEM TESTING** |  |
|  | 7.1 TESTING METHODOLOGY |  |
|  | 7.1.1 UNIT TESTING |  |
|  | 7.1.2 SYSTEM TESTING |  |
| **8** | **RESULT** |  |
| **9** | **CONCLUSION** |  |
| **10** | **FUTURE ENHANCEMENT** |  |
| **11** | **REFERENCE** |  |
| **12** | **APPENDIX** |  |

|  |  |  |
| --- | --- | --- |
|  | **LIST OF FIGURES** |  |
| **FIGURE NO** | **FIGURE NAME** | **PAGE NO** |
| 4.1 | Use Case Diagram |  |
| 4.2 | Activity Diagram |  |
| 4.3 | Sequence Diagram |  |
| 4.4 | ER Diagram |  |
|  |  |  |

|  |  |  |
| --- | --- | --- |
|  | **LIST OF ABBREVATIONS** |  |
| **S.NO** | **ABBREVIATION** | **EXPLANATION** |
| 1 | DOC | DISORDER OF CONSICIOUSNESS |
| 2 | ML | MACHINE LEARNING |
| 3 | PCA | PRINCIPAL COMPONENT ANALYSIS |
| 4 | EEG | ELECTROENCEPHALOGRAM |
| 5 | PSDD | POWER SPECTRAL DENSITY DIFFERENCE |
| 6 | URL | UNIFORM RESOURCE LOCATOR |

**CHAPTER 1**

**INTRODUCTION**

* 1. **OVERVIEW**

Credit Card Fraud can be defined as a case where a person uses someone else’s credit card for personal reasons while the owner and the card issuing authorities are unaware of the fact that the card is being used. Fraud detection involves monitoring the activities of populations of users in order to estimate, perceive or avoid objectionable behavior, which consist of fraud, intrusion, and defaulting. This is a very relevant problem that demands the attention of communities such as machine learning and data science where the solution to this problem can be automated. This problem is particularly challenging from the perspective of learning, as it is characterized by various factors such as class imbalance. The number of valid transactions far outnumber fraudulent ones. Also, the transaction patterns often change their statistical properties over the course of time. ‘Fraud' in credit card transactions is unauthorized and unwanted usage of an account by someone other than the owner of that account.This problem is particularly challenging from the perspective of learning, as it is characterized by various factors such as class imbalance. The number of valid transactions far outnumber fraudulent ones. Also, the transaction patterns often change their statistical properties over the course of time. In real world scenarios. Nowadays credit card frauds are drastically increasing in number as compared to earlier times. Criminals are using fake identity and various technologies to trap the users and get the money out of them. Therefore, it is very essential to find a solution to these types of frauds. In this proposed project we designed a model to detect the fraud activity in credit card transactions.

**1.2 BACKGROUND OF STUDY**

Demonstrate the efficiency of classification models to credit card fraud detection problem and the authors proposed the three classification models ie., decision tree, neural network and logistic regression. Among the three models neural network and logistic regression outperforms than the decision tree the probability theory frame work for making decision under uncertainty. After reviewing Bayesian theory, naïve bayes classifier and k-nearest neighbor classifier is implemented and applied to the dataset for credit card has cited the research for credit card fraud detection and used seven classification methods took a major role.In this work they have included decision trees and SVMs to decrease the risk of the banks. They have suggested Artificial Neural networks and Logistic Regression classification models are more helpful to improve the performance in detecting the frauds, used Artificial Neural Network and Logistic Regression Classification and explained ANN classifiers outperform LR classifiers in solving the problem under investigation. Here the training data sets distribution became more biased and the distribution of the training data sets became more biased and the efficiency of all models decreased in catching the fraudulent transactions. In this proposed project we designed a protocol or a model to detect the fraud activity in credit card transactions. This system is capable of providing most of the essential features required to detect fraudulent and legitimate transactions. As technology changes, it

becomes difficult to track the Modeling and pattern of fraudulent transactions.

With the rise of machine learning, artificial intelligence and other relevant fields of information technology, it becomes feasible to automate this process and to save some of the intensive amount of labour that is put into detecting credit card fraud.

**1.3SIGNIFICANCE OF STUDY**

Credit card fraud detection is the collective term for the policies,tools, methodologies, and practices that credit card companies and financial institutions take to combat identity fraud and stop fraudulent transactions.The key objective of any credit card fraud detection system is to identify suspicious events and report them to an analyst while letting normal transactions be automatically processed. For years, financial institutions have been entrusting this task to rule-based systems that employ rule sets written by experts.Credit card fraud detection is the collective term for the policies, tools, methodologies, and practices that credit card companies and financial institutions take to combat identity fraud and stop fraudulent transactions.

* 1. **OBJECTIVE OF PROJECT**

The specific objectives of the project include:

* Won’t mark any user so misunderstanding of user and fraud is avoided
* Accuracy is better than the existing system.
* Historical data is use so we can get the pattern of user’s original transaction and Fraud transaction.

**1.5 PROJECT JUSTIFICATION**

The key objective of any credit card fraud detection system is to identify suspicious events and report them to an analyst while letting normal transactions be automatically process.

**1.6 SCOPE OF THE PROJECT**

In this proposed project we designed a protocol or a model to detect the fraud activity in credit card transactions. This system is capable of providing most of the essential features required to detect fraudulent and legitimate transactions. As technology changes, it becomes difficult to track the Modeling and pattern of fraudulent transactions. With the rise of machine learning, artificial intelligence and other relevant fields of information technology, it becomes feasible to automate this process and to save some of the intensive amount of labour that is put into detecting credit card fraud.

* Can be highly developed and reduce more fraud solutions.
* Highly complexity can increase the detection of the irregular activities.

**CHAPTER 2**

**LITERATURE REVIEW**

**Literature Review**

Fraudulent Detection in Credit Card System Using SVM & Decision Tree. With growing advancement in the electronic commerce field, fraud is spreading all over the world, causing major financial losses. In the current scenario, Major cause of financial losses is credit card fraud; it not only affects tradesperson but also individual clients. Decision tree, Genetic algorithm, Meta learning strategy, neural network, HMM are the presented methods used to detect credit card frauds. In contemplating system for fraudulent detection, artificial intelligence concept of Support Vector Machine (SVM) & decision tree is being used to solve the problem. Thus by the implementation of this hybrid approach, financial losses can be reduced to greater extent. Machine Learning Based Approach to Financial Fraud Detection Process in Mobile Payment System. Mobile payment fraud is the unauthorized use of mobile transaction through identity theft or credit card stealing to fraudulently obtain money. Mobile payment fraud is a fast growing issue through the emergence of smart phone and online transition services. In the real world, a highly accurate process in mobile payment fraud detection is needed since financial fraud causes financial loss. Therefore, our approach proposed the overall process of detecting mobile payment fraud based on machine learning, supervised and unsupervised method to detect fraud and process large amounts of financial data. Moreover, our approach performed sampling process and feature selection process for fast processing with large volumes of transaction data and to achieve high accuracy in mobile payment detection. F-measure and ROC curve are used to validate our proposed model. We propose a Machine learning

5

model to detect fraudulent credit card activities in online financial transactions. Analyzing fake transactions manually is impracticable due to vast amounts of data and its complexity. However, adequately given informative features, could make it is possible using Machine Learning. This hypothesis will be explored in the project. To classify fraudulent and legitimate credit card transaction by supervised learning Algorithm such as Random forest. To help us to get awareness about the fraudulent and without loss of any financially. New

methods for credit card fraud detection with a lot of research methods and several fraud detection techniques with a special interest in the neural networks, data mining, and distributed data mining. Many other techniques are used to detect such credit card fraud. When done the literature survey on various methods of credit card fraud detection, we can conclude that to detect credit card fraud there aremany other approaches in Machine Learning itself. The research on credit card fraud detection uses both Machine Learning and Deep Learning algorithms. In this section, we enhance the work done in two different points:

(i)the methods that are readily available for fraud detection

(ii) The techniques that are available to handle the imbalanced data. To handle the imbalanced data some of the techniques are available. They are (a) classification methods (b) sampling methods (c) resembling techniques. Here are some of the Machine Learning algorithms that are used for credit fraud detection are support vector machine(SVM), decision trees, logistic regression, gradient boosting, K-nearest neighbor etc.

**CHAPTER 3**

**SYSTEM ANALYSIS AND DESIGN**

**3.1 EXISTING SYSTEM**

The detection of the fraud use of the card is found much faster that the existing system. In case of the existing system even the original card holder is also checked for fraud detection. But in this system no need to check the original user as we maintain a log. The log which is maintained will also be a proof for the bank for the transaction made. We can find the most accuratedetection using this technique. This reduce the tedious work of an employee in the bank. Fraud is detected after the fraud is done, and the fraud is detected after the complaint of the cardholder. The transaction is maintained in a record, we need to up hold a vast data. The fraud transactions are given to alarm which alerts the user to that fraud transaction and prevents the user to that fraud transaction. Fraud is detected is a superlative and easy way. In the past, this was done by employees, who checked all transactions manually. Earlier, fraudulent transactions were detected using the outlier detection in which we find the critical anomalies present in the data. In data mining, anomaly detection (also outlier detection) is the identification of items, events or observations which do not conform to an expected pattern or other items in a dataset. There are few more methods like decision tree, cluster techniques and neural networks etc.

**Disadvantage of exisiting system.**

The main disadvantage of the existing technique is that it is difficult to find the correct number of clusters we want We could run different algorithm first to

7

see the distance between any number of clusters (how discriminate are they to each other). We also tend to lose information from heavy outliers since those have to be put in a cluster as well. Statistical fraud detection problem is a very difficult problem in that there are very few examples of fraud.

**3.2 PROPOSED SYSTEM**

In proposed system, we present a Hidden Markov Model (HMM). Which does not require fraud signatures and yet is able to detect frauds by considering a cardholder’s spending habit. Card transaction processing sequence by the stochastic process of an HMM. The details of items purchased in Individual transactions are usually not known to any Fraud Detection System(FDS) running at the bank that issues credit cards to the cardholders. Hence, we feel that HMM is an ideal choice for addressing this problem. Another important advantage of the HMM-based approach is a drastic reduction in the number of False Positives transactions identified as malicious by an FDS although they are actually genuine. An FDS runs at a credit card issuing bank. Each incoming transaction is submitted to the FDS for verification. FDS receives the card details and the value of purchase to verify, whether the transaction is genuine or not. The types of goods that are bought in that transaction are not known to the FDS. It tries to find any anomaly in the transaction based on the spending profile of the cardholder, shipping address. The performance of fraud detection in creszdit card transactions is greatly affected by the sampling approach on dataset, selection of variables and detection technique(s) used.

8

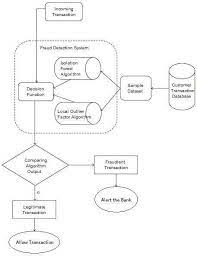
the knowledge of the ones that turned out to be fraud. Hence we are using thetechnique of machine learning for fraud detection. In this we take the real bank dataset and split the dataset into training set and testing set and then apply the Logistic Regression method and billing address, etc. If the FDS confirms the transaction to be of fraud, it raises an alarm, and the issuing bank declines the transaction. The fraud transaction are given to emergency calls to user that fraud transaction has occurred and the user can block the card to prevent further financial loss to him as well as the credit card company. The valid transactions are treated as genuine transactions.The goal of fraud decision has traditionally been viewed as a data mining task, with the goal of properly classifying transaction as legal or fraudulent. It will predict using features of the user already transacted details according to the purchase and card details, Semi supervised approach is used to detect fraudulent.

ADVANTAGE OF PROPOSED SYSTEM:

The benefits of fraud detection and prevention are that you can stop fraudsters from stealing your customers' personal information or loyalty points attached to their accounts. Thus, you provide a better customer experience.  Credit Card advantages include enhanced financial health, improved credit score, balance transfer, easy loan approval, access to funds in case of emergency, affordable EMIs, among multiple other additional benefits.

9

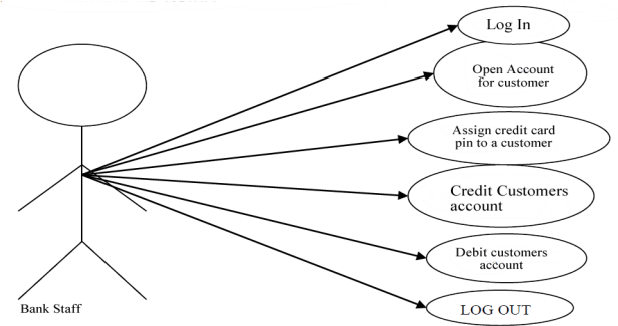
**3.3 ARCHITECTURE**

****

**CHAPTER 4**

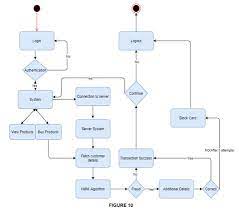
**DESIGN SYSTEM**

**4.1 USECASE DIAGRAM**

****

**Fig: Use case diagram of credit card fraud detection**

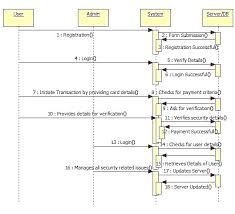
**4.2 ACTIVITY DIAGRAM**

****

**Fig: Activity Diagram of Credit Card Fruad Detection**

11

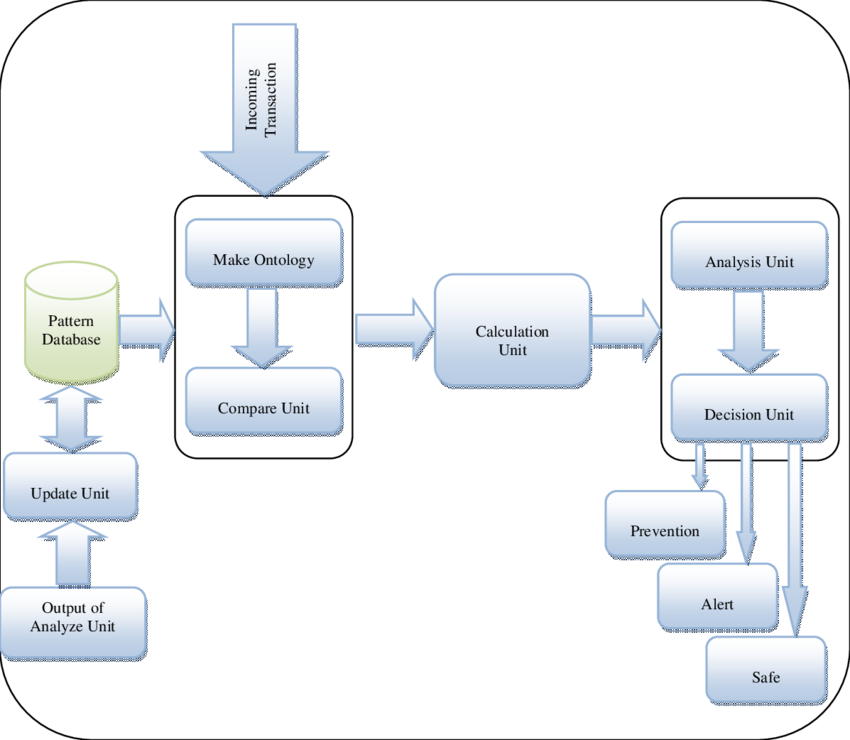
**4.3 SEQUENCE DIAGRAM**

****

**Fig: Sequence Diagram of Credit Card Fraud Detection**

12

**4.4 ER DIAGRAM**

****

**Fig: ER Diagram of Credit Card Fraud Detection**

13

**CHAPTER 5**

**REQUIREMENT SPECIFICATION**

**5.1 SYSTEM REQUIREMENTS**

**5.1.1 SOFTWARE REQUIREMENTS:**

It is a complete description of the system that is to be developed. It includes a set of use cases that describes all the interactions the users will have with the software. Use cases are also known as functionalrequirements. In addition to the use cases, the SRS also contains non-functional requirements. In addition to use cases, there are also functional (or supplementary) requirements. Nonfunctional requirements are requirements which impose constraints on the design or implementation (such as performance engineering supplementary) requirements. Non-functional requirements, quality standards, or design constraints). a) Operating system: An operating system (OS) is software that manages computer hardware and software resources and provides common services for computer programs. Operating system used is windows 10 with 64-bit operating system. Operating System: Windows 10 with 32-bit b) Programming Language: A programming language is a formal language that specifies a set of instructions that can be used to produce various kinds of output. Programming languages generally consist of instructions for a computer. Programming languages can be used to create programs that implement specific algorithms. Python is a general-purpose programming language that is becoming more and more popular for doing data science. Companies worldwide are using Python to harvest insights from their data and get a competitive edge.

14

Unlike any other Python tutorial, this course focuses on Python specifically for data science.

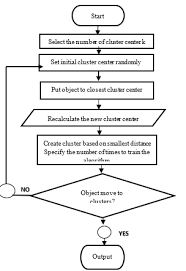
**5.1.2HARDWARE REQUIREMENTS:**

Hard Drives: Hard drives store all of your computer information once yoursystem is turned off. Hard drive of 4GB is minimum required. d) Processor: A processor is the logic circuitry that responds to and processes the basic instructions that drive a computer. Processor being used is Intel® Core i5 @ 2.20 GHz. RAM : 4G Processor : Intel ® Core i5 Speed : 320

**5.2 FLOW DIAGRAM:**

**FLOW DIAGRAM**

Dataflow is a managed service for executing a wide variety of data processing patterns. The documentation on this site shows you how to deploy your batch and streaming data processing pipelines using Dataflow, including directions for using service features.



**CHAPTER 6**

**SYSTEM IMPLEMENTATION**

**6.1 MODULES**

• Data collection

• Data pre-processing

• Feature extraction

• Evaluation model

**6.2 MODULES DESCRIPTION**

Data used in this paper is a set of product reviews collected from credit card transactions records. This step is concerned with selecting the subset of all available data that you will be working with. ML problems start with data preferably, lots of data (examples or observations) for which you already know the target answer. Data for which you already know the target answer is called labelled data.

**6.2.1 DATA COLLECTION**

Data used in this paper is a set of product reviews collected from credit card transactionrecords.Thisstep is concerned with selecting the subset of all available data that you will be working with. ML problems start with data preferably, lots of data (examples or observations) for which you already know the target answer. Data for which you already know the target answer is called labelled data.

17

**6.2.2DATA PRE-PROCESSING:**

Pre-processing is the process of three important and common steps as follows:

Formatting: It is the process of putting the data in a legitimate way that it would be suitable to work with. Format of the data files should be formatted according to the need. Most recommended format is .csv files. Cleaning: Data cleaning is a very important procedure in the path of data science as it constitutes the major part of the work. It includes removing missing data and complexity with naming category and so on. For most of the data scientists, Data Cleaning continues of 80% of work. Sampling: This is the technique of analyzing the subsets from whole large datasets, which could provide a better result and help in understanding the behavior and pattern of data in an integrated way. As there are no NAs nor duplicated variables, the preparation of the dataset was simple the first alteration that was made to be able to open the dataset on Weka program is changing the type of the class attribute from Numeric to Class and identify the class as {1,0} using the program Sublime Text. Another alteration was made on the type as well on the R program to be able to create the model and the visualization.

**6.2.3 Feature extraction:**

Feature extraction is the process of studying the behavior and pattern of the analyzed data and draw the features for further testing and training. Finally,our models are trained using the Classifier algorithm. We can classify module on Natural Language Toolkit library on Python. We use the labeled dataset gathered.

18

The rest of our labeled data will be used to evaluate the models. Some machine learning algorithms were used to classify pre-processed data. The chosen classifiers were Random forest. These algorithms are very popular in textclassificationtask.  
**6.2.4 Evaluation model:**

Evaluation model is an essential part of the model development process. It helps to find the best model that represents our data and how well the selected model will work in the future. Evaluating model performance with the data used for training is not acceptable in data science because it can effortlessly generate overoptimistically and over fitted models. To avoid overfitting, evaluation methods such as hold out and cross-validations are used to test to evaluate model performance. The result will be in the visualized form. Representation of classified data in the form of graphs. Accuracy is well-defined as the proportion of precise predictions for the test data. It can be calculated easily by mathematical calculation i.e. dividing the number of correct predictions by the number of total predictions.

**6.3 IMPLEMENTATION OF SYSTEM**

It gives an overview of implementation and explain how uses can navigate through the newly developed tool in order to use it easily.

**6.4 FORM INPUT & REPORT DESIGN**

The system makes use of encryption to secure transactional data using hashes to maintain a block of transactions in a chain manner which is maintained and verified by every node involved to verify the transaction and save money in a transparent way.

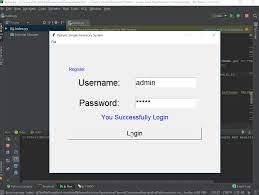
19

The system allows for a full-proof,secure,and authentic fund allocation and fund tracking system to help form an incorruptible government process.The whole process starts with the transaction done by the payer and ends with the transparent database which is publicly provided. Hence the details of the transaction such as payer, cashier,amount of the pay, why the transaction is done, all is noted and saved in the database. Furthermore a block having the transaction details, is added into the network. After the validation process, the block having transactional details with a checksum is then added into the machine learning network. All the transactions and the transactional details, thus published and will be added into the distributed ledger and will be available for the public in order to track the transactions. These forms were also kept as short and simple as possible for easy public awareness on the use of the tool, some of the forms and report.

**6.5 THE LOGIN FORM**

PLEASE LOG IN HERE BY ENTERING YOUR USERNAME AND PASSWORD CORRECTLY:

20



**CHAPTER-7**

**SYSTEM METHODOLOGY**

**Testing**

System Testing is a type of software testing that is performed on a completer integrated to evaluate the compliance of the system with the corresponding requirements. In system testing, integration testing passed components are taken as input. The goal of integration testing is to detect any irregularity between the units that are integrated together. System testing detects defects within both the integrated units and the whole system.

**7.1 TESTING METHODOLOGIES**

**7.1.1 UNIT TESTING:**

All the modules are being separately tested.

**7.1.2 SYSTEM TESTING:**

* Alpha-Testing
* Beta-Testing

**Alpha Testing**

Alpha Testing is a type of acceptance testing; performed to identify all possible issues and bugs before releasing the final product to the end users. Alpha testing is carried out by the testers who are internal employees of the organization. The main goal is to identify the tasks that a typical user might perform and test them.

22

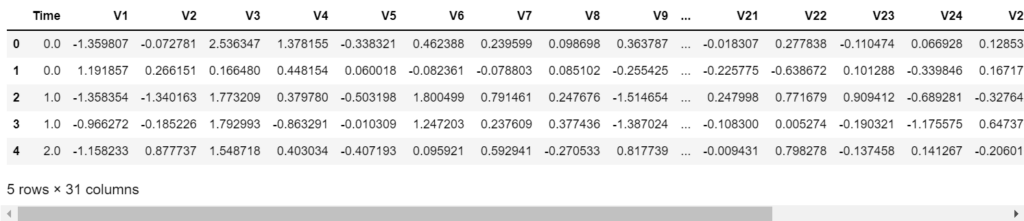
**Beta Testing**

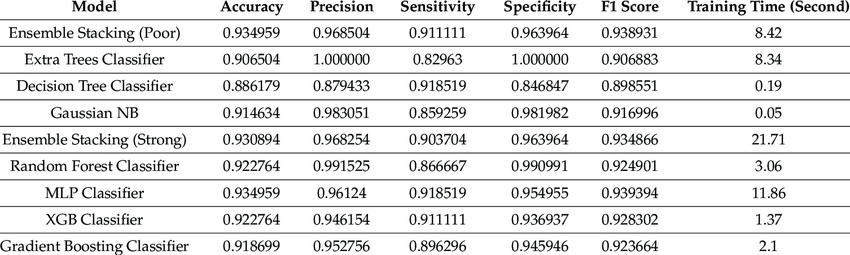
Beta Testing is performed by “real users” of the software application in “real environment” and it can be considered as a form of external User Acceptance Testing. It is the final test before shipping a product to the customers. Direct Feedback from customers is a major advantage of Beta Testing. This testing helps to test products in customer’s environment.

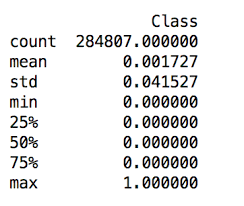
**CHAPTER 8**

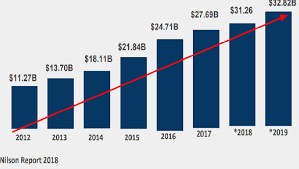
**RESULT**

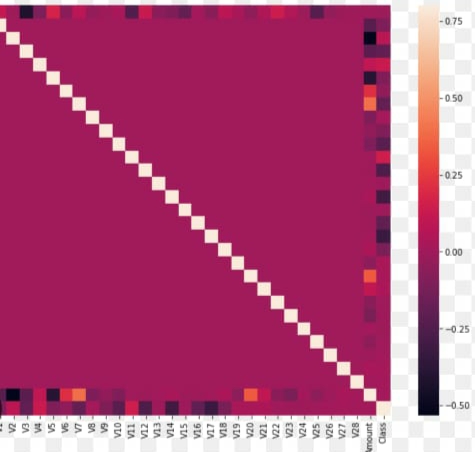
The data set is highly skewed, consisting of 492 frauds in a total of 284,807 observations. This resulted in only 0.172% fraud cases. This skewed set is justified by the low number of fraudulent transactions.The dataset consists of numerical values from the 28 ‘PrincipalComponent Analysis (PCA)’ transformed features, namely V1 to V28.Furthermore, there is no metadata about the original features provided,so pre-analysis or feature study could not be done.  
• The ‘Time’ and ‘Amount’ features are not transformed data.  
• There is no missing value in the dataset.

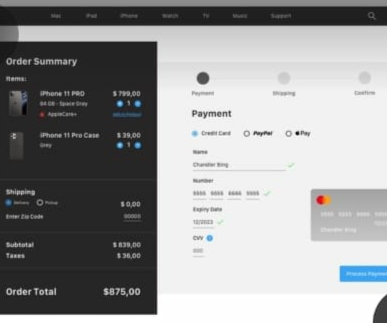






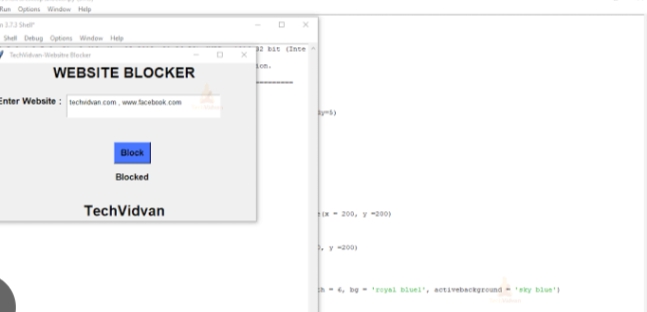








28



**CHAPTER 9**

**CONCLUSION**

Machine learning has been recognized as a successful measure for fraud detection. A great deal of data is transferred during online transaction processes, resulting in a binary result, genuine or fraudulent. In this process we estimate the precision, recall, f1score and support function for the dataset on all the models. In the case of Logistic Regression the 0 classes (transactions without fraud) are predicted with 100% precision and recall and the 1 classes (transactions which are fraud) are predicted with 88% precision. This means that 12% of the transactions which are fraudulent remain undetected by the system. But, 88% is still quite good. In the case of Decision Tree Classifier the the 0 classes (transactions without fraud) are predicted with 100% precision and recall and the 1 classes (transactions which are fraud) are predicted with 82% precision. This means that 18% of the transactions which are fraudulent remain undetected by the system. In the case of Gaussian Naive Bayes the 0 classes (transactions without fraud) are predicted with 100% precision and 98% recall and the 1 classes (transactions which are fraud) are predicted with 6% precision. This means that 94% of the transactions which are fraudulent remain undetected by the system. According to these results it is clear that the•precision with which the Logistic Regression model detects the transactions is much more than the decision tree classifier and the gaussian naive bayes. In this paper, we studied applications of machine learning like Naïve Bayes, Logistic regression, Random forest with boosting and shows that it proves accurate in deducting fraudulent transaction and minimizing the number of false alerts. Supervised learning algorithms are novel one in this literature in terms of application domain. If these algorithmsare applied into bank credit card fraud detection system, the probability of fraud transactions can be predicted soon after credit card transactions. And a series of anti-fraud strategies can be adopted to prevent banks from great losses and reduce risks. The objective of the study was taken differently than the typical classification problems in that we had a variable misclassification cost. Percision,recall.f1-score,support and accuracy are used to evaluate the performance for the proposed system. By comparing all the three methods, we found that random forest classifier with boosting technique is better than the logistic regression and naïve bayes methods.

**CHAPTER 10**

**FUTURE ENHANCEMENT**

From the above comparative analysis of the various credit card fraud detection techniques it is clear that Random Forest with Boosting technique performs best in this scenario. But the drawbacks of this paper by using the abovethree algorithms we cannot determine the names of fraud and unfraud transactions for the given dataset using machine learning. For the further development of the project we can work to solve this problem by using various methods.Preventing known and unknown fraud in real time is not easy but it is feasible.There are many ways to improve the model, such as using it on different datasets with various sizes, different data types or by changing the data splitting ratio, in addition to viewing it from different algorithm perspective. An example can be merging telecom data to calculate the location of people to have better knowledge of the location of the card owner while his/her credit card is being used, this will ease the detection because if the card owner is in Dubai and a transaction of his card was made in Abu Dhabi it will easily be detected as fraud. The Credit Card Fraud Detection Problem includes modelling past credit card transactions with the knowledge of the ones that turned out to be fraud. This model is then used to identify whether a new transaction is fraudulent or not. Our aim here is to detect 100% of the fraudulent transactions while minimizing the incorrect fraud classifications. In this paper, we studied applications of machine learning like Naïve Bayes, Logistic regression, Random forest with boosting and shows that it proves accurate in deducting fraudulent transaction and minimizing the number of false alerts. Supervised learning algorithms are novel one in this literature in terms of application domain. If these algorithmsare applied into bank credit card fraud detection system, the probability of fraud transactions can be predicted soon after credit card transactions. And a series of anti-fraud strategies can be adopted to prevent banks from great losses and reduce risks. The objective of the study was taken differently than the typical classification problems in that we had a variable misclassification cost. Percision,recall.f1-score,support and accuracy are used to evaluate the performance for the proposed system. By comparing all the three methods, we found that random forest classifier with boosting technique is better than the logistic regression and naïve bayes methods.

32

**CHAPTER 11**

**REFERENCE**

[1]Credit Card Fraud Detection Based on Transaction Behaviour -by John Richard D. Kho, Larry A. Vea published by Proc. of the 2017 IEEE Region 10 Conference (TENCON), Malaysia, November 5-8, 2017.

[2] L.J.P. van der Maaten and G.E. Hinton, Visualizing High-Dimensional Data Using t-SNE (2014), Journal of Machine Learning Research. Machine Learning Group — ULB, Credit Card Fraud Detection (2018)

[1] Introduction to Data Mining: Pang-Ning Tan & Michael Steinbach, Vipin Kumar, Pearson.

[2] Credit Card Fraud Detection:A case study by Ayushi Agarwal , Shiv Kumar , Amit Kumar Mishra(IEEE paper).

[3] Credit card fraud detection using Hidden Markov Model by DivyaIyer , Arti Mohanpurkar , Sneha Janardhan, Dhanashree Rathod , Amruta Sardeshmukh (IEEE paper) [4]https://www.youtube.com/watch?v=Z5WKQr4H4X k&t=26 6s [5]https://pandas.pydata.org/pandasdocs/stable/api.htm l#id5

[6]https://www.youtube.com/watch?time\_continue=42 0&v=XWo3nY06RgQ.

[7]https://www.quora.com/What-is-logisticregression/answer/Vinay-Kumar-R- 12? srid=3Okwy.

[8] https://youtu.be/953DXVpN5\_Q

[9]https://www.kaggle.com/agpickersgill/credit-cardfraud-detection/data [10] https://en.wikipedia.org/wiki/Credit\_card\_fraud [11] <https://docs.python.org/2/tutorial>

**CHAPTER 12**

**APPENDIX**

import pandas as pd

import numpy as np

import datetime

import smtplib

from twilio.rest import Client

# Load credit card transaction data

transaction\_data = pd.read\_csv('transaction\_data.csv') # Replace with your transaction data file path

# Set up Twilio credentials

TWILIO\_SID = 'your\_twilio\_sid'

TWILIO\_AUTH\_TOKEN = 'your\_twilio\_auth\_token'

TWILIO\_PHONE\_NUMBER = 'your\_twilio\_phone\_number'

AUTHORIZED\_PHONE\_NUMBER = 'authorized\_person\_phone\_number'

# Function to send alert call

def send\_alert\_call():

client = Client(TWILIO\_SID, TWILIO\_AUTH\_TOKEN)

client.calls.create(

to=AUTHORIZED\_PHONE\_NUMBER,

from\_=TWILIO\_PHONE\_NUMBER,

url='http://demo.twilio.com/docs/voice.xml', # Replace with your desired TwiML URL

method='GET'

)

print('Alert call sent to authorized person!')

# Loop through transactions and detect fraud

for index, transaction in transaction\_data.iterrows():

# Perform fraud detection logic here

if transaction['transaction\_amount'] > 1000:

# Suspicious transaction detected

# Send alert call to authorized person

send\_alert\_call()

# Send email notification

from\_email = 'your\_email@example.com'

to\_email = 'authorized\_person@example.com'

subject = 'Suspicious Credit Card Transaction Detected'

body = f'Transaction ID: {transaction["transaction\_id"]}\nTransaction Amount: {transaction["transaction\_amount"]}\nTransaction Date: {transaction["transaction\_date"]}\nTransaction Time: {transaction["transaction\_time"]}\nTransaction Location: {transaction["transaction\_location"]}'

email\_message = f'Subject: {subject}\n\n{body}'

server = smtplib.SMTP('smtp.gmail.com', 587)

server.starttls()

server.login(from\_email, 'your\_email\_password')

server.sendmail(from\_email, to\_email, email\_message)

server.quit()

# Print alert message

print('Suspicious Credit Card Transaction Detected:')

print('Transaction ID:', transaction['transaction\_id'])

print('Transaction Amount:', transaction['transaction\_amount'])

print('Transaction Date:', transaction['transaction\_date'])

print('Transaction Time:', transaction['transaction\_time'])

print('Transaction Location:', transaction['transaction\_location'])

print('Alert call and email notification sent to authorized person!')

print('------------------------------')

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

from sklearn.linear\_model import LogisticRegression

from sklearn.svm import SVC

from sklearn.neighbors import KNeighborsClassifier

from sklearn.tree import DecisionTreeClassifier

from sklearn.ensemble import RandomForestClassifier

from sklearn.linear\_model import SGDClassifier

from mlxtend.plotting import plot\_learning\_curves

from sklearn.model\_selection import train\_test\_split

from imblearn.over\_sampling import SMOTE

from sklearn.metrics import precision\_score, recall\_score, f1\_score, roc\_auc\_score, accuracy\_score, classification\_report

from sklearn.model\_selection import KFold, StratifiedKFold

from sklearn.preprocessing import StandardScaler

from sklearn.pipeline import Pipeline

from sklearn.model\_selection import GridSearchCV

from sklearn.metrics import make\_scorer, matthews\_corrcoef

import warnings

warnings.filterwarnings("ignore")

# Read Data into a Dataframe

df = pd.read\_csv('creditcard.csv')

# Describe Data

df.describe()

df.columns

Index(['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'],

dtype='object')

df.isna().sum()

def countplot\_data(data, feature):

'''

Method to compute countplot of given dataframe

Parameters:

data(pd.Dataframe): Input Dataframe

feature(str): Feature in Dataframe

'''

plt.figure(figsize=(10,10))

sns.countplot(x=feature, data=data)

plt.show()

def pairplot\_data\_grid(data, feature1, feature2, target):

'''

Method to construct pairplot of the given feature wrt data

Parameters:

data(pd.DataFrame): Input Dataframe

feature1(str): First Feature for Pair Plot

feature2(str): Second Feature for Pair Plot

target: Target or Label (y) '''

sns.FacetGrid(data, hue=target, size=6).map(plt.scatter, feature1, feature2).add\_legend()

plt.show()

countplot\_data(df, df.Class)

pairplot\_data\_grid(df, "Time", "Amount", "Class")

pairplot\_data\_grid(df, "Amount", "Time", "Class")

39

amount\_more = 0

amount\_less = 0

for i in range(df\_refine.shape[0]):

if(df\_refine.iloc[i]["Amount"] < 2500):

amount\_less += 1

else:

amount\_more += 1

print(amount\_more)

print(amount\_less)

fraud = 0

legitimate = 1

for i in range(df\_refine.shape[0]):

if(df\_refine.iloc[i]["Amount"]<2500):

if(df\_refine.iloc[i]["Class"] == 0):

legitimate += 1

else:

fraud+=1

print(fraud)

print(legitimate)

df\_refine = df[["Time", "Amount", "Class"]]

sns.pairplot(df\_refine, hue="Class", size=6)

plt.show()

df.Class.value\_counts()

sns.FacetGrid(df\_refine, hue="Class", size=6).map(sns.distplot,"Time").add\_legend()plt.show()

40

plt.figure(figsize=(20,20))

df\_corr = df.corr()

sns.heatmap(df\_corr)

# Use Synthetic Minority Oversampling

sm = SMOTE(random\_state=42)

X\_res, y\_res = sm.fit\_resample(X\_train, y\_train)

from sklearn.feature\_selection import mutual\_info\_classif

mutual\_infos = pd.Series(data=mutual\_info\_classif(X\_res, y\_res, discrete\_features=False, random\_state=1), index=ex=X\_dex=X\_tdex=X\_t

mutual\_infos.sort\_values(ascending=False)

sns.countplot(y\_res)

# Evaluation of Classifiers

def grid\_eval(grid\_clf):

"""

Method to Compute the best score and parameters computed by grid search

Parameter:

grid\_clf: The Grid Search Classifier

"""

print("Best Score", grid\_clf.best\_score\_)

print("Best Parameter", grid\_clf.best\_params\_)

def evaluation(y\_test, grid\_clf, X\_test):

""" Method to compute the following:

1. Classification Report

2. F1-score

41

3. AUC-ROC score

4. Accuracy

Parameters:

y\_test: The target variable test set

grid\_clf: Grid classifier selected

X\_test: Input Feature Test Set

"""

y\_pred = grid\_clf.predict(X\_test)

print('CLASSIFICATION REPORT')

print(classification\_report(y\_test, y\_pred))

print('AUC-ROC')

print(roc\_auc\_score(y\_test, y\_pred))

print('F1-Score')

print(f1\_score(y\_test, y\_pred))

print('Accuracy')

print(accuracy\_score(y\_test, y\_pred))

# The parameters of each classifier are different

# Hence, we do not make use of a single method and this is not to violate DRY Principles

# We set pipelines for each classifier unique with parameters

param\_grid\_sgd = [{

'model\_\_loss': ['log'],

42

'model\_\_penalty': ['l1', 'l2'],

'model\_\_alpha': np.logspace(start=-3, stop=3, num=20)

}, {

'model\_\_loss': ['hinge'],

'model\_\_alpha': np.logspace(start=-3, stop=3, num=20),

'model\_\_class\_weight': [None, 'balanced']

}]

pipeline\_sgd = Pipeline([

('scaler', StandardScaler(copy=False)),

('model', SGDClassifier(max\_iter=1000, tol=1e-3, random\_state=1, warm\_start=True))

])

MCC\_scorer = make\_scorer(matthews\_corrcoef)

grid\_sgd = GridSearchCV(estimator=pipeline\_sgd, param\_grid=param\_grid\_sgd, scoring=MCC\_scorer, n\_jobs=-1, pre\_dispatch='2\*n\_jobs', cv=5, verbose=1, return\_train\_score=False)

grid\_sgd.fit(X\_res, y\_res)

GridSearchCV(cv=5, error\_score=nan,

estimator=Pipeline(memory=None,

steps=[('scaler',

StandardScaler(copy=False,

with\_mean=True,

with\_std=True)),

('model', SGDClassifier(alpha=0.0001,

43

average=False,

class\_weight=None,

early\_stopping=False:

epsilon=0.1, eta0=0.0,

fit\_intercept=True,

l1\_ratio=0.15,

learning\_rate='optimal',

loss='hinge',

max\_iter=1000,

n\_iter\_no\_change=...

'model\_\_class\_weight': [None, 'balanced'],

'model\_\_loss': ['hinge']}],

pre\_dispatch='2\*n\_jobs', refit=True, return\_train\_score=False,

scoring=make\_scorer(matthews\_corrcoef), verbose=1)

grid\_eval(grid\_sgd)

evaluation(y\_test, grid\_sgd, X\_test)

pipeline\_rf = Pipeline([

('model', RandomForestClassifier(n\_jobs=-1, random\_state=1))

])

param\_grid\_rf = {'model\_\_n\_estimators': [75]}

grid\_rf = GridSearchCV(estimator=pipeline\_rf, param\_grid=param\_grid\_rf, scoring=MCC\_scorer, n\_jobs=-1, pre\_dispatch='2\*n\_jobs', cv=5, verbose=1, return\_train\_score=False)

grid\_rf.fit(X\_res, y\_res)

grid\_eval(grid\_rf)

44

evaluation(y\_test, grid\_rf, X\_test)

pipeline\_lr = Pipeline([

('model', LogisticRegression(random\_state=1))

])

param\_grid\_lr = {'model\_\_penalty': ['l2'],

'model\_\_class\_weight': [None, 'balanced']}

grid\_lr = GridSearchCV(estimator=pipeline\_lr, param\_grid=param\_grid\_lr, scoring=MCC\_scorer, n\_jobs=-1, pre\_dispatch='2\*n\_jobs', cv=5, verbose=1, return\_train\_score=False)

grid\_lr.fit(X\_res, y\_res)

grid\_eval(grid\_lr)

evaluation(y\_test, grid\_lr, X\_test)

pipeline\_knn = Pipeline([

('model', KNeighborsClassifier(n\_neighbors=5))

])

param\_grid\_knn = {'model\_\_p': [2]}

grid\_knn = GridSearchCV(estimator=pipeline\_knn, param\_grid=param\_grid\_knn, scoring=MCC\_scorer, n\_jobs=-1, pre\_dispatch='2\*n\_jobs', cv=5, verbose=1, return\_train\_score=False)

grid\_knn.fit(X\_res, y\_res)

grid\_eval(grid\_knn)

evaluation(y\_test, grid\_knn, X\_test)

45

|  |  |  |
| --- | --- | --- |
|  | **LIST OF ABBREVATIONS** |  |
| **S.NO** | **ABBREVIATION** | **EXPLANATION** |
| 1 | DOC | DISORDER OF CONSICIOUSNESS |
| 2 | ML | MACHINE LEARNING |
| 3 | PCA | PRINCIPAL COMPONENT ANALYSIS |
| 4 | EEG | ELECTROENCEPHALOGRAM |
| 5 | PSDD | POWER SPECTRAL DENSITY DIFFERENCE |
| 6 | URL | UNIFORM RESOURCE LOCATOR |