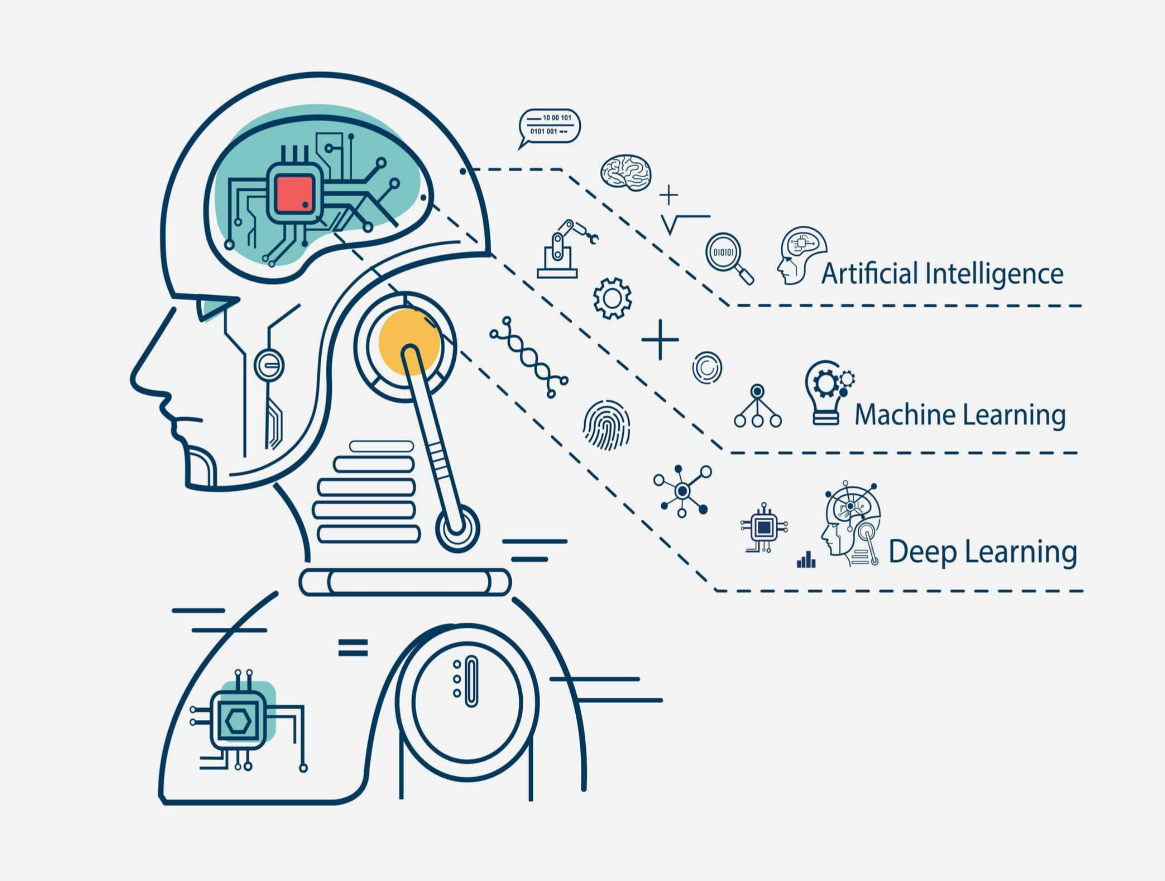
Loan Status Eligibility Prediction Using Machine Learning

– RISHI RAJ
(Analyst Task Report)



Problem Statement

- The objective of this project is to develop a machine learning model to predict loan approval status based on applicant details and financial information.

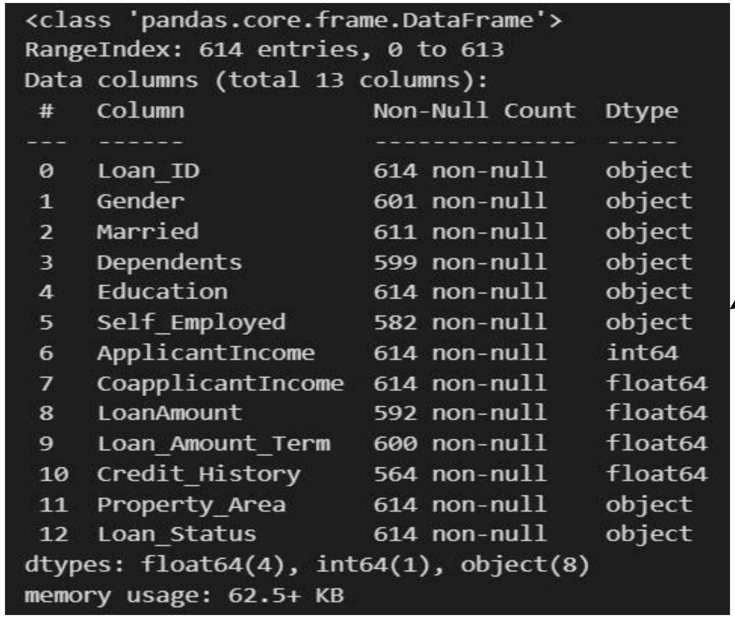
- The project involves preprocessing the data, performing exploratory data analysis, engineering features, selecting and evaluating classification models, and optimizing hyperparameters.

- The deliverables include a detailed report of the entire process, the trained predictive model with performance metrics, and a deployable version of the model.

Column Details

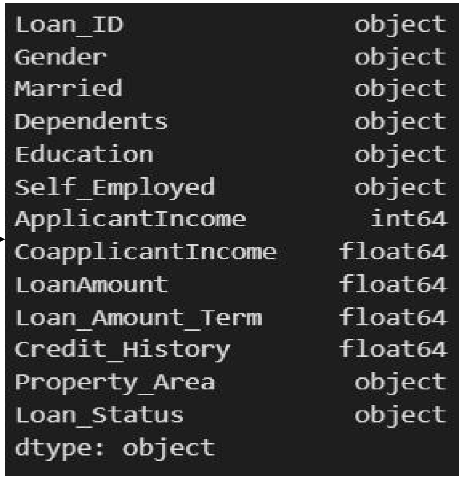
1. Loan_ID : Unique Loan ID
2. Gender : Male/ Female
3. Married : Applicant married (Y/N)
4. Dependents : Number of dependents
5. Education : Applicant Education (Graduate/ Under Graduate)
6. Self_Employed : Self employed (Y/N)
7. ApplicantIncome : Applicant income
8. CoapplicantIncome : Coapplicant income
9. LoanAmount : Loan amount in thousands of dollars
10. Loan_Amount_Term : Term of loan in months
11. Credit_History : Credit history meets guidelines yes or no
12. Property_Area : Urban/ Semi Urban/ Rural
13. Loan_Status : Loan approved (Y/N) this is the target variable

About Datasets



```
<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
```

data.info()



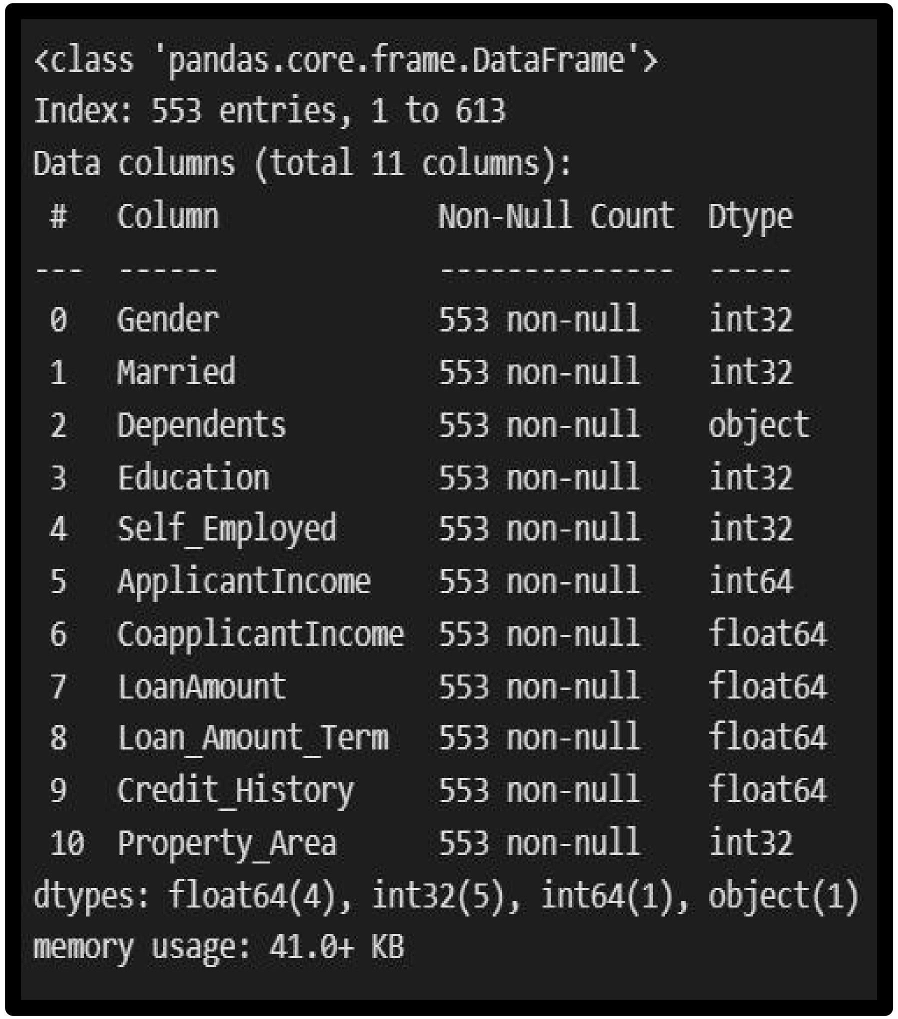
```
Loan_ID               object
Gender                object
Married               object
Dependents            object
Education             object
Self_Employed         object
ApplicantIncome        int64
CoapplicantIncome    float64
LoanAmount           float64
Loan_Amount_Term     float64
Credit_History       float64
Property_Area         object
Loan_Status           object
dtype: object
```

data.dtypes

Workflow Overview

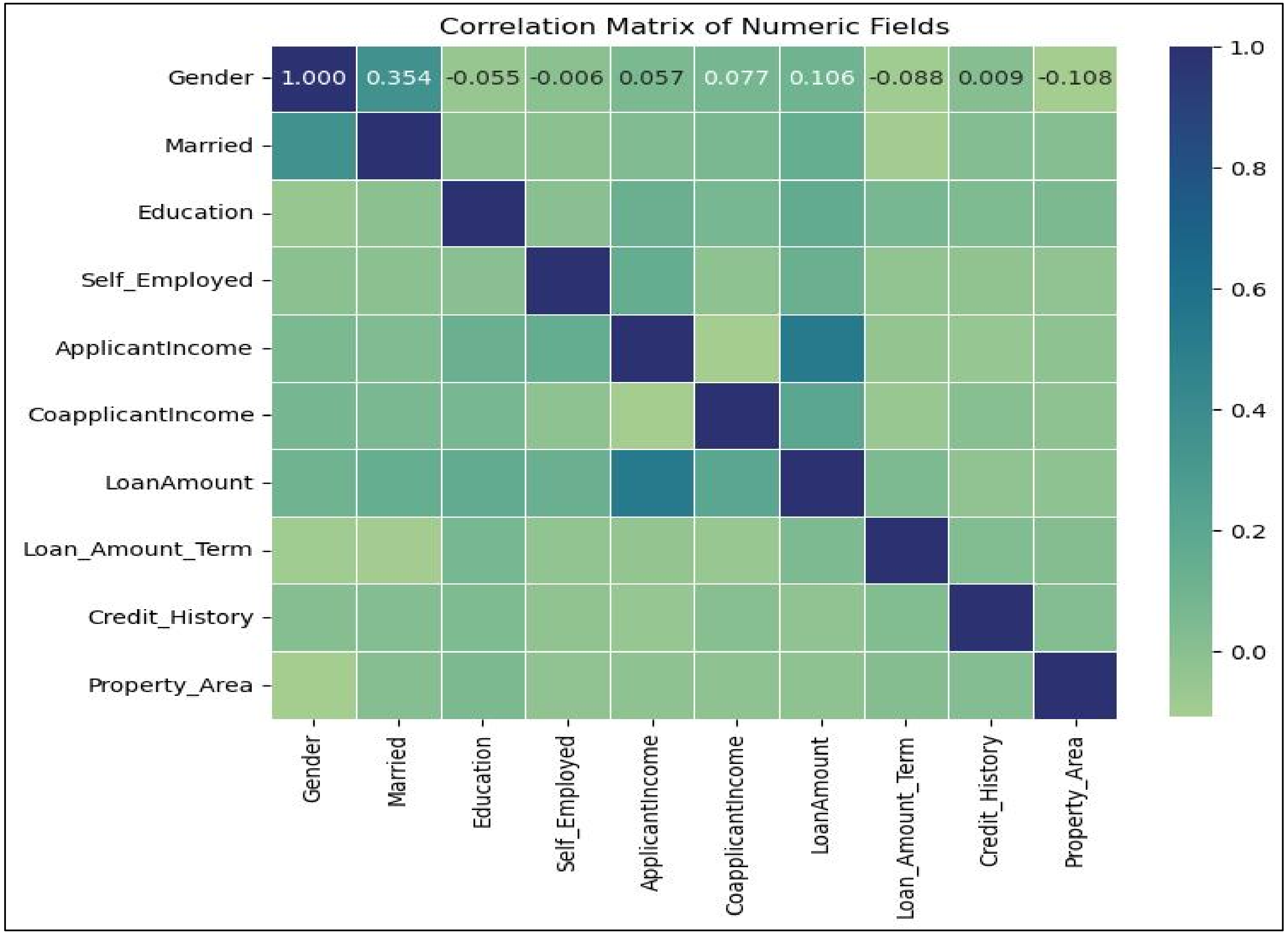
1. Data Reading
2. Data Exploration
3. Data Visualization and Analysis
4. Data Preparation and Data Scaling
5. Train Test Split of Data
6. Model Training
7. Model Prediction and Accuracy Metrics
8. Building a GUI Application

Read Data and Analyse



```
<class 'pandas.core.frame.DataFrame'>
Index: 553 entries, 1 to 613
Data columns (total 11 columns):
 #   Column             Non-Null Count  Dtype
---  ------             --------------  -----
 0   Gender             553 non-null    int32
 1   Married            553 non-null    int32
 2   Dependents         553 non-null    object
 3   Education          553 non-null    int32
 4   Self_Employed      553 non-null    int32
 5   ApplicantIncome    553 non-null    int64
 6   CoapplicantIncome  553 non-null    float64
 7   LoanAmount         553 non-null    float64
 8   Loan_Amount_Term   553 non-null    float64
 9   Credit_History     553 non-null    float64
 10  Property_Area      553 non-null    int32
dtypes: float64(4), int32(5), int64(1), object(1)
memory usage: 41.0+ KB
```

Dataset Info



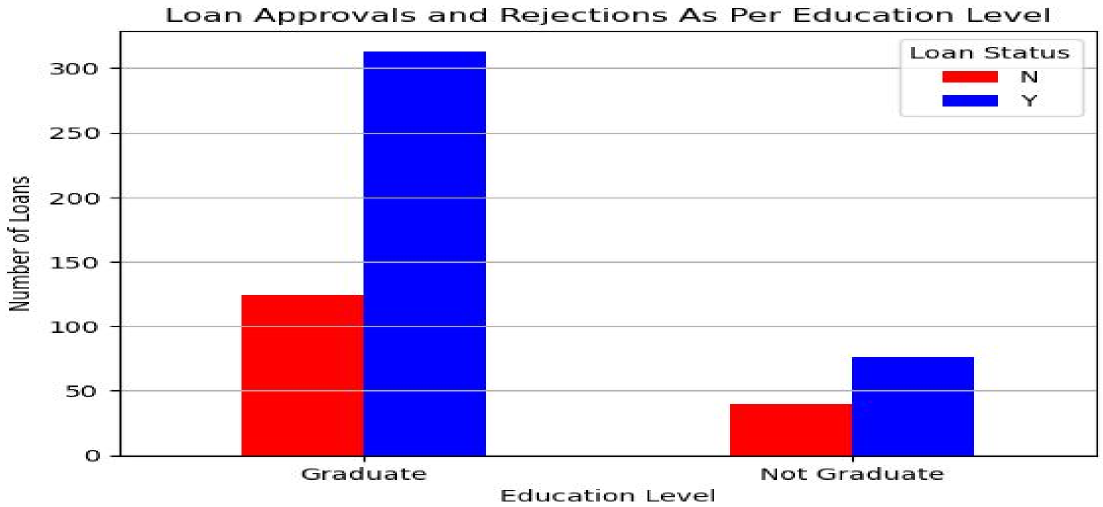
Correlation Matrix of Dataset Fields

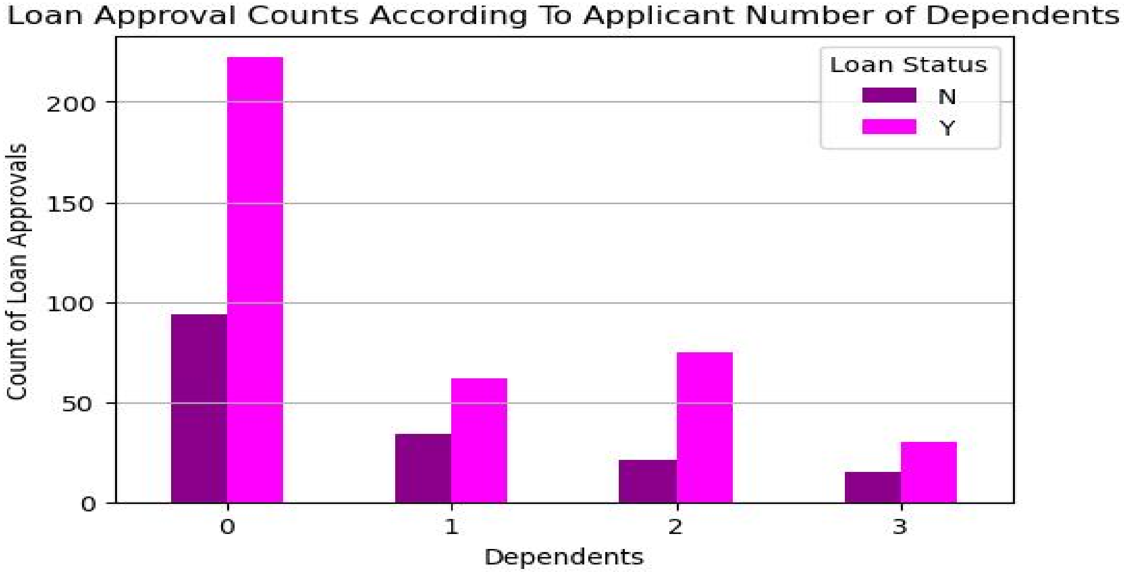
Data Exploration and Correction

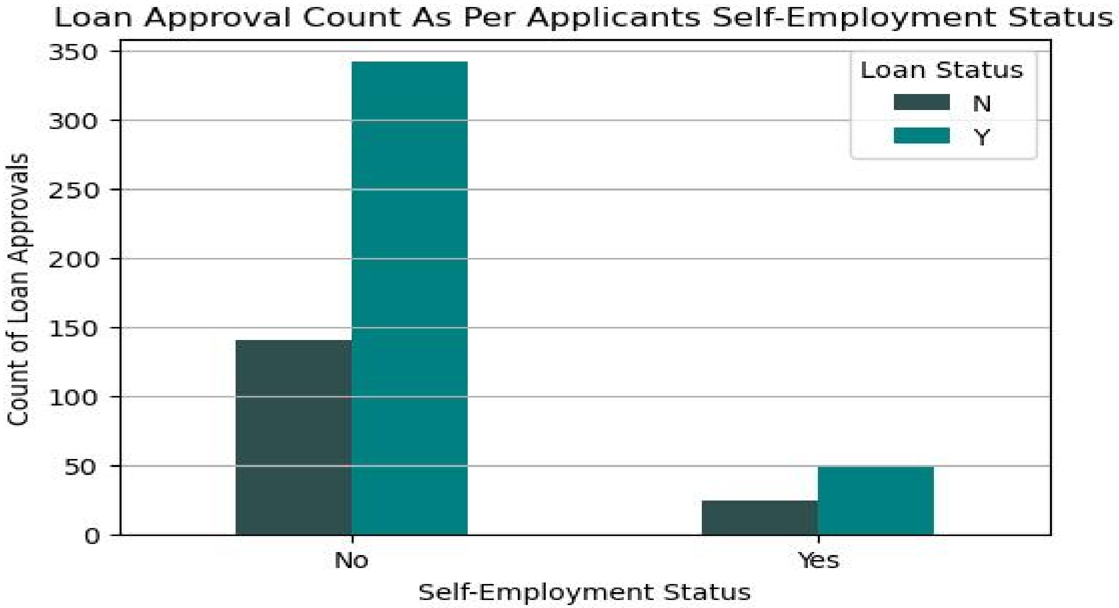
Important Steps:-

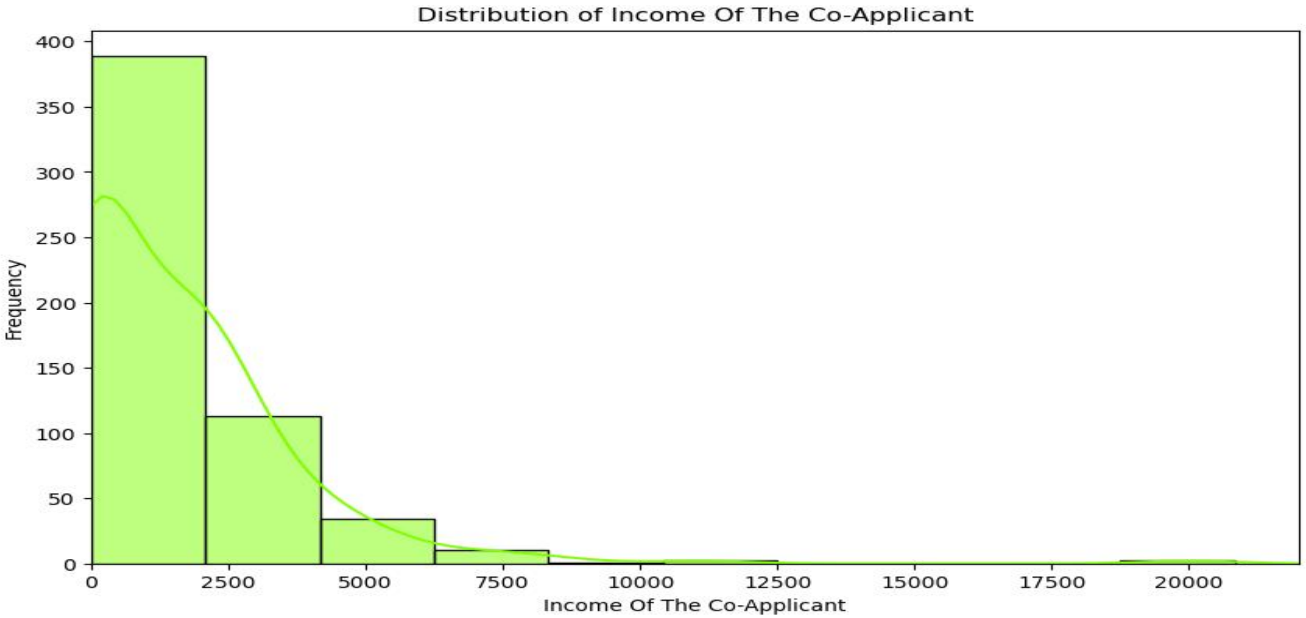
1. Checking Data Types of columns.
2. Checking for null values.
3. Correlation among data.
4. Getting descriptive statistics of the data.
5. Removing some negative values

Data Analysis: Plotting and Charting

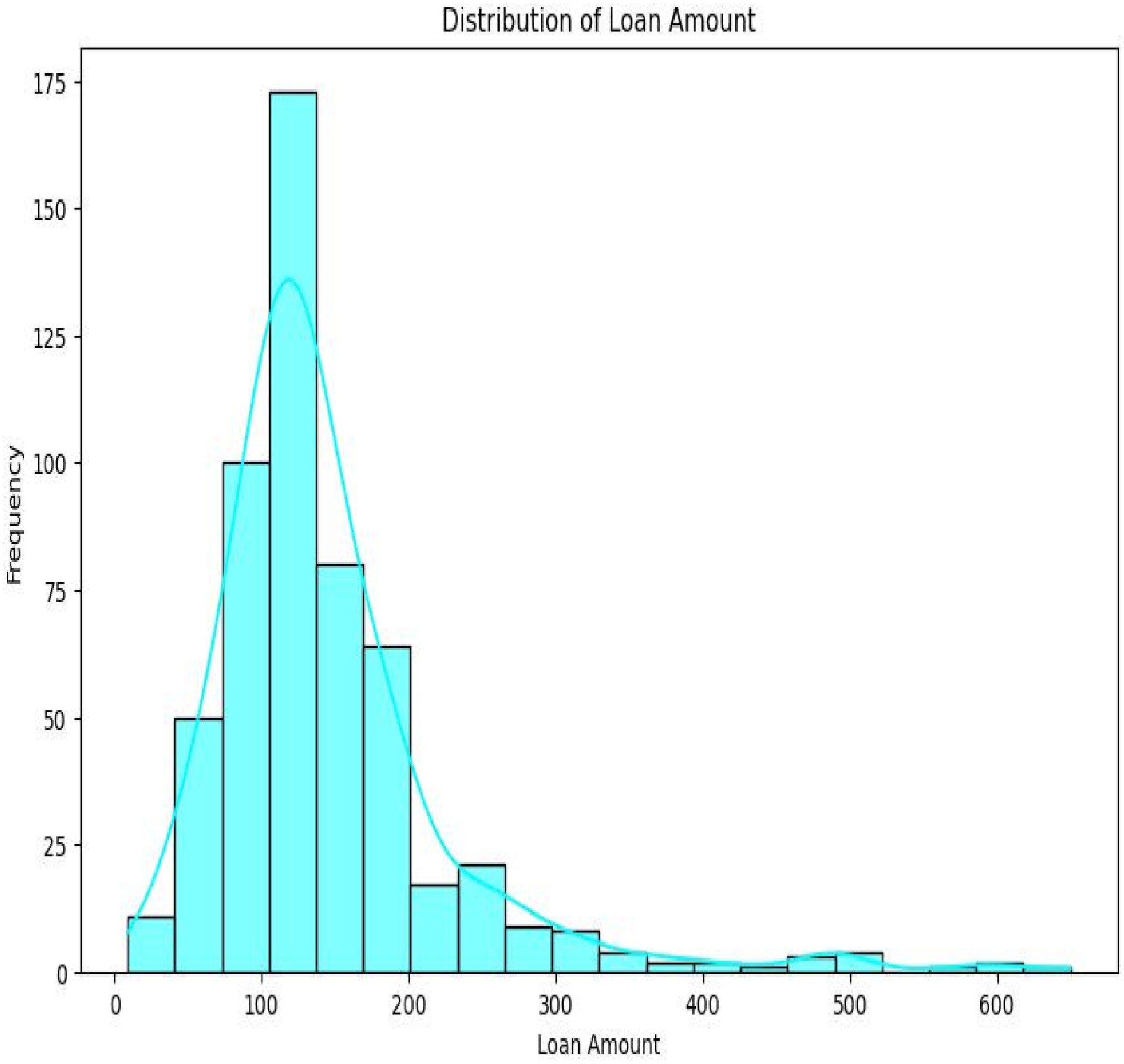

Loan Approvals and Rejections As Per Education Level

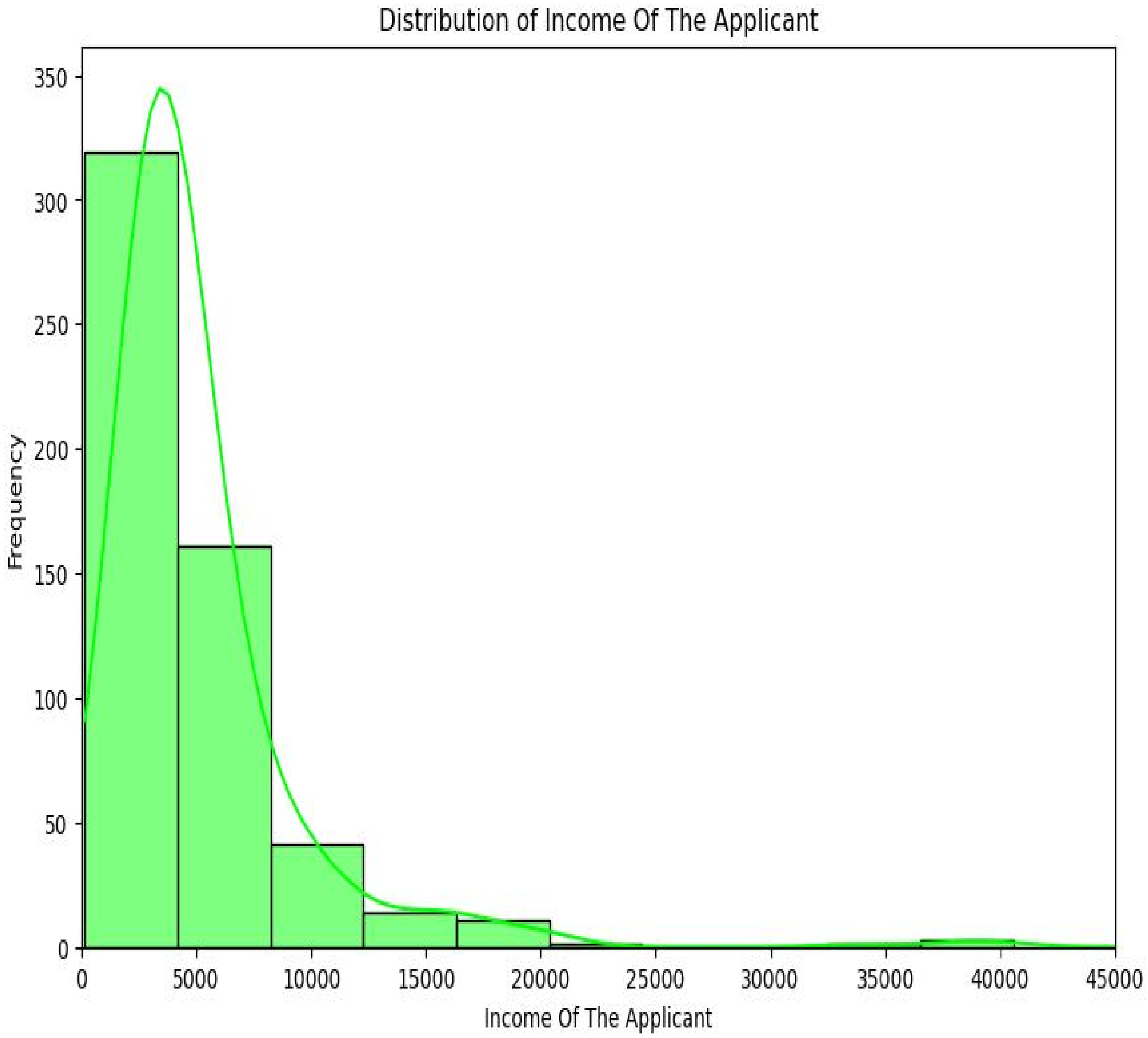

Loan Approval Counts According To Applicant Number of Dependents


Loan Approval Count As Per Applicants Self-Employment Status

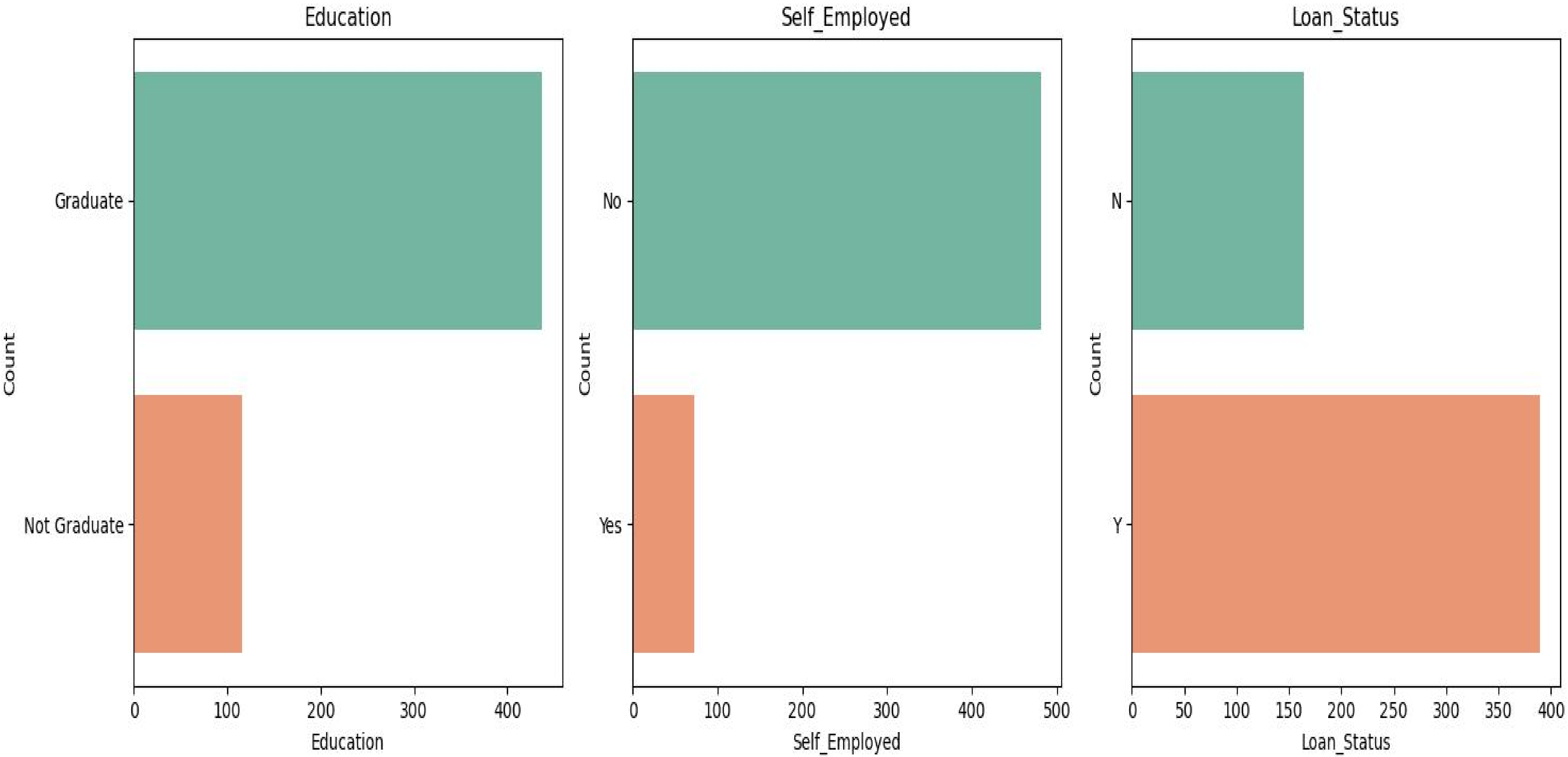

Distribution of Income Of The Co-Applicant

Data Analysis: Plotting and Charting

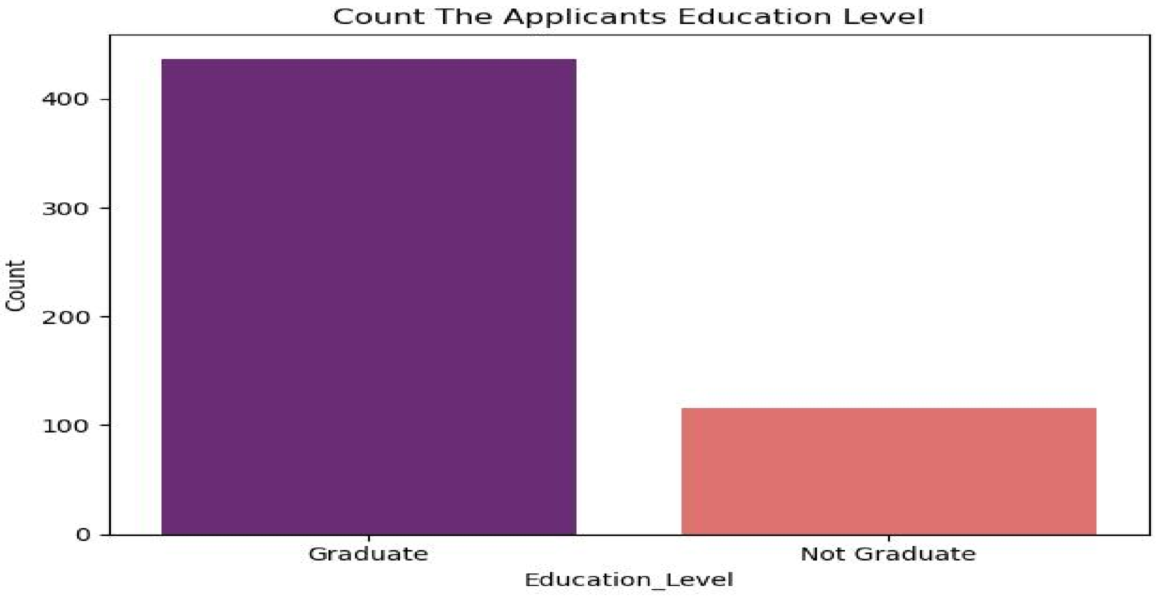


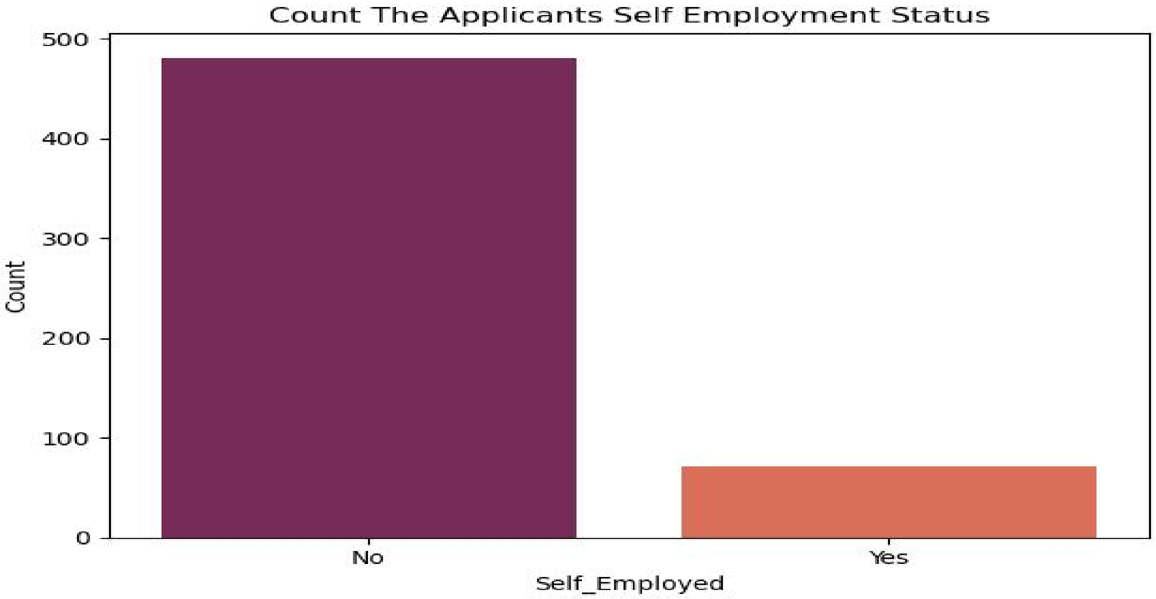


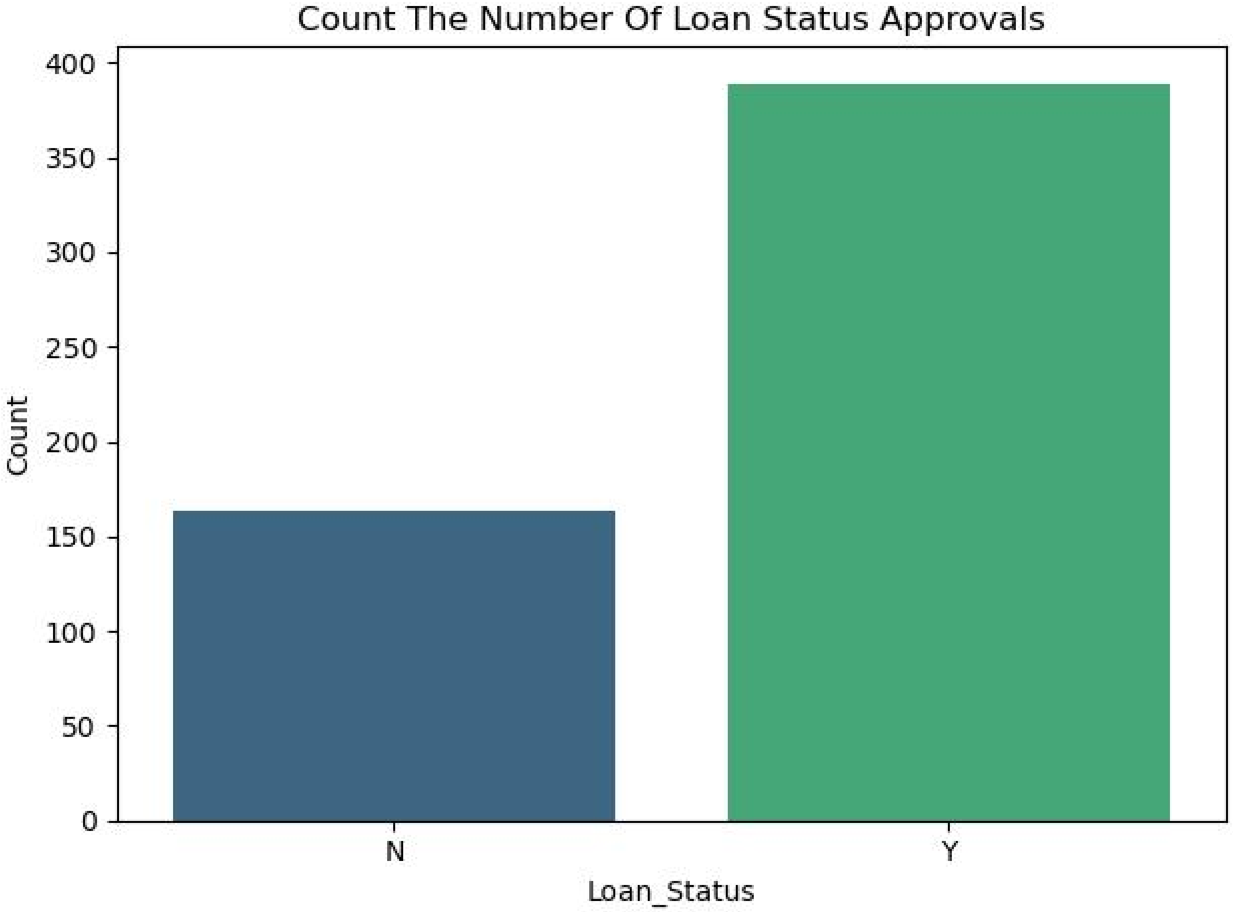
Data Analysis: Plotting and Charting



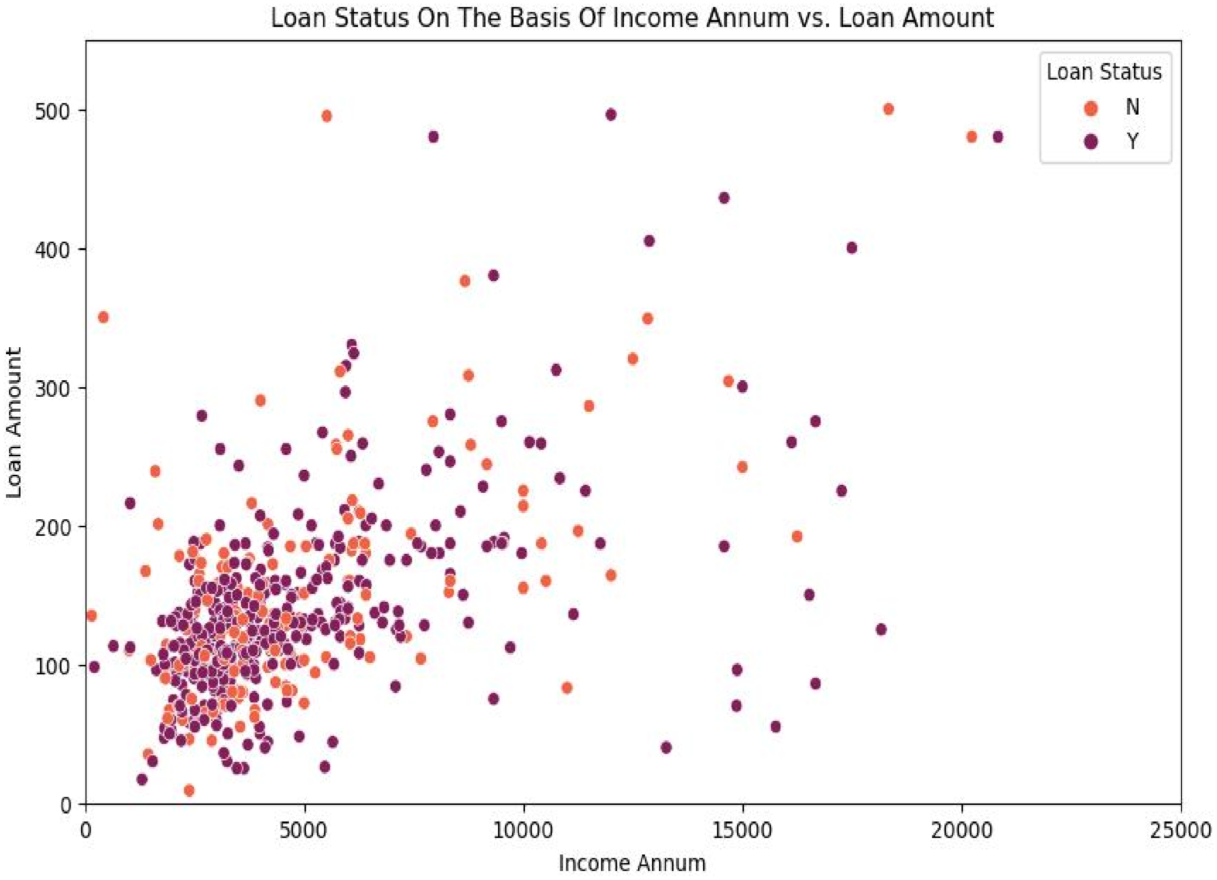
Data Analysis: Plotting and Charting

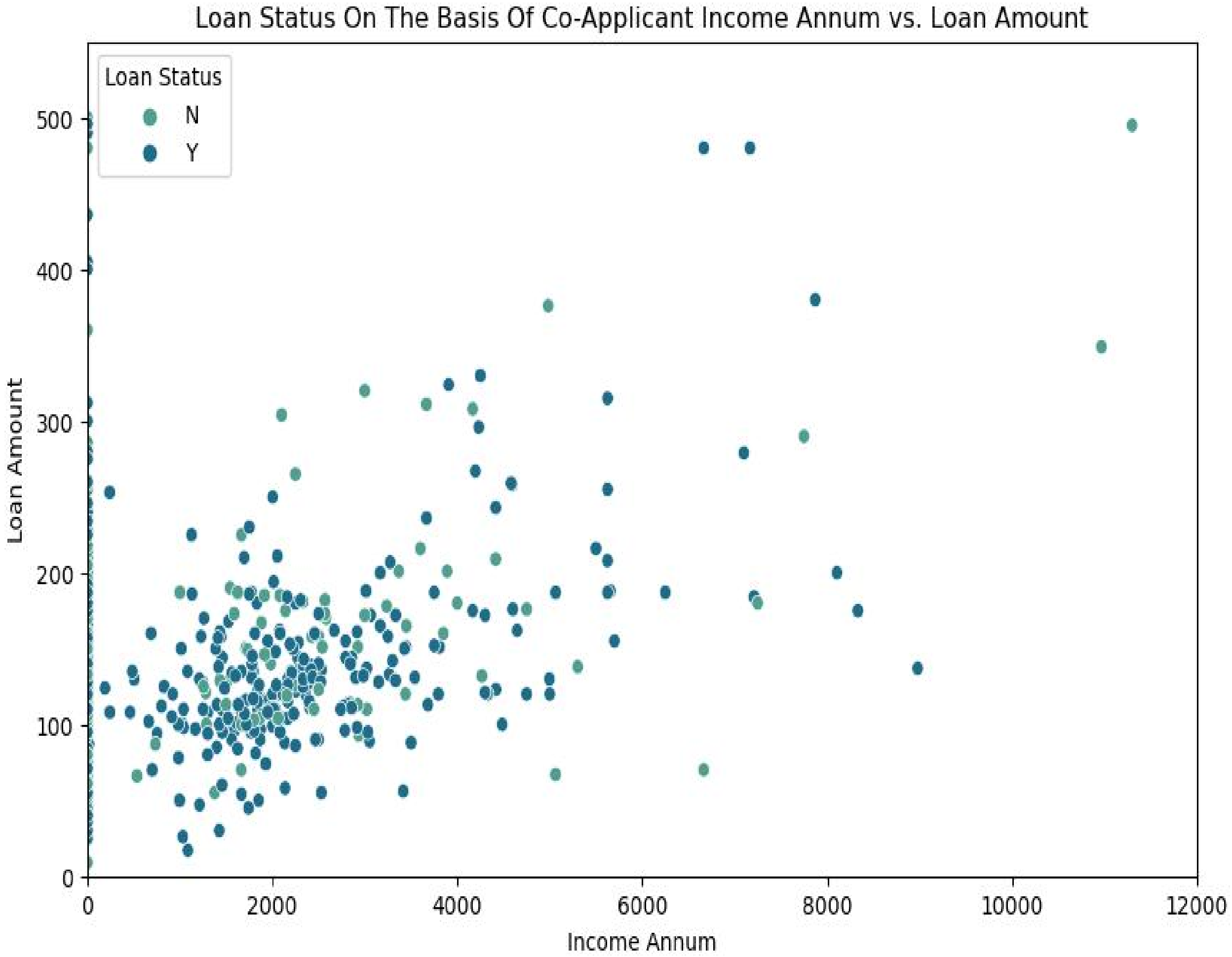

Count The Applicants Education Level


Count The Applicants Self Employment Status


Count The Number Of Loan Status Approvals

Data Analysis: Plotting and Charting

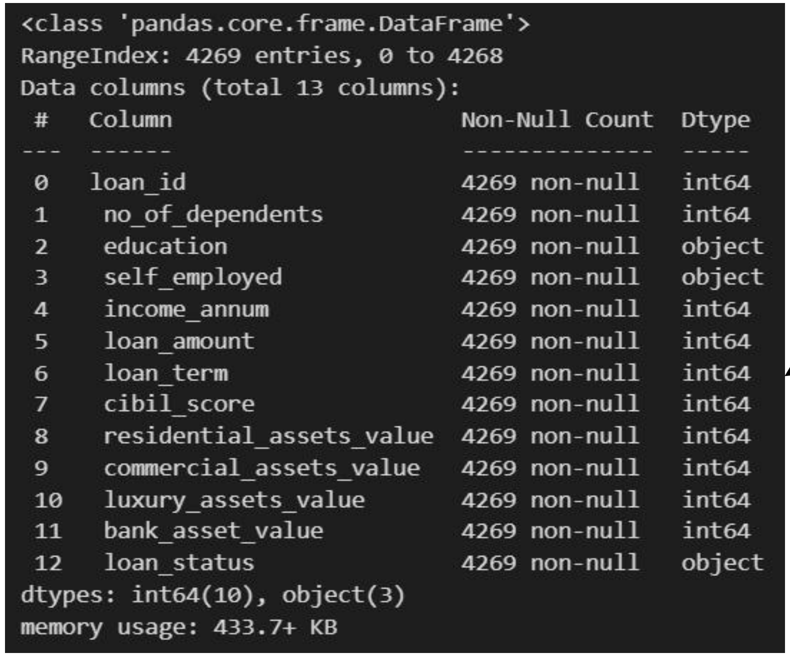




Column Details For CIBIL Data

1. Loan_ID : Unique Loan ID
2. Dependents : Number of dependents
3. Education : Applicant Education (Graduate/ Under Graduate)
4. Self_Employed : Self employed (Y/N)
5. ApplicantIncome : Applicant income
6. LoanAmount : Loan amount in thousands of dollars
7. Loan_Amount_Term : Term of loan in months
8. Cibil_Score : Cibil Score
9. Residential_Assets_Value: Value Of Residential Assets
10. Commercial_Assets_Value: Value Of Commercial Assets
11. Luxury_Assets_Value: Value Of Luxury Assets
12. Bank_Asset_Value: Value Of Bank Assets
13. Loan_Status: Loan approved (Y/N) this is the target variable

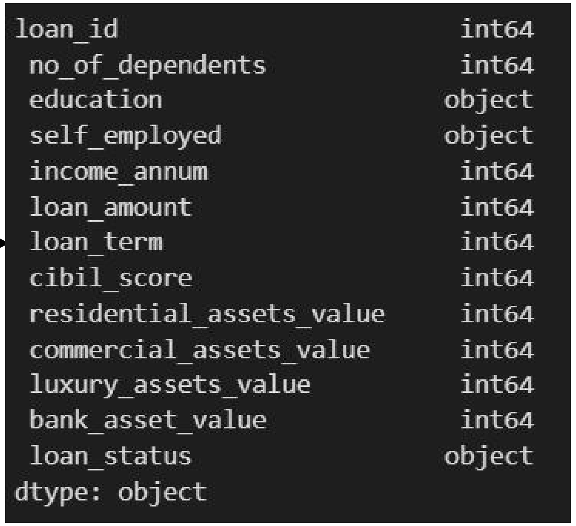
About Datasets



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4269 entries, 0 to 4268
Data columns (total 13 columns):
 #   Column                     Non-Null Count  Dtype 
---  ------                     --------------  ----- 
 0   loan_id                    4269 non-null   int64 
 1    no_of_dependents          4269 non-null   int64 
 2    education                 4269 non-null   object
 3    self_employed             4269 non-null   object
 4    income_annum              4269 non-null   int64 
 5    loan_amount               4269 non-null   int64 
 6    loan_term                 4269 non-null   int64 
 7    cibil_score               4269 non-null   int64 
 8    residential_assets_value  4269 non-null   int64 
 9    commercial_assets_value   4269 non-null   int64 
 10   luxury_assets_value       4269 non-null   int64 
 11   bank_asset_value          4269 non-null   int64 
 12   loan_status               4269 non-null   object
dtypes: int64(10), object(3)
memory usage: 433.7+ KB
```

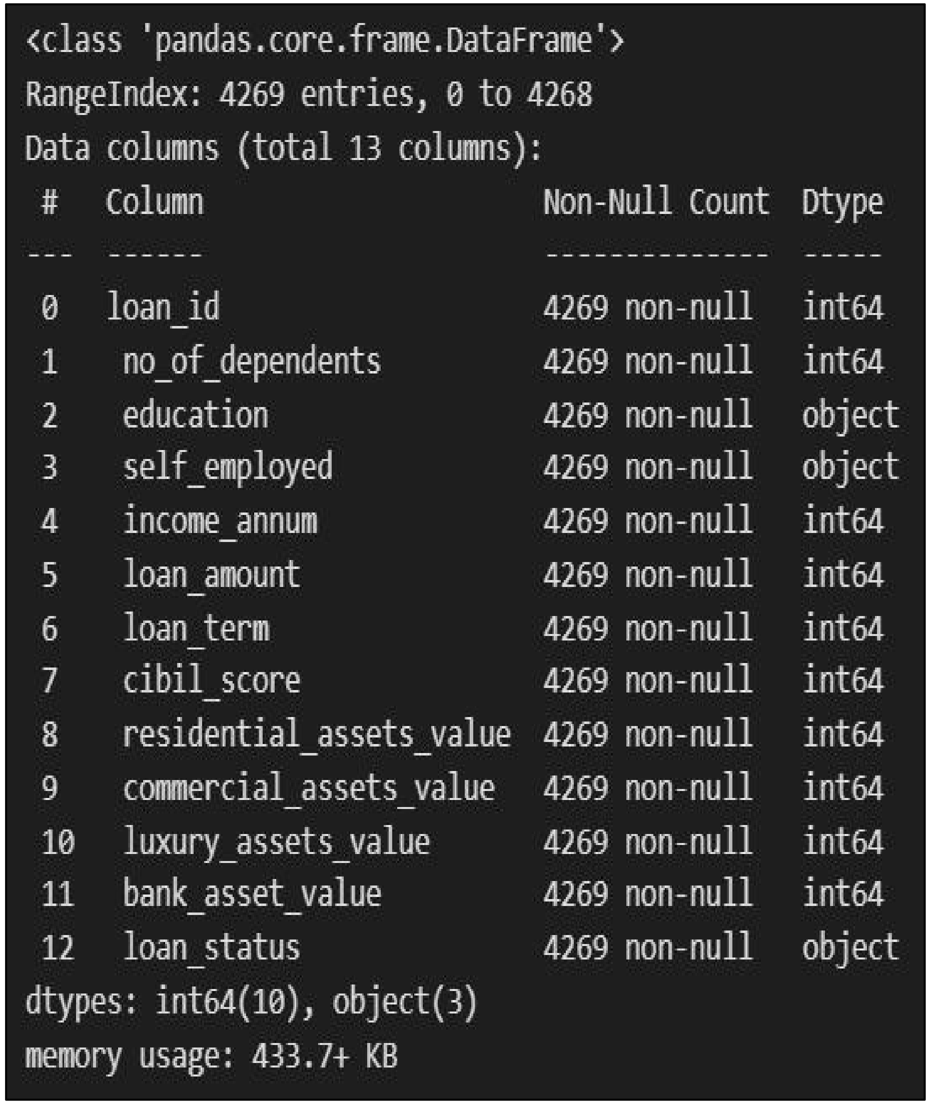
data.info()

data.dtypes



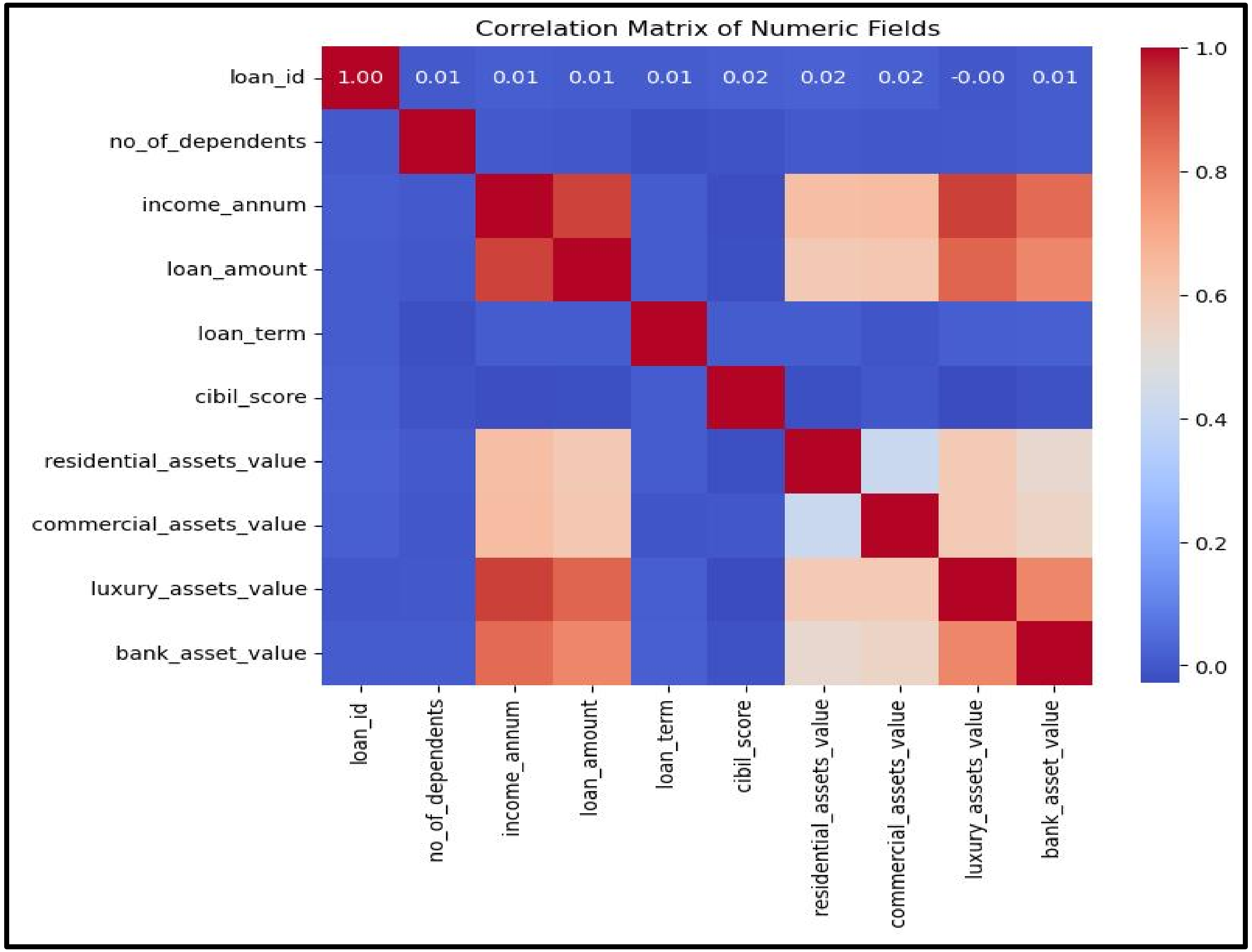
```
loan_id                       int64
 no_of_dependents             int64
 education                   object
 self_employed               object
 income_annum                 int64
 loan_amount                  int64
 loan_term                    int64
 cibil_score                  int64
 residential_assets_value     int64
 commercial_assets_value      int64
 luxury_assets_value          int64
 bank_asset_value             int64
 loan_status                 object
dtype: object
```

Read Data and Analyse



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4269 entries, 0 to 4268
Data columns (total 13 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   loan_id                   4269 non-null   int64
 1    no_of_dependents         4269 non-null   int64
 2    education                4269 non-null   object
 3    self_employed            4269 non-null   object
 4    income_annum             4269 non-null   int64
 5    loan_amount              4269 non-null   int64
 6    loan_term                4269 non-null   int64
 7    cibil_score              4269 non-null   int64
 8    residential_assets_value 4269 non-null   int64
 9    commercial_assets_value  4269 non-null   int64
 10   luxury_assets_value      4269 non-null   int64
 11   bank_asset_value         4269 non-null   int64
 12   loan_status              4269 non-null   object
dtypes: int64(10), object(3)
memory usage: 433.7+ KB
```

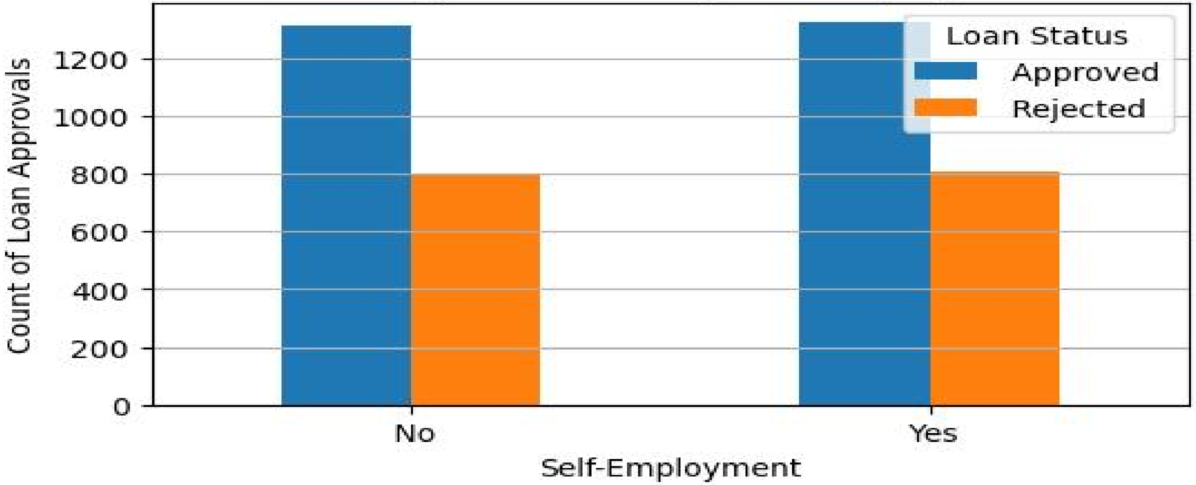
Dataset Info



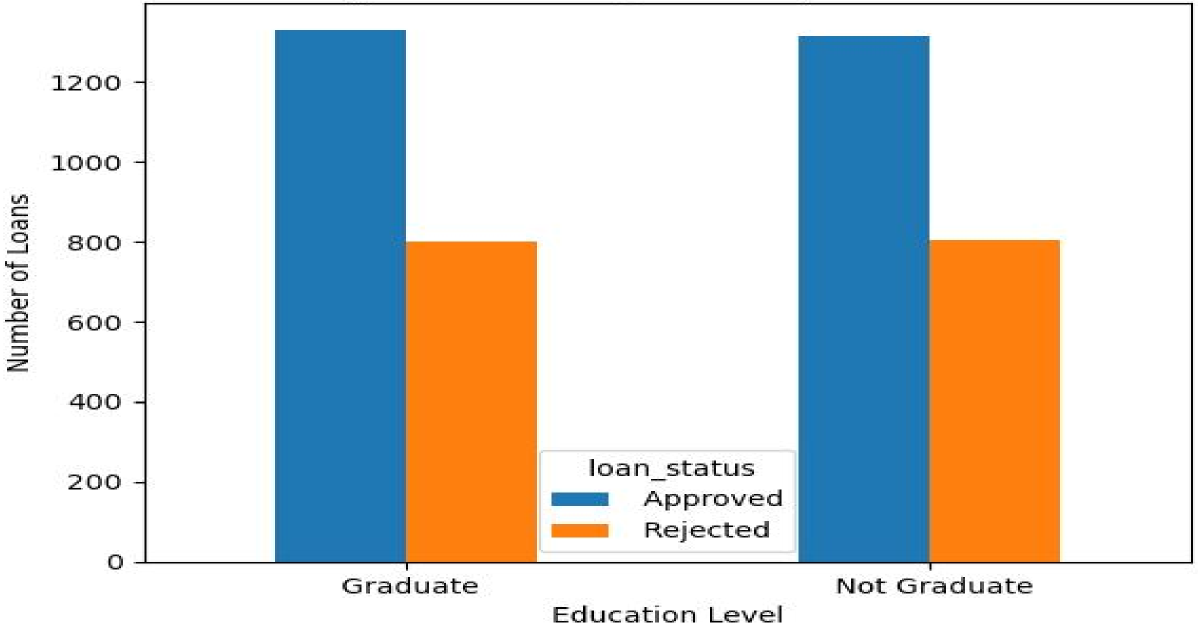
Correlation Matrix of Dataset Fields

Data Analysis: Plotting and Charting

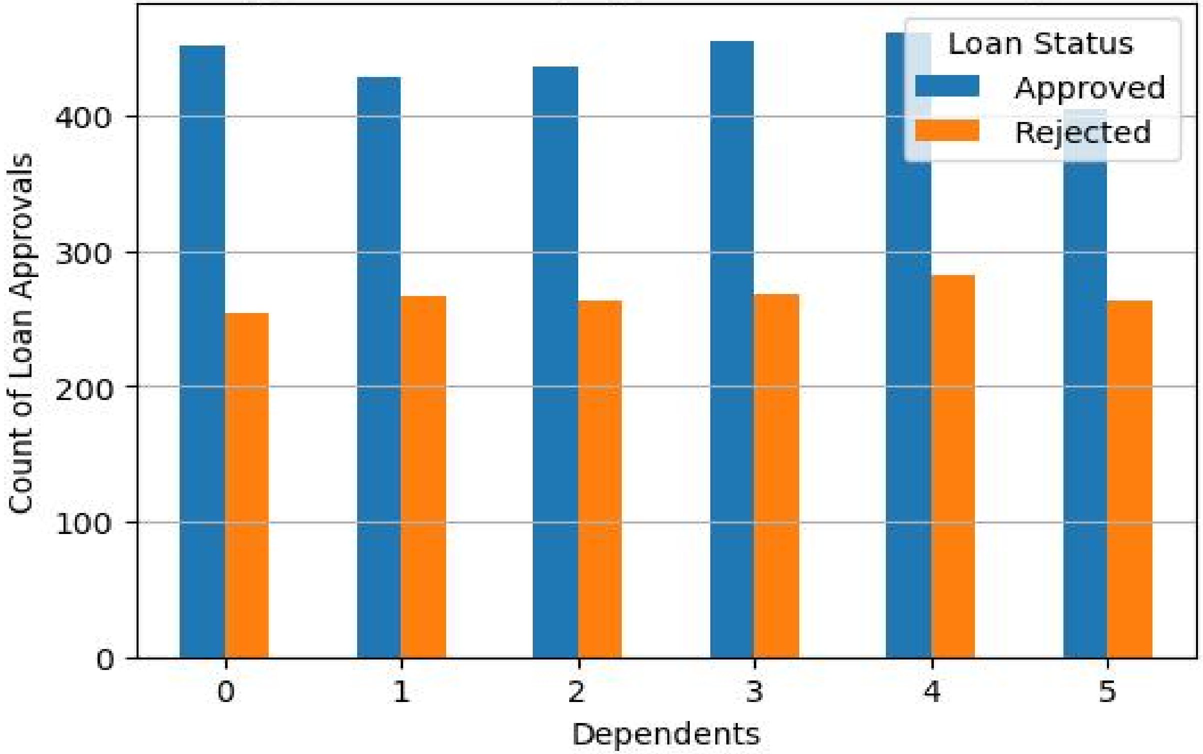
Loan Approval Count by Self-Employment



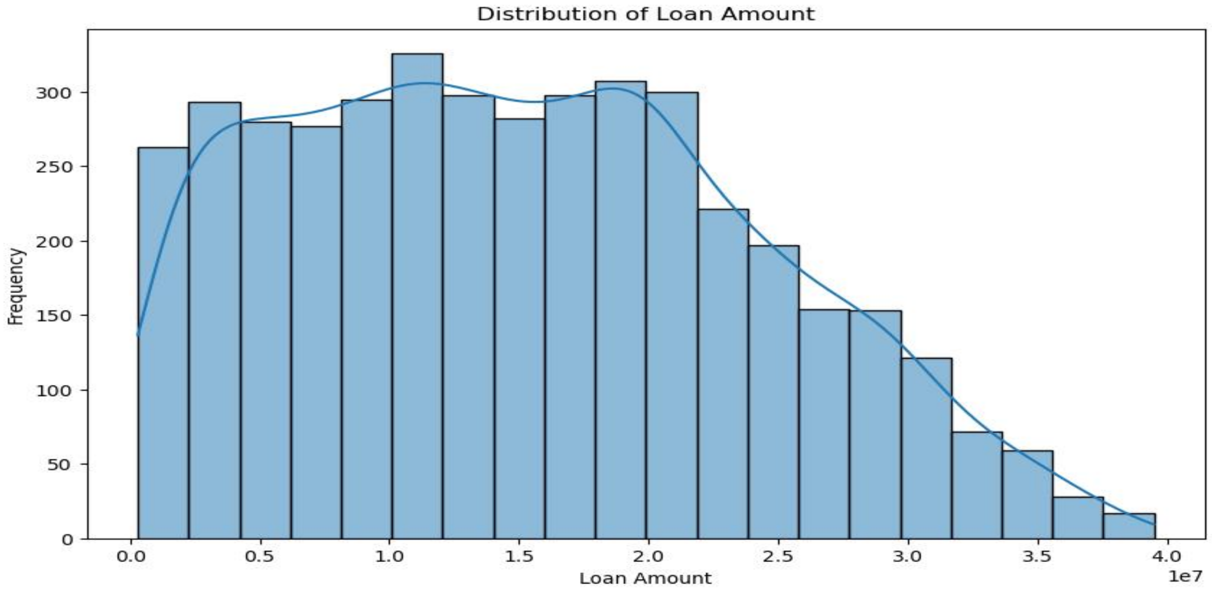
Loan Approvals and Rejections by Education Level

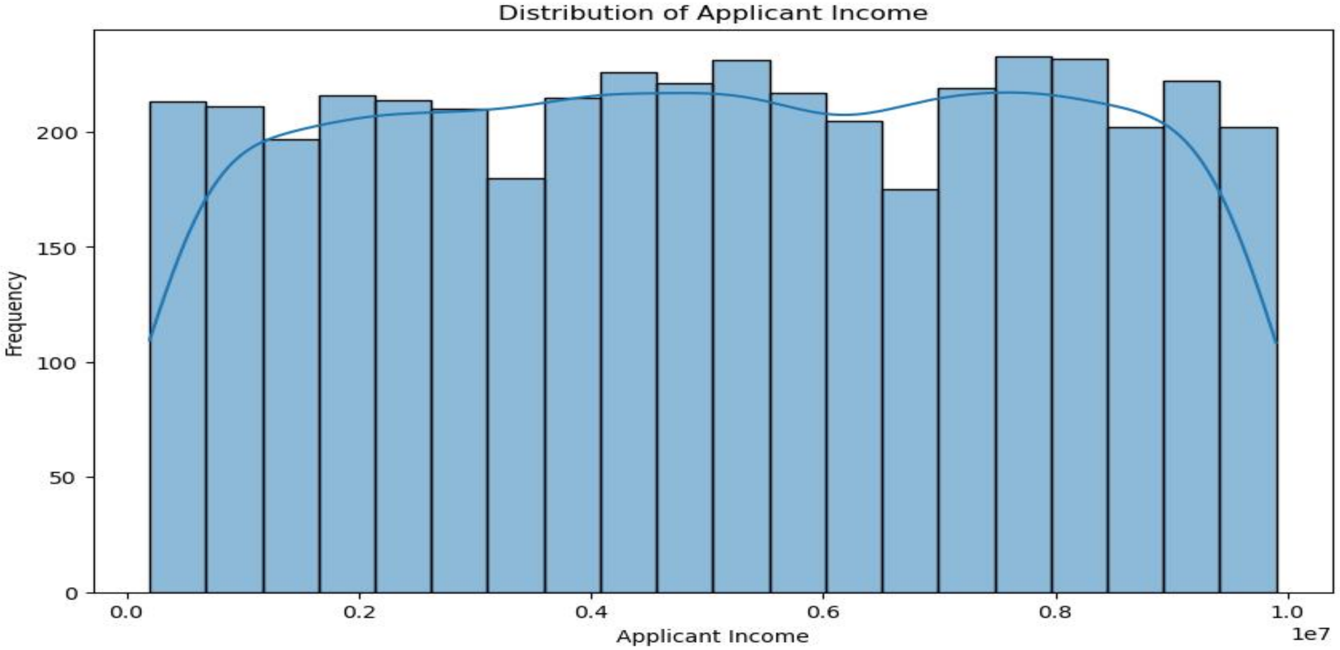


Loan Approval Count by Applicant Number of dependents

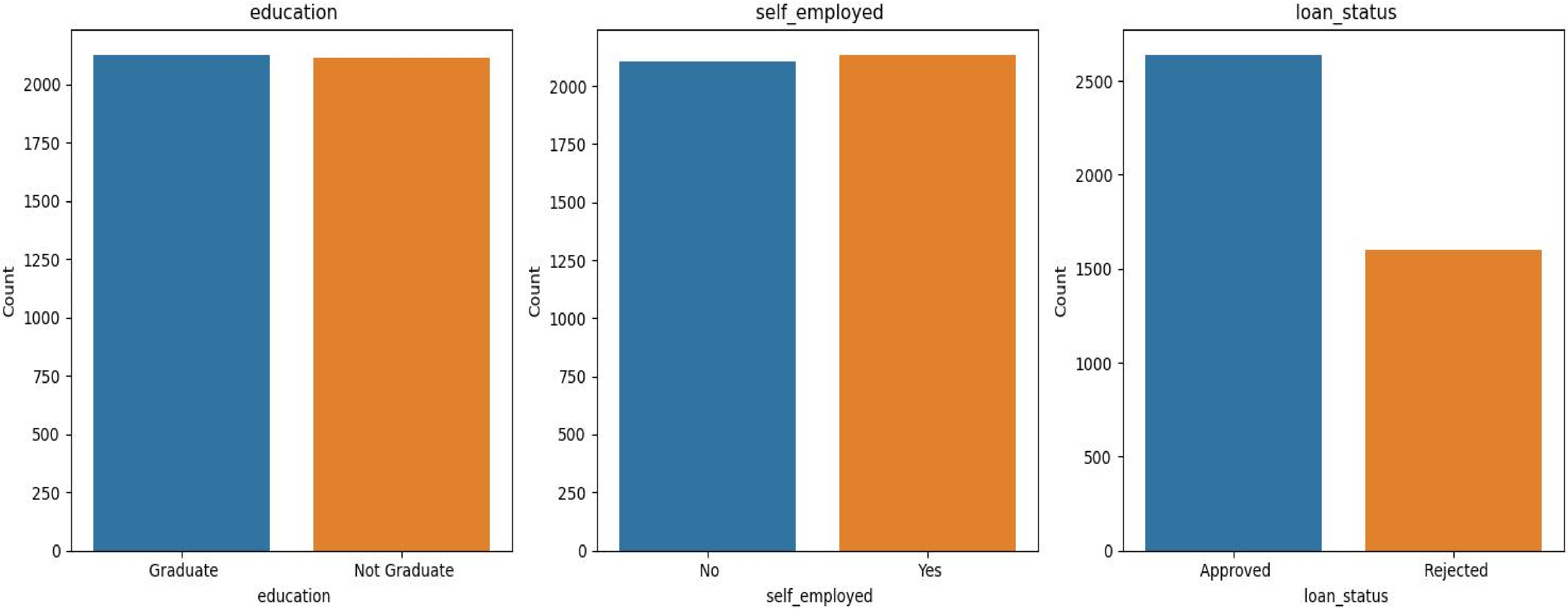


Data Analysis: Plotting and Charting

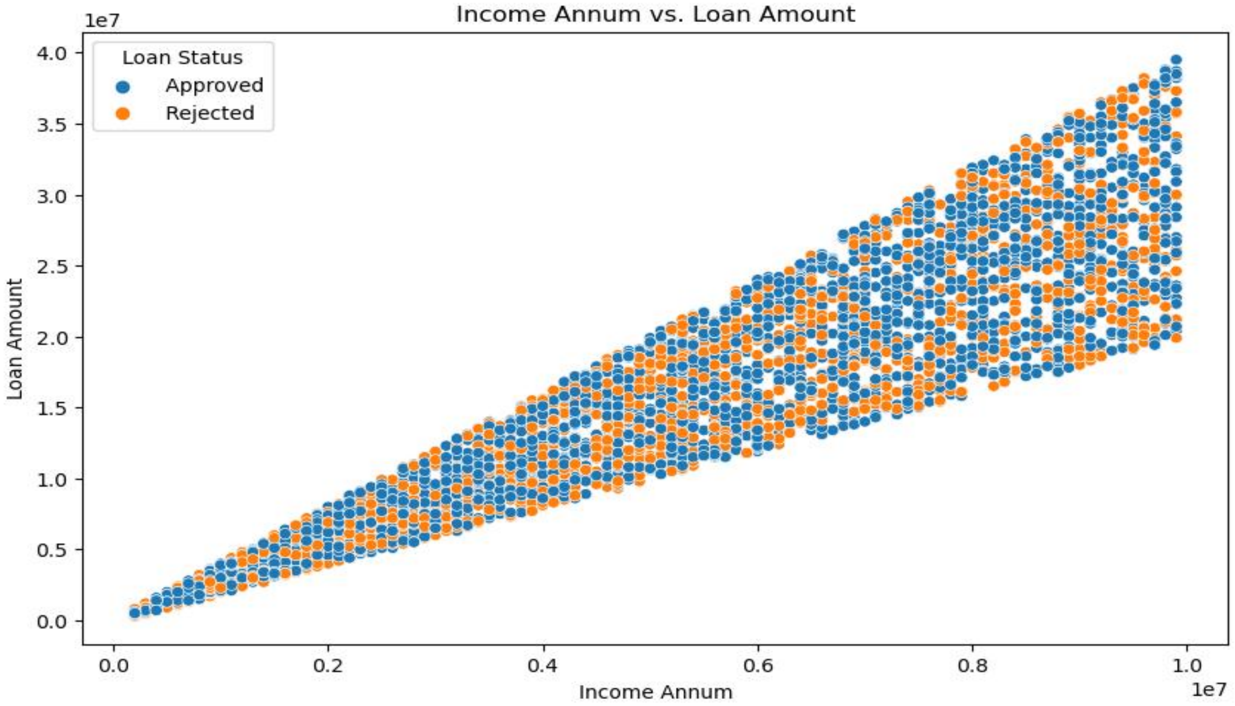


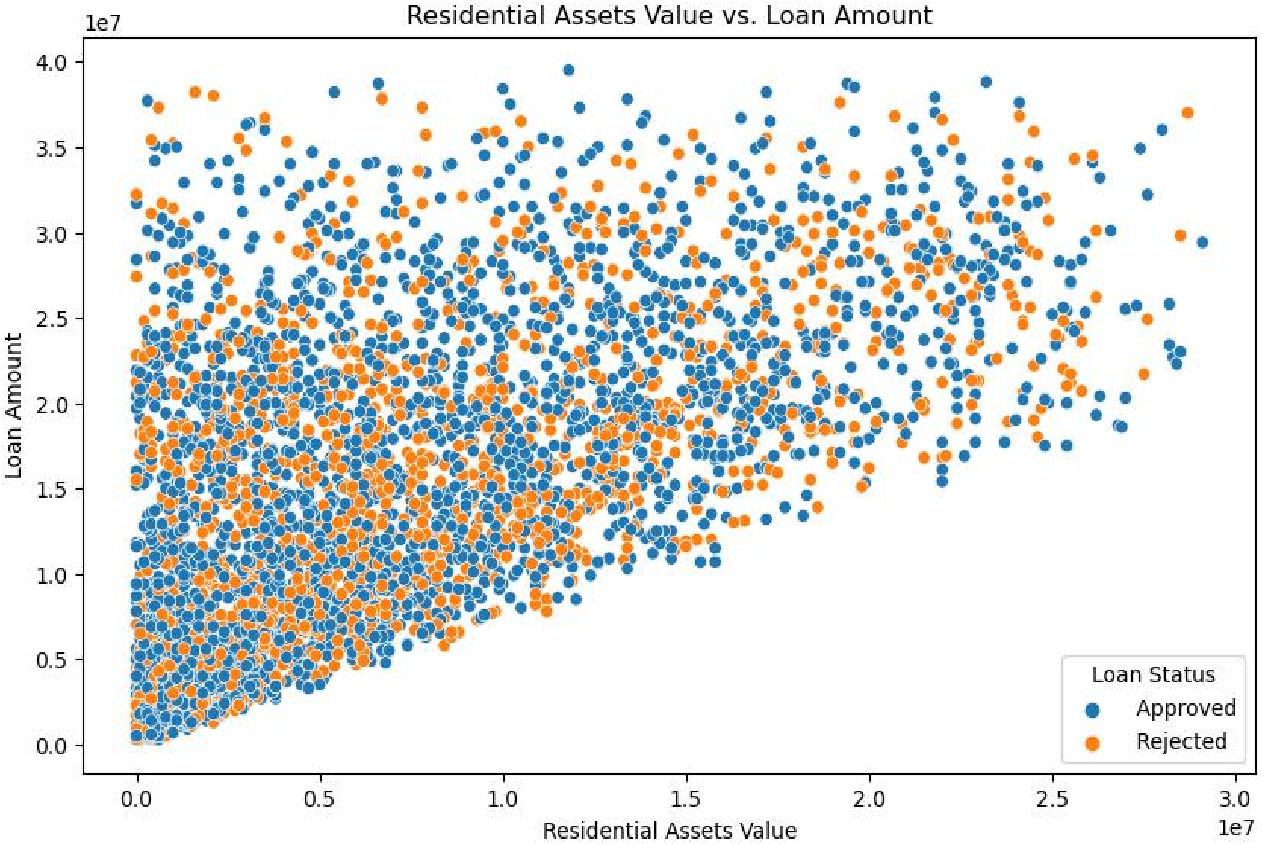


Data Analysis: Plotting and Charting

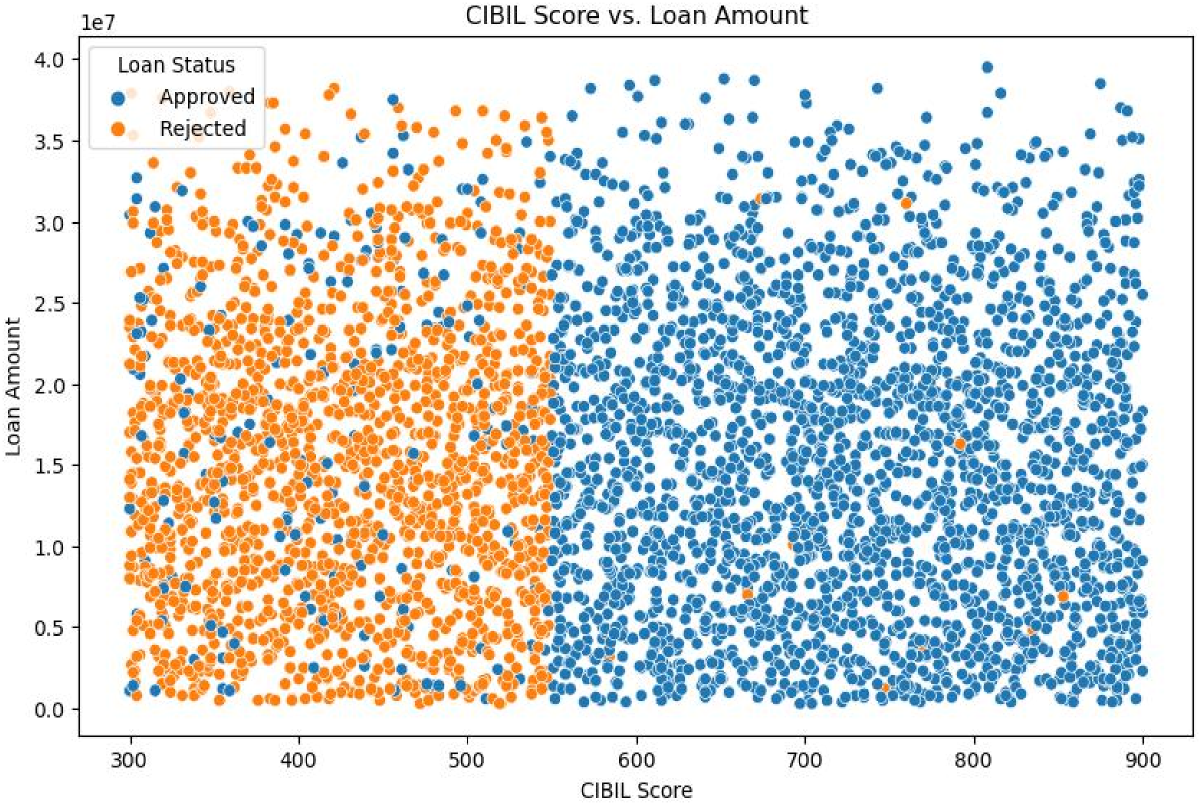


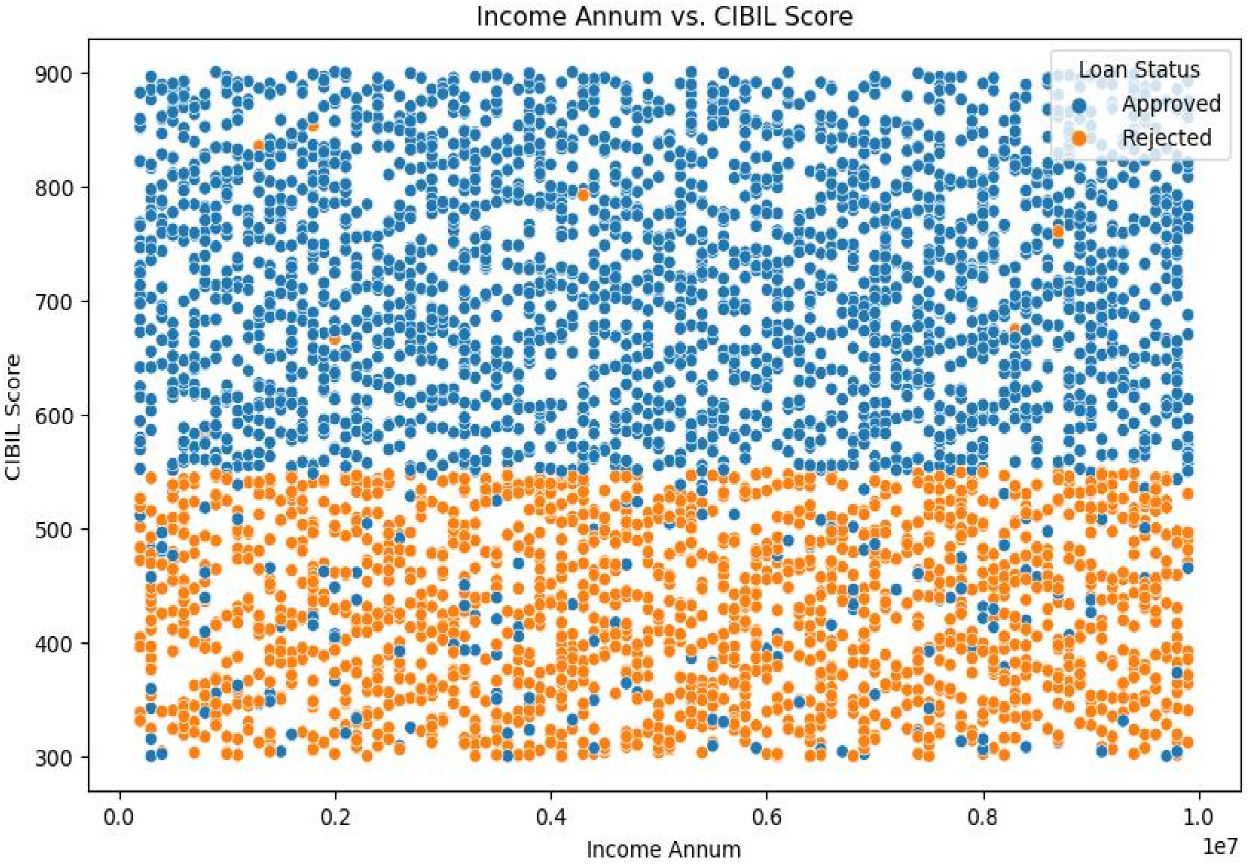
Data Analysis: Plotting and Charting





Data Analysis: Plotting and Charting





Convert Categorical Variables To Numeric

- Categorical features refer to string data types and can be easily understood by human beings.
- However, machines cannot interpret the categorical data directly. Therefore, the categorical data must be converted into numerical data for further processing.
- We mapped categorical variables to numerical values for better processing by machine learning algorithms.

'Graduate': 1, 'Not Graduate’: 0
'Yes': 1, ' No': 0
'Approved': 1, 'Rejected': 0

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Min-max Scaling

- Min-max scaling, also known as normalization, is a technique commonly used in data preprocessing. It is used to transform numerical features into a specific range, typically between 0 and 1. However, machines cannot interpret the categorical data directly.
- Many machine learning algorithms perform better when the input features are normalized. By scaling the features to a specific range, you can prevent any particular feature from dominating the learning process.

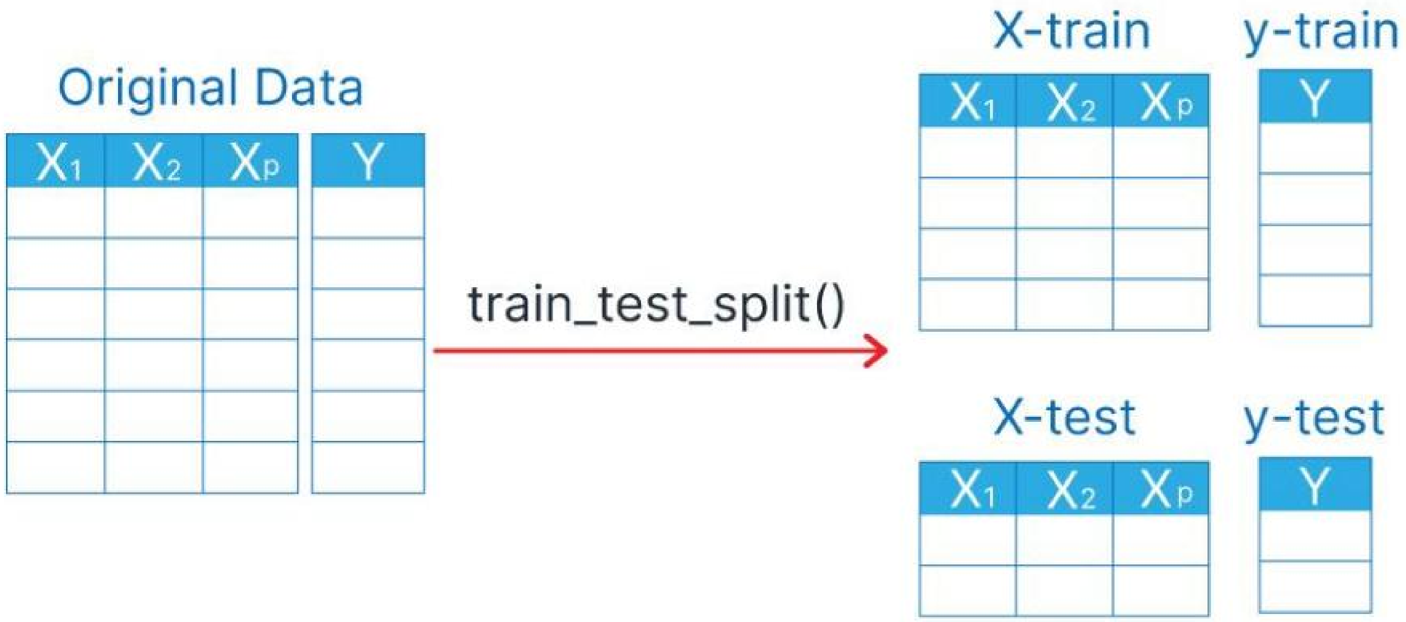
Data Preparation

Input : X
Output : Y

Supervised machine learning is a type of machine learning that learns the relationship between input and output. The inputs are known as features or X variables and output is generally referred to as the target or y variable. The type of data which contains both the features, and the target is known as labeled data.

Train Test Split :-

Train-test split divides the data once into distinct training and test sets used for model evaluation.



Original Data

X_1	X_2	X_p	Y

train_test_split()

X-train

X_1	X_2	X_p

y-train

Y

X-test

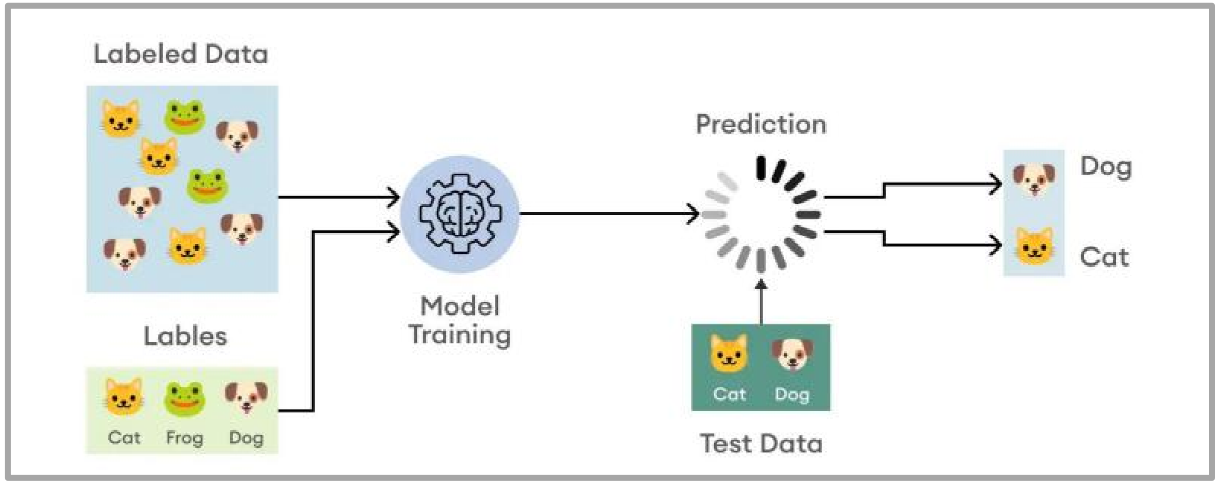
X_1	X_2	X_p

y-test

Y

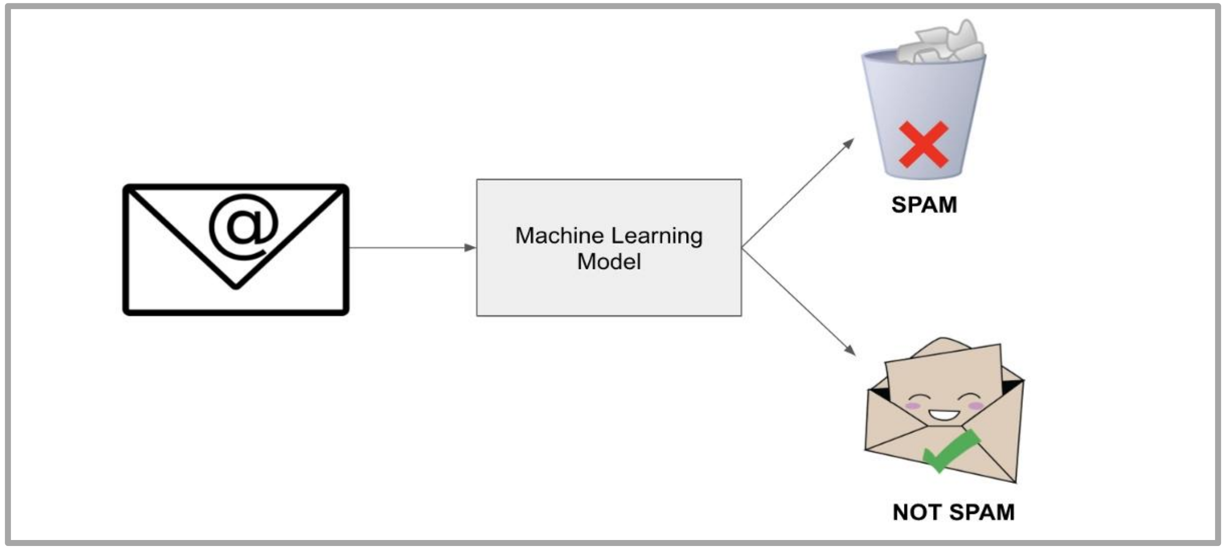
Machine Learning : Classification

- Classification is a supervised machine learning method where the model tries to predict the correct label of a given input data.
- In classification, the model is fully trained using the training data, and then it is evaluated on test data before being used to perform prediction on new unseen data.



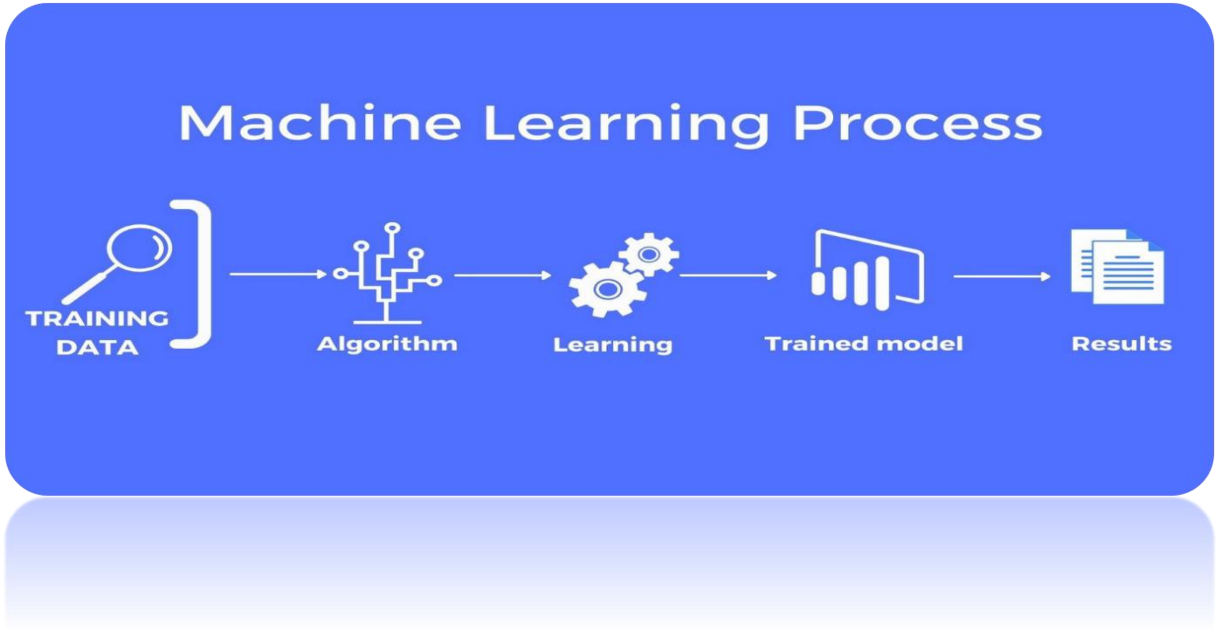
Binary Classification:-

- In a binary classification task, the goal is to classify the input data into two mutually exclusive categories.
- The training data in such a situation is labeled in a binary format: true and false; positive and negative; O and 1; spam and not spam, etc. depending on the problem being tackled.
- The loan approvals prediction is a binary classification problem.



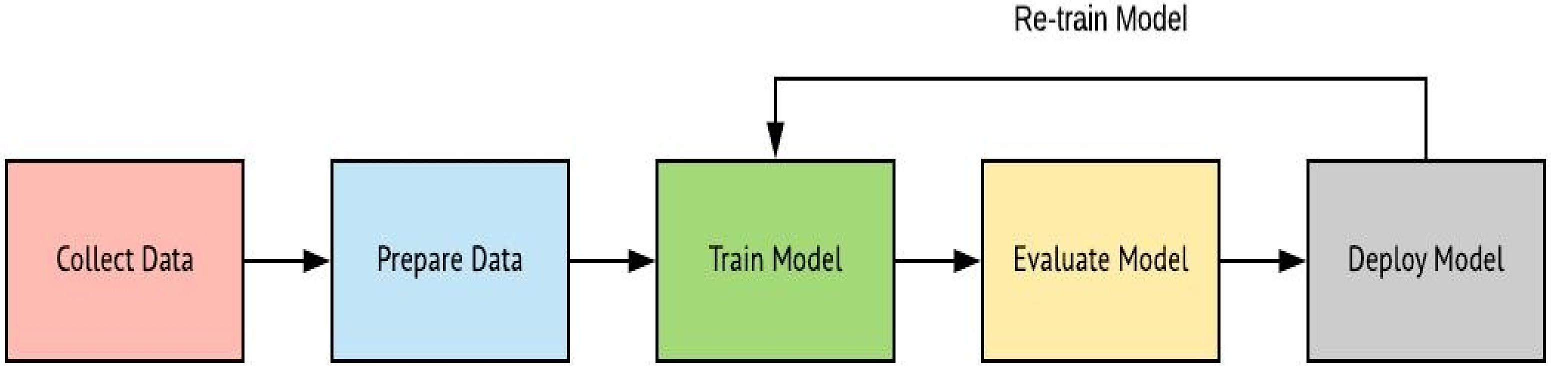
Model Training

- A training model is a dataset that is used to train an ML algorithm.
- It consists of the sample output data and the corresponding sets of input data that have an influence on the output.
- The training model is used to run the input data through the algorithm to correlate the processed output against the sample output.



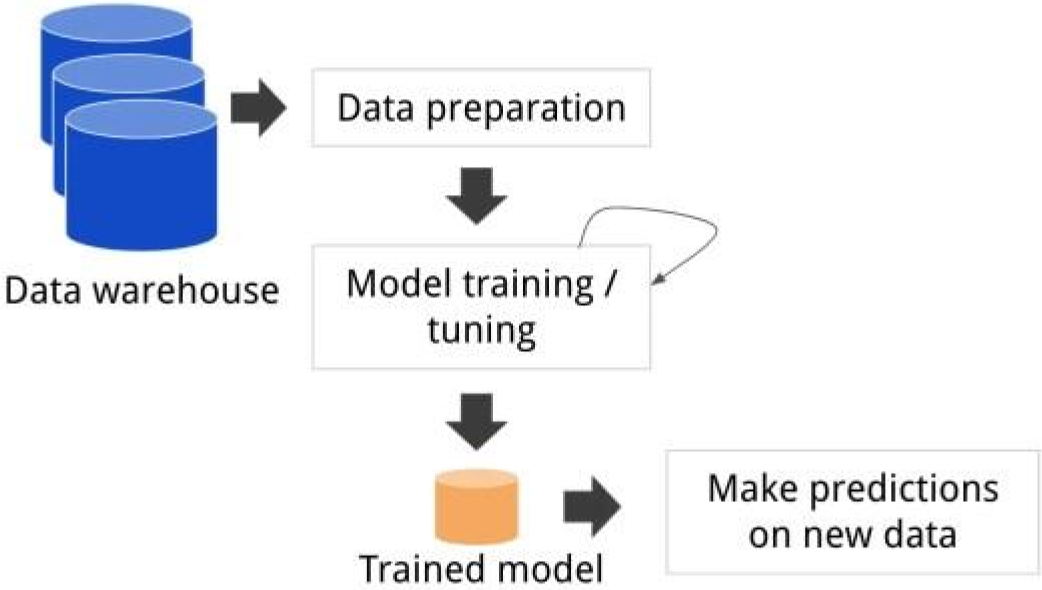
Models Used: Package-Sci-Kit Learn

- Logistic Regression
- Support Vector Machine (SVM)
- Decision Tree Classifier
- Random Forest Classifier
- Gradient Boosting Classifier
- Random Search CV



Model Prediction and Evaluation

Each input variable gets a label marking a category. In other words, the classification technique is used to map the input data to one of the categorial output labels.

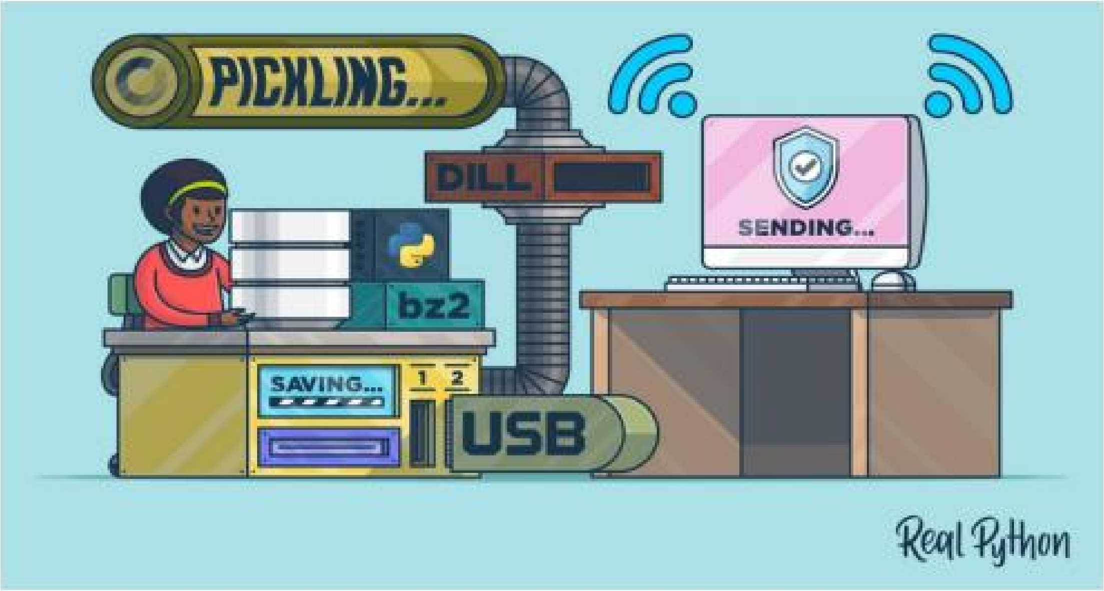


Model Evaluation

Evaluating the performance of your classification model is crucial to ensure its accuracy and effectiveness.

Building GUI Application

1. Saving the Model and Scaler.
2. Taking test Inputs.
3. Scaling the Inputs.
4. Passing the inputs to the model.
5. Getting the Output.
6. Displaying if Loan will be Approved or Rejected.
7. Building a GUI



LOAN APPLICATION

Personal Information

Name (Last) PUBLIC	(First) JOHN	(Middle Initial)	Home Telephone (11)11 - 1111
Address (Mailing Address) 12345 MAIN STREET	(City) ANYWHERE	(State) ZZ (Zip) 999999	Other Telephone (22)22 2222
E-Mail Address JQPJQPJQP@JQP.JQP	APPLICANT IS UNDER REVIEW		
	SUBJECT	REVIEW	

Services needed

UNDER REVIEW

High School Graduate Or General Education (GED) Test Passed? Yes No

highest grade completed

Current Income

Military (Most recent first)

Credits Earned

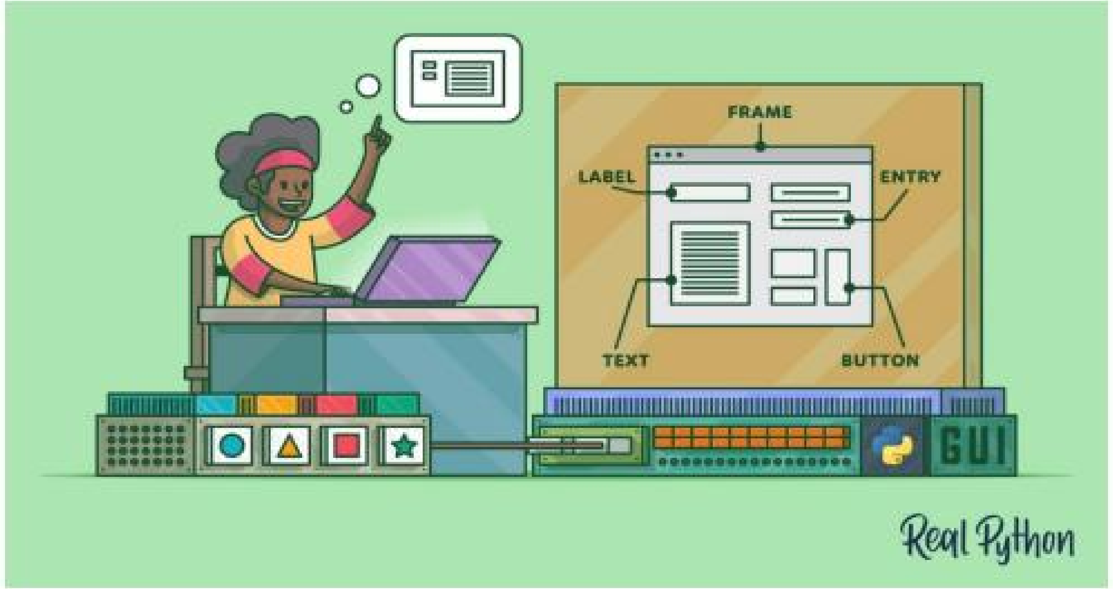
Quarterly or

Other (Specify)

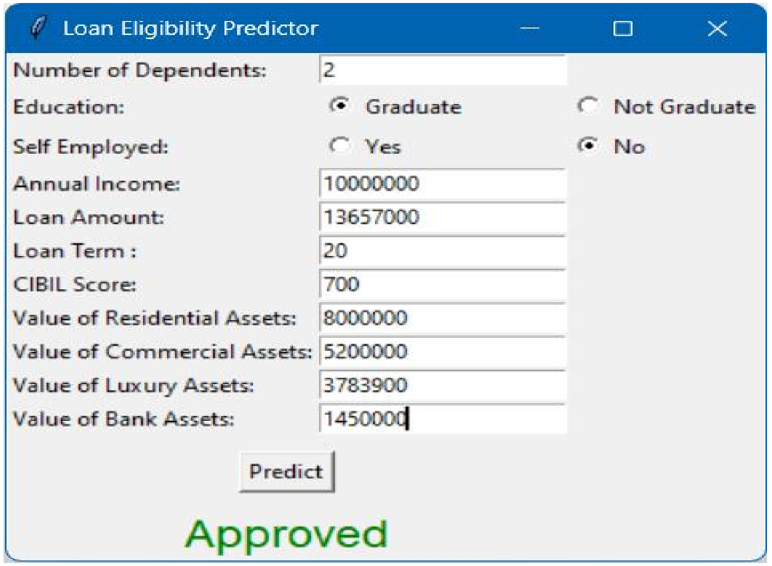
Graduate

Major or Subject

APPROVED



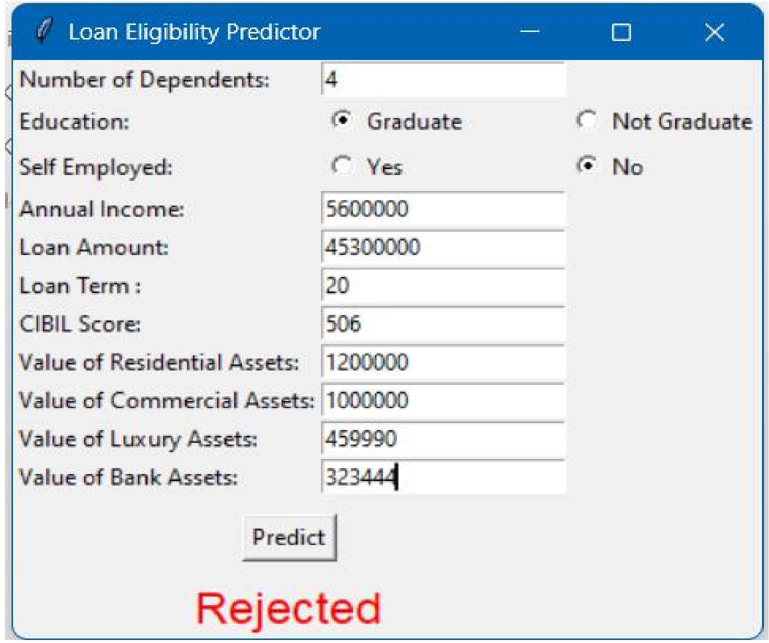
GUI Interface



Loan Eligibility Predictor

Number of Dependents:	2	
Education:	● Graduate	○ Not Graduate
Self Employed:	○ Yes	● No
Annual Income:	10000000	
Loan Amount:	13657000	
Loan Term :	20	
CIBIL Score:	700	
Value of Residential Assets:	8000000	
Value of Commercial Assets:	5200000	
Value of Luxury Assets:	3783900	
Value of Bank Assets:	1450000	

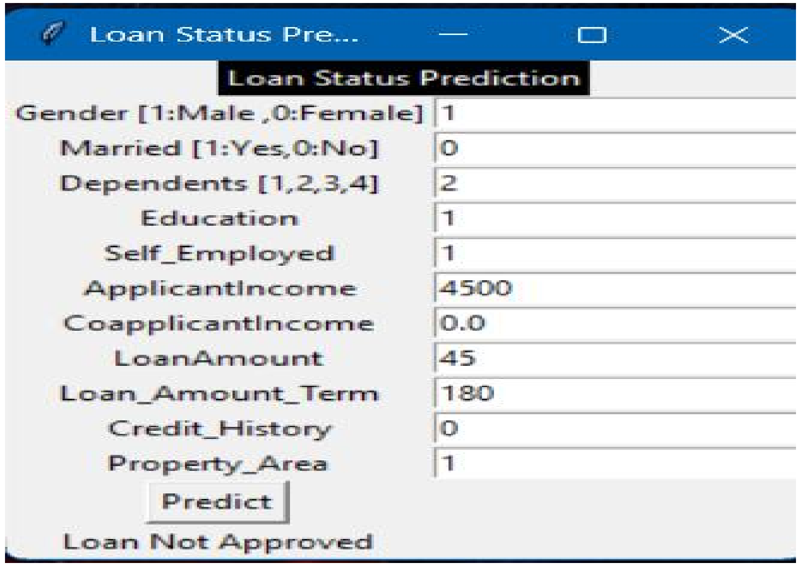
Approved



Loan Eligibility Predictor

Number of Dependents:	4	
Education:	● Graduate	○ Not Graduate
Self Employed:	○ Yes	● No
Annual Income:	5600000	
Loan Amount:	45300000	
Loan Term :	20	
CIBIL Score:	506	
Value of Residential Assets:	1200000	
Value of Commercial Assets:	1000000	
Value of Luxury Assets:	459990	
Value of Bank Assets:	323444	

Rejected

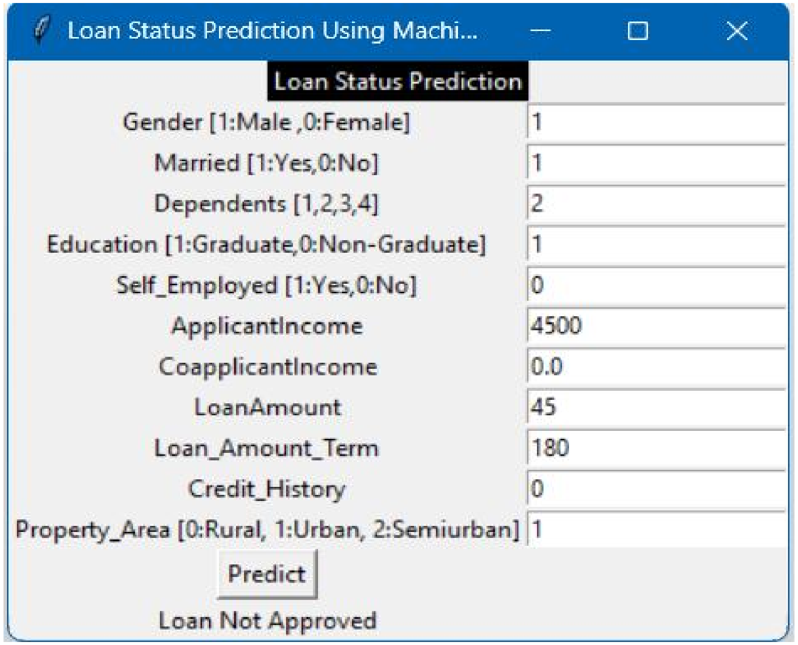


Loan Status Pre...

Loan Status Prediction

Gender [1:Male ,0:Female]	1
Married [1:Yes,0:No]	0
Dependents [1,2,3,4]	2
Education	1
Self_Employed	1
ApplicantIncome	4500
CoapplicantIncome	0.0
LoanAmount	45
Loan_Amount_Term	180
Credit_History	0
Property_Area	1

Loan Not Approved

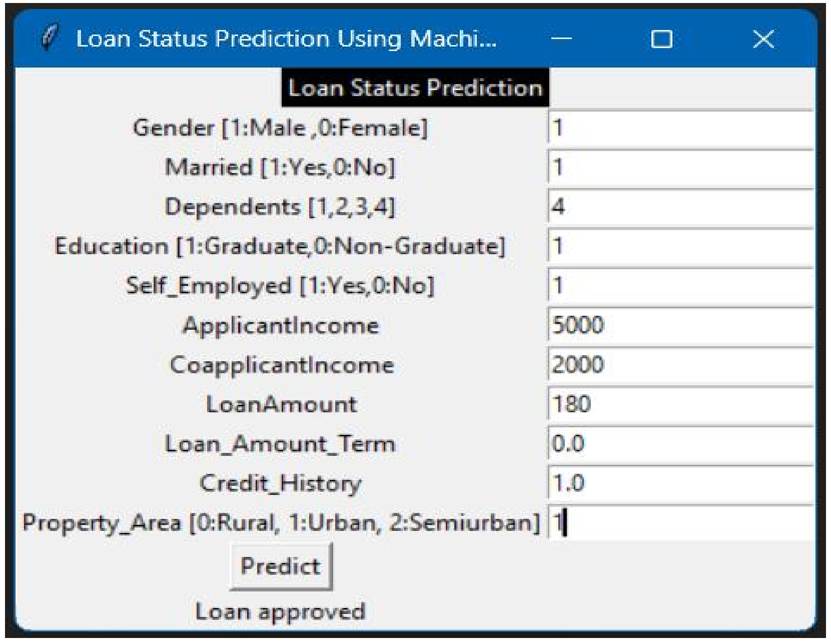


Loan Status Prediction Using Machi...

Loan Status Prediction

Gender [1:Male ,0:Female]	1
Married [1:Yes,0:No]	1
Dependents [1,2,3,4]	2
Education [1:Graduate,0:Non-Graduate]	1
Self_Employed [1:Yes,0:No]	0
ApplicantIncome	4500
CoapplicantIncome	0.0
LoanAmount	45
Loan_Amount_Term	180
Credit_History	0
Property_Area [0:Rural, 1:Urban, 2:Semiurban]	1

Loan Not Approved



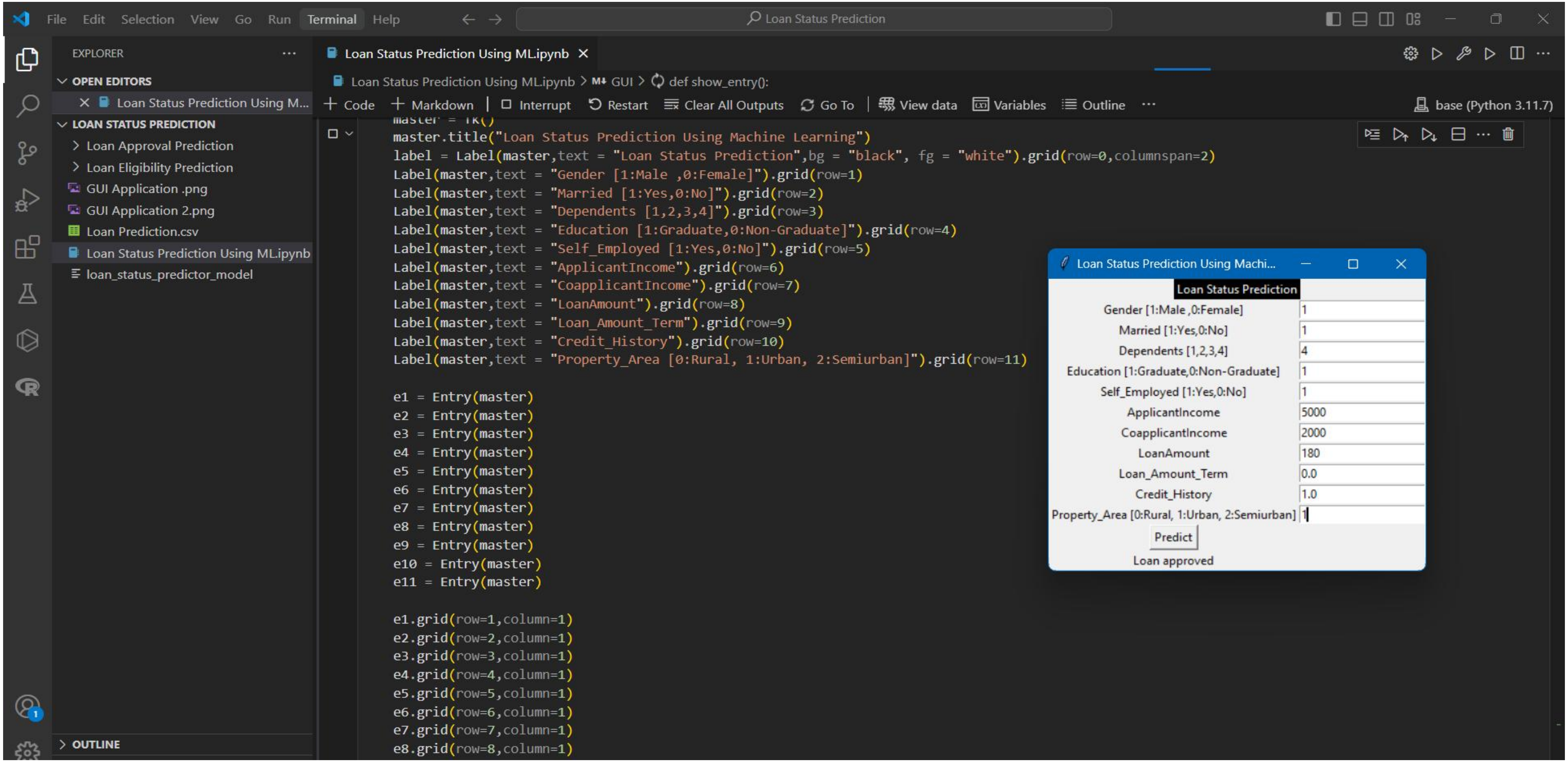
Loan Status Prediction Using Machi...

Loan Status Prediction

Gender [1:Male ,0:Female]	1
Married [1:Yes,0:No]	1
Dependents [1,2,3,4]	4
Education [1:Graduate,0:Non-Graduate]	1
Self_Employed [1:Yes,0:No]	1
ApplicantIncome	5000
CoapplicantIncome	2000
LoanAmount	180
Loan_Amount_Term	0.0
Credit_History	1.0
Property_Area [0:Rural, 1:Urban, 2:Semiurban]	1

Loan approved

GUI Interface



```python
master = Tk()
master.title("Loan Status Prediction Using Machine Learning")
label = Label(master,text = "Loan Status Prediction",bg = "black", fg = "white").grid(row=0,columnspan=2)
Label(master,text = "Gender [1:Male ,0:Female]").grid(row=1)
Label(master,text = "Married [1:Yes,0:No]").grid(row=2)
Label(master,text = "Dependents [1,2,3,4]").grid(row=3)
Label(master,text = "Education [1:Graduate,0:Non-Graduate]").grid(row=4)
Label(master,text = "Self_Employed [1:Yes,0:No]").grid(row=5)
Label(master,text = "ApplicantIncome").grid(row=6)
Label(master,text = "CoapplicantIncome").grid(row=7)
Label(master,text = "LoanAmount").grid(row=8)
Label(master,text = "Loan_Amount_Term").grid(row=9)
Label(master,text = "Credit_History").grid(row=10)
Label(master,text = "Property_Area [0:Rural, 1:Urban, 2:Semiurban]").grid(row=11)

e1 = Entry(master)
e2 = Entry(master)
e3 = Entry(master)
e4 = Entry(master)
e5 = Entry(master)
e6 = Entry(master)
e7 = Entry(master)
e8 = Entry(master)
e9 = Entry(master)
e10 = Entry(master)
e11 = Entry(master)

e1.grid(row=1,column=1)
e2.grid(row=2,column=1)
e3.grid(row=3,column=1)
e4.grid(row=4,column=1)
e5.grid(row=5,column=1)
e6.grid(row=6,column=1)
e7.grid(row=7,column=1)
e8.grid(row=8,column=1)
```

Loan Status Prediction Using Machi...

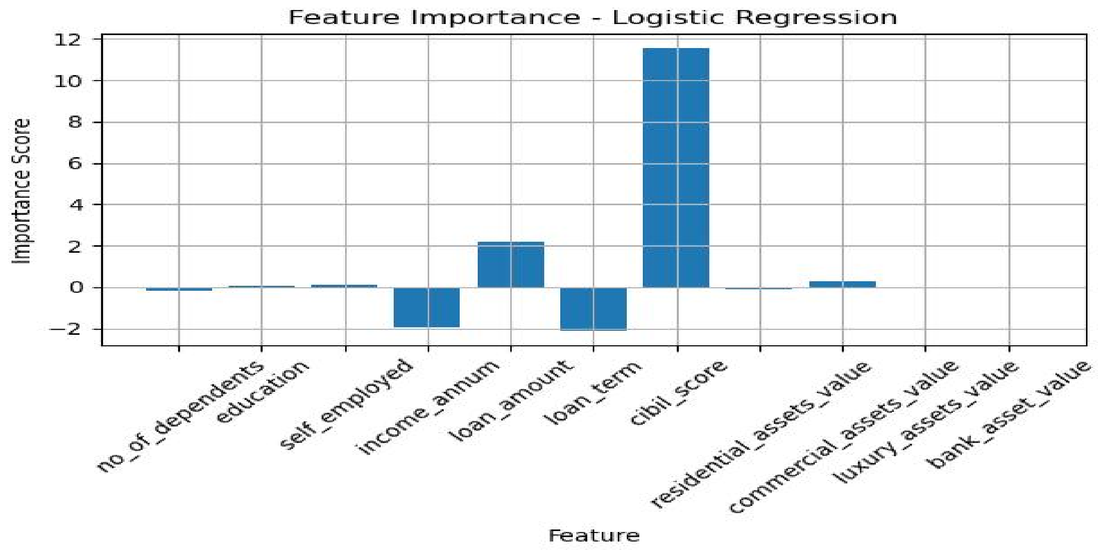
Loan Status Prediction

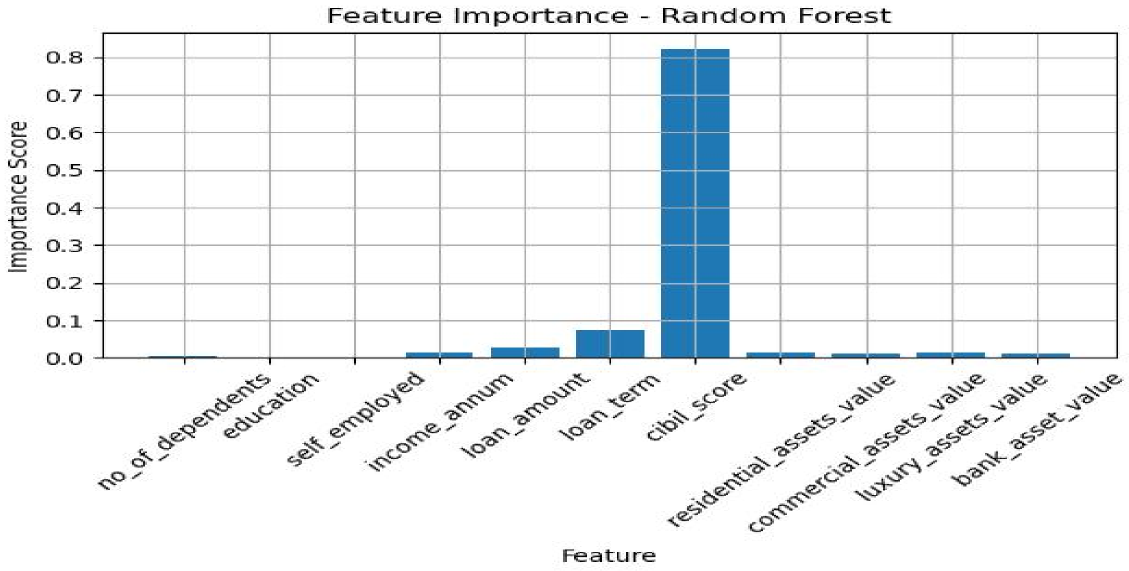
Gender [1:Male ,0:Female]	1
Married [1:Yes,0:No]	1
Dependents [1,2,3,4]	4
Education [1:Graduate,0:Non-Graduate]	1
Self_Employed [1:Yes,0:No]	1
ApplicantIncome	5000
CoapplicantIncome	2000
LoanAmount	180
Loan_Amount_Term	0.0
Credit_History	1.0
Property_Area [0:Rural, 1:Urban, 2:Semiurban]	1

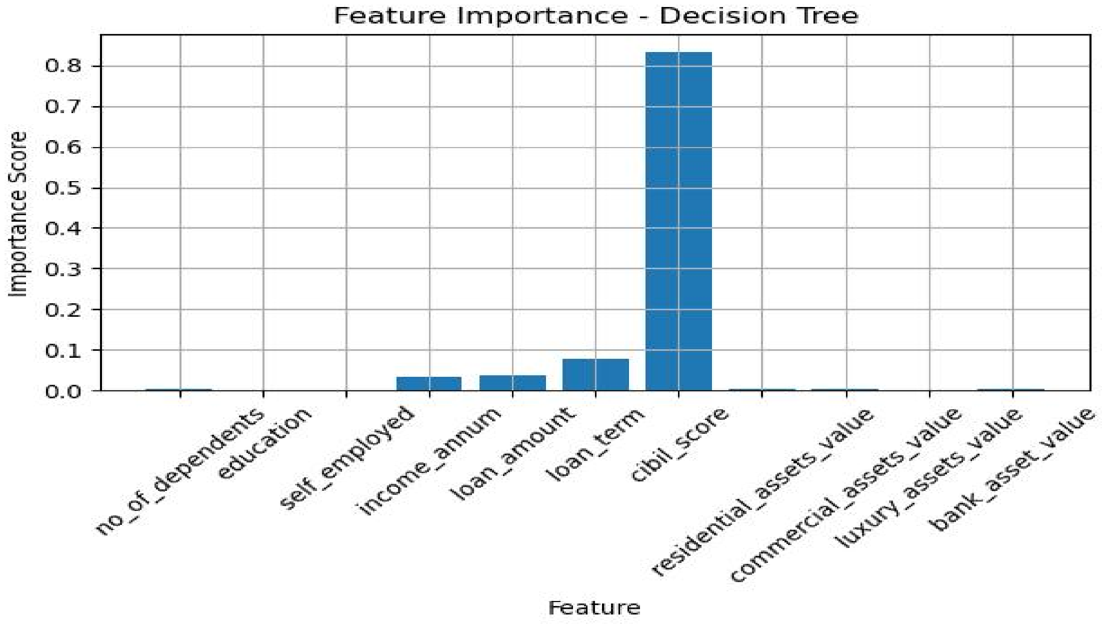
Predict

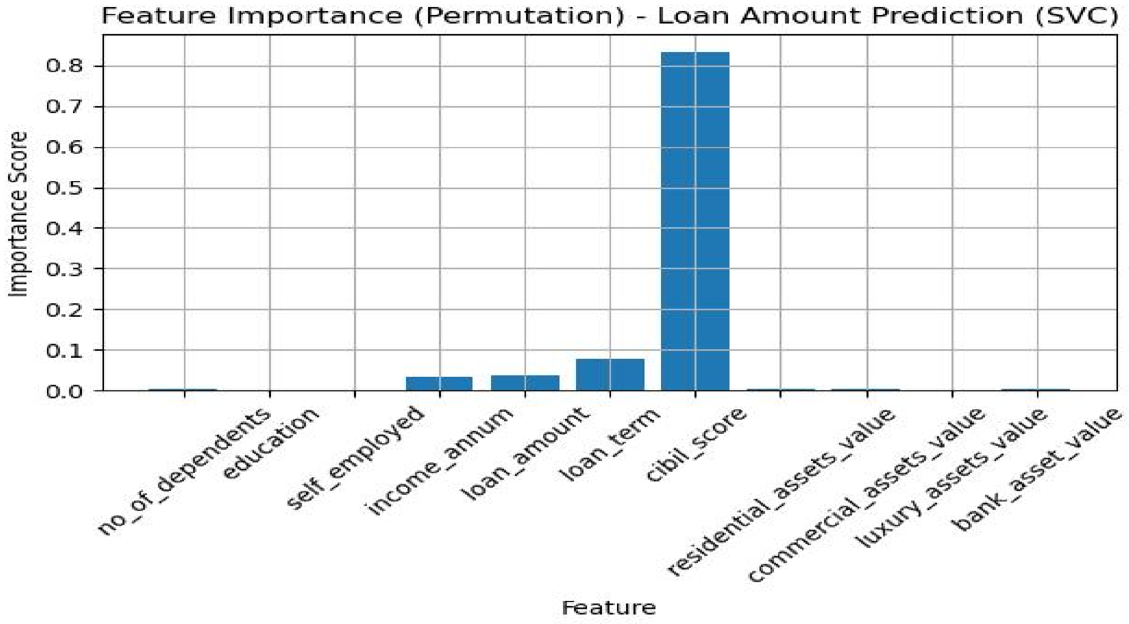
Loan approved

Feature Importance


Feature Importance - Logistic Regression


Feature Importance - Random Forest


Feature Importance - Decision Tree


Feature Importance (Permutation) - Loan Amount Prediction (SVC)

Conclusion:-

1. The project involved assessing the performance of different machine learning models on a dataset. To improve the model, more data can be collected.
2. The models used were Decision Tree, Random Forest, Logistic Regression, and SVC.
3. Among the models examined, the Random Forest Classifier had the most accuracy in the project.
4. Based on current data, model can be built on 4-5 important features for future prospects.
5. The UI based application can be used by the bank to predict if a loan application should be approved or not.
6. Optimal hyper parameters can be found to improve the model.
7. With more data, Neural Networks can be used.
8. Feature importance refers to techniques that assign a score to input features based on how useful they are at predicting a target variable.