NAME OF THE PROJECT

**LOAN APPLICATION STATUS PREDICTION**

PREPARED BY

**KHUSHBOO GUPTA**



**PROBLEM STATEMENT**

We are provided with a dataset which includes the details of applicants who had applied for the loan in the past and the loan status of the applicant whether it is passed or failed. We have to build a Machine Learning Program for the Loan Application Status.

**UNDERSTANDING THE DATASET**

**Loan\_ID:** Unique Loan ID

**Gender:** Male/ Female

**Married:** Applicant married (Y/N)

**Dependents:** Number of dependents

**Education:** Applicant Education (Graduate/ Under Graduate)

**Self\_Employed:** Self employed (Y/N)

**ApplicantIncome:** Applicant income

**CoapplicantIncome:** Coapplicant income

**LoanAmount:** Loan amount in thousands

**Loan\_Amount\_Term:** Term of loan in months

**Credit\_History:** Credit history meets guidelines

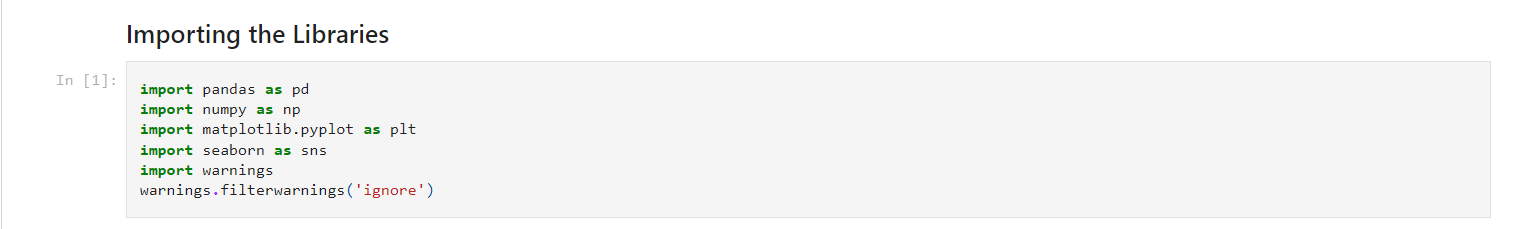
**Property\_Area:** Urban/ Semi Urban/ Rural

**Loan\_Status:** (Output Variable) Loan approved (Y/N)

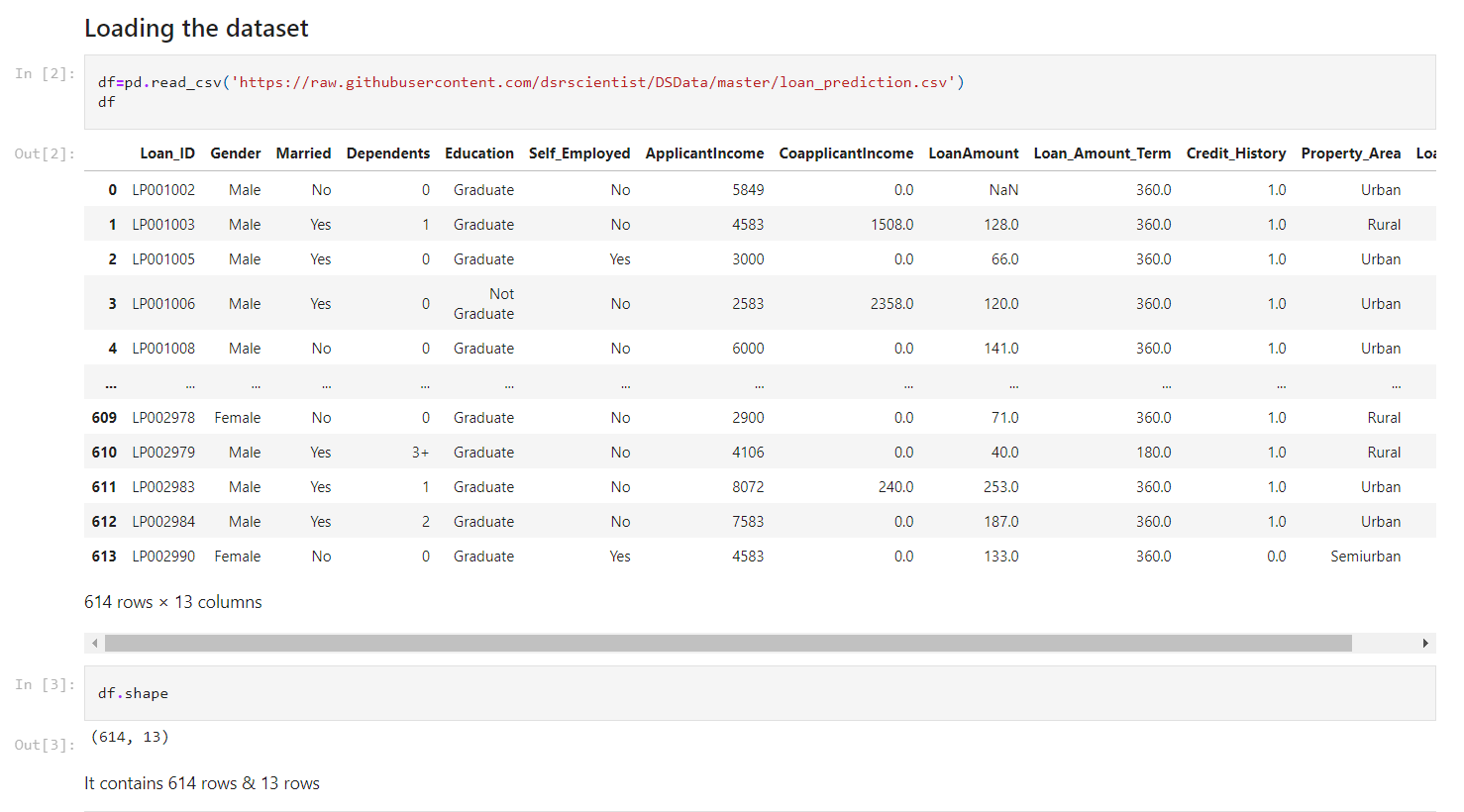
Total Number of Rows – 614

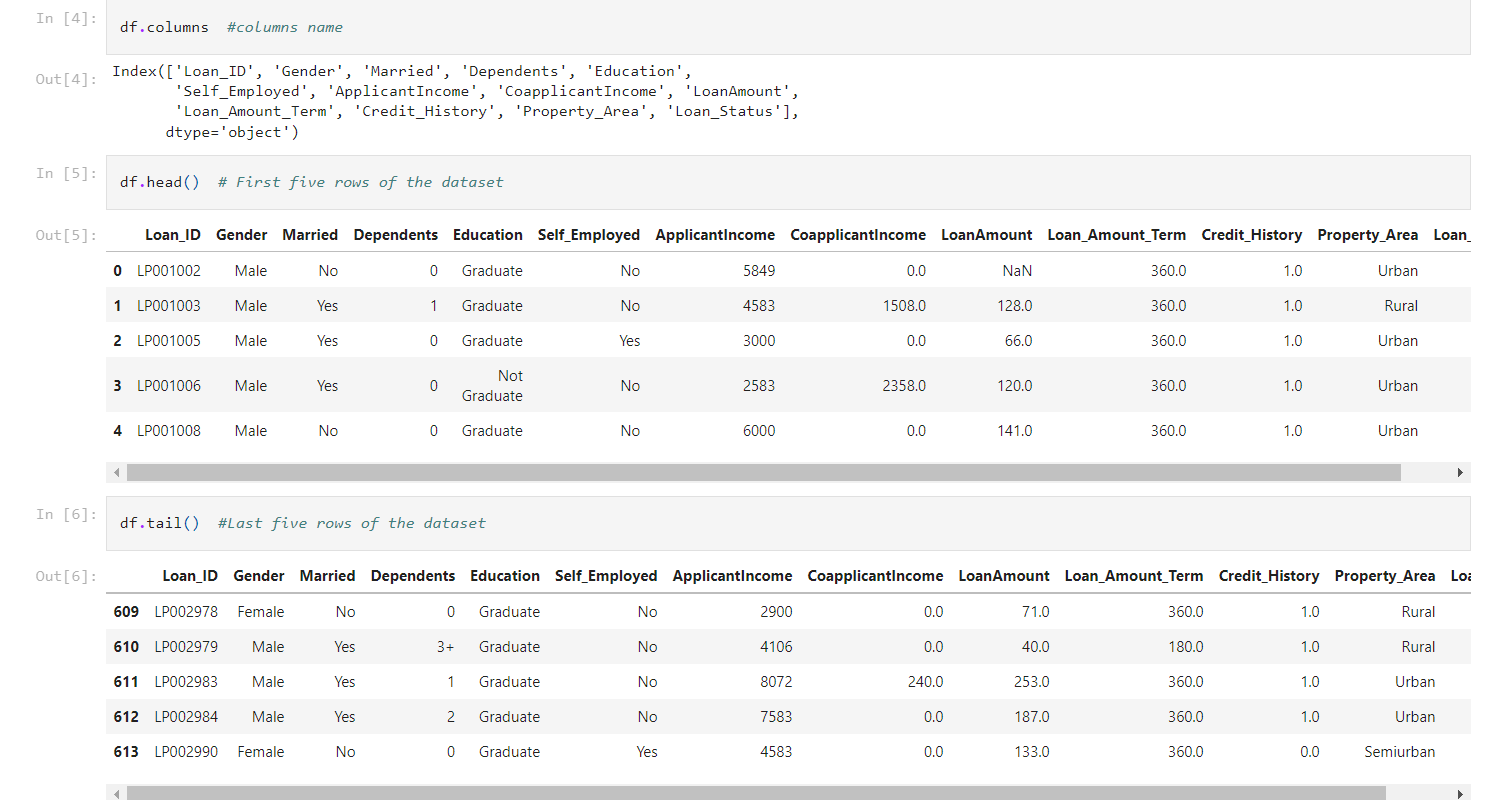
Total Number of Columns – 13

**Importing the Required Libraries**



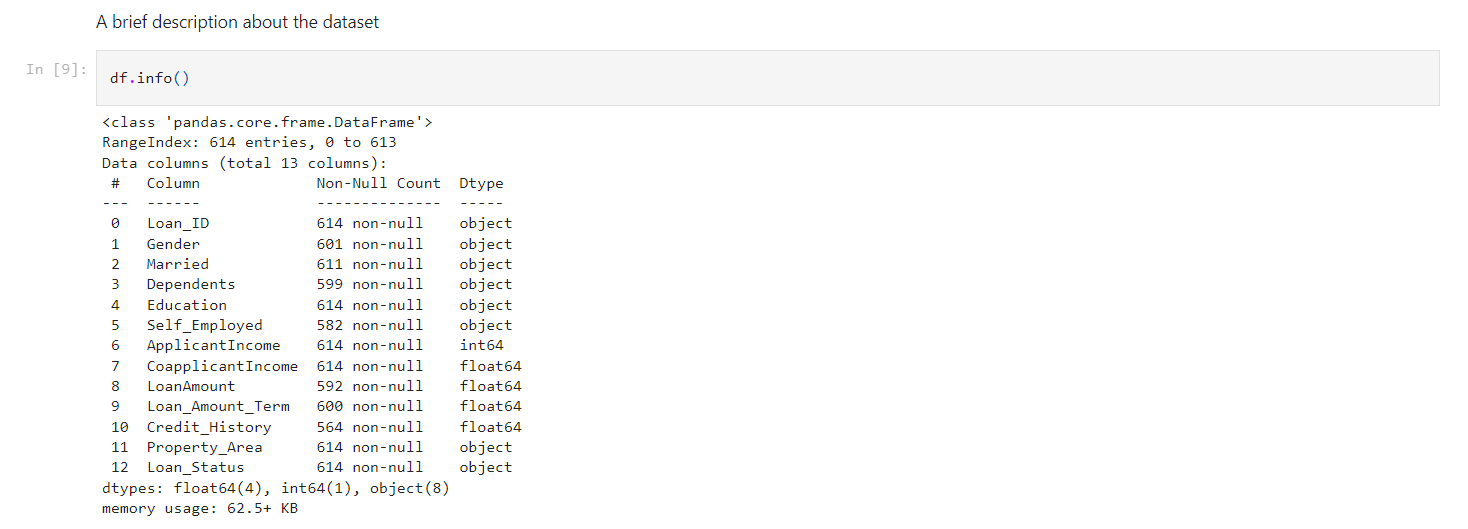
**Loading the Dataset & Verify the Shape & Columns**

****

****

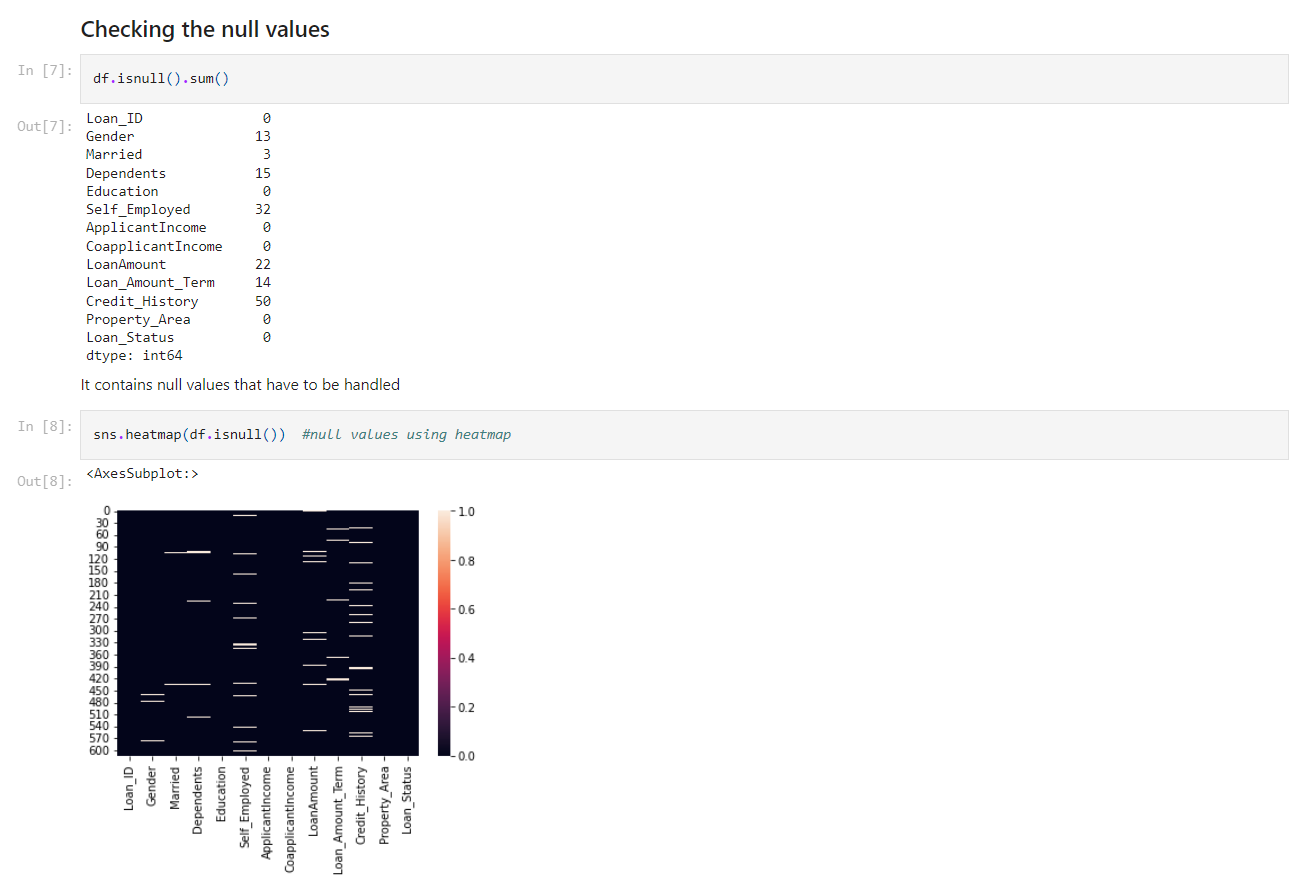
We have successfully load the dataset and verify the shape of the dataset. Also we have seen the different columns and the first five rows from the top and the bottom.

**Checking the Datatypes of Each Column**

****

Using the df.info() we get a brief idea about the dataset. There are 8 columns which have the object datatype and rough idea how many rows are non-null.

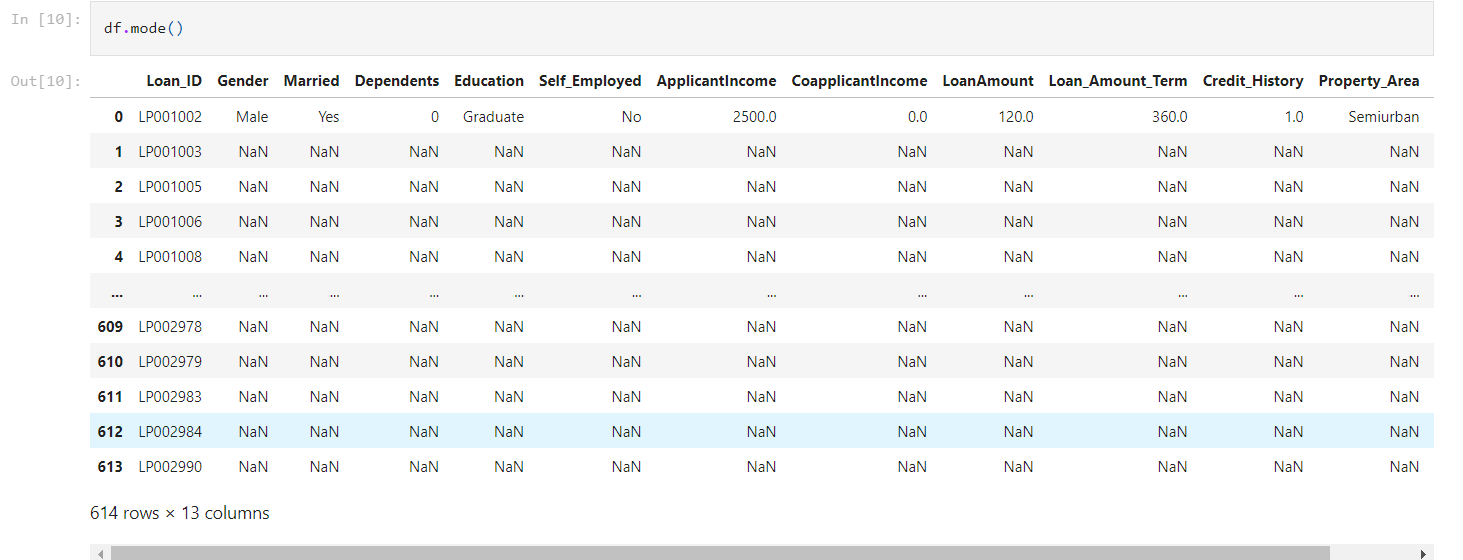
**Checking the Null Values in the Dataset**

****

Using the isnull().sum() we have calculate the total null values present in the each columns of the dataset. With the help of heatmap we try to plot the null values of the dataset. We have Seven columns having the null values which has to be handled accordingly.

**Missing Values Treatment**

Finding the mode of each column which we can use to fulfil the null values.



Now, we got the most repeated terms of each columns and they are either the single numerical value or a variable. So we can use the mode of each column to fulfil the null values.



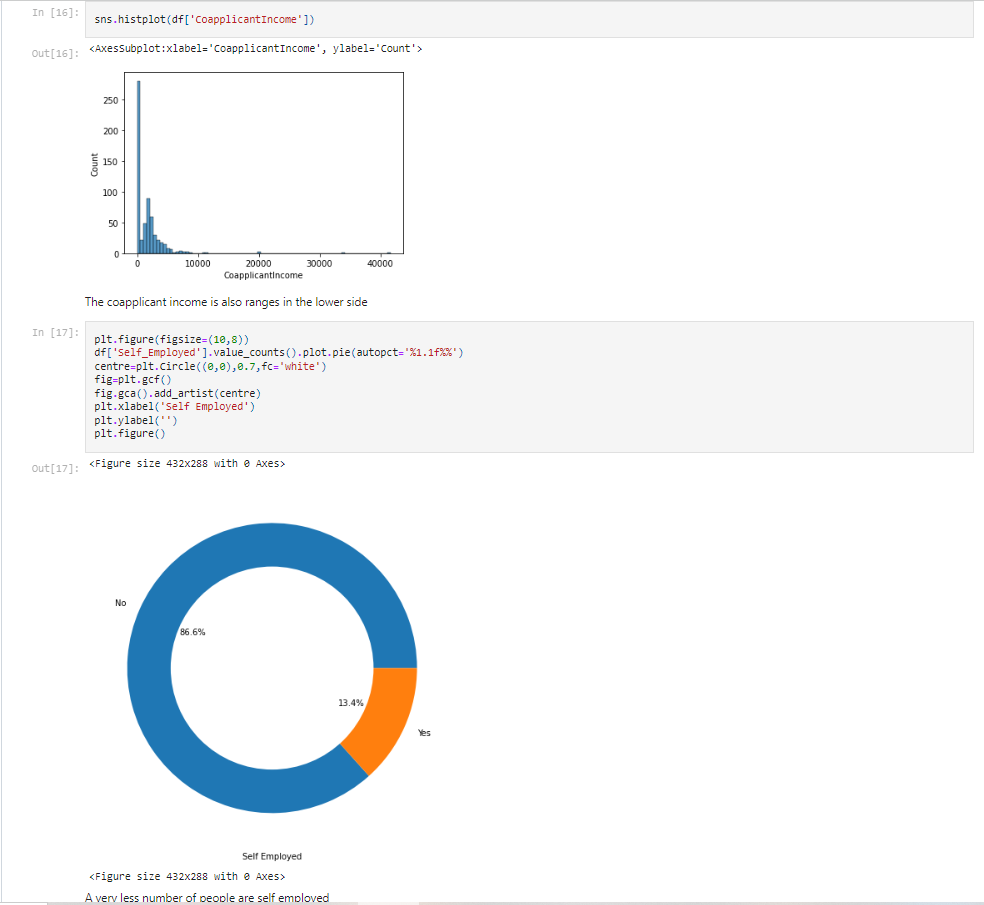
With the help of above method, we dealt with all the null values and now our dataset has no null values, so we can move with the data visualization part.

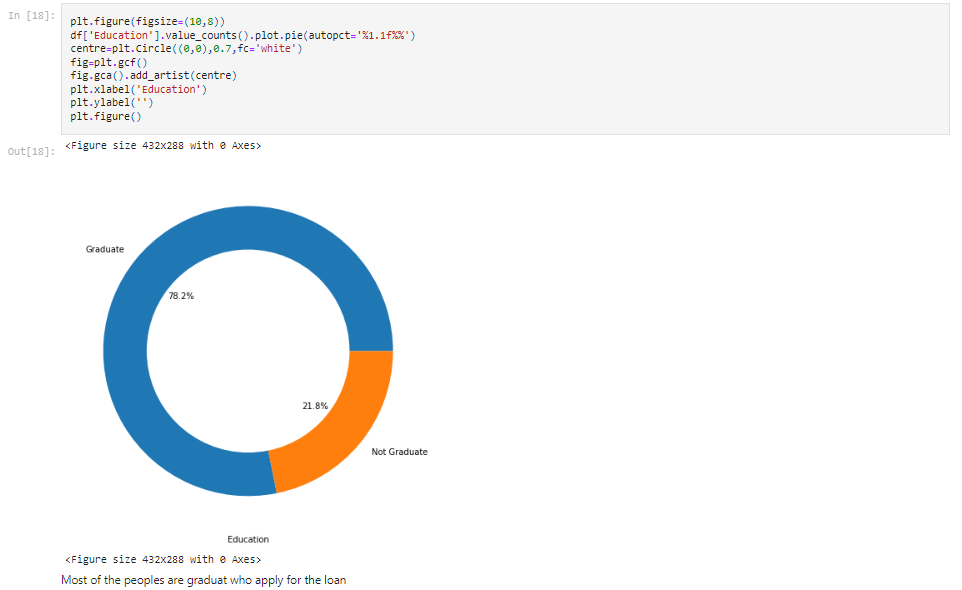
**Data Visualization**

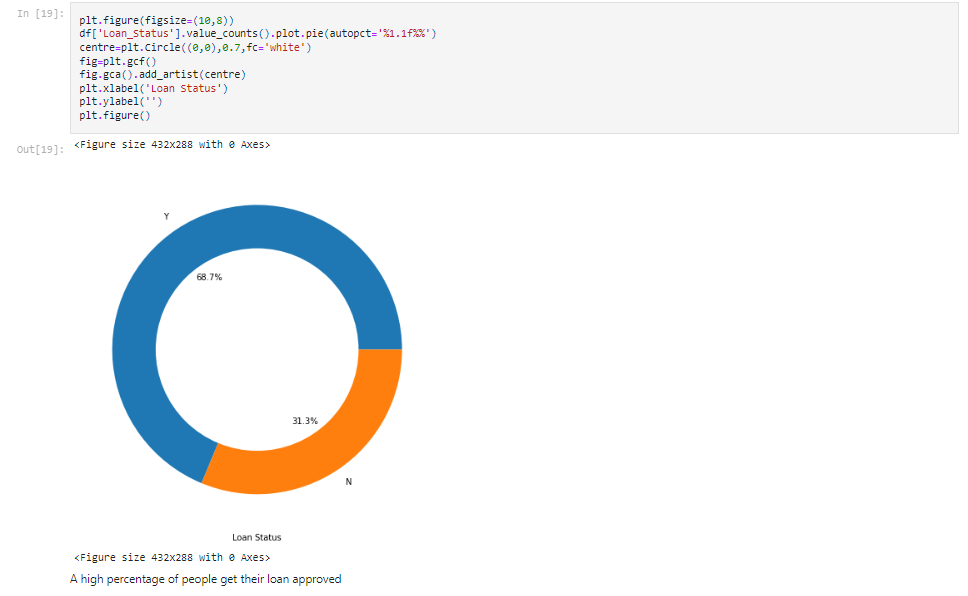
**Univariate Analysis**

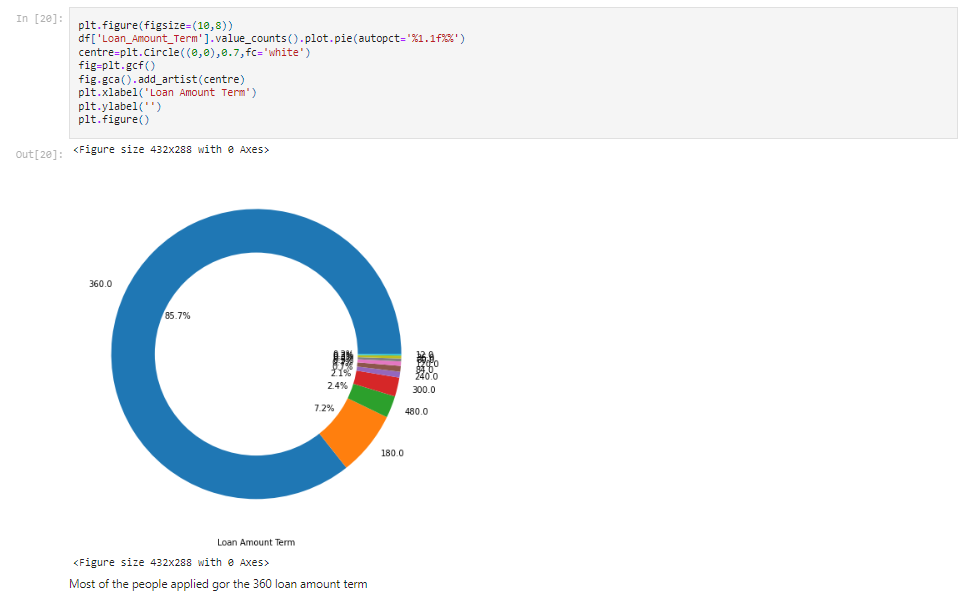
Here we see the variances in the columns one by one.







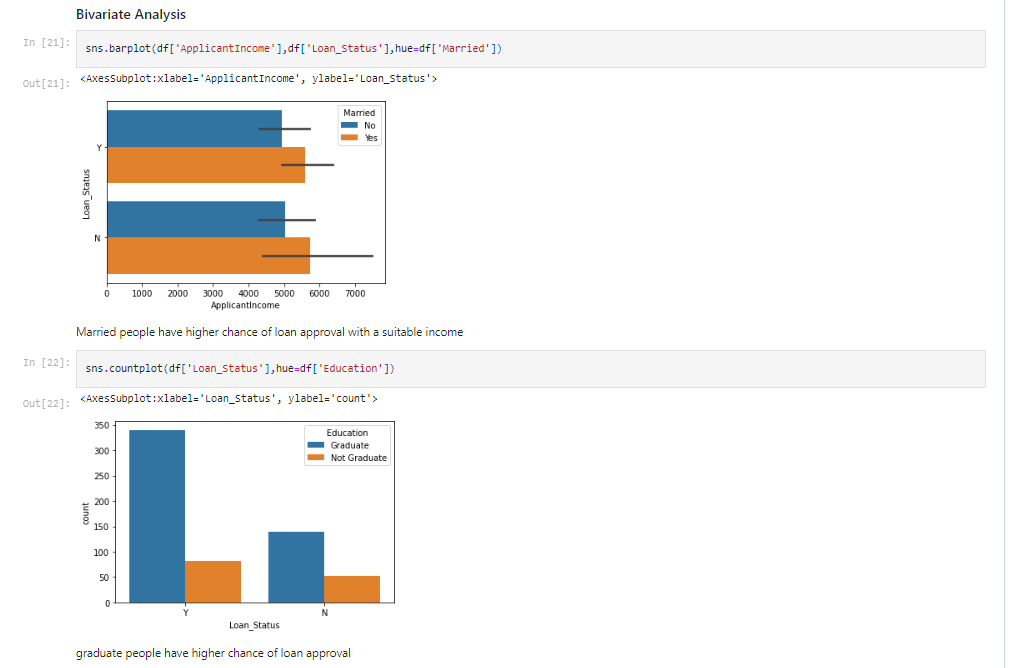




1. Many of the males are married as compared to the females who had applied for the loan.
2. Very less people have the high income, mostly ranges in the lower side.
3. The coapplicant income is also ranges in the lower side.
4. A very less number of people are self-employed i.e. more of the people are working somewhere else.
5. Most of the peoples are graduate who apply for the loan, a very less non-graduates apply for the loan.
6. A high percentage of people get their loan approved according to the provided data.
7. Most of the people go for the 360 months’ loan amount term.

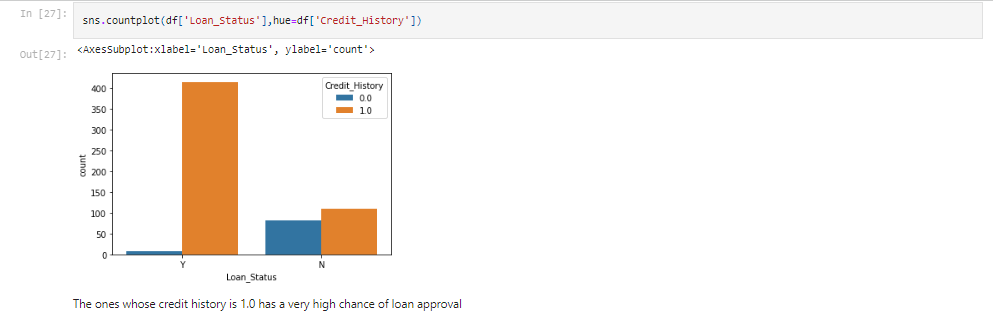
**Bivariate Analysis**

Here we will find the relationship between the different columns of the dataset using the different plots and make our observations.









1. Married people have higher chance of loan approval with a suitable income while unmarried people have almost equal number of chances for the rejection of loan.
2. Graduate people have higher chance of loan approval.
3. Male count is high for the loan approval than the females.
4. People having their loan amount term as 360 have the higher chance of loan approval.
5. Either male or female most of them have their property in Semi urban area followed by Urban.
6. Graduate income is higher than non-graduate, non-graduate has a range of only up to 20,000 but a graduate’s income can be up to 80,000
7. The ones whose credit history is 1.0 has a very high chance of loan approval

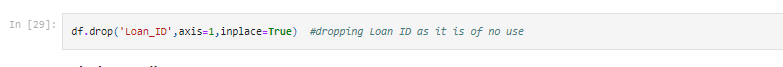
**Multivariate Analysis**

It will analyse the different columns at once. We will use the sns.pairplot() for it.



**Remove Insignificant Variable**

The column Loan\_ID contains the serial number of the Applicant, which is redundant for further analysis. Thus, we drop the column.

****

**Encoding the Categorical Variables**

We have some categorical variables in our dataset which has to be encoded for our further machine learning model. We have the label encoding ensemble technique for our encoding purpose.

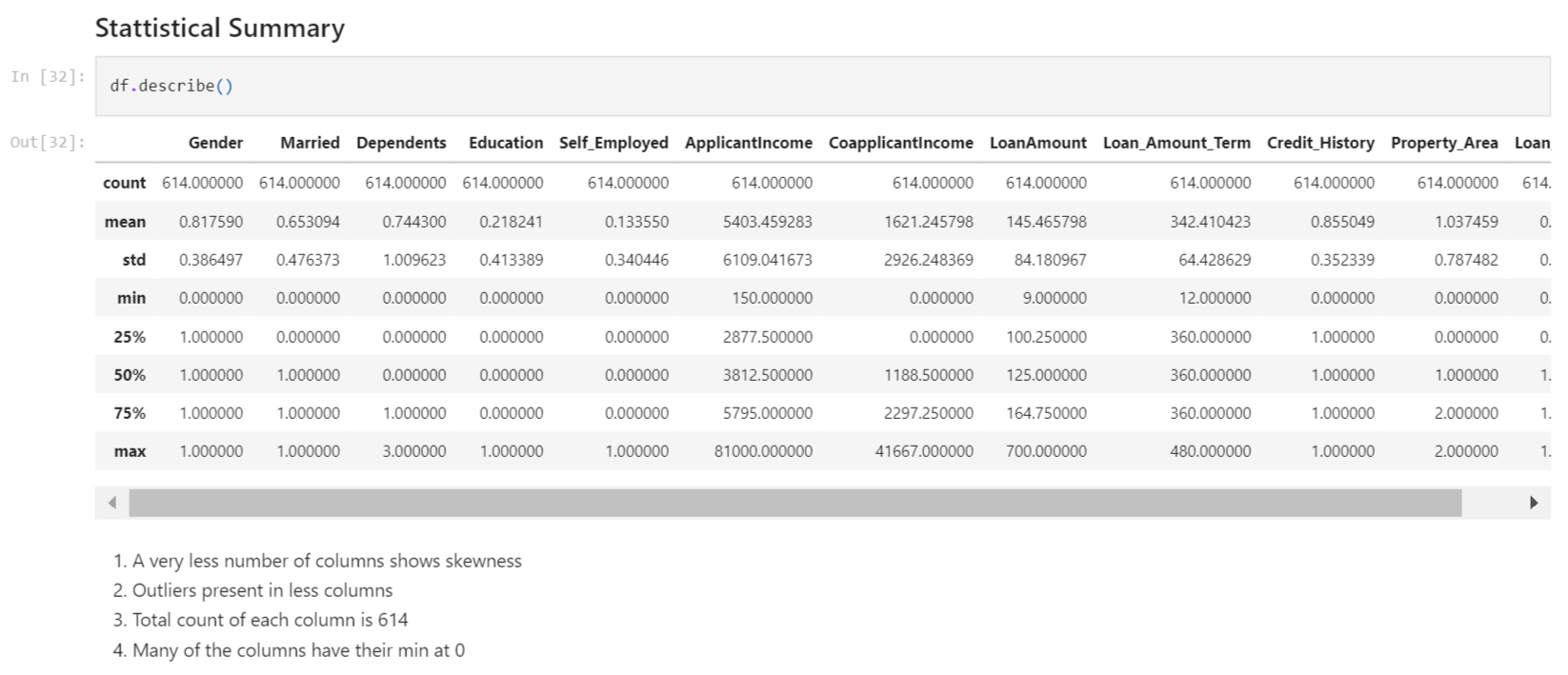


Now, each variable is either float or int.

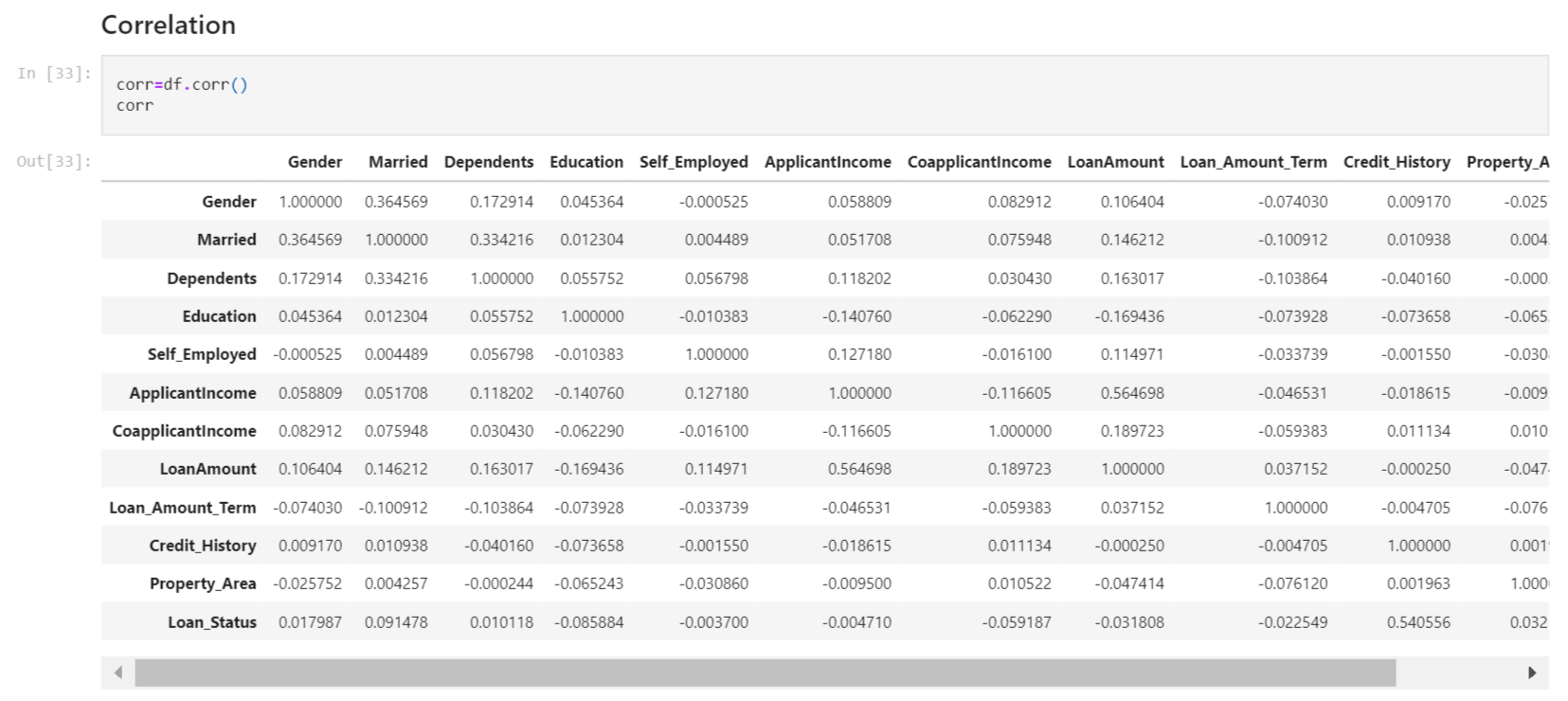
**Statistical Summary & Correlation**

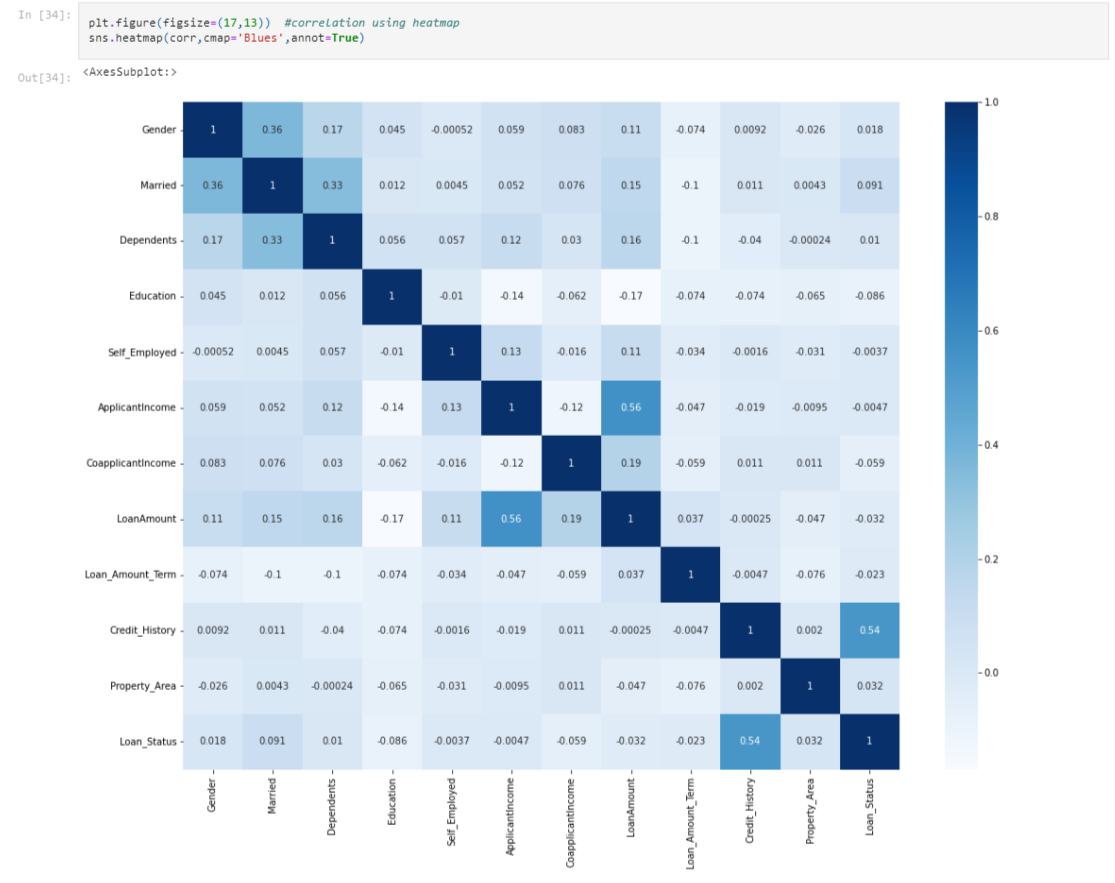
Let’s find out the Statistical Summary & Correlation of our dataset.

df.describe() will provide the statistical summary & df.corr() will find the correlation between each column (i.e. how each column is correlated with each)

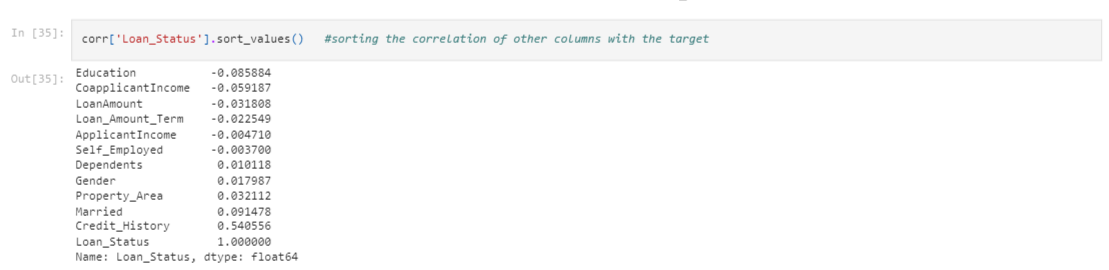


1. A very less number of columns shows skewness
2. Outliers present in less columns
3. Total count of each column is 614
4. Many of the columns have their min at 0



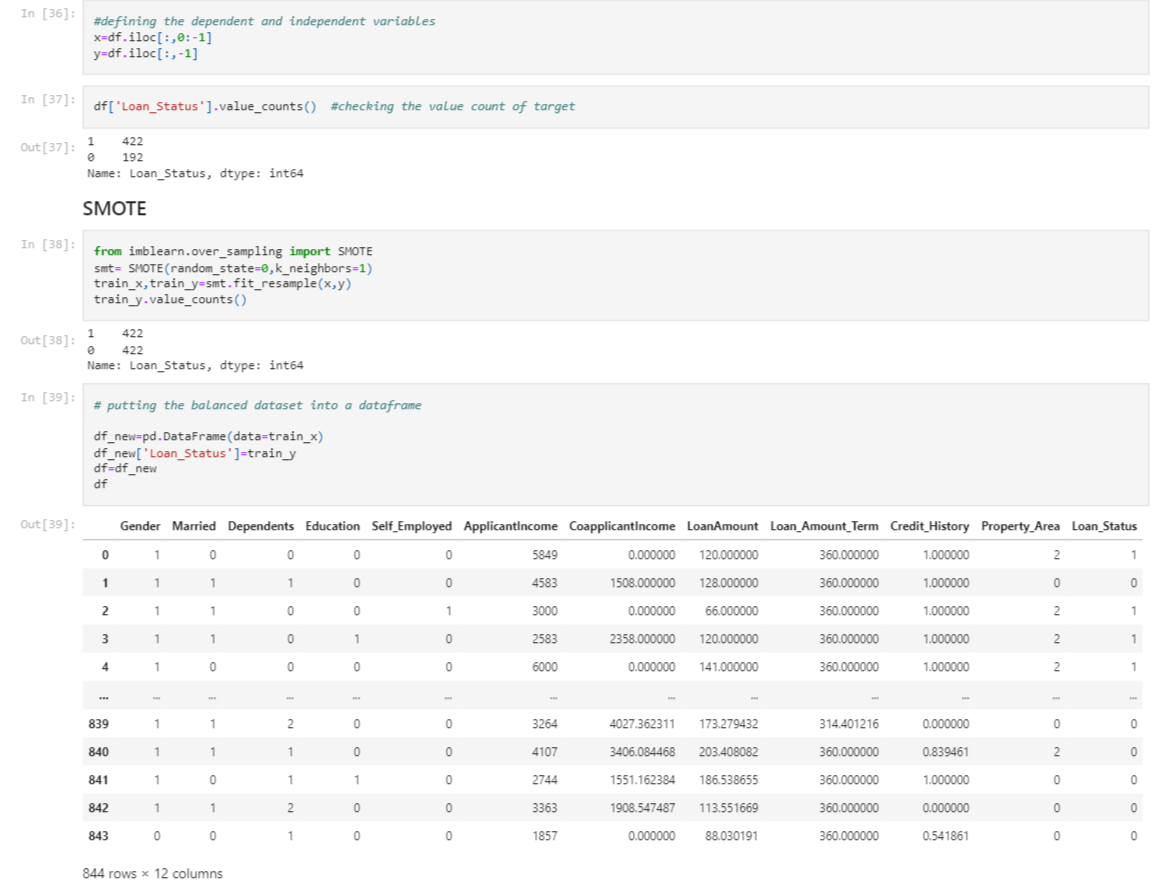


We see that the most correlate variables are Gender – Married - ApplicantIncome — LoanAmount and Credit\_History — Loan\_Status.



**SMOTE for Balancing Data**

As we see there is unbalancing in the target variable, so we will use SMOTE for the oversampling.

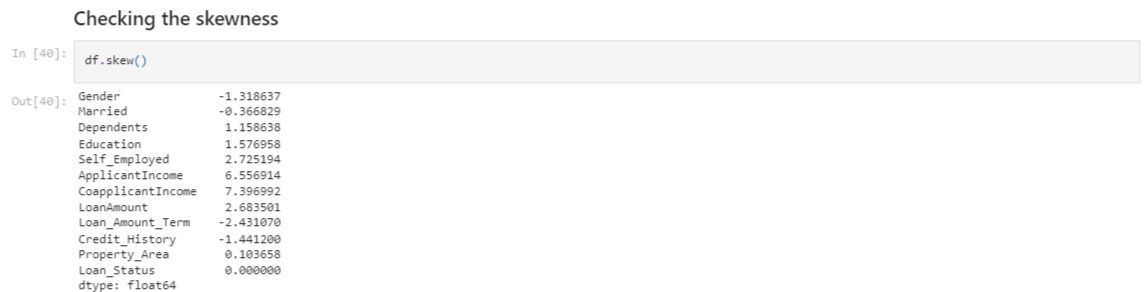


Now the total rows are 844.

**Skewness in the Dataset**

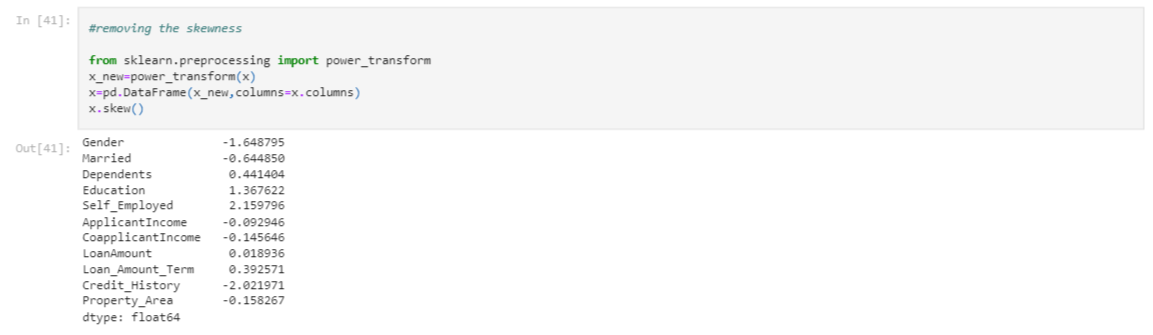
There are two types of skewness one is left skewness and another one is right skewness. It means either the data is left shifted of the right shifted while we plot it. A good graph is Normal distributed graph where mean=median=mode.

**Checking the Skewness**

****

**Removing Skewness**

Skewness can be removed by using the power transform method. We will get the data which is very less skewed.

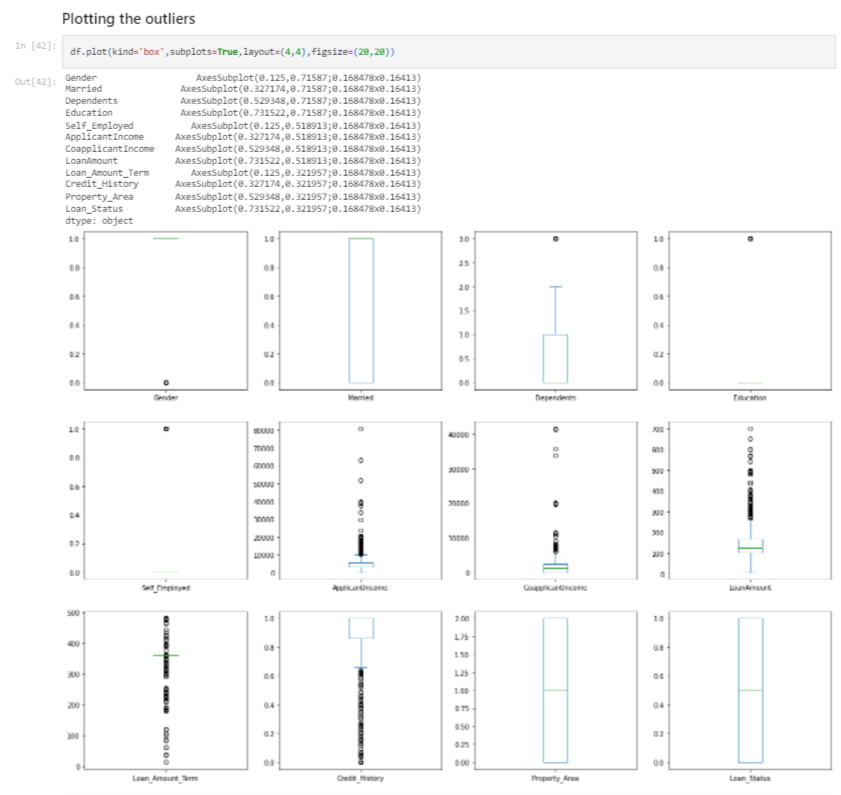


Hence the skewness is removed from the dataset.

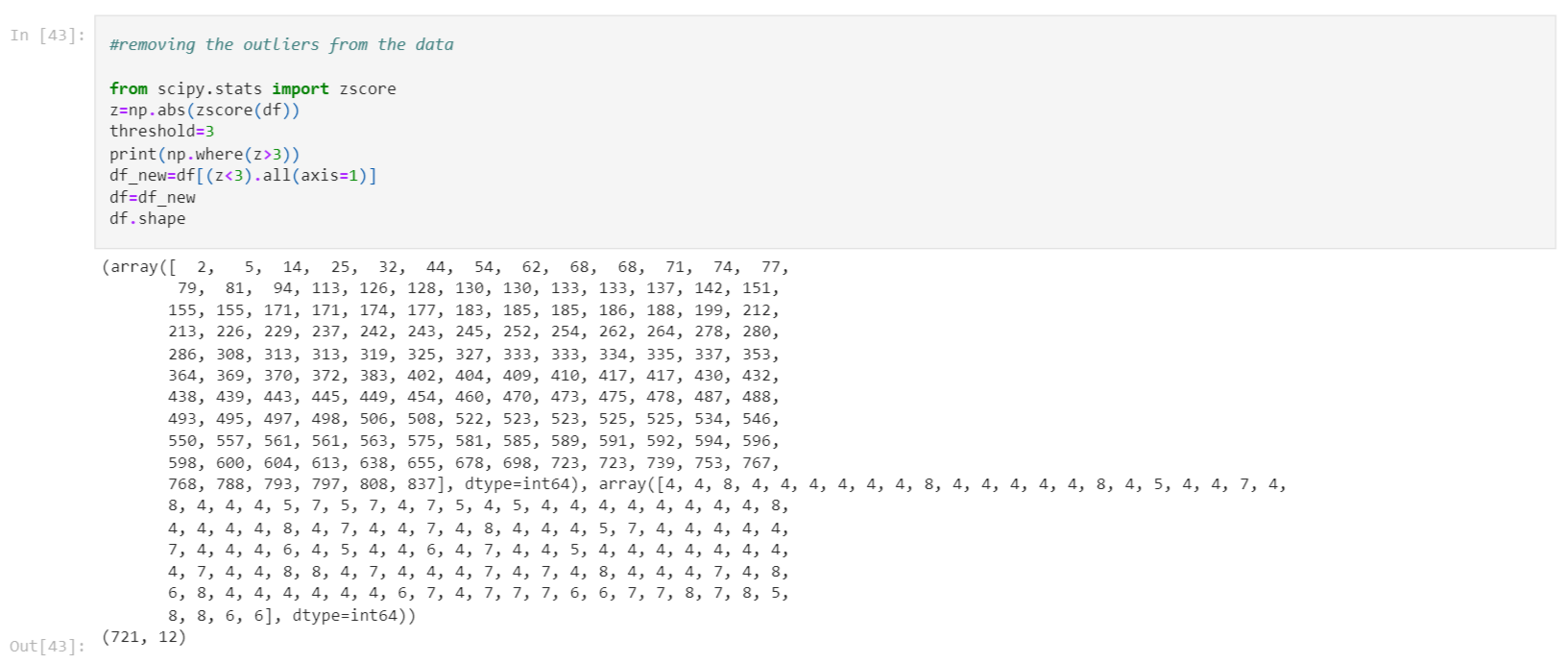
**Outliers in the Dataset**

We have some outliers present in the dataset, so let’s handle them also. As the outliers in the dataset will affect our ML model. We need to remove all the outliers present in the dataset.

There is something called zscore which indicates how many standard deviations away an element is from the mean. We consider the points as outliers whose zscore is above 3 or less than -3. So we need to remove all such points from our dataset.



Using boxplot we get the idea of the outliers present in the dataset.



Using the threshold, we have removed all the points where the zscore is greater than 3. Now the total number of rows after removing the outliers are 721.

**MODEL BUILDING**

We will import important libraries for the building the ML model and defining the different models for our easiness.

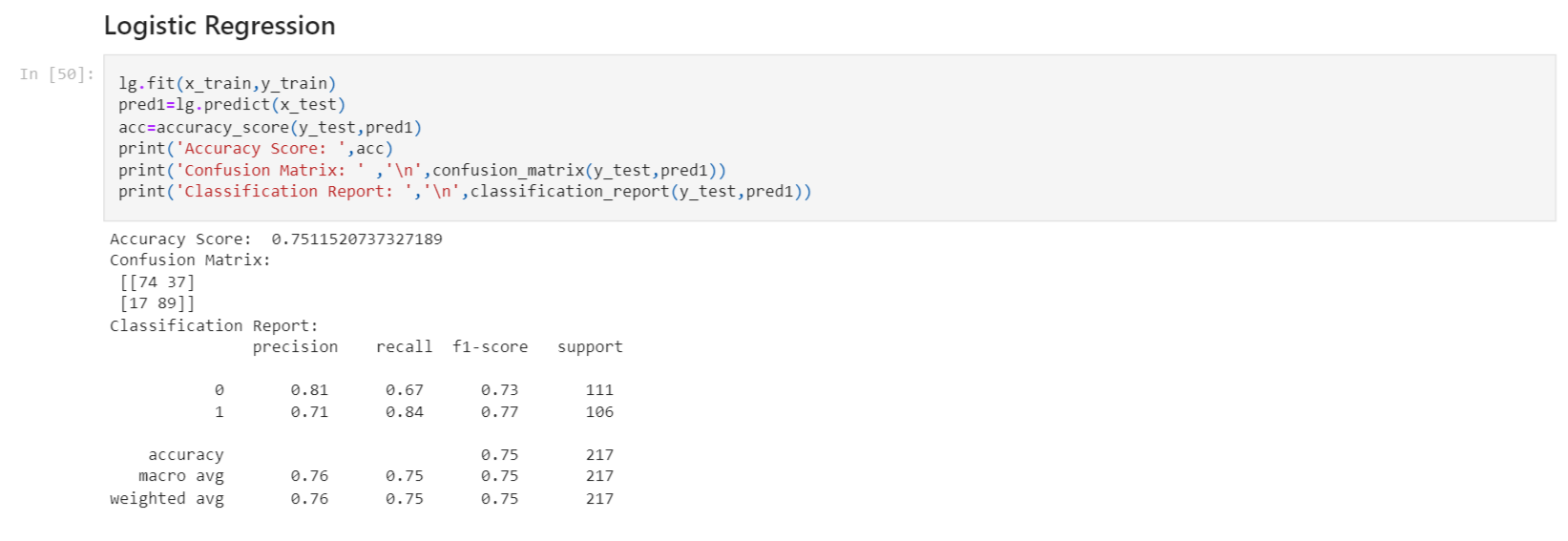
****

**Finding the Best Random State & Train Test Split**

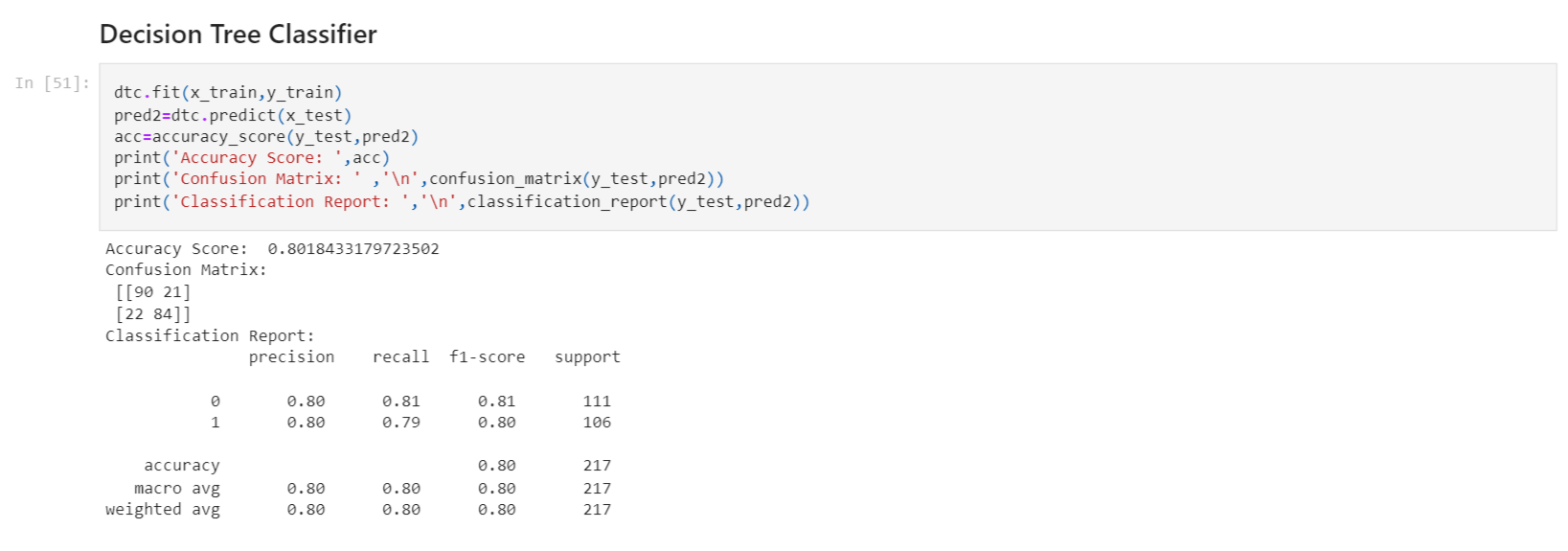
Finding the best random state for the train test split.



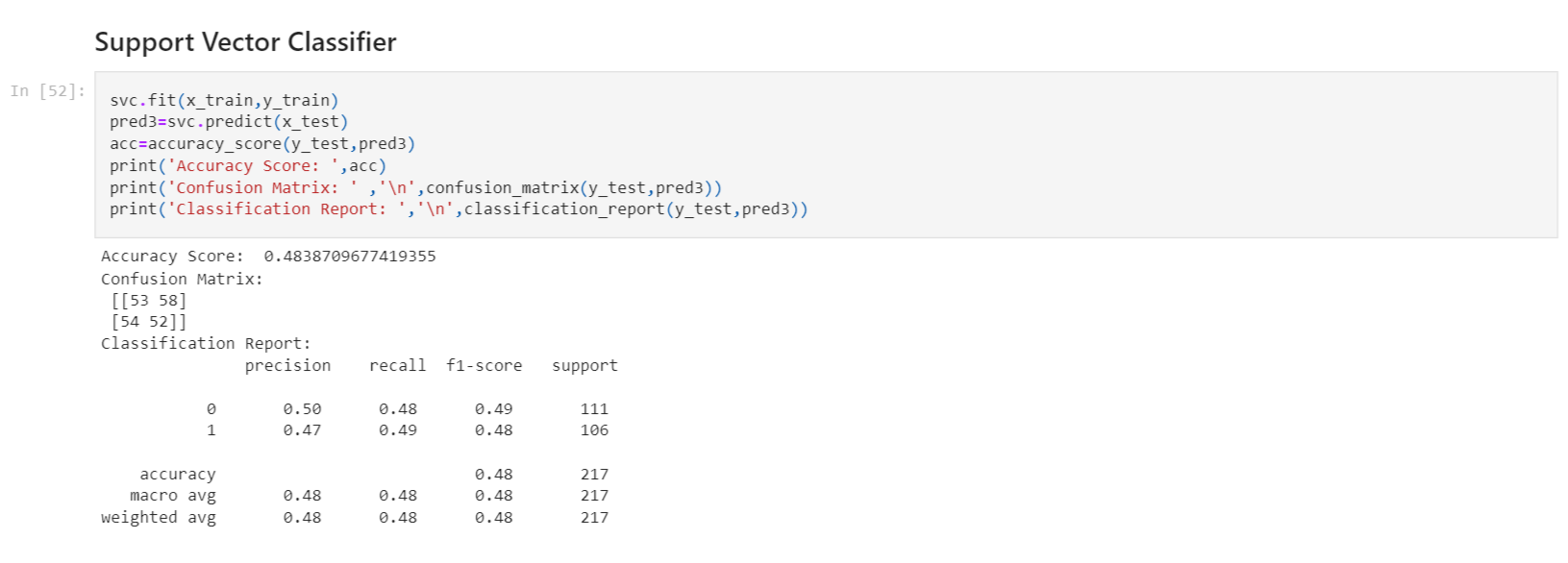
**Logistic Regression**

****

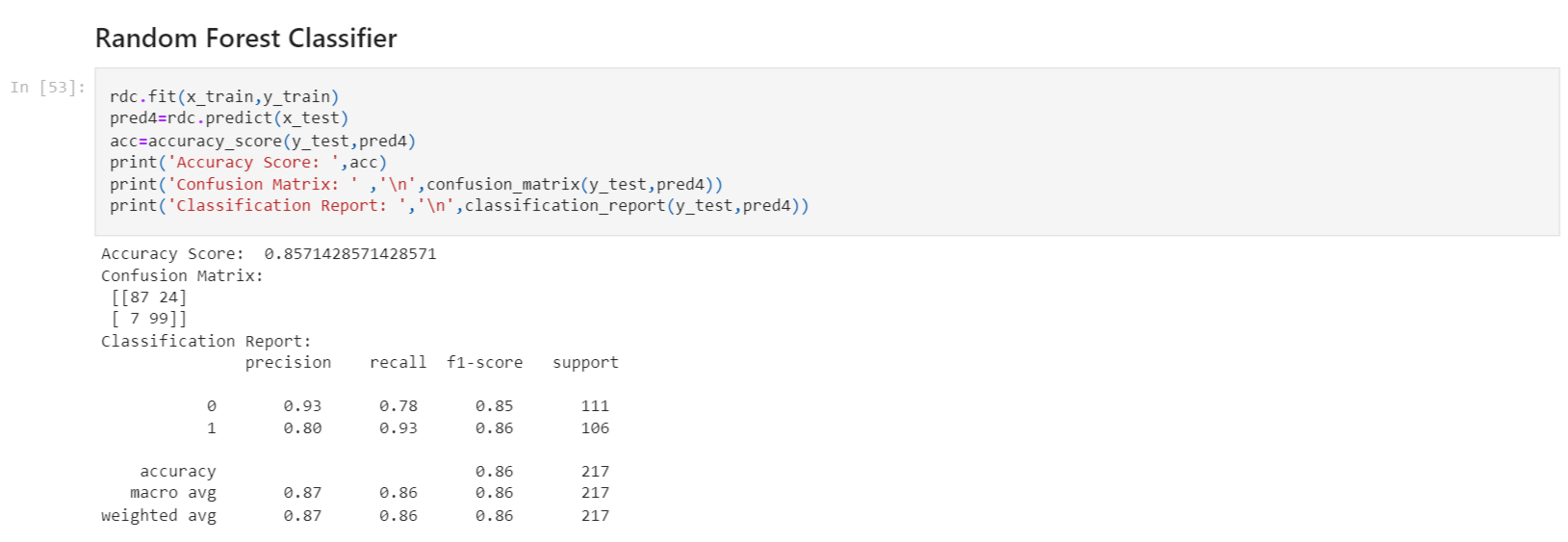
**Decision Tree Classifier**

****

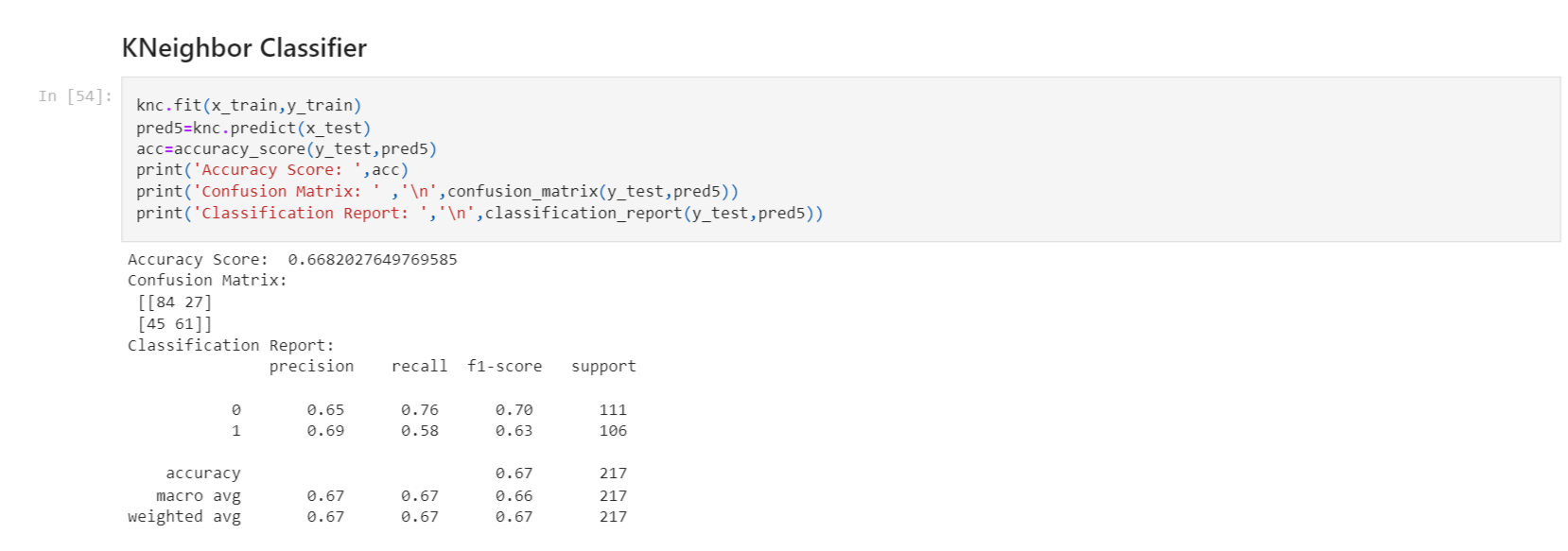
**Support Vector Classifier**

****

**Random Forest Classifier**

****

**KNeighbor Classifier**

****

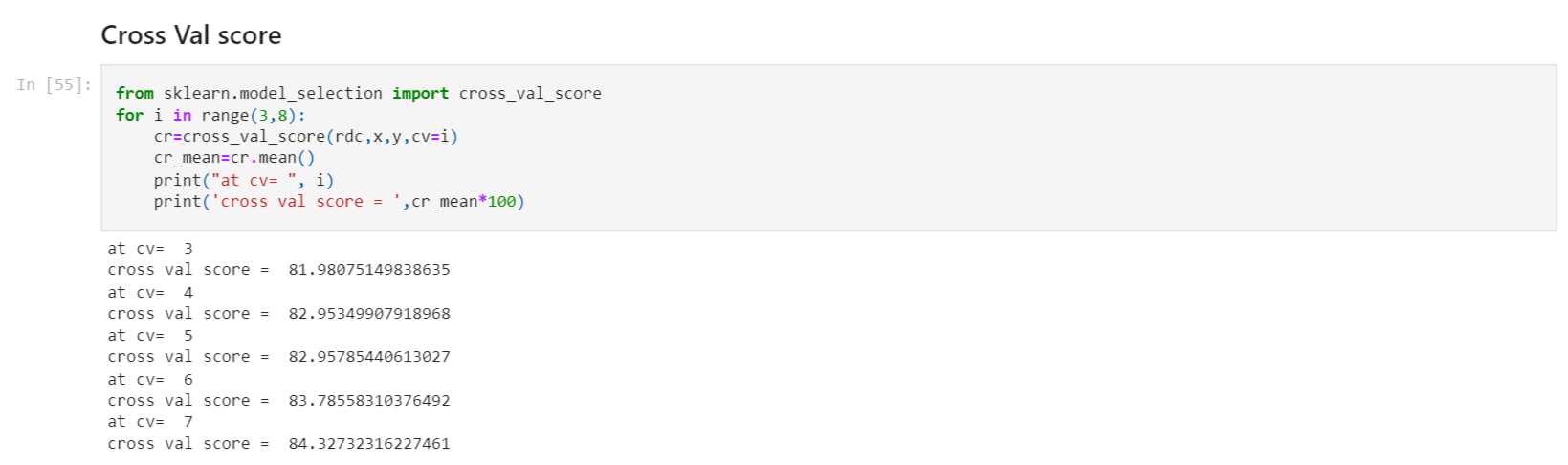
**Accuracy of Each Model**

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Model** | **Accuracy Score** |
| 1 | Logistic Regression | 0.7511520737327189 |
| 2 | Decision Tree Classifier | 0.8018433179723502 |
| 3 | Support Vector Classifier | 0.4838709677419355 |
| 4 | Random Forest Classifier | 0.8571428571428571 |
| 5 | KNeighbor Classifier | 0.6682027649769585 |

Hence, we are getting the best accuracy score through the Random Forest Classifier Model. We will go ahead with this to find the cross val score and hypermeter tuning.

**Cross Val Score**

Cross-validation provides information about how well a classifier generalizes, specifically the range of expected errors of the classifier. Cross Val Score tells how the model is generalized at a particular cross validation.



At CV=7 we get the best results i.e. the Random Forest Classifier more generalized at cv=7, so we calculate the hyper parameters at this value.

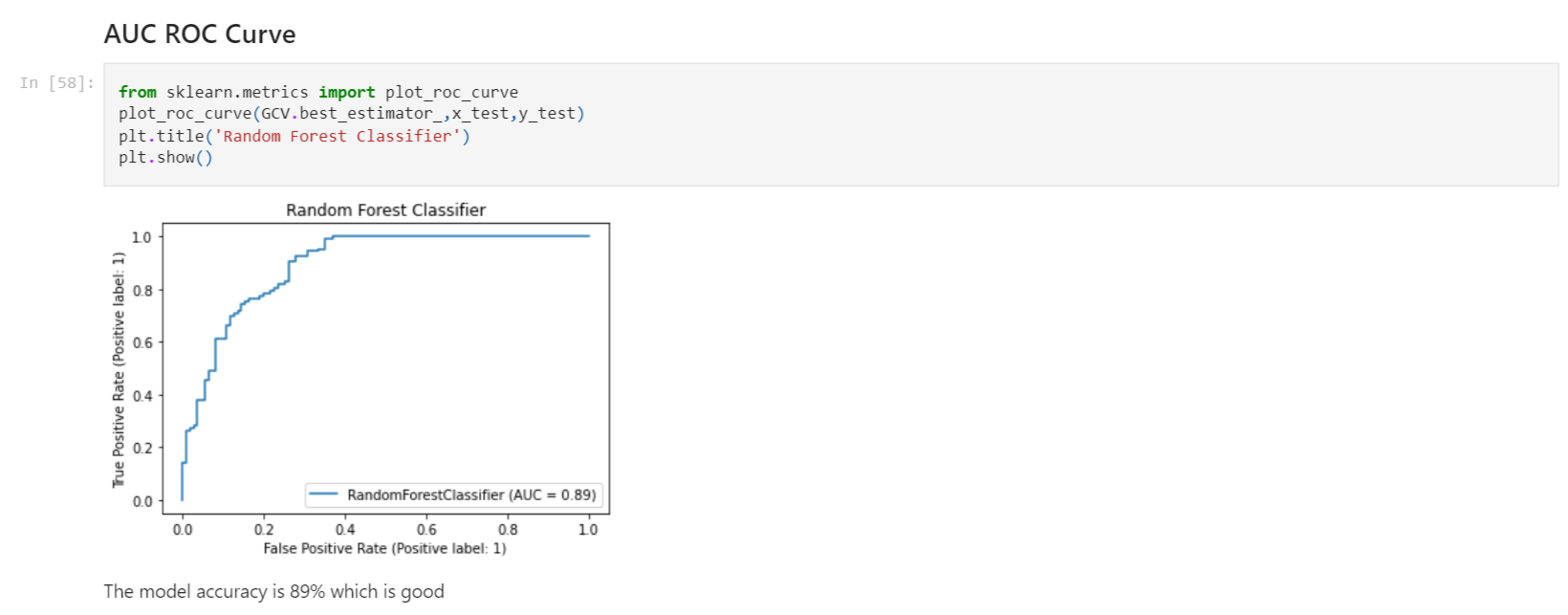
**Hyperparameter Tuning**

We will find which parameters of random forest classifier are the best foe our model. We will do this using Grid Search CV method & also calculate the accuracy score at those best parameters.



**AUC ROC Curve**

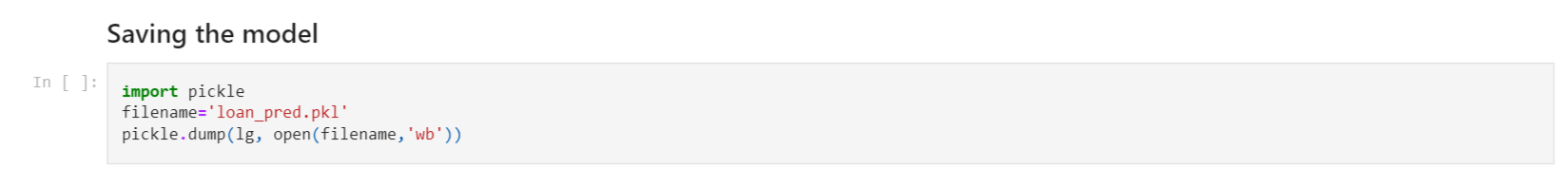
Let’s plot the AUC ROC curve for the Random Forest Classifier & find the accuracy of our model.



Here we are getting the accuracy of the model as 89% which is pretty good. So, we will save this model for the future predictions.

**Saving the Model**

Saving the best model – Random Forest Classifier in this case for future predictions.



**Conclusion**

Let’s see what are the actual test data and what our model predicts.



Hence up to some good extensions our model predicted so well.