DATA TRAINED ACADEMY BLOG ARTICLE



LOAN APPLICATION STATUS PREDICTION



INTRODUCTION:

Loans can be a lifesaver. Be it buying a new home or going for a world tour or completing higher education from the best colleges loans help us make our dreams come true.

A loan is a sum of money that one individual/business/company borrows from another individual/business/company to meet any planned or unplanned financial requirement. The party that gives the money is called the lender and the party that receives the money is called the borrower. By taking a loan, the borrower incurs a debt that he has to pay back along with interest. The interest rate is pre-decided and is levied at periodic intervals.

1.Problem Definition:

This dataset includes details of application who have applied for loan. The dataset includes details like credit history, loan amount, their income, dependents, Education, Self Employed, Applicant income, Co applicant Income, Loan Amount, Loan Amount Term, Credit History, Property Area, Loan Status.

In this project, we are provided a dataset which has the details of the loan application status along with the customer details.

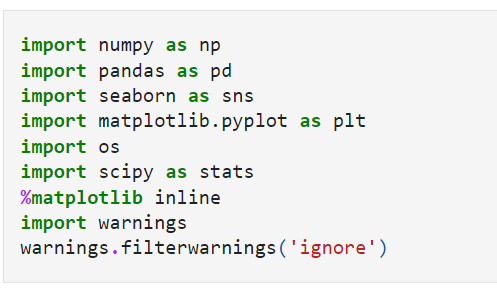
In this particular problem we have to look into the loan person details and analyse the sample to known whether the loan status is accepted or rejected.

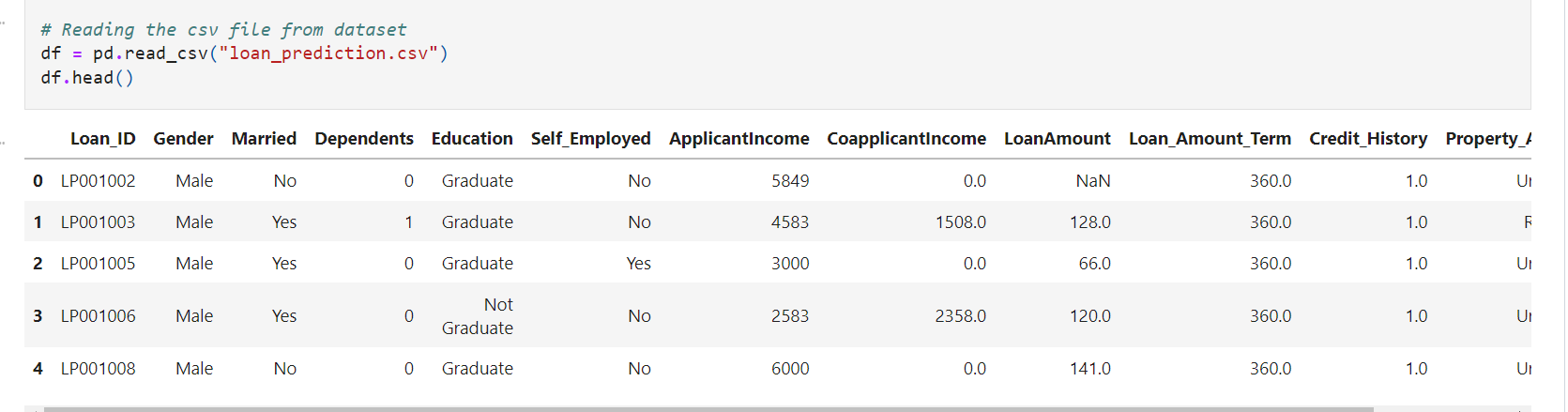
In this problem statement explains that the target variable contains the categories. So it is a “Classification Problem” so let’s predict whether the loan will be approved or not in loan status.

2.Data Analysis:

The process of cleaning, transforming, and extracting data to discover the useful information for business decision making is called data analysis. Data analysis has multiple facets and approaches, encompassing diverse techniques under a variety of names, and is used in different business, science, and social science domains. In today’s business world, data analysis plays a role in making decisions more scientific and helping business operate more effectively.

Importing necessary libraries and dataset.

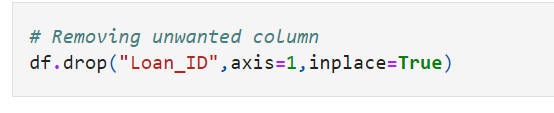


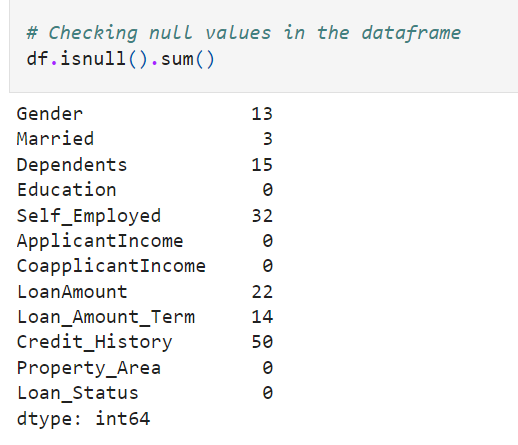


I have imported the dataset which was in the csv file using pandas. The dataset contains 614 rows and 13 columns having both numerical and categorical data.

Data Preparation and cleaning:

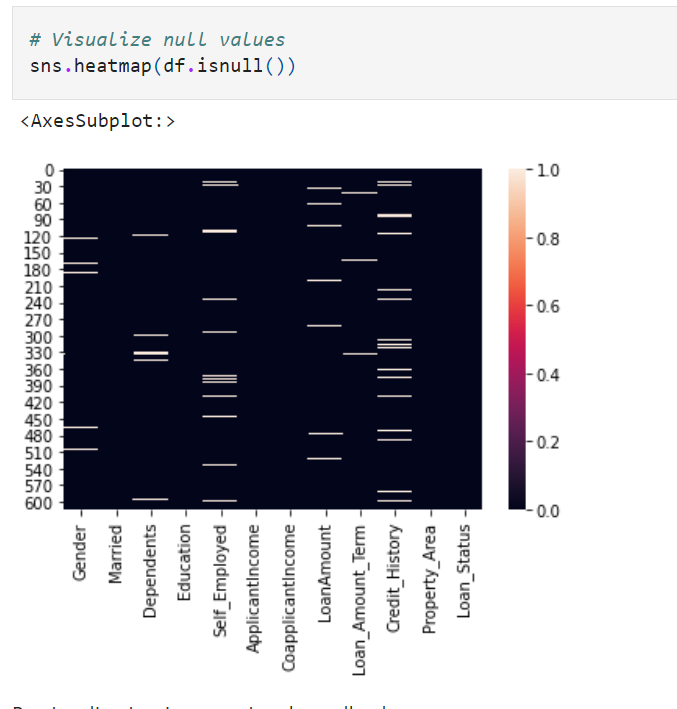
* First, we need to check some statistical information about the dataset like checking shape, datatypes, nunique, value counts, etc.
* After checking the value counts, if we find any unwanted columns in the dataset then we need to do feature engineering on those columns based on the problem.
* In df columns we found Loan\_ ID given to the applications. It has no significance in the prediction. Let’s drop this column.





After dropping the above column, We can can see there are missing values in dataset. The above we can see the missing values in the columns:

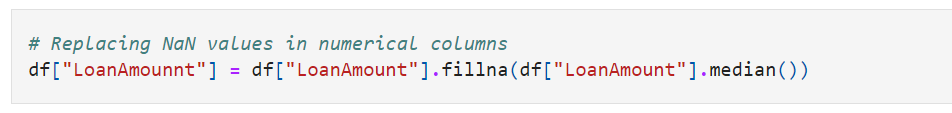
Gender,Married,Dependents,Self\_Employed,LoanAmount,Loan\_Amount\_Term,Credit\_History. We have to replace these null values using imputation techniques.



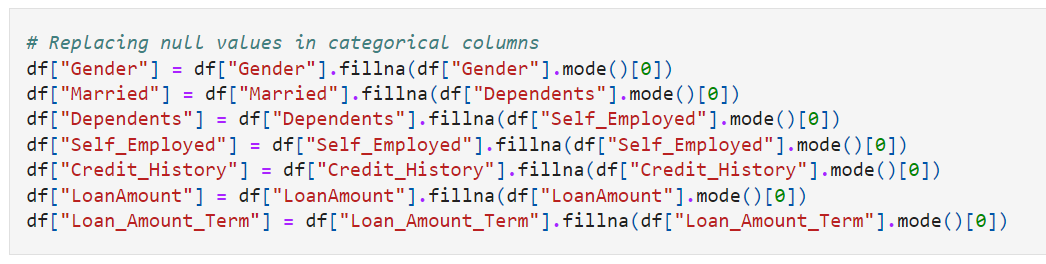
Above we can see there are null values.

IMPUTATION TECHNIQUE:

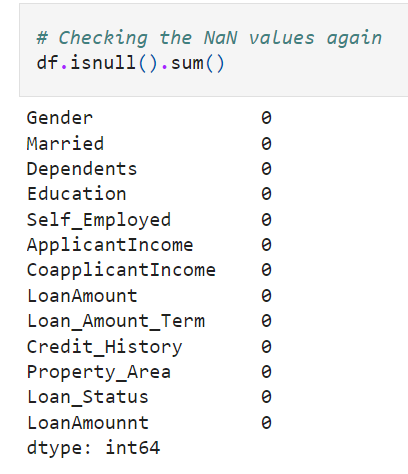
There is skewness in all the numerical so we cannot replace the NaN values with mean I have replaced it with median.

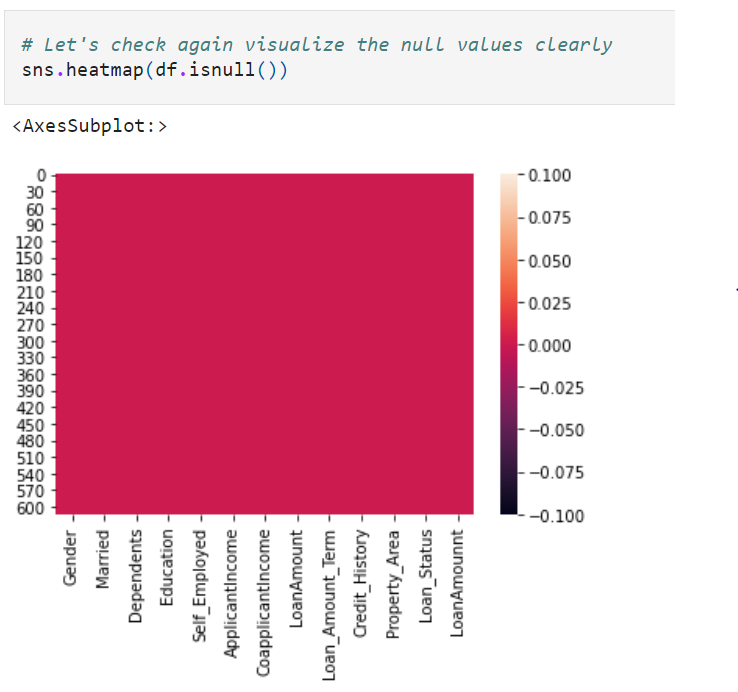


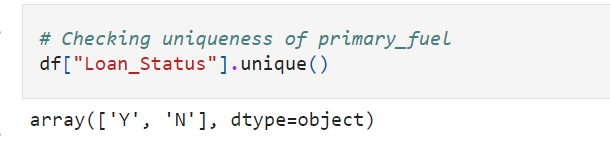
Above we can replaced the null values in numerical columns by there median.

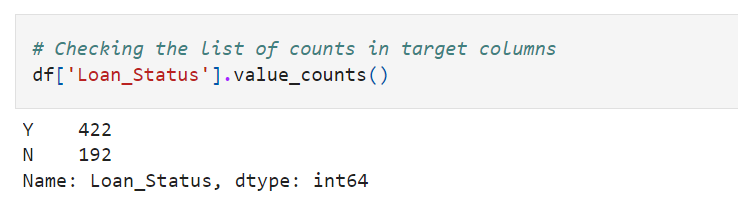


Above we can replaced the null values in categorical columns by there mode.

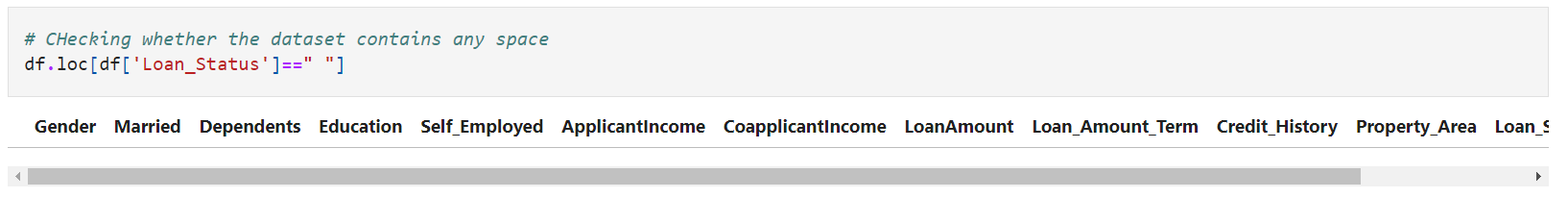


We can see the null values has been removed. 

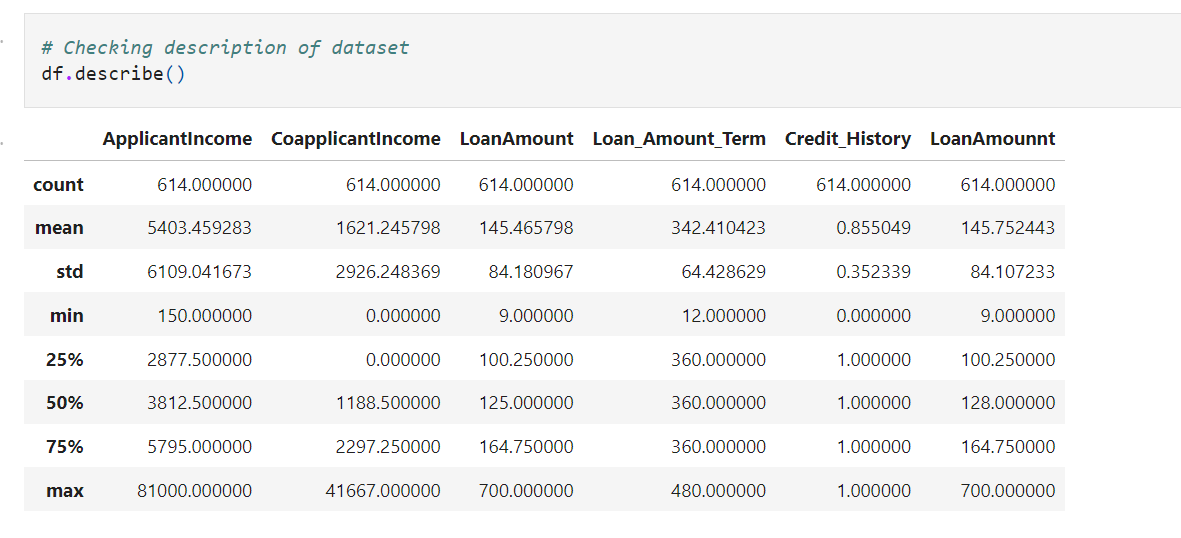




We have two counts in Loan \_Status namely “Y” and “N”. Here “Y” stands for “Yes” that is loan of the applicant is approved and “N” stands for “No” that is the loan of the applicant is not approved.



I can see there are no spaces in the dataset.



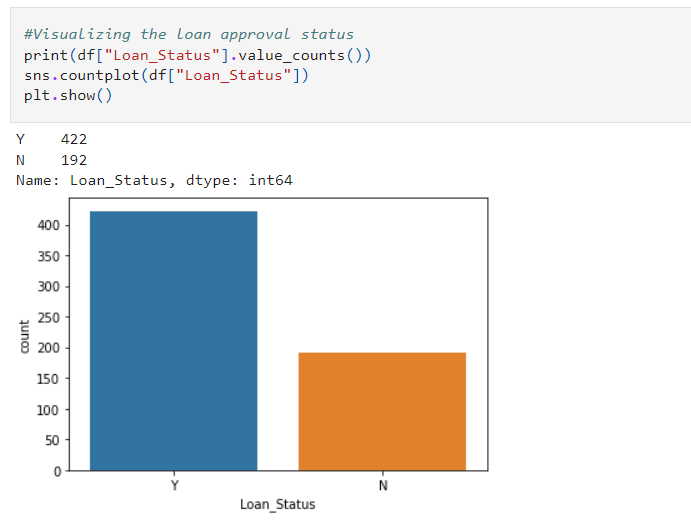
This gives the statistical information of the dataset. There are no negative values and invalid values are present. This gives the summary of numerical data.

The mean value and 2nd quantile value has high difference in almost all the columns which means there is a lot of outliers in all the columns.

The count is same which means no null values.

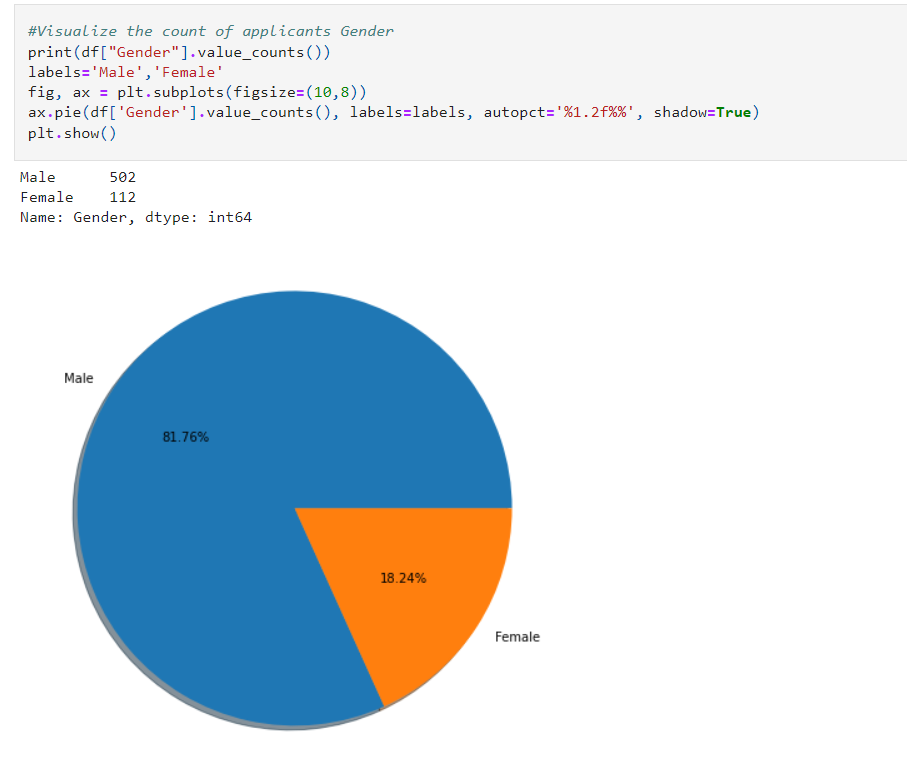
Data Visualization:

In the data visualization we will compare features and labels of the dataset.

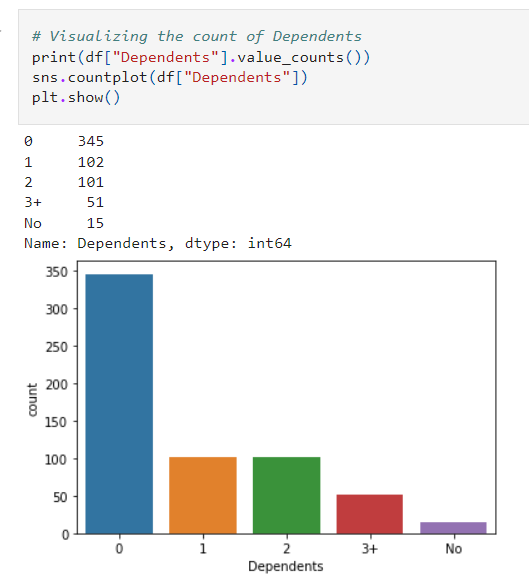


From the plot we can observe that the count of “N” is high compared to “Y”.

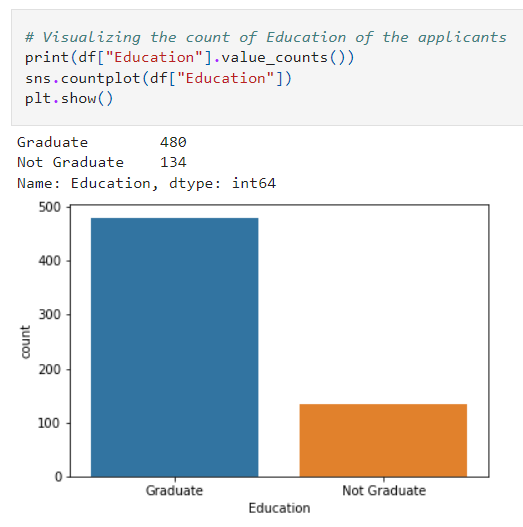
Here we can assume that “Y” stands for “Yes” that is the Loan status and “N” stands for “No” means the Loan status is not prediction. There are 422 “Yes” and 192 “No”.



There are more number of Male applicants applying for loan than Female applicants. There are about 81% of the Male candidates and only 18% of Female candidates are applying for the loan.



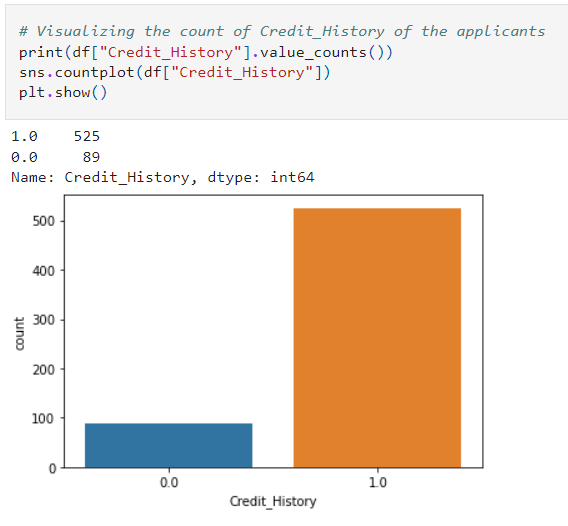
The applicants who have 0 dependents have high counts and the applicants having high counts and the applicants having more than 3 dependents counts are very less.



The count of graduate applicants is high in counts means the maximum number of graduated applicants are applying for the loan.

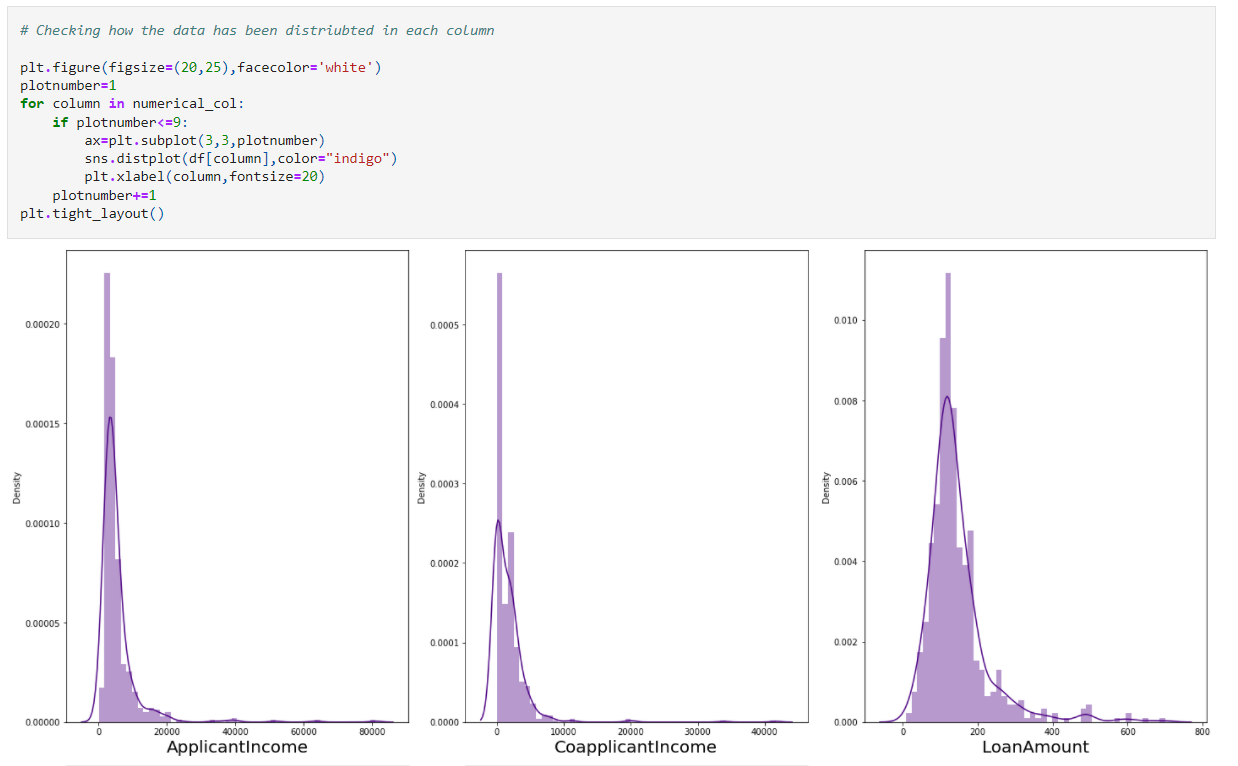
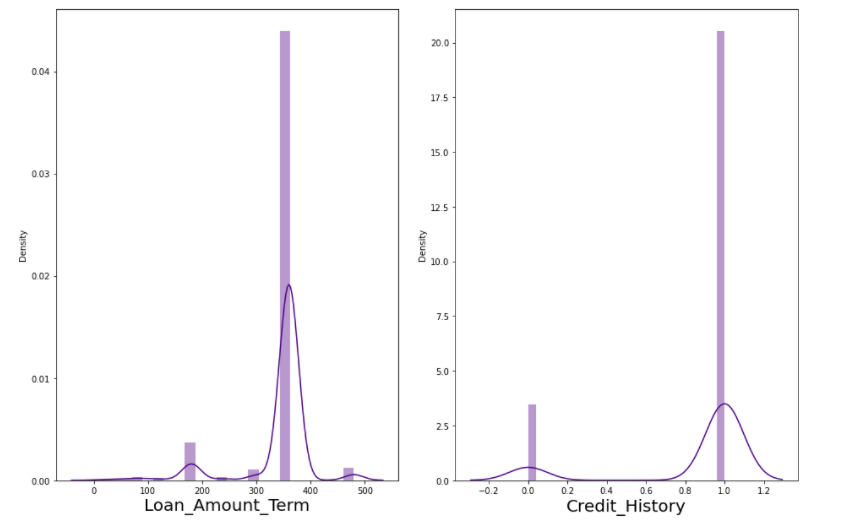


Most of the applicants or not self employed that means they might working in the public sectors and only 82 applicants are self employed and running their own business.



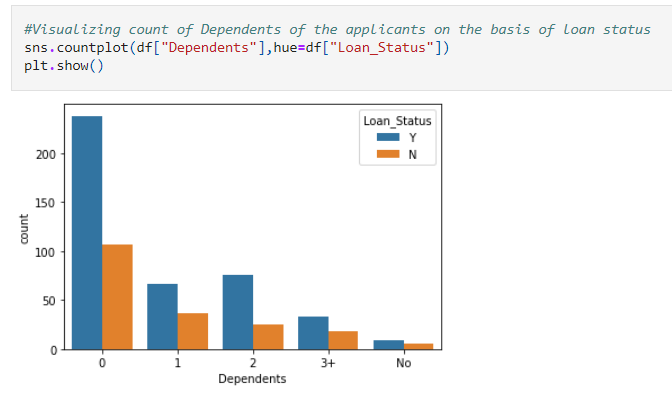
Most of the applicants who have credit history 1 are high in numbers.

Distribution of Skewness:

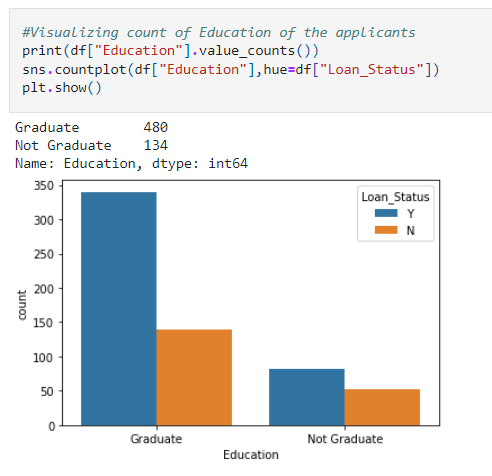
 

The data is not normally distributed in any of the columns. The mean value is greater than the median in Applicant Income, Co applicant Income, Loan Amount and Total Income which means they are skewed to right. The median is greater than the mean in Loan Amount Term and Credit History Columns which means they are skewed to left. We will remove these skewness, using appropriate methods in the later part.

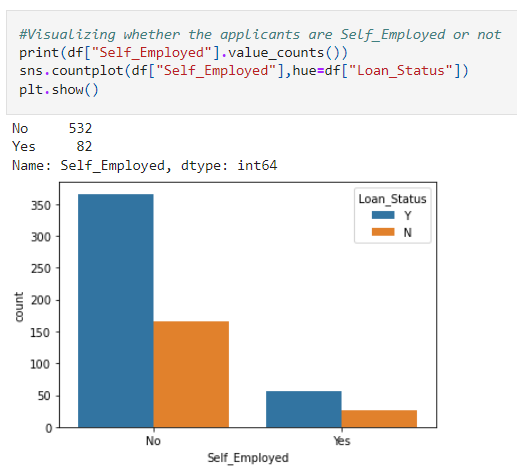
Bivariate Analysis:



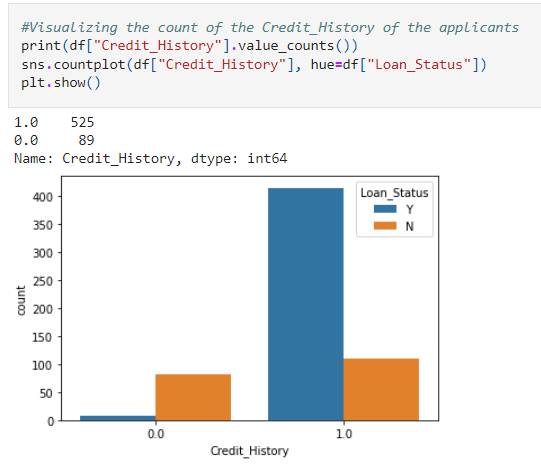
The count of 0 dependents is high which means most of the applicants have no dependents. Having dependents means having commitments. The 3+ dependents more than 3 applicants have dependents. The applicants who have dependents 0 are more likely to get their loan approved.



Most of the applicants who are applying for loan are graduated and only few are not graduated. Also the applicants who are graduated have tendency of getting loans than who are not.



Most of the applicants are not Self Employed means they are working in public sectors and only few of the applicants are Self Employed. The applicants who are not Self Employed have the tendency of getting their loans than self employed applicants.

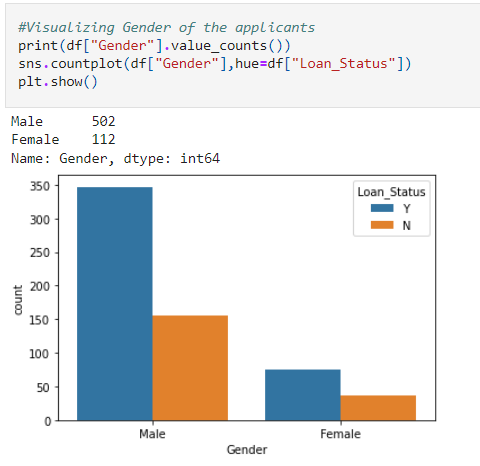


The Credit History gives the information of the applicants who took loan in the past have cleared or not. Here we can notice the applicants who have credit history 1 have high counts which means most of the applicants have cleared their past loan only few of them have to clear the loan.

The applicants who have credit history 1 have got their loan approval which means they have cleared their past loans.



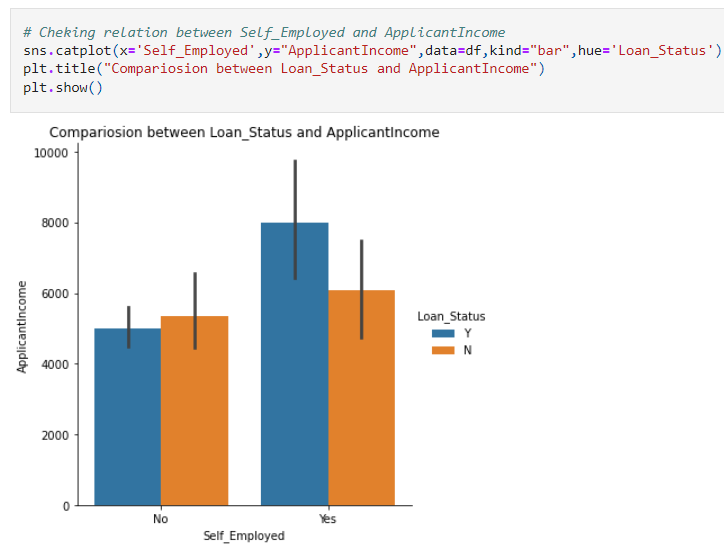
Most of the applicants from the Semiurban are applying for loan followed by Urban area. Also they have more chance of getting their loan approval.



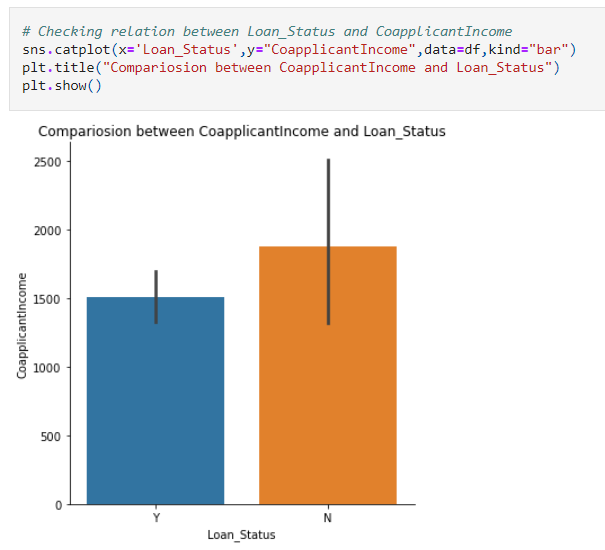
The male applicants who applied for the loan have got approved compared to the female applicants.



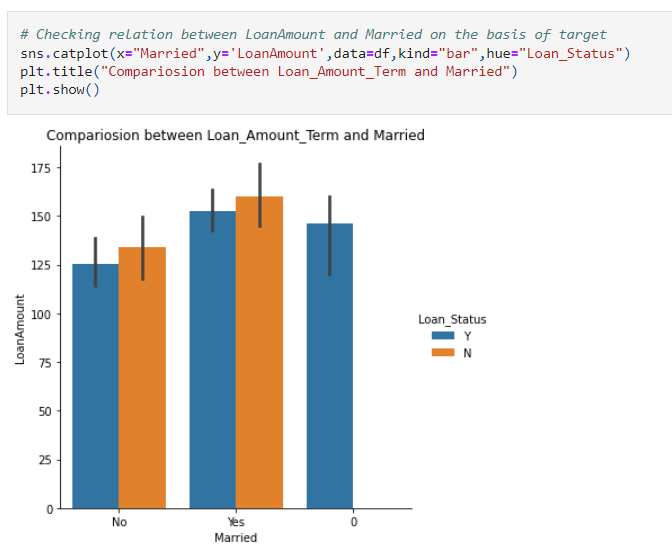
There are more number of Male applicants who are applying for loan than compared to Female applicants. Also more Male candidates loans got approved compared to Female.



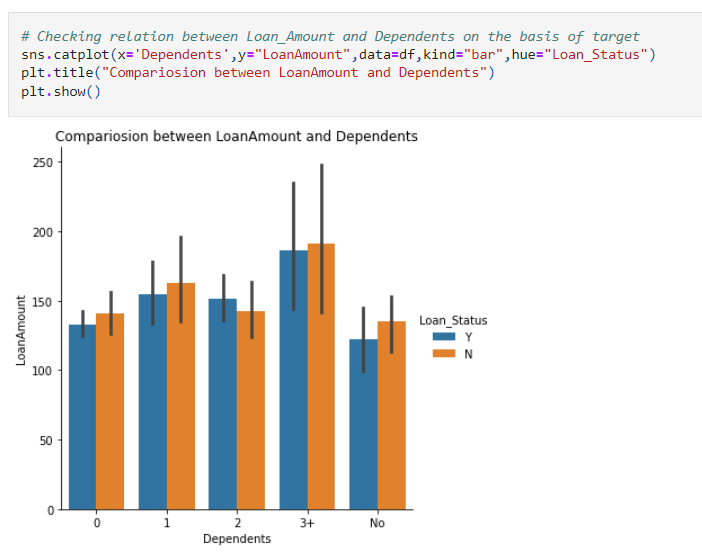
The applicants whose loan got approved have average income and have their own business means they are self employed.



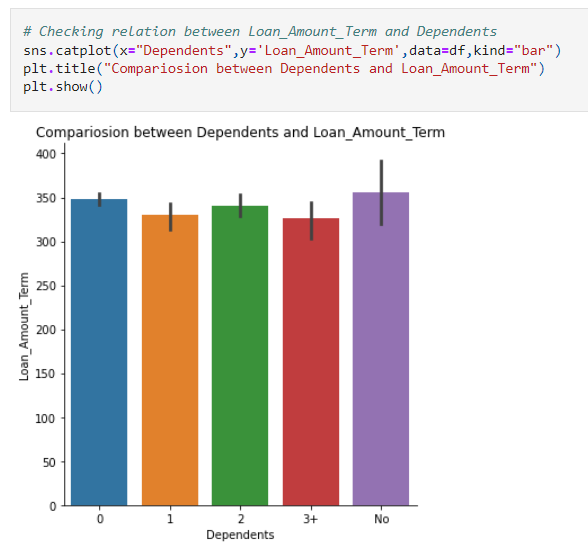
The co applicants who got loan have average income.



The applicants who got married and have average loan amount have more tendency to getting loan.



The applicants who have more than 3 dependents with average loan amount have got their loan approved.



The applicants 0 dependents have high Loan amount term followed by the dependents 2.

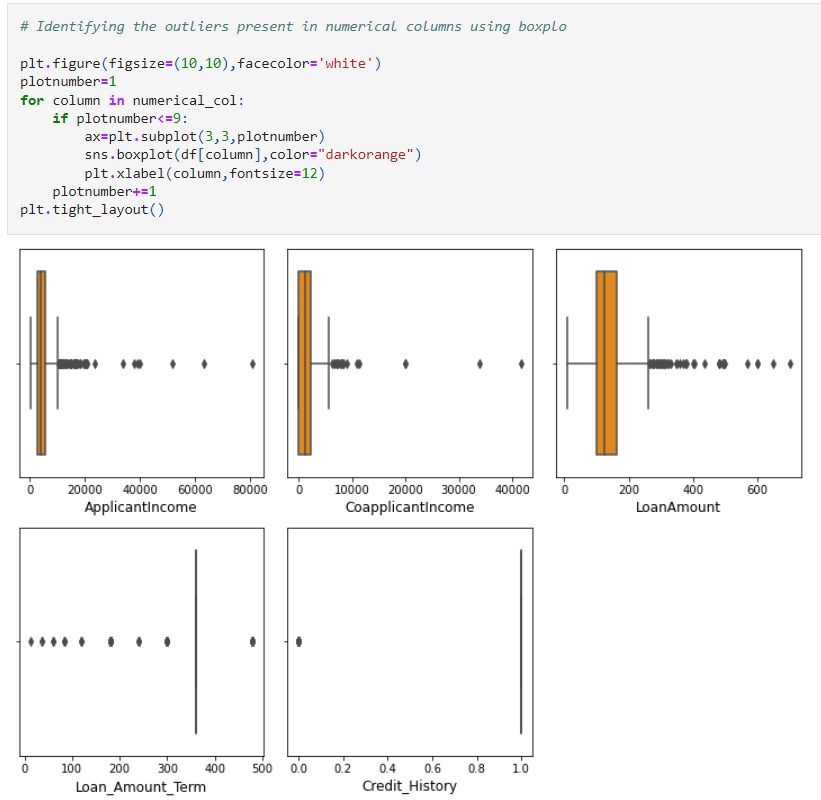
3.EDA Concluding Remark:

We have checked the null values in the dataset and there was no missing values found.

We have dropped some of the irrelevant columns to overcome with the multicollinearity problem.

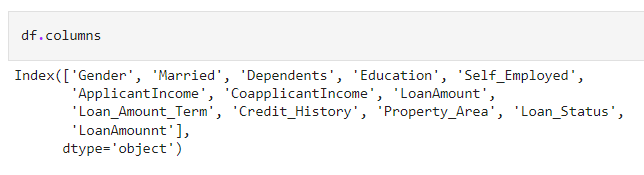
In visualization part, we have found when and where the loan application are more in number.

Checking outliers and Removing outliers and skewness in the data:



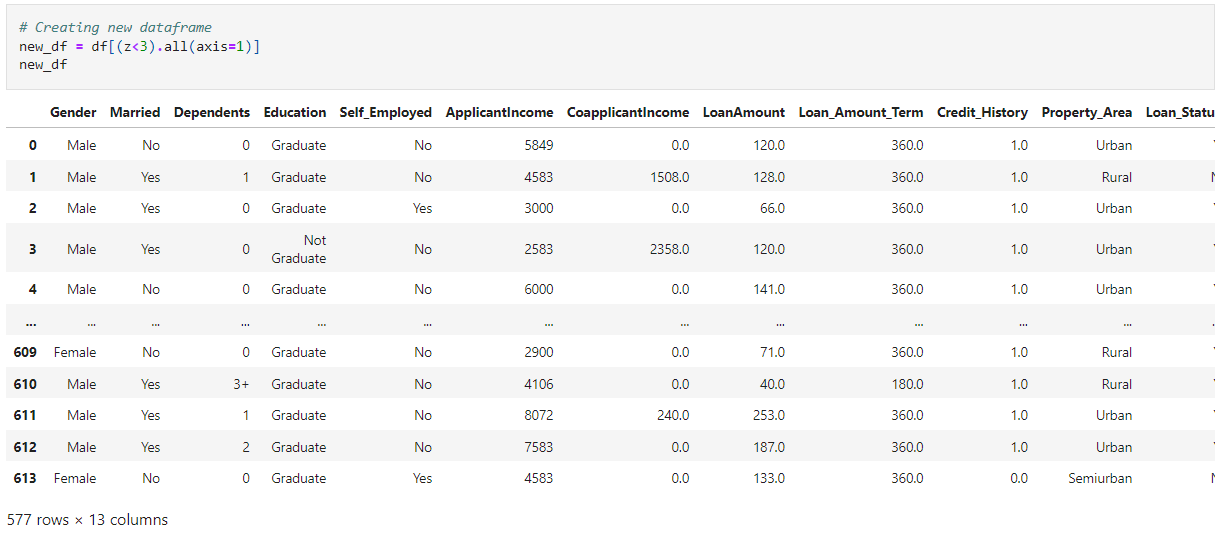
We can observe the outliers present in all the columns. But the columns Credit history has only two unique values so ne need to remove outliers in this column. Let’s remove outliers in remaining columns them using Z score method.

Removing Outliers:

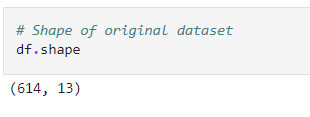




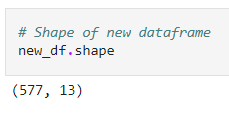
Now we have removed the outliers. So data loss by creating new data frame.



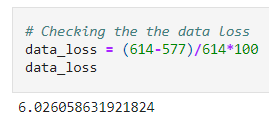
This is the new data frame after removing the outliers. Here we have removed the outliers whose Z score is less than 3.



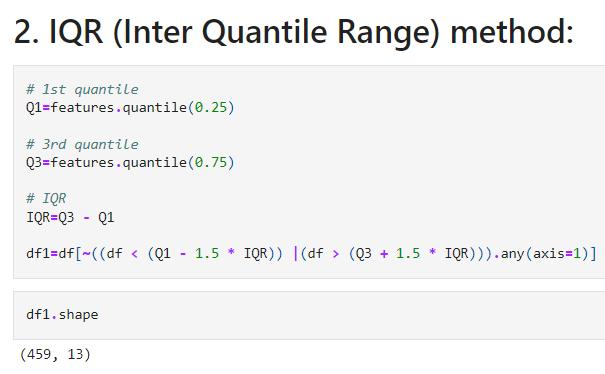
Before removing the outliers we had 614 rows and 12 columns in our dataset.



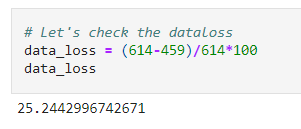
After removing the outliers we have 577 rows and 12 columns.



I am losing only 6% data, hence I am removing outliers. Let’s remove the outliers and check data loss using IQR method.

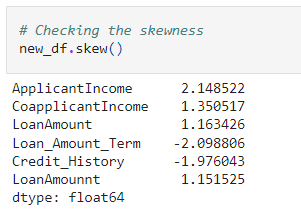


Using IQR method the data frame has 459 rows and 12 columns.



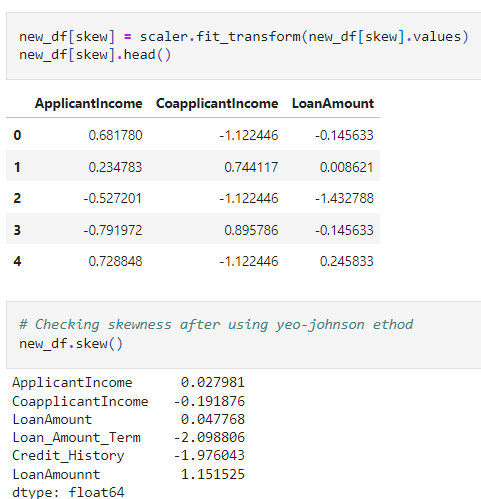
Using IQR method I am losing 25% of data, So considering Z score method.

Checking Skewness:

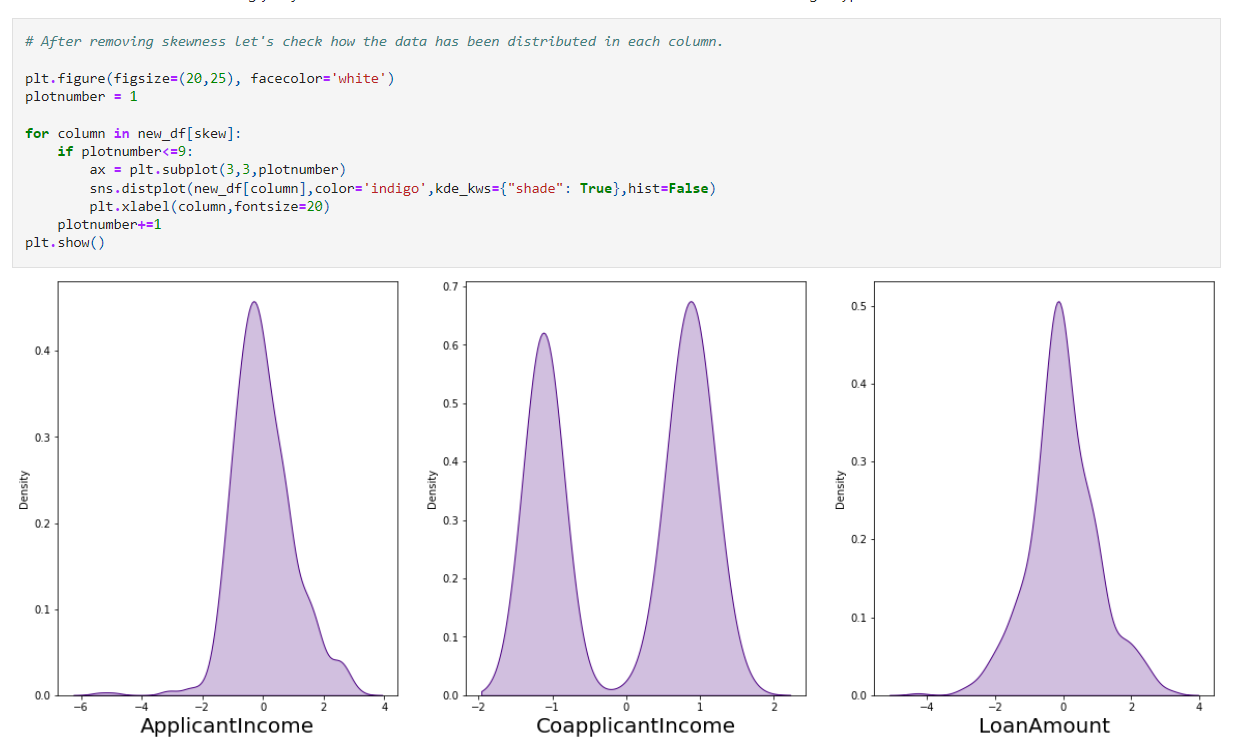


The skewness present in all the above columns. Here the columns Credit History and Loan Amount Term have categorical data of integer type so no need to remove skewness in these columns.



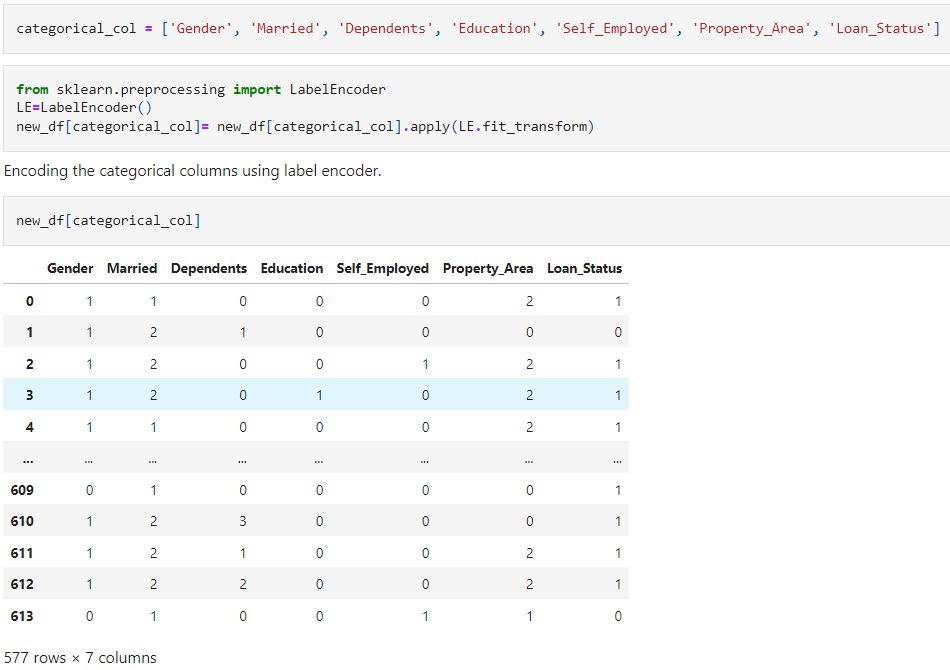


I have removed the skewness using yeo-johnson method. The skewness has been removed in all the numerical integer type columns.

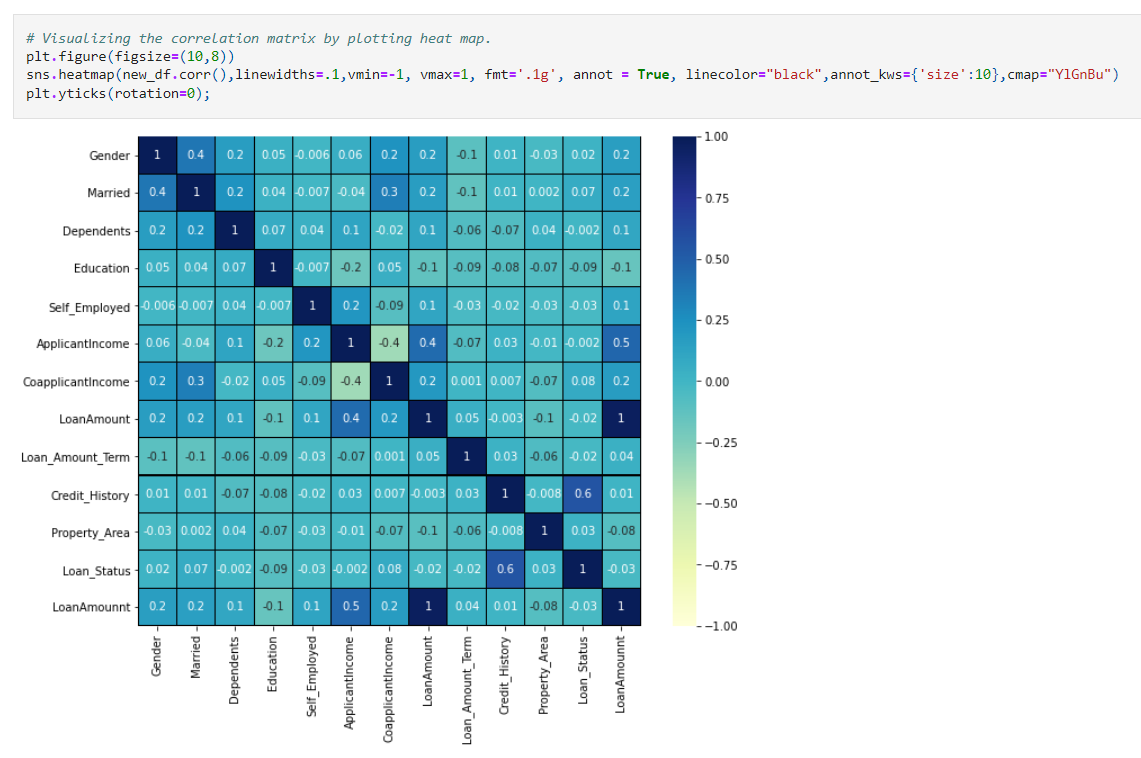


We can notice our data is almost normal and skewness is also removed. Now we can proceed further.

Since our dataset contains object data, we need to encode them using any of the encoding methods. Here I have used label encoding method.



I have applied label encoding method to our cleaned data frame new\_ df and converted the categorical columns in to numerical.

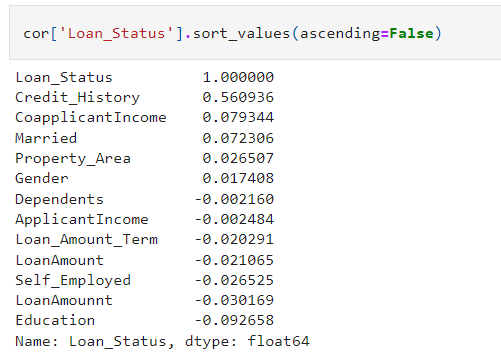


The target column Loan Status is highly positively correlated with the feature Credit\_ History.

The other features have very less correlation with the target column.

Also we can notice there is no multicollinearity issue in the features. Features have moderate level of correlation with each other.

Applicant Income and Gender is very less correlated with the target.



We can see the positive and negative correlation of target and features.

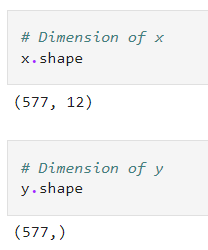

Description automatically generated

Here the columns Applicant Income has vey less correlation with the target so we can drop this column if necessary.

**4.Preprocessing Pipeline:**

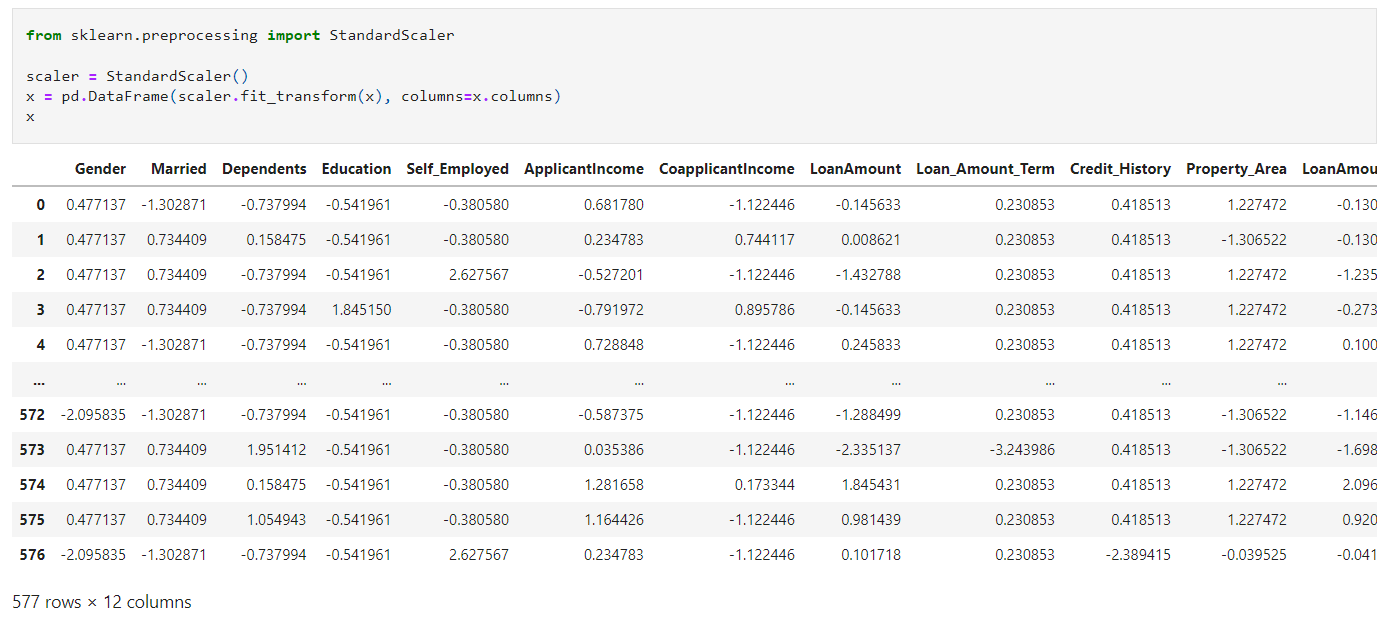
* As a first step I have to separate the dependent and independent features.





Separating the features and label variables into x and y:

Feature Scaling using Standard Scalarization:



I have scaled the data using standard scalarization method to overcome with the issue of data bianess.

Graphical user interface, text, application

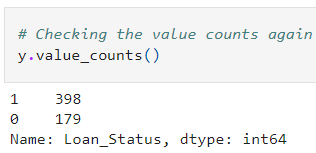
Description automatically generated

Here we can notice the class imbalancing, issue so lets use SMOTE to balance the data.

OVERSAMPLING:

Graphical user interface, text, application, Word

Description automatically generated



The data is balanced Since the highest count of the target is 398 so the data is balanced by oversampling all the classes to the count 398.

Graphical user interface, application

Description automatically generated

We have done with the pre-processing and data cleaning. Now let’s move to build the model.

**5.Building Machine Learning Models:**

Finding Best Random State and Accuracy:



The best accuracy is 88.70% on the Random state 78.

Creating train \_test split:



We have created a new train test split using Random State.

Classification Algorithms:

**Text

Description automatically generated**

# Decision Tree Classification:

Table

Description automatically generated with low confidence

Here we can see accuracy using Decision Tree Classifier is 67.81%.

Chart

Description automatically generated with low confidence

This is the confusion matrix for Decision Tree classifier where we can observe the true positive rate, false positive rate, true negative rate, false negative rate. It is plotted predicted value against True values.

Random Forest Classifier:

Table

Description automatically generated

The accuracy using Random Forest Classifier is 85.05%.

Graphical user interface, application, Teams

Description automatically generated

This is the confusion matrix for Decision Tree classifier where we can observe the true pressure rate, false positive rate, true negative rate, false positive rate, true negative rate, false negative rate. It is plotted predicted value against True values.

Logistic Regression:

Table

Description automatically generated

The accuracy using Logistic Regression Classifier is 87.93%.



This is the confusion matrix for Decision Tree Classifier where we can observe the true positive rate, false positive rate, true negative rate, false negative rate. It is plotted predicted value against True Values.

Support Vector Machine Classifier:

Table

Description automatically generated

The accuracy using SVC is 87.35%.

Graphical user interface

Description automatically generated

This is the confusion matrix for Decision Tree classifier where we can observe the true positive rate, false positive rate, true negative rate, false negative rate. It is plotted predicted value against True values.

KNeighbors Classifier:

Table

Description automatically generated

The accuracy using KNeighors Classifier is 85.05%.



This is the confusion matrix for Decision Tree classifier where we can observe the true positive rate, false positive rate, true negative rate, false negative rate. It is plotted predicted value against True values.

Gradient Boosting Classifier:

Table

Description automatically generated

The accuracy using Gradient Boosting Classifier is 85.05%.

Graphical user interface, application, Teams

Description automatically generated

This is the confusion matrix for Decision Tree classifier where we can observe the true positive rate, false positive rate, true negative rate, false negative rate.

It is plotted predicted value against True values.

AdaBoost Classifier:

Table

Description automatically generated

The accuracy using AdaBoost Classifier is 82.75%.

Graphical user interface, application

Description automatically generated

This is the confusion matrix for Decision Tree classifier where we can observe the true positive rate, false positive rate, true negative rate, false negative rate. It is plotted predicted value against True values.

GaussianNB Classifier:

Table

Description automatically generated

The accuracy using KNN is 86.78%.

Graphical user interface, application

Description automatically generated

This is the confusion matrix for Decision tree classifier where we can observe the true positive rate, false positive rate, true negative rate, false negative rate. It is plotted predicted value against True values.

Checking the cross-validation Score:

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application

Description automatically generated

Above are the cross-validation score for the models.

The difference between accuracy score and cross validation score for the models used: Decision Tree Classifier = 6.95%

Random Forest Classifier = 8.05%

Logistic Regression = 8.16%

SVC = 8.14%

KNeighbors Classifier = 0.78%

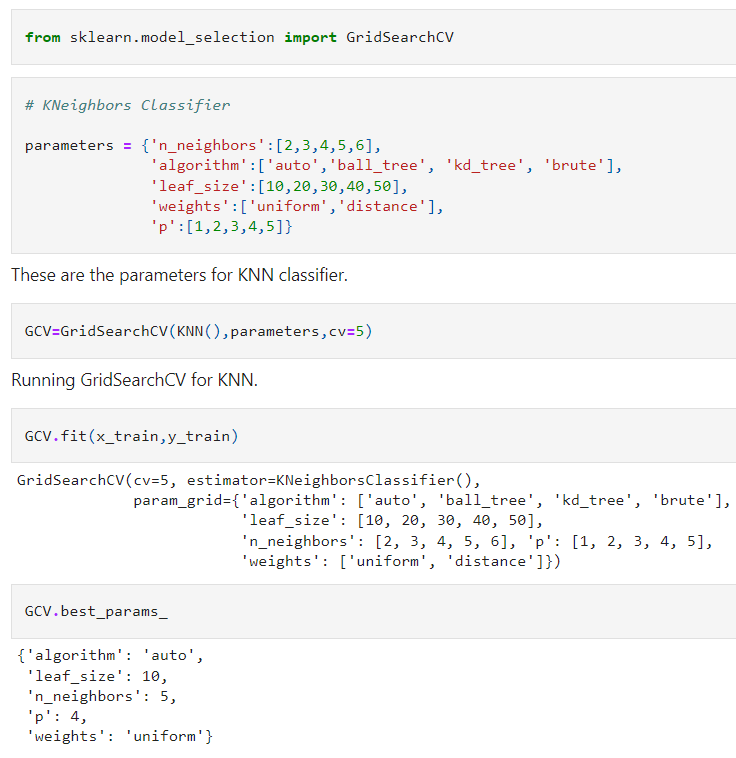
Gradient Boosting Classifier = 7.85%

AdaBoost Classifier = 8.00%

GaussianNB Classifier = 8.11%

From the difference between the accuracy score and the cross validation score we can conclude that KNeighbors Classifier as our best fitting model is giving very less difference compare to other models.

Hyper Parameter Tuning:



These are the best parameters values that we have got for KNN classifier.

Text

Description automatically generated The accuracy of best model increased by 9% after tuning and giving 77.58% which is very good.

Graphical user interface, text, application

Description automatically generated

This the AUC-ROC curve for the models that we have used and is plotted False positive rate against True positive rate.

Graphical user interface, application

Description automatically generated

This is the ROC curve for the best model KNN and AUC for KNN is 91%.

Graphical user interface, text, application

Description automatically generated

We have saved our model using joblib library.

A picture containing calendar

Description automatically generated

These are the predicted loan approval status of the applicants.

Table

Description automatically generated with medium confidence

Chart, line chart

Description automatically generated

Blue line is the actual values and red dots are predicted values and it’s pleasure to see my model is working good.

**6.Concluding Remarks:**

This particular problem needs a good vision on data, and in this problem Feature Engineering is the most crucial thing.

You can see how we have handled numerical and categorical data and how we build different machine learning models on the same dataset.

Using hyper parameter tunning we can improve our model accuracy, for instance in this model the accuracy remained same.

Using this machine Learning Model we people can easily predict the insurance claim is fraudulent or not and we could reject those application which will be considered as fraud claims.