**Loan Application Status Prediction**

**Contents:**

* + - Problem Definition
    - Data Analysis
    - EDA Concluding Remarks
    - Pre-processing Pipeline
    - Building Machine Learning Models
    - Concluding Remarks

**Problem Definition:**

**Load Application Status prediction** is used to predict whether the loan of the applicant will be approved(Loan\_status) or not on the basis of the details provided in the dataset.

This project aims to build a predictive model that helps banks and financial institutions assess the likelihood of a loan being approved based on various applicant features. The dataset includes details like credit history, loan amount, their income, dependents etc.

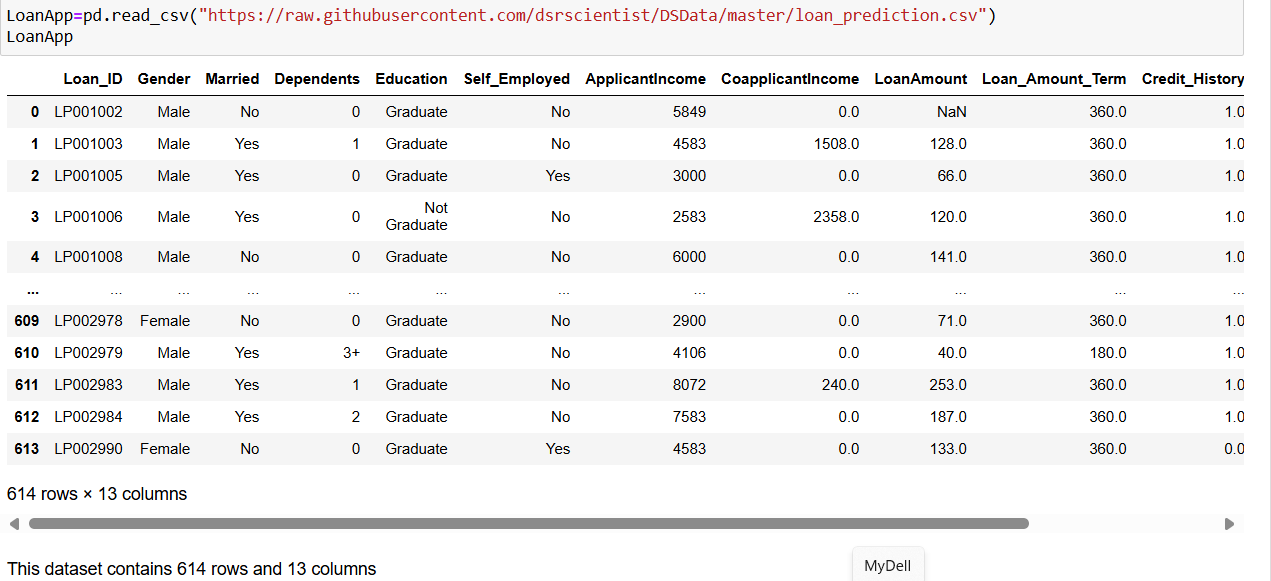
**Data Analysis:**

This dataset includes details of applicants who have applied for loan

Dataset Link:

<https://github.com/dsrscientist/DSData/blob/master/loan_prediction.csv>

This dataset contains 614 rows and 13 columns.

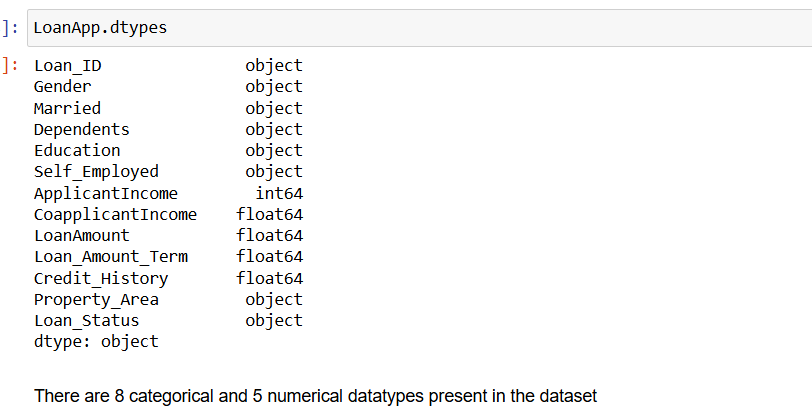


**Variable Definition:**

1. Loan\_ID - This refers to the unique identifier of the applicant's affirmed purchases
2. Gender - This refers to either of the two main categories (male and female) into which applicants are divided on the basis of their reproductive functions
3. Married - This refers to applicant being in a state of matrimony
4. Dependents - This refers to persons who depends on the applicants for survival
5. Education - This refers to number of years in which applicant received systematic instruction, especially at a school or university
6. Self-employed - This refers to applicant working for oneself as a freelancer or the owner of a business rather than for an employer
7. Applicant Income - This refers to disposable income available for the applicant's use under State law.
8. CoapplicantIncome - This refers to disposable income available for the people that participate in the loan application process alongside the main applicant use under State law.
9. Loan Amount - This refers to the amount of money an applicant owe at any given time.
10. Loan\_Amount\_Term - This refers to the duration in which the loan is availed to the applicant
11. Credit History - This refers to a record of applicant's ability to repay debts and demonstrated responsibility in repaying them.
12. Property\_Area - This refers to the total area within the boundaries of the property as set out in Schedule.
13. Loan Status - This refers to whether applicant is eligible to be availed the Loan requested.

Here Loan Status is the Target Variable.

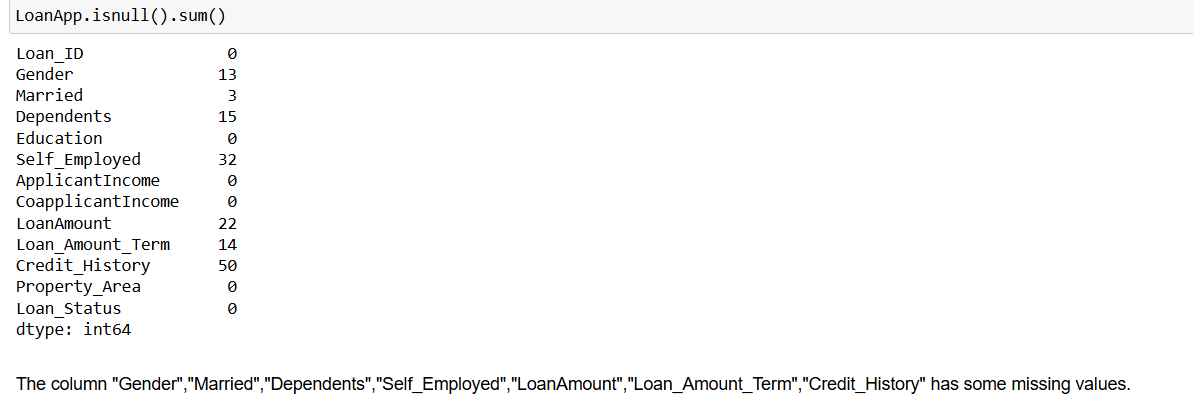
Datatypes:

****

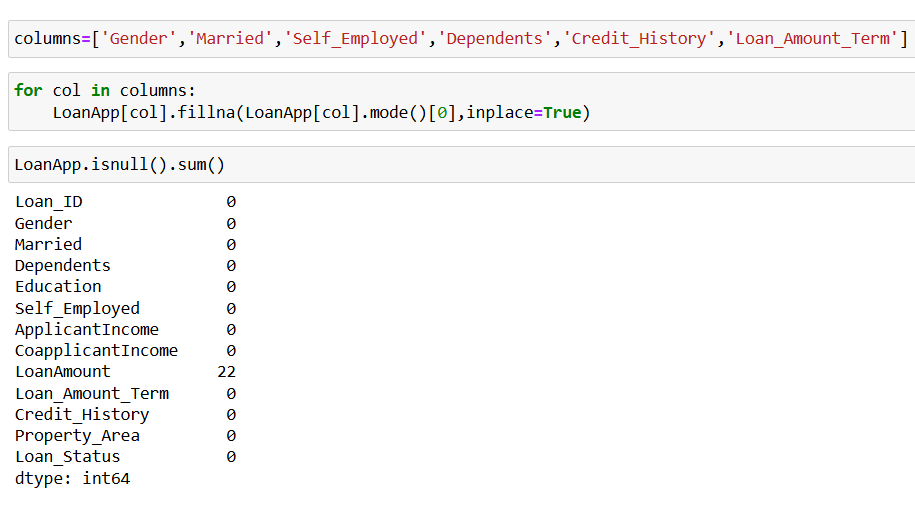
In this dataset there are 8 categorical data and 5 numerical datatypes present.

Missing Values:

We could see there are certain missing values in the columns “Gender, Married, Self -Employed, LoanAmount,Loan\_Amount\_Term, Credit history”.



We are filling the missing values by mean and mode values. For categorical datatype we are filling the missing values by mode of that value and for Numerical datatype we are filling with the mean of the specific column.



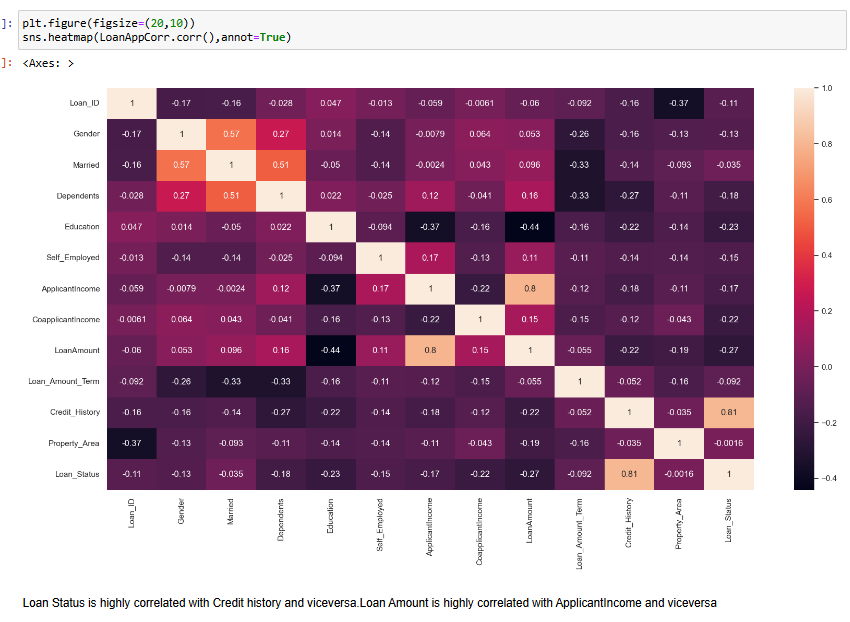


Hence now we could see that there are no null values in the dataset

**EDA Concluding Remarks:**

* From the EDA Analysis, We can conclude the Loan Approval is high for Employee who are not self Employed.
* Marital status with yes has the high chance of Loan Approval when compared to Single Relationship.
* Dependents is less loan approved rate is high.
* Property Area with Semiurban Area has high Chance of Loan Approval.
* Graduate has the high chance of loan approval when compared to Non graduate.
* Credit history score is 1.0, loan Approved else loan rejected.
* Loant\_Amount\_term with long period has the highest count.
* Applicant and CoApplicant Income is high and there is a high chance of loan approval.

**Correlation:**

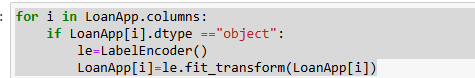


* From the Correlation matrix, we could confirm that Loan status is highly Correlated with Credit History and vice versa.
* Loan Amount is highly correlated with Applicant Income and viceversa.
* Education is correlated with LoanAmount and viceversa.

**Pre-processing Pipeline**

**Encoding Technique:**

There are 8 categorical data and inorder to predict the model we need to convert the categorical into numerical by using LabelEncoder technique.

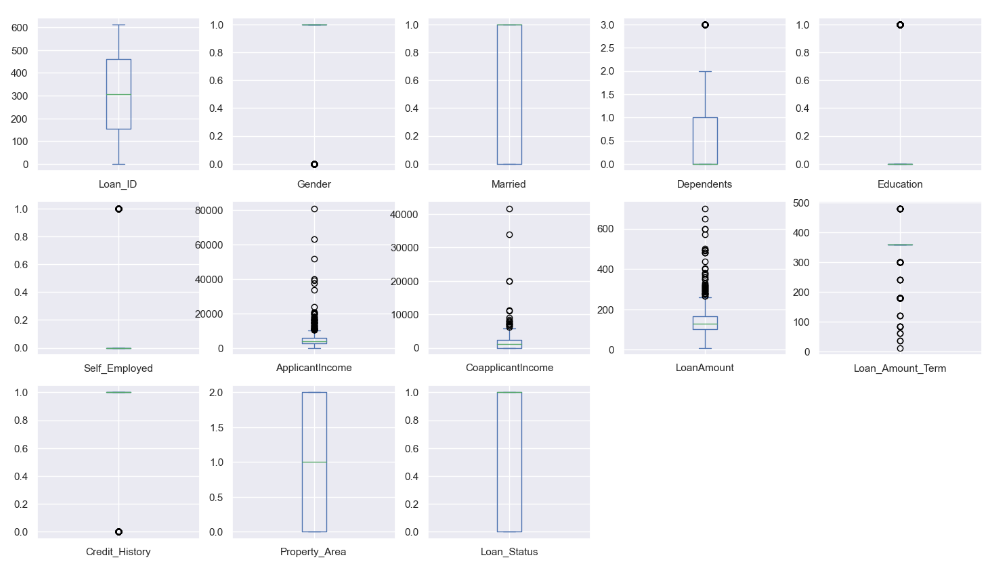
****

**Detecting Outliers using Boxplot:**

As we find there is gap between 75% and max value, outliers are present in the columns“ApplicantIncome,CoapplicantIncome,LoanAmount,LoanAmountTerm”.

Hence we are confirming the same using boxplot.

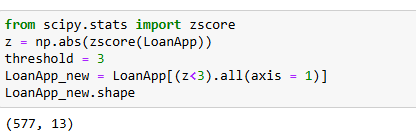
LoanApp.plot(kind='box',subplots=True,layout=(3,5),figsize=(18,10))

****

From the boxplot, we could see Outliers are present in the ApplicantIncome,CoapplicantIncome,LoanAmount,LoanAmountTerm.

**Removing Outliers:**

We are removing the outliers using zscore. Hence the dataset is reducted to 577 rows and 13 columns.

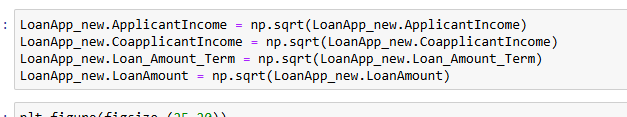


**Treating Skewness:**

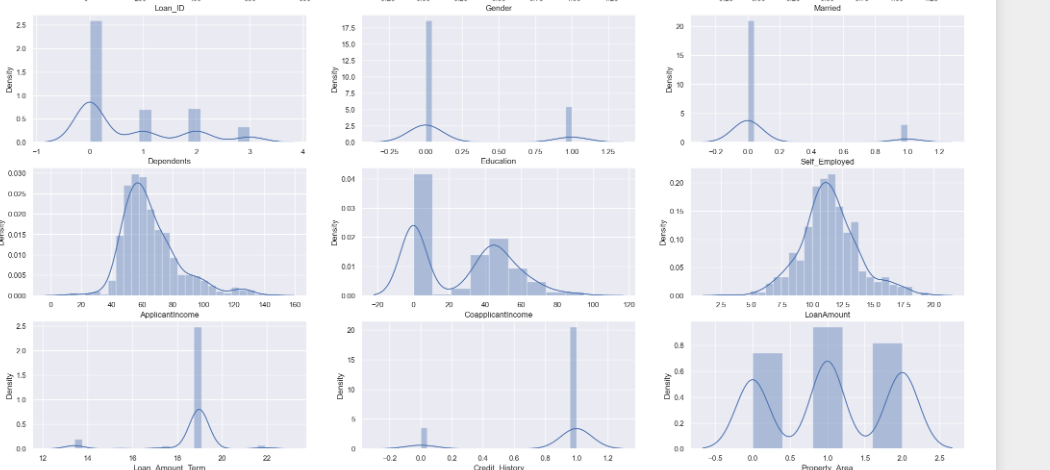
As mean >median : Right Skewness present in the columns “ApplicantIncome,CoapplicantIncome,LoanAmount”.

Mean<Median: Left Skewness present in columns “Loan\_Amount\_Term,Credit\_History”

We are treating the skewness by square root method.



Now the skewness removed.



**Building Machine Learning Models:**

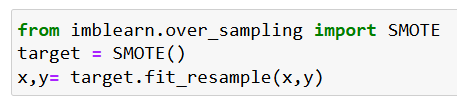
**Data Splitting:**

As mentioned earlier “Loan status” is the target variable. Hence, we are dropping the Loan status and taking the rest of the columns as the input variable.

**Balancing Data using SMOTE:**



As the target variable is not balanced, we are balancing the data using SMOTE.



We are splitting the data into 75% and 25%

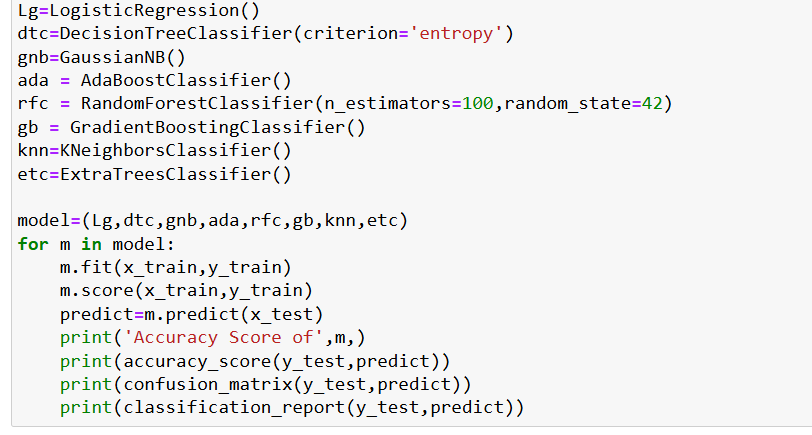
x\_train,x\_test,y\_train,y\_test=train\_test\_split(x,y,test\_size=0.25,random\_state=42)

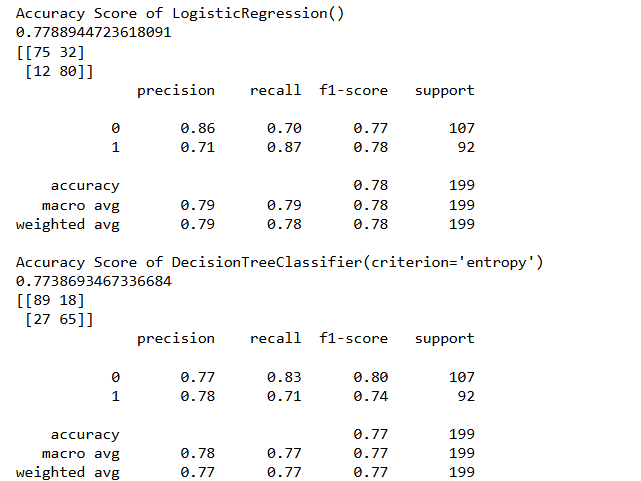
**Model Training:**

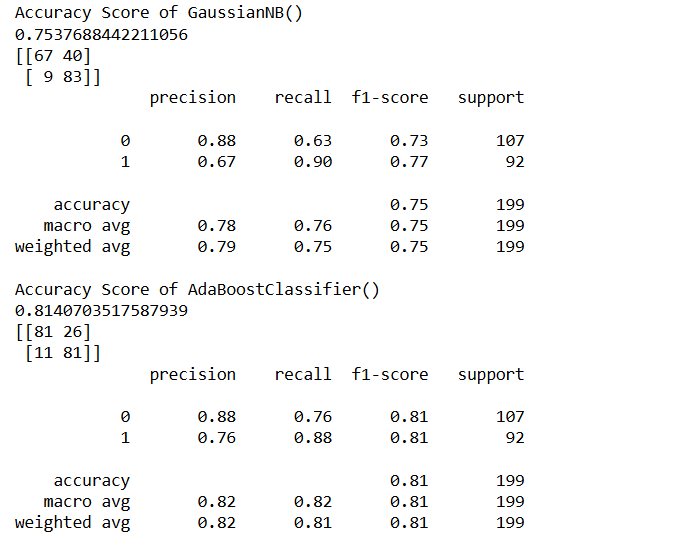
As the target variable is in terms of 0’s and 1’s , we are taking the Logistic Regression and Classifiers models into account and then predicting the target using the below models.

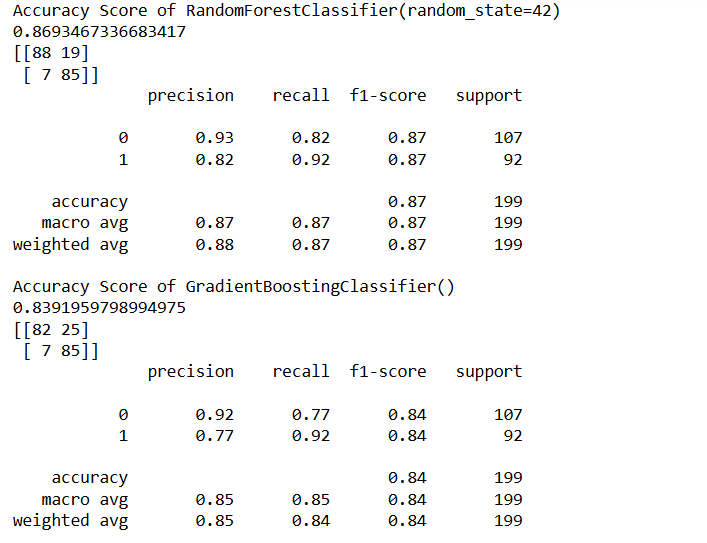
* Logistic Regression
* GaussianNB
* AdaBoostClassifier
* RandomForestClassifier
* GradientBoostingClassifier
* KNeighborsClassifier
* ExtraTreeClassifier

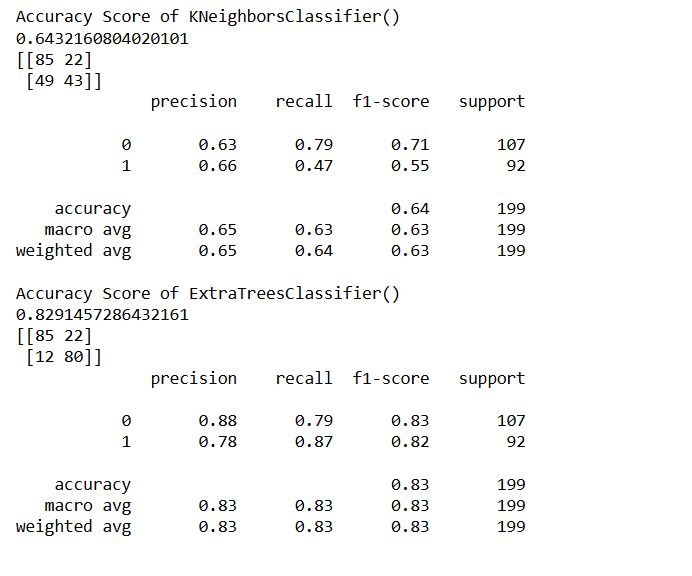
Also, we need to consider three metrics such as Accuracy Score, Confusion matrix and classification report. Comparing all the models along with metrics and we need to build the best model among them.





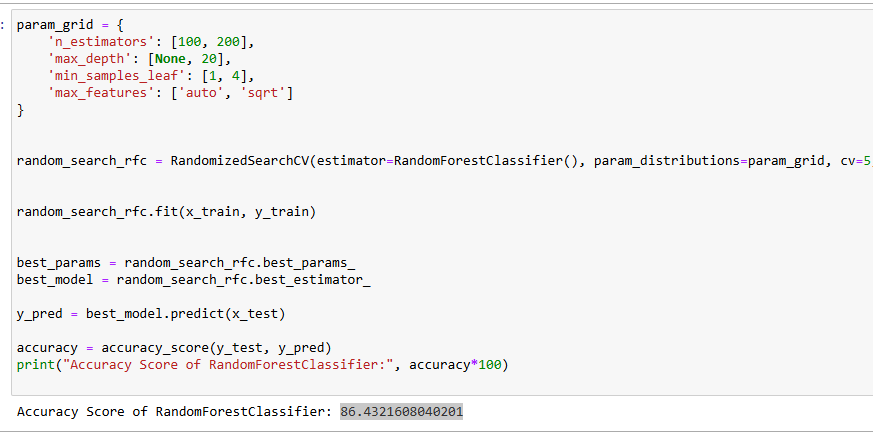
****

****

****

**Among all the models, RandomForestClassifier works best and gives the accuracy score of 86.93%**

**Conculsion:**

****

After HyperParameter Tuning using RandomSearchCV, RandomForestClassifier works with Accuracy score of 86.4321608040201%

Out of 199 rows testing data, 85 rows are Loan Approved (1’s)and 88 rows Loan Rejected (0’s).

Accuracy Score of RandomForestClassifier(random\_state=42)

0.8693467336683417

[[88 19]

[ 7 85]]

precision recall f1-score support

0 0.93 0.82 0.87 107

1 0.82 0.92 0.87 92

We are saving the best model using pickle.

