Global Power Plant: Classification and Regression ML project

## Introduction

In this blog post, I will walk you through my project on predicting the Primary Fuel using classification and the Capacity (in Mega Watts) using regression machine learning techniques. I have also done EDA on the dataset, and we will come to certain conclusions.

This project aims to provide accurate fuel and capacity predictions using the features in the dataset. We have done Data Cleaning, Feature Selection, Model Selection and Hyper parameter tuning in the classification model.

## Problem Statement

We will make the following two predictions:

1. Primary Fuel - energy source used in primary electricity generation or export.
2. Capacity\_mw - electrical generating capacity in megawatts

The data is collected from the following source: [link](https://github.com/FlipRoboTechnologies/ML_-Datasets/tree/main/Global%20Power%20Plant%20Database)

## Methodology

### EDA (Exploratory Data Analysis)

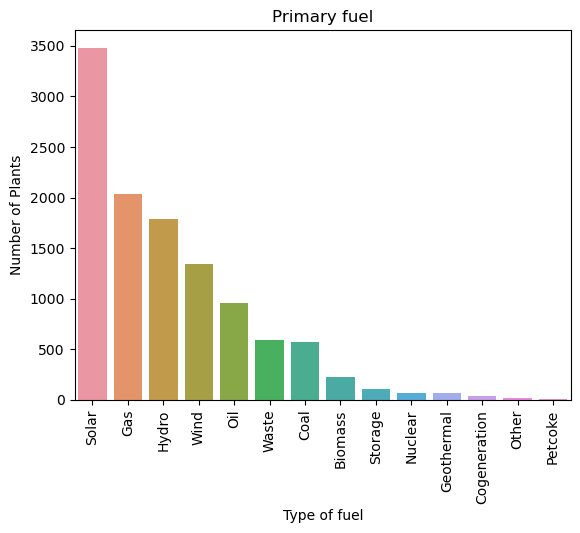
1. Number of power plants existing in the respective countries:

A graph of plants growing

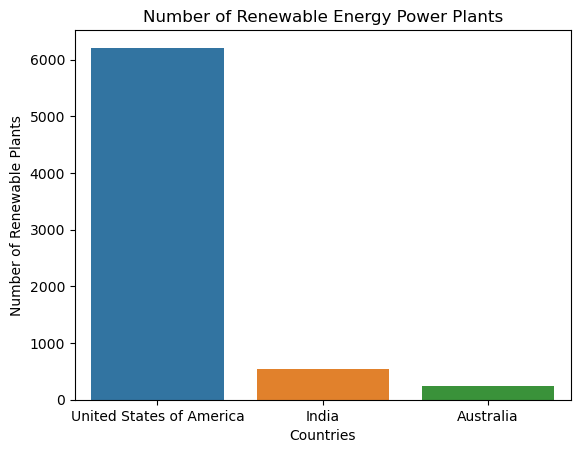
Description automatically generated

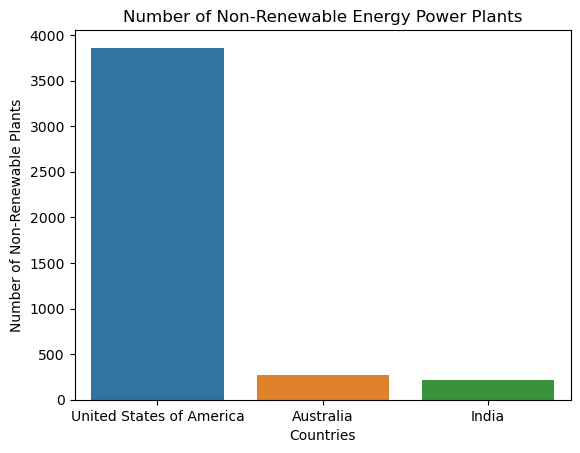
According to this, we can see that USA has the highest number of power plants. It is not surprising that this is the case as it is the third largest countries and deemed as a superpower.

1. Number of plants of all the different categories of plants:



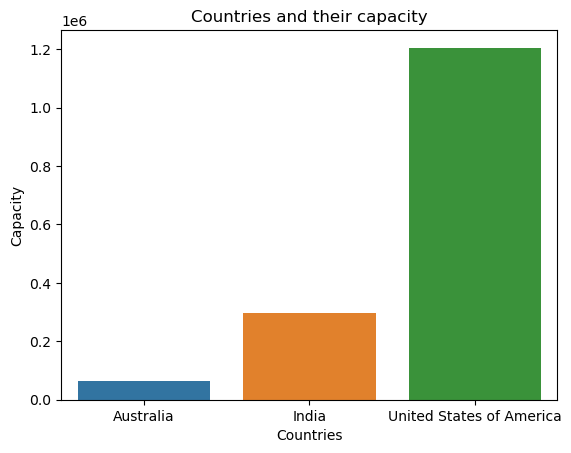
Not surprisingly, solar plants are leading in the power generation as compared to the other conventional power plants. Thankfully the fuels which are harmful to the society are significantly less. Let is check the same:





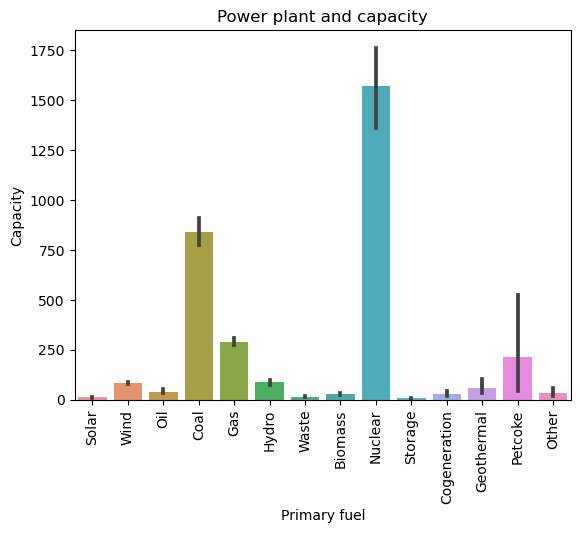
India has more renewable energy plants than non-renewable energy plants. This is a positive observation, and it proves that we are moving towards sustainability.

1. Let us compare the countries and their capacities:



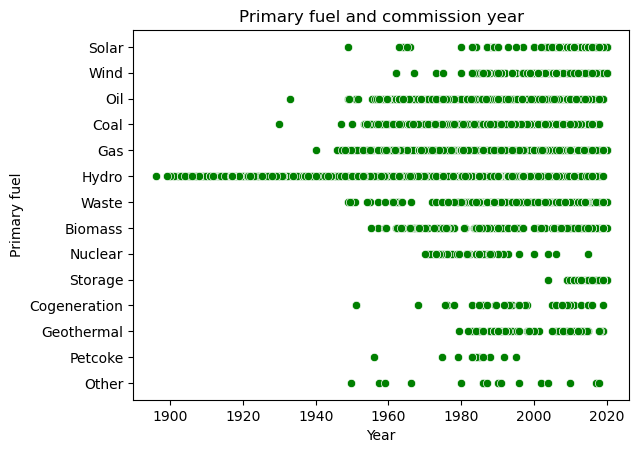
Compared to the number of power plants India has, the capacity generated is huge.

1. Let us check the type of power plant and the capacity it generates. This will enable us to determine which plant is more efficient and hence profitable.



We see that nuclear plant has the most generation capacity, followed by coal and gas.

1. Let us see at what year the plants have been commissioned, and therefore, the density of the manufacturing of the plants:



We can see that Hydro power plant has consistently been commissioned since the 1900’s to the current decade. It can be observed that the more un-conventional plants are recently being commissioned.

1. Let us check the generation of the respective countries in the respective years:

A graph of different colored bars

Description automatically generated

### Data Preprocessing

1. Importing the Libraries:

A white background with black and red text

Description automatically generated

1. Getting the datasets:

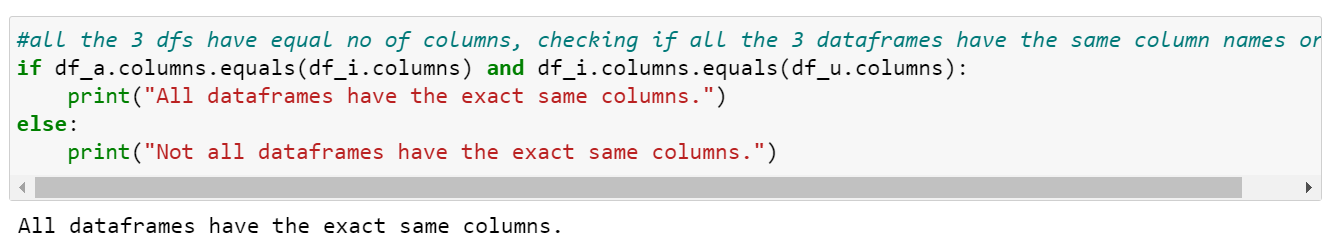


1. Checking the shape:

A white rectangular object with a white background

Description automatically generated

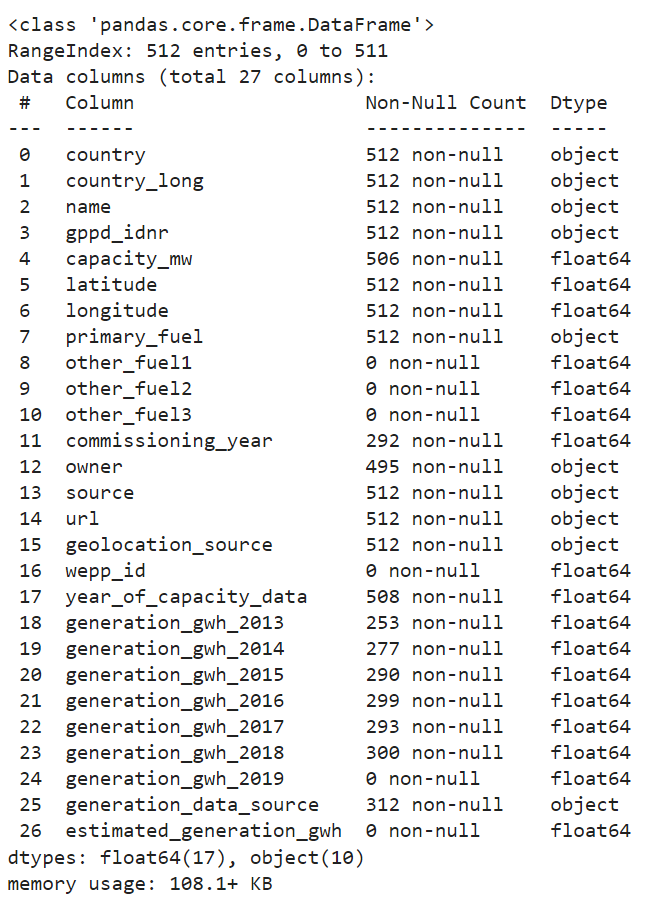
1. All the 3 dfs have equal no of columns, checking if all the 3 dataframes have the same column names or not:



A screen shot of a computer code

Description automatically generated

1. Getting info() on all the three dataframes:



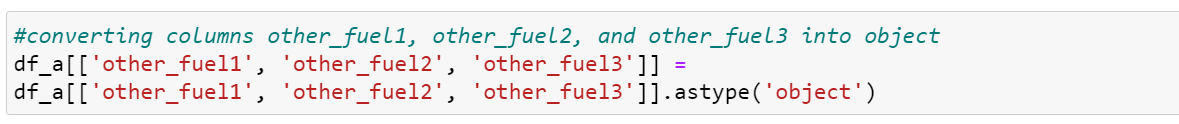
A screenshot of a computer code

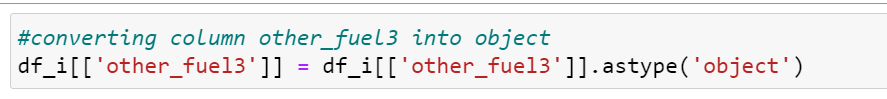
Description automatically generated

A screenshot of a computer

Description automatically generated

1. Converting the columns 'other\_fuel1', 'other\_fuel2', 'other\_fuel3' into object datatype:





the column 'generation\_gwh\_2019' is of type 'object' in df\_i, whereas it is of type float in the other two datatypes.

Checking the df\_i dataset:

A close up of a text

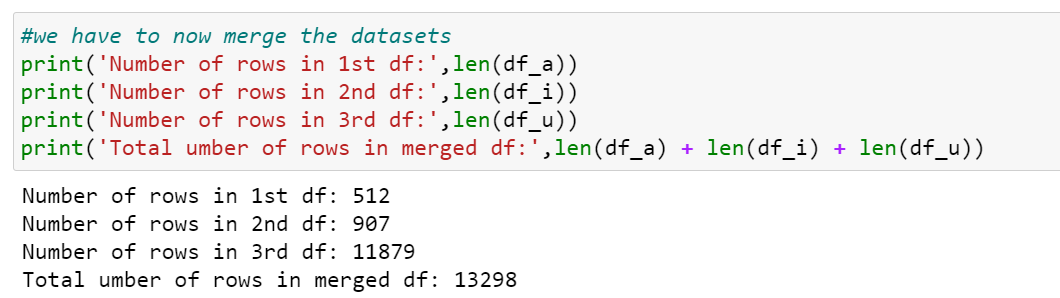
Description automatically generated

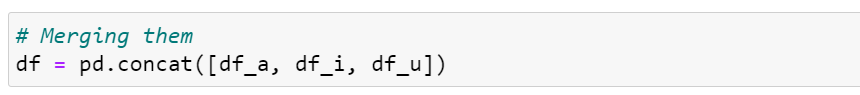
Since it has only 1 value, we can convert it into Nan value, and it can be then changed.

A screenshot of a computer

Description automatically generated

1. Merging the Datasets:





We have now merged the datasets.

1. Checking Null values:

A screenshot of a computer

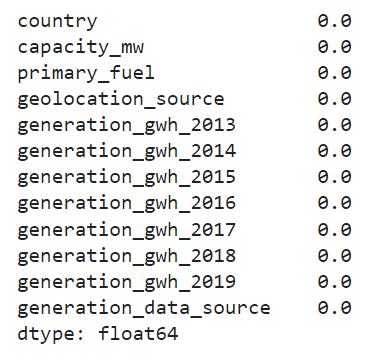
Description automatically generated

We can see that the columns 'wepp\_id' and 'estimated\_generation\_gwh' do not have any values. The columns, 'other\_fuel1', 'other\_fuel2', and 'other\_fuel3' also have very high null values. Let us drop them.

1. Also, let us drop other columns as they contain redundant values and do not give important values:

Columns dropped: 'name','gppd\_idnr', 'latitude','longitude','commissioning\_year','owner', 'source', 'url','year\_of\_capacity\_data', ‘country\_long’

1. Filling the Nan values with median for numerical and with mode for categorical:



1. Plotting histogram and checking the skewness:

A graph of a graph

Description automatically generated with medium confidence

A graph of a number of data

Description automatically generated with medium confidence

Skewness:

A screenshot of a computer code

Description automatically generated

1. Skewness after removing outliers:

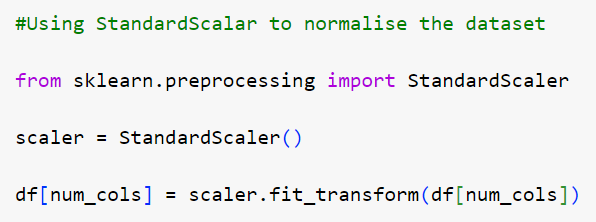
A screenshot of a computer code

Description automatically generated

The skewness remains, still we cannot process the data more as it will reduce the integrity of the data.

### Feature Engineering

1. Applying the Standard Scaler to the numerical data:



1. Checking the cardinality of Primary\_fuel:

A screenshot of a computer

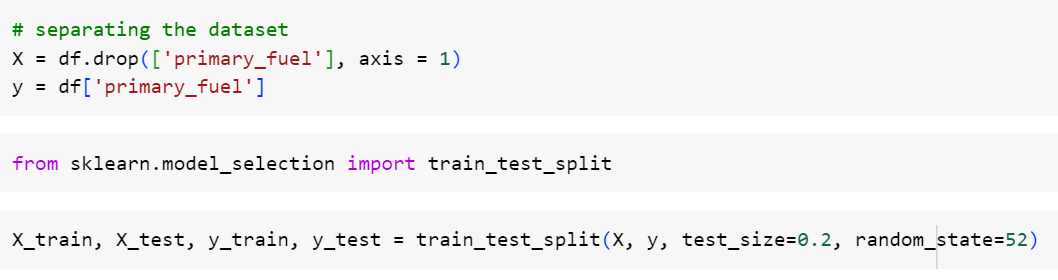
Description automatically generated

1. Encoding the categorical columns:

A computer code with text

Description automatically generated with medium confidence

### Splitting the dataframe into Training and Testing



### Feature Selection

A screenshot of a computer code

Description automatically generated

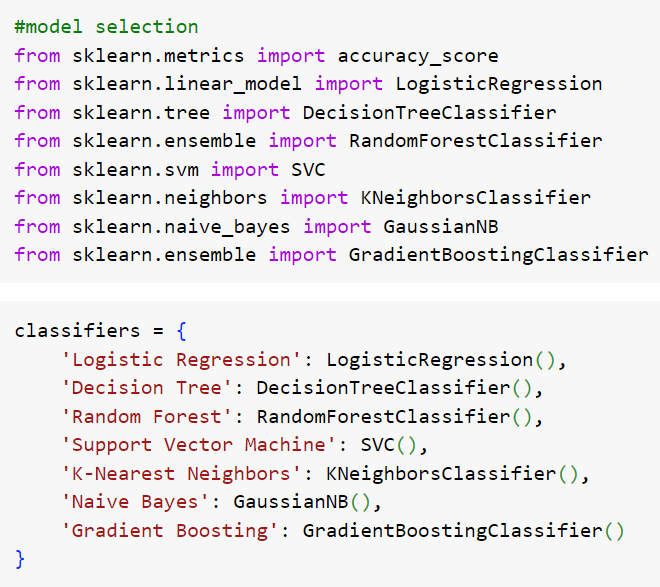
The results which we get are:

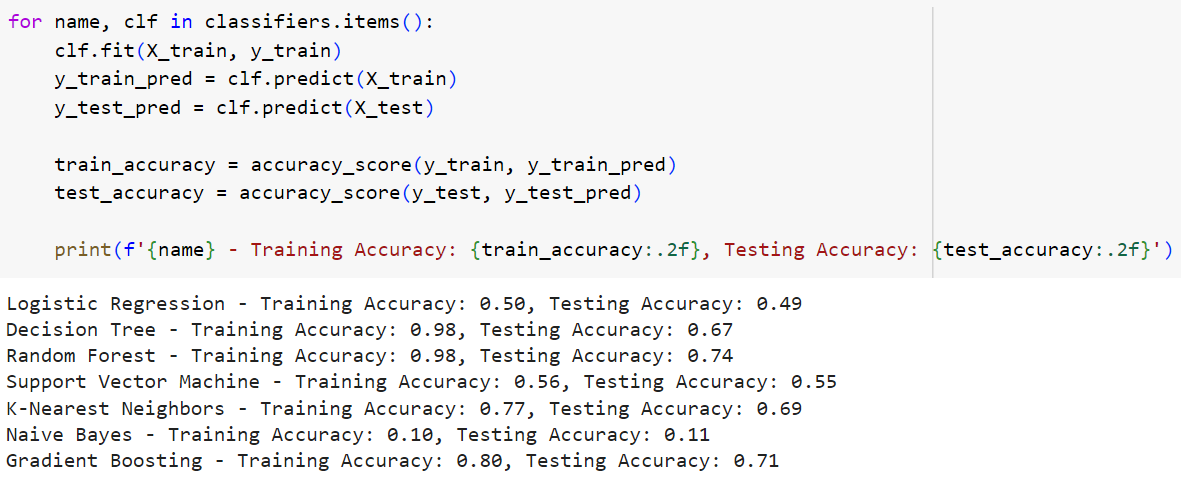
A screenshot of a computer

Description automatically generated

From the above result, we can drop the last 3 datasets.

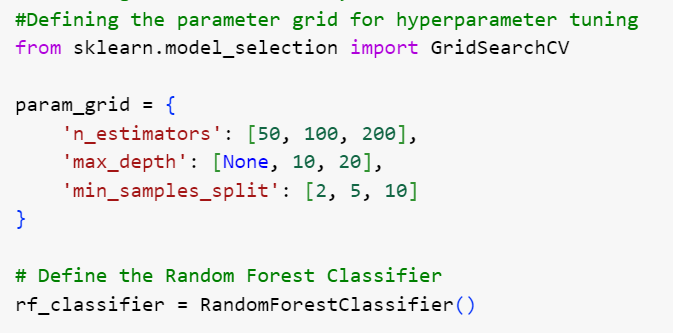
### Model Selection





We can select the Random Forest Classifier. Let us do the hyperparameter tuning for the same.

### Hyperparameter Tuning



A screenshot of a computer program

Description automatically generated

A screenshot of a computer code

Description automatically generated

After evaluating the model on testing set,

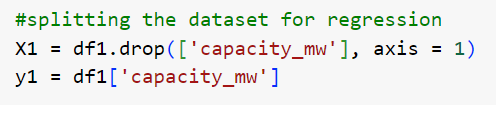
A screenshot of a computer code

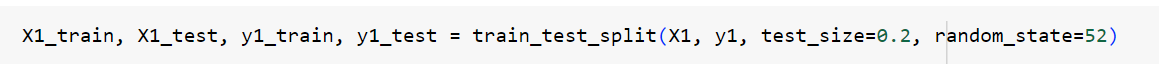
Description automatically generated

Hence, we have selected Random Forest Classifier for the classification of Primary Fuel.

## Code to predict the ‘capacity\_mw’

1. Splitting the dataset





1. Feature selection

A screenshot of a computer code

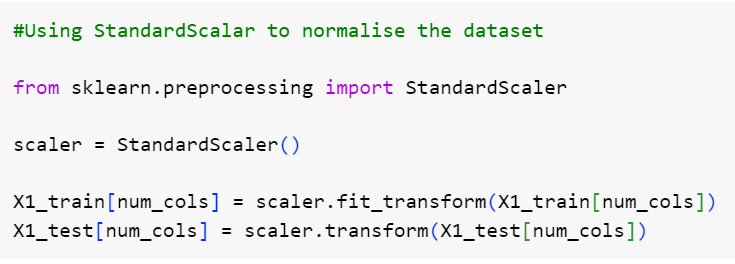
Description automatically generated

A screenshot of a computer

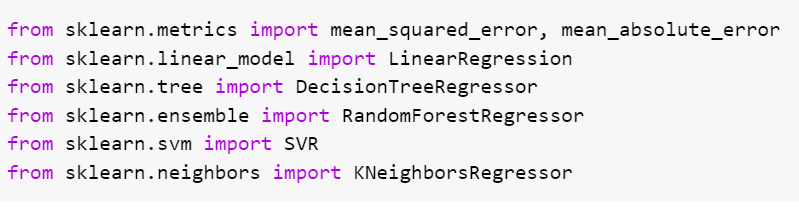
Description automatically generated

We can again see that the last three columns are not important to the model. Hence we can remove the last three columns.

1. Feature Engineering



1. Model Selection



A computer code with text

Description automatically generated

A screenshot of a computer program

Description automatically generated

We get the results as:

A screenshot of a computer program

Description automatically generated

(The models are not performing as we would like them to.)