**Problem Statement**

This dataset includes details of applicants who have applied for loan. The dataset includes details like credit history, loan amount, their income, dependents etc.

**Independent Variables:**

- Loan ID

- Gender

- Married

- Dependents

- Education

- Self Employed

- Applicant Income

- Coapplicant Income

- Loan Amount

- Loan Amount Term

- Credit History

- Property Area

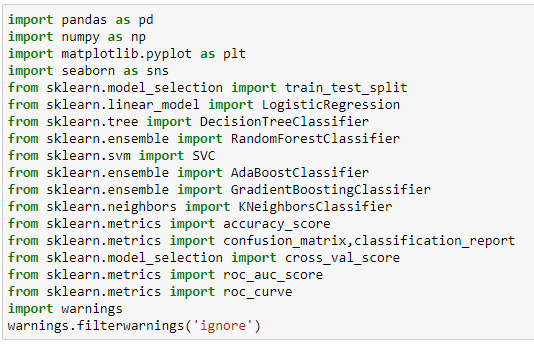
**Dependent Variable (Target Variable):**

- Loan Status

You have to build a model that can predict whether the loan of the applicant will be approved or not on the basis of the details provided in the dataset.

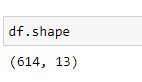
**Importing libraries**

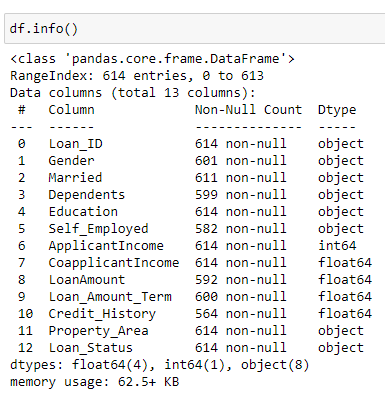
As our first step we need to import the libraries in our jupyter notebook, for Loan application status we are imported different libraries

****

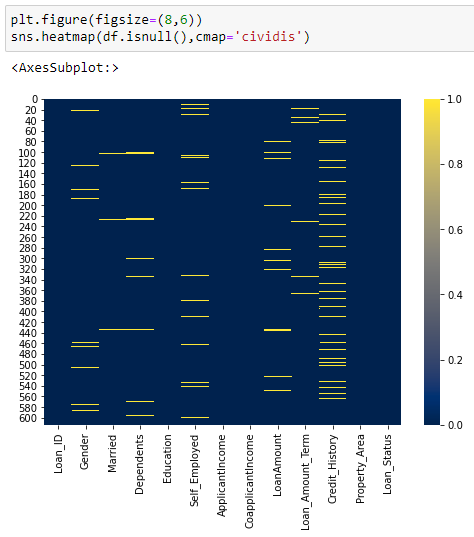
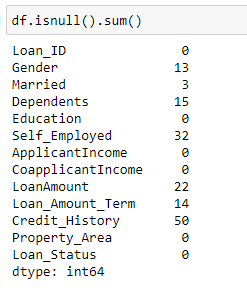
We import pandas, numpy, seaborn, matplotlib, model selection and some machine learning algorithms and some metrics to know accuracy and finally we imported warnings (to ignore warnings)

****After we import, we need to upload our data into our file by using pandas reading file. Here we using csv data file

**** Shape function gives us dimension of dataframe. Here we have 614 entries and 13 columns



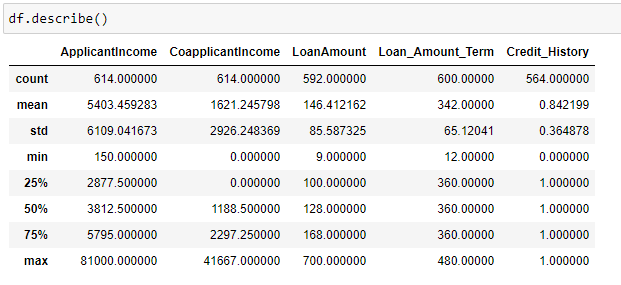
we have 614 entries are as RangeIndex 0 to 613. and total 13 columns in this dataframe, in Gender has 601 non-null values, Married 611 non-null values, Dependents we have 599 non-null values and Self\_Employed we have 582 non-null values, LoanAmount has 592 Non-null values and Credit\_Histroy has 564 nano-null values and remaining columns has 614 non-null values and 4columns are in float64 dtype, one column is in int64 dtype and 8 columns are in object dtypes and we have 62.5+ KB

Checking null values

Gender column has 13 null values, Married has 3 null values, Dependents has 15 null values, Self\_Employed has 32 null values, LoanAmount has 22 null values, Loan\_Amount\_Term has 14 null values and Credit\_History has 50 null values in dataframe.

Above plot is

**Describe** is gives the information of count of variables that used to calculate if there is null values that shows less number of rows than given in dimension, and function also it gives the information of mean, standard deviation, minimum value,25th percentile, 50th percentile(which is median),75th percentile and in last it gives maximum value of each column



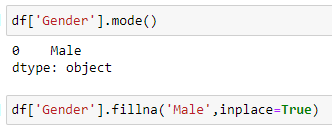
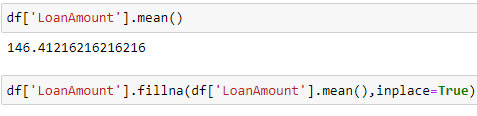
we can observe that all the columns, mean is more than median (50th percentile) and also observe that difference between max value and 75th percentile is very high except Credit\_Histroy column. The both observe clear that data has outliers

**Handling missing values**

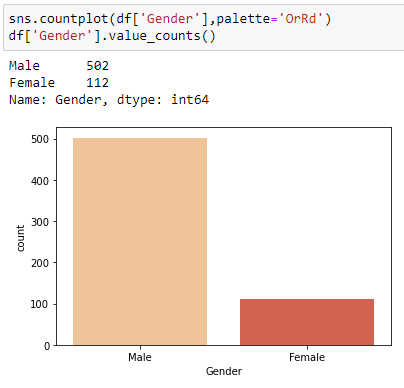
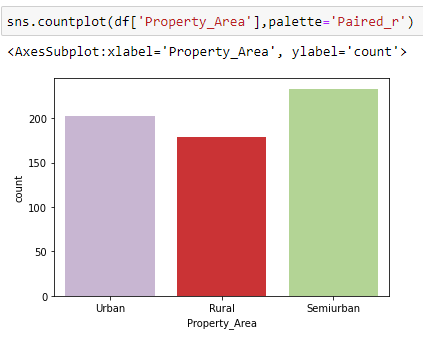
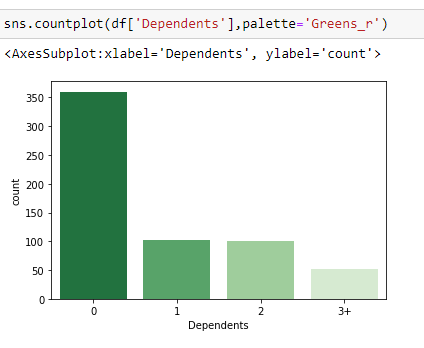
We have different technique to handling null values. We can drop the rows where we have null values but we lose the data if we have more null values. For this project I use filling the null values with mean and mode values

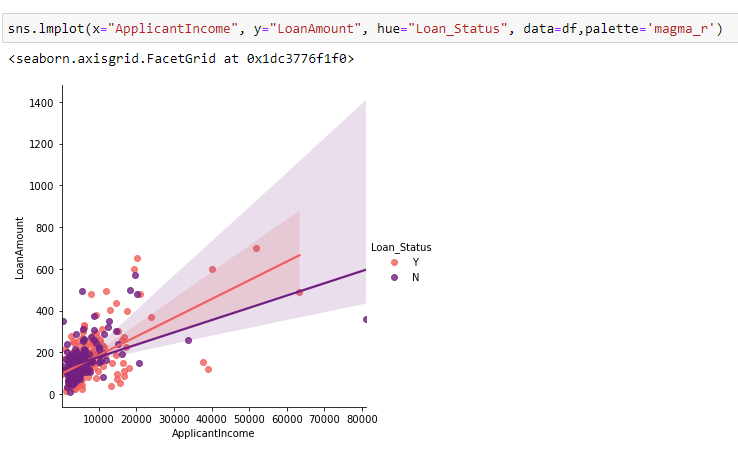
If the column is object dtype we need to fill the null values with the mode value of the column

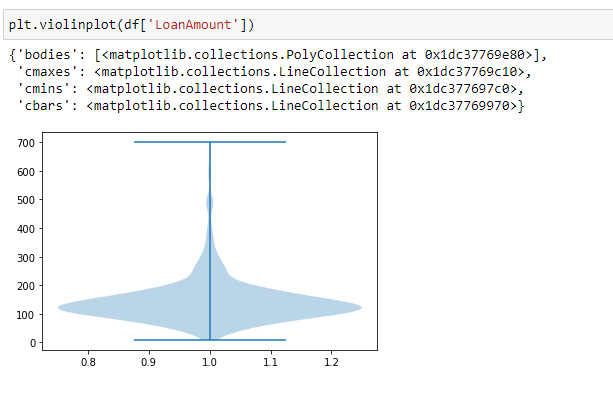
If the column is numeric type data we need to fill the null values with mean value of the column



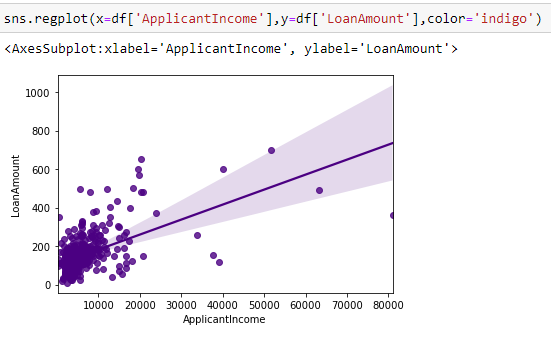
**Data visualization**

****

Most of the column has object type data so we use to find how the data was distributed in the dataframe. And we are looking for unique values and value counts in these to know about the data more



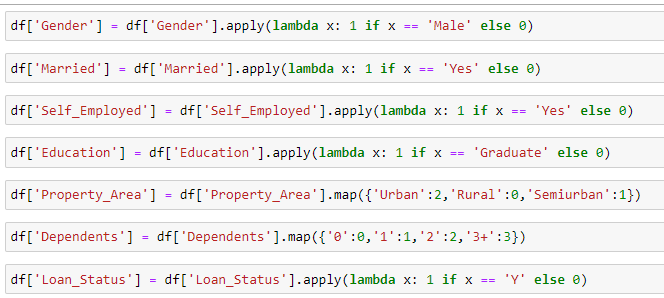
Violin plots are similar to box plots, except that they also show the probability density of the data at different values, usually smoothed by a kernel density estimator.



regplot is used to plot data and a linear regression model fit.

both columns are has good correlation and it has best fit for linear regression

Next step is converting the categorical data to numerical data. We can convert our data in different model here I use map function to convert



Above image, we can see that how I converted my categorical data as numeric

**Checking correlation**

Correlation means association - more precisely it is a measure of the extent to which two variables are related. There are three possible results of a correlational study: a positive correlation, a negative correlation, and no correlation.

A positive correlation is a relationship between two variables in which both variables either increase or decrease at the same time. An example would be height and weight. Taller people tend to be heavier.

A negative correlation is a relationship between two variables in which an increase in one variable is associated with a decrease in the other. An example would be height above sea level and temperature. As you climb the mountain (increase in height) it gets colder (decrease in temperature).

A zero correlation exists when there is no relationship between two variables. For example, there is no relationship between the amount of tea drunk and level of intelligence.

### Strength of correlation

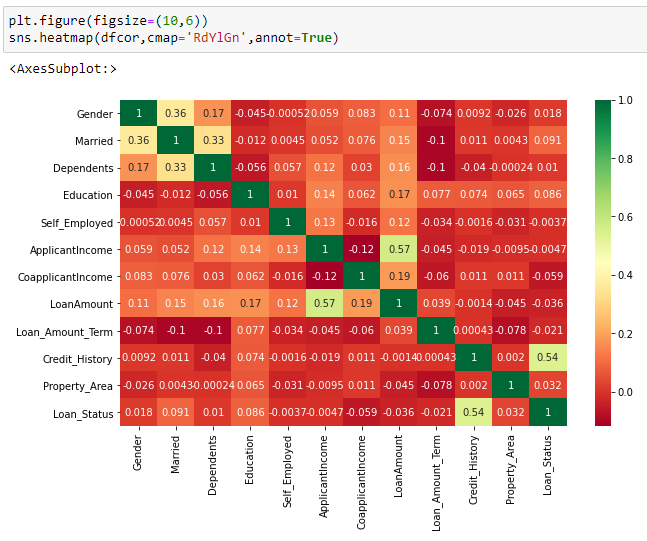
prefect +1,-1

strong +(0.9 to 0.7) & (-0.9 to -0.7)

moderate +(0.6 to 0.4) & (-0.6 to -0.4)

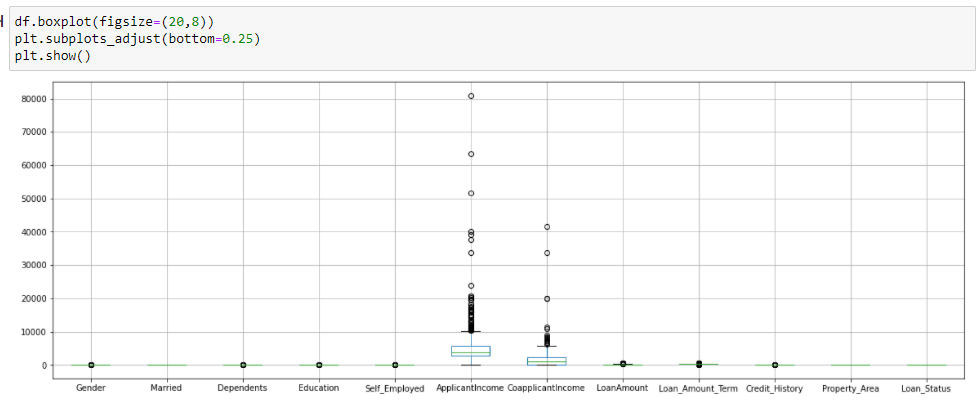
week +(0.3 to 0.1) & (-0.3 to -0.1)

Zero 0



We have to see correlation between target variable with features if we have zero we no need to keep that column if we drop that column also it won’t effect on predication

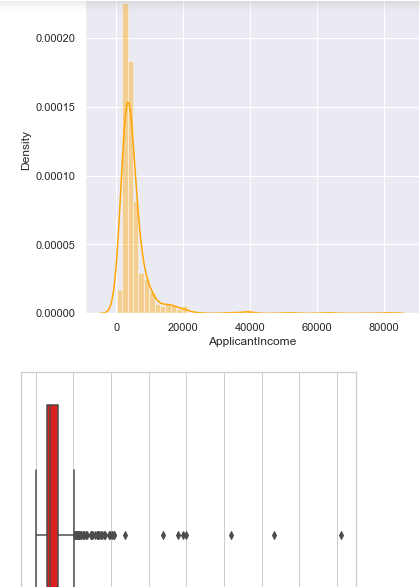
Next step is to checking outliers

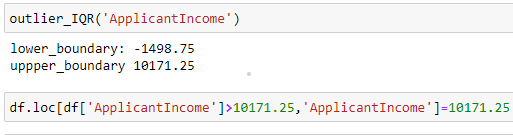


We have outliers in the most of the columns

We can remove outliers with zscore method and we can do by using IQR also

Here I use the IQR method by defining the function

Here is my defined functions here 

here I use only one column to explain ApplicationIncome

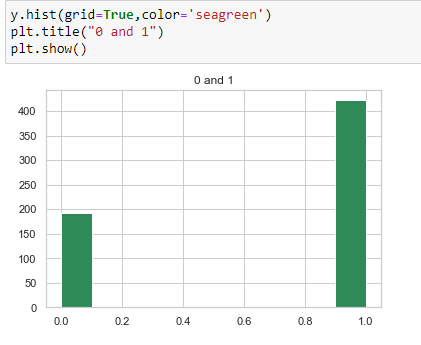
data was skewed so use outlie\_IQR function, if it is normally distributed then I go with outlier\_norm function

After completion of data cleaning, we need to go next

Dividing target variable and features

here our target variable is Loan\_status

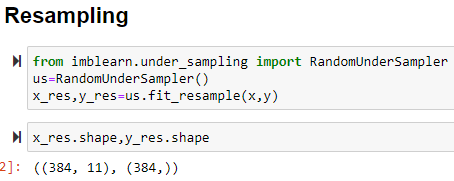
This project is classification model so we need to check imbalance in the data



We have to deal this imbalance by resampling, we have 2 different ways to resample

We can do with over sampling and under sampling

Here I use the under sampling



By using random under sampling from imblearn.under\_sampling library

After this we need to move our next step to applying algorithms, before that we have to find the best random state 

Here we got random\_state 78 is the best for this dataframe

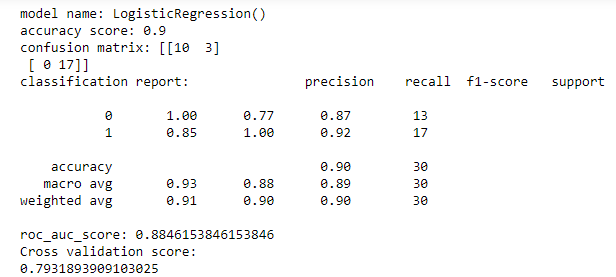
Tain test split the data and applying the algorithms and get a good out all the algorithms



I prepared list for algorithms to loop It automatically

Here we are using accuracy score confusion matric and classification reports and roc\_auc\_score for all algorithms and also find the cross -validation score.

Example for result of logistic regression



Here I got accuracy 0.9 and

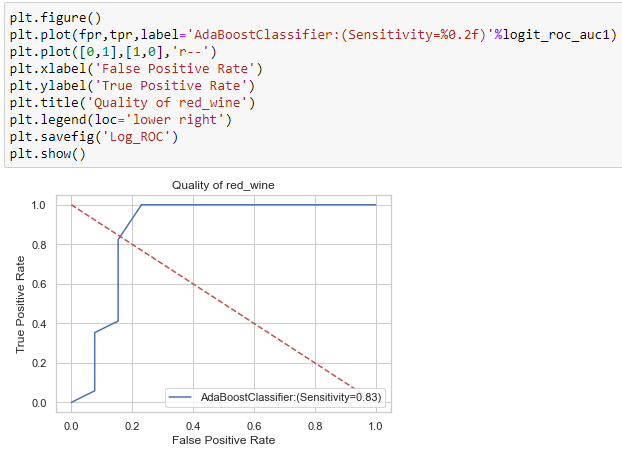
A **confusion matrix** is a technique for summarizing the performance of a classification algorithm. Classification accuracy alone can be misleading if you have an unequal number of observations in each class or if you have more than two classes in your data

The **classification report** visualizer displays the precision, recall, F1, and support scores for the model. There are four ways to check if the predictions are right or wrong: TN / True Negative: the case was negative and predicted negative. TP / True Positive: the case was positive and predicted positive

**Conclusion**

After analysing data and data cleaning and modelling the data at the end we need to conclude that which is the best suitable mode for the project. Here we need to see the different between accuracy score and cross-validation which give less difference we need to consider that as our best solution

For this project I got AdaBoostClassifier

****Here I use the roc curve for AdaBoostClassifier

**An ROC curve (receiver operating characteristic curve) is a graph showing the performance of a classification model at all classification thresholds. This curve plots two parameters: True Positive Rate. False Positive Rate.**