**SAMPLE EXAMPLE OF RANDOM FOREST ALGORITHM:**

**CODE:**

#The problem here is to predict the gas consumption (in millions of gallons) in 48 of the US states based on petrol tax (in cents),

#per capita income (dollars), paved highways (in miles) and the proportion of population with the driving license.

import pandas as pd

import numpy as np

dataset = pd.read\_csv('petrol\_consumption.csv')

dataset.head()

dataset.tail()

#Preparing Data for training

X = dataset.iloc[:, 0:4].values

y = dataset.iloc[:, 4].values

from sklearn.model\_selection import train\_test\_split

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=0)

#Feature Scaling

from sklearn.preprocessing import StandardScaler

sc = StandardScaler()

X\_train = sc.fit\_transform(X\_train)

X\_test = sc.transform(X\_test)

#Training

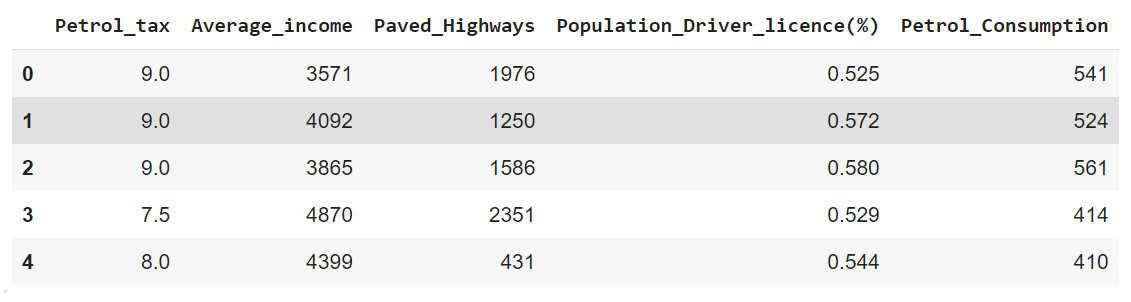
from sklearn.ensemble import RandomForestRegressor

regressor = RandomForestRegressor(n\_estimators=20, random\_state=0)

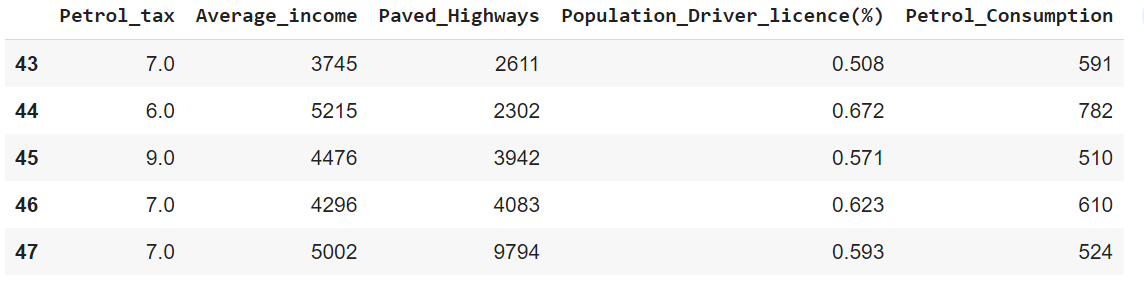
regressor.fit(X\_train, y\_train)

y\_pred = regressor.predict(X\_test)

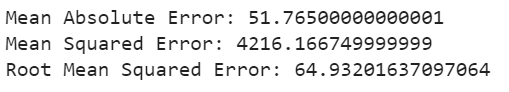
**OUTPUT:**



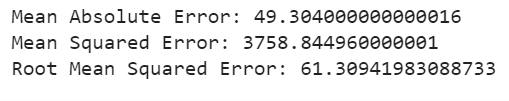
*Data set glimpses*



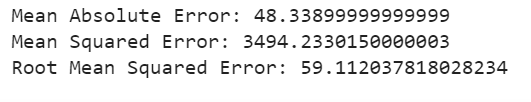
*Data set glimpses*



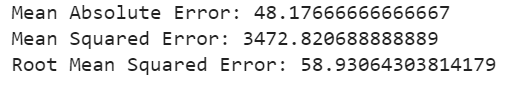
*RMSE with 20 Trees*



*RMSE with 50 Trees*



*RMSE with 200 Trees*



*RMSE with 300 Trees*

*You can see that the error values decreases with the increase in number of estimator. After 200 the rate of decrease in error diminishes, so therefore 200 is a good number for n\_estimators*