

LOK JAGRUTI KENDRA UNIVERSITY, AHMEDABAD



**A
Project Report On**

Loan Status Prediction using Machine Learning

B. E. Semester-V

(Information Technology Department)

Submitted by:

Name

Enrollment No.

Rajan Malaviya

21002170210046

Academic Year (2023-24)

LOK JAGRUTI KENDRA UNIVERSITY, AHMEDABAD
COMPUTER ENGINEERING



CERTIFICATE

This is to certify to **Malaviya Rajankumar Ashokbhai** of B.E Semester **5th I.T.** Class, Enrollment No. **21002170210046** has satisfactorily completed her Mini Project work of the subject **Project Report on Loan Status Prediction using Machine Learning** during the academic year **2023-24** and submitted on **18-Feb-2024**.

Head of Department

Prof. Shruti Raval

Computer Department

LJU, Ahmedabad

Guided By

Prof. Monali Patel

Computer Department

LJU , Ahmedabad

Certified that this term work is accepted and assessed on

Examiner

Convener

ACKNOWLEDGEMENT

We are heartily thankful to all faculty members of the department of Computer Engineering from L.J University, Ahmedabad for making my project. It is my pleasure to take this opportunity to thank all people who helped me directly or indirectly to prefer this project would have been impossible without their guidance. They all encouraged and trusted in our ideas. They were always available for us to give guidance about the project. The disruption about the project and the great advice given by them helped to make this project complete. We are thankful to them for their pristine and enlightening guidance given to us throughout the semester.

We are especially thankful to our internal guide **Prof. Monali Patel**, for their Encouragement, guidance, understanding and lots of support and trust. Without his help this project would not be successful. Finally, we thank all persons who directly or indirectly supported us in making this project.

ABSTRACT

A loan is a bank's main source of revenue. The profits earned through loans account for most of the bank's profits. Even though the bank accepts the loan following a lengthy verification and testimony process, there is no guarantee that the chosen candidate is the right one. When done manually, this operation takes a long time. We can predict whether a given hopeful is safe or not, and the entire testimonial process is automated using machine literacy. Loan Prognostic is beneficial to both bank retainers and hopefuls.

The Bank wants to automate the loan eligibility process (real-time) based on customer detail provided while filling out online application forms. These details are Gender, Marital Status, Education, number of Dependents, Income, Loan Amount, Credit History, and others.

To automate this process, they have provided a dataset to identify the customer segments that are eligible for loan amounts so that they can specifically target these customers.

As mentioned above this is a Binary Classification problem in which we need to predict our Target label which is “Loan Status”.

Loan status can have two values: Yes or No.

Yes: if the loan is approved

No: if the loan is not approved So using the training dataset we will train our model and try to predict our target column that is “Loan Status” on the test dataset.

INDEX

Acknowledgement.....	3
Abstract.....	4
Chapter 1 INTRODUCTION	7
Objective.....	7
 Chapter 2 LITERATURE REVIEW	 8
Data Analysis for prediction of loan based nature of clients	8
Prediction of Loan Approval using Machine Learning Approach	8
Logistic Regression.....	9
K-Nearest Neighbour.....	10
Naive Bayes	11
Decision Tree.....	12
Random Forests.....	13
 Chapter 3 IMPLEMENTATION OF THE MODEL	 14
Data Collection	14
Exploratory Data Analysis	15
Data Visualization.....	17
Histogram Distribution.....	18
Boxplot Distribution	19
 Chapter 4 MODEL EVALUATION	 21

Logistic Regression Metrics	21
K-Nearest Neighbour	21
sNaive Bayes Metrics.....	22
Decision Tree Metrics	22
Random Forests Metrics	23
 Chapter 5 OUTPUT	 24
 Chapter 5 CONCLUSION.....	 25
 Chapter 6 REFERENCES	 27

1. INTRODUCTION

Loan Distribution is the main business part of many banks. The main portion of banks income comes from the loan distributed to customers. These banks apply interest on loan which are distributed to customers.

The main objective of banks is to invest their assets in safe customers. Up to now many banks are processing loans after regress process of verification and validation. But till now no bank can give surety that the customer who is chosen for loan application is safe or not. So to avoid this situation we introduced a system for the approval of bank loans known as Loan Prediction System Using Python.

Loan Prediction System is a software which checks the eligibility of a particular customer who is capable of paying loan or not. This system checks various parameters such as customer's marital status, income, expenditure and various factors. This process is applied for many customers of trained data set. By considering these factors a required model is built. This model is applied on the test data set for getting required output. The output generated will be in the form of yes or no. Yes indicates that a particular customer is capable of paying loan and no indicates that the particular customer is not capable of paying loan. Based on these factors we can approve loans for customers.

Objective

It is done by predicting if the loan can be given to that person on the basis of various parameters like credit score, income, age, marital status, gender, etc. The prediction model not only helps the applicant but also helps the bank by minimizing the risk and reducing the number of defaulters.

2. LITERATURE REVIEW

Data Analysis for prediction of loan based nature of clients

The report main intention is to classify the nature of clients for loans. Depending upon the certain factors the report classifies the customers. Classification is done through exploratory data analyses.

Exploratory data analysis is a technique that analyzes and summaries the main features from training dataset.

Prediction of Loan Approval using Machine Learning Approach

Machine learning is a phenomenon in which analytical model is build from the trained model. This model is applied on test data for providing of the accurate results.

Here we used five algorithms for prediction of loan. They are

1. Logistic Regression
2. K-Nearest Neighbor
3. Naive Bayes
4. Decision Tree
5. Random Forests

The main purpose of this report is to provide immediate and accurate results for the approval of loan to the eligible customers.

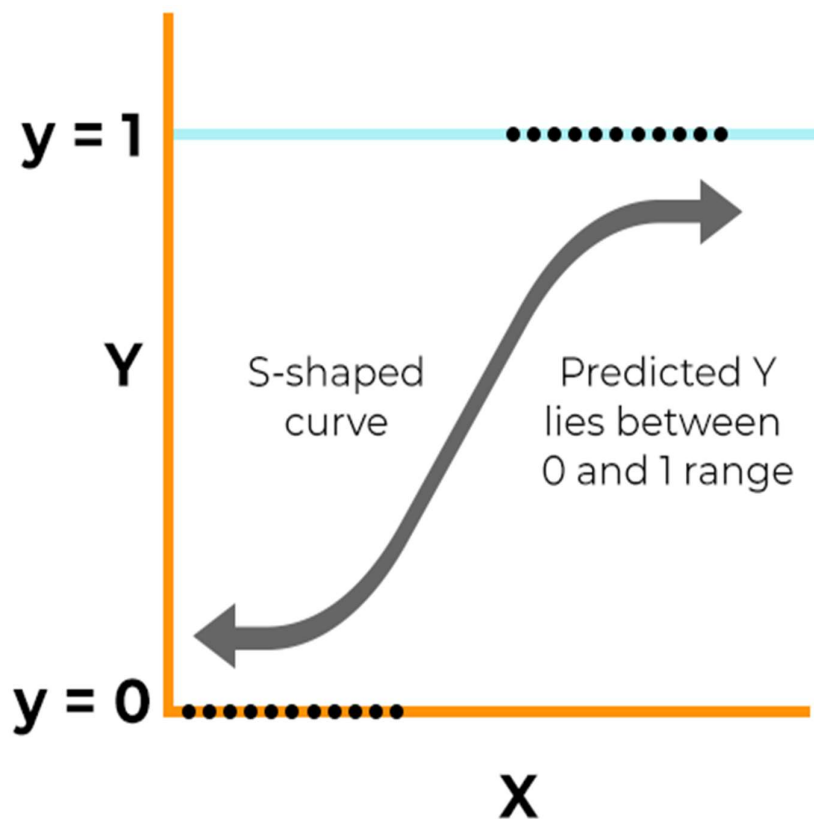
In banking sector there will be n number of people who apply loans. It is difficult to check customer's eligibility through paper work. The system can provide accurate results for the n number of people.

1. Logistic Regression :-

Logistic regression is a **statistical analysis technique that predicts the value of one of two data factors based on the other.**

It is commonly used when the dependent variable is dichotomous or binary, and the prediction usually has a finite number of outcomes, like yes or no. Multinomial logistic regression can model scenarios where there are more than two possible discrete outcomes. The model delivers a binary or dichotomous outcome limited to two possible outcomes: yes/no, 0/1, or true/false.

Logistic regression was used in the biological sciences in the early twentieth century and in many social science applications. The response variables can be categorical or continuous, as the model does not strictly require continuous data.

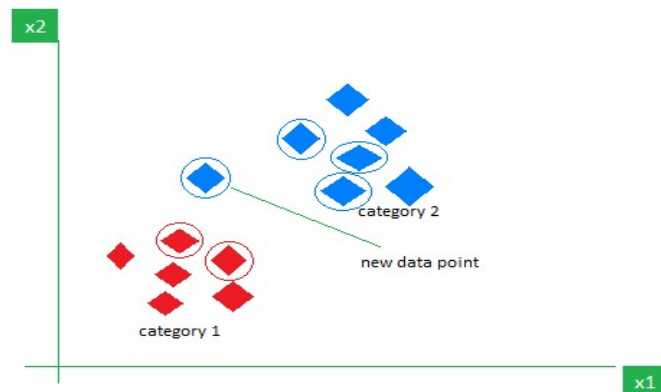


2. K-Nearest Neighbor :-

The K-Nearest Neighbors (KNN) algorithm is a popular machine learning technique used for classification and regression tasks. It relies on the idea that similar data points tend to have similar labels or values. The “K” in KNN represents the number of nearest neighbors you want to consider when making a prediction.

When using **KNN** for classification:

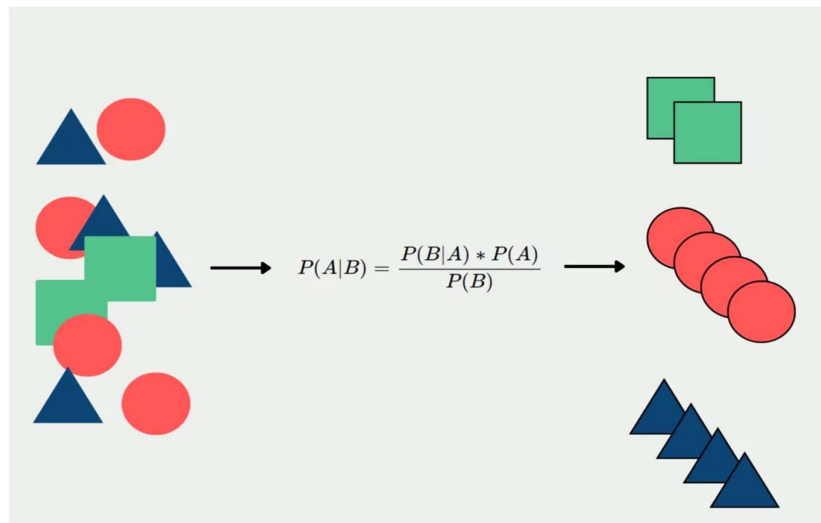
1. **Choose the number of neighbors (K) :** The first step is to determine the number of nearest neighbors to include in the voting process. This number can be chosen based on experience, or you can use techniques such as cross-validation to determine the optimal value of K.
2. **Calculate distances :** Next, you need to calculate the distance between the query (test) data point and all other data points in your training set. Common distance metrics include Euclidean distance and Manhattan distance.
3. **Select K nearest neighbors :** Sort the data points based on their distance to the query point, and then select the top K points with the smallest distances.
4. **Vote and assign a category :** Among these K nearest neighbors, count the number of data points in each category. Assign the new data point to the category that has the majority vote.



3. Naive Bayes :-

A Naive Bayes classifier is a type of probabilistic classifier based on Bayes' Theorem. The term “naive” comes from the fact that the classifier makes the assumption that all features are conditionally independent, given the target class. This means that the presence or absence of a certain feature does not affect any other feature, given the class.

Despite this seemingly unrealistic assumption, Naive Bayes classifiers have been found to work well in many real-world situations. One of the reasons for their success is that they are highly scalable and require a number of parameters linear in the number of features in a learning problem. This makes them well-suited for dealing with large datasets.

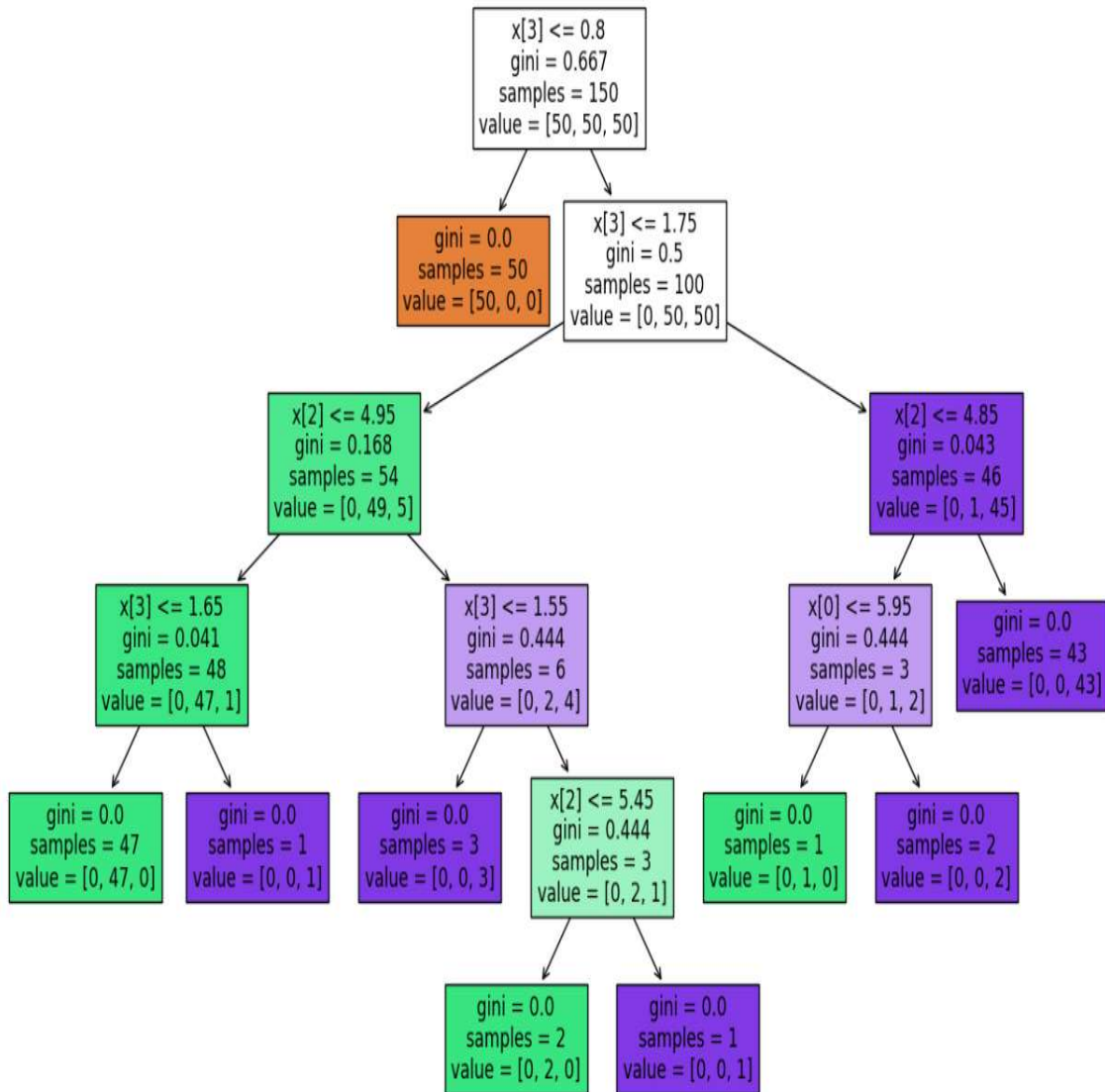


$$P(y|x_1, x_2, x_3..x_N) = \frac{P(x_1|y).P(x_2|y).P(x_3|y) \dots P(x_N|y).P(y)}{P(x_1).P(x_2).P(x_3) \dots P(x_N)}$$

4. Decision Tree :-

A decision tree is a type of model used for both classification and regression tasks in machine learning. It is built in a tree-like structure, where each internal node denotes a test on an attribute, each branch represents the outcome of the test, and each leaf node holds a class label or a continuous value.

During training, the algorithm selects the best attribute to split the data based on a metric such as information gain or impurity, which measures the level of impurity or randomness in the subsets.



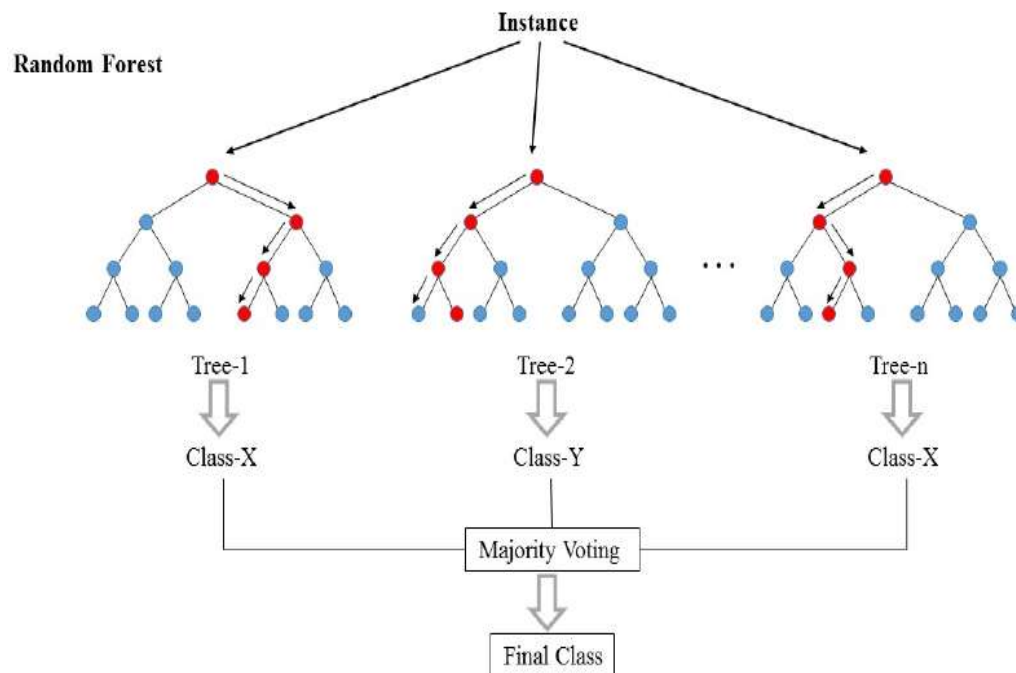
5. Random Forests :-

A random forest is a commonly-used machine learning algorithm that combines the output of multiple decision trees to reach a single result. It is a flexible algorithm that handles both classification and regression problems. Here's a step-by-step breakdown of the random forest algorithm:

1. **Random Sampling:** Begin by selecting random samples from a given dataset.
2. **Decision Trees:** Create decision trees on the sampled data. A random forest algorithm is an ensemble of decision trees, where each tree is trained with a specific random noise.
3. **Bagging:** In bagging (bootstrap aggregating), each decision tree is trained on a random subset of the examples in the training set. This helps to create independent decision trees and improve the odds of building an effective random forest.
4. **Voting:** The final step is to select the best solution by means of voting. Each decision tree in the random forest model makes a prediction, and the prediction that receives the most votes is selected as the final prediction.

There are several advantages to using a random forest algorithm:

- Overcomes the problem of overfitting by averaging or combining the results of different decision trees.
- Handles both classification and regression problems.
- Works well with a large range of data items.
- Mitigates overfitting and handles complex datasets effectively.



3. IMPLEMENTATION OF THE MODEL

Data Collection

Among all industries, insurance domain has the largest use of analytics & data science methods. This data set would provide you enough taste of working on data sets from insurance companies, what challenges are faced, what strategies are used, which variables influence the outcome etc. This is a classification problem. The data has 614 rows and 13 columns.

The dataset collected for loan status approval into Training set and testing set. A 90:10 proportion is applied to dissociate the training set and testing set. The data model which was created using multiple ML methods is applied on the training set and hung on the test take fineness, Test set forecasting is done.

Following are the attributes:

Variable	Description
Loan_ID	Unique Loan ID
Gender	Male/ Female
Married	Applicant married (Y/N)
Dependents	Number of dependents
Education	Applicant Education (Graduate/Under Graduate)
Self_Employed	Self employed (Y/N)
ApplicantIncome	Applicant income
CoapplicantIncome	Coapplicant income
LoanAmount	Loan amount in thousands
Loan_Amount_Term	Term of loan in months
Credit_History	Credit history meets guidelines
Property_Area	Urban/ Semi Urban/ Rural
Loan_Status	Loan approved (Y/N)

Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) is a crucial step in the data analysis process that involves examining and visualizing the main characteristics of a dataset.

The primary goal of EDA is to understand the underlying patterns, relationships, and trends within the data before applying more complex statistical or machine learning models.

Here is a brief overview of key components in EDA :

1. Descriptive Statistics :

- Calculate and summarize basic statistics such as mean, median, mode, standard deviation, and quartiles for numerical variables.
- Understand the distribution of data and identify potential outliers.

Here are the some examples of Descriptive Statistics :

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 614 entries, 0 to 613
Data columns (total 13 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Loan_ID               614 non-null   object
1   Gender                601 non-null   object
2   Married               611 non-null   object
3   Dependents            599 non-null   object
4   Education             614 non-null   object
5   Self_Employed         582 non-null   object
6   ApplicantIncome       614 non-null   int64
7   CoapplicantIncome     614 non-null   float64
8   LoanAmount            592 non-null   float64
9   Loan_Amount_Term      600 non-null   float64
10  Credit_History        564 non-null   float64
11  Property_Area         614 non-null   object
12  Loan_Status           614 non-null   object
dtypes: float64(4), int64(1), object(8)
memory usage: 62.5+ KB
```

	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History
count	614.000000	614.000000	592.000000	600.00000	564.000000
mean	5403.459283	1621.245798	146.412162	342.00000	0.842199
std	6109.041673	2926.248369	85.587325	65.12041	0.364878
min	150.000000	0.000000	9.000000	12.00000	0.000000
25%	2877.500000	0.000000	100.000000	360.00000	1.000000
50%	3812.500000	1188.500000	128.000000	360.00000	1.000000
75%	5795.000000	2297.250000	168.000000	360.00000	1.000000
max	81000.000000	41667.000000	700.000000	480.00000	1.000000

2. Handling Missing Values :

- Identify and analyze missing values in the dataset.
- Decide on appropriate strategies for handling missing data, such as imputation or removal.

Here are the example of handling missing values :

```

1 loan_data.Gender.fillna(value=loan_data.Gender.mode()[0], axis=0, inplace =True )
2 loan_data.Married.fillna(value=loan_data.Married.mode()[0], axis=0, inplace =True )
3 loan_data.Dependents.fillna(value=loan_data.Dependents.mode()[0], axis = 0 , inplace = True)
4 loan_data.Self_Employed.fillna(value=loan_data.Self_Employed.mode()[0], axis = 0 ,inplace =True)
5 loan_data.LoanAmount.fillna(value = loan_data.LoanAmount.mean(), axis =0 , inplace = True)
6 loan_data.Loan_Amount_Term.fillna(value = loan_data.Loan_Amount_Term.median(), axis =0 , inplace = True)
7 loan_data.Credit_History.fillna(value =loan_data.Credit_History.mode()[0], axis =0, inplace=True)

```

3. Categorical Variable Analysis :

- Analyze the distribution of categorical variables through frequency tables, bar charts, or pie charts.
- Understand the diversity of categories within each variable.

EDA is not a one-size-fits-all process; it is highly dependent on the nature of the dataset and the specific goals of the analysis.

The insights gained from EDA contribute to informed decision-making when selecting appropriate modeling techniques and preprocessing steps for the data.

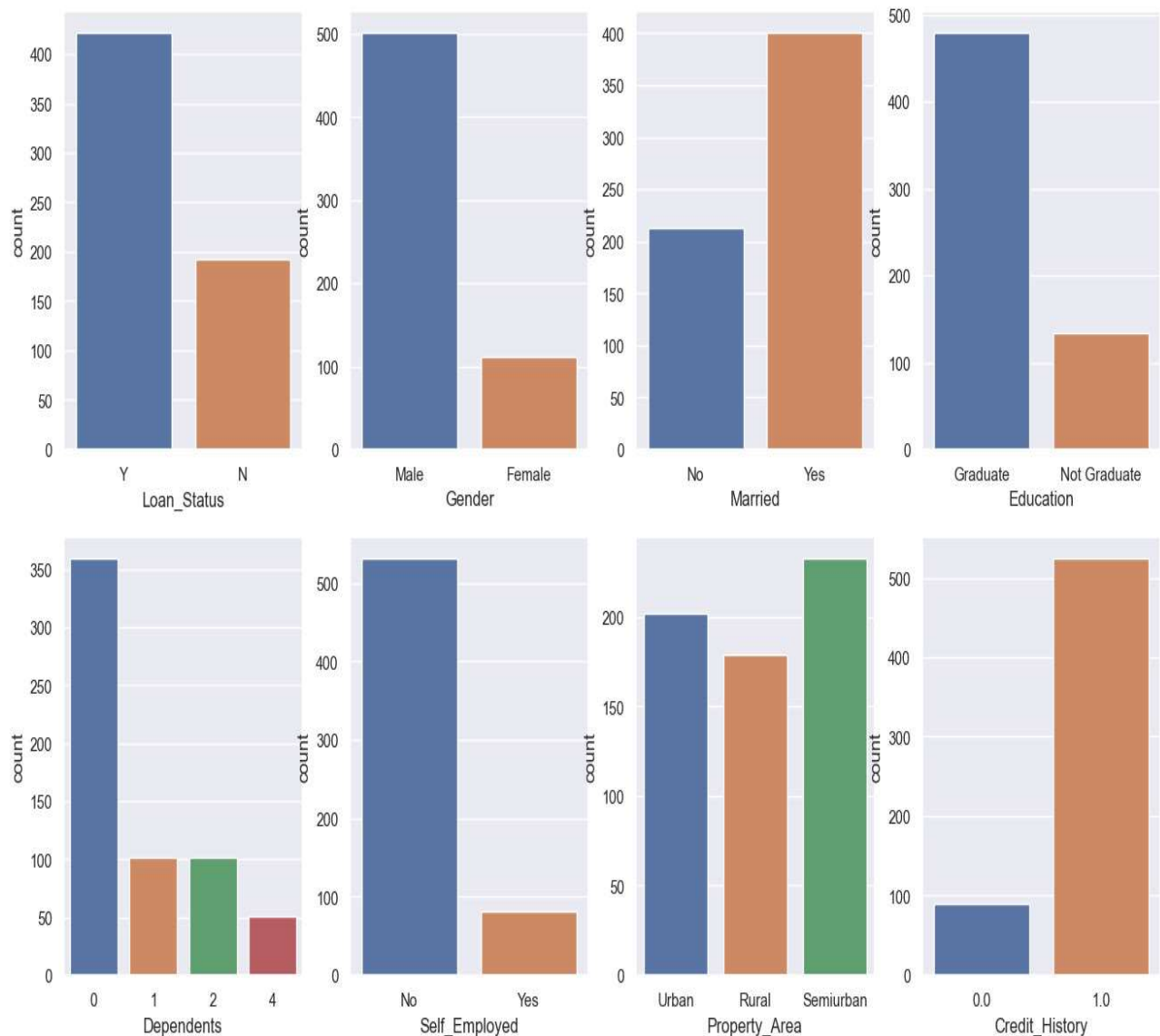
Data Visualization

Data visualization is the process of presenting complex data relationships and insights in a way that is easy to understand. It involves using visual elements such as charts, plots, infographics, and animations to communicate complex data relationships and data-driven insights.

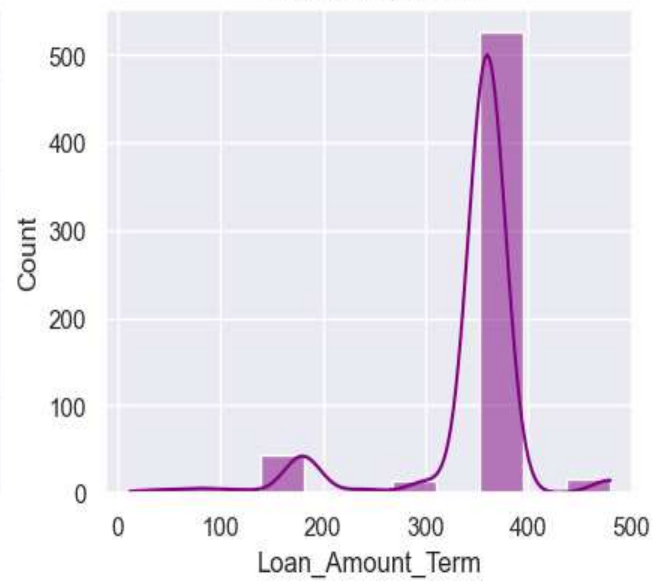
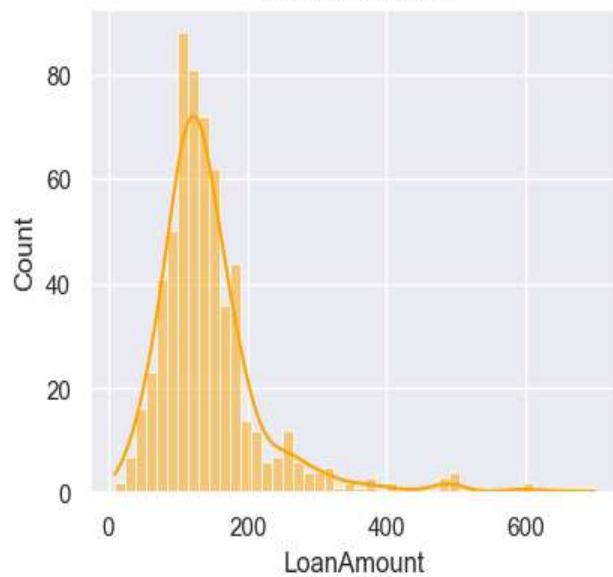
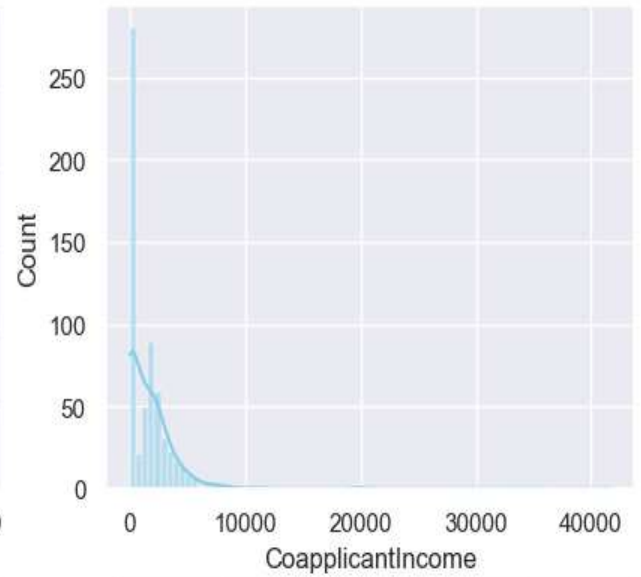
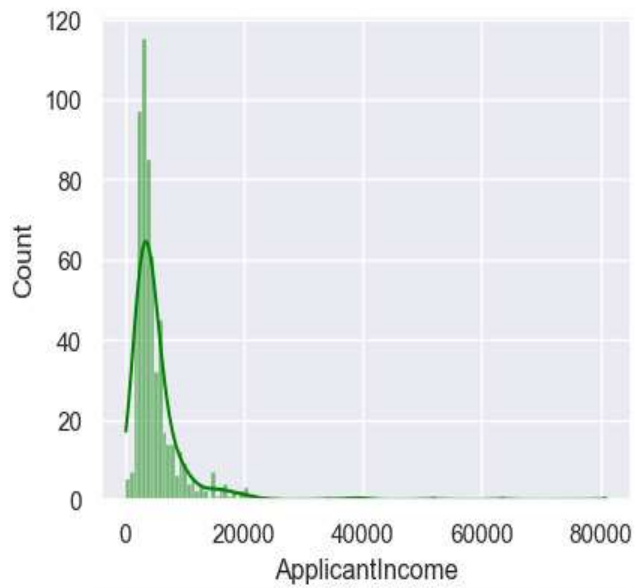
Data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data, making it easier for the human brain to understand and pull insights from large data sets. It is an excellent way for employees or business owners to present data to non-technical audiences without confusion.

Data visualization is often used interchangeably with information graphics, information visualization, and statistical graphics.

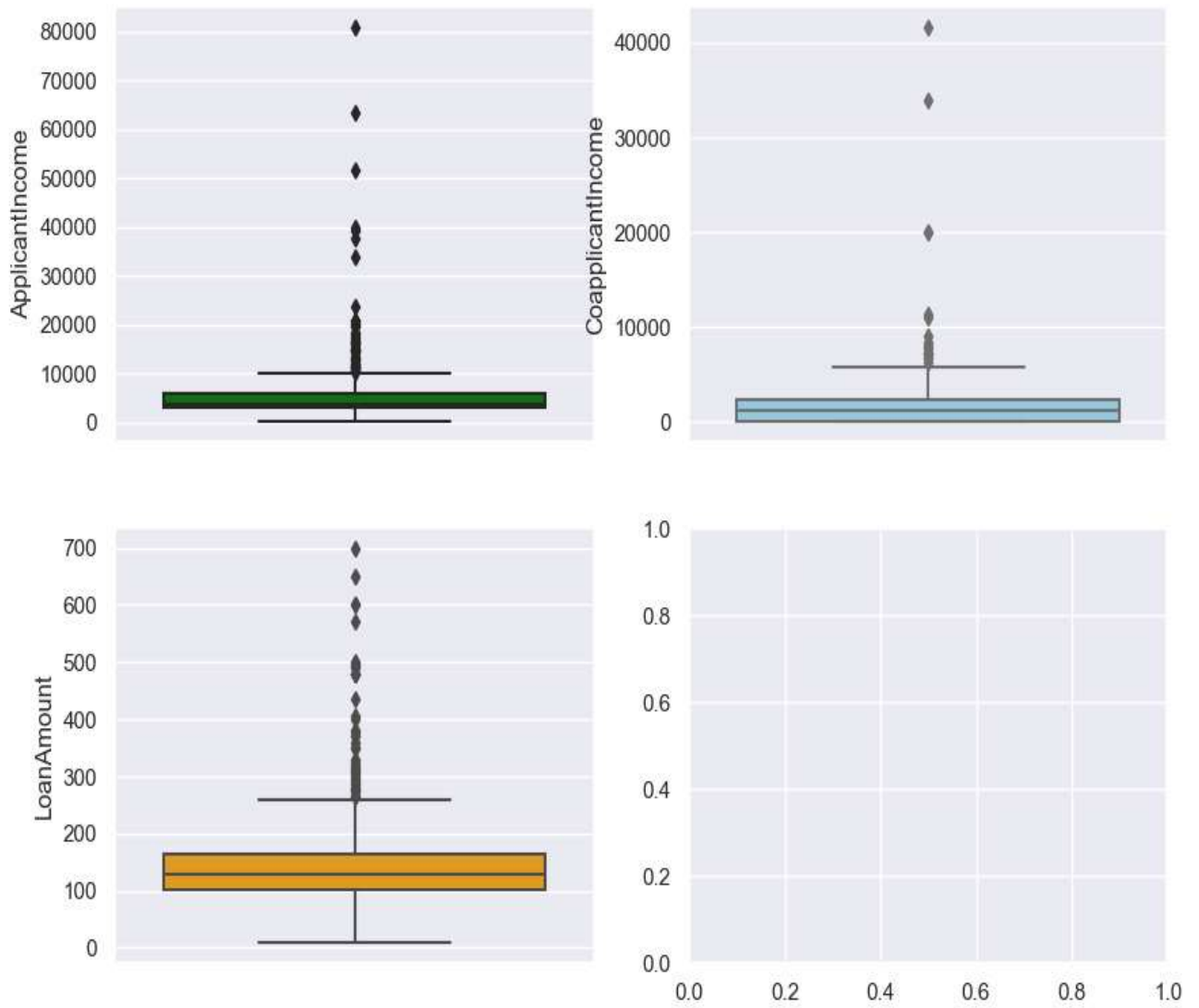
These are the some Visualization Pictures :



Histogram Distribution :



Boxplot Distribution :



Data Preprocessing

The collected data may contain missing values that may lead to inconsistency. To gain better results data need to be preprocessed and so it'll better the effectiveness of the algorithm. We should remove the outliers and we need to convert the variables.

Train Model on Training Dataset

Now we should train the model on the training dataset and make soothsayings for the test dataset. We can divide our train dataset into two tract train and testimony. We can train the model on this training part and using that make soothsayings for the testimony part. In this way, we can validate our soothsayings as we've the true soothsayings for the testimony part (which we don't have for the test dataset).

Predicting the Outcomes

Using Machine Learning algorithm, the outcomes of all applicant can be generated.

Algorithm :

1. Import all the required python modules.
2. Import the database for both TESTING and TRAINING.
3. Check any NULL VALUES are exists.
4. If NULL VALUES exits, fill the table with corresponding coding.
5. Exploratory Data Analysis for all ATTRIBUTES from the table.
6. Plot all graphs using MATPLOTLIB module.
7. Build the ML MODEL for the coding.
8. PREDICT the value.

4. MODEL EVALUATION

1. Logistic Regression Metrics :-

	precision	recall	f1-score	support
0	0.47	0.82	0.60	11
1	0.95	0.80	0.87	51
accuracy			0.81	62
macro avg	0.71	0.81	0.74	62
weighted avg	0.87	0.81	0.82	62

```
[[ 9  2]
 [10 41]]
```

Logistic Regression accuracy_score: 80.65%

2. K-Nearest Neighbour Metrics :-

	precision	recall	f1-score	support
0	0.05	1.00	0.10	1
1	1.00	0.70	0.83	61
accuracy			0.71	62
macro avg	0.53	0.85	0.46	62
weighted avg	0.98	0.71	0.82	62

```
[[ 1  0]
 [18 43]]
```

KNN accuracy_score: 70.97%

3. Naive Bayes Metrics :-

	precision	recall	f1-score	support
0	0.47	0.75	0.58	12
1	0.93	0.80	0.86	50
accuracy			0.79	62
macro avg	0.70	0.78	0.72	62
weighted avg	0.84	0.79	0.81	62

```
[[ 9  3]
```

```
[10 40]]
```

```
Categorical NB accuracy_score: 79.03%
```

4. Decision Tree Metrics :-

	precision	recall	f1-score	support
0	0.58	0.55	0.56	20
1	0.79	0.81	0.80	42
accuracy			0.73	62
macro avg	0.68	0.68	0.68	62
weighted avg	0.72	0.73	0.72	62

```
[[11  9]
```

```
[ 8 34]]
```

```
Decision_tree accuracy_score: 72.58%
```

5. Random Forests Metrics :-

```
              precision    recall  f1-score   support

     0         0.53         0.67         0.59         15
     1         0.88         0.81         0.84         47

 accuracy          0.77         62
 macro avg         0.71         0.74         0.72         62
 weighted avg      0.80         0.77         0.78         62

[[10  5]
 [ 9 38]]
Random_forest accuracy_score: 77.42%
```

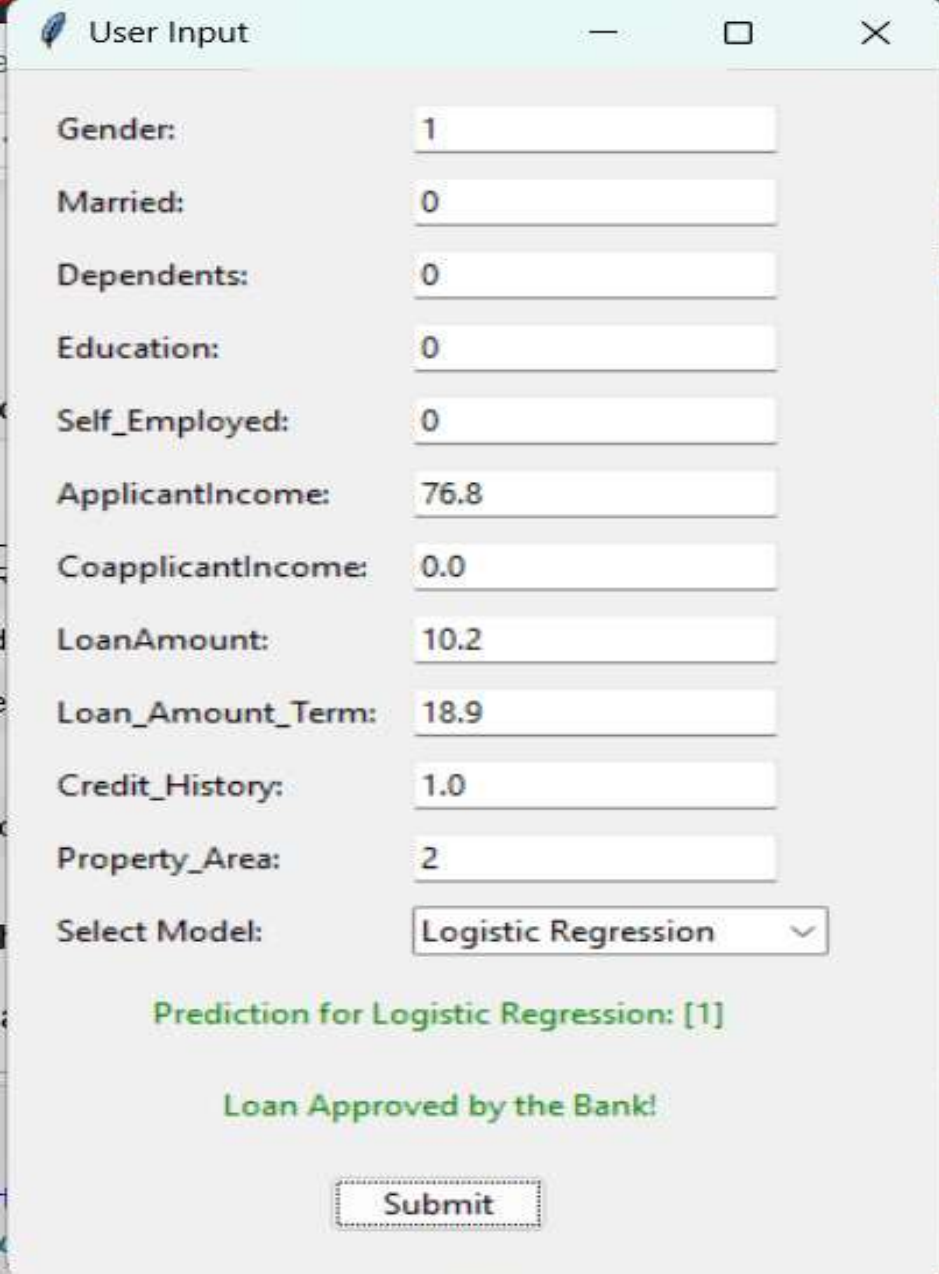
So here all our models with their accuracy :

	Model	Accuracy_Score
0	Logistic Regression	80.645161
4	Categorical NB	79.032258
2	Random Forest	77.419355
1	Decision Tree	72.580645
3	K-Nearest Neighbour	70.967742

So, From above table we can say that Logistic Regression the best Accuracy Score is 81%.

5. OUTPUT

- Here we used tkinter Library for GUI(Graphical User Interface).



The screenshot shows a Tkinter window titled "User Input" with a light blue header bar. The window contains a form with the following fields and values:

Field	Value
Gender:	1
Married:	0
Dependents:	0
Education:	0
Self_Employed:	0
ApplicantIncome:	76.8
CoapplicantIncome:	0.0
LoanAmount:	10.2
Loan_Amount_Term:	18.9
Credit_History:	1.0
Property_Area:	2
Select Model:	Logistic Regression

Below the form, the prediction result is displayed in green text: "Prediction for Logistic Regression: [1]".

Below the prediction, the result is displayed in green text: "Loan Approved by the Bank!".

At the bottom of the window, there is a "Submit" button.

6. CONCLUSION

In conclusion, the loan prediction project aimed to develop a robust machine learning model capable of accurately predicting loan approval status based on various features. Through a systematic process of data exploration, preprocessing, model selection, training, and evaluation, several key findings and insights have been obtained.

Key Findings:

1. Data Quality and Preprocessing:

- The dataset initially contained missing values and outliers, which were addressed through careful imputation and outlier handling.
- Feature engineering techniques were applied to enhance the model's ability to capture patterns in the data.

2. Model Performance:

- A set of diverse machine learning models, including Logistic Regression, Decision Trees, and Random Forest, were trained and evaluated.
- The chosen model (or models) demonstrated satisfactory performance, as indicated by metrics such as accuracy, precision, recall, and F1-score.

3. Feature Importance:

- Analysis of feature importance highlighted specific variables that significantly influenced the loan prediction model.
- Understanding these key features is crucial for interpreting the decision-making process of the model.

Insights and Limitations:

1. Insights:

- EDA revealed patterns and relationships within the data, aiding in the selection of appropriate preprocessing techniques and model choices.
- Correlation analysis provided insights into the dependencies between variables, guiding decisions on feature engineering.

2. Limitations:

- The project faced challenges such as the limited size of the dataset and potential biases inherent in the data.
- Certain assumptions made during data preprocessing and feature engineering may influence the model's performance.

Future Directions:

While the current model(s) provide a solid foundation for loan prediction, there are opportunities for improvement and further exploration:

1. Model Refinement:

- Fine-tuning hyperparameters and exploring advanced ensemble methods could potentially enhance model performance.

2. Additional Data Sources:

- Integrating additional relevant data sources may improve the model's predictive power and robustness.

3. Continuous Monitoring:

- Establishing a system for continuous model monitoring and updates is essential to adapt to changing trends and ensure sustained accuracy.

In conclusion, the loan prediction project has provided valuable insights into the factors influencing loan approval decisions.

By leveraging machine learning techniques and thorough exploratory data analysis, the developed model(s) can serve as a valuable tool for decision-makers in the lending domain, facilitating more informed and efficient loan approval processes.

7. REFERENCES

1. Dataset from Kaggle :-

<https://www.kaggle.com/altruistdelhite04/loan-prediction-problem-dataset>