LAB TASK 3

Ques: Construct the Linear Regression Plot of Covid Cases.

Dataset Used: https://www.kaggle.com/imdevskp/covid19-coronavirus-india-dataset?select=nation_level_daily.csv

Procedure: -

- Firstly, We import data using Pandas
- Then Decode our date attribute to date time stamp
- We have to select an independent and a dependent attribute to be used in our regression model.
- Next, we have to divide our dataset into training set and test set.
- Initialize our Linear regression model and fit it to the X_train and Y_train.
- Create another variable to store the results of X_test as predicted by our regression model.
- Find the scatter plot of our training sets and the best fit Regression line.
- Find the scatter plot of our test set and the best fit line of the training set
- Finally, Calculate our evaluation metrics to check the accuracy of our model.

CODE

```
#Libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# Importing the dataset
import datetime as dt
data = pd.read_csv("nation_level_daily.csv")
data['Date'] = pd.to_datetime(data['Date'], format = "%d %B ", errors='coerce')
data['Date'] = data['Date'].map(dt.datetime.toordinal)

# Creating the X and Y Variables and setting  Date to X and No of cases on that day in Y
X = data.iloc[123:153, 0].values
y = np.asarray(data.iloc[123:153, 1].values)

# Splitting the datset into Training and Test Set
from sklearn.model_selection import train_test_split
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.5,
random_state=0)

#Training Linear Regression Model
from sklearn.linear_model import LinearRegression
regressor = LinearRegression()
regressor.fit(X_train.reshape(-1,1), y_train)

# Results Prediction
y_pred = regressor.predict(X_test.reshape(-1,1))

# Visualisng the training results
plt.scatter(X_train, y_train, color='red')
plt.plot(X_train, regressor.predict(X_train.reshape(-1,1)), color='green')
plt.title('National_Level Covid cases in the month of June')
plt.xlabel('Date')
plt.ylabel('Number of cases confirmed')
plt.show()

# Visualisng the test results
```

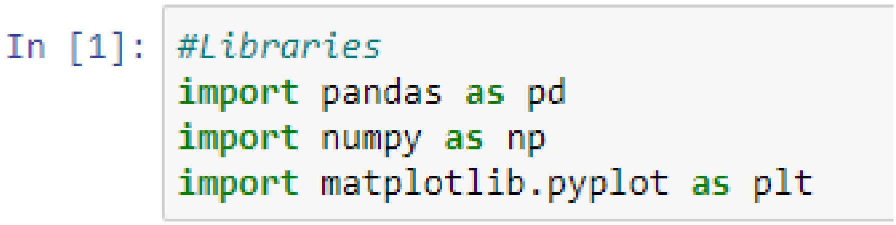
```
plt.scatter(X_test, y_test, color='red')
plt.plot(X_test, y_pred, color='green')
plt.title('National_Level Covid Cases in month of July')
plt.xlabel('Dates')
plt.ylabel('Number of cases confirmed')
plt.show()

#Mean Absolute Error
from sklearn.metrics import mean_absolute_error
mean_absolute_error(y_test, y_pred)

#Mean Squared Error
from sklearn.metrics import mean_squared_error
mean_squared_error(y_test, y_pred)
#Root Mean Squared Error
np.sqrt(mean_squared_error(y_test, y_pred))

#Root Mean Squared Log Error
np.log(np.sqrt(mean_squared_error(y_test, y_pred)))
```

R Square

from sklearn.metrics import r2_score

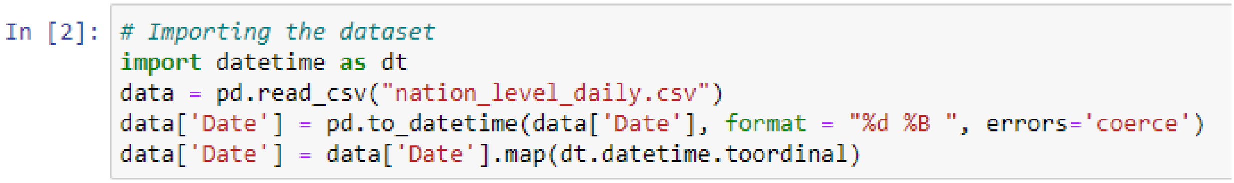
r2_score(y_test, y_pred)

OUTPUT:



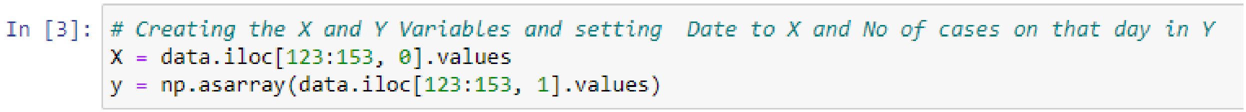
```
In [1]: #Libraries
        import pandas as pd
        import numpy as np
        import matplotlib.pyplot as plt
```

Importing Libraries



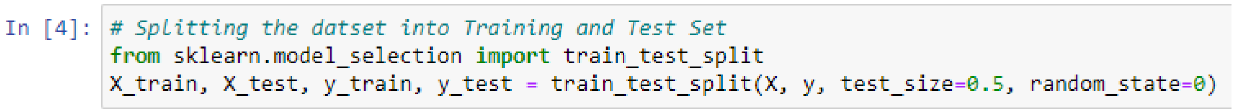
```
In [2]: # Importing the dataset
        import datetime as dt
        data = pd.read_csv("nation_level_daily.csv")
        data['Date'] = pd.to_datetime(data['Date'], format = "%d %B ", errors='coerce')
        data['Date'] = data['Date'].map(dt.datetime.toordinal)
```

Importing the data set and formatting the date into Timestamp



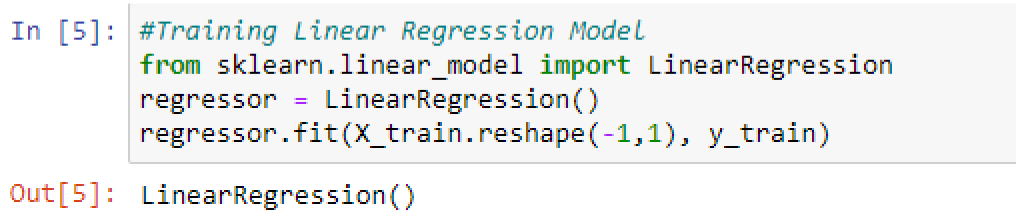
```
In [3]: # Creating the X and Y Variables and setting  Date to X and No of cases on that day in Y
        X = data.iloc[123:153, 0].values
        y = np.asarray(data.iloc[123:153, 1].values)
```

Taking the cases in the month of July from the whole data set



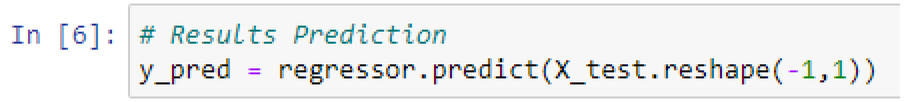
```
In [4]: # Splitting the datset into Training and Test Set
        from sklearn.model_selection import train_test_split
        X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.5, random_state=0)
```

Here, we split our dataset with 50% of data in training set and 50% of the data in test set.



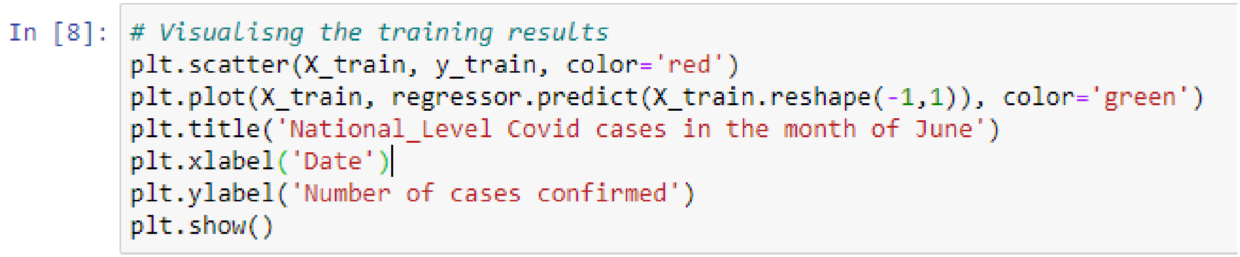
```
In [5]: #Training Linear Regression Model
        from sklearn.linear_model import LinearRegression
        regressor = LinearRegression()
        regressor.fit(X_train.reshape(-1,1), y_train)

Out[5]: LinearRegression()
```

Here we have trained our Linear regression model with training dataset.

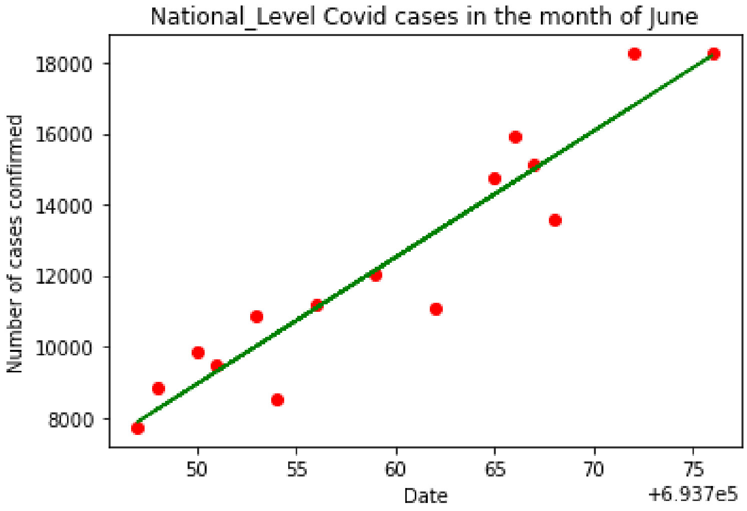


```
In [6]: # Results Prediction
        y_pred = regressor.predict(X_test.reshape(-1,1))
```

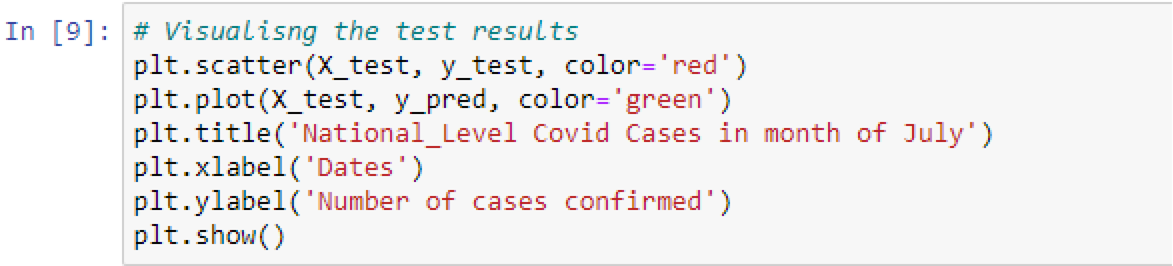
we are creating an array and storing the results of X_test dataset as predicted by our regressor.



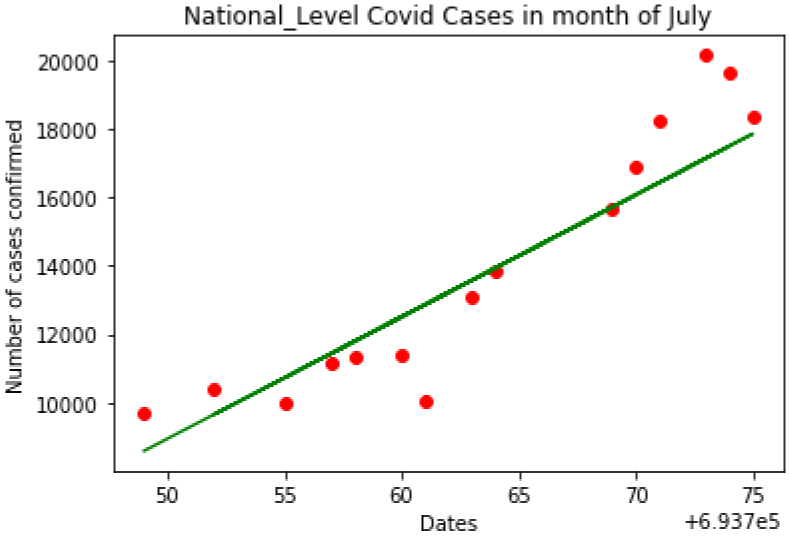
```
In [8]: # Visualisng the training results
        plt.scatter(X_train, y_train, color='red')
        plt.plot(X_train, regressor.predict(X_train.reshape(-1,1)), color='green')
        plt.title('National_Level Covid cases in the month of June')
        plt.xlabel('Date')
        plt.ylabel('Number of cases confirmed')
        plt.show()
```



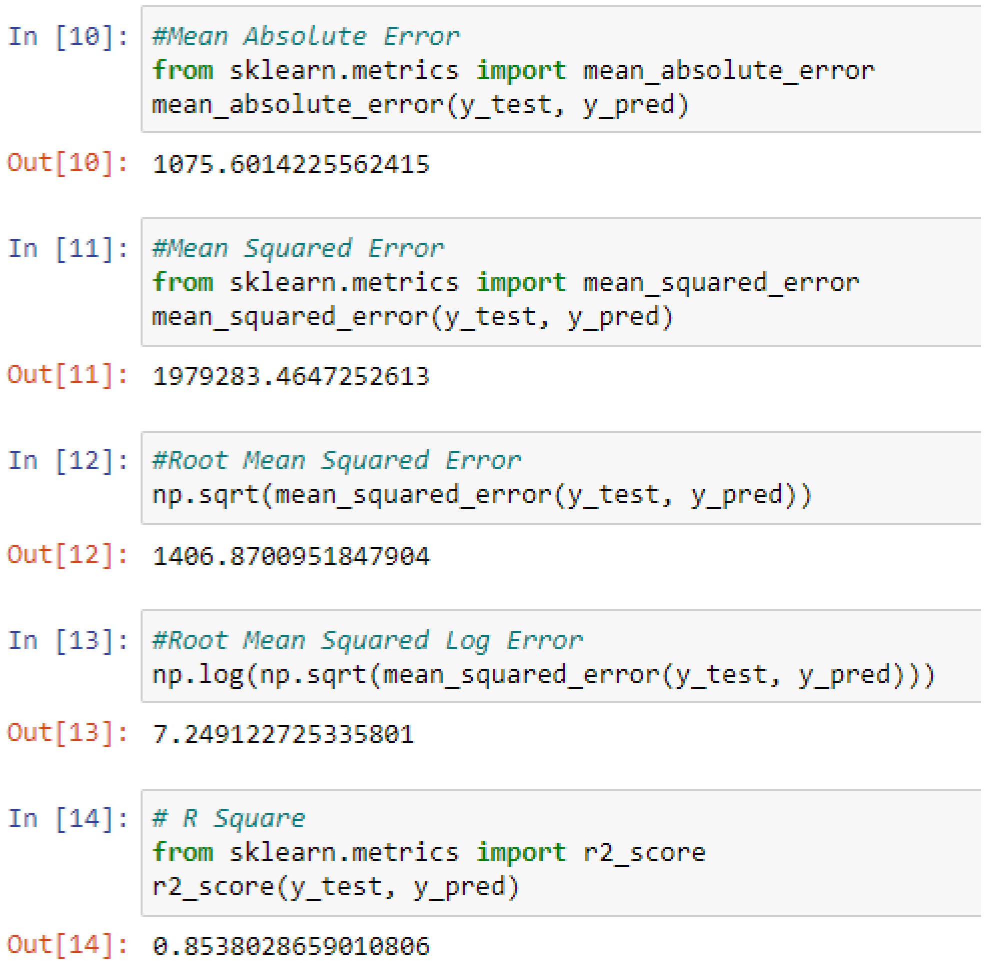
We are plotting the training sets with the best fit regression line. The dates are encoded instead of the whole Date .



```
In [9]: # Visualisng the test results
        plt.scatter(X_test, y_test, color='red')
        plt.plot(X_test, y_pred, color='green')
        plt.title('National_Level Covid Cases in month of July')
        plt.xlabel('Dates')
        plt.ylabel('Number of cases confirmed')
        plt.show()
```



Here we have plotted our test set result with the regression line. Again, here we have dates in encoded format instead of the conventional date format.

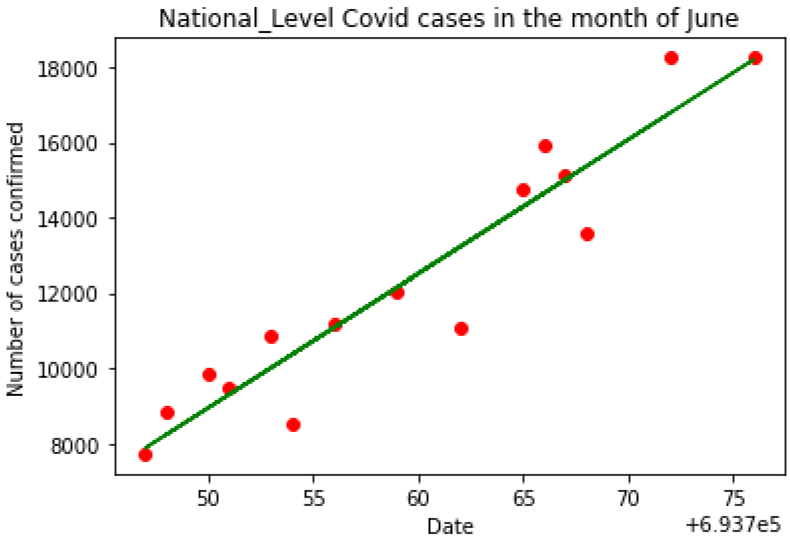


```
In [10]: #Mean Absolute Error
         from sklearn.metrics import mean_absolute_error
         mean_absolute_error(y_test, y_pred)

Out[10]: 1075.6014225562415

In [11]: #Mean Squared Error
         from sklearn.metrics import mean_squared_error
         mean_squared_error(y_test, y_pred)

Out[11]: 1979283.4647252613

In [12]: #Root Mean Squared Error
         np.sqrt(mean_squared_error(y_test, y_pred))

Out[12]: 1406.8700951847904

In [13]: #Root Mean Squared Log Error
         np.log(np.sqrt(mean_squared_error(y_test, y_pred)))

Out[13]: 7.249122725335801

In [14]: # R Square
         from sklearn.metrics import r2_score
         r2_score(y_test, y_pred)

Out[14]: 0.8538028659010806
```

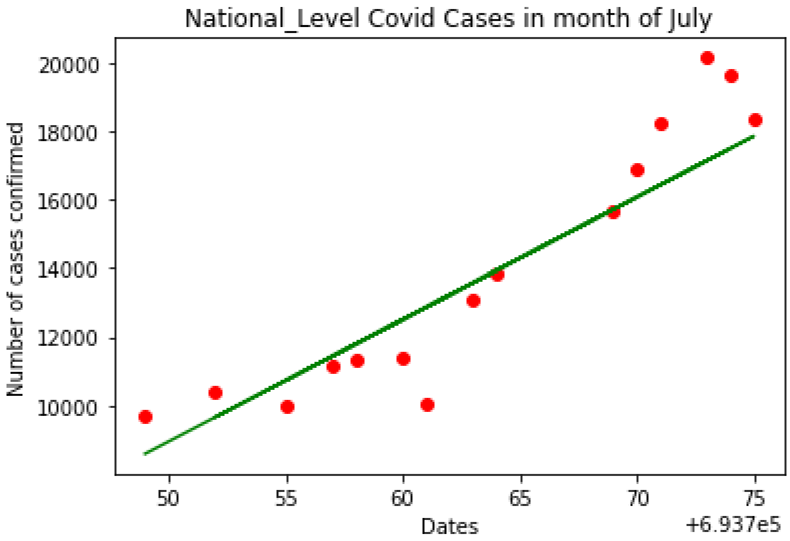
RESULTS:

- Mean Absolute Error : 1075.6014225562415
- Mean Squared Error : 1979283.4647252613
- Root Mean Squared : 1406.8700951847904
- Root Mean Squared Log Error : 7.249122725335801
- R Square Value : 0.8538028659010806

- Training set Plot:



- Test set Plot:


National_Level Covid Cases in month of July
Number of cases confirmed
Dates
+6.937e5