IJMRSET

e-ISSN:2582-7219

# INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH IN SCIENCE, ENGINEERING AND TECHNOLOGY

Volume 6, Issue 12, December 2023

ISSN
INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 7.54

☎ 6381 907 438 ☎ 6381 907 438 ✉ ijmrset@gmail.com @ www.ijmrset.com

International Journal Of Multidisciplinary Research In Science, Engineering and Technology (IJMRSET)

| ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 7.54| Monthly Peer Reviewed & Referred Journal |

| Volume 6, Issue 12, December 2023 |

| DOI:10.15680/IJMRSET.2023.0612030 |

# Enhancing Bank Loan Approval with Predictive Analytic Models

**Prof.Pranali Vhora[1], Prof. Minakshi Retharekar[2] , Prof. Veena Badgujar[3]**

Assistant Professor, Shah and Anchor Kutchhi Engineering College, Chembur, India[1]

Assistant Professor, Shah and Anchor Kutchhi Engineering College, Chembur, India[2]

Assistant Professor, K.J.Somaiya College of Engineering, Vidyavihar, India[3]

**ABSTRACT:** In the dynamic landscape of banking, where diverse products contribute to revenue streams, the primary income source for banks often lies in their credit lines. Profits and losses of a bank hinge significantly on the repayment behavior of loans extended to customers. The identification and prediction of potential loan defaulters are crucial for mitigating Non-Performing Assets (NPAs) and optimizing profitability. This study delves into the various methodologies employed to address the challenge of controlling loan defaults, emphasizing the importance of accurate predictions for profit maximization.Drawing on previous research, this paper focuses on predictive analytics, specifically employing the Logistic Regression model,Linear Regression,naive bayes and decision Tree to study and predict loan defaulters. The dataset used for analysis is sourced from Kaggle, offering a comprehensive foundation for evaluating the effectiveness of the model. The study involves the execution of Logistic Regression models, and performance measures such as sensitivity and specificity are computed to compare their efficacy.

## I.INTRODUCTION

Loan distribution stands as a pivotal revenue stream for many banks, with a significant portion of their income derived from the interest on loans extended to customers. Despite rigorous verification and validation processes, banks often face uncertainties regarding the safety of customers chosen for loan applications. To address this challenge, we propose the implementation of a Loan Prediction System using Python, aiming to enhance the accuracy of loan approval decisions.The primary objective of this system is to evaluate the eligibility of customers for loans by assessing various parameters, including marital status, income, expenditure, and other relevant factors. Leveraging a trained dataset, the system employs a regression process to build a predictive model. This model is then applied to a test dataset to generate outputs indicating whether a particular customer is deemed capable of repaying a loan ("yes") or not ("no").

The proposed Loan Prediction System serves as a valuable tool for banks to make informed decisions in the loan approval process. By automating the evaluation of multiple factors and leveraging predictive analytics, the system enhances the accuracy of identifying safe customers for loan applications. The outcomes of this system, presented as binary results (yes or no), provide a clear basis for approving or denying loans to individual customers.This research contributes to the ongoing efforts of banks to invest in safe customers, emphasizing the importance of leveraging technology, specifically Python programming, to streamline and improve the loan approval process. The implementation of such a system not only enhances the efficiency of loan distribution but also contributes to the overall risk management strategy of banks.

In the contemporary financial landscape, the multifaceted risks associated with bank loans necessitate a meticulous analysis before loan authorization. Banks play a pivotal role in a market economy, and their ability to assess credit risk profoundly influences the success of businesses. Credit risk assessment involves a critical determination by banks to categorize borrowers into either good (non-defaulter) or bad (defaulter). Predicting the future status of borrowers, particularly whether they will default on a loan, presents a formidable challenge for financial institutions.The essence of predicting loan defaulters lies in addressing a binary classification problem, where the customer's credit history influences the magnitude of the loan. This paper explores the complexities associated with developing a predictive model for classifying borrowers into defaulter or non-defaulter categories. The task is inherently challenging due to the burgeoning demand for loans in today's dynamic financial environment.This study delves into the critical aspects of credit risk prediction, emphasizing the significance of accurate classification in determining loan sizes. The challenges posed by the evolving loan landscape are explored, and the paper underscores the intricate nature of developing a robust model capable of effectively categorizing borrowers. As financial institutions grapple with the increasing demands for loans, the research presented herein aims to contribute valuable insights and strategies for enhancing the efficiency of risk assessment in bank loans.

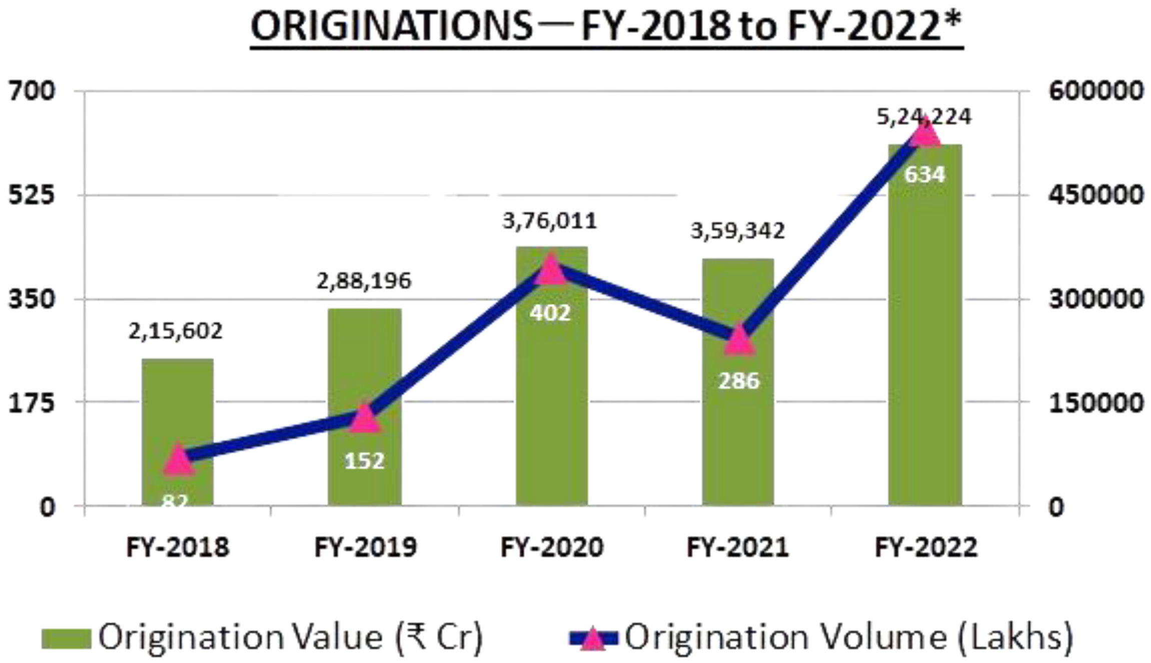


**Figure 1: Loan Survey 2018-2022**

The research utilizes a comprehensive dataset encompassing diverse customer attributes such as income, credit history, marital status, and other relevant factors. Each machine learning algorithm is implemented and fine-tuned to optimize its performance in predicting loan approval outcomes. The study analyzes the strengths and limitations of Logistic Regression, Linear Regression, Decision Tree, and Naive Bayes models in the context of their application to the loan approval process.Performance metrics, including accuracy, precision, recall, and F1 score, are computed to assess the effectiveness of each algorithm. The comparative analysis aims to identify the machine learning tool that yields the highest accuracy in predicting whether a loan should be approved or not.The results of this study provide valuable insights into the strengths and weaknesses of different machine learning approaches in the context of loan approval prediction. By leveraging the predictive power of these algorithms, banks and financial institutions can enhance their decision-making processes, minimize risks associated with loan approvals, and optimize their overall lending strategies.

## II.LITERATURE REVIEW

A thing that is borrowed, especially a sum of money that is expected to be paid back with interest.A loan is the lending of money by one or more individuals, organizations, or other entities to other individuals, organizations etc.A loan is money, property, or other material goods given to another party in exchange for future repayment of the loan value or principal amount, along with interest or finance charges

.

**Mohammad Ahmad Sheikh and Amit Kumar Goel and Tapas Kumar proposed An Approach for Prediction of Loan Approval using Machine Learning Algorithm[1].**

It appears that they have described a typical data analysis and predictive modeling workflow, focusing on the prediction of loan approval based on various features in the dataset. This involves handling missing values, removing outliers, and transforming variables to prepare the data for analysis.Filling in missing data points with estimated values to ensure completeness of the dataset.Exploratory data analysis to understand the distribution of variables, identify patterns, and gain insights into the characteristics of the dataset.Developing a predictive model, possibly using machine learning algorithms, to capture the relationships between the features and the target variable (loan approval).Assessing the performance of the model using metrics like accuracy, precision, recall, or F1 score to understand how well it predicts loan approval.Applying the trained model to a separate set of data (test data) to evaluate its generalization performance.The best-case accuracy obtained on the original dataset is 0.811, indicating that the model performs reasonably well.Applicants with a poor credit score are more likely to fail in obtaining loan approval, possibly due to a higher probability of defaulting on loan payments.High-income applicants requesting lower loan amounts are more likely to get approved, suggesting a correlation between income and loan approval likelihood.Gender and marital status do not seem to be significant factors considered by the company in the loan approval process.

**P. L. Srinivasa Murthy and G. Soma Shekar and P. Rohith and G. Vishnu Vardhan Reddy proposed a Loan Approval Prediction System Using Machine Learning.[2]**

It's great to hear that the system they have described appears to be effective for loan approval prediction. Your positive assessment and future considerations suggest a robust and scalable solution.The system is versatile and can be used for various banking requirements, indicating its potential applicability across different scenarios within the financial domain.The system is easily uploadable to any operating system, ensuring flexibility and accessibility.Given the increasing trend towards online technologies, the system is well-positioned for the future, aligning with the evolving landscape of digital banking.Emphasis on the system's security and reliability suggests a trustworthy and robust solution, which is crucial in financial applications.The use of the Random Forest algorithm is highlighted for its accuracy in providing reliable results. This choice implies that the model can handle complex relationships within the data.The system is designed to handle a large number of customers applying for loans, indicating scalability and efficiency.Future plans include the addition of more algorithms to further enhance the system's accuracy. This commitment to continuous improvement aligns with best practices in data science and machine learning.It's important to continue monitoring and refining the system over time, considering changes in data patterns and the evolving landscape of banking and technology.

**Anant Shinde and Yash Patil and Ishan Kotian and Abhinav Shinde and Reshma Gulwani proposed a Loan Prediction System Using Machine Learning[3]**

The above research employs a logistic regression algorithm-based prediction model. To create a logistic classification model that predicts loan status,over 600 sample data were collected and evaluated.The goal is to create a logistic classification model that predicts the loan status based on a loan application.Over 600 sample data points were collected and evaluated. This dataset likely includes various features or variables related to loan applications, such as income, credit score, debt-to-income ratio, employment status, etc.The logistic regression algorithm is employed for the prediction model. Logistic regression is a statistical method used for binary classification problems, such as predicting whether a loan will be approved or not.The model claims a maximum accuracy of about 82 percent. Accuracy is a measure of how well the model predicts the correct outcome. In this case, it suggests that the model is reasonably effective, but further evaluation (e.g., precision, recall) might provide a more comprehensive understanding of its performance.The mention of regression models may indicate that additional regression analyses were conducted, possibly to fine-tune or improve the predictive accuracy of the model.The model is described as quickly adaptable to a wide range of inputs. This adaptability is a valuable characteristic, allowing the model to handle diverse types of loan applications and applicant profiles.The strategy is noted to save a significant amount of time for the banking industry and its staff..

**Ms. Kathe Rutika Pramod and Ms. Panhale Sakshi Dattatray and Ms. Avhad Pooja Prakash and Ms. Dapse Punam Laxman and Mr. Ghorpade Dinesh B. proposed An Approach For Prediction Of Loan Approval Using Machine Learning Algorithm[4]**

It appears that they are describing the analytical process and insights gained from a data analysis project, possibly related to loan approval. The process started with data cleaning and processing, indicating the initial steps involved in preparing the dataset for analysis.Missing value imputation was performed using the "mice" package, suggesting a method for handling incomplete data.Exploratory analysis was conducted, likely involving visualizations and statistical summaries to understand the characteristics of the data.The final step involved model building and evaluation, where a predictive model was developed and assessed.The best accuracy achieved on the public test set is mentioned to be 0.811. This represents the model's ability to correctly predict outcomes and suggests a reasonably good performance.Applicants with a credit history not passing are mentioned to be less likely to get approved. This makes sense, as credit history is a crucial factor in assessing an individual's creditworthiness and likelihood of repayment.The insight that applicants with high income but seeking a low loan amount are more likely to get approved aligns with the idea that individuals with greater financial stability may pose lower risk for lenders.The observation that some basic characteristics like gender and marital status don't seem to influence the approval decision implies that the company focuses more on other factors during the approval process.The insights drawn from the analysis provide a better understanding of the factors influencing loan approval decisions

**Miraz Al Mamun and Afia Farjana and Muntasir Mamun proposed Predicting Bank Loan Eligibility Using Machine Learning Models and Comparison Analysis.[5]**

In this paper they are highlighting the importance of leveraging technology, specifically predictive models, in the fast-growing IT sector, particularly in the context of loan approval processes within the banking system.The rapidly growing IT sector demands continuous development of new technologies and updates to existing ones. The goal is to reduce human interference and enhance job productivity.The predictive model is applied to the banking system, specifically for anyone applying for a loan. The objective is to streamline and optimize the loan approval process.The model, through data examination, is claimed to reduce fraud during the loan approval process. This reduction not only benefits the bank but also minimizes waiting time for applicants.The prediction process involves several steps, including data cleaning and processing, imputation of missing values, exploratory analysis of the dataset, model construction, and testing on a separate dataset.The best-case accuracy achieved on the original dataset is reported to be 0.9189. This indicates a high level of accuracy in predicting loan outcomes.Applicants with the lowest credit scores are likely to be denied a loan due to a higher risk of default.High-income applicants seeking smaller loans are more likely to be approved, reflecting a logical expectation of their ability to repay.Factors like gender and marital status do not seem to influence the corporation's loan approval decisions.

| No. | Paper Title | Author Name | Key Points | Remark |
|---|---|---|---|---|
| 1 | An Approach for Prediction of Loan Approval using Machine Learning Algorithm | Mohammad Ahmad Sheikh and Amit Kumar Goel and Tapas Kumar,2020 | The model concludes that a bank should not only target the rich customers for granting loan but it should assess the other attributes of a customer as well which play a very important part in credit granting decisions and predicting the loan defaulters.[1] | Improves the performance of database cleaning. |
| 2 | Loan Approval Prediction System Using Machine Learning | P. L. Srinivasa Murthy and G. Soma Shekar and P. Rohith and G. Vishnu Vardhan | Trained Dataset, Random Forests, Bank Loans, Safe Customers.use random forest algorithm for the classification of data. Random forests algorithm builds a model from a trained dataset and this | This system accepts data for N no. of customers. In future they can add more algorithms to this system for getting more accurate results. |

| | | | | |
|---|---|---|---|---|
| | | Reddy,2020 | model is applied on test data and we get the required output.[2] | |
| 3 | Loan Prediction System Using Machine Learning | Anant Shinde and Yash Patil and Ishan Kotian and Abhinav Shinde and Reshma Gulwani,2022 | The machine learning approach is ideal for reducing human effort and effective decision making in the loan approval process by implementing machine learning tools that use classification algorithms to predict eligible loan applicants.[3] | They are required to compare with other models also. |
| 4 | An Approach For Prediction Of Loan Approval Using Machine Learning Algorithm | Ms. Kathe Rutika Pramod and Ms. Panhale Sakshi Dattatray and Ms. Avhad Pooja Prakash and Ms. Dapse Punam Laxman and Mr. Ghorpade Dinesh B.,2021 | The analytical process started from data cleaning and processing, Missing value imputation with mice package, then exploratory analysis and finally model building and evaluation. The best accuracy on the public test set is 0.811. This brings some of the following insights about approval.[4] | Virtual machines that are present on a physical system or running on a portable storage device can be detected or analyzed. |
| 5 | Predicting Bank Loan Eligibility Using Machine Learning Models and Comparison Analysis | Miraz Al Mamun and Afia Farjana and Muntasir Mamun,2022 | future. The system is based on the previous training data but in the future, it is possible to make changes to software, which can accept new testing data and should also take part in training data and predict accordingly.[5]. | To design and implement the system using machine learning and data mining to predict the probability of the user to get loan or not from bank to improve the accuracy and to minimize the frauds., |

In summary, the work presented in this paper is built on previous research to explore how data represent people's trust. and how we can trust on bank using the different methods for approval and for bank also be aware from fraud customers

## III.METHODOLOGY OF PROPOSED SURVEY

A Company wants to automate the loan eligibility process (real time) based on customer detail provided while filling online application form.To automate this process, they have given a problem to identify the customers' segments, those are eligible for loan amount so that they can specifically target these customers.These details are Gender, Marital Status, Education, Number of Dependents, Income, Loan Amount, Credit History and others.The first thing we need to do and before jumping to analyse the data is to understand the problem statement.The next step is to identify our independent variables and our dependent variables. The below mind map illustrates the process I have conducted to structure and plan the project.

The next step is to look at the data we're working with.Normally, we must answer the following questions:

1) What do we want exactly from data?

2) What is missing in data and what we have to remove from data?

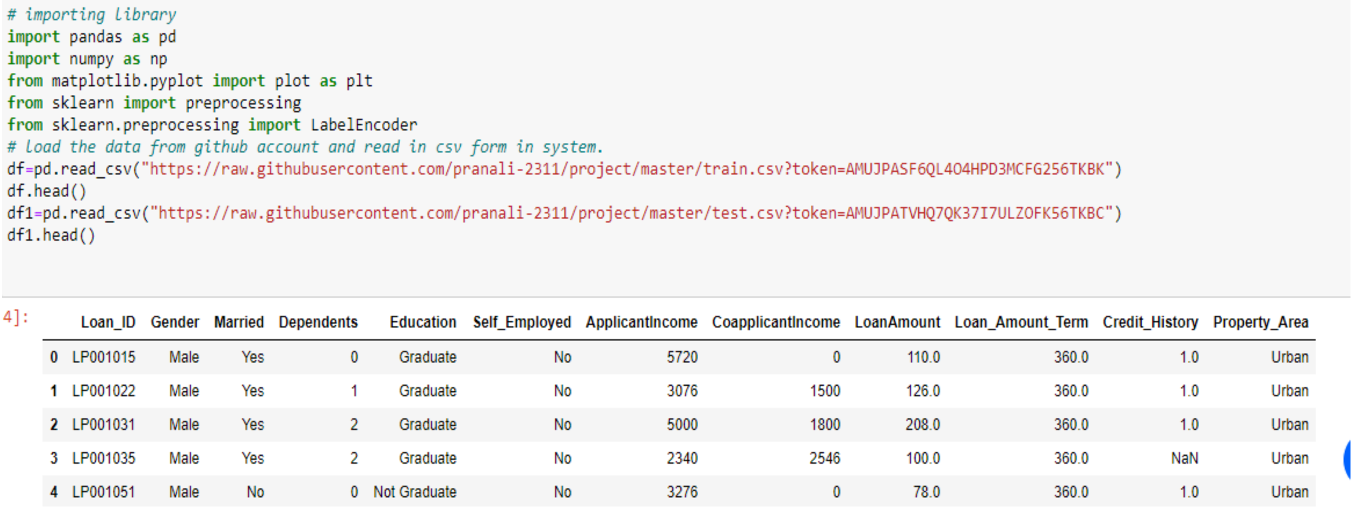
3) Which prediction is used and which is suitable for us?

4) What is accuracy from our result and compare with which algorithm?

The goal is to predict whether a bank should grant a loan to customers. This is a classification problem, where the model aims to categorize loan applications into approved or denied classes.The described approach follows a systematic process for developing a predictive model for loan approval using logistic regression. Each step, from preprocessing to model evaluation, contributes to creating a robust and accurate model for predicting whether a bank should grant a loan to its customers.

## A. DATA COLLECTION:-

We collect the data from kaggle.It's common to use Kaggle as a data source, given its popularity for providing datasets for educational and learning purposes.There are two datasets obtained from Kaggle: one for training the model and another for testing the model.The training dataset is used to train the predictive model. It's common to split this dataset into two parts, such as an 80:20 or 70:30 ratio. The major portion of the dataset (80% or 70%) is used for training the model.The testing dataset is used to evaluate the performance of the trained model. The remaining portion of the dataset (20% or 30%) is used to assess how well the model generalizes to new, unseen data.The major part of the training dataset is used to teach the model to recognize patterns and make predictions. The model is then tested on the separate testing dataset to measure its accuracy and performance.The accuracy of the developed model is calculated based on its performance on the testing dataset. Accuracy is a common metric used to evaluate classification models, indicating the proportion of correctly predicted instances.



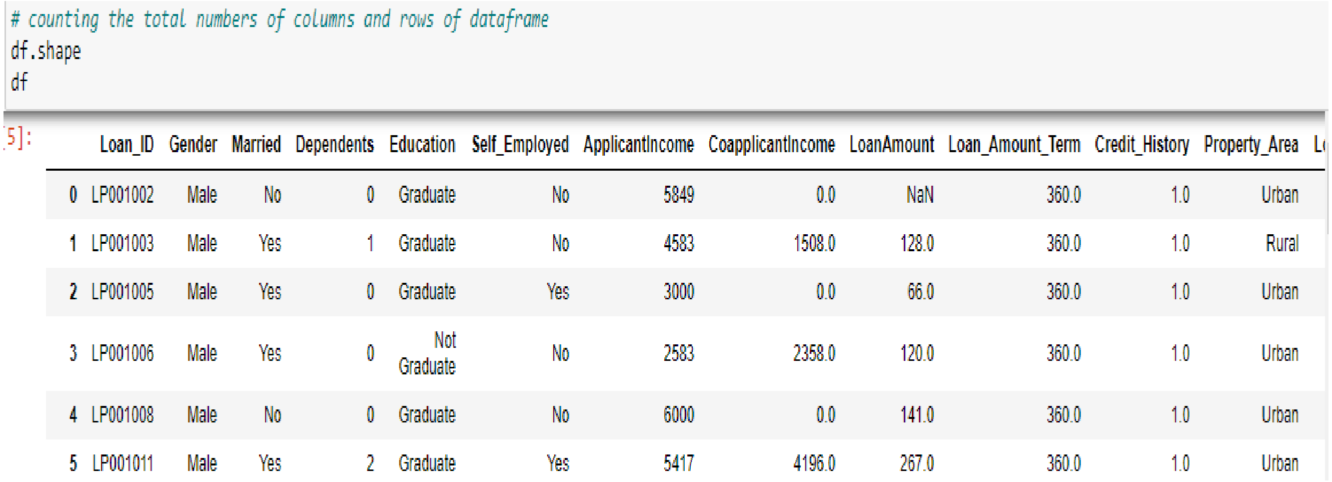
```
# importing library
import pandas as pd
import numpy as np
from matplotlib.pyplot import plot as plt
from sklearn import preprocessing
from sklearn.preprocessing import LabelEncoder
# load the data from github account and read in csv form in system.
df=pd.read_csv("https://raw.githubusercontent.com/pranali-2311/project/master/train.csv?token=AMUJPASF6QL4O4HPD3MCFG256TKBK")
df.head()
df1=pd.read_csv("https://raw.githubusercontent.com/pranali-2311/project/master/test.csv?token=AMUJPATVHQ7QK37I7ULZOFK56TKBC")
df1.head()
```

4]:

| | Loan_ID | Gender | Married | Dependents | Education | Self_Employed | ApplicantIncome | CoapplicantIncome | LoanAmount | Loan_Amount_Term | Credit_History | Property_Area |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | LP001015 | Male | Yes | 0 | Graduate | No | 5720 | 0 | 110.0 | 360.0 | 1.0 | Urban |
| 1 | LP001022 | Male | Yes | 1 | Graduate | No | 3076 | 1500 | 126.0 | 360.0 | 1.0 | Urban |
| 2 | LP001031 | Male | Yes | 2 | Graduate | No | 5000 | 1800 | 208.0 | 360.0 | 1.0 | Urban |
| 3 | LP001035 | Male | Yes | 2 | Graduate | No | 2340 | 2546 | 100.0 | 360.0 | NaN | Urban |
| 4 | LP001051 | Male | No | 0 | Not Graduate | No | 3276 | 0 | 78.0 | 360.0 | 1.0 | Urban |

## B. DATA PREPROCESSING:-

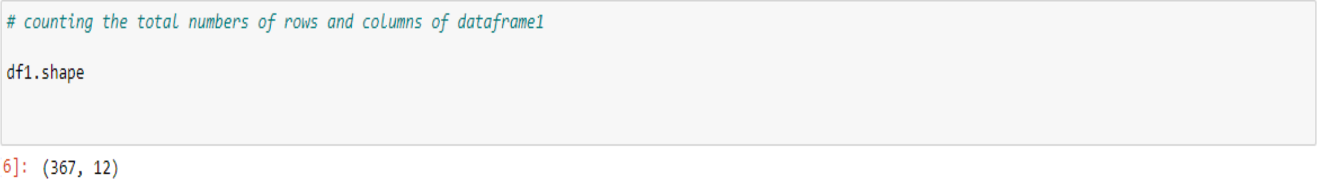
We are describing the use of data mining techniques in the preprocessing phase of handling data collected through online forms.Raw data is collected using online forms. This data may be unstructured, containing irrelevant, missing, or noisy information.Data mining techniques are employed in the preprocessing stage to transform raw data into useful and efficient formats. This is crucial as it helps in handling issues like irrelevant information, missing data, and noise.Data cleaning techniques are applied to address issues such as irrelevant, missing, or noisy data. This step is essential for improving data quality before further analysis.Data reduction techniques are used to handle large volumes of data before data mining. This is done to make data analysis more manageable and efficient, aiming to achieve accurate results.Data reduction helps increase data storage capacity and reduces the cost of data analysisEncoding mechanisms are employed to reduce the size of data. This reduction can be either lossy or lossless.Lossless reduction implies that the original data can be obtained after reconstruction from the compressed data, while lossy reduction means some information is lost in the compression process.Wavelet transforms and PCA are specific techniques mentioned for data reduction.Wavelet transforms are mathematical transformations that can be used to represent data in a more compact form, particularly effective for signal and image processing.PCA is a statistical method that transforms high-dimensional data into a lower-dimensional form while preserving the most important information.



```
# counting the total numbers of columns and rows of dataframe
df.shape
df
```

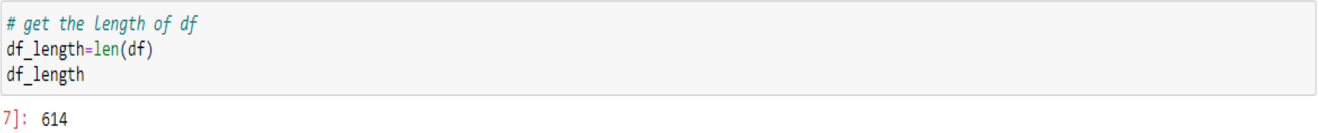
5]:

| | Loan_ID | Gender | Married | Dependents | Education | Self_Employed | ApplicantIncome | CoapplicantIncome | LoanAmount | Loan_Amount_Term | Credit_History | Property_Area | L |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | LP001002 | Male | No | 0 | Graduate | No | 5849 | 0.0 | NaN | 360.0 | 1.0 | Urban | |
| 1 | LP001003 | Male | Yes | 1 | Graduate | No | 4583 | 1508.0 | 128.0 | 360.0 | 1.0 | Rural | |
| 2 | LP001005 | Male | Yes | 0 | Graduate | Yes | 3000 | 0.0 | 66.0 | 360.0 | 1.0 | Urban | |
| 3 | LP001006 | Male | Yes | 0 | Not Graduate | No | 2583 | 2358.0 | 120.0 | 360.0 | 1.0 | Urban | |
| 4 | LP001008 | Male | No | 0 | Graduate | No | 6000 | 0.0 | 141.0 | 360.0 | 1.0 | Urban | |
| 5 | LP001011 | Male | Yes | 2 | Graduate | Yes | 5417 | 4196.0 | 267.0 | 360.0 | 1.0 | Urban | |



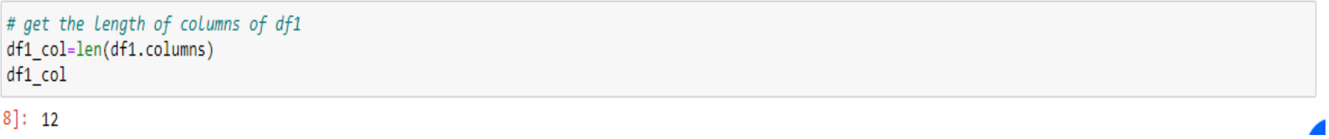
```
# counting the total numbers of rows and columns of dataframe1

df1.shape
```

```
6]: (367, 12)
```



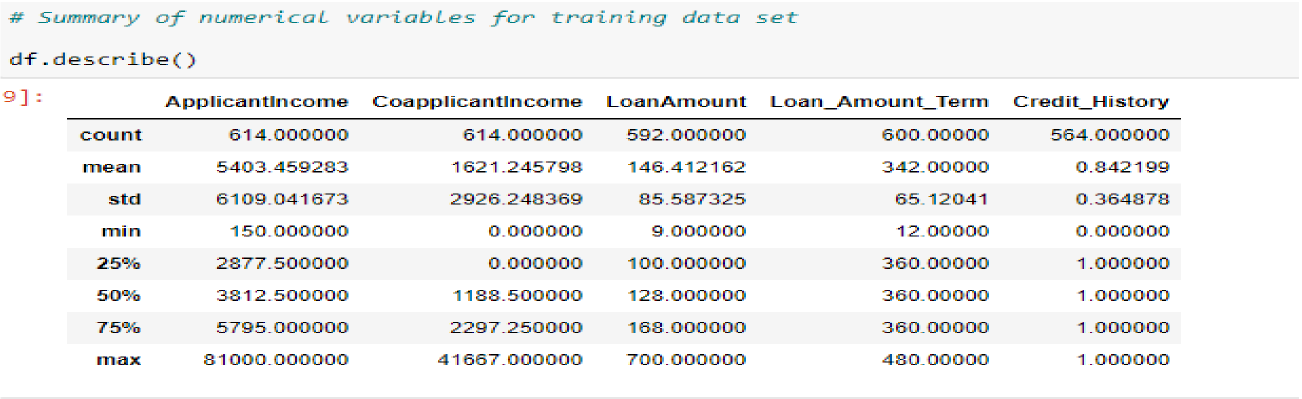
```
# get the length of df
df_length=len(df)
df_length
```

```
7]: 614
```



```
# get the length of columns of df1
df1_col=len(df1.columns)
df1_col
```

```
8]: 12
```



```
# Summary of numerical variables for training data set

df.describe()
```

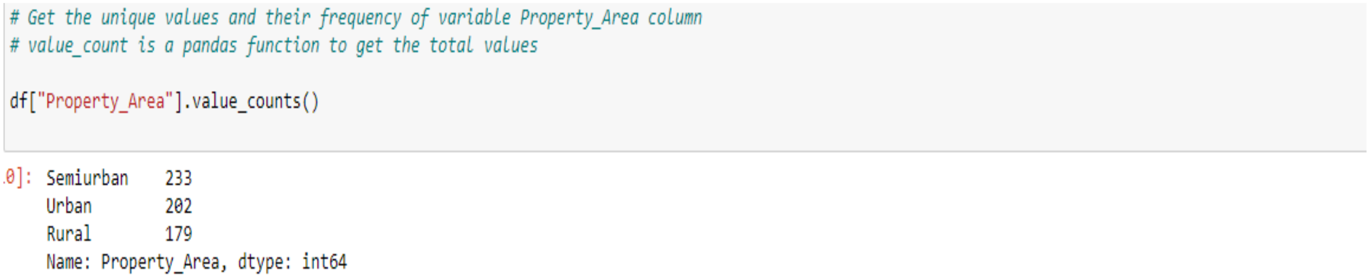
9]:

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

| ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 7.54| Monthly Peer Reviewed & Referred Journal |

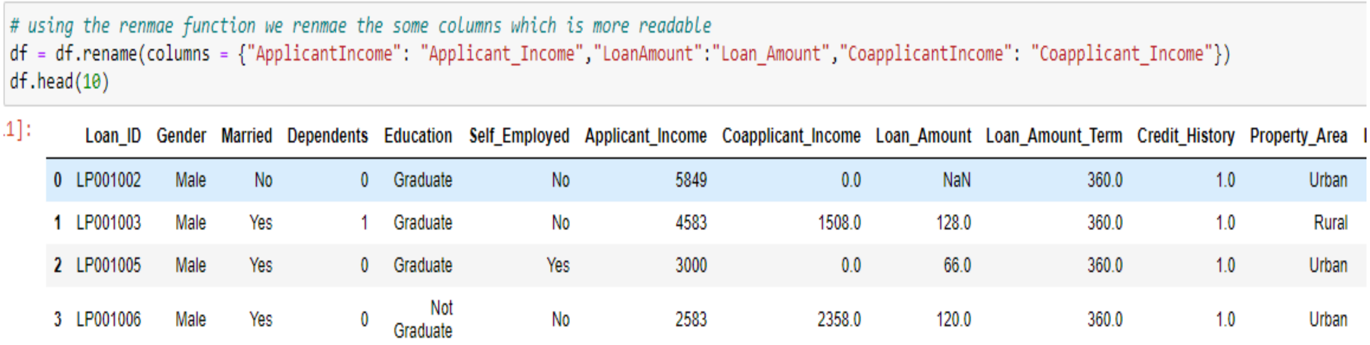
| Volume 6, Issue 12, December 2023 |

| DOI:10.15680/IJMRSET.2023.0612030 |



```
# Get the unique values and their frequency of variable Property_Area column
# value_count is a pandas function to get the total values

df["Property_Area"].value_counts()
```

```
.0]: Semiurban    233
     Urban        202
     Rural        179
     Name: Property_Area, dtype: int64
```



```
# using the renmae function we renmae the some columns which is more readable
df = df.rename(columns = {"ApplicantIncome": "Applicant_Income","LoanAmount":"Loan_Amount","CoapplicantIncome": "Coapplicant_Income"})
df.head(10)
```

.1]:

| | Loan_ID | Gender | Married | Dependents | Education | Self_Employed | Applicant_Income | Coapplicant_Income | Loan_Amount | Loan_Amount_Term | Credit_History | Property_Area |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | LP001002 | Male | No | 0 | Graduate | No | 5849 | 0.0 | NaN | 360.0 | 1.0 | Urban |
| 1 | LP001003 | Male | Yes | 1 | Graduate | No | 4583 | 1508.0 | 128.0 | 360.0 | 1.0 | Rural |
| 2 | LP001005 | Male | Yes | 0 | Graduate | Yes | 3000 | 0.0 | 66.0 | 360.0 | 1.0 | Urban |
| 3 | LP001006 | Male | Yes | 0 | Not Graduate | No | 2583 | 2358.0 | 120.0 | 360.0 | 1.0 | Urban |

## C.FEATURE EXTRACTION:-

Feature engineering involves the creation, modification, or selection of features in a dataset to enhance the performance of machine learning models.It helps in preparing a proper input dataset that is compatible with the requirements of machine learning algorithms.

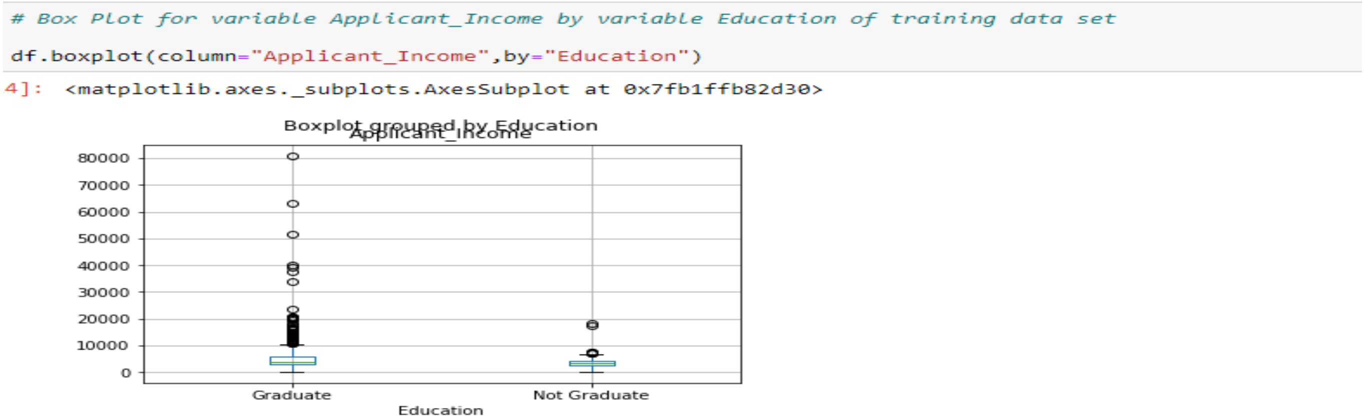
```
import pandas as pd
import numpy as np
```

Pandas is used for data manipulation and analysis, providing data structures like DataFrame that are particularly useful for handling structured data.

NumPy is used for numerical operations and provides support for large, multi-dimensional arrays and matrices.

## D. LIST OF TECHNIQUES:-

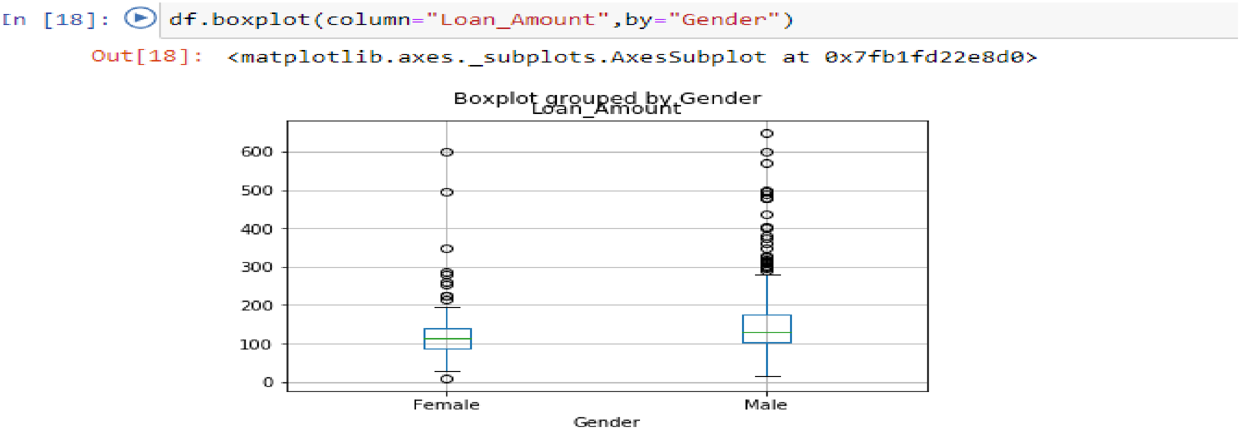
### 1.1 Handling outliers:-



```
# Box Plot for variable Applicant_Income by variable Education of training data set
df.boxplot(column="Applicant_Income",by="Education")
4]:  <matplotlib.axes._subplots.AxesSubplot at 0x7fb1ffb82d30>
```

We can see that there is no substantial difference between the mean income of graduates and non-graduates. But there are a higher number of graduates with very high incomes, which are appearing to be the outliers

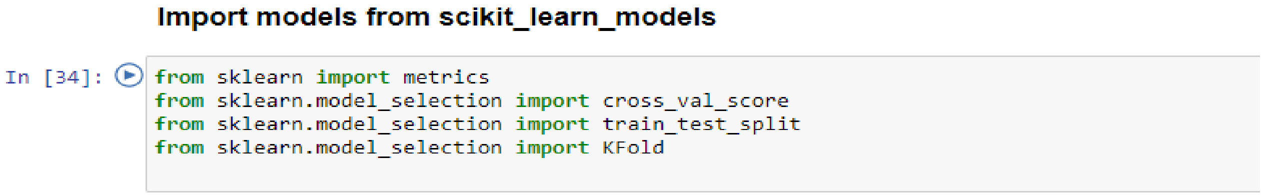
Boxplot on Loan_Amount by Gender for comparison



```
In [18]:  df.boxplot(column="Loan_Amount",by="Gender")
   Out[18]:  <matplotlib.axes._subplots.AxesSubplot at 0x7fb1fd22e8d0>
```

### E.MODEL SELECTION:-

Import models from scikit_learn_models



**Import models from scikit_learn_models**

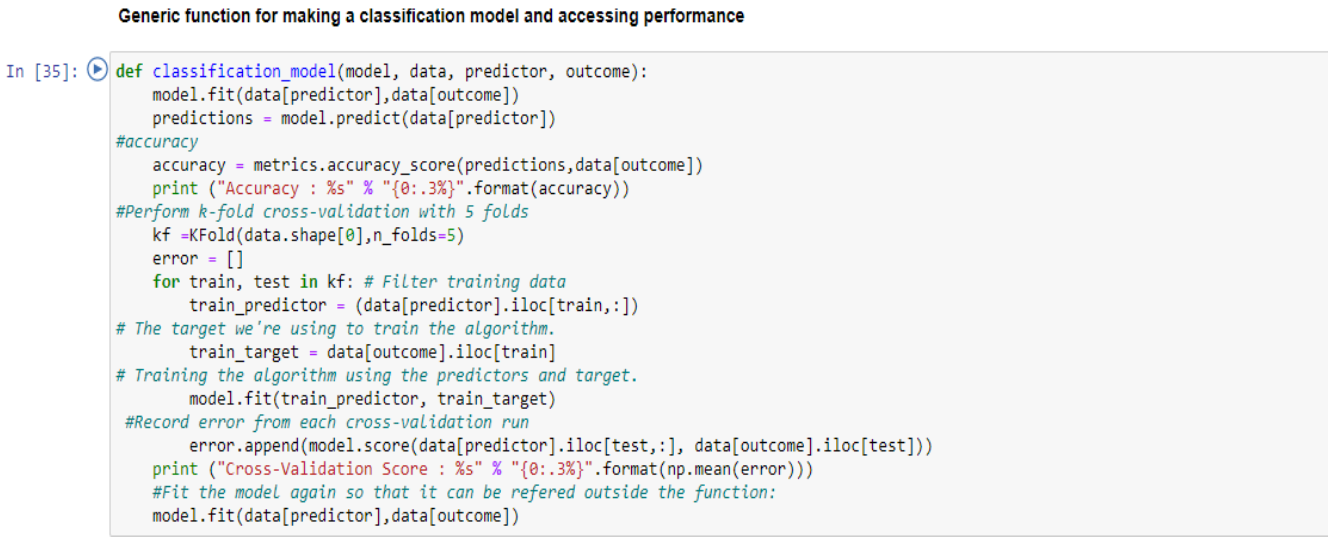
```
In [34]:  from sklearn import metrics
          from sklearn.model_selection import cross_val_score
          from sklearn.model_selection import train_test_split
          from sklearn.model_selection import KFold
```

Generic function for making a classification model and accessing performance

| ISSN: 2582-7219 | www.ijmrset.com | Impact Factor: 7.54| Monthly Peer Reviewed & Referred Journal |

| Volume 6, Issue 12, December 2023 |

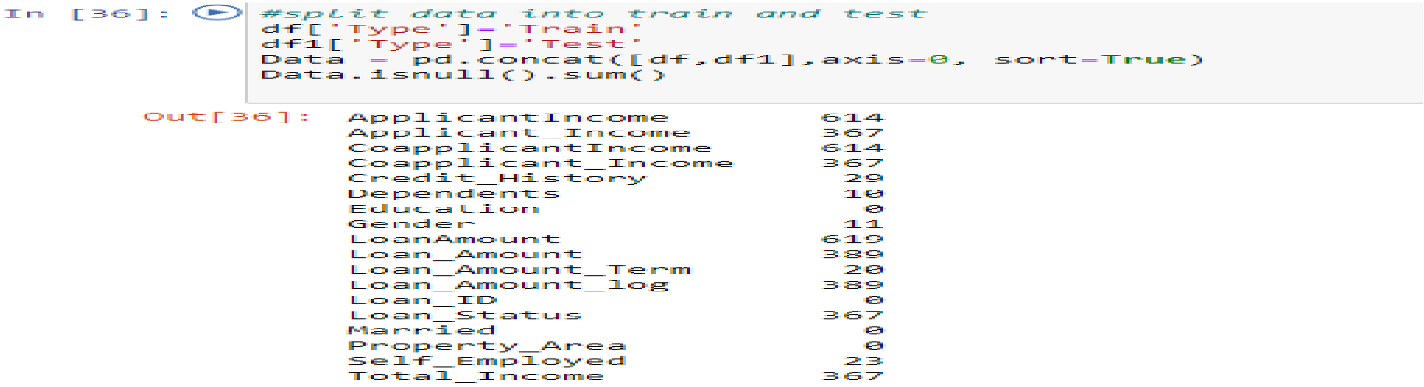
| DOI:10.15680/IJMRSET.2023.0612030 |



Generic function for making a classification model and accessing performance

```
In [35]: def classification_model(model, data, predictor, outcome):
             model.fit(data[predictor],data[outcome])
             predictions = model.predict(data[predictor])
         #accuracy
             accuracy = metrics.accuracy_score(predictions,data[outcome])
             print ("Accuracy : %s" % "{0:.3%}".format(accuracy))
         #Perform k-fold cross-validation with 5 folds
             kf =KFold(data.shape[0],n_folds=5)
             error = []
             for train, test in kf: # Filter training data
                 train_predictor = (data[predictor].iloc[train,:])
         # The target we're using to train the algorithm.
                 train_target = data[outcome].iloc[train]
         # Training the algorithm using the predictors and target.
                 model.fit(train_predictor, train_target)
          #Record error from each cross-validation run
                 error.append(model.score(data[predictor].iloc[test,:], data[outcome].iloc[test]))
             print ("Cross-Validation Score : %s" % "{0:.3%}".format(np.mean(error)))
             #Fit the model again so that it can be refered outside the function:
             model.fit(data[predictor],data[outcome])
```

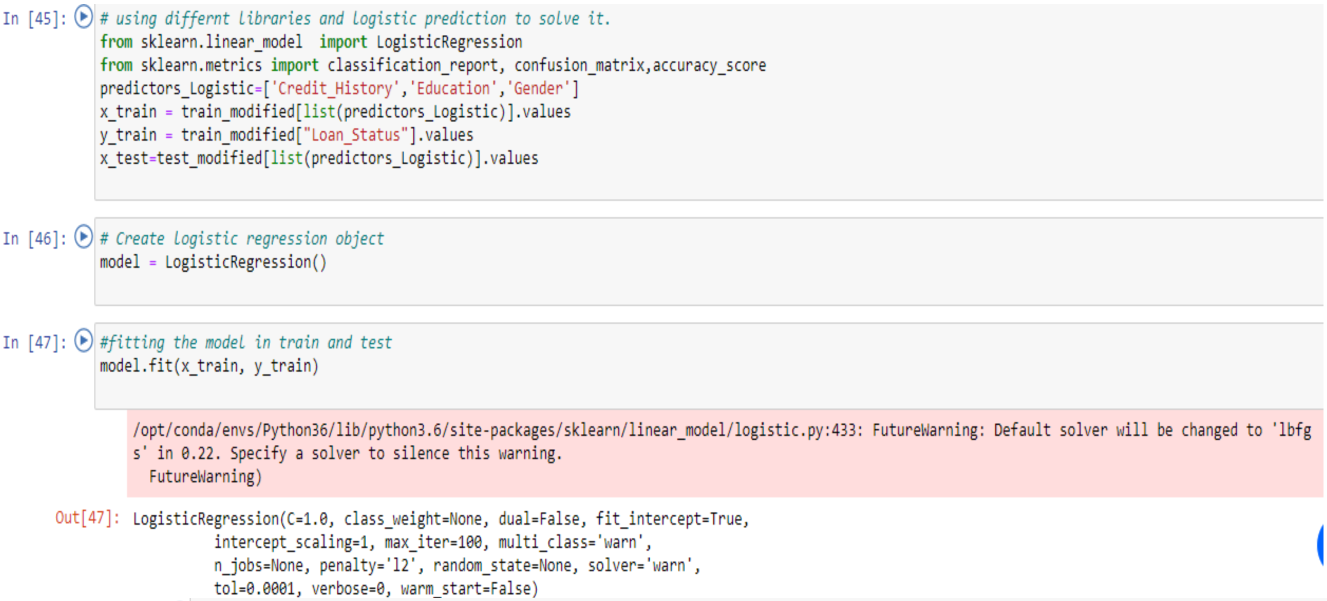
Split data into train and test



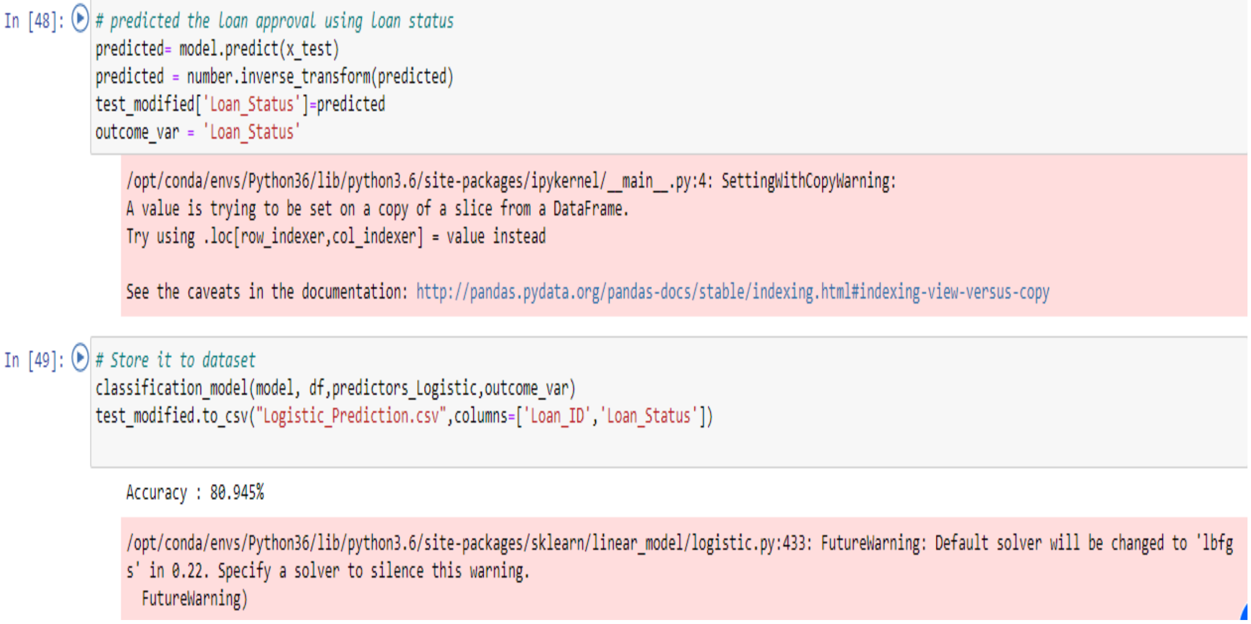
```
In [36]: #split data into train and test
         df['Type']='Train'
         df1['Type']='Test'
         Data = pd.concat([df,df1],axis=0, sort=True)
         Data.isnull().sum()

Out[36]: ApplicantIncome          614
         Applicant_Income         367
         CoapplicantIncome        614
         Coapplicant_Income       367
         Credit_History            29
         Dependents                10
         Education                  0
         Gender                    11
         LoanAmount               619
         Loan_Amount              389
         Loan_Amount_Term          20
         Loan_Amount_log          389
         Loan_ID                    0
         Loan_Status              367
         Married                    0
         Property_Area              0
         Self_Employed             23
         Total_Income             367
```

## 1)LOGISTIC REGRESSION:-

The chances of getting a loan will be higher for*:*
Applicants having a credit history (we observed this in exploration.)Applicants with higher applicant and co-applicant incomes.Applicants with higher education level.Properties in urban areas with high growth perspectivesSo let’s make our model with ‘Credit_History’, ’Education’ & ’Gender’Using different libraries and logistic prediction to solve it



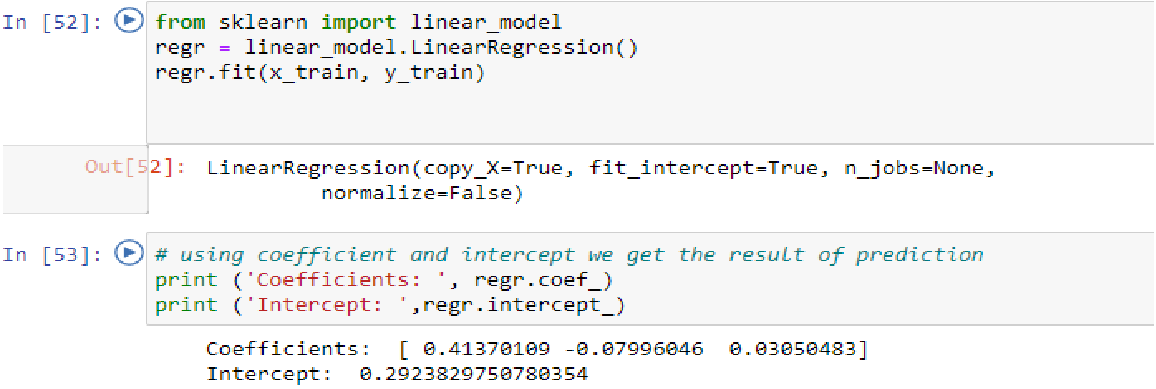
```
In [45]: # using differnt libraries and logistic prediction to solve it.
         from sklearn.linear_model  import LogisticRegression
         from sklearn.metrics import classification_report, confusion_matrix,accuracy_score
         predictors_Logistic=['Credit_History','Education','Gender']
         x_train = train_modified[list(predictors_Logistic)].values
         y_train = train_modified["Loan_Status"].values
         x_test=test_modified[list(predictors_Logistic)].values

In [46]: # Create logistic regression object
         model = LogisticRegression()

In [47]: #fitting the model in train and test
         model.fit(x_train, y_train)

           /opt/conda/envs/Python36/lib/python3.6/site-packages/sklearn/linear_model/logistic.py:433: FutureWarning: Default solver will be changed to 'lbfg
           s' in 0.22. Specify a solver to silence this warning.
             FutureWarning)

Out[47]: LogisticRegression(C=1.0, class_weight=None, dual=False, fit_intercept=True,
                   intercept_scaling=1, max_iter=100, multi_class='warn',
                   n_jobs=None, penalty='l2', random_state=None, solver='warn',
                   tol=0.0001, verbose=0, warm_start=False)
```



```
In [48]: # predicted the loan approval using loan status
         predicted= model.predict(x_test)
         predicted = number.inverse_transform(predicted)
         test_modified['Loan_Status']=predicted
         outcome_var = 'Loan_Status'

         /opt/conda/envs/Python36/lib/python3.6/site-packages/ipykernel/__main__.py:4: SettingWithCopyWarning:
         A value is trying to be set on a copy of a slice from a DataFrame.
         Try using .loc[row_indexer,col_indexer] = value instead

         See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy

In [49]: # Store it to dataset
         classification_model(model, df,predictors_Logistic,outcome_var)
         test_modified.to_csv("Logistic_Prediction.csv",columns=['Loan_ID','Loan_Status'])

         Accuracy : 80.945%

         /opt/conda/envs/Python36/lib/python3.6/site-packages/sklearn/linear_model/logistic.py:433: FutureWarning: Default solver will be changed to 'lbfg
         s' in 0.22. Specify a solver to silence this warning.
           FutureWarning)
```

## 2) LINEAR REGRESSION:-

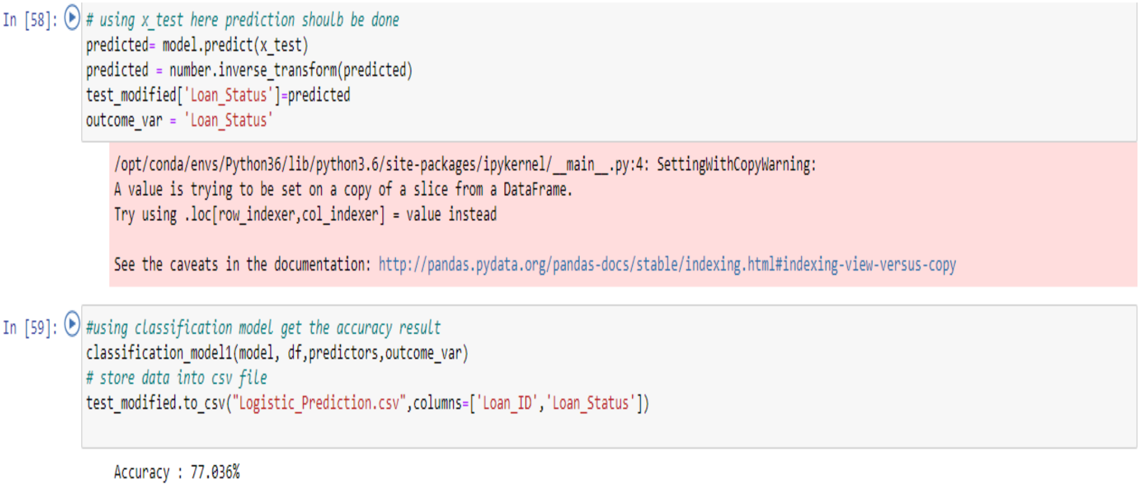
Linear Regression is a linear approach to modeling the relationship between a scalar response (or dependent variable) and one or more explanatory variables (or independent variables). In linear regression, the relationships are modeled using linear predictor functions whose unknown model parameters are estimated from the data.Linear regression was the first type of regression analysis to be studied rigorously, and to be used extensively in practical applications



```
In [52]: from sklearn import linear_model
         regr = linear_model.LinearRegression()
         regr.fit(x_train, y_train)

Out[52]: LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None,
                  normalize=False)

In [53]: # using coefficient and intercept we get the result of prediction
         print ('Coefficients: ', regr.coef_)
         print ('Intercept: ',regr.intercept_)

    Coefficients:  [ 0.41370109 -0.07996046  0.03050483]
    Intercept:  0.2923829750780354
```

### 3)NAIVE BAYES CLASSIFIER:-

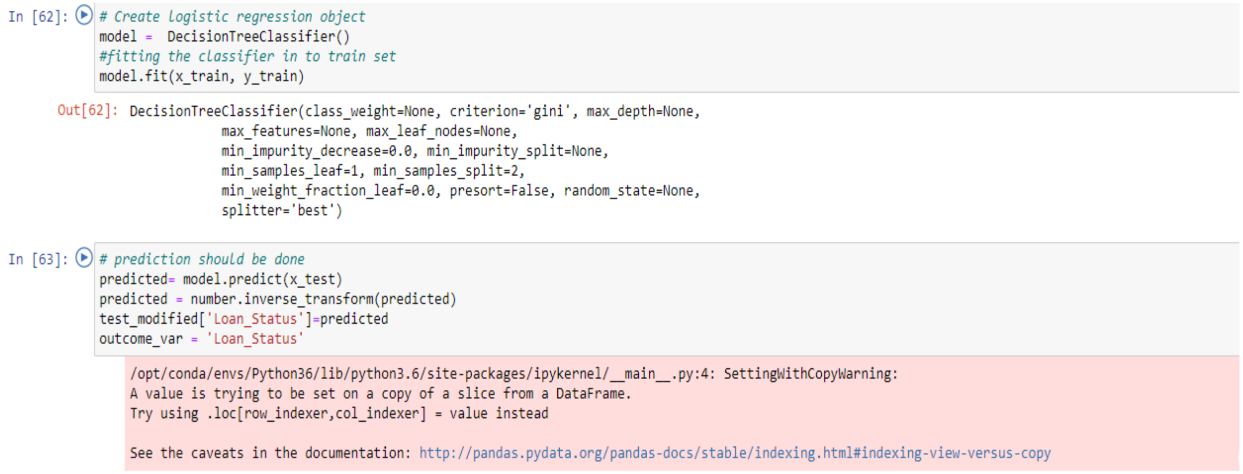
In machine learning, naive Bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes theorem with strong (naive) independence assumptions between the features. They are among the simplest Bayesian network models.Naive Bayes classifiers are highly scalable, requiring a number of parameters linear in the number of variables (features/predictors) in a learning problemMaximum-likelihood training can be done by evaluating a closed-form expression, which takes linear time, rather than by expensive iterative approximation as used for many other types of classifiers.In the statistics and computer science literature, naive Bayes models are known under a variety of names, including simple Bayes and independence BayesAll these names reference the use of Bayes' theorem in the classifier's decision rule, but naive Bayes is not (necessarily) a Bayesian method.



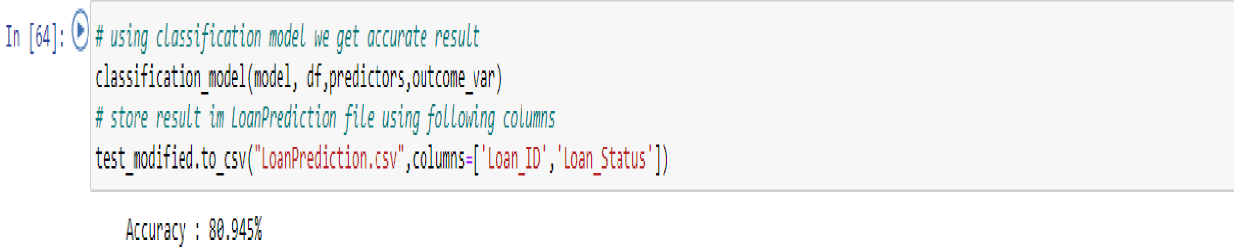
```
In [58]: # using x_test here prediction shoulb be done
         predicted= model.predict(x_test)
         predicted = number.inverse_transform(predicted)
         test_modified['Loan_Status']=predicted
         outcome_var = 'Loan_Status'

    /opt/conda/envs/Python36/lib/python3.6/site-packages/ipykernel/__main__.py:4: SettingWithCopyWarning:
    A value is trying to be set on a copy of a slice from a DataFrame.
    Try using .loc[row_indexer,col_indexer] = value instead

    See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy

In [59]: #using classification model get the accuracy result
         classification_model1(model, df,predictors,outcome_var)
         # store data into csv file
         test_modified.to_csv("Logistic_Prediction.csv",columns=['Loan_ID','Loan_Status'])

    Accuracy : 77.036%
```

### 4) DECISION TREE CLASSIFIER:-

In statistics, Decision tree learning uses a decision tree (as a predictive model) to go from observations about an item (represented in the branches) to conclusions about the item's target value (represented in the leaves)It is one of the predictive modelling approaches used in statistics, data mining and machine learning.Tree models where the target variable can take a discrete set of values are called classification trees; in these tree structures, leaves represent class labels and branches represent conjunctions of features that lead to those class labels.Decision trees where the target variable can take continuous values (typically real numbers) are called regression trees.



```
In [62]: # Create logistic regression object
         model =  DecisionTreeClassifier()
         #fitting the classifier in to train set
         model.fit(x_train, y_train)

Out[62]: DecisionTreeClassifier(class_weight=None, criterion='gini', max_depth=None,
                     max_features=None, max_leaf_nodes=None,
                     min_impurity_decrease=0.0, min_impurity_split=None,
                     min_samples_leaf=1, min_samples_split=2,
                     min_weight_fraction_leaf=0.0, presort=False, random_state=None,
                     splitter='best')

In [63]: # prediction should be done
         predicted= model.predict(x_test)
         predicted = number.inverse_transform(predicted)
         test_modified['Loan_Status']=predicted
         outcome_var = 'Loan_Status'

    /opt/conda/envs/Python36/lib/python3.6/site-packages/ipykernel/__main__.py:4: SettingWithCopyWarning:
    A value is trying to be set on a copy of a slice from a DataFrame.
    Try using .loc[row_indexer,col_indexer] = value instead

    See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/indexing.html#indexing-view-versus-copy
```



```
In [64]: # using classification model we get accurate result
         classification_model(model, df,predictors,outcome_var)
         # store result im LoanPrediction file using following columns
         test_modified.to_csv("LoanPrediction.csv",columns=['Loan_ID','Loan_Status'])

    Accuracy : 80.945%
```

## IV. CONCLUSION AND FUTURE WORK

In this paper, We collected over 600 sample data points to train and evaluate the model. This dataset likely included features related to loan applicants, such as income, credit score,Loan id,dependent, etc.The logistic regression algorithm achieved a maximum accuracy of about 82 percent. Accuracy is a common metric used to evaluate the performance of classification models, indicating the proportion of correctly predicted outcomes.The model is designed to anticipate outcomes, suggesting its ability to make predictions about whether a loan application is likely to be approved or denied.The strategy of using this logistic regression-based model is said to save the banking industry and its staff a significant amount of time. This implies that the model facilitates a more efficient and streamlined decision-making process compared to traditional methods.

## REFERENCES

[1] Mohammad Ahmad Sheikh and Amit Kumar Goel and Tapas Kumar,"An Approach for Prediction of Loan Approval using Machine Learning Algorithm" of the International Conference on Electronics and Sustainable Communication Systems,IEEE (ICESC 2020)
[2] P. L. Srinivasa Murthy and G. Soma Shekar and P. Rohith and G. Vishnu Vardhan Reddy"Loan Approval Prediction System Using Machine Learning " , 2015.
[3] Anant Shinde and Yash Patil and Ishan Kotian and Abhinav Shinde and Reshma Gulwani"Loan Prediction System Using Machine Learning", 2022.
[4] Ms. Kathe Rutika Pramod and Ms. Panhale Sakshi Dattatray and Ms. Avhad Pooja Prakash and Ms. Dapse Punam Laxman and Mr. Ghorpade Dinesh B "An Approach For Prediction Of Loan Approval Using Machine Learning Algorithm" ACM, 2021.
[5] Miraz Al Mamun and Afia Farjana and Muntasir Mamun "Predicting Bank Loan Eligibility Using Machine Learning Models and Comparison Analysis", 2022
[6] X.Frencis Jensy, V.P.Sumathi,Janani Shiva Shri, "An exploratory Data Analysis for Loan Prediction based on nature of clients", InternationalJournal of Recent Technology and Engineering (IJRTE),2018.
[7] Sivasree M S, Rekha Sunny T, "Loan Credibility Prediction System Based on Decision Tree Algorithm", International Journal of Engineering Research & Technology ,2015.

INNO SPACE
SJIF Scientific Journal Impact Factor
Impact Factor
7.54

INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

निस्केयर
NISCAIR

# INTERNATIONAL JOURNAL OF MULTIDISCIPLINARY RESEARCH IN SCIENCE, ENGINEERING AND TECHNOLOGY

| Mobile No: +91-6381907438 | Whatsapp: +91-6381907438 | ijmrset@gmail.com |

www.ijmrset.com