***Prediction of Loan Application Status***

***Problem Statement:***

There is a company named Dream Housing Finance that deals in all home loans. They have presence across all urban, semi urban and rural areas. Customer first apply for home loan after that company validates the customer eligibility for loan. However doing this manually takes a lot of time. Hence it wants to automate the loan eligibility process (real time) based on customer information

So the final thing is to identify the factors/ customer segments that are eligible for taking loan. How will the company benefit if we give the customer segments is the immediate question that arises. The solution is ….Banks would give loans to only those customers that are eligible so that they can be assured of getting the money back. Hence the more accurate we are in predicting the eligible customers the more beneficial it would be for the Dream Housing Finance Company.

**TYPE OF PROBLEM:**

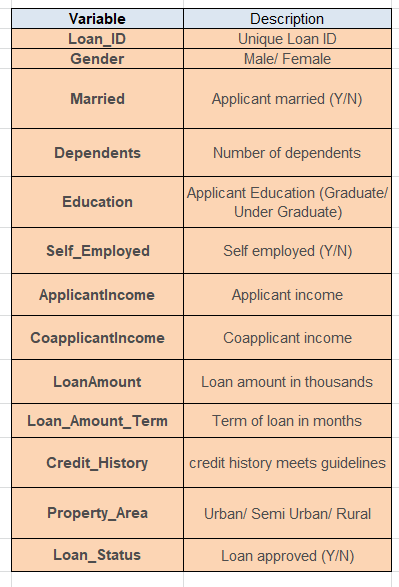
The above problem is a clear classification problem as we need to classify whether the Loan\_Status is yes or no. So this can be solved by any of the classification techniques like

1. Logistic Regression .
2. Decision Tree Algorithm.
3. Random Forest Technique.

I have mentioned only few. We will be dealing with each of techniques later in this blog.

# Description about the Data Columns:

It’s very useful to know about the data columns before getting in to the actual problem for avoiding confusion at a later state. Now let us understand the data columns (that has been already given by the company itself ) first so that we will get a glance.



There are altogether 13 columns in our data set. Of them Loan\_Status is the response variable and rest all are the variables /factors that decide the approval of the loan or not.

Now let us look in to the each variable and can make some assumptions.(It’s just assumptions right, there is no harm in just assuming few statements)

Loan ID -> As the name suggests each person should have a unique loan ID.

Gender -> In general it is male or female. No offence for not including the third gender.

Married -> Applicant who is married is represented by Y and not married is represented as N. The information regarding whether the applicant who is married is divorced or not has not been provided. So we don’t need to worry regarding all these.

Dependents -> the number of people dependent on the applicant who has taken loan has been provided.

Education -> It is either non -graduate or graduate. The assumption I can make is “ The probability of clearing the loan amount would be higher if the applicant is a graduate”.

Self\_Employed -> As the name suggests Self Employed means , he/she is employed for himself/herself only. So freelancer or having a own business might come in this category. An applicant who is self employed is represented by Y and the one who is not is represented by N.

Applicant Income -> Applicant Income suggests the income by Applicant.So the general assumption that i can make would be “The one who earns more have a high probability of clearing loan amount and would be highly eligible for loan ”

Co Applicant income -> this represents the income of co-applicant. I can also assume that “ If co applicant income is higher , the probability of being eligible would be higher “

Loan Amount -> This amount represents the loan amount in thousands. One assumption I can make is that “ If Loan amount is higher , the probability of repaying would be lesser and vice versa”

Loan\_Amount\_Term -> This represents the number of months required to repay the loan.

Credit\_History -> When I googled it , I got this information. A credit history is a record of a borrower’s responsible repayment of debts. It suggests → 1 denotes that the credit history is good and 0 otherwise.

Property\_Area -> The area where they belong to is my general assumption as nothing more is told. Here it can be three types. Urban or Semi Urban or Rural

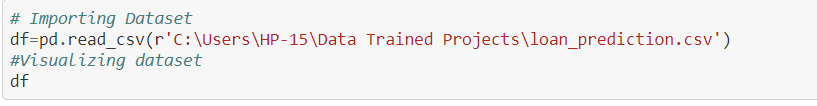
Loan\_Status -> If the applicant is eligible for loan it’s yes represented by Y else it’s no represented by N.

***Data Analysis:***

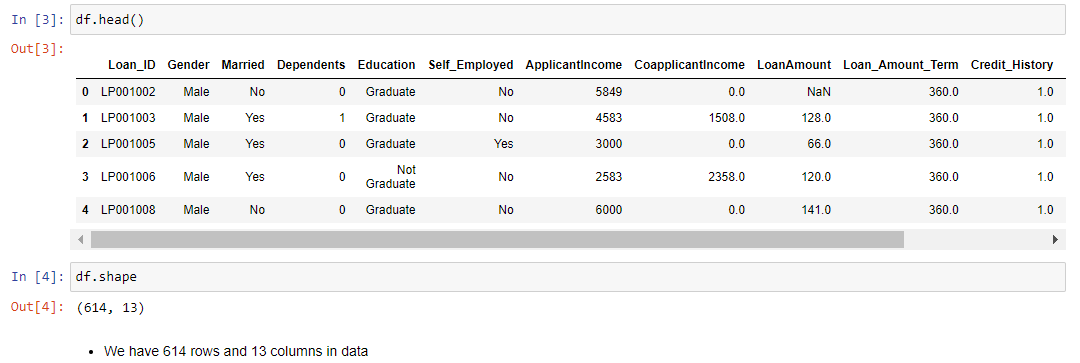
Now let me walk through the code. Firstly I just imported the necessary packages like pandas, numpy, seaborn etc. so that i can carry the necessary operations further.



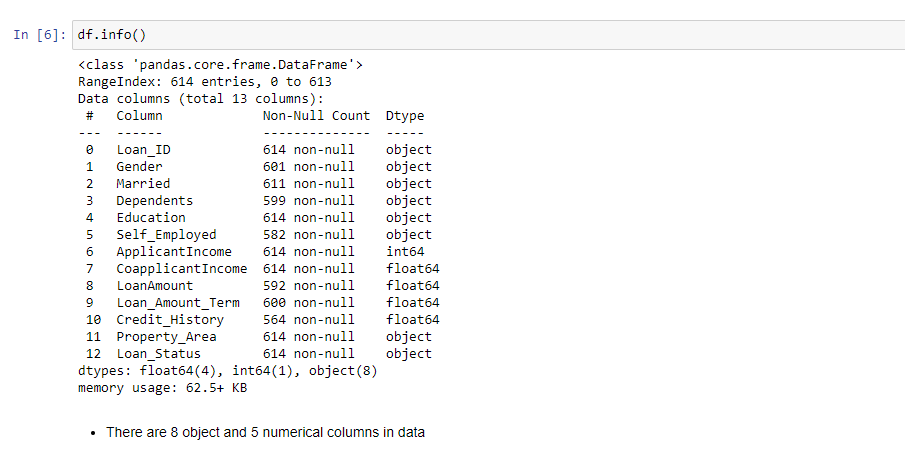
Now I am going to upload or read the files/data-sets using pandas. For this we used read\_csv



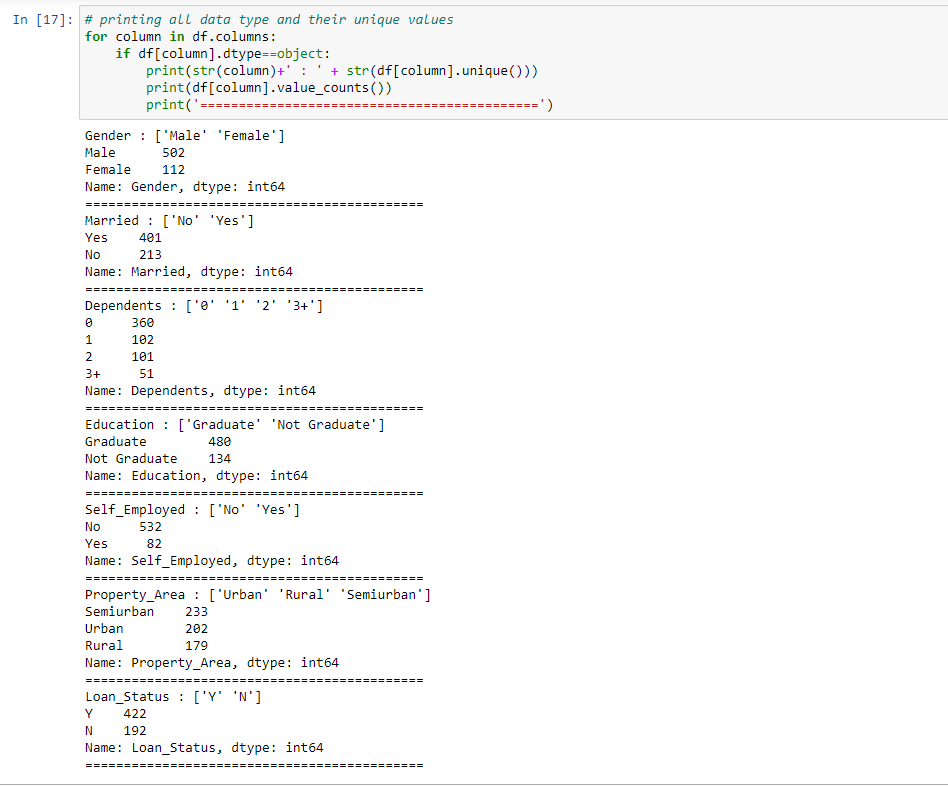
Let me get the top 5 values. We can get using the head function. Hence the code would be df.head().



­ Checking all variable datatype using df.info()



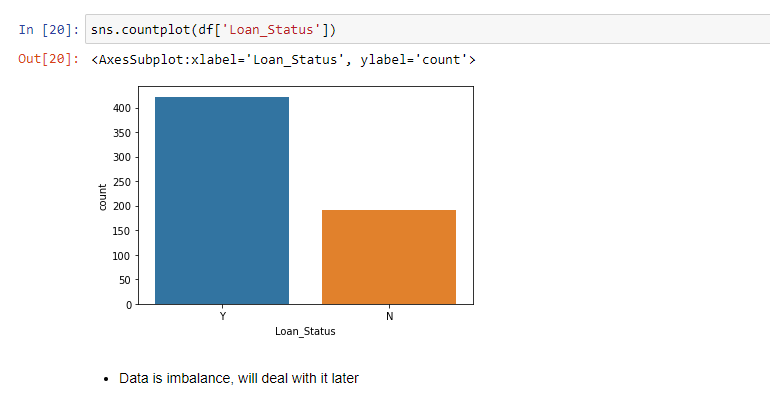
Now let us analyse the data using single variable.



**Conclusions**: (Through Single Variable Analysis)

1. We can see that approximately 81% are Male and 19% are female.
2. Percentage of applicants with no dependents is higher.
3. There are more number of graduates than non graduates.
4. Semi Urban people is slightly higher than Urban people among the applicants.
5. Larger Percentage of people have a good credit history.
6. The percentage of people that the loan has been approved has been higher rather than the percentage of applicant for which the loan has been declined.

Now let me check the target variable



We can notice we large number of data for loan approved and less data in declines loan, will have to deal with problem in data imbalance.

**Exploratory Data Analysis:**

Well don’t get to worry about the fancy names like exploratory data analysis and all. By looking at the columns description in the above paragraph, we can make many assumptions like

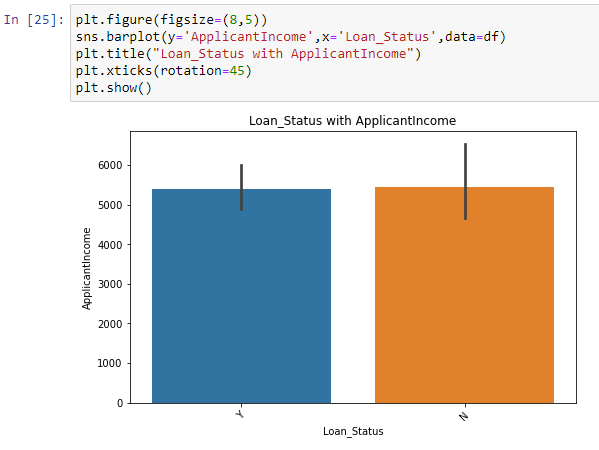
1. The one whose salary is more can have a greater chance of loan approval.
2. The one who is graduate has a better chance of loan approval.
3. Married people would have a upper hand than unmarried people for loan approval .
4. The applicant who has less number of dependents have a high probability for loan approval.
5. The lesser the loan amount the higher the chance for getting loan.

Why are we doing EDA?

Like these there are many more we can assume. But one basic question you may get it …”Why are we doing all these ? Why can’t we do directly modeling the data instead of knowing all these…..” Well in some cases we can easily come to conclusion if we just to do EDA. Then there is no necessary for going through next models.

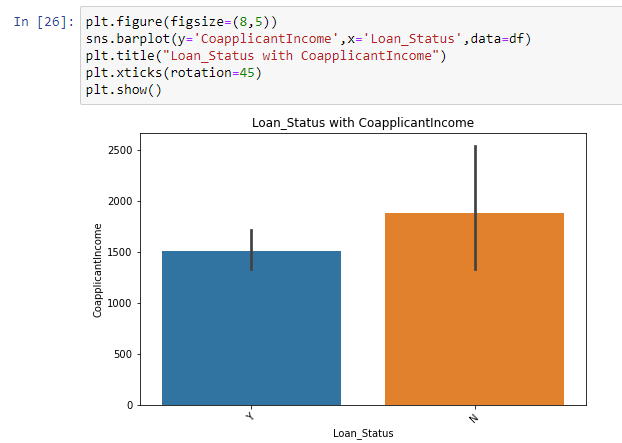
Lets import one library for visualization



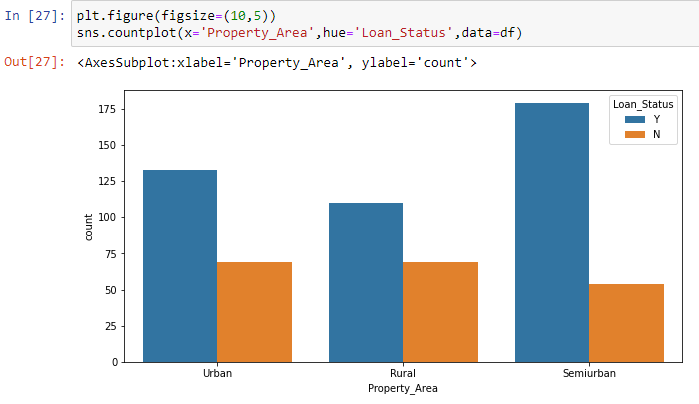


**Conclusions**:

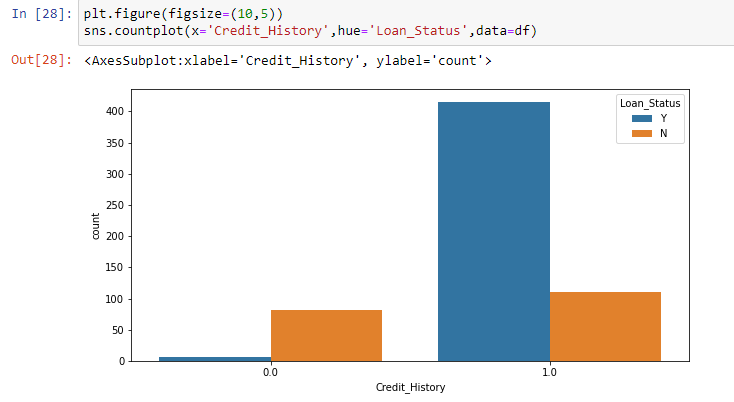
Applicant income is not deciding the loan status, whether customer should get the loan or not



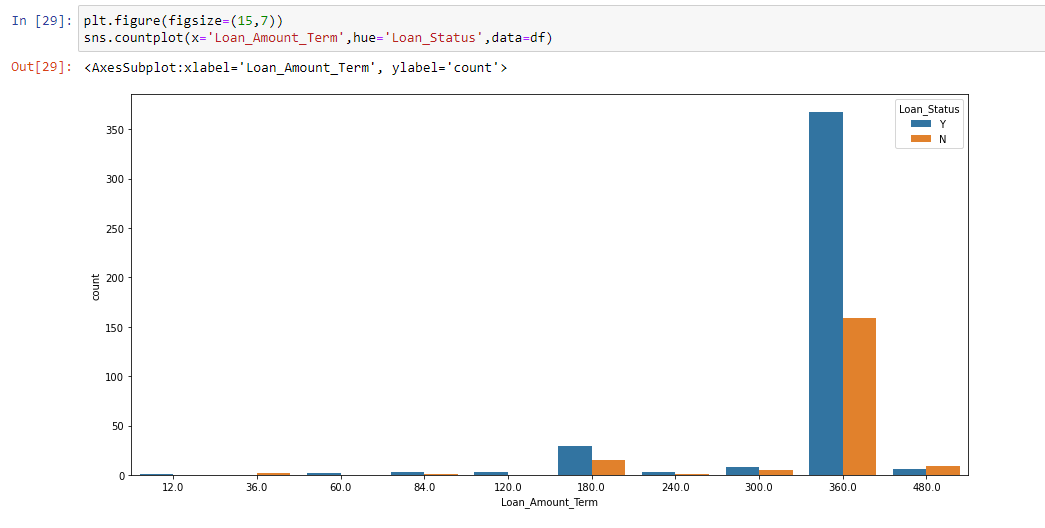
Here we notice some interesting figure there are chances not to be approved doesn’t matter if Coapplicant Income is high might be that’s depends on all other figures



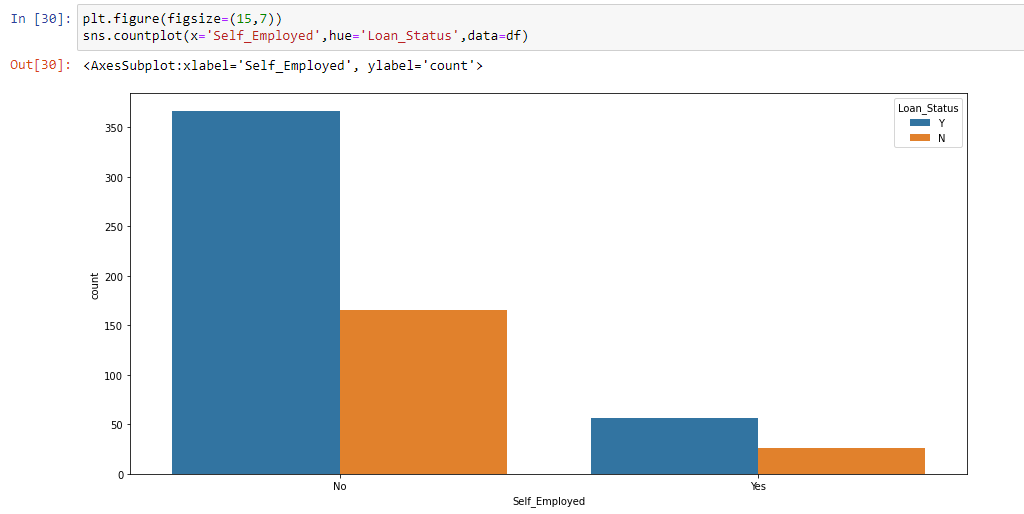
* People those are from Urban area having 50-50 chances of approval depends and other parameters as well
* In RuralProperty\_area not approval chances are increasing
* In Semiurban area there are high chances to get approval



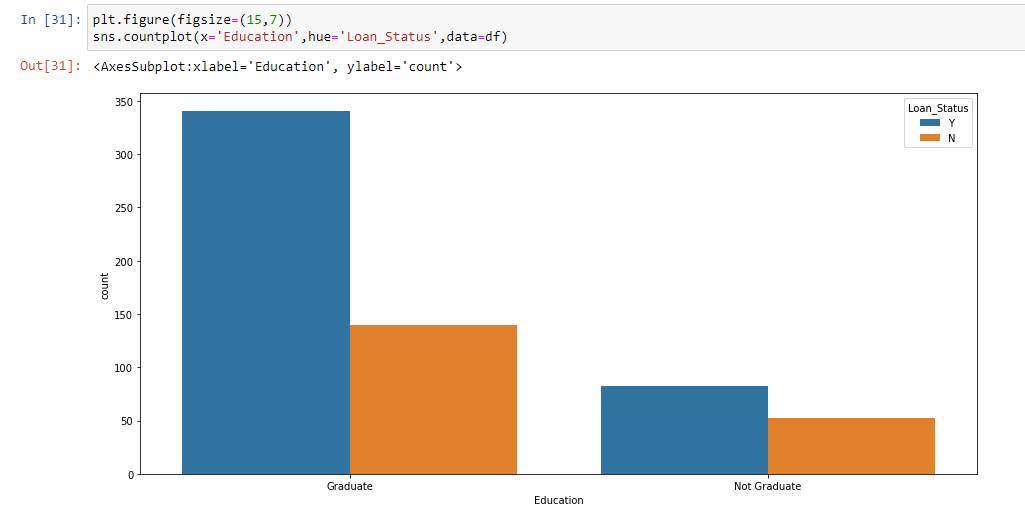
* Here we have the logical figure most of the people those are not getting approval, not having credit hsitory
* the people those having credit history, most of them getting approval



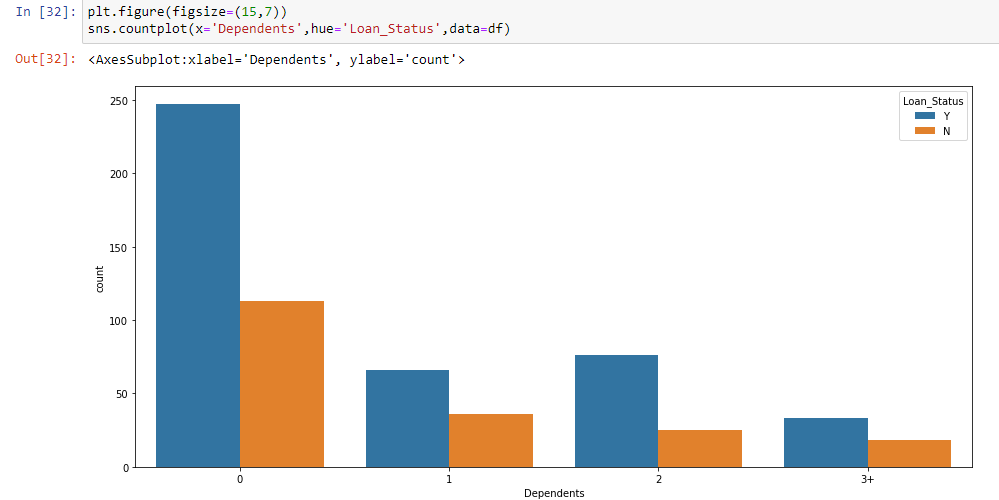
* Most of the people who applied for loan asked for 360 months of term more half are getting approved
* People those are taking loan for 480 months, most of them are getting approval



* most of the data we have from not employed person
* Here we see if the person is not self employed there is high chances of approval
  + There are more than 350 people are approved and more than 150 are not approved those are not self\_employed
* In self\_employed we see there around 50% of diffrence between approved and not approved people

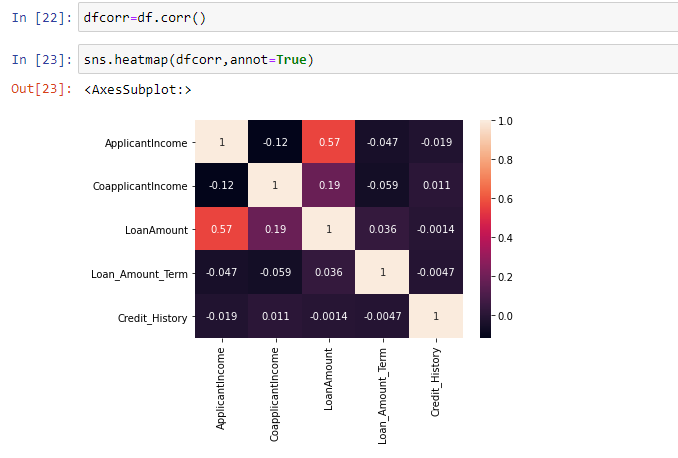


* There are high chances of approval for Graduate people
  + In Graduate category. we have around 350 approved loan and around 150 not approved
* In other side we can notice of the person is not graduate difference is very less in approved and not approved category



* We can notice people those are not having dependents are getting chances of Loan approval
* when dependents numbers are 3+ there is less chances of approval

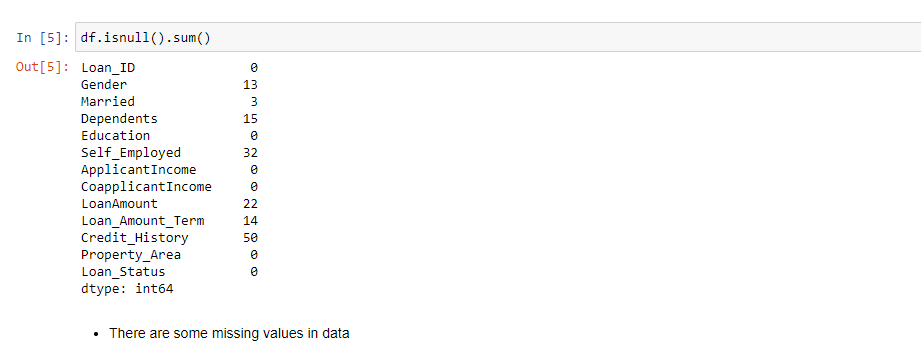
**Checking the correlation between variables:**

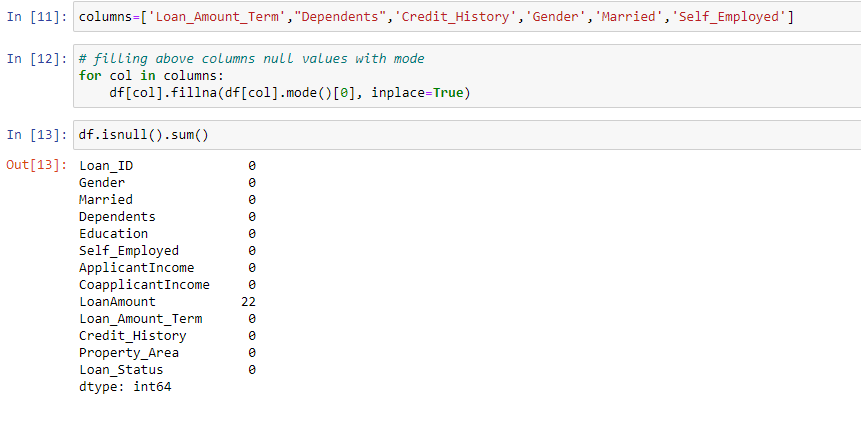


* We can not notice LoanAmount is correlated with ApplicantIncome
* all the other featurs are negatively correlated with each other

**DATA CLEANING AND STRUCTURING:**

Before we go for modelling the data, we have to check whether the data is cleaned or not. And after cleaning part, we have to structure the Data. For cleaning part, First I have to check whether there exists any missing values. For that I am using the code snippet isnull().sum()

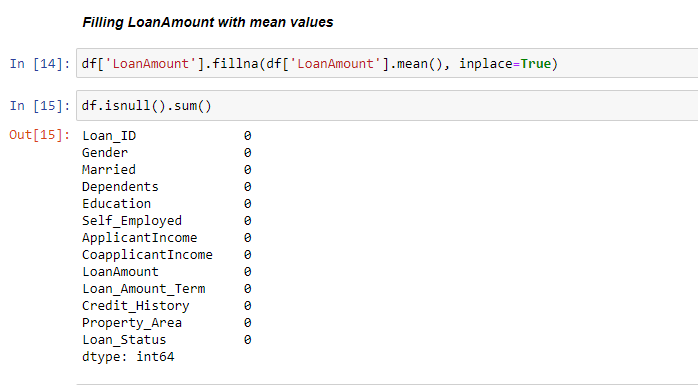


The above code suggests that there are 13 missing values in Gender, 3 in Married, 15 in Dependents, 32 in Self\_Employed, 22 in Loan Amount, 14 in Loan\_Amount\_Term and 50 in Credit History.

Except the Loan Amount and Loan\_Amount\_Term everything else which is missing is of type categorical. Hence we can replace the missing values by mode of that particular column. Before getting in to the code , I would like to say few things about mean , median and mode.

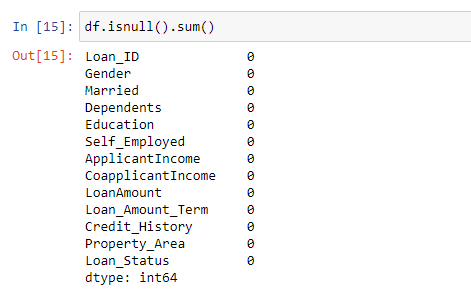
Mean is nothing but the average value where as median is nothing but the central value and mode the most occurring value. Replacing the categorical variable by mode makes some sense. Foe example if we take the above case, 398 are married, 213 are not married and 3 are missing. So As married people are higher in number we are considering the missing values as married. This may be right or wrong. But the probability of them being married is higher. Hence I replaced the missing values by Married.

For categorical values this is fine. But what do we do for continuous variables. Should we replace by mean or by median.



In the above code, missing values of Loan-Amount is replaced by mean.

After replacing let us check if there are further any missing values by the following code df.isnull().sum().

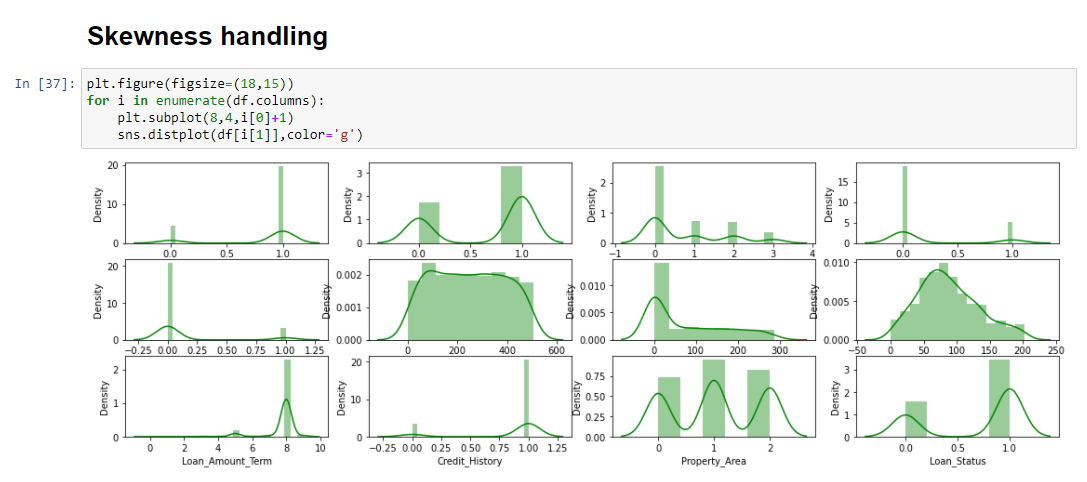


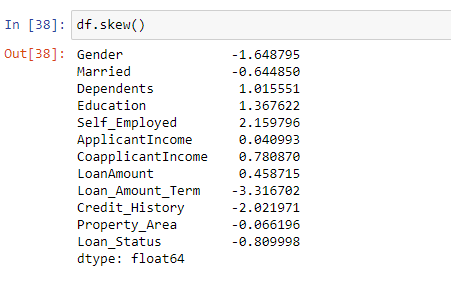
Now we found that there are no missing values anymore.

Lets remove the Loan\_ID as well it’s just unique no for everyloan there will no use of this for model building and prediction the loan status.



***Skewness handling:***





#### skewness more than+/- 0.5 will we treated but not treating the object and Target column

most of the columns are skewed but we have to check most of them are categorical features as well they must be imbalance that’s why showing as skewness

##### will only deal with numercal columns

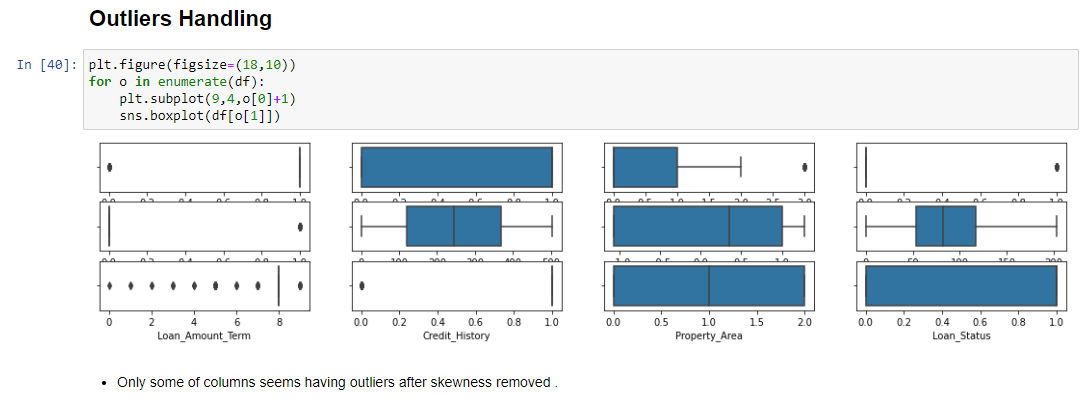
* CoapplicantIncome

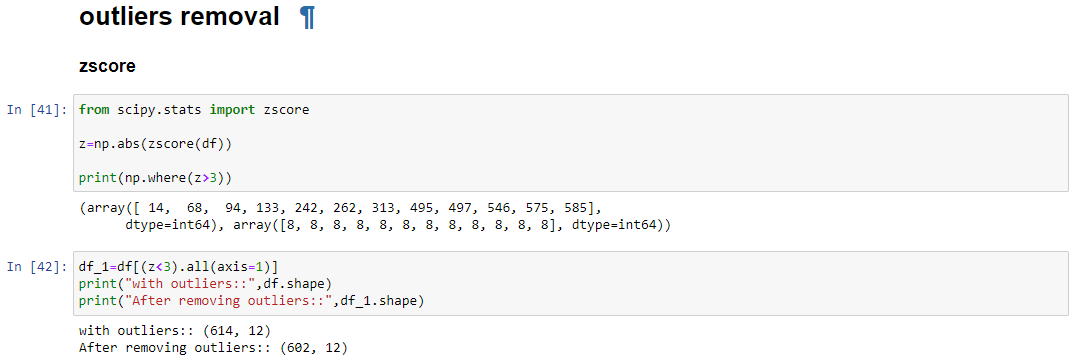
will be treated

choosing the Power Transform for skewness removal



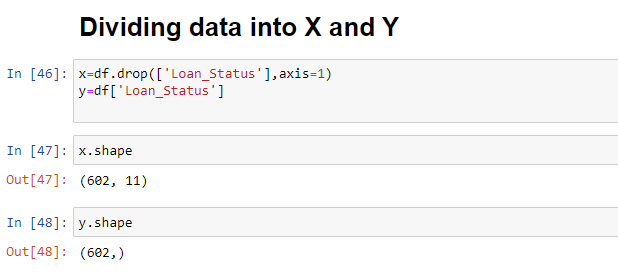
***Outliers Handling:***





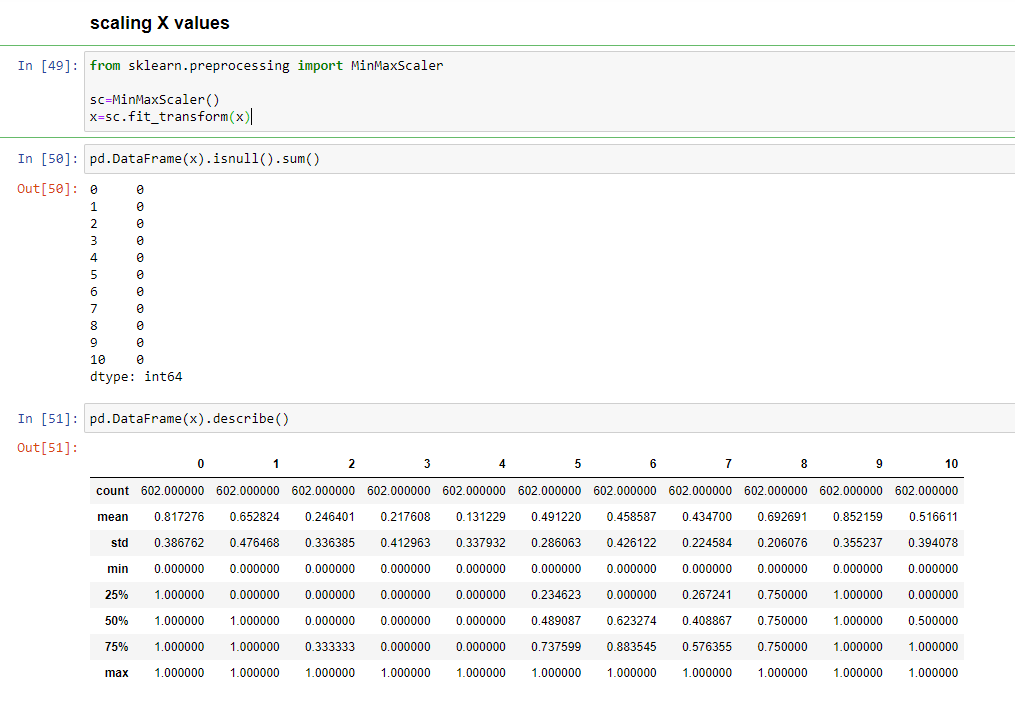
Using ZSCORE for outliers removal we can its removed 12 rows from data as outliers.

As the data cleaning and data structuring are done, we will be going to our next section which is nothing but Model Building.



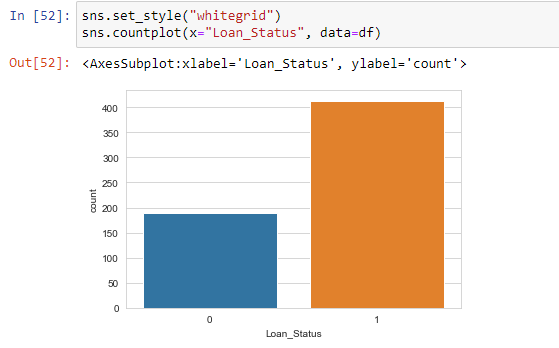
Divided the data into two parts features (x) and target (y) for model

***Data Scaling:***



scaling the data using MinMaxScaler in min 0 to max 1 so model won’t be bias for any number

***Imbalanced learn:***



* We can see here Data is not fully balance but we'll treat the imbalance.
* we have around 200 in yes and around 400 in no.

Using OverSamling\_BorderlineSMOTE

* because there is less data in Loan\_Status for yes



Out data is balanced not we have same number of data for both category now we can go for model building.

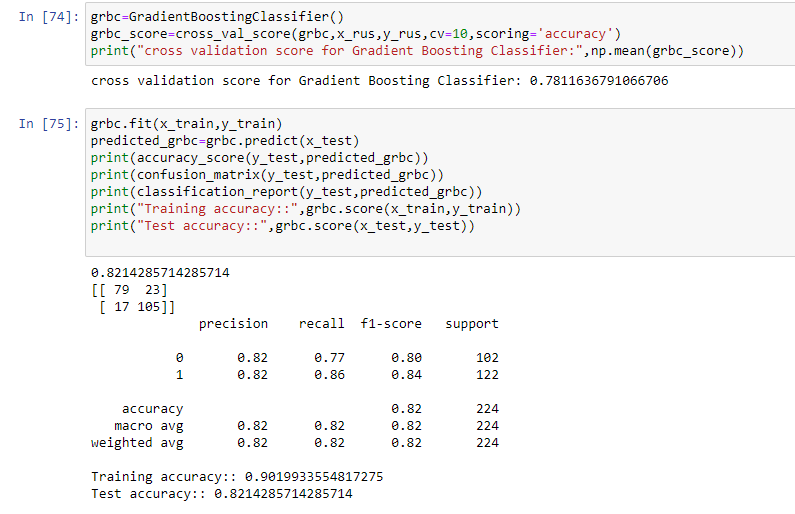
***Building Machine Learning Models***:

***SAMPLING TECHNIQUES AND NEED FOR THAT*** :

There are many sampling techniques like Random Sampling, Stratified Sampling etc. The major purpose is to improve the accuracy which can be obtained by hiding some portion of train data and running the model so that on an average the one that gives higher accuracy can be taken for test data.

## ***Gradient Boosting classifier***

## As here we wan’t to classify between the people who have taken loan or not we have used Gradient Boosting classifier. The purpose of this algorithm is to find a plane that separates two types. Y variable belongs to 1 or 0.

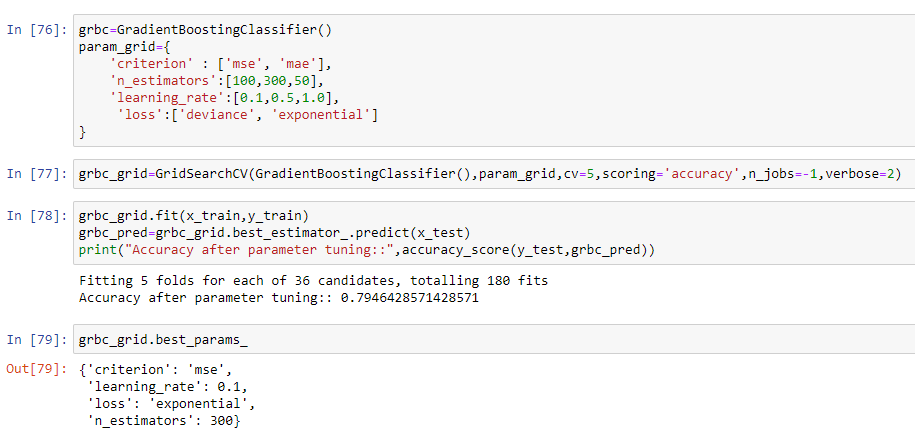


I have tried various techniques like Random Forest, Ada Boost, Decision Tree etc. and came to conclusion that the above code gave maximum accuracy. However there is still a lot of room to enhance accuracy which I have to figure it out still.

# HYPER PARAMETER TUNING:

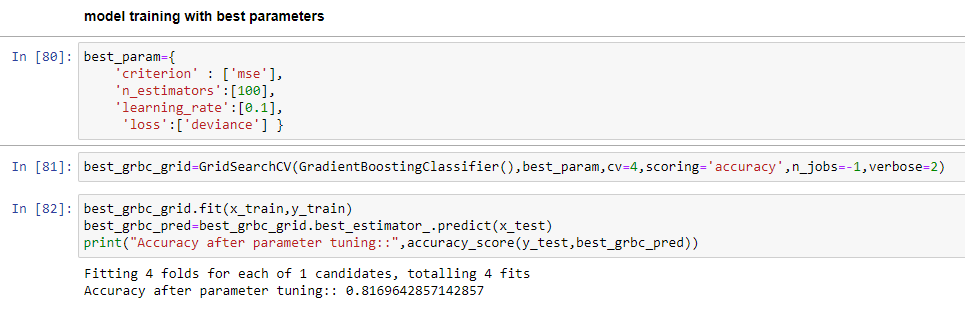
To enhance the accuracy tuning the model .

I have used the **GridSearchCV** grid for tuning the model so we can get the best parameters



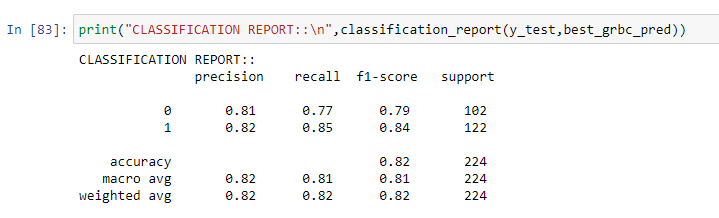
After tuning the model we got the best parameters for model

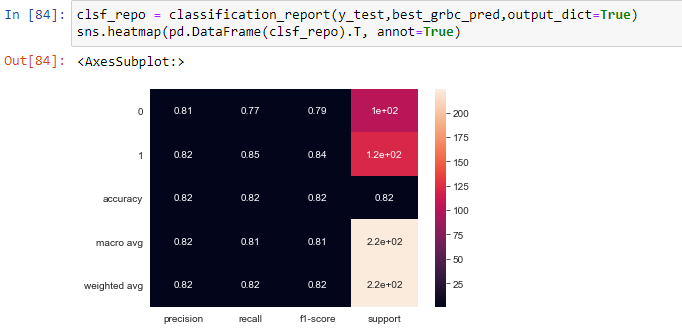
Lets train the model with best parameters



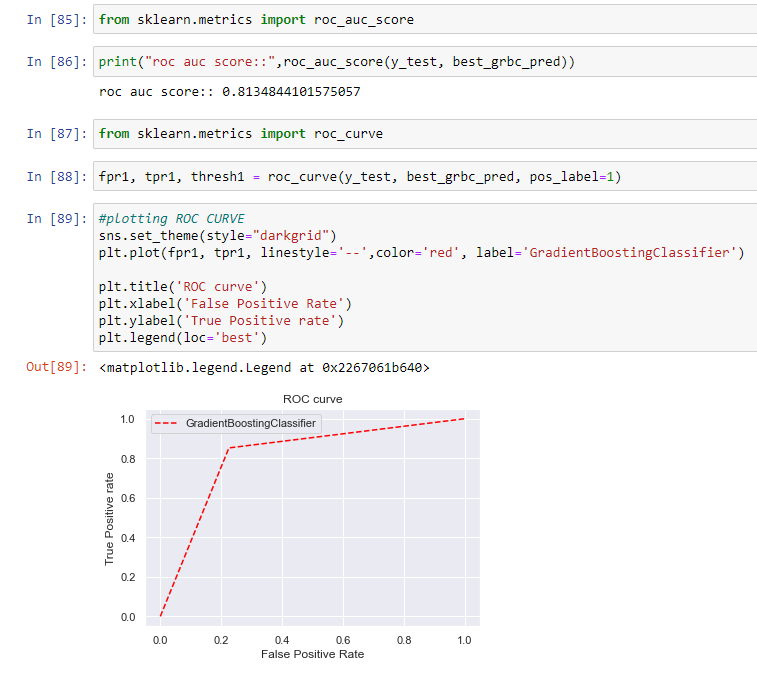
As we can see average accuracy is 81.69% . I have tried with other parameters as well but this the best I got.

**Classification report:**





**roc\_auc\_score:**



roc\_auc\_score = 81.34%

***CONCLUSION*:**

Key Findings and Conclusions of the Study:

* 1. There are high chances of approval for Graduate people comparing not graduate
  2. There is high chances of Loan Approved when you have credit history, people those are not having any credit history mostly getting not approved
  3. People from Semiurban area are having high chances to get their loan approved comparing people from other area.
  4. There are high chances for loan approval when you taking loan for less tenurity