Global Power Plant Database

Submitted by:

DEEPTHI MALLU



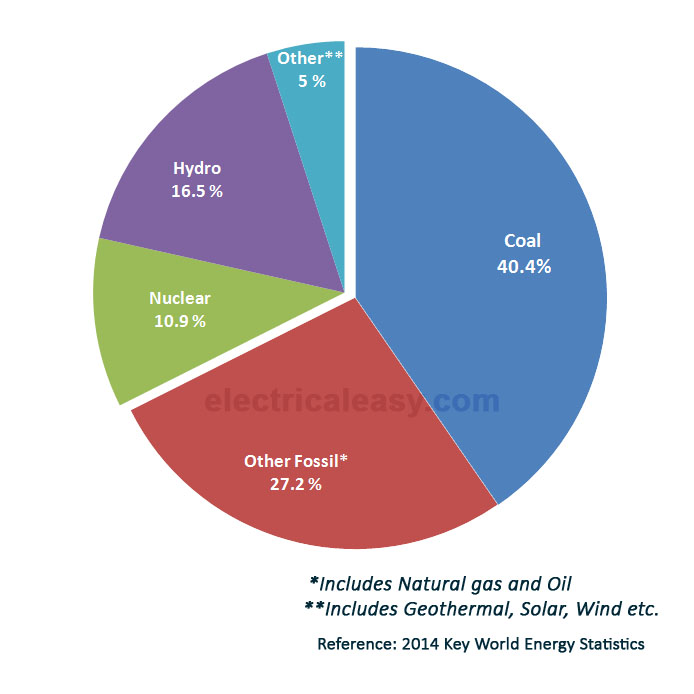
**INTRODUCTION**

**Power Generation, India**

Power generation, one of the most important aspects of utility systems in the world. India is the world's [third largest producer](https://en.wikipedia.org/wiki/List_of_countries_by_electricity_production) and [third largest consumer](https://en.wikipedia.org/wiki/List_of_countries_by_electricity_consumption) of electricity. Our article focuses on, the power generation capacity and the type of fuel used for power generation. Power is Utilized by every Individual, without power one cannot survive. Our dataset is all about power generation plants in India. We use different sources of energy for generating the power, which are Renewable Energy resources and Non-Renewable Energy resources. Based on availability of resources one uses the source of energy to generate the power.

In India, every region uses different type of fuels for power generation because the resources are not common in all the places. This article throws light upon the seven main sources of energy that are found in India. The sources are: 1. Coal 2. Mineral Oil 3. Natural Gas 4. Thermal Power 5. Hydro Power 6. Nuclear Power 7. Solar power.

Furthermore, we use machine learning models to predict the amount of capacity of power to be generated and the primary type of fuel used for power generation.



**Problem Definition**

The main Aim of this project is to build machine learning models to predict power capacity in Mega Watts and the primary fuel used for power generation. In India most of the power is generated from Thermal power plants and the Hydel power plants.

[Thermal power](https://en.wikipedia.org/wiki/Thermal_power) is the "largest" source of power in India. There are different types of [thermal power plants](https://en.wikipedia.org/wiki/Thermal_power_station) based on the [fuel](https://en.wikipedia.org/wiki/Fuel) that are used to generate the steam such as [coal](https://en.wikipedia.org/wiki/Coal), [gas](https://en.wikipedia.org/wiki/Fuel_gas), and [diesel](https://en.wikipedia.org/wiki/Diesel_fuel), [natural gas](https://en.wikipedia.org/wiki/Natural_gas). Across India, 71% of consumed electricity is generated from thermal power plants. The [national electric grid](https://en.wikipedia.org/wiki/National_Grid_(India)) in [India](https://en.wikipedia.org/wiki/India) has an installed capacity of 383.37 [GW](https://en.wikipedia.org/wiki/Watt#Gigawatt) as of 31 May 2021.

The units of power generation databases are collected around the world, which are used to compare our problem with those databases and draw insights and make further analysis, and can be able to build machine learning models and check the performance and patterns drawn from the models to predict our label.

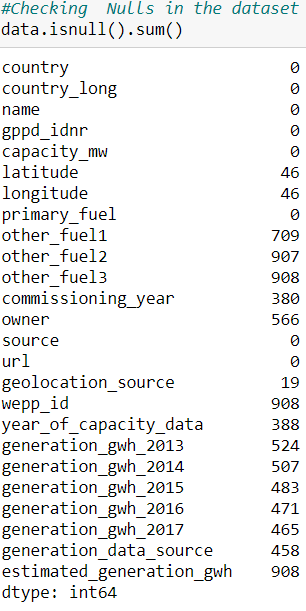
**Data Analysis**

In this project, we have a dataset which contains geolocation of the power plants and the entries information on plant capacity, generation, ownership, and fuel type.

The given dataset contains 908 rows and 28columns. The columns are country, country-long, name, gppd-idnr, capacity-mw, latitude, longitude, primary-fuel, other-fuel1, other-fuel2, commissioning-year, etc.

The obvious con of this data set is the small sample size. However, there is no information in some of the columns. As a data scientist we need to pre-process the data and get insights from the data and build machine learning models based on the type of problem.

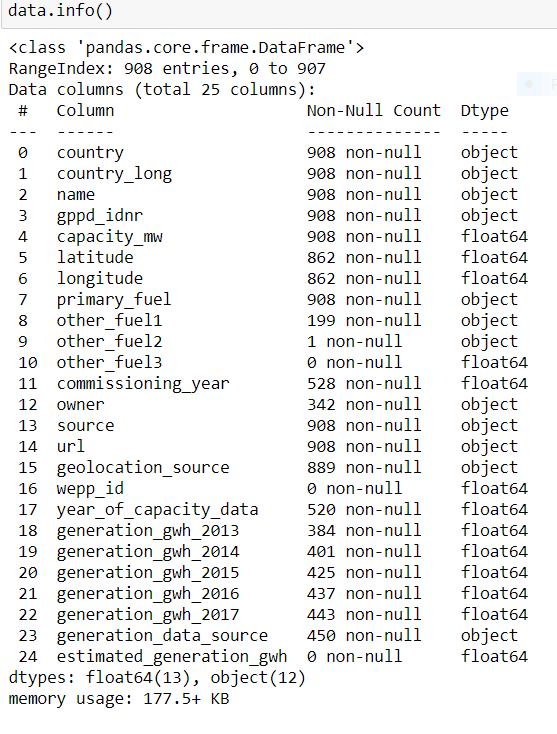
Our dataset consists of many null values, which to filled by some imputing techniques based on the type of the column. The Number of Null’s given below:



In some of the columns the data is completely filled with NaN’s so for that type of columns, I didn’t use any imputing techniques I just dropped those columns which are of no use.



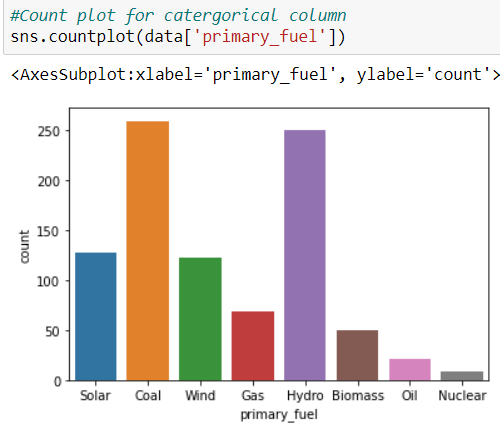
The column which is of object type that column to be filled by using Mode, if the column is of either integer type or float type that column to be filled either by mean or median based on the distribution of the data in the column. If the column is normally distributed use mean or the data is skewed then use median method to fill NaN’s.



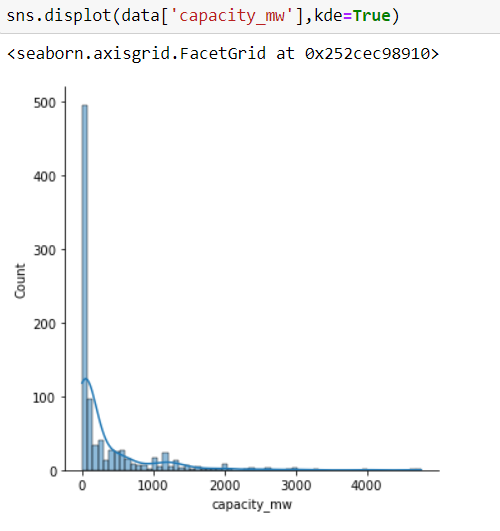
**Exploratory data analysis**

**Dependent variable**: In the dataset we have two labels to be predicted, the capacity in Mega-watts and the primary fuel.

Since the type of fuel is categorical type, it comes under multi classification problem. Exploratory data analysis was conducted starting with the dependent variable primary fuel. In India, the usage of coal and hydro is more for power generation. There are completely 253 power plants which uses coal as primary fuel and 250 power plants uses hydro as the source to generate power.

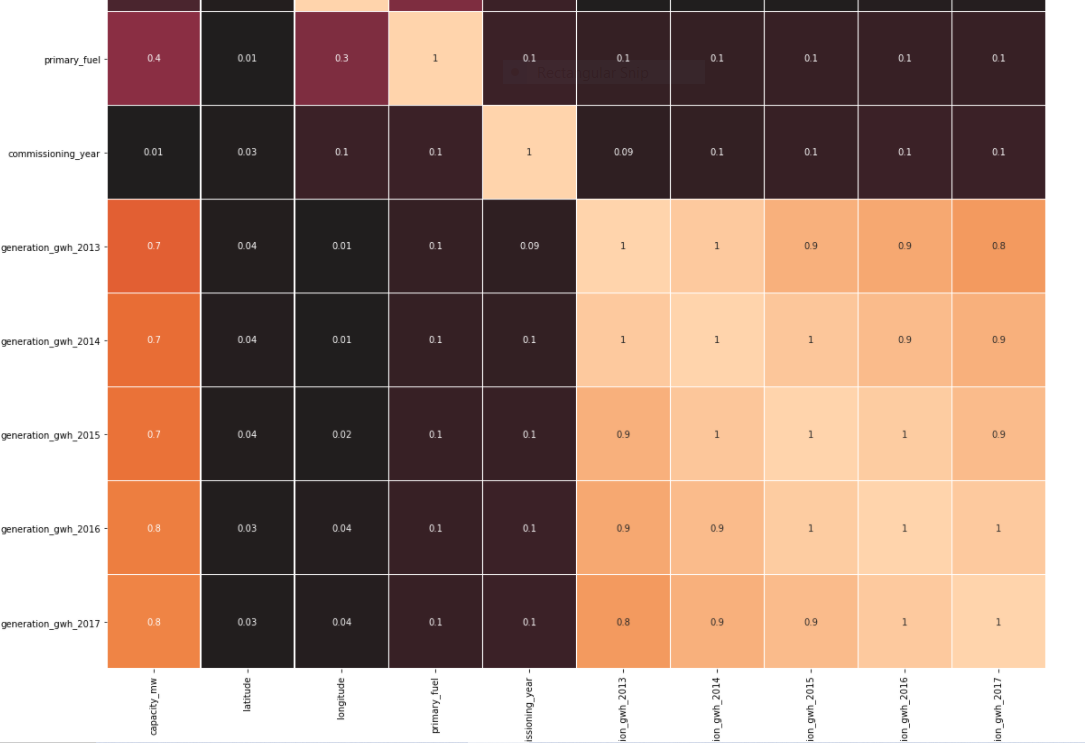


Second dependent variable, the capacity of the power plant which is of numeric type, so I have used distribution plot to check how the data is distributed in the column. The maximum capacity of the power generated is 4760 Mega Watts.



**Correlations among variables:**

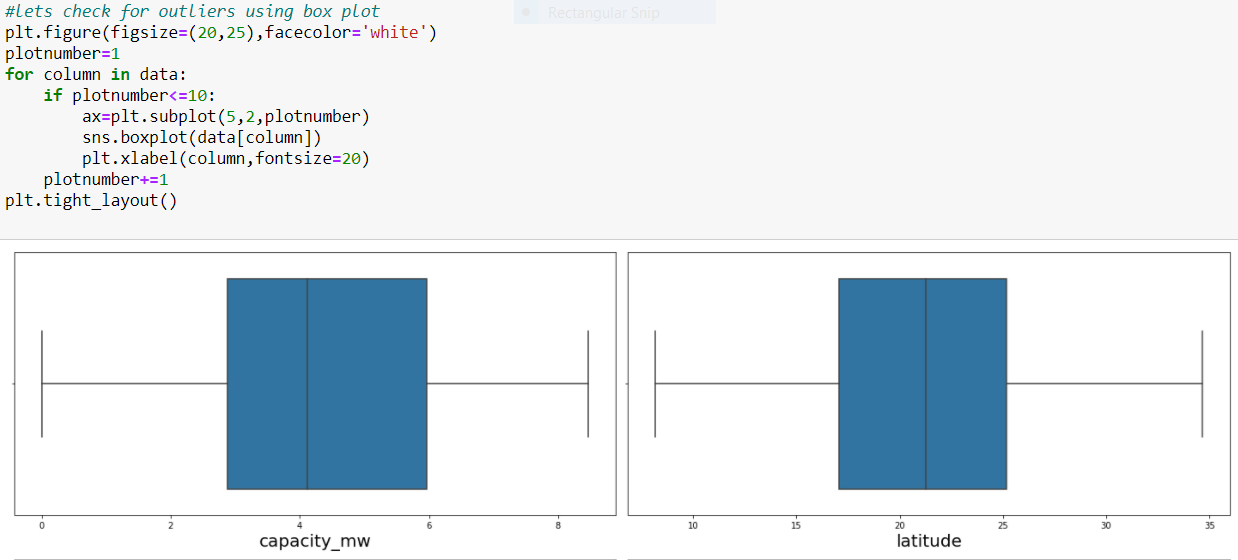
To check the correlation, I have plotted Heat map to visualise the correlation between columns. Then I found labels are highly correlated with the generation of power in Gigawatts with different years and very less correlated with commissioning year, there is also 100% multi collinearity (feature to feature correlation) problem exists with some of the features.



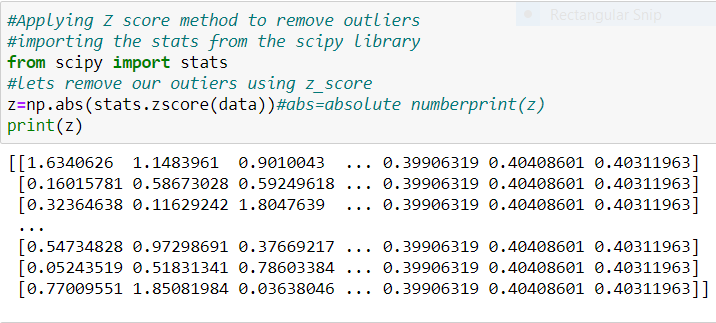
**Visualizing variables:**

In order to find outliers and skewness present in the data, I have plotted box plots and distribution plots to check how much data is affected and tried removing them using Z-Score and Log transformation methods respectively.

**Box Plots**



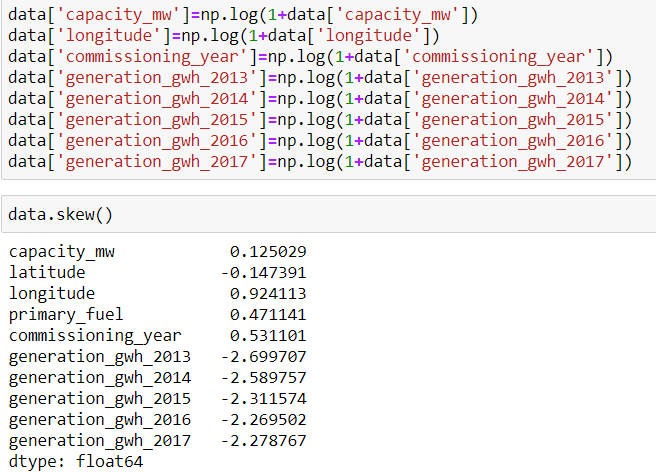
**Z-Score**



**Distribution Plots**



**Log-Transformation**



## Even I plotted some Scatter plots to visualize how the features and labels are correlated.

## 

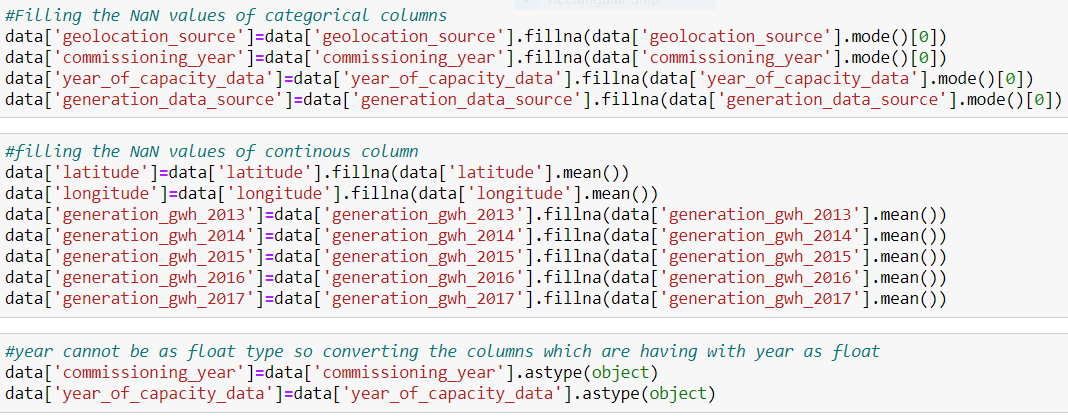
## Pre-processing Pipeline

## Data pre-processing is a data mining technique which is considered as a predominant step in machine learning to yield highly accurate and insightful results.

## It involves transforming raw data into an understandable format. Real-world data is often noisy, incomplete, inconsistent, and lack in certain behaviours or trends, and is likely to contain many errors.

**Treating Missing Values**

## In this dataset we have many missing values, in some of the columns the data is completely filled with NaN’s which are of no use so I have dropped those columns. The Columns which are of object type I filled the NaN’s with mode and the columns of numeric type I filled using mean.



**Dropping garbage columns**

## I have dropped some of the columns which do not contribute much for our label’s prediction.

## 

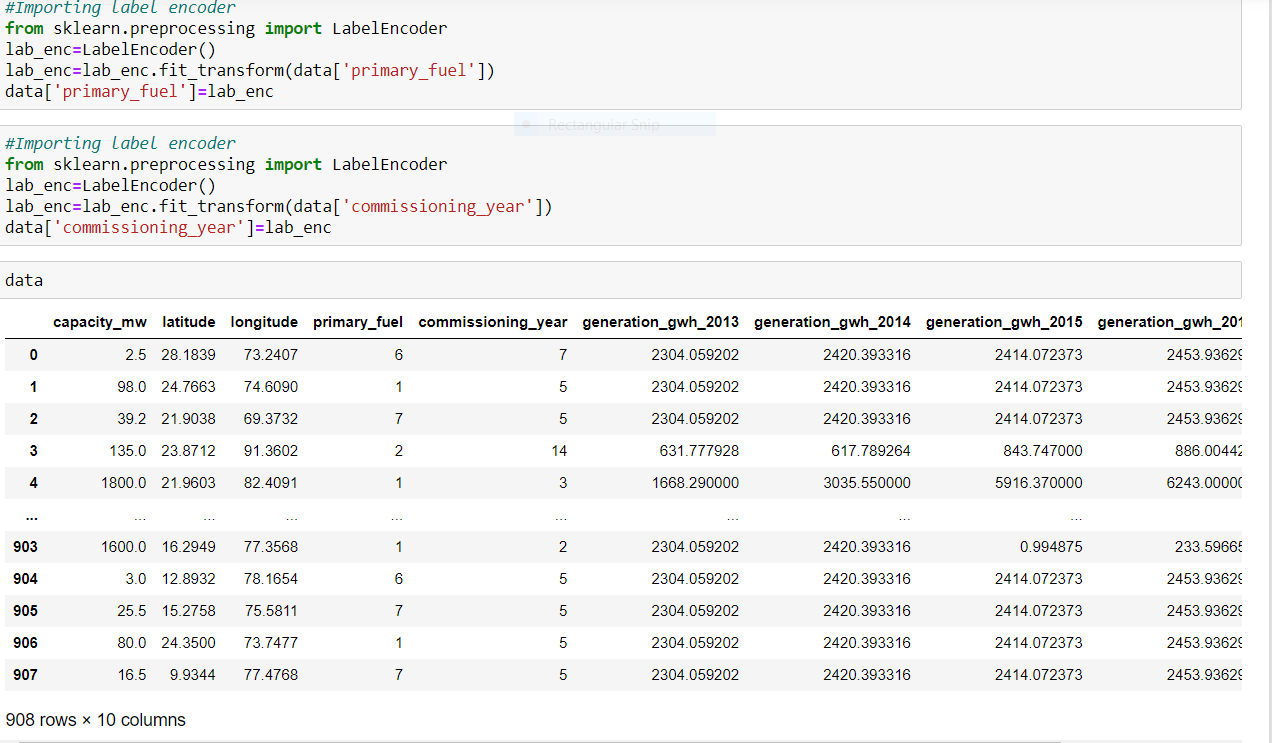
**Encoding**

In machine learning, we usually deal with the datasets having of categorical data and numerical data. Machine Learning Algorithms accepts only Numerical data as input. As our models cannot understand the labels, we need to convert them to 0’s and 1’s using some encoding techniques.

In our dataset there is a column with categorical variables. The primary fuel which is our label and the other columns are commissioning\_year and year\_of\_capacity\_data columns which are of big in number, which affects our model accuracy so in order overcome the problem, I have subtracting one column with the other column and named new column with commissioning\_year.

**Label Encoder**

Label Encoding is a popular encoding technique for handling categorical variables. It refers to converting the labels into numeric form so as to convert it into the machine-readable form. Machine learning algorithms can then decide in a better way on how those labels must be operated. It is an important pre-processing step for the structured dataset in supervised learning.



**Balancing our imbalanced data**

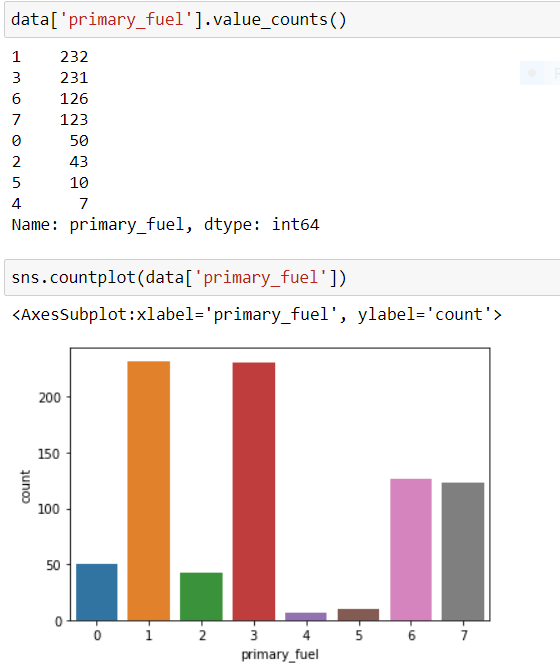
**SMOTE**

## SMOTE is an over-sampling technique focused on generating synthetic tabular data. The general idea of SMOTE is the generation of synthetic data between each sample of the minority class and its “k” nearest neighbors.

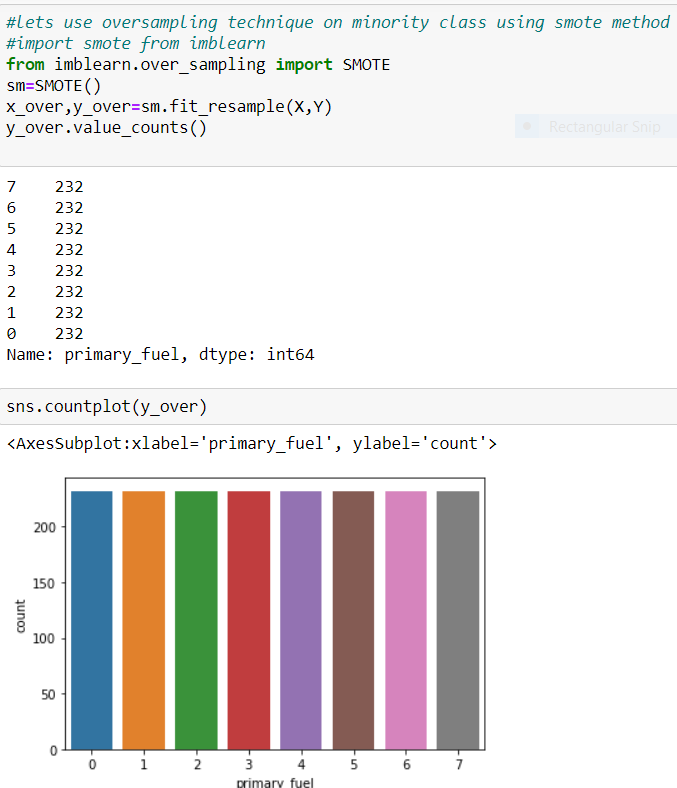
In our dataset we have two predictions, Smote is applied on Categorical labels as it comes under classification problem. Since the values of the label are not equally distributed, we use SMOTE for up-sampling to solve the imbalance problem and aims to balance class distribution by randomly increasing minority class examples by replicating them.

So, I have applied SMOTE on our label primary fuel, because all the classes in the label are imbalanced, so in order to make the classes balanced, I used one of the popular over sampling technique SMOTE.

**Before SMOTE**

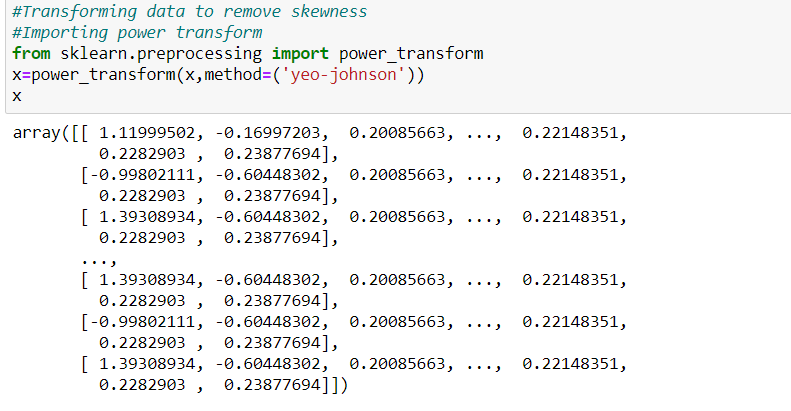


**After SMOTE**



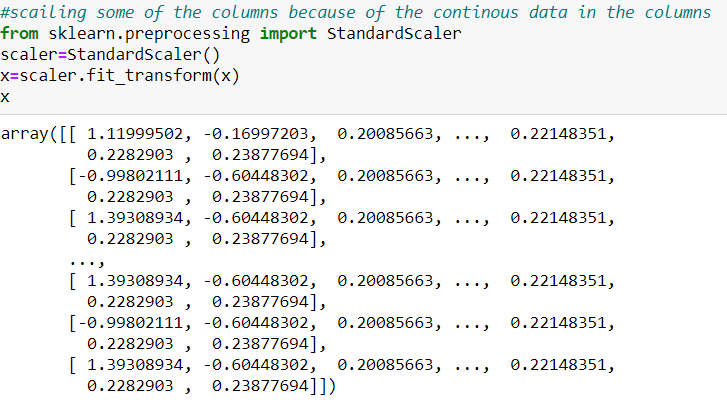
**Data Transformation**

First, I have split the dataset into x and y variables, in ‘x’ we have our features and in ‘y’ we have our label. After splitting I have applied power transform on the features to remove the skewness.



**Standardization:**

Standard Scaler follows Standard Normal Distribution, which makes mean = 0 and scales the data to unit variance. Since our data in the columns are of different scales with which our models cannot give good patterns and performance on unequal scales, in order to bring all the values to standard form we use standard scaler.



**Building machine learning models**

Machine learning is a method of data analysis that automates analytical model building. It is a branch of artificial intelligence based on the idea that systems can learn from data, identify patterns and make decisions with minimal human intervention.

All the machine learning models are imported from sklearn. Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modelling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python.

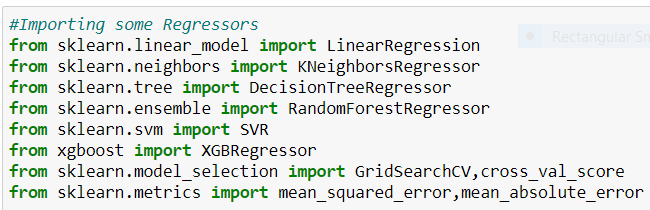
In this global power generation data base, we have to predict two labels one is of continuous type and other of categorical type. The continuous label prediction comes under Regression problem and the categorical label comes under Classification label. We will see one after the other in first case we will predict the continuous label which is capacity in Mega Watts.

**Label 1**

**Capacity in MW**

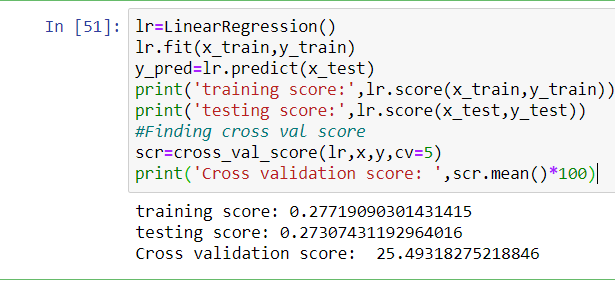
First, we need to split the x(features) and y(label) values into train data & test data using train test split which is also imported from sklearn. In the train test split we are going to divide the training data 70% and testing data as 30% using train data we are going to fit the model and using test data we are going to predict and evaluate the model performance. Since the label is a continuous variable, it comes under Regression problem. So, we need to use regression models for predicting the label.

I have used 6 different algorithms for prediction, the model which gives the best performance, that model will be considered as the best model for prediction.



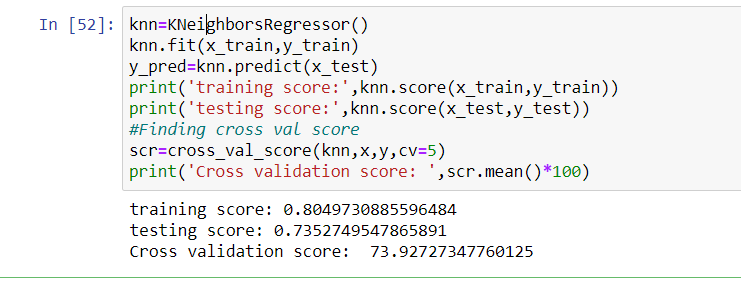
**Linear Regression**

 Linear Regression is a machine learning algorithm based on supervised learning. It performs a regression task. Regression models predicts our target variables based on independent variables. It is mostly used for finding out the relationship between variables and forecasting. It is used to predict, or visualize, a relationship between two different features/variables.



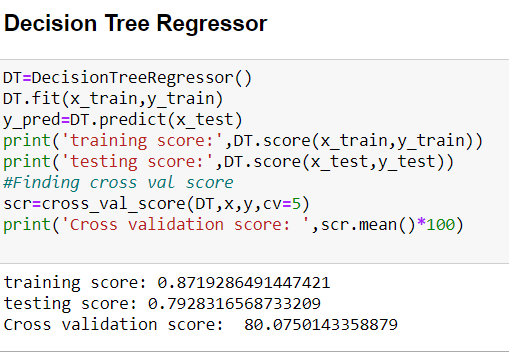
**KNeighbors Regression**

K-nearest neighbors is a supervised machine learning algorithm which can be used to solve both classification and regression problems. It belongs to the class of non-parametric models. The models don’t learn parameters from training data set to come up with a discriminative function in order to classify the test or unseen data set. Rather model memorizes the training data set. This is why K-NN classifier is also called as lazy learner.



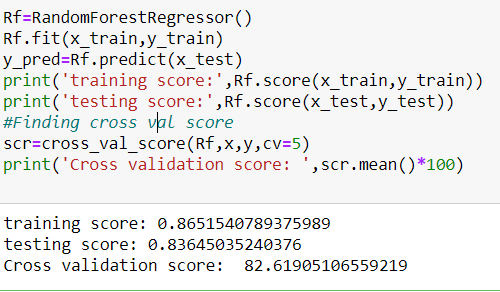
**Decision Tree Regression**

The decision tree is a supervised machine learning algorithm which breaks down the data set into smaller subsets. A decision leaf splits into two or more branches that represent the value of the attribute under examination. The topmost node in the decision tree is the best predictor called the root node. ID3 is the algorithm that builds up the decision tree. It employs a top to down approach and splits are made based on standard deviation.



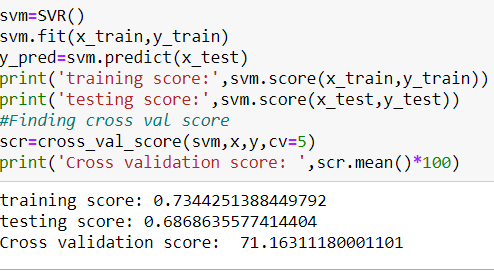
**Random Forest Regression**

Random Forest is a Supervised learning algorithm that is based on the ensemble learning method and many Decision Trees. Random Forest is a Bagging technique, so all calculations are run in parallel and there is no interaction between the Decision Trees when building them. RF can be used to solve both Classification and Regression tasks.



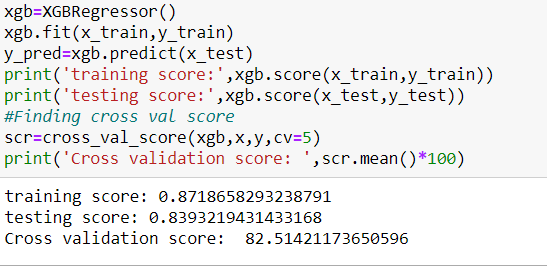
**Support Vector Regression**

In machine learning, support-vector machines (SVMs, also support-vector networks) are supervised learning models with associated learning algorithms that analyze data for classification and regression analysis.



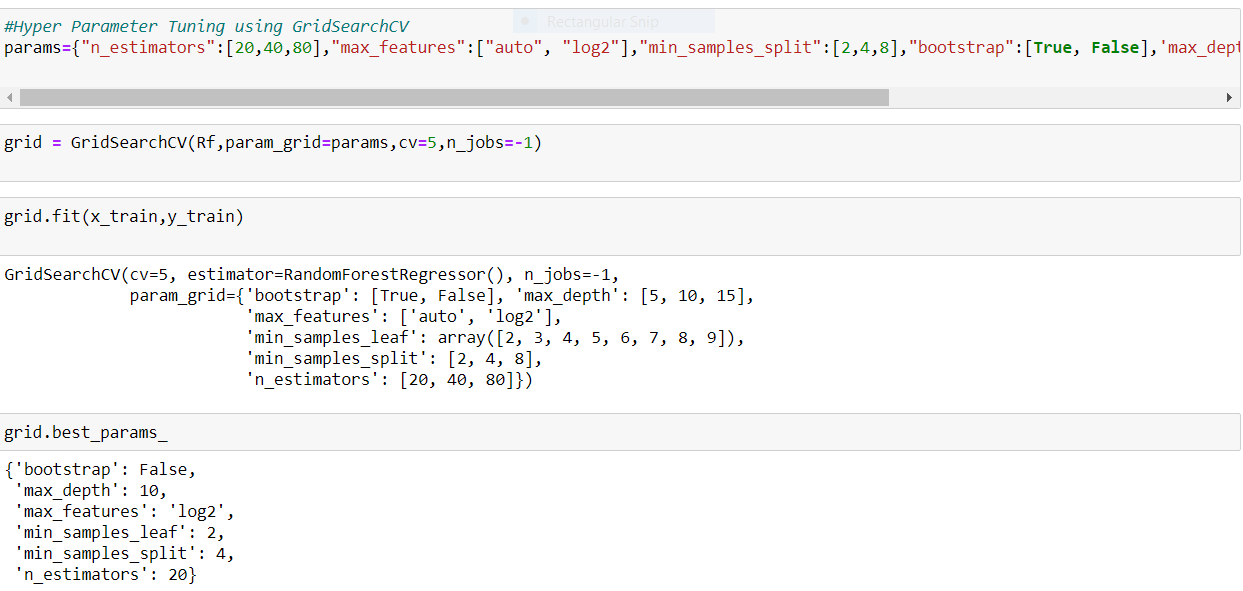
**Extreme Gradient Boost Regression**

XGBoost is one of the most popular machine learning Algorithm these days. Regardless of the type of prediction task at hand; regression or classification. XGBoost is well known to provide better solutions than other machine learning algorithms.



**Hyper Parameter Tuning**

“Hyper parameter tuning is choosing a set of optimal hyperparameters for learning an algorithm”. It is also defined as a parameter whose value is set before the learning process begins.



**Evaluation Metrics**

Evaluation metrics in data modelling are tools that describe how well a particular model is solving your problem. These metrics help in the selection of the best parameters, can give you insights of the data you are working with, and ultimately, they support the reliability and integrity of your model.

**Mean Absolute Error (MAE):**

It is the average of the absolute differences between the actual value and the model’s predicted value.

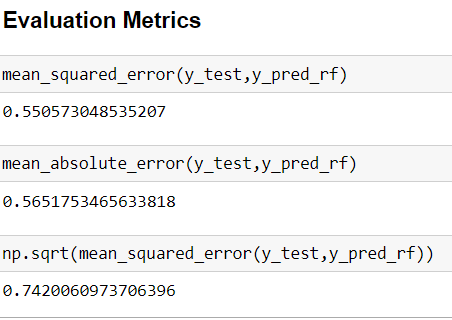
**Mean Squared Error (MSE):**

It is the average of the squared differences between the actual and the predicted values.

**Root Mean Squared Error (RMSE):**

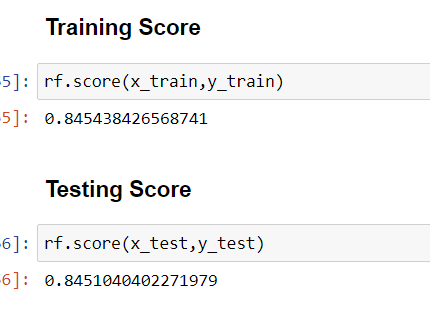
It is the average root-squared difference between the real value and the predicted value. By taking a square root of MSE, we get the Root Mean Square Error.

We want the value of RMSE to be as low as possible, as lower the RMSE value is, the better the model is with its predictions. A Higher RMSE indicates that there are large deviations between the predicted and actual value.

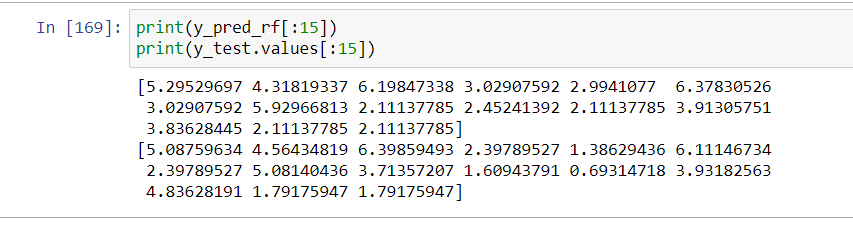


**Conclusions from models**

We got our best model as Random Forest Regressor with the score of 84% and both the training and testing scores are balanced and the RMSE value is also less. So, we can go further build our model as Random Forest.



Comparing the predicted and the Actual values.



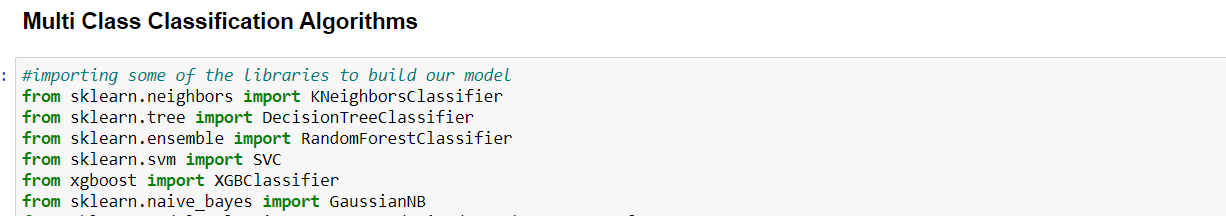
**Label 2**

**Primary Fuel**

The same procedure will be employed for predicting the type the fuel but the change is building the models because the above prediction is of regression problem but predicting primary fuel involves the classification problem because the values in the label are of categorical type and that termed as classes so this is a problem under classification. So, we employee classification algorithms to build our models.

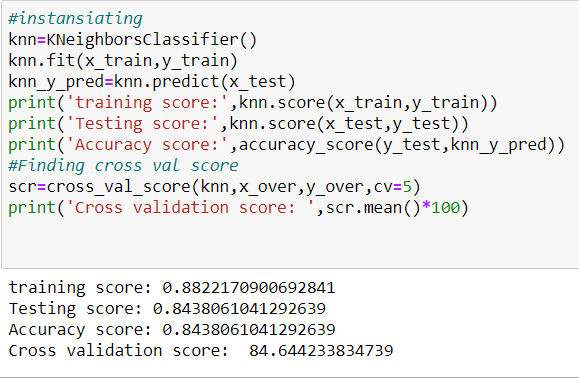
In the classification algorithms we use evaluation metrics as accuracy score, precision, Recall, F1 score all comes under classification report and confusion matrix.

I have used 5 classification algorithms for model building.



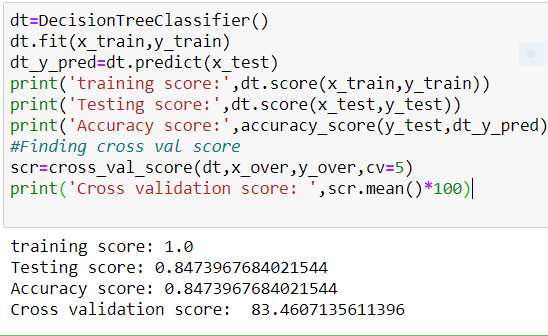
**KNeighbors Classifier**

 K-Nearest Neighbors is one of the most basic yet essential classification algorithms in Machine Learning. It belongs to the supervised learning domain and finds intense application in pattern recognition, data mining and intrusion detection.



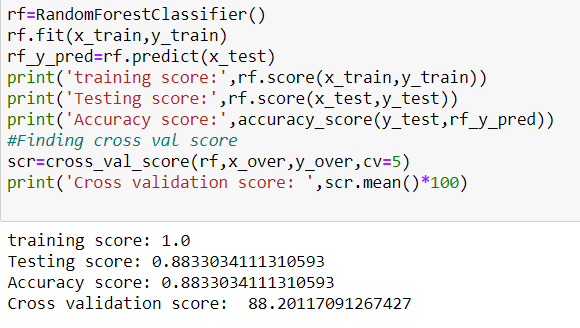
**Decision Tree Classifier**

A decision tree is a structure that includes a root node, branches, and leaf nodes. Each internal node denotes a test on an attribute, each branch denotes the outcome of a test, and each leaf node holds a class label. The topmost node in the tree is the root node. We make some assumptions while implementing the Decision-Tree algorithm.



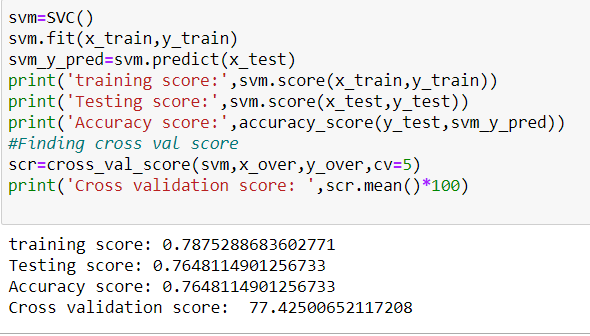
**Random Forest Classifier**

Random forest is a supervised Machine learning algorithm used for both classification and regression, and other tasks using decision trees. It creates a set of decision trees from a randomly selected subset of the training set.



**Support Vector Classifier**

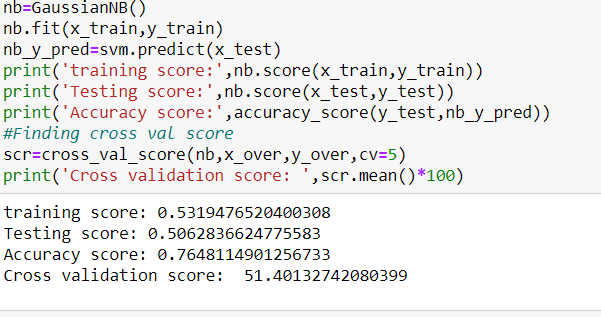
In machine learning, support-vector machines (SVMs, also support-vector networks) are supervised learning models with associated learning algorithms that analyze the data for classification and regression analysis.



**Gaussian Naive Bayes Classifier**

Naive Bayes is a probabilistic machine learning algorithm that can be used in several classification tasks.

Typical applications of Naïve Bayes are classification of documents, filtering spam, prediction and so on. It is not a single algorithm but a family of algorithms where all of them share a common principle, i.e. every pair of features being classified is independent of each other.



**Conclusion from models**

We got our best model as Random Forest Classifier with the accuracy score of 88% & where the training score and testing scores are balanced.

**Hyper Parameter Tuning**

Hyper parameter optimisation in machine learning intends to find the hyper parameters of a given machine learning algorithm that deliver the best performance as measured on a validation set. Hyper parameters, in contrast to model parameters, are set by the machine learning engineer before training. The number of trees in a random forest is a hyper parameter while the weights in a neural network are model parameters learned during training. I like to think of hyper parameters as the model settings to be tuned so that the model can optimally solve the machine learning problem. So, will use Randomised Search CV for the hyper parameter tuning.

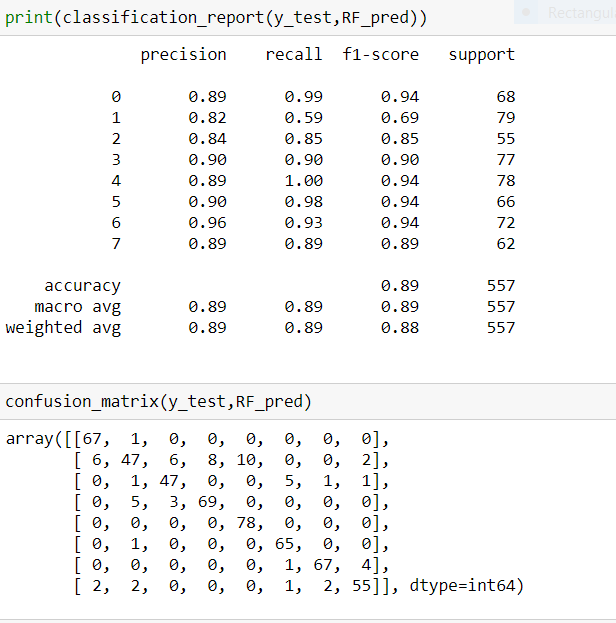
**Randomised Search CV**

Randomized Search CV is very useful when we have many parameters to try and the training time is very long. For this example, I use a random-forest classifier, so I suppose you already know how this kind of algorithm works. The first step is to write the parameters that we want to consider and from these parameters select the best ones.



**Evaluation Metrics for Final model Random Forest Classifier**

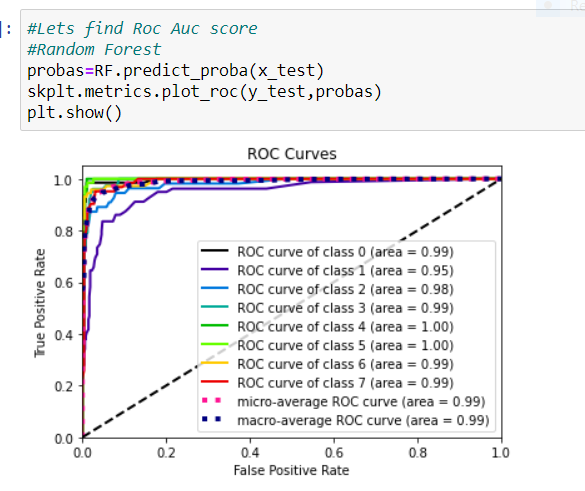
The evaluation metrics are important for a model to decide a model is good or bad which will be decided on the values of metrics such as accuracy score, cross validation score, precision, recall, F1 score, sensitivity, ROC AUC score, confusion matrix. The values for evaluation are shown below.



**ROC curve**

It is a performance measurement for the classification problems at various threshold settings. ROC is a probability curve and AUC represents the degree or measure of separability. It tells how much the model is capable of distinguishing between classes. Higher the AUC, the better the model is at predicting 0s as 0s and 1s as 1s. By analogy, the Higher the AUC, the better the model.

 This ROC curve is plotted in between True positive Rate and False Positive Rate, which are on y-axis and x-axis respectively.



**Remarks**

The Renewable resources are abundantly available in nature. In India we use more of the renewable resources for power generation which are naturally available. So, there is no extra cost for purchasing the fuels, so based on the past data available we can easily predict the power generation capacity of a plant, based on the available resources.

We have many powerplants which are used for power generation and we use different source of fuel for power generation, but in our dataset very few powerplants information was included which is very less in number, which is not sufficient for our prediction, because our statistical methods mostly perform well on longer datasets. So, if in future if we collect more than the data available then we can achieve better results.