An Extensive Analysis of Loan Approval Status based on Customers' profiles

N. Arjun - COHNDDS232F-005

Akithma Fernando - COHNDDS232F-003

Samrooth - COHNDDS232F-019

Higher National Diploma in Data Science

National Institute of Business Management

Colombo, Sri Lanka

01tst December 2023

A Project Proposal submitted for the partial fulfillment of the requirements of the Advanced Diploma in Data Science (Full-time) Programme.

Abstract

Our aim throughout this project is to analyze the machine learning models to identify which model most accurately predicts the loan approval status. The dataset is obtained from the Kaggle website. It gives insights about the applicant's profile that is analyzed by the bank or financial institution to decide whether they can provide the loan or not for the applicant. In this case, we analyze the accuracy scores and AUC scores to identify what model has having high accuracy score to predict correctly. On the other hand, with the help of graphical representations, we have identified the main factors that are affecting the loan approval status. And how the banks consider these factors in applicants to provide loans. With the analysis of the applicant's income area, education qualifications, dependents, credit histories, and employment they will decide on loan approval.

Table of Contents

List of Abbreviations:

ROC - Receiver operating characteristic

AUC - Area under the curve

CSV - Comma-Separated Values

Q1 - 25th percentile

Q2 - 75th percentile

Std - Standard deviation

Max - Maximum

Min - Minimum

1. Chapter 1: Introduction

1.1 Background

Loan approval is an important decision that impacts all individuals and businesses in the field of finance. There is a responsibility for all financial institutions to assess the creditworthiness, and accuracy of loan application evaluation and minimize the risk while maximizing the efficiency and the profitability. The traditional approach that is followed usually is time-consuming and can lead to inconsistencies and potential biases. Therefore, to overcome these limitations, it is necessary to introduce a data analytics and machine learning model for an automated loan approval system.

In this study, we have explored the complexities of approving loans for customers based on their profiles. Our objective is to conduct an extensive analysis of loan approval for the customers based on their profiles by using a comprehensive dataset that includes the customer attributes such as Gender, Marital status, Dependents, Education, Self-Employment status, Applicant and Co-Applicant Income, Loan amount, Term, Credit History, Area, and Status.

By achieving the objectives of our project, we will be able to present a proper analysis of the financial institution that can help to improve the loan approval process.

1.2 Research Problem

Normally, people apply for loans because of their purposes such as housing developments, business startup plans, and purchase of vehicles. To decide on whether to approve or not approve the loan amount, banks should consider the customer details of those who applying for the loan. They consider the applicant's past credit history, applicant's income, co-applicant income, and dependents of the loan applicant. Some of the limitations are challenging to make decisions. Our study approaches the best understanding of the loan process. With all of this information, the bank will decide whether yes or no for the loan approval and they determine the loan amount.

The credit history represents that if the applicant was approved for the loan in the past period, did he pay the interest amount for the loan properly or did the applicant not pay back the amount properly? According to that if the applicant has a good credit history, it is a plus point for the applicant. Also, there are some other factors that mainly affect the loan approval status. Therefore, the bank will consider the customer profile to make a decision about the loan amount.

1.3 Research Questions

To analyze the loan status, it is crucial to understand the suitable machine learning model and the variables that are affecting the loan status. We can make decisions with the answers to the following research questions.

- Which machine learning model has a high accuracy score for predicting the loan approval status?

- What are the variables that are statistically significant? and by analyzing p-values, coefficients, F-statistic, and r-squared values, evaluating the independent variable to explain model fit.

- How do loan amounts depend on the other related variables with different applicant profiles?

1.4 Objectives of the Project

Objectives of this project are:

- Analyzing machine learning models such as Logistic regression, Random Forest, and Decision tree to identify which model is suitable for predicting loan approval status with applicant information.

- Analyze the linear regression model to identify the variables and whether they are statistically significant or not for target variable loan amounts based on the OLS regression report values such as p-value, coefficient, and r-squared value.

- Visualize and explore the factors such as education, self-employed, income distributions, loan amounts, credit histories, and all other specific variables that are involved in loan approvals to identify how they affect the loan status.

1.5 Scope of the Research

As already mentioned, based on the applicant's details and past credit history banks will decide to give loan approval status. We are specifically analyzing the variables that influence the approval status.

Our main focus of this project is to analyze the dataset with machine learning techniques and identify the best model with the higher accuracy score. It will give insights into the loan approval status. In this project, we used Classification models, statistical methods, and graphical representations for the analysis.

1.6 Justification of the Research

When it comes to the loan approval status, banks consider the variables that are to identify the customer profile. In this case, past credit history, applicant’s income, co-applicant income, and dependents are the main factors that the banks analyze about the applicant to decide the approval status.

While considering the dependents, it represents the family member count of the loan applicant. Even though the applicant has a high income, because of the higher family member count the income will mostly go for their personal purposes. Therefore, the bank will take it as a minus point for the applicant to decide on the loan approval status. In summary, we use machine learning techniques for our project to give an insight into the factors that affect the loan approval status for making the decision to approve or not. Banks can make decisions based on all of this customer profile information.

1.7 Expected Limitations

Our project is based on a secondary dataset, which we gathered from the Kaggle dataset. It is challenging to find out the variable descriptions. The author did not clearly mention the all variables. However, we are clear about the variables from the explanations about the loan statements which are given in the description. We took this dataset as a balanced dataset.

According to our objectives, the linear regression model will give insights into the linear relationship between the X (independent) and Y(dependent) variables. If the output is related to a non-linear relationship, then we should consider this as an unsuitable model. To determine the loan amount, the bank should consider the various factors not only the variables that are included in the dataset. Since they have mentioned the basic main factors, we analyze these variables.

1.8 Proposed work schedule

Project Initiation; 2nd of November 2023

Data collection; 6th of November 2023

presentation; 04th of December 2023

Completion of the report; 30th of November 2023

Data analysis; From 11th of November 2023 to 24th of November 2023

Submission; 01st of December 2023

2. Chapter 2: Literature Review

2.1 Introduction to the research theme

The theme of this research is to explore the factors influencing lending decisions which means the interplay of various customer attributes and their impact on the loan approval process within financial institutions using automated loan approval systems. Understanding how the elements of customer profiles such as credit history, income, employment status, and demographic information influence loan approval can significantly enhance the accuracy of predicting who is more likely to get a loan. This research can help financial institutions to expand access to loans for deserving individuals based on their genuine qualifications and needs.

2.2 Findings by other researchers

A study published in 2022 “Loan Approval Prediction Based on Machine Learning Approach” focused on machine learning algorithms to predict loan approval based on a bank credit dataset. Eight algorithms such as the Logistic Regression algorithm, Random Forest, Decision trees, Linear Regression, Support Vector Machine (SVM), Naïve Bayes, K-means, and K Nearest Neighbors (KNN) were done and the accuracy, precision, recall, and F1-score were evaluated. Logistic Regression has high accuracy followed by Naïve Bayes and Random Forest and Support Vector Machine has higher precision. K-means and K-nearest neighbors are less effective. Overall, the algorithms received accuracy rates between 70% and 80%. The study aimed to focus on the factors influencing creditworthiness by analyzing 13 features and future studies will be exploring more dependable outcomes in determining creditworthiness for loan approval.(Azeez A. Nureni, O.E. Adekola, 2021)

A study published in 2020 on “ Accurate Loan Approval Prediction Based on Machine Learning Approach” focused on the prediction of loan approvals using machine learning algorithms such as Logistic Regression, Decision Tree, and Random Forest. It explores the significant role of loans in a bank by emphasizing the importance of credit risk assessment in banking. The ML-based prediction aims to automate the validation of loan application features and ease efficient applicant selection while maintaining privacy. To validate the applicant’s loan eligibility this study uses training and testing datasets and among three ML algorithms, Decision Tress has a high accuracy. (J. Tejaswini, T.Mohana, R.Devi, 2020)

Research on “Loan Approval Prediction Model A Comparative Analysis” found that the most accurate model to predict loan approvals is Decision Tress among the models used here such

as Logistic Regression, Decision Tree, and Random Forest. These models have focused on factors like credit score, income, and demographic details to predict loan eligibility by banks to minimize the approval time. Logistic Regression has the lowest accuracy in predicting loan approval while Random Forest is better at generalization. (Afrah Khan, Abishek Kumar, 2021)

Another study on "Prediction for Loan Approval using Machine Learning Algorithm" compares the effectiveness of different models and founds that models like SVM and Naïve Bayes are most suitable to predict loan approval. In that Naïve Bayes gives better results when compared to other models. Also, algorithms like SVM and Naive Bayes were employed to evaluate loan safety using past customer records. The automated system aids loan processing and loan approval for both banks and loan applicants. (Ashwini, Shraddha, Ankita, 2021)

A study published in 2022 on "Loan Approval Prediction Using Machine Learning" states the need for accurate loan approval systems in the banking sector using a Random Forest algorithm. The system identifies the customers who are eligible for loans based on some factors such as marital status, income, and expenditures. The system provides a binary output of 'yes' or 'no' which means the customer can repay the loan or not. Finally, this study highlights the accuracy of the system for diverse loan applicants and improves the algorithm to make the system even better. (P.L.Srinivasa, G.Soma Shekar, P.Rohith, 2020)

Research on "Loan Approval Prediction" focussed on predicting loan approvals and subprime mortgage risks using machine learning algorithms. The models that are used in this research study are GRU, Bi-LSTM, and Logistic Regression and GRU has the highest accuracy at 97%. This study helps financial institutions to provide loans by highlighting the importance of investigating factors such as payment history, and income. Also, financial institutions can

improve by implementing subprime mortgage risk model to reduce risk. (Pallapothu Nishita, Bipasha Bhowamik, 2023)

A study published in 2023 on “An ensemble machine learning based bank loan approval predictions system with a smart application” focused on estimating loan default risks using machine learning and deep learning algorithms in the banking sector. This study has identified the best models and built an ensemble classifier to predict bank loan default using models like Logistic Regression, Decision Tree, Random Forest, and neural networks such as DNN, RNN, and LSTM. They also developed a user-friendly interface for banks to evaluate loan eligibility. (Nazim Uddin, Sunil Aryal, Khabir Uddin, 2023)

Another study was done on “Predicting Bank Loan Eligibility Using Machine Learning Model and Comparison Analysis ” based on a dataset that contains applicant information such as age, income type, and credit history which has been used to predict the eligibility of loan applicants using machine learning algorithms. Among all the models Logistic Regression has the highest accuracy at 92%. The study has revealed that credit scores have the highest impact on loan approval while gender and marital status have less impact. Applicants with high incomes and smaller loan amounts are more likely to be approved. (Miraz Al Mamum, Afia, Muntasir, 2022)

A study published in 2021 on “Loan Forecast by Using Machine Learning” focuses on the loan approval process in the banking sector using machine learning algorithms such as Logistic Regression and decision Trees. The main aim is to predict whether providing a loan to an applicant will be safe for the bank. The achieved accuracy is 77.27% and cross-validation is 80.80%. (A.Kulothugan, 2021)

Another study “An Approach to Loan Approval Prediction Using Machine Learning” focused on the loan approval by banks to predict loan eligibility using machine learning algorithms like Logistic Regression, Random Forest tree, Extreme Gradient Boost, Decision- tree classifier and Support Vector Machine. According to the results obtained from the evaluations XGBoost has the best performance in accuracy, mean square error, recall, and F1-score. This study approaches a new way that loan decisions are made by both financial and nonfinancial elements such as gender, marital status, and education. (Yamuna, Praneeth, 2022)

2.3 Table for Variables, their definitions and sources

Table 2; Operationalization Table

Variables	**Definition**	**Sources**
Gender	The gender of the person (Male/Female)	Kaggle Loan Dataset
Married	Marital status of the loan applicant	Kaggle Loan Dataset
Dependents	Number of people dependent on the loan applicant	Kaggle Loan Dataset
Education	Educational level of the loan applicant	Kaggle Loan Dataset
Self Employed	Self-employment status of the loan applicant	Kaggle Loan Dataset
Applicant Income	Income of the loan applicant	Kaggle Loan Dataset

Co-applicant Income	The income of another individual involved in the loan application	Kaggle Loan Dataset
Loan Amount	The money requested/selected by the loan applicant/bank	Kaggle Loan Dataset
Term	It represents the maturity period that the applicant should pay back the loan.	Kaggle Loan Dataset
Credit History	Applicant's History of borrowing and repayment	Kaggle Loan Dataset
Area	The geographic location of the property for which the loan is being applied is situated	Kaggle Loan Dataset
Status	Whether the bank or the financial institution approved or not for the loan.	Kaggle Loan Dataset

3. Chapter 3: Methodology

3.1 Population, Sample, and Sampling Technique

The dataset that I retrieved from Kaggle represents real-world related circumstances about loan approvals. The loan approval status includes details on numerous factors the banks or other financial institutions consider. We took all the variables without removing anything for our

analysis. Our dataset contains 614 rows and 12 columns. The dataset has information about the approval status populations.

The loan dataset has self-employed information. From the analysis, we have gathered the information that they mostly approved loan amounts for the no self-employed category. This means they assume that if the applicant is self-employed, they are mostly having trouble repaying the loan back. Banks make this kind of decision based on their past records. This analysis explains whether they are yes or no for approving loan amounts. Our analysis will find clear information about the approval status based on the related variables that influenced factors.

3.2 Type of Data to be collected and data sources

The dataset for this project has been retrieved from Kaggle datasets. This is a secondary dataset. The secondary dataset stands for it is collected by someone else and I have used this dataset for my research purposes. It contains both numerical and categorical variables. The categorical variables are mostly binary. To understand the loan approval status the analysis part has graphical representations and machine learning models in our project. With the existing records, the dataset is focused on predictive data analysis methods.

By analysis of the related variables from the secondary dataset, we can conclude the factors that affect the loan approval status. Machine learning models and visualizations are done in this project to find the best predictive model.

3.3 Conceptual Framework

There are 12 variables under this dataset. We have changed the Term and Credit history from float to categorical for our research purpose. Except for the applicant's income, co-applicant income, and loan amount, all other variables are considered categorical type variables. To understand our research objectives, we should consider the dependent and independent variables. In this case, we mostly used the Status variable as our target variable. Other than that, all other variables are considered as independent variables. With the target variable, we are doing our project to predict the loan approval status. With all of the independent variables, we analyze whether the applicant is eligible or not for the loan amount.

Further explaining our workings, firstly we have done with the machine learning models. In this case, we took three classification techniques and linear regression analysis. The classification methods we used here are logistic regression, random forest, and decision tree. With all of these techniques, we identified the classification method that shows high accuracy in making predictions. After all of this analysis, we graphically visualize the variables to identify the variables and analyze how those variables depend on the loan approval status.

3.4 Hypothesis

There is a high chance of deciding to loan approval with the applicant's income and dependents. It shows the positive relationship between applicant income and co-applicant income to the loan amount. If the applicant's and co-applicant income are high then there is a high chance to get a

loan approval status as yes. On the other hand, an alternative hypothesis shows a high chance for loan approval status as ‘yes’ for a positive credit history. Also, between the urban, semi-urban, and rural areas, the loan approval status is dynamic. When it comes to whether self-employed or not, the banks or the financial institutions have mostly fixed the loan approval status as ‘yes’ for the not self-employed category. In this case, maybe they consider the self-employed might face losses in the future and it is hard to repay the loan amount. Therefore, they mostly give loan for the no-self-employee category.

3.5 Operationalization Table

Table 2; Operationalization Table

Variables	Indicators	Measures
Gender	Whether the loan applicant is Female or Male	Categorical (Male, Female) Binary
Married	Whether the loan applicant is married or not	Categorical (Yes, No)- Binary
Dependents	Family members count that applicant have. (0, 1, 2, 3+)	Categorical(0,1,2,3+)
Education	Whether the loan applicant is a graduate or not graduate.	Categorical (Graduate, Not graduate) - Binary

Self-employed	Whether the loan applicant is Self-employed or not	Categorical (Yes, No)- Binary
Applicant Income	Loan applicant's salary and other incomes.	Numerical
Coapplicant_Income	Co-applicant that the person who goes for the back support, his income.	Numerical
Loan Amount	The amount that the applicant has applied for the loan.	Numerical
Term	The term represents the duration for repaying the loan amount.	Categorical
Credit History	It explains did the applicant repaid the loan amount if he borrowed in the past years.	Categorical(0-bad, 1-good)- Binary
Area	The area where the applicant is from?	Categorical (Urban, Semiurban, Rural)
Status	It explains whether the loan is accepted or not.	Categorical (Y-Yes, N-No) - Binary

3.6 Methods of data analysis

To analyze our objectives, we have used machine learning techniques. We dropped null values. Also, completed all unique value identifications, converted the variables into proper data types, removed outliers, assigned dummy variables, and then the feature selection part. After completing the data preprocessing methods, we split the dataset into training and testing.

The three machine learning techniques are used here to achieve our first objective. They are logistic regression, random forest, and decision tree. According to previous research, strongly mentioned that logistic regression is the machine learning technique that most accurately predicts the loan approval status. Therefore, we have a clear understanding that logistic regression is a good model for this project. For the proving purpose, we took the other two classification methods. In our case, it shows logistic regression is a good model for our project.

The linear regression model has been taken for our 2nd objective. To identify the modal other than the classification methods, is there anything to predict most accurately? Our linear regression analysis shows there is no linear relationship between our independent variable and the target variable. This means linear regression is not a good model to make predictions.

Finally, we did a graphical representation analysis for our final objective. It explains how loan approval status depends on other factors. With all of this analysis, we can conclude how these factors affect the loan approvals for the applicants, and how the banks or other financial institutions analyze the applicant's profiles.

4. Chapter 4: Data Analysis

4.1 Data Analysis

Our project's main objective is to identify the best machine-learning model. With the most accurate prediction of loan approval status, we can conclude. As I already mentioned above, we are done with the three machine learning models. Graphical visualizations give insights into the dataset and it analyze the factors that are involved in the loan approval status. All analyses here are done with the Python Jupyter Notebook.

In the very first part, we input the dataset into the Jupyter Notebook to check the overview of the dataset. The dataset contains 614 rows and 12 columns.

4.1.1 Overview of the dataset.

	Gender	Married	Dependents	Education	Self_Employed	Applicant_Income	Coapplicant_Income	Loan_Amount	Term	Credit_History	Area	Status
0	Male	No	0	Graduate	No	584900	0.0	15000000	360.0	1.0	Urban	Y
1	Male	Yes	1	Graduate	No	458300	150800.0	12800000	360.0	1.0	Rural	N
2	Male	Yes	0	Graduate	Yes	300000	0.0	6600000	360.0	1.0	Urban	Y
3	Male	Yes	0	Not Graduate	No	258300	235800.0	12000000	360.0	1.0	Urban	Y
4	Male	No	0	Graduate	No	600000	0.0	14100000	360.0	1.0	Urban	Y
...	...	...	...	...	...	...	...	...	...	...	...	...
609	Female	No	0	Graduate	No	290000	0.0	7100000	360.0	1.0	Rural	Y
610	Male	Yes	3+	Graduate	No	410600	0.0	4000000	180.0	1.0	Rural	Y
611	Male	Yes	1	Graduate	No	807200	24000.0	25300000	360.0	1.0	Urban	Y
612	Male	Yes	2	Graduate	No	758300	0.0	18700000	360.0	1.0	Urban	Y
613	Female	No	0	Graduate	Yes	458300	0.0	13300000	360.0	0.0	Semiurban	N

614 rows × 12 columns

The dataset has object, int, and float data types. Also, it has the null values.

The null values and data types are as below.

4.1.2 The data type of the dataset

```
Gender                object
Married               object
Dependents            object
Education             object
Self_Employed         object
Applicant_Income       int64
Coapplicant_Income   float64
Loan_Amount            int64
Term                 float64
Credit_History       float64
Area                  object
Status                object
dtype: object
```

4.1.3 Missing values in each column

```
Gender               13
Married               3
Dependents           15
Education             0
Self_Employed        32
Applicant_Income      0
Coapplicant_Income    0
Loan_Amount           0
Term                 14
Credit_History       50
Area                  0
Status                0
dtype: int64
```

Since there are null values in the dataset, we have done with the removing null or missing values part. For that, we used “drop.na()” code. And then identified the unique values to analyze the

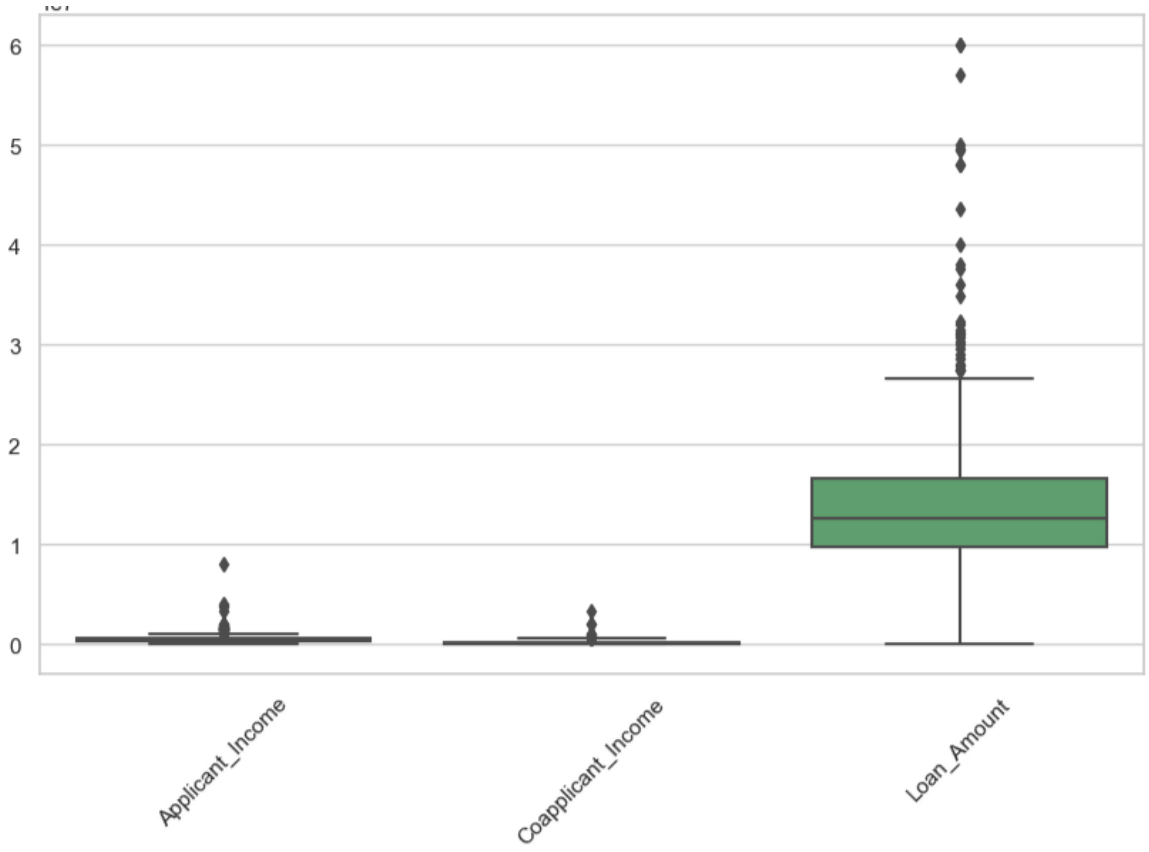
dataset to understand the categorical variables in the dataset. The unique values for each variable are given below:

4.1.4 Unique values:

```
The number of values for feature Gender :2 -- ['Female' 'Male']
The number of values for feature Married :2 -- ['No' 'Yes']
The number of values for feature Dependents :4 -- ['0' '1' '2' '3+']
The number of values for feature Education :2 -- ['Graduate' 'Not Graduate']
The number of values for feature Self_Employed :2 -- ['No' 'Yes']
The number of values for feature Applicant_Income :420
The number of values for feature Coapplicant_Income :241
The number of values for feature Loan_Amount :187
The number of values for feature Term :9 -- [ 36.  60.  84. 120. 180. 240. 300. 360. 480.]
The number of values for feature Credit_History :2 -- [0. 1.]
The number of values for feature Area :3 -- ['Rural' 'Semiurban' 'Urban']
The number of values for feature Status :2 -- ['N' 'Y']
```

After identifying the unique values, we processed them with the outliers. In this case, we took the dataset as balanced data. Furthermore, the dataset has outliers in the numerical variables' applicant income, co-applicant income, and loan amount.

4.1.5 Checking the outliers with boxplot:



For those numerical variables, we have done with the outliers-removing process. In this case, we calculate the first, third, and interquartile ranges. With that calculated the lower and the upper limits for the variables. Then, we removed the outliers with the below code:

4.1.6 Excluded the rows that have outliers:

```
data=data[(data['Loan_Amount']<26925000.0)&
          (data['Loan_Amount']>-475000.0)&
          (data['Applicant_Income']<1023425.0)&
          (data['Applicant_Income']>-152375.0)&
          (data['Coapplicant_Income']<562875.0)&
          (data['Coapplicant_Income']>-337725.0)]
data
```

The output is as below:

4.1.7 Output after removing the outliers:

	Gender	Married	Dependents	Education	Self_Employed	Applicant_Income	Coapplicant_Income	Loan_Amount	Term	Credit_History	Area	Status
0	Male	No	0	Graduate	No	584900	0.0	15000000	360.0	1.0	Urban	Y
1	Male	Yes	1	Graduate	No	458300	150800.0	12800000	360.0	1.0	Rural	N
2	Male	Yes	0	Graduate	Yes	300000	0.0	6600000	360.0	1.0	Urban	Y
3	Male	Yes	0	Not Graduate	No	258300	235800.0	12000000	360.0	1.0	Urban	Y
4	Male	No	0	Graduate	No	600000	0.0	14100000	360.0	1.0	Urban	Y
...	...	...	...	...	...	...	...	...	...	...	...	...
609	Female	No	0	Graduate	No	290000	0.0	7100000	360.0	1.0	Rural	Y
610	Male	Yes	3+	Graduate	No	410600	0.0	4000000	180.0	1.0	Rural	Y
611	Male	Yes	1	Graduate	No	807200	24000.0	25300000	360.0	1.0	Urban	Y
612	Male	Yes	2	Graduate	No	758300	0.0	18700000	360.0	1.0	Urban	Y
613	Female	No	0	Graduate	Yes	458300	0.0	13300000	360.0	0.0	Semiurban	N

433 rows × 12 columns

To find out whether standardization or normalization is suitable, we analyzed the p-values for each numerical variable. All the p-values are less than 0.05. And, the dataset contains outliers. In this case, if the p-values are below 0.05 then we can take it as a normal distribution. Hence, we did the normalization method.

4.1.8 P- Values:

```
Shapiro-Wilk Test for Applicant_Income: Statistic=0.9120369553565979, p-value=3.6021084051196355e-15
Shapiro-Wilk Test for Coapplicant_Income: Statistic=0.8434571623802185, p-value=2.6576626456796875e-20
Shapiro-Wilk Test for Loan_Amount: Statistic=0.9818273186683655, p-value=2.97612768918043e-05
```

Assigned dummy variables and values for categorical variables in the dataset with binary categorical. The output shows 30 columns with the dummy variables. And then, we are done with the feature selection part. Here, we took all variables for independent variables excluding the target Status variable. And, with the feature selection values, we dropped the categorical variables that have the 0.000 values as their feature score values to get the suitable proper variables. Finally, for the target variable, we converted the categorical values "Y" and "N" into the numerical values "1" and "0".

4.1.9 Dummy variables:

	Applicant_Income	Coapplicant_Income	Loan_Amount	Status	Gender_Female	Gend
0	0.570528	0.000000	0.561798	Y	0	
1	0.443788	0.268089	0.479401	N	0	
2	0.285314	0.000000	0.247191	Y	0	
3	0.243568	0.419200	0.449438	Y	0	
4	0.585644	0.000000	0.528090	Y	0	
...	...	...	...	...	...	
609	0.275303	0.000000	0.265918	Y	1	
610	0.396036	0.000000	0.149813	Y	0	
611	0.793072	0.042667	0.947566	Y	0	
612	0.744119	0.000000	0.700375	Y	0	
613	0.443788	0.000000	0.498127	N	1	

433 rows × 30 columns

4.1.10 Feature score value in descending order:

8	NaN	0.020335	Dependents_2
9	NaN	0.018644	Term_360.0
10	NaN	0.018527	Gender_Male
11	NaN	0.018419	Dependents_3+
12	NaN	0.015989	Area_Urban
13	NaN	0.012208	Married_No
14	NaN	0.009158	Term_240.0
15	NaN	0.009120	Gender_Female
16	NaN	0.008642	Term_180.0
17	NaN	0.007061	Education_Graduate
18	NaN	0.000000	Education_Not Graduate
19	NaN	0.000000	Dependents_1
20	NaN	0.000000	Term_60.0
21	NaN	0.000000	Term_84.0
22	NaN	0.000000	Term_120.0
23	NaN	0.000000	Term_300.0
24	NaN	0.000000	Dependents_0
25	NaN	0.000000	Married_Yes
26	NaN	0.000000	Credit_History_0.0
27	NaN	0.000000	Area_Rural
28	NaN	0.000000	Self_Employed_Yes

4.1.11 Dropping the variables that are having 0.000 values:

	Applicant_Income	Coapplicant_Income	Loan_Amount	Status	Gender_Female	Gender_Male	Mar
0	0.570528	0.000000	0.561798	Y	0	1	
1	0.443788	0.268089	0.479401	N	0	1	
2	0.285314	0.000000	0.247191	Y	0	1	
3	0.243568	0.419200	0.449438	Y	0	1	
4	0.585644	0.000000	0.528090	Y	0	1	
...	...	...	...	...	...	...	
609	0.275303	0.000000	0.265918	Y	1	0	
610	0.396036	0.000000	0.149813	Y	0	1	
611	0.793072	0.042667	0.947566	Y	0	1	
612	0.744119	0.000000	0.700375	Y	0	1	
613	0.443788	0.000000	0.498127	N	1	0	

433 rows × 19 columns

4.1.12 Converted the status categorical values as “0” and “1”:

	Applicant_Income	Coapplicant_Income	Loan_Amount	Status	Gender_Female	Gender_Male	Married_No	I
0	0.570528	0.000000	0.561798	1	0	1	1	
1	0.443788	0.268089	0.479401	0	0	1	0	
2	0.285314	0.000000	0.247191	1	0	1	0	
3	0.243568	0.419200	0.449438	1	0	1	0	
4	0.585644	0.000000	0.528090	1	0	1	1	
...	...	...	...	...	...	...	...	
609	0.275303	0.000000	0.265918	1	1	0	1	
610	0.396036	0.000000	0.149813	1	0	1	0	
611	0.793072	0.042667	0.947566	1	0	1	0	
612	0.744119	0.000000	0.700375	1	0	1	0	
613	0.443788	0.000000	0.498127	0	1	0	1	

433 rows × 19 columns

4.2 Finding and Interpretation

As we discussed before, to achieve our objectives we have used machine learning techniques. In this case we took three machine learning classification techniques to identify which models is fit

for make loan approval status prediction. At the very beginning, we split the dataset into training and testing. Here, status is the target variable and other variables are considered as independent variables. For train purpose we assigned 80% and for the test purpose it is 20%.

4.2.1 Split the dataset into training and testing:

```
from sklearn.model_selection import train_test_split
y=DataNew['Status']
x=DataNew.drop('Status',axis=1)
```

```
# spliting the data into training and testing sets to evaluvate the performance of the model

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=42)
```

4.2.2 Shape of the dataset after split

```
print(x_train.shape)
print(x_test.shape)
print(y_train.shape)
print(y_test.shape)
```

It represents the shape of the dataset after split into the training and test process.

```
(346, 18)
(87, 18)
(346,)
(87,)
```

Logistic regression is a machine learning technique used to analyze and predict the output. Also, it is used for binary dependent or target variables. Firstly, it is a must to fit the logistic regression model. And then we trained the model over the training data. In our project, we use min-max scaler scaling techniques to standardize the array of x-variables in the dataset.

To predict the loan approval status with logistic regression, it shows the accuracy score as 85%. The output is given below.

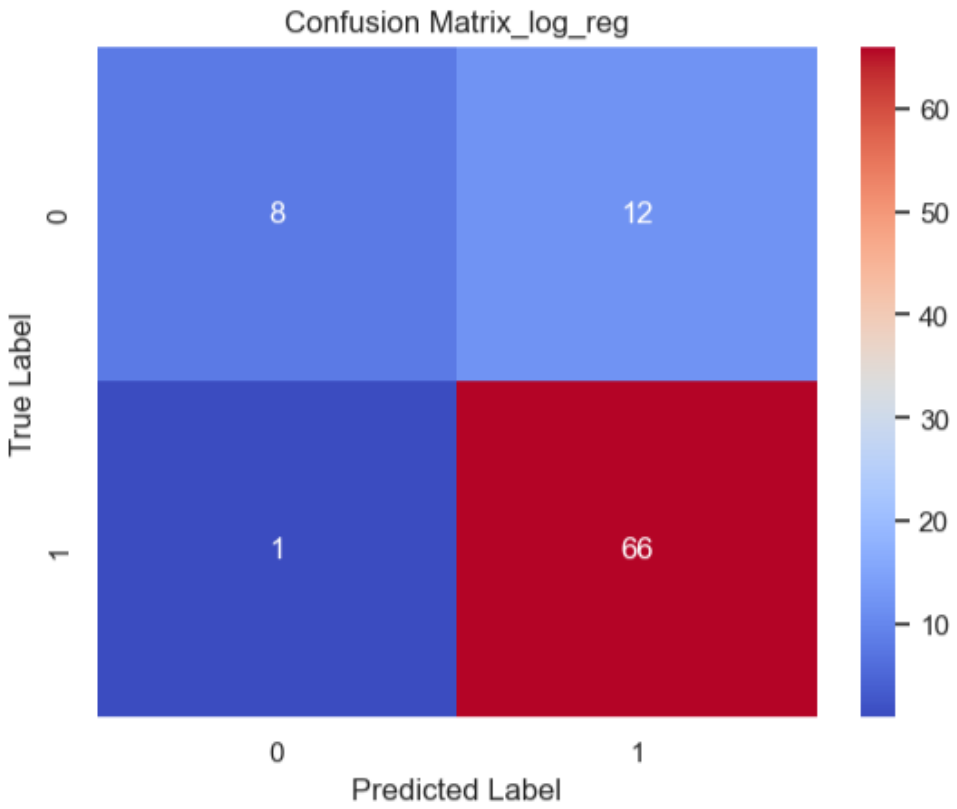
4.2.3 Classification report of logistic regression:

```
Accuracy_log_reg: 0.8505747126436781
Confusion Matrix_log_reg:
[[ 8 12]
 [ 1 66]]
Classification Report_log_reg:
              precision    recall  f1-score   support

           0       0.89      0.40      0.55        20
           1       0.85      0.99      0.91        67

    accuracy                           0.85        87
   macro avg       0.87      0.69      0.73        87
weighted avg       0.86      0.85      0.83        87
```

The accuracy score of logistic regression is around 85%. It represents that logistic regression is 85% accurate in predicting the loan approval status. The '0' and '1' in the above report stand for as follows: '0' – rejection of loan, '1' – approval of loan. In this case, the precision explains that 85% correctly predicted for class '1' and 89% positively predicted for class '0'. In the recall, it shows a 0.40 positive score for class '0' and 0.99 for class '1'. The f-1 score represents the mean score of precision and recall. It shows 0.55 for class '0' and 0.91 for class '1'. And to identify the actual existence of the '0' and '1' classes, support is calculated in the report. According to the classification report, it explains that the quantity of the 0 class is 20, and 1 is 67. On the other hand, there are some other values also presented in the report. They are macro average and weighted average. With the macro average, we can calculate the overall average value for the both 0 and 1 classes. And with the weighted average we can identify the average of both classes while it is having different count of samples. The above report mentioned the macro and weighted averages for all precision, recall, and f-1 scores for both classes.

Overall, the logistic regression model shows 85% accuracy, and high values in precision, recall, and f-1 score for class ‘1’. Which means high for the loan approvals yes and low for the loan approval no status.

4.2.4 Confusion matrix for logistic regression:



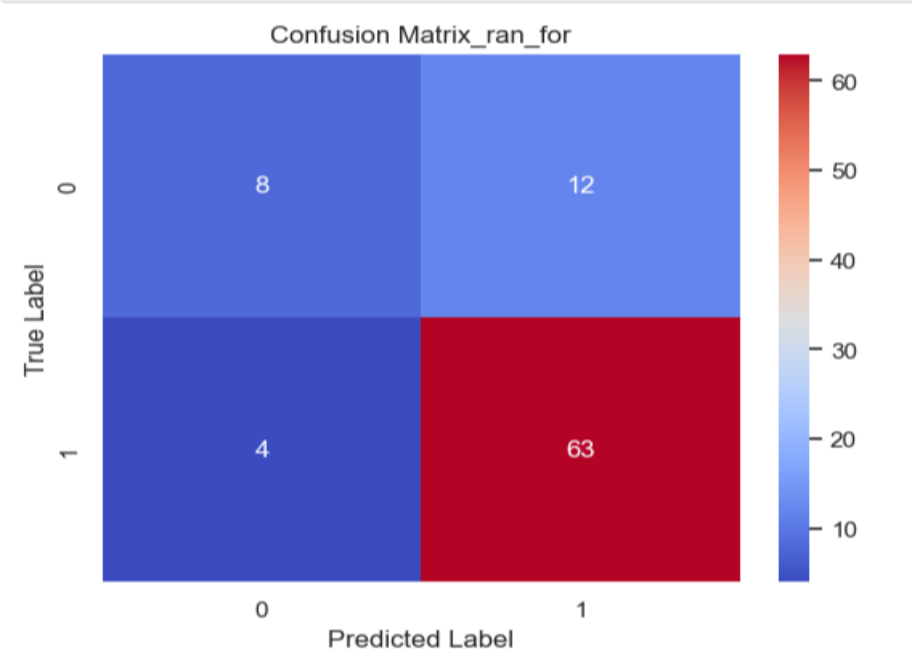
With the help of a confusion matrix, we can analyze the correct prediction and incorrect prediction. According to the above confusion matrix, it represents 66 as true positive (TP) which means for the class ‘1’ it correctly predicted the 66 instances. 8 as true negative (TN). It stands for, that the model correctly predicted the 8 instances over the class ‘0’. On the other hand, it clearly explains that the 12 instances correctly predicted for class ‘1’ for false positive (FP), and false negative (FN) for class ‘0’ it correctly predicted only one instance. Also, the AUC score of logistic regression is 69% whereas the accuracy score is 85%.

4.2.5 Classification report of the random forest:

```
Accuracy_ran_for: 0.8160919540229885
Confusion Matrix_ran_for:
[[ 8 12]
 [ 4 63]]
Classification Report_ran_for:
              precision    recall  f1-score   support

           0       0.67      0.40      0.50        20
           1       0.84      0.94      0.89        67

    accuracy                           0.82        87
   macro avg       0.75      0.67      0.69        87
weighted avg       0.80      0.82      0.80        87
```

The accuracy score for the random forest machine learning technique is around 81.6%. This means, that to predict the loan approval status, a random forest of 81.6% correctly predicts the status. It is something below the logistic regression method. Also, the above report gives insights about the all precision, recall, f1-score, and support values.

4.2.6 Confusion matrix of random forest:



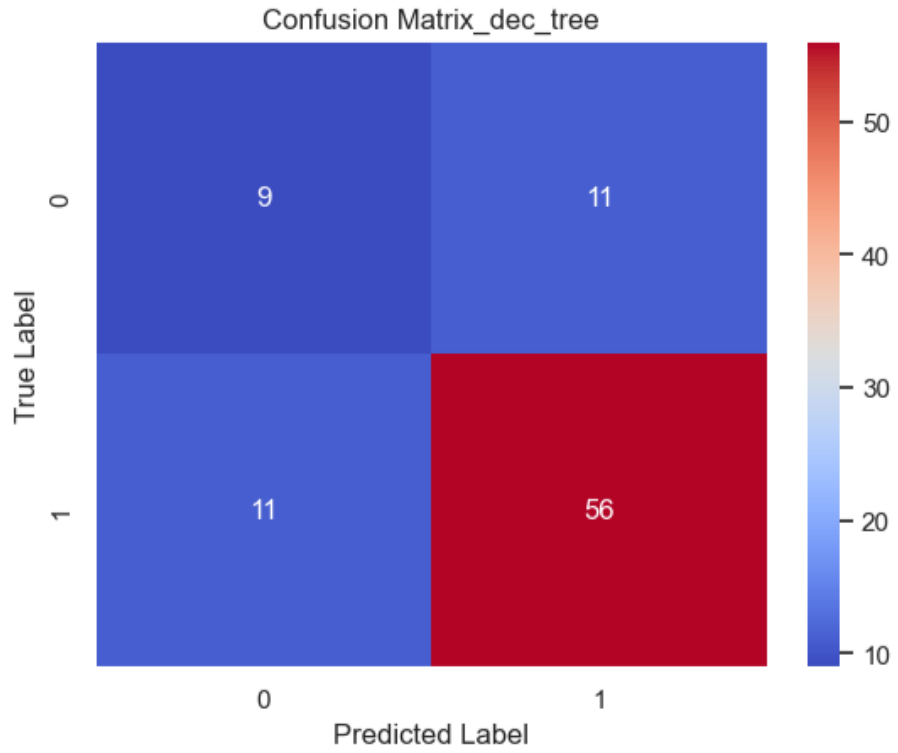
For the loan approval status yes, Random Forest correctly predicted 66 as true positive, and for the loan approval status no, it predicted only 8 as true negative. On the other hand, it correctly predicted 12 for false positive whereas 4 for false negative. And the AUC score is 67%.

4.2.7 Classification report of Decision tree

```
Accuracy_dec_tree: 0.7471264367816092
Confusion Matrix_dec_tree:
[[ 9 11]
 [11 56]]
Classification Report_dec_tree:
              precision    recall  f1-score   support

           0       0.45      0.45      0.45        20
           1       0.84      0.84      0.84        67

    accuracy                           0.75        87
   macro avg       0.64      0.64      0.64        87
weighted avg       0.75      0.75      0.75        87
```

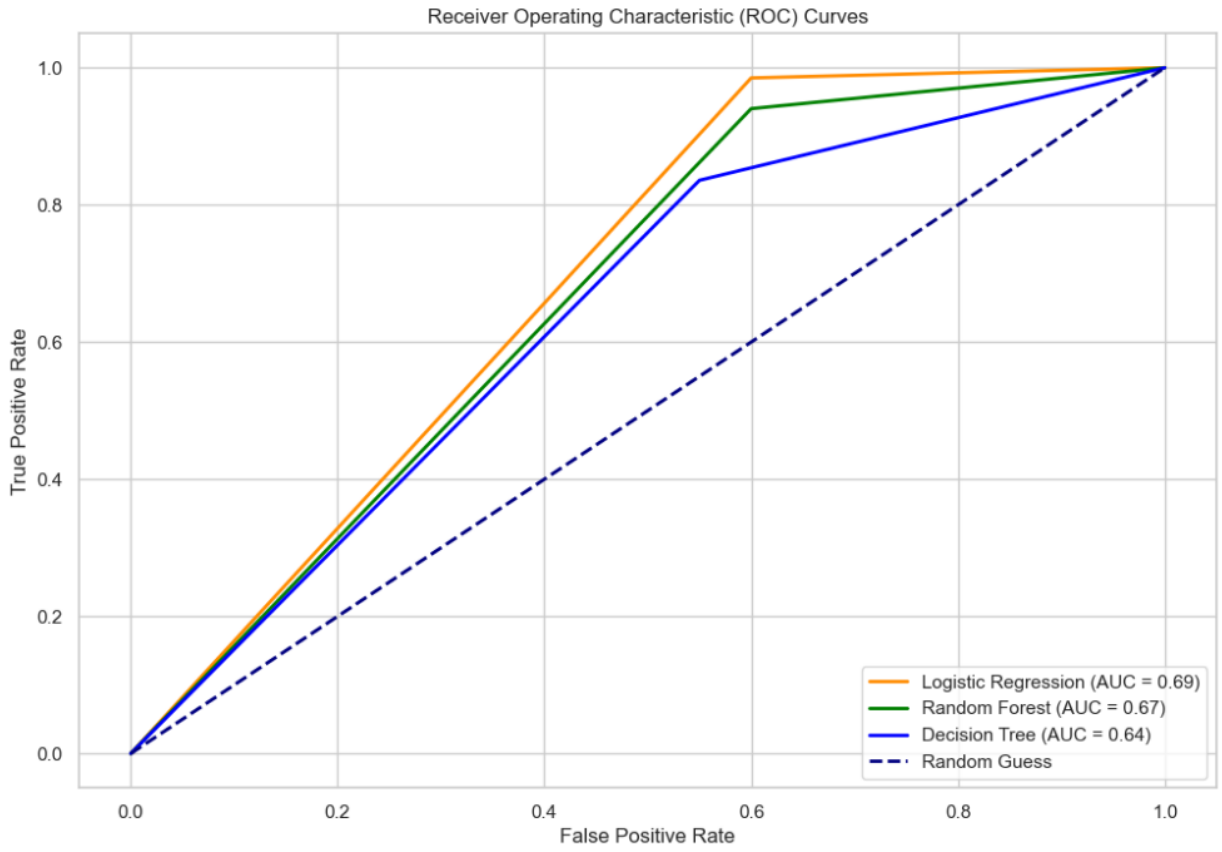
The accuracy score for the decision tree classification model is around 74.7%. It represents that the decision tree model is 74.7% and accurately predicts the loan approval status. The accuracy score of the decision tree is lower than both logistic regression and random forest. On the other hand, the precision, recall, and f1-scores also showing lower scores than the other two models.

4.2.8 Confusion matrix of Decision Tree:



According to the above confusion matrix, it explains the correct and incorrect prediction of the model. 56 for TP, 9 for TN, 11 for FP and FN for the both 0 and 1 classes. The AUC score of this model is around 64.4%.

4.2.9 ROC Curves.



The above ROC curves represent the values between the false positive rate and true positive rate for the models that we have done above. According to the curves, all curves of the models are above the random line. It represents a good model fit. According to all accuracy scores and AUC scores of the logistic regression, random forest, and decision tree, logistic regression is the best model to predict the loan approval status.

To achieve our second support objective, we analyzed it with a linear regression model. For the linear regression model, we should get numerical variables for our target variable. In this case, we took the loan amount as our target variable. In this case, it shows 7.4% as the model score for

training and -16.7% for testing. These scores represent the model performance by training and testing. It is not a good model for the data. Because it's showing a weak and negative fit. On the other hand, the r-squared value is showing as 40.9%. This means overall the variance of target variable from the x variables are not good fit. It will not give good prediction. The output is given below:

4.2.10 OLS regression results:

OLS Regression Results

Dep. Variable:	Loan_Amount	R-squared:	0.409
Model:	OLS	Adj. R-squared:	0.385
Method:	Least Squares	F-statistic:	16.89
Date:	Thu, 30 Nov 2023	Prob (F-statistic):	1.04e-37
Time:	13:15:35	Log-Likelihood:	206.89
No. Observations:	433	AIC:	-377.8
Df Residuals:	415	BIC:	-304.5
Df Model:	17		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
const	0.0673	0.036	1.857	0.064	-0.004	0.139
Applicant_Income	0.5810	0.043	13.423	0.000	0.496	0.666
Coapplicant_Income	0.3090	0.034	9.215	0.000	0.243	0.375
Status	-0.0086	0.020	-0.425	0.671	-0.048	0.031
Gender_Female	0.0377	0.022	1.692	0.091	-0.006	0.081
Gender_Male	0.0296	0.020	1.495	0.136	-0.009	0.069
Married_No	-0.0326	0.018	-1.817	0.070	-0.068	0.003
Dependents_2	0.0351	0.021	1.705	0.089	-0.005	0.076
Dependents_3+	0.0106	0.028	0.378	0.706	-0.045	0.066
Education_Graduate	0.0211	0.018	1.148	0.252	-0.015	0.057
Self_Employed_No	0.0174	0.024	0.733	0.464	-0.029	0.064
Term_36.0	0.1450	0.117	1.235	0.218	-0.086	0.376
Term_180.0	-0.0331	0.049	-0.672	0.502	-0.130	0.064
Term_240.0	0.0685	0.116	0.589	0.556	-0.160	0.297
Term_360.0	0.0531	0.041	1.299	0.195	-0.027	0.133
Term_480.0	0.0711	0.060	1.181	0.238	-0.047	0.189
Credit_History_1.0	-0.0114	0.025	-0.449	0.654	-0.061	0.039
Area_Semiurban	0.0201	0.019	1.080	0.281	-0.016	0.057
Area_Urban	-0.0225	0.019	-1.160	0.247	-0.061	0.016

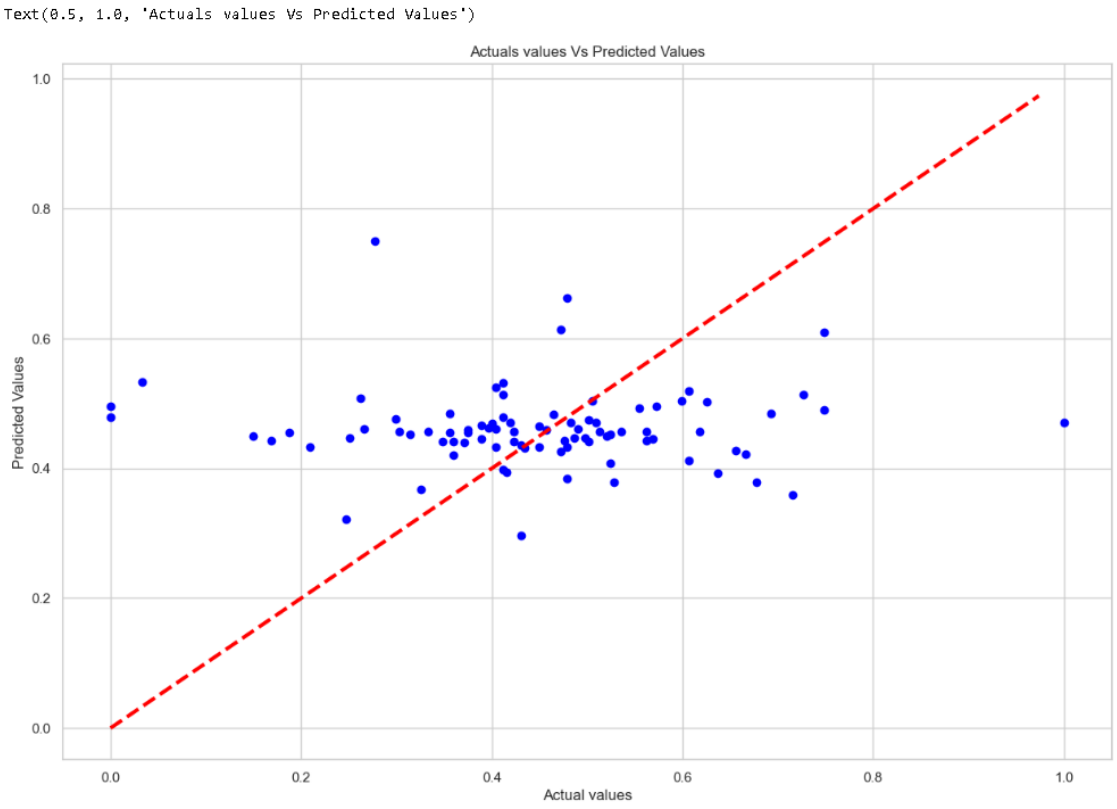
Omnibus:	79.450	Durbin-Watson:	2.128
Prob(Omnibus):	0.000	Jarque-Bera (JB):	224.442
Skew:	-0.867	Prob(JB):	1.83e-49
Kurtosis:	6.071	Cond. No.	8.08e+15

According to the p-values, we can conclude that the applicant's income and co-applicant income are the variables that are statistically significant for predicting loan amount. Because these variables are having lower p-values than 0.05. This means, that when the applicant income and co-applicant income are increase, the loan amount also will get increase.

Coefficients also give insights into the relationships between independent variables and target variables. In this case, also, applicant income and co-applicant income also show a good positive relationship with the target variable loan amount. Because these variables only indicated a positive and bit strong relationship between them. This means, that if a unit gets an increase in the applicant and co-applicant income, the loan amount is also expected to increase according to the coefficient score. The other variables are mostly not strong and also some are showing negative values.

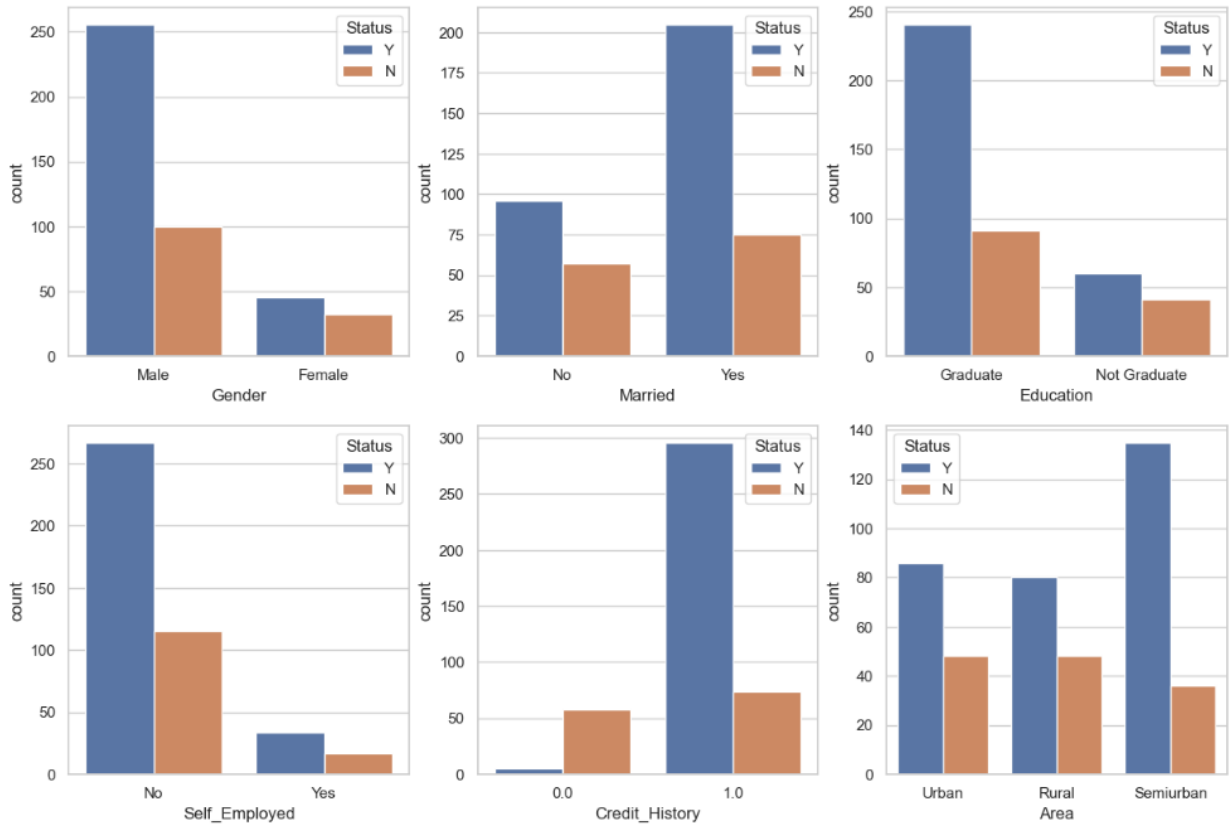
Overall, the linear regression model is not a good fit for the data. The regression line graph is given below. It shows actual values to predicted values.

4.2.11 Regression line:



In the last part, we have done with the graphical representations to achieve our last objective. This analysis is used to analyze the factors that are involved in the loan approval status and identify how they affect the loan approval status. The below bar chat gives insights about the main factors that how the banks or financial institutions making decision to give loan approvals.

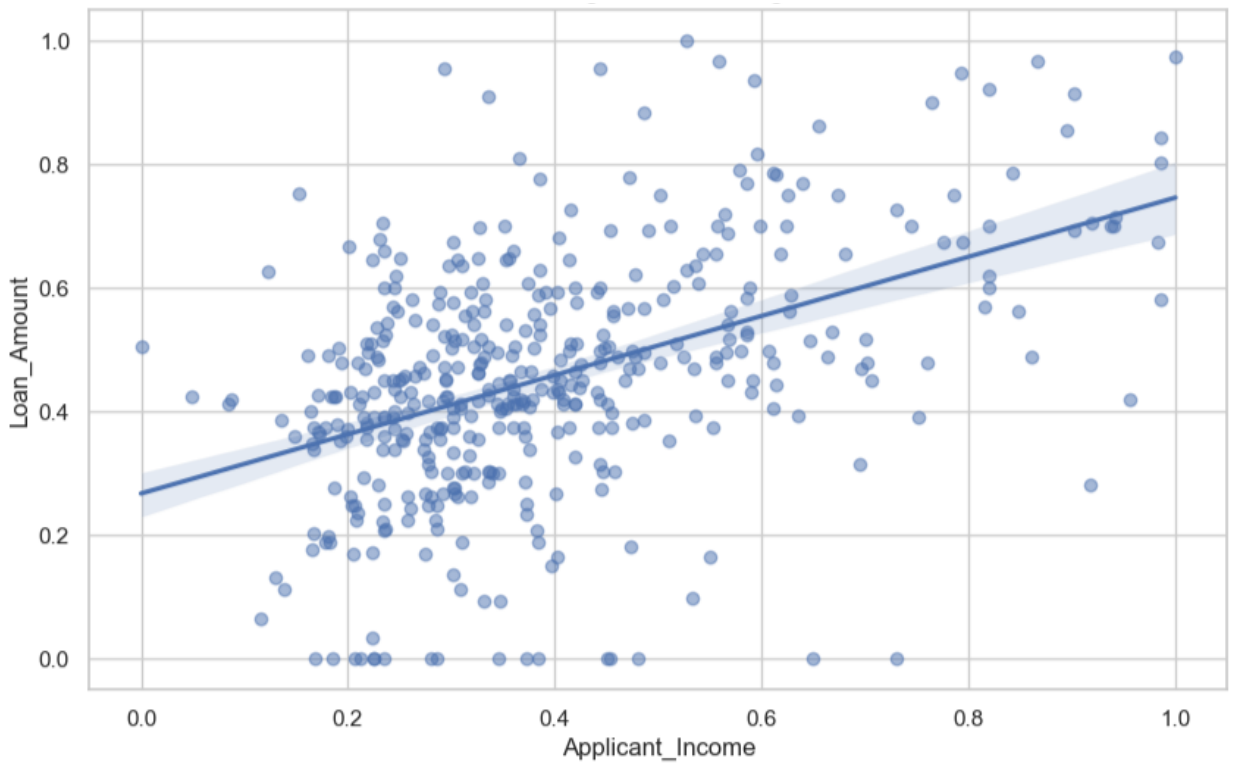
4.2.12 Bar chart for the involved variables:



According to the above charts banks or financial institutions are deciding to approve the loan. Mostly married applicants are selected for the loan approvals. When we consider the education qualification of the applicant, they make loan approval status as ‘yes’ for the graduate loan applicants. On the other hand, they mostly reject the loan selection for self-employed applicants. This is because they might think with their previous applicant's history that the self-employed have a high chance of getting losses in their business. If this happens, then applicants will face trouble

in repaying the loan amount. Also, they consider the applicant's residential area. They mostly provide loans for semiurban area applicants. And finally, they consider the applicant's past credit history. It represents whether the applicant repaid properly or not for the previous loan repayment process. In this case, if the applicant has any good past credit history, that means if the applicant repaid his loan properly in his past life, they will decide to provide loan for them.

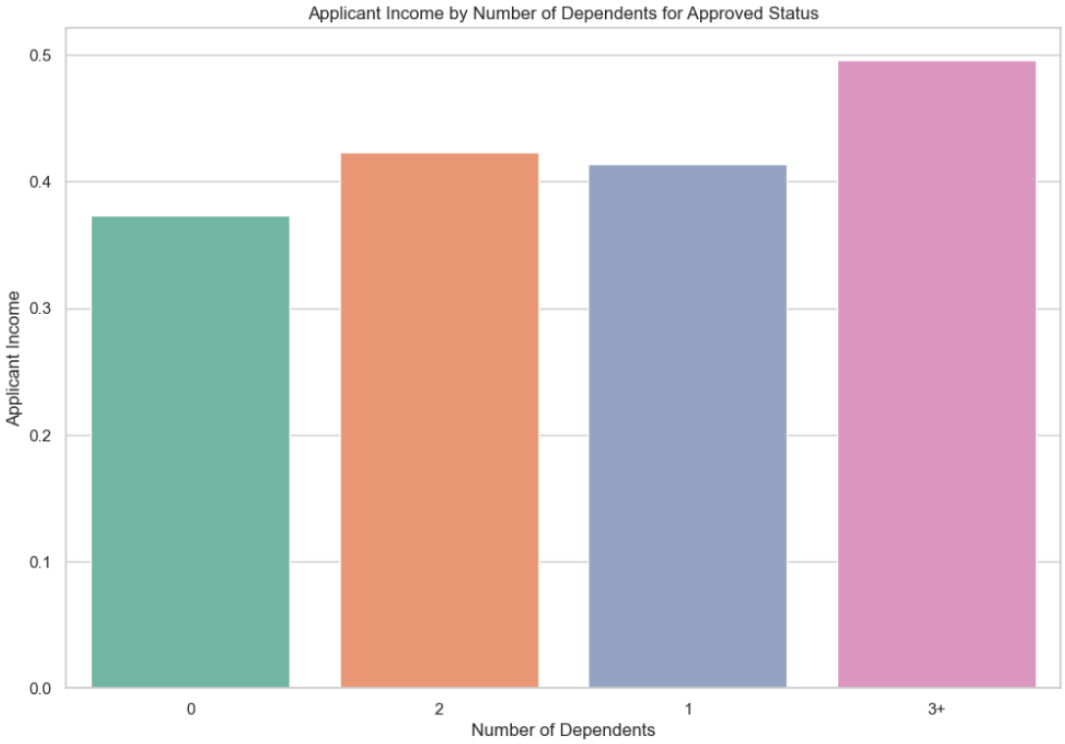
4.2.13 Regression line for applicant income vs loan amount



The above scatterplot with regression line is clearly explains that if the applicant income is high then the loan amount also will increase. In this case loan amount is strongly depends on applicant income.

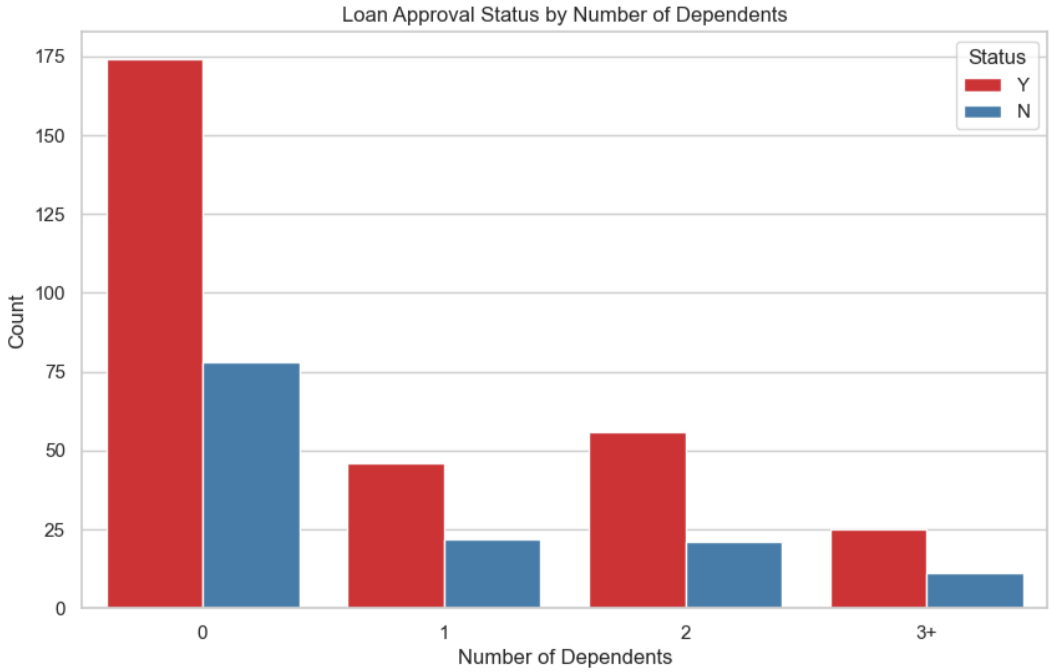
There is another main factor that affects the loan approval status is the dependents variable. Dependents is stands for the applicant's family members count. In the dataset, it shows most of the loan applicants are having 3+ (more than 3) dependents.

4.2.14 Dependents visualization:



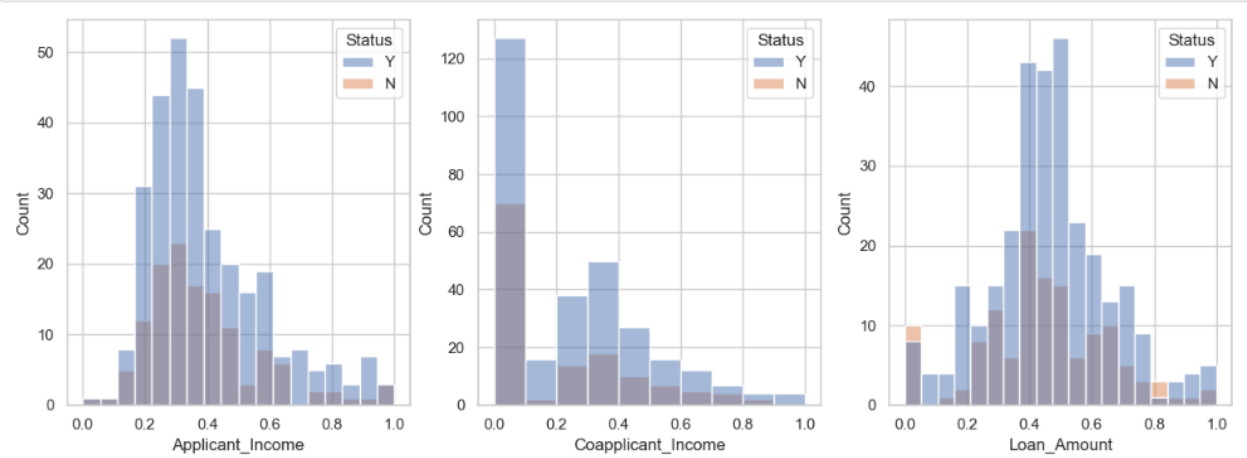
However, when they decide to give a loan, they will consider this as well. This means, that even if the loan applicant's income is high, because of the high dependents a big part of his income will mostly be used for their purposes. In this case, it will not be considered a good income. After deducting their expenses for their dependents, the applicant will only have a small part as his income. Therefore, this type of applicants will have high chance to face troubles while repay the loan. Hence, the banks more likely to give loan for '0' dependents. The output is given below:

4.2.15 Loan approval status by dependents:



The below histogram gives insights into the distribution of applicant income, co-applicant income, and loan amount. In this case, the distribution of the applicant's income is right-skewed. This means, there are only a few applicants have a high income. The distribution of the co-applicant income and loan amount also shows the same as the applicant income. Most of the applicants have a lower amount of the loan.

4.2.16 Histogram of the numerical variables.



5. Chapter 5: Conclusion

Based on the analysis we concluded that the logistic regression is the most suitable machine learning model to make predictions. We identified that the logistic regression has the highest accuracy score and the highest AUC (area under the curve) score than random forest and decision tree. This means, the logistic regression model mostly predicts accurately to analyze the loan approval status. On the other hand, analyze the loan amount variable with all other independent variables. It shows that the linear regression model is not a good model to analyze this. Because the r-squared values are lower. Also, it did not show the linear relationship between the variables. Applicant income and co-applicant income are the only variables that show statistical significance.

Finally, the graphical representation analysis, it strongly mentioned that the credit history, applicant's income, dependents, education and self employe variable are the main factors to make decision for loan approval status.

Appendices

```
# Importing necessary libraries
import warnings
warnings.filterwarnings("ignore")
import pandas as pd
import numpy as np
from sklearn.metrics import accuracy_score
from matplotlib import pyplot as plt
import seaborn as sns
```

```
# Loading the dataset

data = pd.read_csv(r"D:\HND.DS\ML\loan_train.csv")
data
```

```
# Summary of the data information

data.info()
```

```
# Descriptive statistics of the data

data.describe().style.background_gradient(cmap='plasma')
```

```
# Displaying the data types of each columns

data.dtypes
```

```
# Identifying the missing values in each column of the dataset

data.isnull().sum()
```

```
# Removing the missing values from the dataset

data = data.dropna()
data
```

```
# Analyzing unique values in the each columns

for column in data:
    unique_vals = np.unique(data[column])
    nr_values = len(unique_vals)
    if nr_values < 10:
        print('The number of values for feature {} :{} -- {}'.
              format(column,
        nr_values,unique_vals))
    else:
        print('The number of values for feature {} :{}'.
              format(column, nr_values))
```

```
# Cnverting the datatypes

columns_to_convert = ['Term', 'Credit_History']

for column in columns_to_convert:
    data[column] = data[column].astype('category')
```

```
# Count the occurrences of each category in the "Status" variable

count_data = data['Status'].value_counts().reset_index()
count_data.columns = ['Status', 'count']

# Create a bar plot
sns.set(style="whitegrid")
plt.figure(figsize=(8, 6))

# Create the bar chart
sns.barplot(x="Status", y="count", data=count_data)

# Adding labels and title
plt.xlabel("Status")
plt.ylabel("Count")
plt.title("Bar Chart of 'Status' Variable")
plt.xticks(rotation=0)
plt.show()

print("count_data: ")
print(count_data)
```

```
# Boxplot Checking for the outliers

numerical_features = ['Applicant_Income', 'Coapplicant_Income', 'Loan_Amount']

plt.figure(figsize=(10, 6))
sns.boxplot(data=data[numerical_features])
plt.xticks(rotation=45)
plt.show()
```

```
# For the Loan_Amount variables first quartile (Q1) and the third quartile (Q3)

Q1=data.Loan_Amount.quantile(0.25)
Q3=data.Loan_Amount.quantile(0.75)
Q1,Q3
```

```
# Calculating the inter quartile rang (IQR)

IQR=Q3-Q1
IQR
```

```
#  For the Loan_Amount variable calculates the lower and upper limits for identifying potential outliers in a dataset

lower_limit = Q1 - 1.5*IQR
upper_limit = Q3 + 1.5*IQR
lower_limit,upper_limit
```

```
# Exclude the raws that have values considered as outliers

data=data[(data['Loan_Amount']<26925000.0)&
          (data['Loan_Amount']>-475000.0)&
          (data['Applicant_Income']<1023425.0)&
          (data['Applicant_Income']>-152375.0)&
          (data['Coapplicant_Income']<562875.0)&
          (data['Coapplicant_Income']>-337725.0)]
data
```

```
# distribution analysis

numerical_features = ['Applicant_Income', 'Coapplicant_Income', 'Loan_Amount']

fig, axes = plt.subplots(nrows=1, ncols=len(numerical_features), figsize=(15, 4))
fig.suptitle('Distribution of Numerical Features')

for i, feature in enumerate(numerical_features):
    sns.histplot(data[feature], kde=True, ax=axes[i])
    axes[i].set_title(feature)

plt.show()
```

```
# Findout the P-value

from scipy.stats import shapiro

stat, p_value = shapiro(data['Applicant_Income'])
print(f'Shapiro-Wilk Test for Applicant_Income: Statistic={stat}, p-value={p_value}')

stat, p_value = shapiro(data['Coapplicant_Income'])
print(f'Shapiro-Wilk Test for Coapplicant_Income: Statistic={stat}, p-value={p_value}')

stat, p_value = shapiro(data['Loan_Amount'])
print(f'Shapiro-Wilk Test for Loan_Amount: Statistic={stat}, p-value={p_value}')
```

Normalization

```
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()

numerical_columns = data.select_dtypes(include=np.number).columns
data[numerical_columns] = scaler.fit_transform(data[numerical_columns])
```

Assigning the dummy variables

```
# Creating dummy variables for categorical variables

dummy = pd.get_dummies(data,columns=['Gender', 'Married', 'Dependents', 'Education', 'Self_Employed','Term',
       'Credit_History', 'Area'],dtype=int)

dummy
```

```
# separate the target variable from the input features. The x DataFrame will contain all the columns except 'status'
# 'y' - represents what you want to predict (target variable)
# spliting

x = dummy.drop("Status", axis=1).values
y = dummy["Status"]
```

```
# Initialize an empty DataFrame
feature = pd.DataFrame(columns=['Status', 'Feature Importance Score'])


for i, column in enumerate(dummy.drop('Status', axis=1)):
    print('Importance of feature {}: {:.3f}'.format(column, dt.feature_importances_[i]))

    fi = pd.DataFrame({'Variable': [column], 'Feature Importance Score': [dt.feature_importances_[i]]})

# Append data to the DataFrame
    feature = feature.append(fi, ignore_index=True)

# Sort the data
feature = feature.sort_values('Feature Importance Score', ascending=False).reset_index(drop=True)
feature
```

```
columns_to_drop = [
'Education_Not Graduate',
'Dependents_1',
'Term_60.0',
'Term_84.0',
'Term_120.0',
'Term_300.0',
'Dependents_0',
'Married_Yes',
'Credit_History_0.0',
'Area_Rural',
'Self_Employed_Yes']
DataNew = dummy.drop(columns_to_drop,axis=1)
DataNew
```

```
# Converting categorical values Y and N in the Status column into numerical values 1 and 0.

Status_mapping = { 'Y': 1,
                   'N': 0
}

DataNew['Status'] = DataNew['Status'].map(Status_mapping)
DataNew
```

```
# spliting the data into training and testing sets to eveluvate the performance of the model

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=42)
```

```
# After spliting, findout the shape of the resulting dataset

x_train, x_test, y_train, y_test = train_test_split(x, y, train_size=0.80, test_size=0.2, random_state=15)

print(x_train.shape)
print(x_test.shape)
print(y_train.shape)
print(y_test.shape)
```

```
from sklearn.linear_model import LogisticRegression

# Create the logistic regression model

logistic_regression_model = LogisticRegression()
logistic_regression_model.fit(x_train, y_train)
```

```
# Training the logistic regression model to make prediction

model=LogisticRegression()
model.fit(x_train, y_train)
y_predict_log_reg = model.predict(x_test)
y_predict_log_reg
```

```
#  feature scaling, specifically Min-Max scaling, on the input features and then make predictions with logistic regression

scaler= MinMaxScaler()
x_train_scale = scaler.fit_transform(x_train)
x_testscale = scaler.transform(x_test)
regression=model.fit(x_train_scale,y_train)
pred_logis_reg = regression.predict(x_testscale)
print(pred_logis_reg)
```

```
# evaluating the performance of a logistic regression model on a test dataset and generating several key evaluation metrics.

y_predict_log_reg = logistic_regression_model.predict(x_test)

from sklearn.metrics import accuracy_score, confusion_matrix, classification_report

accuracy = accuracy_score(y_test, y_predict_log_reg)
confusion = confusion_matrix(y_test, y_predict_log_reg)
classification_report = classification_report(y_test, y_predict_log_reg)

print(f'Accuracy_log_reg: {accuracy}')
print(f'Confusion Matrix_log_reg:\n{confusion}')
print(f'Classification Report_log_reg:\n{classification_report}')
```

```
# confusion matrix heatmap visualization

sns.heatmap(confusion, annot=True, fmt="d" , cmap='coolwarm')
plt.title("Confusion Matrix_log_reg")
plt.xlabel("Predicted Label")
plt.ylabel("True Label")
plt.show()
```

```
fpr, tpr, thresholds = roc_curve(y_test, pred_logis_reg)
roc_auc = auc(fpr, tpr)

# Plot ROC curve
plt.figure(figsize=(8, 6))
plt.plot(fpr, tpr, color='darkorange', lw=2, label=f'ROC curve (AUC = {roc_auc:.2f})')
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--', label='Random Guess')

plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic (ROC) Curve_log_reg')
plt.legend(loc='lower right')
plt.show()
```

```
# Create and train the Random Forest model

from sklearn.ensemble import RandomForestClassifier

random_forest_model = RandomForestClassifier()
random_forest_model.fit(x_train, y_train)
```

```
# Creating the decision tree model
from sklearn.tree import DecisionTreeClassifier
decision_tree_model = DecisionTreeClassifier()
decision_tree_model.fit(x_train, y_train)
```

```
from sklearn.metrics import roc_curve, auc
import matplotlib.pyplot as plt

# Assigning y_predict_log_reg, y_predict_ran_for, y_predict_dec_tree are your predicted probabilities
fpr_log_reg, tpr_log_reg, _ = roc_curve(y_test, y_predict_log_reg)
roc_auc_log_reg = auc(fpr_log_reg, tpr_log_reg)

fpr_ran_for, tpr_ran_for, _ = roc_curve(y_test, y_predict_ran_for)
roc_auc_ran_for = auc(fpr_ran_for, tpr_ran_for)

fpr_dec_tree, tpr_dec_tree, _ = roc_curve(y_test, y_predict_dec_tree)
roc_auc_dec_tree = auc(fpr_dec_tree, tpr_dec_tree)

# Plot ROC curves
plt.figure(figsize=(12, 8))

# Logistic Regression ROC Curve
plt.plot(fpr_log_reg, tpr_log_reg, color='darkorange', lw=2, label=f'Logistic Regression (AUC = {roc_auc_log_reg:.2f})')

# Random Forest ROC Curve
plt.plot(fpr_ran_for, tpr_ran_for, color='green', lw=2, label=f'Random Forest (AUC = {roc_auc_ran_for:.2f})')

# Decision Tree ROC Curve
plt.plot(fpr_dec_tree, tpr_dec_tree, color='blue', lw=2, label=f'Decision Tree (AUC = {roc_auc_dec_tree:.2f})')

# Random Line
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--', label='Random Guess')

plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic (ROC) Curves')
plt.legend(loc='lower right')
plt.show()
```

```
x = sm.add_constant(x)
model = sm.OLS(y, x)
results = model.fit()
print(results.summary())
```

```
mse = mean_squared_error(ytest, ypredic)
rmse = np.sqrt(mse)
print(f'Root Mean Squared Error: {rmse}')
```

```
# Plotting Actuals Vs Predicted

plt.figure(figsize=(15,10))

plt.scatter(ytest, ypredic, c='blue')
plt.plot([ytrain.min(), ytrain.max()], [ytrain.min(),
                ytrain.max()], 'k--', c='red', lw=3)
plt.xlabel('Actual values')
plt.ylabel('Predicted Values')
plt.title('Actuals values Vs Predicted Values')
```

```
# Bar Plot for Categorical Variables:


categorical_vars = ['Gender', 'Married', 'Education', 'Self_Employed', 'Credit_History', 'Area']

fig, axes = plt.subplots(nrows=2, ncols=3, figsize=(15, 10))

for i, var in enumerate(categorical_vars):
    sns.countplot(x=var, hue='Status', data=data, ax=axes[i//3, i%3])

plt.show()
```

```
# Pair Plot for Multiple Variables:

sns.pairplot(data, hue='Status', diag_kind='kde')
plt.show()
```

```
# Create a scatter plot with a regression line
plt.figure(figsize=(10, 6))
sns.regplot(x=data['Applicant_Income'], y=data['Loan_Amount'], scatter_kws={'alpha':0.5})
plt.xlabel('Applicant_Income')
plt.ylabel('Loan_Amount')
plt.title('Scatter Plot with Regression Line: applicant income vs. Loan amount')
plt.grid(True)
plt.show()
```

```
sns.set(style="whitegrid")
plt.figure(figsize=(10, 6))

# Box plot for Status vs Applicant Income
sns.boxplot(x="Status", y="Applicant_Income", data=data, palette="Set2", showfliers=False)

plt.xlabel("Loan Approval Status")
plt.ylabel("Applicant Income")
plt.title("Applicant Income Distribution by Loan Approval Status")

plt.show()
```

```
sns.set(style="whitegrid")
plt.figure(figsize=(12, 8))

# Filter data for approved status ('Y')
approved_data = data[data['Status'] == 'Y']

# Bar chart for Applicant Income by Dependents for 'Y' status
sns.barplot(x="Dependents", y="Applicant_Income", data=approved_data, palette="Set2", ci=None)

plt.xlabel("Number of Dependents")
plt.ylabel("Applicant Income")
plt.title("Applicant Income by Number of Dependents for Approved Status")

plt.show()
```

```
sns.set(style="whitegrid")
plt.figure(figsize=(10, 6))

# Bar chart for Status by Dependents
sns.countplot(x="Dependents", hue="Status", data=data, palette="Set1")

plt.xlabel("Number of Dependents")
plt.ylabel("Count")
plt.title("Loan Approval Status by Number of Dependents")

plt.show()
```

References:

1. Anon, n.d. [online] Accurate loan approval prediction based on machine learning approach. Available at: <https://jespublication.com/upload/2020-110471.pdf> [Accessed 30 Nov. 2023a].

2. Anon, n.d. [online] Prediction for loan approval using Machine Learning Algorithm - IRJET. Available at: <https://www.irjet.net/archives/V8/i4/IRJET-V8I4785.pdf> [Accessed 30 Nov. 2023a].

3. Anon, n.d. [online] Loan approval prediction model A comparative analysis - mililink.com. Available at: <https://www.mililink.com/upload/article/1759044670aams_vol_203_january_2020_a10_p427-435_afrah_khan_and_nidhi_singh.pdf> [Accessed 30 Nov. 2023a].

4. Anon, n.d. [online] Innovation-journals.org. Available at: <https://innovation-journals.org/JIIT/IT5-1/IV5i1-1.pdf> [Accessed 30 Nov. 2023].

5. Nureni, A.A., and Adekola, O.E., 2023. *Loan approval prediction based on machine learning approach*. [online] FUDMA JOURNAL OF SCIENCES. Available at: <https://fjs.fudutsinma.edu.ng/index.php/fjs/article/view/830> [Accessed 1 Dec. 2023].

6. Author links open overlay panelNazim Uddin a,, a,, b,, c,, Highlights•Developed an ensemble ML model for loan approval prediction outperforming individual ML and DL models.•Performed data balancing with SMOTE to enhance model performance.•Deployed the proposed ensemble model into a user-friendly desktop app for c, and AbstractBanks rely heavily on loans as a primary source of revenue; however, 2023. *An ensemble machine learning based bank loan approval predictions system with a smart application*. [online] International Journal of Cognitive Computing in Engineering. Available at: <https://www.sciencedirect.com/science/article/pii/S2666307423000293> [Accessed 1 Dec. 2023].

7. Mamun, M.A.,, Farjana, A., and Mamun, M., 2022. [online] Predicting Bank Loan Eligibility Using Machine Learning Model and Comparison Analysis . Available at: <https://ieomsociety.org/proceedings/2022orlando/328.pdf> [Accessed 1 Dec. 2023].

8. Kulothungan, et. al., n.d. *Loan forecast by using Machine Learning*. [online] Turkish Journal of Computer and Mathematics Education (TURCOMAT). Available at:

<https://www.turcomat.org/index.php/turkbilmat/article/view/2673> [Accessed 1 Dec. 2023].

9. Yamuna, and Praneeth, 2022. [online] An Approach to Loan Approval Prediction Using Machine Learning. Available at: <https://www.journal-dogorangsang.in/no_2_Online_22/14_aug.pdf> [Accessed 1 Dec. 2023].

10. Anon, n.d. [online] Loan approval prediction. Available at: <https://ijaem.net/issue_dcp/Loan%20Approval%20Prediction.pdf> [Accessed 30 Nov. 2023a].

11. Frost, J., 2023. *How to interpret p-values and coefficients in regression analysis*. [online] Statistics By Jim. Available at: <https://statisticsbyjim.com/regression/interpret-coefficients-p-values-regression> [Accessed 1 Dec. 2023].

12. Bhandari, P., 2023. *An easy introduction to statistical significance (with examples)*. [online] Scribbr. Available at: <https://www.scribbr.com/statistics/statistical-significance> [Accessed 1 Dec. 2023].

13. Zach, 2020. *Introduction to logistic regression*. [online] Statology. Available at: <https://www.statology.org/logistic-regression> [Accessed 1 Dec. 2023].

14. Anon, n.d. *Sklearn.metrics.accuracy_score*. [online] scikit. Available at: <https://scikit-learn.org/stable/modules/generated/sklearn.metrics.accuracy_score.html> [Accessed 1 Dec. 2023].

15. Sarker, I.H., 2021. *Machine learning: Algorithms, real-world applications and Research Directions*. [online] SN computer science. Available at: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7983091/> [Accessed 1 Dec. 2023].