

BANK LOAN APPROVAL DATA ANALYZE AND PREDICTION USING DATA SCIENCE TECHNIQUE(ML)

Submitted in partial fulfillment of the requirements for the award of Bachelor
of Engineering degree in Computer Science and Engineering

by

GUNNAM SRINIVASU (Reg. No. 38110180)

ALLAM SHANKAR PAVAN KALYAN (Reg. No. 38110026)



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING**

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

MAY - 2022



SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **GUNNAM SRINIVASU (Reg. No: 38110180)**, **ALLAM SHANKAR PAVAN KALYAN (Reg. No: 38110026)** who carried out the project entitled "**BANK LOAN APPROVAL DATA ANALYZE AND PREDICTION USING DATA SCIENCE TECHNIQUE (ML)**" under my supervision from November 2021 to April 2022.

Internal Guide
Dr. M. MAHESWARI M.E., Ph.D.,

External Guide (if Applicable)

Head of the Department
Dr. L. LAKSHMANAN M.E., Ph.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

I **GUNNAM SRINIVASU, ALLAM SHANKAR PAVAN KALYAN** hereby declare that the Project Report entitled **BANK LOAN APPROVAL DATA ANALYZE AND PREDICTION USING DATA SCIENCE TECHNIQUE (ML)** done by me under the guidance of **Dr. M. Maheswari M.E., Ph.D., (Internal)** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering Degree in Computer Science and Engineering.

DATE: 15-03-2022

G. Srinivasu

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to Board of Management of SATHYABAMA for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to Dr. T. Sasikala M.E., Ph.D, Dean, School of Computing Dr. L. Lakshmanan M.E., Ph.D. , and Dr. S. Vigneshwari M.E., Ph.D. Head of the Department, Dept. of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide Dr. M. Maheswari ME., Ph.D for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the Department of Computer Science and Engineering who were helpful in many ways for the completion of the project.

ABSTRACT

Loans are no longer considered a last resort to buy a sought-after smartphone or a dream house. Over the last decade or so, people have become less hesitant in applying for a loan, whether it's personal, vehicle, education, business, or home especially when they don't have a lump sum at their disposal. Besides, Home and Education Loans provide tax advantages that reduce tax liability and increase the cash in hand from salary income. to get loans with minimal paperwork, quick eligibility checks, and competitive interest rates. They have opened an online channel to apply and submit documents for the approval process. If you still find the loan application and review process intimidating, credit history is indicative of your future repayment behaviour, based on your pattern in settling past loans. It helps the bank to know if you will be punctual and regular with your payments. Banks weigh your employment history and current engagement to ensure that your source of income is reliable.

TABLE OF CONTENTS

CHAPTER No.	TITLE	PAGE No
	ABSTRACT	i
	LIST OF FIGURES	v
1	INTRODUCTION	1
	1.1 Objective of the project	1
	1.1.1 Necessity	1
	1.1.2 Software development method	1
	1.1.3 Layout of the document	1
	1.2 Overview of the designed project	2
2	LITERATURE SURVEY	3
	2.1 Literature Survey	3
3	AIM AND SCOPE OF THE PRESENT INVESTIGATION	8
	3.1 Project Proposal	8
	3.1.1 Mission	8
	3.1.2 Goal	8
	3.2 Scope of the Project	8
	3.3 Overview of the project	8
	3.4 Existing system	9
	3.4.1 Disadvantages	9
	3.5 Preparing the dataset	9
	3.6 Proposed system	10
	3.6.1 Exploratory Data Analysis of loan approval	10
	3.6.2 Data Wrangling	10
	3.6.3 Data collection	10
	3.6.4 Building the classification model	10
	3.6.5 Advantages	11
	3.8 Flow chart	12

4	EXPERIMENTAL OR MATERIALS AND METHODS; ALGORITHMS USED	13
4.1	System Study	13
4.1.1	System requirement specifications	13
4.2	System Specifications	13
4.2.1	Machine Learning Overview	13
4.2.2	Flask Overview	14
4.3	Steps to download & install Python	14
4.3.1	IDE Installation for python	14
4.3.2	Python File Creation	14
4.4	Python Libraries needed	14
4.4.1	Numpy library	15
4.4.2	Pandas library	15
4.4.3	Matplotlib library	15
4.4.4	Seaborn library	16
4.4.5	Scikit Learn library	16
4.4.6	Flask	16
4.5	Modules	17
4.6	UML diagrams	18
4.6.1	Use Case Diagram	18
4.6.2	Class Diagram	19
4.6.3	Activity Diagram	20
4.6.4	Sequence Diagram	21
4.6.5	Entity Relationship Diagram	22
4.7	Module Details	23
4.7.1	Data Pre-processing	23
4.7.2	Data Validation /Cleaning /Preparing Process	24
4.7.3	Exploration data analysis of visualization	25
4.7.4	Comparing Algorithm with prediction in the form of best accuracy result	26
4.7.5	Algorithm and Techniques	30
4.7.6	Deployment Using Flask	40

5	RESULTS AND DISCUSSION, PERFORMANCE ANALYSIS	42
	5.1 Performance Analysis	42
	5.2 Discussion	42
6	SUMMARY AND CONCLUSION	43
	6.1 Summary	43
	6.2 Conclusion	43
	6.3 Future Work	43
	REFERENCES	44
	APPENDIX	
	A. SOURCE CODE	45-49
	B. SCREENSHOTS	50-52

LIST OF FIGURES

FIGURE NO.	FIGURE NAME	PAGE NO.
3.1	Architecture of Proposed Model	11
3.2	Flow chart	12
4.1	System Architecture	18
4.2	Use Case Diagrams	18
4.3	Class Diagram	19
4.4	Activity Diagram	20
4.5	Sequence Diagram	21
4.6	Entity Relationship Diagram	22
4.7	Logistic Regression	32
4.8	Random Forest Classifier	34
4.9	Decision Tree Classifier	35
4.10	Naïve Bayes Classifier	37
4.11	K-Nearest Neighbour	38
4.12	Support Vector Classifier	39
4.13	Gradient Boost	40
6.1	Machine Learning Algorithms Accuracy	50
6.2	Home Page	50
6.3	Inputs Page	51
6.4	Inputs Given By The User	51
6.5	Loan Approved	52
6.6	Loan Rejected	52

CHAPTER-1

INTRODUCTION

1.1 OBJECTIVE OF THE PROJECT:

The goal is to develop a machine learning model for Bank Loan Approval Prediction, to potentially replace the updatable supervised machine learning classification models by predicting results in the form of best accuracy by comparing supervised algorithm.

1.1.1 Necessity:

This online bank loan approval system helps in overcoming the time management. This Application is very easy to use. It can work accurately and very smoothly in a different scenario. It reduces the effort workload and increases efficiency in work. In aspects of time value, it is worthy. In this website the user can check the loan status easily whether approved or not.

1.1.2 Software development method:

In many software applications program different methods and cases are followed such as, Waterfall model, Iterative model, Spiral model, V-model and Big Bang model. I used waterfall model in this application. I tried to use test case and case software approaches.

1.1.3 Layout of the document:

This documentation starts with formal introduction. After introduction analysis and design of the project are described. In analysis and design of the project have many parts such as project proposal, mission, goal, target audience, environment. Use cases and test cases are in chapter 2 and chapter 3 respectively. Finally, this documentation finished with result and Conclusion part.

1.2 OVERVIEW OF THE DESIGNED PROJECT:

At first, we take the dataset from our resource then we have to perform data-preprocessing, visualization methods for cleaning and visualizing the dataset respectively and we applied the Machine Learning algorithms on the dataset then we generate the pickle file for best algorithm and flask is used as user interface for displaying the result.

CHAPTER-2

LITERATURE SURVEY

2.1 LITERATURE SURVEY:

General

A literature review is a body of text that aims to review the critical points of current knowledge on and/or methodological approaches to a particular topic. It is secondary sources and discuss published information in a particular subject area and sometimes information in a particular subject area within a certain time period. Its ultimate goal is to bring the reader up to date with current literature on a topic and forms the basis for another goal, such as future research that may be needed in the area and precedes a research proposal and may be just a simple summary of sources. Usually, it has an organizational pattern and combines both summary and synthesis.

A summary is a recap of important information about the source, but a synthesis is a re-organization, reshuffling of information. It might give a new interpretation of old material or combine new with old interpretations or it might trace the intellectual progression of the field, including major debates. Depending on the situation, the literature review may evaluate the sources and advise the reader on the most pertinent or relevant of them.

Review of Literature Survey

Title : A benchmark of machine learning approaches for credit score prediction.

Author: Vincenzo Moscato, Antonio Picariello, Giancarlo Sperli

Year : 2021

Credit risk assessment plays a key role for correctly supporting financial institutes in defining their bank policies and commercial strategies. Over the last decade, the emerging of social lending platforms has disrupted traditional services for credit risk assessment. Through these platforms, lenders and borrowers can easily interact among them without any involvement of financial institutes. In particular, they support borrowers in the fundraising process, enabling the participation of any number and size of lenders. However, the lack of lenders' experience and missing or uncertain information about

borrower's credit history can increase risks in social lending platforms, requiring an accurate credit risk scoring. To overcome such issues, the credit risk assessment problem of financial operations is usually modeled as a binary problem on the basis of debt's repayment and proper machine learning techniques can be consequently exploited. In this paper, we propose a benchmarking study of some of the most used credit risk scoring models to predict if a loan will be repaid in a P2P platform. We deal with a class imbalance problem and leverage several classifiers among the most used in the literature, which are based on different sampling techniques. A real social lending platform (Lending Club) data-set, composed by 877,956 samples, has been used to perform the experimental analysis considering different evaluation metrics (i.e. AUC, Sensitivity, Specificity), also comparing the obtained outcomes with respect to the state-of-the-art approaches. Finally, the three best approaches have also been evaluated in terms of their explainability by means of different eXplainable Artificial Intelligence (XAI) tools.

Title : An Approach for Prediction of Loan approval using Machine Learning Algorithm.

Author: Mohammad Ahmad Sheikh, Amit Kumar Goel, Tapas Kumar

Year : 2020

In our banking system, banks have many products to sell but main source of income of any banks is on its credit line. So they can earn from interest of those loans which they credits. A bank's profit or a loss depends to a large extent on loans i.e. whether the customers are paying back the loan or defaulting. By predicting the loan defaulters, the bank can reduce its NonPerforming Assets. This makes the study of this phenomenon very important. Previous research in this era has shown that there are so many methods to study the problem of controlling loan default. But as the right predictions are very important for the maximization of profits, it is essential to study the nature of the different methods and their comparison. A very important approach in predictive analytics is used to study the problem of predicting loan defaulters: The Logistic regression model. The data is collected from the Kaggle for studying and prediction. Logistic Regression models have been performed and the different measures of performances are computed. The models are compared on the basis of the performance measures such as sensitivity and

specificity. The final results have shown that the model produce different results. Model is marginally better because it includes variables (personal attributes of customer like age, purpose, credit history, credit amount, credit duration, etc.) other than checking account information (which shows wealth of a customer) that should be taken into account to calculate the probability of default on loan correctly. Therefore, by using a logistic regression approach, the right customers to be targeted for granting loan can be easily detected by evaluating their likelihood of default on loan. The model concludes that a bank should not only target the rich customers for granting loan but it should assess the other attributes of a customer as well which play a very important part in credit granting decisions and predicting the loan defaulters.

Title : Predict Loan Approval in Banking System Machine Learning Approach for Cooperative Banks Loan Approval.

Author: Amruta S. Aphale, Dr. Sandeep R. Shinde.

Year : 2020

In today's world, taking loans from financial institutions has become a very common phenomenon. Everyday a large number of people make application for loans, for a variety of purposes. But all these applicants are not reliable and everyone cannot be approved. Every year, we read about a number of cases where people do not repay bulk of the loan amount to the banks due to which they suffers huge losses. The risk associated with making a decision on loan approval is immense. So the idea of this project is to gather loan data from multiple data sources and use various machine learning algorithms on this data to extract important information. This model can be used by the organizations in making the right decision to approve or reject the loan request of the customers. In this paper, we examine a real bank credit data and conduct several machine learning algorithms on the data for that determine credit worthiness of customers in order to formulate bank risk automated system.

Title : Loan Approval Prediction Using Machine Learning

Author: Yash Divate, Prashant Rana, Pratik Chavan

Year : 2021

With the upgrade in the financial area loads of individuals are applying for bank advances however the bank has its restricted resources which it needs to allow to restricted individuals just, so discovering to whom the credit can be conceded which will be a more secure choice for the bank is a commonplace interaction. So in this task we attempt to decrease this danger factor behind choosing the protected individual in order to save bunches of bank endeavors and resources. This is finished by mining the Data of the past records of individuals to whom the advance was conceded previously and based on these records/encounters the machine was prepared utilizing the AI model which give the most precise outcome. The principle objective of this paper is to anticipate whether relegating the advance to specific individual will be protected or not. This paper is separated into four areas (i)Data Collection (ii) Comparison of AI models on gathered information (iii) Training of framework on most encouraging model (iv) Testing.

Title : Prediction for Loan Approval using Machine Learning Algorithm

Author: Ashwini S. Kadam, Shraddha R. Nikam, Ankita A. Aher, Gayatri V. Shelke, Amar S. Chandgude

Year : 2021

In our banking system, banks have many products to sell but main source of income of any banks is on its credit line. So they can earn from interest of those loans which they credits. A bank's profit or a loss depends to a large extent on loans i.e. whether the customers are paying back the loan or defaulting. By predicting the loan defaulters, the bank can reduce its Non-performing Assets. This makes the study of this phenomenon very important. Previous research in this era has shown that there are so many methods to study the problem of controlling loan default. But as the right predictions are very important for the maximization of profits, it is essential to study the nature of the different methods and their comparison. A very important approach in predictive analytics is used to study the problem of predicting loan defaulters (i) Collection of Data, (ii) Data Cleaning and (iii) Performance Evaluation. Experimental tests found that the Naïve Bayes model has better performance than other models in terms of loan forecasting.

Title : Modern Approach for Loan Sanctioning in Banks Using Machine Learning

Author: Golak Bihari Rath, Debasish Das, BiswaRanjan Acharya

Year : 2021

Loan analysis is a process adopted by banks used to check the credibility of loan applicants who can pay back the sanction loan amount within regulations and loan amount term mentioned by the bank. Most banks use their common recommended procedure of credit scoring and background check techniques to analyze the loan application and to make decisions on loan approval. This is overall a risk-oriented and a time-consuming process. In some cases, people suffer through financial problems while some intentionally try to fraud. As a result, such delay and default in payment by the loan applicants can lead to loss of capital of the banks. Hence to overcome this, banks need to adopt a better procedure to find the trustworthy applicants for granting loan from the list of all applicants applied for the loan, who can pay can their loan amount in stipulated time. In the modern-day age and advance of technology, we adopt a machine learning approach to reduce the risk factor and human errors in the loan sanction process and determine where an applicant is eligible for loan approval or not. Here, we examine various features such as applicant income, credit history, education from past records of loan applicants irrespective of their loan sanction, and the best features are determined and selected which have a direct impact on the outcome for loan approval.

CHAPTER-3

AIM AND SCOPE OF THE PRESENT INVESTIGATION

3.1 PROJECT PROPOSAL:

The project proposal is the term of documents. A project can describe the project proposal. It is the set of all plans of a project. Like, how the software works, what are the steps to complete the entire projects, and what are the software requirements and analysis for this project. In my project, I am doing all the steps and also risk and reward and other project dependencies in the project proposal.

3.1.1 Mission:

An online Web based machine learning application is very popular and well known to everyone. Now a day's everybody wants to get it and work with it. Loan prediction is mostly useful for bank employees in approving the loan application. This simple method gives fast and accurate results in approving the customer application.

3.1.2 Goal:

The goal is to develop a machine learning model for Bank Loan Approval Prediction.

3.2 SCOPE OF THE PROJECT:

The scope of this paper is to implement and investigate how different supervised binary classification methods impact default prediction. The model evaluation techniques used in this project are limited to precision, sensitivity, F1-score.

3.3 OVERVIEW OF THE PROJECT:

The overview of the project is to provide a web-based machine learning application to the user. Therefore, the user can directly check the loan approval in our website over the internet. So, the user can easily check into this loan system prediction whether their loan will be approved or not.

3.4 EXISTING SYSTEM:

Anomaly detection relies on individuals' behavior profiling and works by detecting any deviation from the norm. When used for online banking fraud detection, however, it mainly suffers from three disadvantages. First, for an individual, the historical behavior data are often too limited to profile his/her behavior pattern. Second, due to the heterogeneous nature of transaction data, there lacks a uniform treatment of different kinds of attribute values, which becomes a potential barrier for model development and further usage.

Third, the transaction data are highly skewed, and it becomes a challenge to utilize the label information effectively. Anomaly detection often suffers from poor generalization ability and a high false alarm rate. We argue that individuals' limited historical data for behavior profiling and the highly skewed nature of fraud data could account for this defect. Since it is straightforward to use information from other similar individuals, measuring similarity itself becomes a great challenge due to heterogeneous attribute values. We propose to transform the anomaly detection problem into a pseudo-recommender system problem and solve it with an embedding based method. By doing so, the idea of collaborative filtering is implicitly used to utilize information from similar users, and the learned preference matrices and attribute embedding provide a concise way for further usage.

3.4.1 Disadvantages:

1. They had proposed a mathematical model and machine learning algorithms is not used.
2. Class Imbalance problem was not addressed and the proper measure were not taken.

3.5 PREPARING THE DATASET:

This dataset contains 665 records of features extracted from Bank Loan datas, which were then classified into 2 classes:

- Approve
- Reject

3.6 PROPOSED SYSTEM:

3.6.1 Exploratory Data Analysis of loan approval

Multiple datasets from different sources would be combined to form a generalized dataset, and then different machine learning algorithms would be applied to extract patterns and to obtain results with maximum accuracy.

3.6.2 Data Wrangling

In this section of the report will load in the data, check for cleanliness, and then trim and clean given dataset for analysis. Make sure that the document steps carefully and justify for cleaning decisions.

3.6.3 Data collection

The data set collected for predicting given data is split into Training set and Test set. Generally, 7:3 ratios are applied to split the Training set and Test set. The Data Model which was created using machine learning algorithms are applied on the Training set and based on the test result accuracy, Test set prediction is done.

3.6.4 Building the classification model

The predicting the loan approval, ML algorithm prediction model is effective because of the following reasons: It provides better results in classification problem.

- It is strong in preprocessing outliers, irrelevant variables, and a mix of continuous, categorical and discrete variables.
- It produces out of bag estimate error which has proven to be unbiased in many tests and it is relatively easy to tune with.

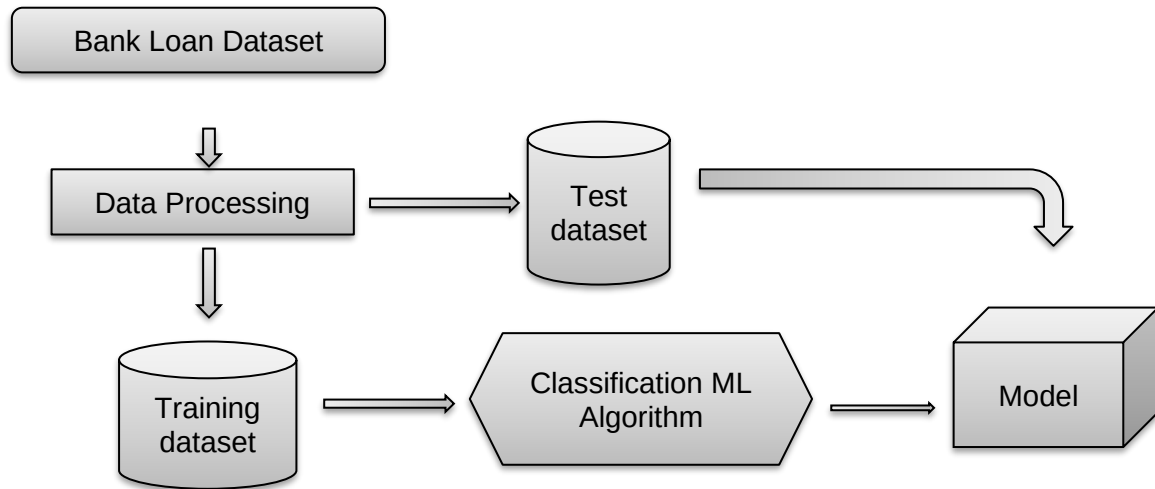


Fig:3.1: Architecture of Proposed model

3.6.5 Advantages:

- Performance and accuracy of the algorithms can be calculated and compared.
- Class imbalance can be dealt with machine learning approaches.

3.8 FLOW CHART:

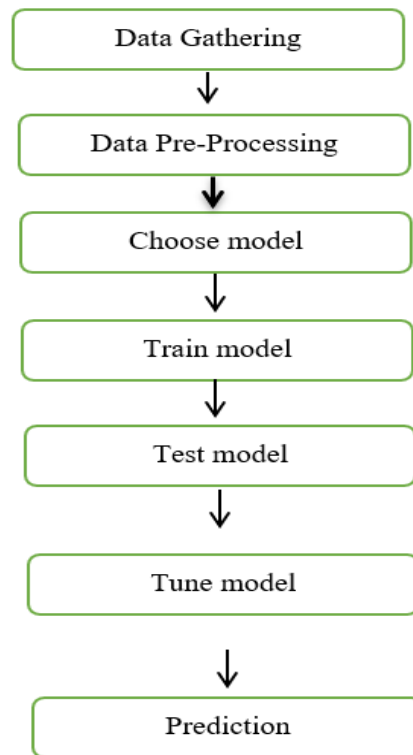


Fig:3.2: FLOW CHART

CHAPTER-4

EXPERIMENTAL OR MATERIALS AND METHODS

ALGORITHMS USED

4.1 SYSTEM STUDY:

To develop this model we use new modern technologies which are Machine Learning using Python for predicting and Flask is for user interface.

4.1.1 System requirement specifications:

a) Hardware requirements:

- Processor : Intel
- RAM : 2GB
- Hard Disk : 80GB

b) Software requirements:

- OS : Windows
- Framework : Flask
- Technology : Machine Learning using Python
- Web Browser : Chrome, Microsoft Edge
- Code editor : Visual Studio Code, Google Colab, Anaconda or Jupyter notebook.

4.2 SYSTEM SPECIFICATIONS:

4.2.1 Machine Learning Overview:

Machine learning is a field of study that looks at using computational algorithms to turn empirical data into usable models. The machine learning field grew out of traditional statistics and artificial intelligences communities. Through their business processes immense amounts of data have been and will be collected. This has provided an opportunity to re-invigorate the statistical and computational approaches to autogenerate useful models from data. Machine learning algorithms can be used to (a) gather understanding of the cyber phenomenon that produced the data under study, (b) abstract the

understanding of underlying phenomena in the form of a model, (c) predict future values of a phenomena using the above-generated model, and (d) detect anomalous behavior exhibited by a phenomenon under observation.

4.2.2 Flask Overview:

Flask is an API of Python that allows us to build up web-applications. It was developed by Armin Ronacher. Flask's framework is more explicit than Django's framework and is also easier to learn because it has less base code to implement a simple web-Application.

4.3 STEPS TO DOWNLOAD & INSTALL PYTHON:

Download the Latest version of the **Python** executable installer (<https://www.python.org/downloads/>). Watch the PIP list where pip is the package installer for python. Now upgrade the pip and setuptools using the command

Pip install --upgrade pip and Pip install --upgrade setuptools

4.3.1 IDE INSTALLATION FOR PYTHON

IDE stands for Integrated Development Environment. It is a GUI (Graphical User Interface) where programmers write their code and produce the final products. Best IDE is Pycharm. So download the pycharm new version and install the software (<https://www.jetbrains.com/pycharm/download/>)

4.3.2 PYTHON FILE CREATION

GO To FILE MENU > CREATE > NEW > PYTHON FILE
>(Name Your Python File as "HOUSE PRICE PPREDICTION" > SAVE

4.4 PYTHON LIBRARIES NEEDED

There are many libraries in python. In those we only use few main libraries needed.

4.4.1 NUMPY LIBRARY

NumPy is an open-source numerical Python library. NumPy contains a multi-dimensional array and matrix data structures. It can be utilized to perform a number of mathematical operations on arrays such as trigonometric, statistical, and algebraic routines like mean, mode, standard deviation etc...,

Installation- (<https://numpy.org/install/>)

```
pip install NUMPY
```

Here we mainly use array, to find mean and standard deviation.

4.4.2 PANDAS LIBRARY

Pandas is a high-level data manipulation tool developed by Wes McKinney. It is built on the Numpy package and its key data structure is called the DataFrame. DataFrames allow you to store and manipulate tabular data in rows of observations and columns of variables. There are several ways to create a DataFrame.

Installation- (https://pandas.pydata.org/getting_started.html)

```
pip install PANDAS
```

Here we use pandas for reading the csv files, for grouping the data, for cleaning the data using some operations.

4.4.3 MATPLOTLIB LIBRARY

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible. Use interactive figures that can zoom, pan, update, visualize etc.,

Installation- (<https://matplotlib.org/users/installing.html>)

```
pip install Matplotlib
```

Here we use pyplot mainly for plotting graphs.

matplotlib.pyplot is a collection of functions that make matplotlib work like MATLAB. Each pyplot function makes some change to a figure: e.g., creates a figure, creates a plotting area in a figure, plots some lines in a plotting area, decorates the plot with labels, etc.

4.4.4 SEABRON LIBRARY

Seaborn package was developed based on the Matplotlib library. It is used to create more attractive and informative statistical graphics. While seaborn is a different package, it can also be used to develop the attractiveness of matplotlib graphics.

Installation-(<https://seaborn.pydata.org/installing.html>)

```
pip install Seaborn
```

4.4.5 SCIKIT-LEARN LIBRARY

Scikit-learn is a free machine learning library for the Python. It features various algorithms like support vector machine, random forests, regression and k-neighbors, and it also supports Python numerical and scientific libraries like NumPy and SciPy.

```
Pip install Scikit-Learn
```

Installation-(<https://scikit-learn.org/stable/install.html>)

Here use scikit-learn's regression methods for prediction purpose.

4.4.6 FLASK

Flask is an API of Python that allows us to build up web-applications. It was developed by Armin Ronacher. Flask's framework is more

explicit than Django's framework and is also easier to learn because it has less base code to implement a simple web-Application.

```
pip install flask
```

Here we use flask for the user-interface.

4.5 MODULES:

A modular design reduces complexity, facilitates change (a critical aspect of software maintainability), and results in easier implementation by encouraging parallel development of different parts of system. Software with effective modularity is easier to develop because functions may be compartmentalized and interfaces are simplified. Software architecture embodies modularity that is software is divided into separately named and addressable components called modules that are integrated to satisfy problem requirements.

Modularity is the single attribute of software that allows a program to be intellectually manageable. The five important criteria that enable us to evaluate a design method with respect to its ability to define an effective modular design are: Modular decomposability, Modular Comprehensibility, Modular Understandability, Modular continuity, Modular Protection.

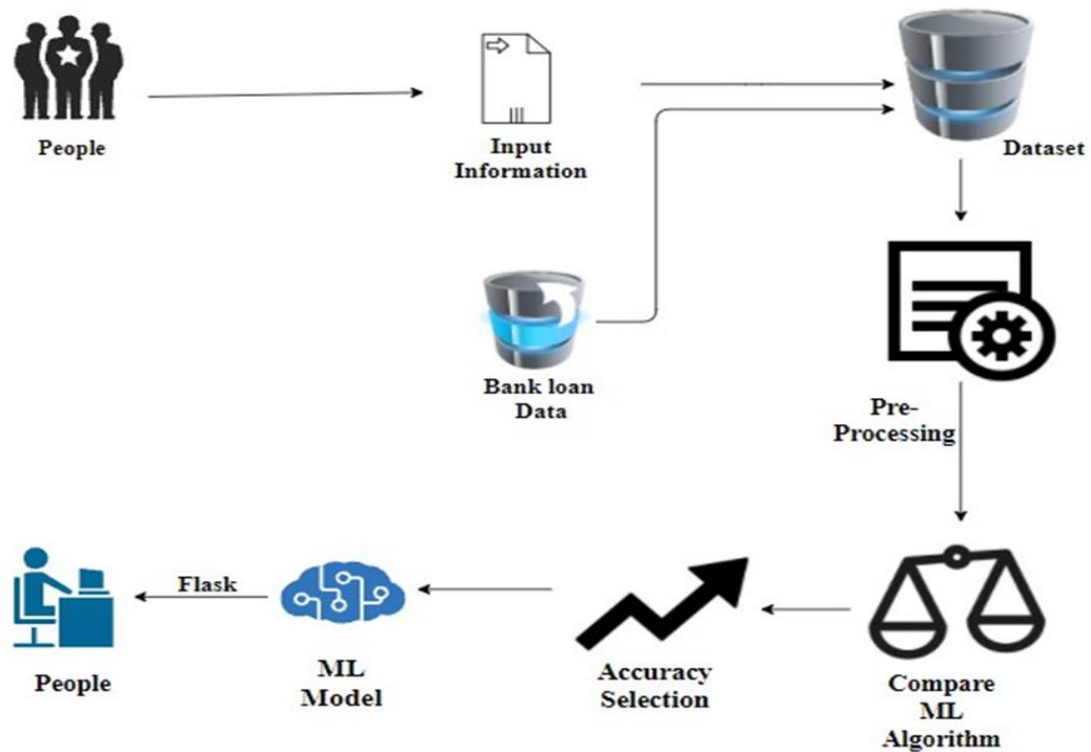


Fig:4.1: SYSTEM ARCHITECTURE

4.6 UML DIAGRAMS

4.6.1 Use Case Diagram

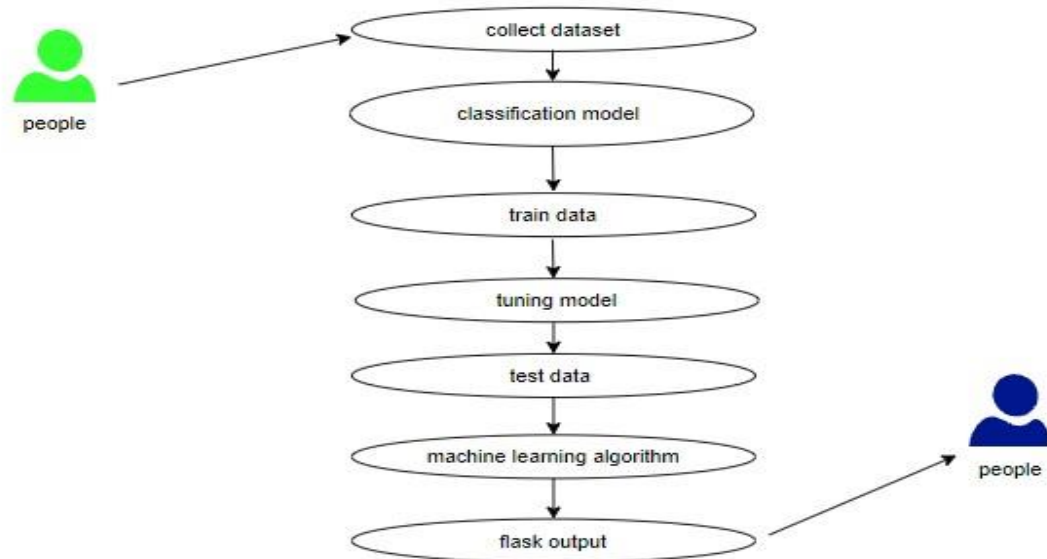


Fig:4.2: USE CASE DIAGRAM

Use case diagrams are considered for high level requirement analysis of a system. So when the requirements of a system are analyzed the functionalities are captured in use cases. So, it can say that uses cases are nothing but the system functionalities written in an organized manner.

4.6.2 Class Diagram

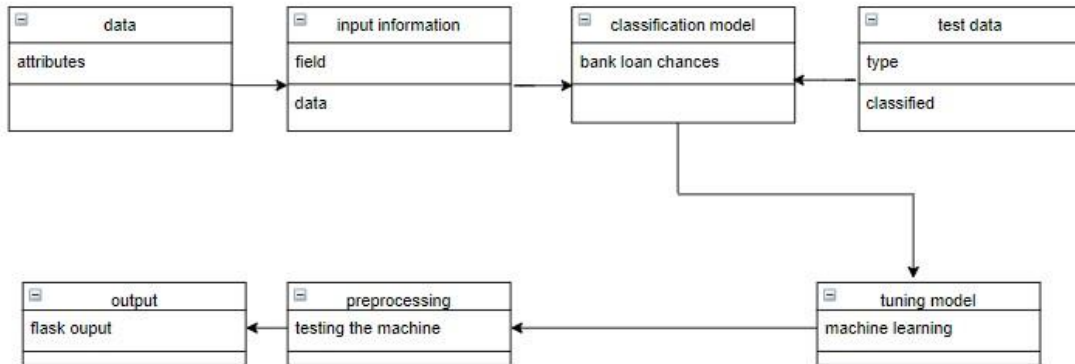


Fig:4.3: CLASS DIAGRAM

Class diagram is basically a graphical representation of the static view of the system and represents different aspects of the application. So a collection of class diagrams represent the whole system. The name of the class diagram should be meaningful to describe the aspect of the system. Each element and their relationships should be identified in advance Responsibility (attributes and methods) of each class should be clearly identified for each class minimum number of properties should be specified and because, unnecessary properties will make the diagram complicated. Use notes whenever required to describe some aspect of the diagram and at the end of the drawing it should be understandable to the developer/coder. Finally, before making the final version, the diagram should be drawn on plain paper and rework as many times as possible to make it correct.

4.6.3 Activity Diagram

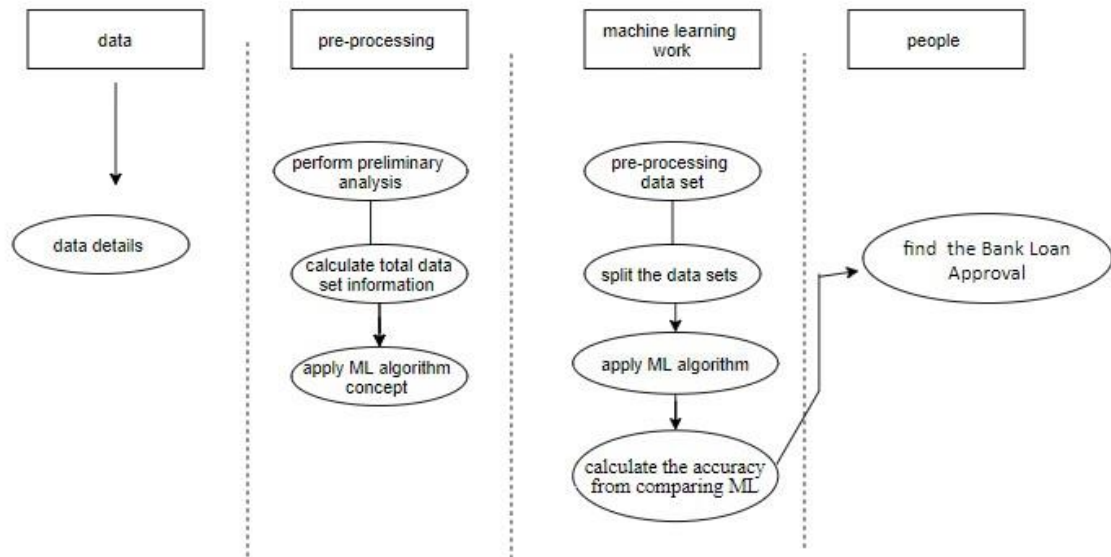


Fig:4.4: ACTIVITY DIAGRAM

Activity is a particular operation of the system. Activity diagrams are not only used for visualizing dynamic nature of a system but they are also used to construct the executable system by using forward and reverse engineering techniques. The only missing thing in activity diagram is the message part. It does not show any message flow from one activity to another. Activity diagram is some time considered as the flow chart. Although the diagrams looks like a flow chart but it is not. It shows different flow like parallel, branched, concurrent and single.

4.6.4 Sequence Diagram

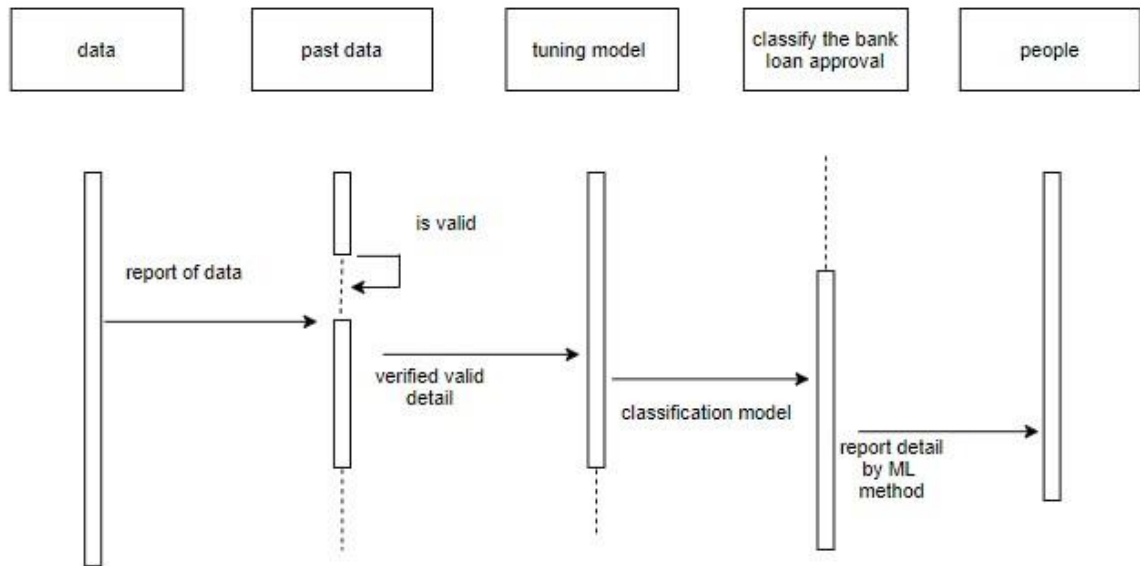


Fig:4.5: SEQUENCE DIAGRAM

Sequence diagrams model the flow of logic within your system in a visual manner, enabling you both to document and validate your logic, and are commonly used for both analysis and design purposes. Sequence diagrams are the most popular UML artifact for dynamic modeling, which focuses on identifying the behavior within your system. Other dynamic modeling techniques include activity diagramming, communication diagramming, timing diagramming, and interaction overview diagramming. Sequence diagrams, along with class diagrams and physical data models are in my opinion the most important design-level models for modern business application development.

4.6.5 Entity Relationship Diagram (ERD)

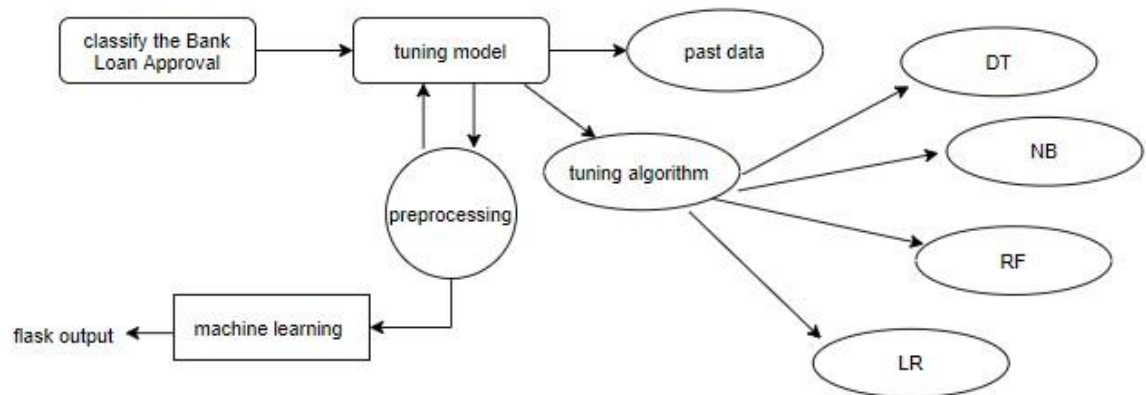


Fig:4.6: ENTITY RELATIONSHIP DIAGRAM

An entity relationship diagram (ERD), also known as an entity relationship model, is a graphical representation of an information system that depicts the relationships among people, objects, places, concepts or events within that system. An ERD is a data modelling technique that can help define business processes and be used as the foundation for a relational database. Entity relationship diagrams provide a visual starting point for database design that can also be used to help determine information system requirements throughout an organization. After a relational database is rolled out, an ERD can still serve as a referral point, should any debugging or business process re-engineering be needed later.

The following are the modules of the project, which is planned in aid to complete the project with respect to the proposed system, while overcoming existing system and also providing the support for the future enhancement.

4.7 MODULE DETAILS:

4.7.1 *Data Pre-processing*

Validation techniques in machine learning are used to get the error rate of the Machine Learning (ML) model, which can be considered as close to the true error rate of the dataset. If the data volume is large enough to be representative of the population, you may not need the validation techniques. However, in real-world scenarios, to work with samples of data that may not be a true representative of the population of given dataset. To finding the missing value, duplicate value and description of data type whether it is float variable or integer. The sample of data used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyper parameters.

The evaluation becomes more biased as skill on the validation dataset is incorporated into the model configuration. The validation set is used to evaluate a given model, but this is for frequent evaluation. It as machine learning engineers use this data to fine-tune the model hyper parameters. Data collection, data analysis, and the process of addressing data content, quality, and structure can add up to a time-consuming to-do list. During the process of data identification, it helps to understand your data and its properties; this knowledge will help you choose which algorithm to use to build your model.

A number of different data cleaning tasks using Python Pandas library and specifically, it focuses on probably the biggest data cleaning task, missing values and it able to more quickly clean data. It wants to spend less time cleaning data, and more time exploring and modeling.

Some of these sources are just simple random mistakes. Other times, there can be a deeper reason why data is missing. It's important to understand these different types of missing data from a statistics point of view. The type of missing data will influence how to deal with filling in the missing values and to detect missing values, and do some basic imputation and detailed statistical approach for dealing with missing data. Before, joint into code, it's important to understand the sources of missing data. Here are some typical reasons why data is missing:

- User forgot to fill in a field.
- Data was lost while transferring manually from a legacy database.
- There was a programming error.
- Users chose not to fill out a field tied to their beliefs about how the results would be used or interpreted.

Variable identification with Uni-variate, Bi-variate and Multi-variate analysis:

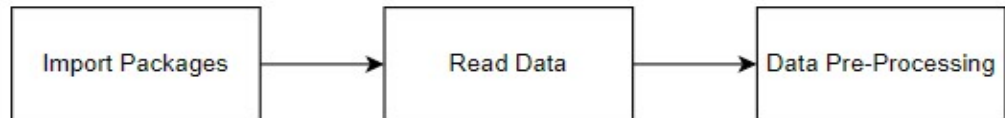
- import libraries for access and functional purpose & read the given dataset
- General Properties of Analyzing the given dataset
- Display the given dataset in the form of data frame
- show columns
- shape of the data frame
- To describe the data frame
- Checking data type and information about dataset
- Checking for duplicate data
- Checking Missing values of data frame
- Checking unique values of data frame
- Checking count values of data frame
- Rename and drop the given data frame
- To specify the type of values
- To create extra columns

4.7.2 Data Validation/ Cleaning/Preparing Process

Importing the library packages with loading given dataset. To analyzing the variable identification by data shape, data type and evaluating the missing values, duplicate values. A validation dataset is a sample of data held back from training your model that is used to give an estimate of model skill while tuning model's and procedures that you can use to make the best use of validation and test datasets when evaluating your models. Data cleaning / preparing by rename the given dataset and drop the column etc. to analyze the uni-variate, bi-variate and multi-variate process.

The steps and techniques for data cleaning will vary from dataset to dataset. The primary goal of data cleaning is to detect and remove errors and anomalies to increase the value of data in analytics and decision making.

MODULE DIAGRAM



GIVEN INPUT EXPECT OUTPUT

input: data

output: removing noisy data

4.7.3 Exploration data analysis of visualization

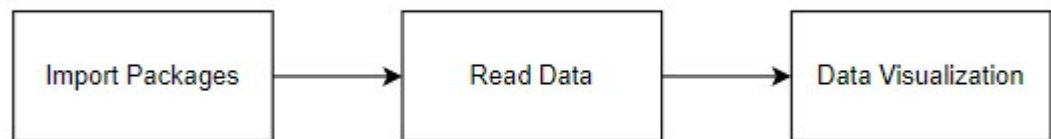
Data visualization is an important skill in applied statistics and machine learning. Statistics does indeed focus on quantitative descriptions and estimations of data. Data visualization provides an important suite of tools for gaining a qualitative understanding. This can be helpful when exploring and getting to know a dataset and can help with identifying patterns, corrupt data, outliers, and much more. With a little domain knowledge, data visualizations can be used to express and demonstrate key relationships in plots and charts that are more visceral and stakeholders than measures of association or significance. Data visualization and exploratory data analysis are whole fields themselves and it will recommend a deeper dive into some the books mentioned at the end.

Sometimes data does not make sense until it can look at in a visual form, such as with charts and plots. Being able to quickly visualize of data samples and others is an important skill both in applied statistics and in applied machine learning. It will discover the many types of plots that you will need to know when visualizing data in

Python and how to use them to better understand your own data.

- How to chart time series data with line plots and categorical quantities with bar charts.
- How to summarize data distributions with histograms and box plots.

MODULE DIAGRAM



GIVEN INPUT EXPECT OUTPUT

input: data

output: visualized data

4.7.4 Comparing Algorithm with prediction in the form of best accuracy result

It is important to compare the performance of multiple different machine learning algorithms consistently and it will discover to create a test harness to compare multiple different machine learning algorithms in Python with scikit-learn. It can use this test harness as a template on your own machine learning problems and add more and different algorithms to compare. Each model will have different performance characteristics. Using resampling methods like cross validation, you can get an estimate for how accurate each model may be on unseen data. It needs to be able to use these estimates to choose one or two best models from the suite of models that you have created. When have a new dataset, it is a good idea to visualize the data using different techniques in order to look at the data from different perspectives. The same idea applies to model selection. You should use a number of different ways of looking at the estimated accuracy of your machine learning algorithms in order to choose the one or two to finalize. A way to do this is

to use different visualization methods to show the average accuracy, variance and other properties of the distribution of model accuracies.

In the next section you will discover exactly how you can do that in Python with scikit-learn. The key to a fair comparison of machine learning algorithms is ensuring that each algorithm is evaluated in the same way on the same data and it can achieve this by forcing each algorithm to be evaluated on a consistent test harness.

Pre-processing refers to the transformations applied to our data before feeding it to the algorithm. Data Preprocessing is a technique that is used to convert the raw data into a clean data set. In other words, whenever the data is gathered from different sources it is collected in raw format which is not feasible for the analysis. To achieving better results from the applied model in Machine Learning method of the data has to be in a proper manner. Some specified Machine Learning model needs information in a specified format, for example, Random Forest algorithm does not support null values. Therefore, to execute random forest algorithm null values have to be managed from the original raw data set. And another aspect is that data set should be formatted in such a way that more than one Machine Learning and Deep Learning algorithms are executed in given dataset.

In the example below these 7 different algorithms are compared:

- Logistic Regression
- Random Forest
- Decision Tree Classifier
- Naïve Bayes
- Support Vector Classifier
- K Nearest Neighbor
- Gradient Boost

The K-fold cross validation procedure is used to evaluate each algorithm, importantly configured with the same random seed to ensure that the same splits to the training data are performed and that each algorithm is evaluated in precisely

the same way. Before that comparing algorithm, Building a Machine Learning Model using install Scikit-Learn libraries. In this library package have to done preprocessing, linear model with logistic regression method, cross validating by KFold method, ensemble with random forest method and tree with decision tree classifier. Additionally, splitting the train set and test set. To predicting the result by comparing accuracy.

False Positives (FP): A person who will pay predicted as defaulter. When actual class is no and predicted class is yes. E.g. if actual class says this passenger did not survive but predicted class tells you that this passenger will survive.

False Negatives (FN): A person who default predicted as payer. When actual class is yes but predicted class in no. E.g. if actual class value indicates that this passenger survived and predicted class tells you that passenger will die.

True Positives (TP): A person who will not pay predicted as defaulter. These are the correctly predicted positive values which means that the value of actual class is yes and the value of predicted class is also yes. E.g. if actual class value indicates that this passenger survived and predicted class tells you the same thing.

True Negatives (TN): A person who default predicted as payer. These are the correctly predicted negative values which means that the value of actual class is no and value of predicted class is also no. E.g. if actual class says this passenger did not survive and predicted class tells you the same thing.

Prediction result by accuracy:

Logistic regression algorithm also uses a linear equation with independent predictors to predict a value. The predicted value can be anywhere between negative infinity to positive infinity. It needs the output of the algorithm to be classified variable data. Higher accuracy predicting result is logistic regression model by comparing the best accuracy.

True Positive Rate (TPR) = $TP / (TP + FN)$

False Positive Rate (FPR) = $FP / (FP + TN)$

Accuracy: The Proportion of the total number of predictions that is correct otherwise overall how often the model predicts correctly defaulters and non-defaulters.

Accuracy calculation:

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

Accuracy is the most intuitive performance measure and it is simply a ratio of correctly predicted observation to the total observations. One may think that, if we have high accuracy then our model is best. Yes, accuracy is a great measure but only when you have symmetric datasets where values of false positive and false negatives are almost same.

Precision: The proportion of positive predictions that are actually correct.

Precision = $TP / (TP + FP)$

Precision is the ratio of correctly predicted positive observations to the total predicted positive observations. The question that this metric answer is of all passengers that labeled as survived, how many actually survived? High precision relates to the low false positive rate. We have got 0.788 precision which is pretty good.

Recall: The proportion of positive observed values correctly predicted. (The proportion of actual defaulters that the model will correctly predict)

Recall = $TP / (TP + FN)$

Recall (Sensitivity) - Recall is the ratio of correctly predicted positive observations to the all observations in actual class - yes.

F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. Intuitively it is not as

easy to understand as accuracy, but F1 is usually more useful than accuracy, especially if you have an uneven class distribution. Accuracy works best if false positives and false negatives have similar cost. If the cost of false positives and false negatives are very different, it's better to look at both Precision and Recall.

General Formula:

$$F\text{- Measure} = 2TP / (2TP + FP + FN)$$

F1-Score Formula:

$$F1 \text{ Score} = 2 * (\text{Recall} * \text{Precision}) / (\text{Recall} + \text{Precision})$$

4.7.5 ALGORITHM AND TECHNIQUES

Algorithm Explanation

In machine learning and statistics, classification is a supervised learning approach in which the computer program learns from the data input given to it and then uses this learning to classify new observation. This data set may simply be bi-class (like identifying whether the person is male or female or that the mail is spam or non-spam) or it may be multi-class too. Some examples of classification problems are: speech recognition, handwriting recognition, bio metric identification, document classification etc. In Supervised Learning, algorithms learn from labeled data. After understanding the data, the algorithm determines which label should be given to new data based on pattern and associating the patterns to the unlabeled new data.

Used Python Packages:

sklearn:

- In python, sklearn is a machine learning package which include a lot of ML algorithms.
- Here, we are using some of its modules like `train_test_split`, `DecisionTreeClassifier` or `Logistic Regression` and `accuracy_score`.

NumPy:

- It is a numeric python module which provides fast maths functions for calculations.
- It is used to read data in numpy arrays and for manipulation purpose.

Pandas:

- Used to read and write different files.
- Data manipulation can be done easily with data frames.

Matplotlib:

- Data visualization is a useful way to help with identify the patterns from given dataset.
- Data manipulation can be done easily with data frames.

Logistic Regression:

It is a statistical method for analyzing a data set in which there are one or more independent variables that determine an outcome. The outcome is measured with a dichotomous variable (in which there are only two possible outcomes). The goal of logistic regression is to find the best fitting model to describe the relationship between the dichotomous characteristic of interest (dependent variable = response or outcome variable) and a set of independent (predictor or explanatory) variables. Logistic regression is a Machine Learning classification algorithm that is used to predict the probability of a categorical dependent variable. In logistic regression, the dependent variable is a binary variable that contains data coded as 1 (yes, success, etc.) or 0 (no, failure, etc.).

In other words, the logistic regression model predicts $P(Y=1)$ as a function of X .

Logistic regression Assumptions:

- Binary logistic regression requires the dependent variable to be binary.
- For a binary regression, the factor level 1 of the dependent variable should represent the desired outcome.

- Only the meaningful variables should be included.
- The independent variables should be independent of each other. That is, the model should have little.
- The independent variables are linearly related to the log odds.
- Logistic regression requires quite large sample sizes.

MODULE DIAGRAM

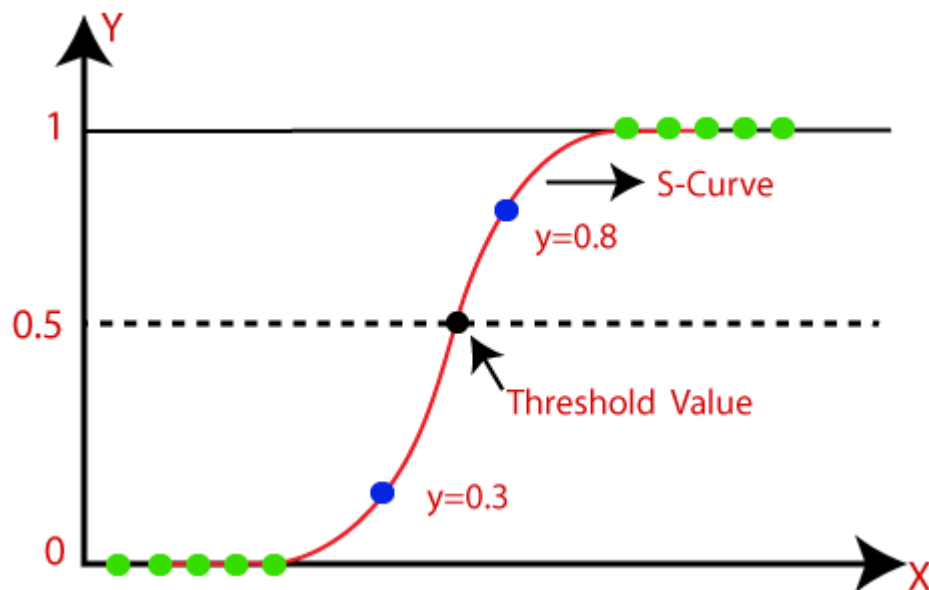
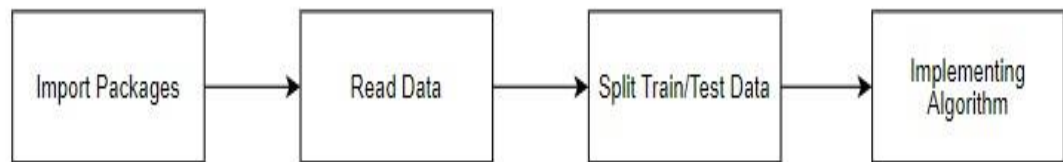


Fig:4.7: LOGISTIC REGRESSION

GIVEN INPUT EXPECT OUTPUT

input: data

output: getting accuracy

Random Forest Classifier:

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of over fitting to their training set. Random forest is a type of supervised machine learning algorithm based on ensemble learning. Ensemble learning is a type of learning where you join different types of algorithms or same algorithm multiple times to form a more powerful prediction model. The random forest algorithm combines multiple algorithm of the same type i.e. multiple decision *trees*, resulting in a *forest of trees*, hence the name "Random Forest". The random forest algorithm can be used for both regression and classification tasks.

The following are the basic steps involved in performing the random forest algorithm:

- Pick N random records from the dataset.
- Build a decision tree based on these N records.
- Choose the number of trees you want in your algorithm and repeat steps 1 and 2.

In case of a regression problem, for a new record, each tree in the forest predicts a value for Y (output). The final value can be calculated by taking the average of all the values predicted by all the trees in forest. Or, in case of a classification problem, each tree in the forest predicts the category to which the new record belongs. Finally, the new record is assigned to the category that wins the majority vote.

Random Forest Classifier

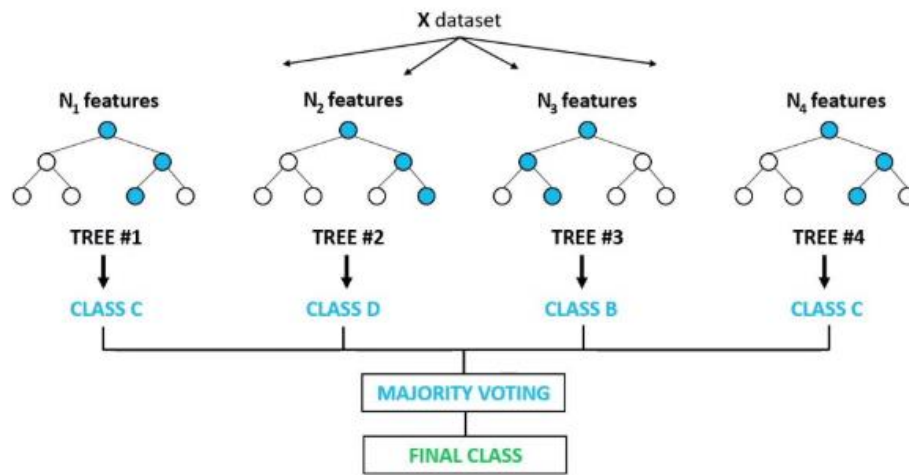


Fig:4.8: RANDOM FOREST CLASSIFIER

GIVEN INPUT EXPECT OUTPUT

input: data

output: getting accuracy

Decision Tree Classifier:

It is one of the most powerful and popular algorithm. Decision-tree algorithm falls under the category of supervised learning algorithms. It works for both continuous as well as categorical output variables. Assumptions of Decision tree:

- At the beginning, we consider the whole training set as the root.
- Attributes are assumed to be categorical for information gain, attributes are assumed to be continuous.
- On the basis of attribute values records are distributed recursively.
- We use statistical methods for ordering attributes as root or internal node.

Decision tree builds classification or regression models in the form of a tree structure. It breaks down a data set into smaller and smaller subsets while at the

same time an associated decision tree is incrementally developed. A decision node has two or more branches and a leaf node represents a classification or decision. The topmost decision node in a tree which corresponds to the best predictor called root node. Decision trees can handle both categorical and numerical data. Decision tree builds classification or regression models in the form of a tree structure. It utilizes an if-then rule set which is mutually exclusive and exhaustive for classification. The rules are learned sequentially using the training data one at a time. Each time a rule is learned, the tuples covered by the rules are removed. This process is continued on the training set until meeting a termination condition. It is constructed in a top-down recursive divide-and-conquer manner. All the attributes should be categorical. Otherwise, they should be discretized in advance. Attributes in the top of the tree have more impact towards in the classification and they are identified using the information gain concept. A decision tree can be easily over-fitted generating too many branches and may reflect anomalies due to noise or outliers.

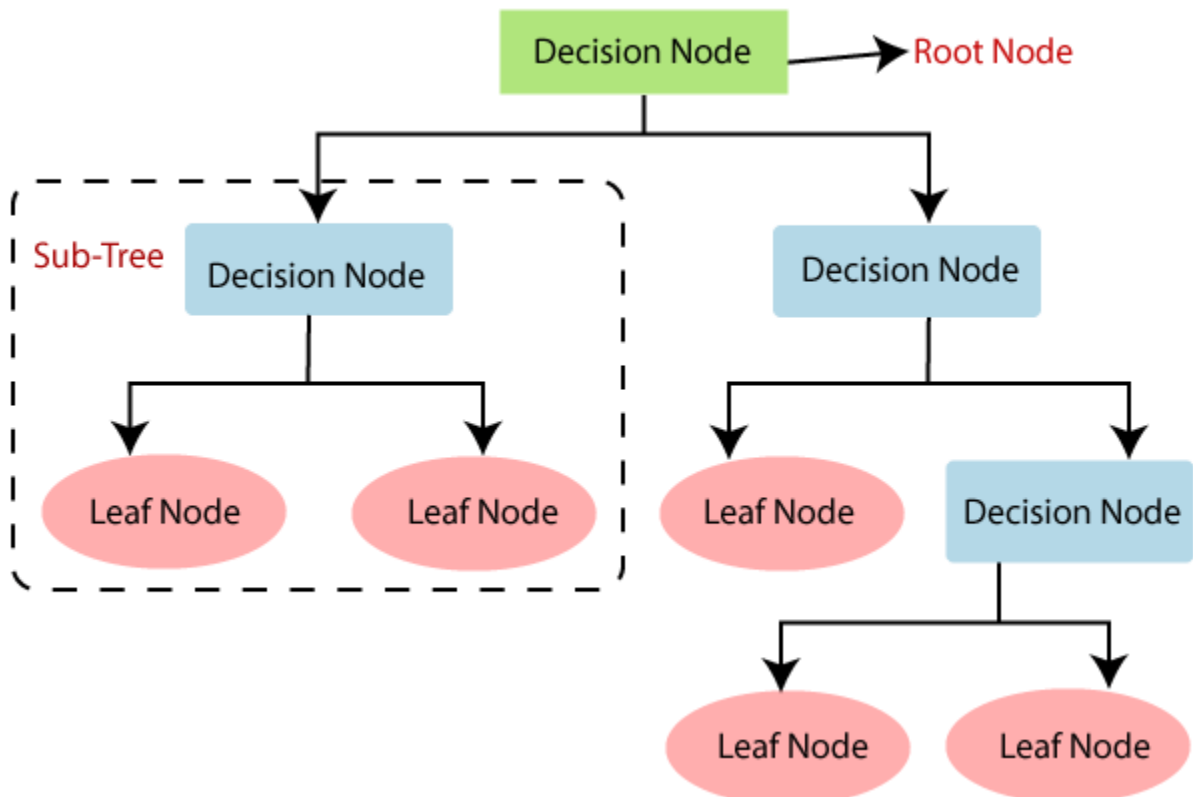


Fig:4.9: DECISION TREE CLASSIFIER

Naive Bayes algorithm:

The Naive Bayes algorithm is an intuitive method that uses the probabilities of each attribute belonging to each class to make a prediction. It is the supervised learning approach you would come up with if you wanted to model a predictive modeling problem probabilistically.

- Naive Bayes simplifies the calculation of probabilities by assuming that the probability of each attribute belonging to a given class value is independent of all other attributes. This is a strong assumption but results in a fast and effective method.
- The probability of a class value given a value of an attribute is called the conditional probability. By multiplying the conditional probabilities together for each attribute for a given class value, we have a probability of a data instance belonging to that class. To make a prediction we can calculate probabilities of the instance belonging to each class and select the class value with the highest probability.
- Naive Bayes is a statistical classification technique based on Bayes Theorem. It is one of the simplest supervised learning algorithms. Naive Bayes classifier is the fast, accurate and reliable algorithm. Naive Bayes classifiers have high accuracy and speed on large datasets.
- Naive Bayes classifier assumes that the effect of a particular feature in a class is independent of other features. For example, a loan applicant is desirable or not depending on his/her income, previous loan and transaction history, age, and location.
- Even if these features are interdependent, these features are still considered independently. This assumption simplifies computation, and that's why it is considered as naive. This assumption is called class conditional independence.

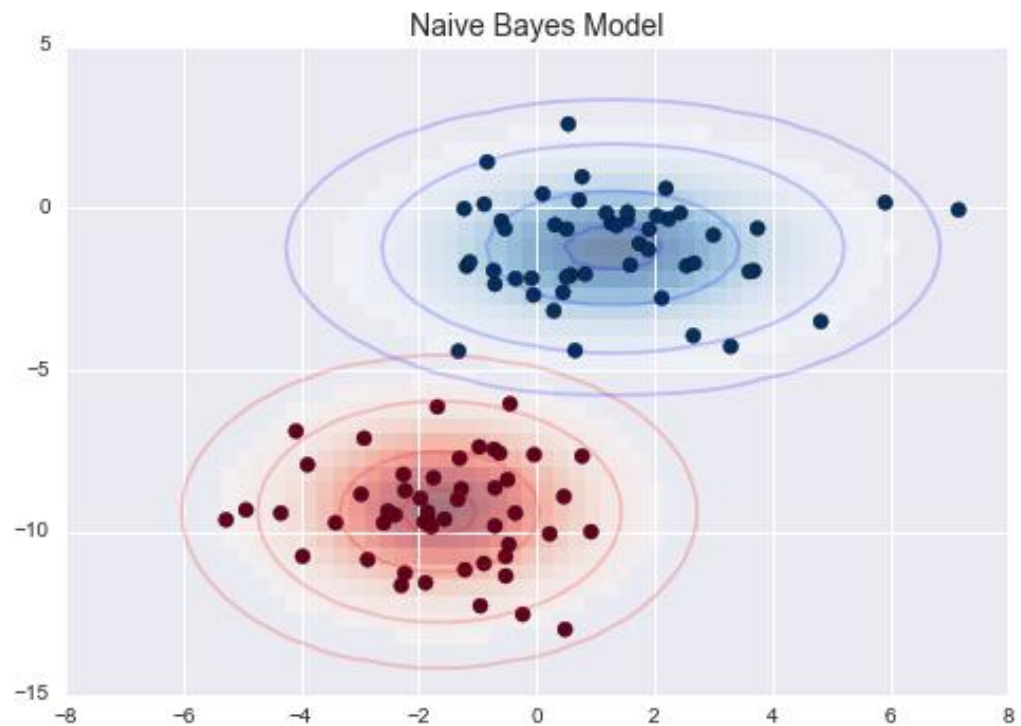


Fig:4.10: NAÏVE BAYES CLASSIFIER

K-Nearest Neighbor

K-Nearest Neighbor is one of the simplest Machine Learning algorithms based on Supervised Learning technique. It 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. It 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. K- NN algorithm can be used for Regression as well as for Classification but mostly it is used for the Classification problems.

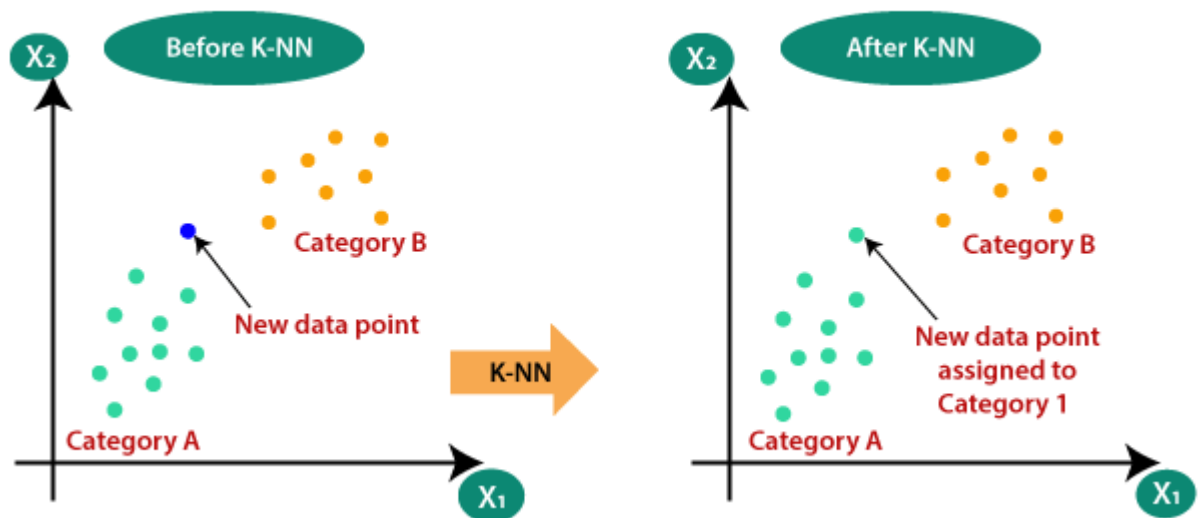


Fig:4.11: K-NEAREST NEIGHBOR

Support Vector Classifier

Support Vector Classifier or SVC is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. Primarily, it is used for Classification problems in Machine Learning. The goal of the SVC algorithm 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. This best decision boundary is called a hyperplane. SVC chooses the extreme points/vectors that help in creating the hyperplane. These extreme cases are called as support vectors, and hence algorithm is termed as Support Vector Machine.

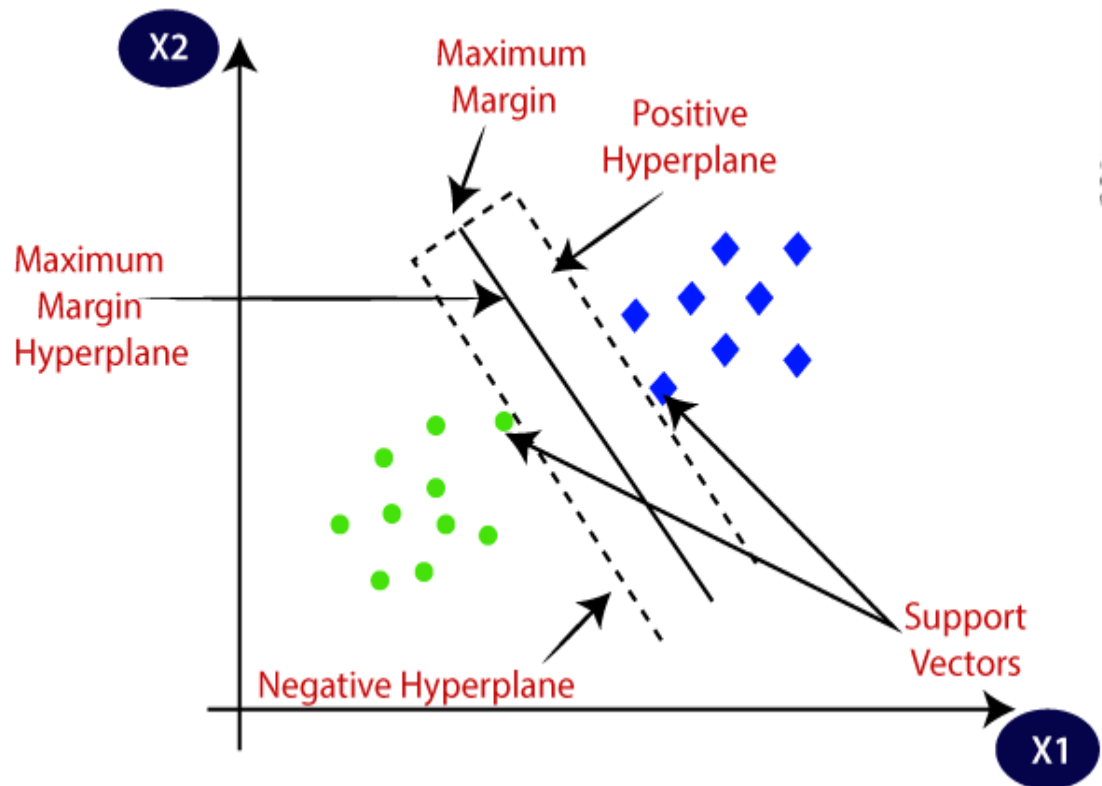


Fig:4.12: SUPPORT VECTOR CLASSIFIER

Gradient Boost

Gradient boosting algorithm is one of the most powerful algorithms in the field of machine learning. As we know that the errors in machine learning algorithms are broadly classified into two categories i.e., Bias Error and Variance Error. As gradient boosting is one of the boosting algorithms it is used to minimize bias error of the model.

Gradient boosting algorithm can be used for predicting not only continuous target variable (as a Regressor) but also categorical target variable (as a Classifier).

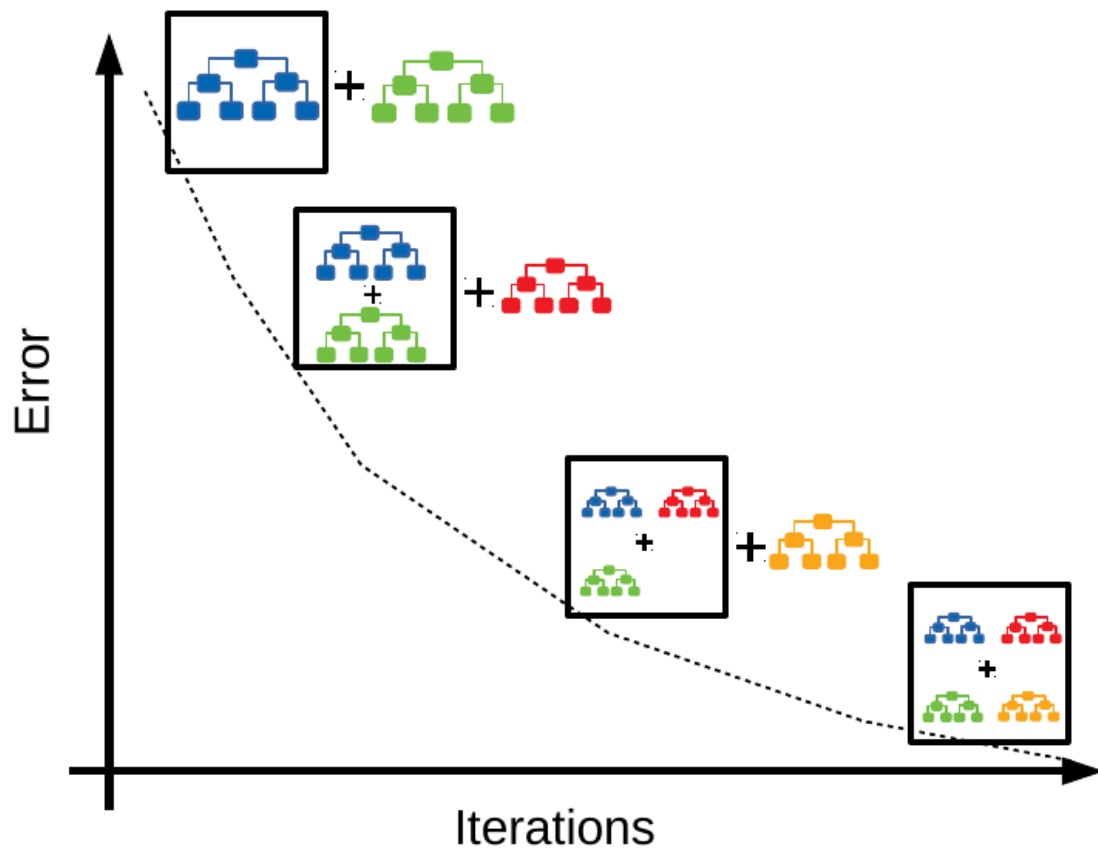


Fig:4.13: GRADIENT BOOST

4.7.6 Deployment Using Flask (Web Framework):

Flask is a micro web framework written in Python. It is classified as a micro-framework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself.

Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

Flask was created by Armin Ronacher of Poccoo, an international group of Python

enthusiasts formed in 2004. According to Ronacher, the idea was originally an April Fool's joke that was popular enough to make into a serious application. The name is a play on the earlier Bottle framework.

When Ronacher and Georg Brand created a bulletin board system written in Python, the Pocoo projects Werkzeug and Jinja were developed.

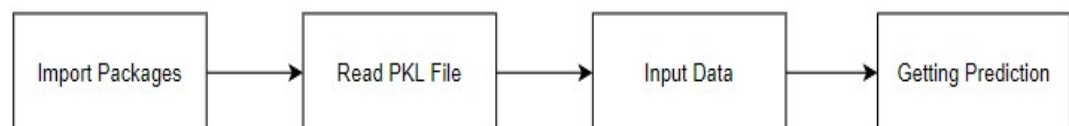
In April 2016, the Pocoo team was disbanded and development of Flask and related libraries passed to the newly formed Pallets project.

Flask has become popular among Python enthusiasts. As of October 2020, it has second most stars on GitHub among Python web-development frameworks, only slightly behind Django, and was voted the most popular web framework in the Python Developers Survey 2018.

The micro-framework Flask is part of the Pallets Projects, and based on several others of them.

Flask is based on Werkzeug, Jinja2 and inspired by Sinatra Ruby framework, available under BSD licence. It was developed at pocoo by Armin Ronacher. Although Flask is rather young compared to most Python frameworks, it holds a great promise and has already gained popularity among Python web developers. Let's take a closer look into Flask, so-called "micro" framework for Python.

MODULE DIAGRAM



GIVEN INPUT EXPECTED OUTPUT

input : data values

output : predicting output

CHAPTER-5

RESULTS AND DISCUSSION, PERFORMANCE ANALYSIS

5.1 PERFORMANCE ANALYSIS:

Website performance optimization, the focal point of technologically superior website designs is the primary factor dictating success for Loan approval process. After all, unimpressive website performance kills admission process when the torture of waiting for slow Web pages to load frustrates visitors into seeking alternatives – impatience is a digital virtue! And also the ml algorithms used in our project will give the best accurate result to the user for Loan approval.

We created the following six chapter in-depth speed optimization guide to show you how important it is to have a fast loading, snappy website! Countless research papers and benchmarks prove that optimizing your sites' speed is one of the most affordable and highest ROI providing investments!

Lightning-fast page load speed amplifies visitor engagement, retention, and boosts sales. Instantaneous website response leads to higher conversion rates, and every 1 second delay in page load decreases customer satisfaction by 16 percent, page views by 11 percent and conversion rates by 7 percent according to a recent Aberdeen Group research.

5.2 DISCUSSION:

While discussions provide avenues for exploration and discovery, leading a discussion can be anxiety-producing: discussions are, by their nature, unpredictable, and require us as instructors to surrender a certain degree of control over the flow of information. Fortunately, careful planning can help us ensure that discussions are lively without being chaotic and exploratory without losing focus. When planning a discussion, it is helpful to consider not only cognitive, but also social/emotional, and physical factors that can either foster or inhibit the productive exchange of ideas.

CHAPTER-6

SUMMARY AND CONCLUSION

6.1 SUMMARY:

This project objective is to predict the Loan Approval of the user. So this online banking loan approval system will reduce the paper work and reduce the wastage of bank asserts and efforts and also saves the valuable time of the customer.

6.2 CONCLUSION:

The analytical process started from data cleaning and processing, missing value, exploratory analysis and finally model building and evaluation. The best accuracy on public test set is higher accuracy score will be find out. This application can help to find the Prediction of Bank Loan Approval.

6.3 FUTURE WORK:

- Bank Loan Approval prediction to connect with cloud.
- To optimize the work to implement in Artificial Intelligence environment.

REFERENCES:

- [1] Amruta S. Aphale , Dr. Sandeep R. Shinde, 2020, Predict Loan Approval in Banking System Machine Learning Approach for Cooperative Banks Loan Approval, International Journal Of Engineering Research & Technology (IJERT) Volume 09, Issue 08 (August 2020)
- [2] Ashwini S. Kadam, Shraddha R. Nikam, Ankita A. Aher, Gayatri V. Shelke, Amar S.Chandgude, "Prediction for Loan Approval using Machine Learning Algorithm" (IRJET) Volume: 08 Issue: 04 | Apr 2021.
- [3] M. A. Sheikh, A. K. Goel and T. Kumar, "An Approach for Prediction of Loan Approval using Machine Learning Algorithm," 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), 2020, pp. 490-494, doi: 10.1109/ICESC48915.2020.9155614.
- [4] Rath, Golak & Das, Debasish & Acharya, Biswaranjan. (2021). Modern Approach for Loan Sanctioning in Banks Using Machine Learning. Pages={179-188} 10.1007/978-981-15-5243-4_15.
- [5] Vincenzo Moscato, Antonio Picariello, Giancarlo Sperlí, A benchmark of machine learning approaches for credit score prediction, Expert Systems with Applications, Volume 165, 2021, 113986, ISSN 0957-4174.
- [6] Yash Divate, Prashant Rana, Pratik Chavan, "Loan Approval Prediction Using Machine Learning" International Research Journal of Engineering and Technology (IRJET) Volume: 08 Issue: 05 | May 2021.

APPENDIX:

A. SOURCE CODE:

Jupyter Notebook

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import missingno as mso
import seaborn as sns

from sklearn.model_selection import train_test_split
from imblearn.over_sampling import SMOTE

from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.naive_bayes import GaussianNB
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import GradientBoostingClassifier
```

```
df=pd.read_csv("Loan_Train.csv")
df.head()
```

```
countMale = len(df[df.Gender == 'Male'])
countFemale = len(df[df.Gender == 'Female'])
countNull = len(df[df.Gender.isnull()])

print("Percentage of Male applicant: {:.2f}%".format((countMale /
(len(df.Gender))*100))
print("Percentage of Female applicant: {:.2f}%".format((countFemale /
(len(df.Gender))*100))
print("Missing values percentage: {:.2f}%".format((countNull /
(len(df.Gender))*100))
```

```
df.Married.value_counts(dropna=False)
sns.countplot(x="Married", data=df, palette="Paired")
plt.show()
```

```
df = pd.get_dummies(df)
df = df.drop(['Gender_Female', 'Married_No', 'Education_Not Graduate',
            'Self_Employed_No', 'Loan_Status_N'], axis = 1)
```

```
Q1 = df.quantile(0.25)
Q3 = df.quantile(0.75)
IQR = Q3 - Q1
```

```
df = df[~((df < (Q1 - 1.5 * IQR)) |(df > (Q3 + 1.5 * IQR))).any(axis=1)]
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2,
random_state = 0)
```

```
LRclassifier = LogisticRegression(solver='saga', max_iter=500, random_state=1)
LRclassifier.fit(X_train, y_train)
y_pred = LRclassifier.predict(X_test)
```

```
scoreListknn = []
for i in range(1,21):
    KNclassifier = KNeighborsClassifier(n_neighbors = i)
    KNclassifier.fit(X_train, y_train)
    scoreListknn.append(KNclassifier.score(X_test, y_test))
```

```
SVCclassifier = SVC(kernel='rbf', max_iter=500)
SVCclassifier.fit(X_train, y_train)
y_pred = SVCclassifier.predict(X_test)
```

```
NBclassifier2 = GaussianNB()
NBclassifier2.fit(X_train, y_train)
y_pred = NBclassifier2.predict(X_test)
```

```
scoreListDT = []
for i in range(2,21):
    DTclassifier = DecisionTreeClassifier(max_leaf_nodes=i)
    DTclassifier.fit(X_train, y_train)
    scoreListDT.append(DTclassifier.score(X_test, y_test))
```

```
scoreListRF = []
for i in range(2,25):
    RFclassifier = RandomForestClassifier(n_estimators = 1000, random_state =
1, max_leaf_nodes=i)
    RFclassifier.fit(X_train, y_train)
    scoreListRF.append(RFclassifier.score(X_test, y_test))
```

```
GBclassifier = GradientBoostingClassifier(subsample=1, n_estimators=200,
max_depth=5, max_leaf_nodes=40)
GBclassifier.fit(X_train, y_train)
y_pred = GBclassifier.predict(X_test)
```

```
compare = pd.DataFrame({'Model': ['Logistic Regression', 'Random Forest', 'K
Neighbors', 'Decision Tree', 'SVM', 'Gaussian NB', 'Gradient Boost'], 'Accuracy':
[LRAcc*100, RFAcc*100, KNAcc*100, DTAcc*100, SVCAcc*100,
NBAcc2*100, GBAcc*100]})
compare.sort_values(by='Accuracy', ascending=False)
```

```
import pickle
file=open("RFclassifier.pkl", 'wb')
pickle.dump(RFclassifier, file)
```

Html pages source code: Index.html

```
<!doctype html>
<html lang="en">
  <head><title>loan Prediction</title></head><body>
    <span class="block xl:inline">Loan Prediction</span>
    <span class="block text-indigo-600 xl:inline">Machine Learning </span>
    </h1><p class="mt-3 text-base text-gray-500 sm:mt-5 sm:text-lg sm:max-w-xl sm:mx-auto md:mt-5 md:text-xl lg:mx-0">Stay Ahead of time and find out now with ML powered predictions and it helps in knowing that if you are Eligible or Not for taking loan.</p><div class="mt-5 sm:mt-8 sm:flex sm:justify-center lg:justify-start"><div class="rounded-md shadow"><a href="/predict" class="w-full flex items-center justify-center px-8 py-3 border border-transparent text-base font-medium rounded-md text-white bg-indigo-600 hover:bg-indigo-700 md:py-4 md:text-lg md:px-10">Prediction</a></div></div></div>
</body></html>
```

Predict.html

```
<!doctype html>
<html lang="en">
  <head><title>prediction</title></head><body>
    <section class="text-gray-600 body-font"><div class="container px-5 py-24 mx-auto"><div class="flex flex-col text-center w-full mb-20">
<h1 class="sm:text-3xl text-2xl font-medium title-font mb-4 text-gray-900">Loan prediction</h1><p class="lg:w-2/3 mx-auto leading-relaxed text-base">fill the form for prediction</p></div><div>
    <a class="btn btn-primary" href="/" role="button">Back</a>
  </div></div><div class="mb-3">
    <label for="exampleFormControlInput1" class="form-label"> Gender</label>
    <select class="form-select" id="gender" name="gender" aria-label="Default select example"><option selected>-- select gender --</option>
    <option value="Male">Male</option><option value="Female">Female</option>
  </select></div><div class="mb-3">
    <label for="exampleFormControlInput1" class="form-label">Enter ApplicantIncome</label><input type="text" class="form-control" id="ApplicantIncome" name="ApplicantIncome" placeholder="ApplicantIncome">
  </div></div><button type="submit" class="btn btn-primary">Predict</button>
</div></div></section>
</body></html>
```

LA. Html (Loan Approval):

```
<!DOCTYPE html>
<html lang="en">
<head>
</style><title>Loan Approval</title>
</head>
<body>
  <div class="bg-image"></div>
  <div class="bg-text">
    <div class="col-md-6 my-2 d-flex align-items-end justify-content-around">
      <a href="/predict">
        <button type="submit" class="btn btn-danger button" style="margin-right: 100%;">Back</button></a>
      </div>
      <h1>LOAN STATUS</h1>
      <p>You Will Get the Approval from the Bank </p>
    </div>
  </body>
</html>
```

LR. Html (Loan Reject):

```
<!DOCTYPE html>
<html lang="en">
<head>
  -webkit-filter: blur(3px);height: 100%;
  background-position: center;
  background-repeat: no-repeat;
  background-size: cover;
</style>
  <title>Loan Reject</title>
</head>
<body>
  <div class="bg-image"></div>

  <div class="bg-text">
    <div class="col-md-6 my-2 d-flex align-items-end justify-content-around">
      <a href="/predict"> <button type="submit" class="btn btn-danger button" style="margin-right: 100%;">Back</button></a>
    </div>
    <h1>LOAN STATUS</h1>
    <p>Your Details are not satisfied For Loan Approval</p>
  </div></body>
</html>
```

App.py:

```
from flask import Flask, escape, request, render_template
import pickle
import numpy as np
app = Flask(__name__)
model = pickle.load(open('RFclassifier.pkl', 'rb'))
@app.route('/')
def home():
    return render_template("index.html")
@app.route('/predict', methods=['GET', 'POST'])
def predict():
    if request.method == 'POST':
        gender = request.form['gender'] married = request.form['married']
        dependents = request.form['dependents']
        education = request.form['education']
        self_employed = request.form['employed']
        credit = float(request.form['credit']) area = request.form['area']
        ApplicantIncome = float(request.form['ApplicantIncome'])
        CoapplicantIncome = float(request.form['CoapplicantIncome'])
        LoanAmount = float(request.form['LoanAmount'])
        Loan_Amount_Term = float(request.form['Loan_Amount_Term'])
        if (gender == "Male"):
            Gender_Male=1
        else:
            Gender_Male=0
        if(married=="Yes"):
            Married_Yes = 1
        else:
            Married_Yes=0
        if(dependents=='1'):
            dependents_0 = 0
            dependents_1 = 1
        arr=np.array([[credit, ApplicantIncome, CoapplicantIncome, LoanAmount,
Loan_Amount_Term, Gender_Male, Married_Yes, dependents_0, dependents_1,
dependents_2, dependents_3, education_graduate, self_employed_yes,semiurban,
urban, rural ]])
        prediction = model.predict(arr)
        if(prediction==0):
            return render_template("LR.html")
        elif(prediction!=0):
            return render_template("LA.html")
    else:
        return render_template("prediction.html")
if __name__ == "__main__":
    app.run(debug=True)
```

B. SCREENSHOTS

	Model	Accuracy
2	K Neighbors	91.111111
1	Random Forest	88.888889
4	SVM	86.666667
3	Decision Tree	84.444444
0	Logistic Regression	82.222222
6	Gradient Boost	75.555556
5	Gaussian NB	71.111111

Fig:6.1: MACHINE LEARNING ALGORITHMS ACCURACY

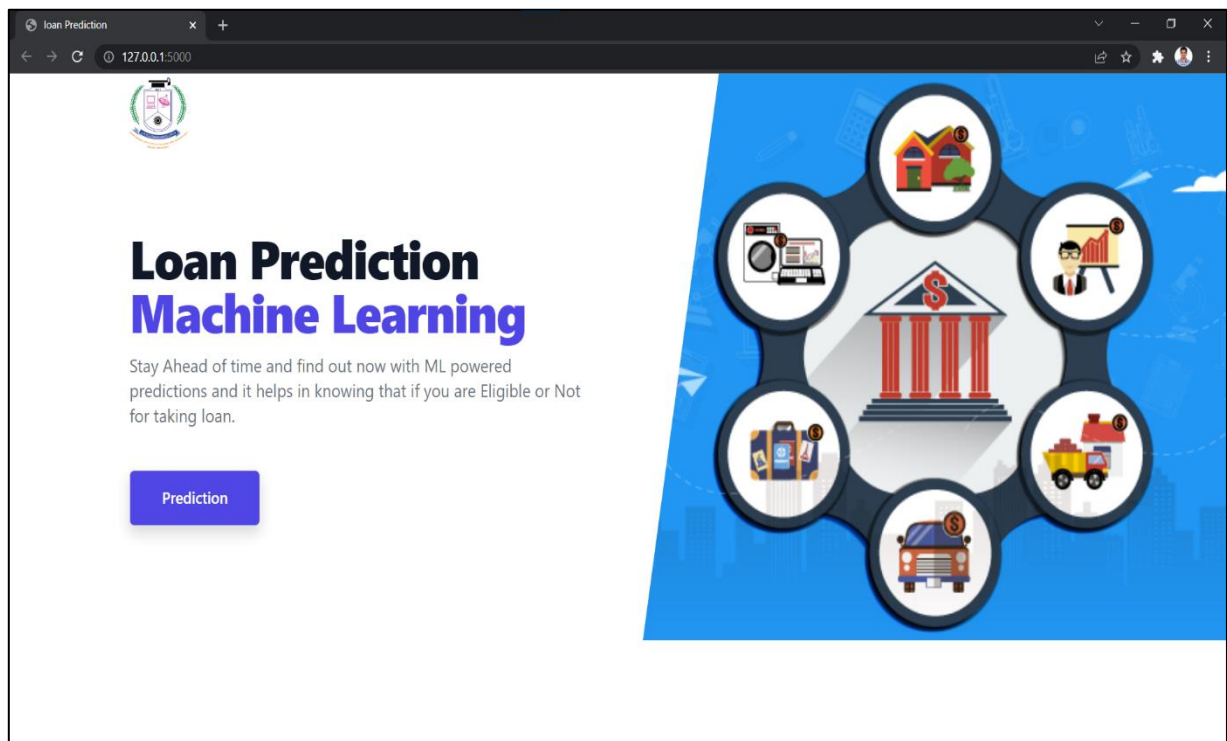


Fig:6.2: HOME PAGE

prediction x +
← → ↻ 127.0.0.1:5000/predict

Loan prediction

fill the form for prediction

[Back](#)

Gender
-- select gender --

Married status
-- select married status --

Dependents
-- select dependents --

Education
-- select education --

Self_Employed
-- select Self_Employed --

Fig:6.3: INPUTS PAGE

prediction x +
← → ↻ 127.0.0.1:5000/predict

Loan prediction

fill the form for prediction

[Back](#)

Gender
Male

Married status
Yes

Dependents
2

Education
Graduate

Self_Employed
Yes

Fig:6.4: INPUTS GIVEN BY THE USER

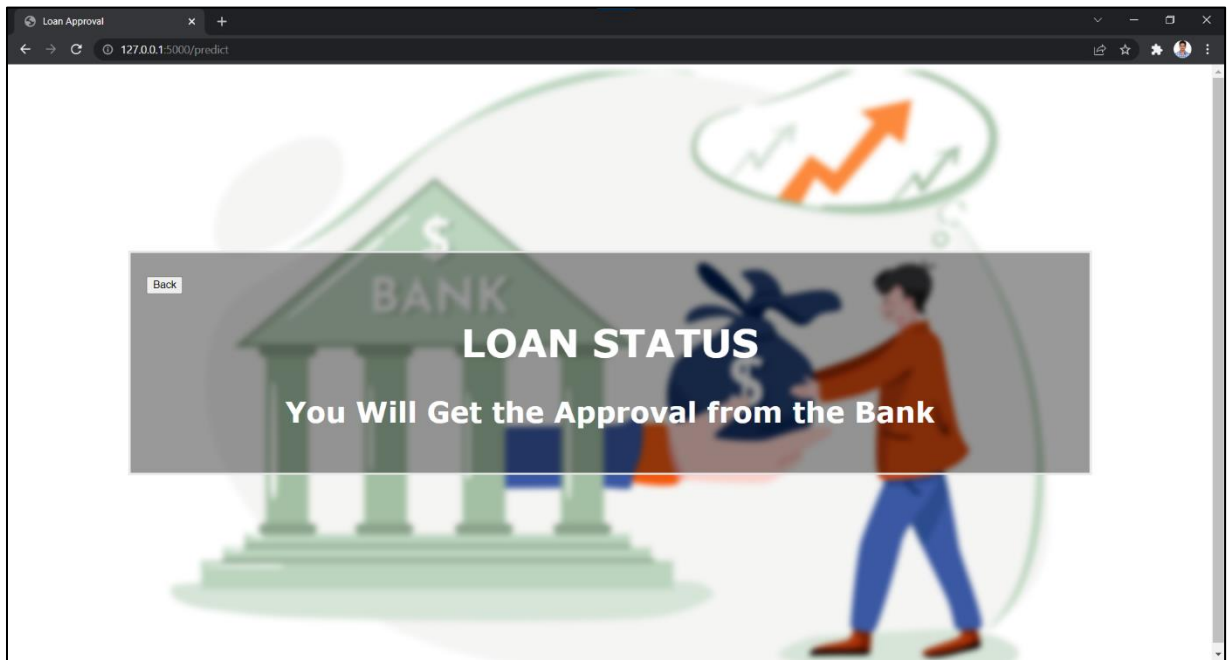


Fig:6.5: LOAN APPROVED

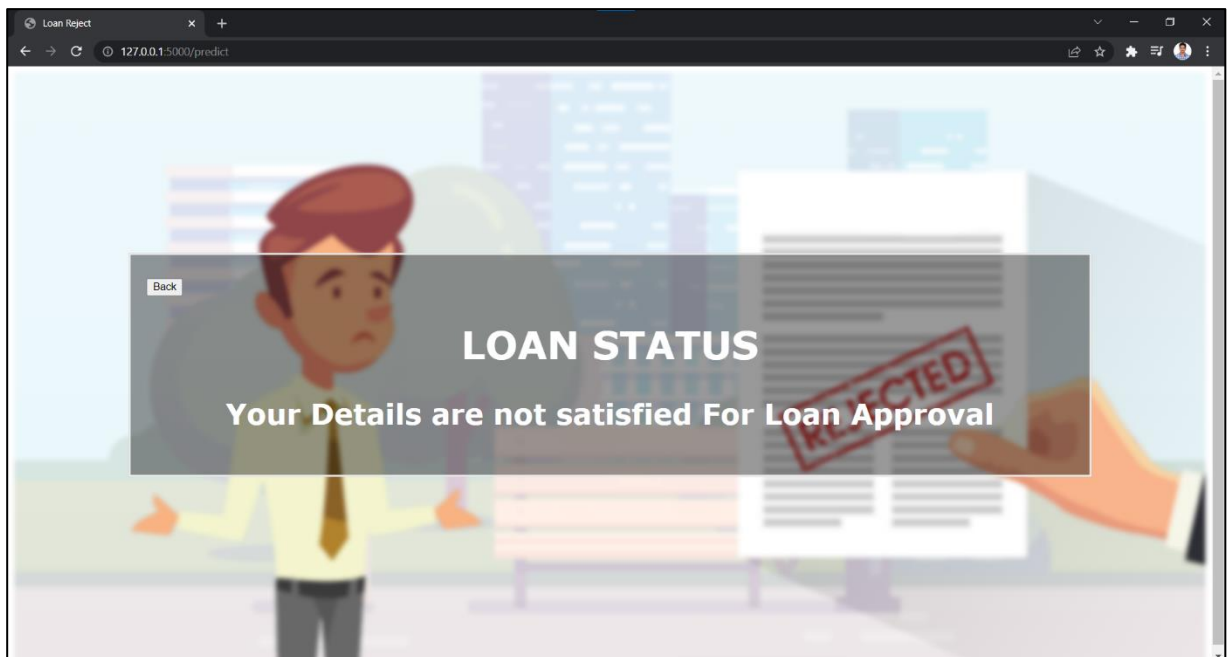


Fig:6.6: LOAN REJECTED

Bank loan approval research paper (1).docx

by Bank Loan Approval Research Paper (1).docx Bank Loan Approval
Research Paper (1).docx

Submission date: 28-Jan-2022 04:44PM (UTC+1000)

Submission ID: 1749835604

File name: Bank_loan_approval_research_paper_1.docx (474.98K)

Word count: 2360

Character count: 13597

BANK LOAN APPROVAL DATA ANALYZE AND PREDICTION USING DATA SCIENCE TECHNIQUE (ML)

Gunnam Srinivasu¹, Allam Shankar Pavan Kalyan², M.Maheswari³, M.Selvi⁴

^{1,2} Student, ^{3,4} Associate professor, CSE, Sathyabama Institute of Science and Technology, Chennai, India

ABSTRACT: Loans are not generally viewed if all else fails to purchase a sought-after cell phone or a fantasy house. Throughout the last ten years or somewhere in the vicinity, individuals have become less reluctant in applying for an Loan, regardless of whether it's private, vehicle, instruction, business, or home - particularly when they don't have a singular amount available to them. Also, Home and Education Loans give charge benefits that diminish charge responsibility and increment the money close by from compensation income. to get credits with negligible administrative work, fast qualification checks, and cutthroat financing costs. This paper is separated into four segments. Proposed framework is to carry out and research what different managed paired arrangement strategies mean for default forecast. The model assessment strategies utilized in this undertaking are restricted to accuracy, awareness, F1-score.

Keywords: Loan, compensation income, financing costs, credits.

1. INTRODUCTION

Loan prediction is mostly useful for bank workers for approving the application. This simple method gives fast and accurate results in approving the customer application. So the user can easily check into this loan system prediction whether their loan is approved or rejected. This online banking loan approval system will reduce the paper work and reduce wastage of bank asserts and efforts. This application can also be changed depends on the regulation and conditions of the banks. The entire course expectation has completed secretly and no other have the option to change the handling.

2. RELATED WORK

Sharmili et al [1] introduced an Loan forecast model using Support vector machines and Naïve Bayesian techniques. Guileless Bayes, a free hypothesis proceeds, envelops likelihood hypothesis with respect to the information order. Then again, Support Vector Machine works with this statistical learning model for grouping of forecasts. Dataset from UCI archive with 21 ascribes was taken on to assess the proposed technique.

Vikas et al [2] introduced a machine learning built credit forecast model. The libraries in the current methodology were information assortment and pre-handling, implementing the machine learning algorithms, prepares after testing the information. In the pre-handling level, identification, expulsion

of anomalies, attribution evacuation handling was completed. Here KNN, DT, SVM also slope helping models were utilized to anticipate the potential outcomes of current status in regards to the Loan endorsement process.

Shiva Sri et al [3] proposed an Exploratory DataAnalysis with respect to the credit expectation method in light of the customer's tendency and their necessities. The central point concentrated during the information investigation were yearly pay versus Loan intention, client's trust, Loan residency vs delinquent months, Loan residency vs credit class, Loan residency vs no. of years in the present place of employment, and opportunities aimed at house proprietorship and Loan reimbursement.

Kaur et al [4] introduced a credit forecast model utilizing a few Machine Learning (ML) calculations. The dataset with highlights, specifically, orientation, conjugal status, training, number of wards, business status, pay, coapplicant's pay, Loan sum, Loan residency, record as a consumer are utilized for deciding a credit qualification in regards to the credit authorizing process.

Mohammed Ahmed et al [5] introduced a credit hazard expectation model in used with NB, J48 also BN approaches. This approach was carried out utilizing ascribes, to be specific, orientation, work, age, credit sum, record as a consumer, reason, lodging, and class

Goyal et al [6] suggested a group method based Loan forecast strategy for the clients. The sub cycles in the current technique incorporates, information assortment, sifting the information, highlight extraction, applying the model, lastly examination the outcomes. The different credit forecast procedures implemented with SVM,RF,TGA,ROC.

Sudhamathy et al [7] recommended a danger examination technique in endorsing an Loan for the clients utilizing R bundle. To calibrate the forecast perfection, the pre-handling activity contains the below these methods: discovery, positioning, expulsion of anomalies, evacuation of ascription, adjusting of data by relative division with respect to validating, preparing process. After that it includes the determination method for further develops the forecast precision. When assessed, the DT model brought about 94.3% expectation exactness. The method involved with examining information according to alternate points of view and removing helpful information from it.

3. EXISTING SYSTEM

Irregularity recognition depends on people's conduct profiling and works by recognizing any deviation from the standard. When utilized for web based financial misrepresentation recognition, in any case, it predominantly experiences three hindrances. To begin with, for an individual, the chronicled conduct information are regularly too restricted to even consider profiling his/her standard of conduct. Second, because of the heterogeneous idea of exchange information, there comes up short on uniform treatment of various types of trait esteems, which turns into a possible boundary for model turn of events and further use. Third, the exchange information are profoundly slanted, and it turns into a test to use the mark data viably. Irregularity identification frequently experiences helpless speculation capacity and a high bogus caution rate. We contend that people's restricted recorded information for conduct profiling and the profoundly slanted nature of misrepresentation information could represent this deformity. Since it is direct to utilize data from other comparable people, estimating similitude itself turns into an extraordinary test because of heterogeneous quality qualities. We propose to change the oddity discovery issue into a pseudorecommender framework issue and tackle it with an installing based technique. Thusly, the

possibility of cooperative sifting is verifiably used to use data from comparable clients, and the learned inclination lattices and ascribe installing give a compact method for promoting utilization.

DISADVANTAGES OF THE EXISTING SYSTEM

They had proposed a numerical model and AI calculations isn't used. Class Imbalance issue was not tended to and the legitimate measure were not taken.

4. PROPOSED SYSTEM

Bank loan is important for more business and as well as for housing and education purpose. The current system is good it also have some disadvantages like time taking and more paper work. So we introduce this new system like online process for bank loan approval. Therefore, This modules aims helps in predicting the bank loan status by using some attributes like Credit History, Loan Term, Self Employed, Applicant Income.

The model uses high-level architecture where it is trained and fit using the Machine Learning algorithm and then the prediction is used to assess the execution of the algorithm. This algorithm type helps in predicting the loan approval and helps in reduces the loss resulting due to them.

We are using different ML Algorithms for obtaining the best from that to use. We are using a dataset from Kaggle. the dataset taken by Kaggle is thoroughly Pre-processing and cleaning has been done. After this data visualization is done to the dataset by that dataset we split the dataset into test data set and we trained that test data set with train data set and finally, we take the best accuracy Algorithm by comparing different Algorithms. After that, we implemented the model by that best model.

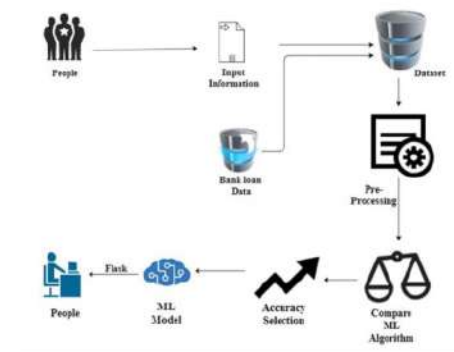


Fig: 1.1 OVERVIEW OF THE SYSTEM ARCHITECTURE

ADVANTAGES OF THE PROPOSED SYSTEM

- Execution and exactness of the calculations can be determined and analyzed.
- Class irregularity can be managed AI draws near.

5. IMPLEMENTATION

This model helps in predicting the loan approval and We develop this application in the following way at first we build this module by collecting all the information from the previous year data how bank loan is approved and this data is collected from external resources from Kaggle. After Collecting the Dataset we have some work for developing the algorithms by using the ML methods. Firstly, we have to do Data-Preprocessing. In this data preprocessing the following are done. Data cleaning which helps in removing the unwanted data like missing values, Noisy data and removing outliers.

Data integration helps in storing the large data into single unit. Data Transformation helps in formatting the data or structured the data. Data reduction helps in reducing the volume of the data which helps us in analysis the data easier and after this our dataset is prepared. For this dataset we apply data visualization technique which helps in showing the data in the format like static, animated, and interactive visualization. Here the data representation shown in graphical way. Which helps in analysis the data quiet easy. After this we train the data with the given input test dataset. After this we apply different machine learning model to get their accuracy. From this accuracy we take the best and apply to our train and test dataset.

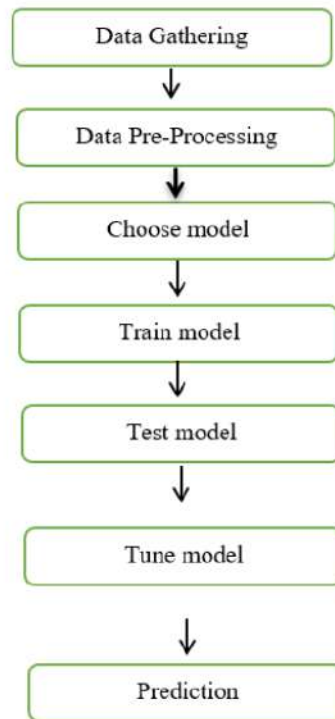


Fig: 1.2 PROJECT WORKFLOW

For more info related to the project are followed the steps involved in doing the project is Data preprocessing, Data Analysis of visualization, compare the algorithms and get valuable accuracy score for better result.

6. MODULES DESCRIPTION

6.1 DATASET

The dataset is collected from kaggle and it most popular website for DataScience and is Mostly helpful for Students who are working on DataScience environment. My dataset name is bank dataset(Fig: 1.2) and it consist of following attributes like Credit_History, Loan_ID, Self_Employed, ApplicantIncome, etc.,

Loan_ID	Gender	Married	Dependent	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property_Area	Loan_Status
1	Male	No	0	Graduate	No	5849	0	960	360	1	Urban	Y
2	Male	Yes	1	Graduate	No	4583	1526	128	360	1	Rural	N
3	Male	Yes	0	Graduate	Yes	3000	0	66	360	1	Urban	Y
4	Male	Yes	0	Not Graduate	No	2047	2345	125	360	1	Urban	Y
5	Male	Yes	0	Graduate	No	6000	0	141	360	1	Urban	Y
6	Male	No	0	Graduate	No	4000	0	0	360	1	Urban	Y
7	Male	Yes	2	Graduate	Yes	3417	4396	267	360	1	Urban	Y
8	Male	Yes	0	Not Graduate	No	2333	1254	95	360	1	Urban	Y
9	Male	Yes	0	Graduate	No	3036	2254	128	360	1	Semiurban	N
10	Male	Yes	2	Graduate	No	4006	1225	168	360	1	Urban	Y
11	Male	Yes	1	Graduate	No	12941	10966	349	360	1	Semiurban	N
12	Male	Yes	2	Graduate	No	1200	791	70	360	1	Urban	Y
13	Male	Yes	2	Graduate	No	2200	2840	109	360	1	Urban	Y
14	Male	Yes	2	Graduate	No	3075	4036	200	360	1	Urban	Y
15	Male	No	0	Graduate	No	1853	2840	114	360	1	Rural	N
16	Male	Yes	2	Graduate	No	1279	3896	17	120	1	Urban	Y
17	Male	No	0	Graduate	No	4950	0	125	360	1	Urban	Y
18	Male	No	1	Not Graduate	No	2196	0	109	360	1	Urban	Y
19	Female	No	0	Graduate	No	3510	0	76	360	1	Urban	N
20	Male	Yes	0	Not Graduate	No	4887	0	133	360	1	Rural	N

Fig: 1.3 DataSet

6.2 DATA VALIDATION/ CLEANING/PREPARING PROCESS

After taking the dataset, the first and most important step is data cleaning or validation. In this step by using some pandas libraries like shape, isnull, describe, info etc., for analysing and cleaning the dataset. Here we remove the unwanted data like missing values, noisy data and null values removing outliers from the dataset. For removing missing values here, we followed by removing or ignoring the unwanted tuples and also by filling the missing values in the dataset. After this we check for the noisy data if there is any noisy data identified in the dataset after checking the missing values. We have used some methods like binning, regression and clustering for removing the noisy data in the dataset. After removing missing values and noisy data from the dataset if there any outliers are present we remove them by using clustering technique.

6.3 SUMMARIZING THE DATA VISUALISATION

The dataset contains various attributes like Applicantincome, credithistory, selfemployed etc., mainly loan approval prediction depends on this.

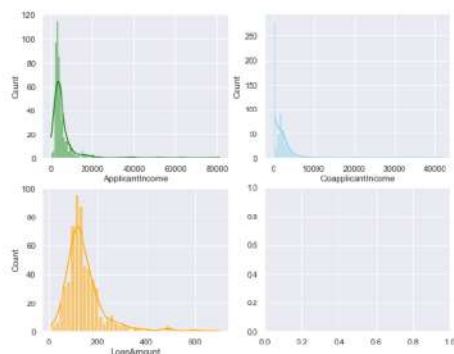


Fig: 1.4 Relation count between some attributes like Applicantincome, Coapplicantincome, Loanamount

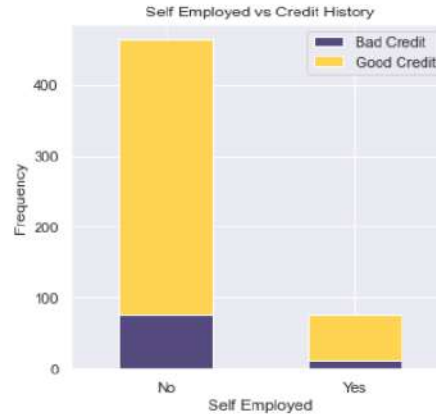


Fig: 1.5 Most not self-employed applications has more credits compared to self-employed applicants

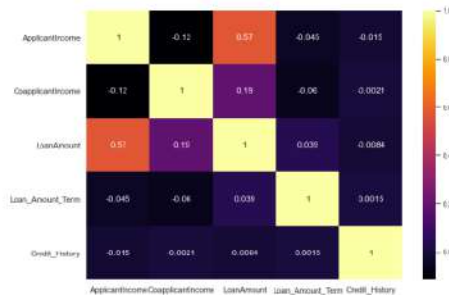


Fig: 1.6 HEATMAP

It tells the Correlation between all the features

6.4 COMPARING ALGORITHM WITH EXPECTATION AS BEST PRECISION RESULT

Because more than one class may be given to a single instance, multi-label classification (MLC) is the best and optimum option. The MLC is embedded into the algorithms used for categorization and model building, which are as follows:

2

1. Random Forest

2. Decision tree

3. Logistic regression

4. Naïve Bayes

Algorithm	Accuracy
Random Forest	86.66
Decision Tree	80.00
Logistic Regression	75.55
Naïve Bayes	64.44
KNN	84.44
Support Vector Machine	77.77
Gradient Boost	80.00

6.5 DEPLOYMENT USING FLASK

The project is deployed using flask for making an interface. The interface contains inputs like loan id, gender, marital status, self-employed, applicant income etc., By giving this inputs we get the loan status as output.

7. RESULT

With the created Random Forest model we acquired the best accuracy with a voting classifier where we created an array called Random forest. which consists of our models and we made the voting parameter set to hard, which tells that our model to make the predictions by the highest votes now, we can implement Random Forest model and we predicted the accuracy score with our test data.

8. CONCLUSION

This record tells the best way to concentrate on loan approval or rejection in utilizing a machine preparing machine to decrease paper work. We utilized different calculations to decide the whether the loan approved or not.

At present, this online bank loan approval system helps people to save their time and work by predicting the loan status in online.

REFERENCES:

- [1] S. Vimala, K.C. Shamili, —Prediction of Loan Risk using NB and Support Vector Machinel, International Conference on Loanments in Computing Technologies (ICACT 2018), vol. 4, no. 2, pp. 110-113, 2018.
- [2] Pidikiti Supriya, Myneedi Pavani, Nagarapu Saisushma, Namburi Vimala Kumari, K. Vikas, —Loan Prediction by using Machine Learning.
- [3] X. Francis Jency, V.P.Sumathi, Janani Shiva Sri, —An Exploratory Data Analysis for Loan Prediction Based on Nature of the Clientsl, International Journal of Recent Technology and Engineering (IJRTE), Vol. 7, No. 48, pp. 176-179, 2018.
- [4] Anchal Goyal, Ranpreet Kaur, —Loan Prediction Using Ensemble Technique, International Journal of Loand Research in Computer and Communication Engineering, Vol. 5, Issue 3, pp. 523 – 526, March 2016.
- [5] Aboobyda Jafar Hamid and Tarig Mohammed Ahmed, —Developing Prediction Model of Loan Risk in Banks using Data Miningl, Machine Learning andApplications: An International Journal (MLAIJ), Vol.3, No.1, pp. 1-9, March 2016.
- [6] Anchal Goyal, Ranpreet Kaur, —Accuracy Prediction for Loan Risk using Machine Learning Modelsl, International Journal of Computer Science Trends and Technology (IJCST), Vol. 4, Issue 1, pp. 52-57, Jan- Feb 2016.
- [7] Sudhamathy, Credit Risk Analysis and Prediction Modelling of Bank Loans using Rl, International Journal of Engineering and Technology (IJET), Vol. 8, No. 5, pp. 1954-1966, Oct-Nov 2016.
- [8] Rattle data mining tool: available from <http://rattle.togaware.com/rattle-download.htm>
- [9] Aafer Y, Du W &Yin H 2013, DroidAPIMiner: 'Mining API-Level Features for Robust Malware Detection in Android', in:

Security and privacy in Communication Networks Springer, pp 86-103.

[10] Ekta Gandotra, Divya Bansal, Sanjeev Sofat 2014, 'Malware Analysis and Classification:ASurvey'available from <http://www.scirp.org/journal/jis>

[11] Vincenzo Moscato, Antonio Picariello, Giancarlo Sperli, "A benchmark of machine learning approaches for credit score prediction, Expert Systems with Applications," Volume 165, 2021, 113986, ISSN 0957-4174, <https://doi.org/10.1016/j.eswa.2020.113986>.

[12] Monika Papouskova, Petr Hajek, "Two-stage consumer credit risk modelling using heterogeneous ensemble learning," Decision Support Systems, Volume 118, 2019, Pages 33-45, ISSN 0167-9236, <https://doi.org/10.1016/j.dss.2019.01.002>.

[13] Lin Ma, Xi Zhao, Zhili Zhou, Yuanyuan Liu, "A new aspect on P2P online lending default prediction using meta-level phone usage data in China, Decision Support Systems," Volume 111, 2018, Pages 60-71, ISSN 0167-9236, <https://doi.org/10.1016/j.dss.2018.05.001>.

[14] Yanzhang Wang, Yu Song, Yunqiang Yin, Yuyan Wang, Xin Ye, Dujuan Wang, Multi-view ensemble learning based on distance-to-model and adaptive clustering for imbalanced credit risk assessment in P2P lending, Information Sciences, Volume 525, 2020, Pages 182-204, ISSN 0020-0255, <https://doi.org/10.1016/j.ins.2020.03.027>.

Bank loan approval research paper (1).docx

ORIGINALITY REPORT

2%

SIMILARITY INDEX

2%

INTERNET SOURCES

1%

PUBLICATIONS

1%

STUDENT PAPERS

PRIMARY SOURCES

1

www.ijitee.org

Internet Source

1%

2

Submitted to University of Central Lancashire

Student Paper

1%

3

ebooks.iospress.nl

Internet Source

<1%

4

essay.utwente.nl

Internet Source

<1%

Exclude quotes On

Exclude matches Off

Exclude bibliography On