Program_28

October 26, 2022

1 PROGRAM 28

1.0.1 Aim : Multiple regression

1.0.2 Date : 26/10/2022

1.0.3 By : Anu C Scharia

[ ]:

```python
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
```

[ ]:

```python
df=pd.read_csv('FuelConsumption.csv')
df.head()
df.info
```

[ ]:

```
<bound method DataFrame.info of       MODELYEAR   MAKE       MODEL
VEHICLECLASS  ENGINESIZE  CYLINDERS  \
0          2014  ACURA         ILX         COMPACT         2.0          4
1          2014  ACURA         ILX         COMPACT         2.4          4
2          2014  ACURA  ILX HYBRID         COMPACT         1.5          4
3          2014  ACURA     MDX 4WD     SUV - SMALL         3.5          6
4          2014  ACURA     RDX AWD     SUV - SMALL         3.5          6
…         …    …         …             …          …        …
1062       2014  VOLVO    XC60 AWD     SUV - SMALL         3.0          6
1063       2014  VOLVO    XC60 AWD     SUV - SMALL         3.2          6
1064       2014  VOLVO    XC70 AWD     SUV - SMALL         3.0          6
1065       2014  VOLVO    XC70 AWD     SUV - SMALL         3.2          6
1066       2014  VOLVO    XC90 AWD  SUV - STANDARD         3.2          6

     TRANSMISSION FUELTYPE  FUELCONSUMPTION_CITY  FUELCONSUMPTION_HWY   \
0             AS5        Z                   9.9                  6.7
1              M6        Z                  11.2                  7.7
2             AV7        Z                   6.0                  5.8
3             AS6        Z                  12.7                  9.1
4             AS6        Z                  12.1                  8.7
```

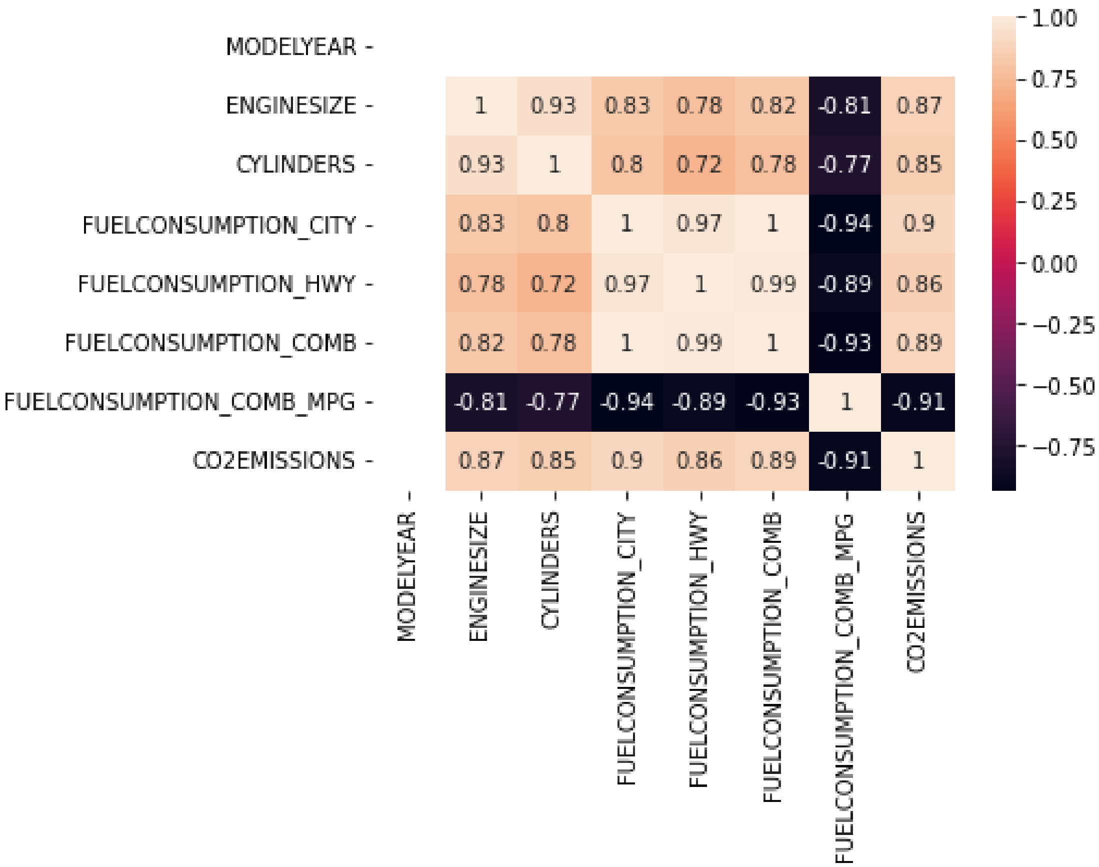
```
…              …      …                   …                  …
1062           AS6         X                  13.4                     9.8
1063           AS6         X                  13.2                     9.5
1064           AS6         X                  13.4                     9.8
1065           AS6         X                  12.9                     9.3
1066           AS6         X                  14.9                    10.2

      FUELCONSUMPTION_COMB  FUELCONSUMPTION_COMB_MPG  CO2EMISSIONS
0                      8.5                        33           196
1                      9.6                        29           221
2                      5.9                        48           136
3                     11.1                        25           255
4                     10.6                        27           244
…                    …                        …            …
1062                  11.8                        24           271
1063                  11.5                        25           264
1064                  11.8                        24           271
1065                  11.3                        25           260
1066                  12.8                        22           294

[1067 rows x 13 columns]>
```

```
[ ]: sns.heatmap(df.corr(),annot=True)
```

```
[ ]: <AxesSubplot:>
```



```
[ ]: cdf=df[['ENGINESIZE','CYLINDERS','FUELCONSUMPTION_CITY','FUELCONSUMPTION_HWY','FUELCONSUMPTION
     cdf.head()
```

```
[ ]:    ENGINESIZE  CYLINDERS  FUELCONSUMPTION_CITY  FUELCONSUMPTION_HWY  \
     0         2.0          4                   9.9                  6.7
     1         2.4          4                  11.2                  7.7
     2         1.5          4                   6.0                  5.8
     3         3.5          6                  12.7                  9.1
     4         3.5          6                  12.1                  8.7

        FUELCONSUMPTION_COMB  CO2EMISSIONS
     0                   8.5           196
     1                   9.6           221
     2                   5.9           136
     3                  11.1           255
     4                  10.6           244
```

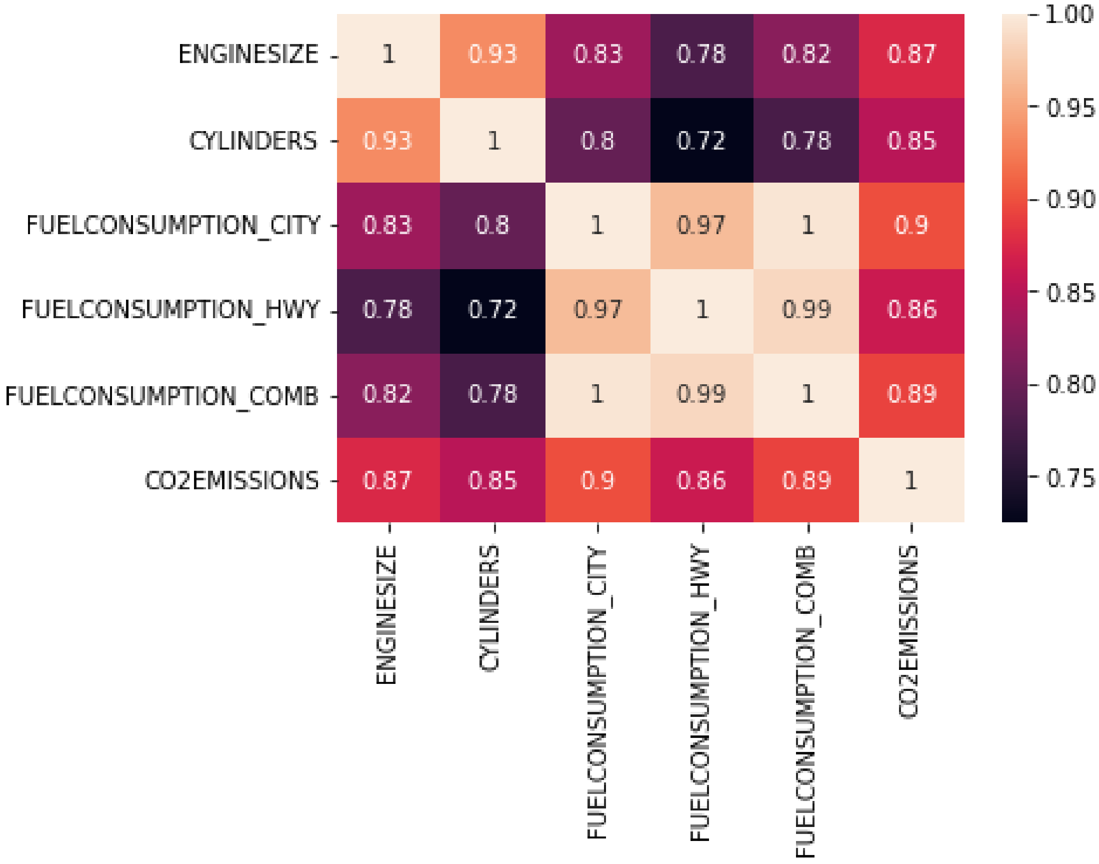
```
[ ]: x= cdf.iloc[:, :-1].values
     y= cdf.iloc[:, -1].values
     print(x)
print(y)
```

```
[[ 2.   4.    9.9  6.7  8.5]
 [ 2.4  4.   11.2  7.7  9.6]
 [ 1.5  4.    6.   5.8  5.9]
 …
 [ 3.   6.   13.4  9.8 11.8]
 [ 3.2  6.   12.9  9.3 11.3]
 [ 3.2  6.   14.9 10.2 12.8]]
[196 221 136 … 271 260 294]
```

[ ]:

```
sns.heatmap(cdf.corr(),annot=True)
```

[ ]: <AxesSubplot:>



	ENGINESIZE	CYLINDERS	FUELCONSUMPTION_CITY	FUELCONSUMPTION_HWY	FUELCONSUMPTION_COMB	CO2EMISSIONS
ENGINESIZE	1	0.93	0.83	0.78	0.82	0.87
CYLINDERS	0.93	1	0.8	0.72	0.78	0.85
FUELCONSUMPTION_CITY	0.83	0.8	1	0.97	1	0.9
FUELCONSUMPTION_HWY	0.78	0.72	0.97	1	0.99	0.86
FUELCONSUMPTION_COMB	0.82	0.78	1	0.99	1	0.89
CO2EMISSIONS	0.87	0.85	0.9	0.86	0.89	1

[ ]:

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=1/3,random_state=1)
```

[ ]:

```
regressor= LinearRegression()
regressor.fit(x_train, y_train)
```

```
[ ]: LinearRegression()
```

```
[ ]: y_test_pred=regressor.predict(x_test)
     y_train_pred=regressor.predict(x_train)
```

```
[ ]: print("Intersect :",regressor.intercept_)
     print("Coffecients :",regressor.coef_)
```

```
Intersect : 67.61239058772759
Coffecients : [11.54382994  6.76843584 -5.55515389 -6.2638602  21.09978711]
```

```
[ ]: print("Residual sum squares : %.2f" % np.mean((y_test_pred-y_test)**2))
```

```
Residual sum squares : 567.73
```

```
[ ]: print("Testing : ",regressor.score(x_test,y_test))
```

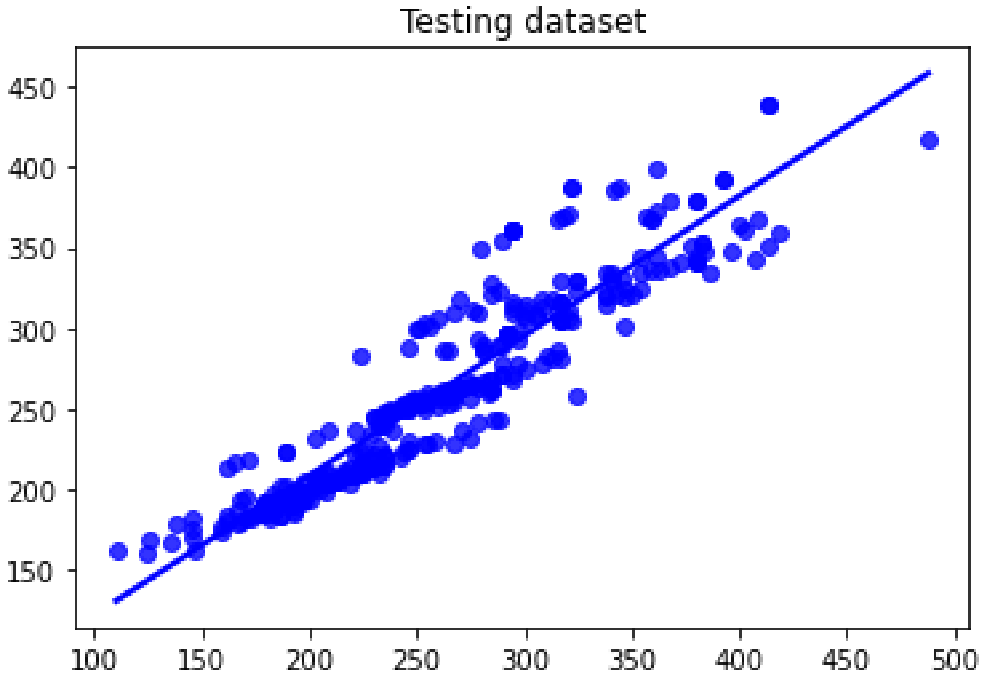
```
Testing :  0.8684262550943238
```

```
[ ]: print("Training : ",regressor.score(x_train,y_train))
```

```
Training :  0.8616244046401831
```

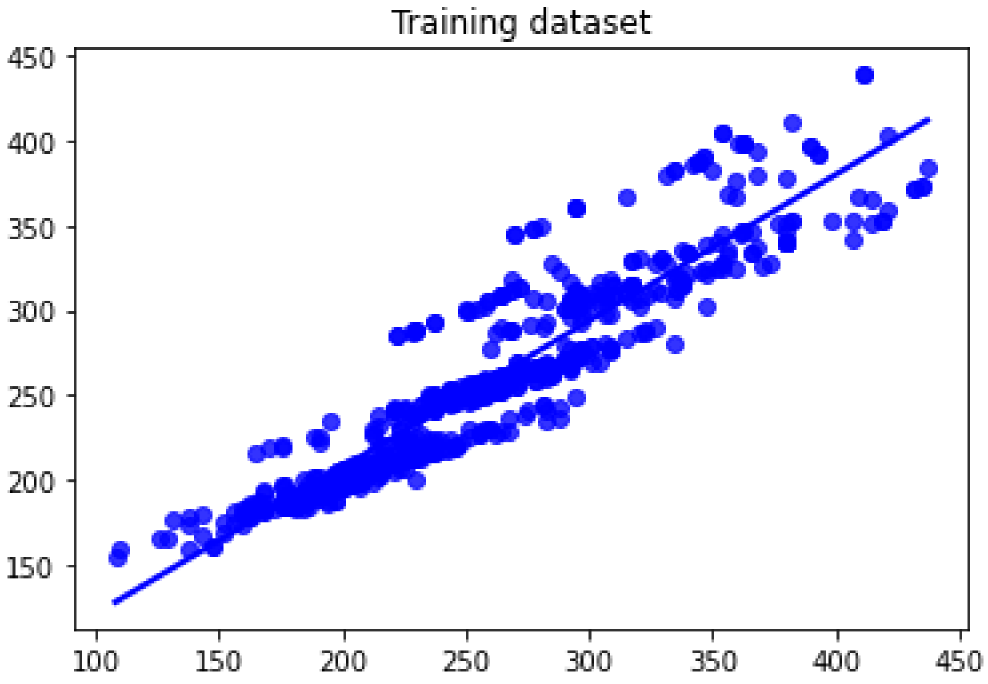
```
[ ]: sns.regplot(x=y_test,y=y_test_pred,ci=None,color="b")
     plt.title("Testing dataset")
```

```
[ ]: Text(0.5, 1.0, 'Testing dataset')
```



```
[ ]: sns.regplot(x=y_train,y=y_train_pred,ci=None,color="b")
     plt.title("Training dataset")
```

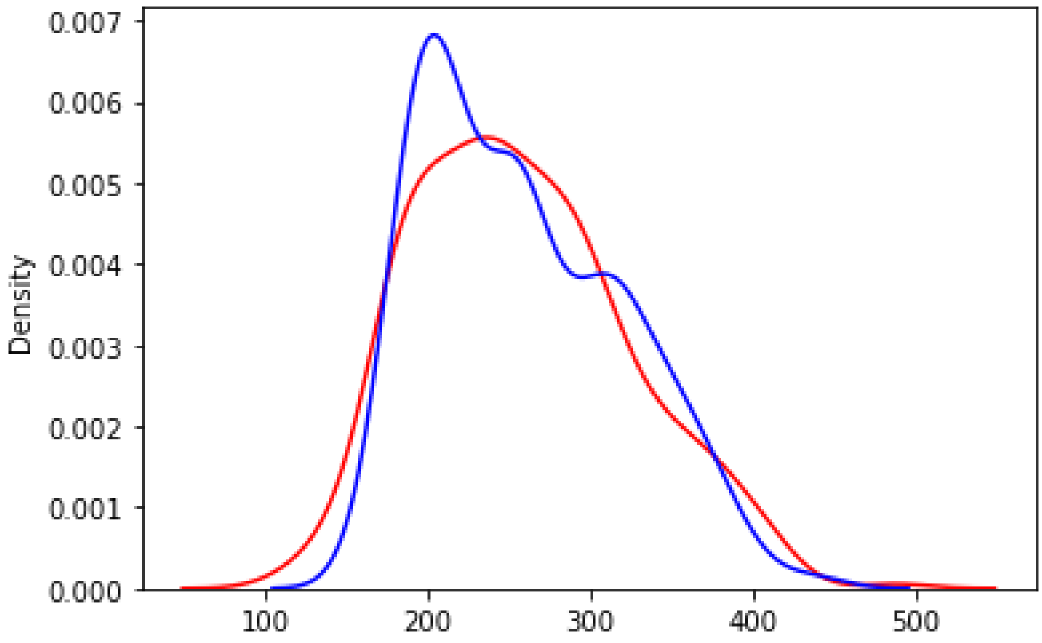
```
[ ]: Text(0.5, 1.0, 'Training dataset')
```



```
[ ]: ax1=sns.distplot(y_test,hist=False,color="r",label="Actual Value")
     sns.distplot(y_test_pred,hist=False,color="b",label="Fitted Values",ax=ax1)
```

```
/opt/anaconda3/lib/python3.9/site-packages/seaborn/distributions.py:2619:
FutureWarning: `distplot` is a deprecated function and will be removed in a
future version. Please adapt your code to use either `displot` (a figure-level
function with similar flexibility) or `kdeplot` (an axes-level function for
kernel density plots).
  warnings.warn(msg, FutureWarning)
/opt/anaconda3/lib/python3.9/site-packages/seaborn/distributions.py:2619:
FutureWarning: `distplot` is a deprecated function and will be removed in a
future version. Please adapt your code to use either `displot` (a figure-level
function with similar flexibility) or `kdeplot` (an axes-level function for
kernel density plots).
  warnings.warn(msg, FutureWarning)
```

```
[ ]: <AxesSubplot:ylabel='Density'>
```



[ ]:

```
 
```

[ ]:

```
x= cdf.iloc[:, :3].values
y= cdf.iloc[:, -1].values
print(x)
print(y)
```

```
[[ 2.   4.   9.9]
 [ 2.4  4.  11.2]
 [ 1.5  4.   6. ]
 …
 [ 3.   6.  13.4]
 [ 3.2  6.  12.9]
 [ 3.2  6.  14.9]]
[196 221 136 … 271 260 294]
```

[ ]:

```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size= 1/
 ↪3,random_state=0)
```

[ ]:

```
regressor= LinearRegression()
regressor.fit(x_train, y_train)
```

[ ]:

```
LinearRegression()
```

[ ]:

```
y_pred= regressor.predict(x_test)
```

```
[ ]: print("Testing : ",regressor.score(x_test,y_test))
```

```
Testing :  0.8418565610806696
```

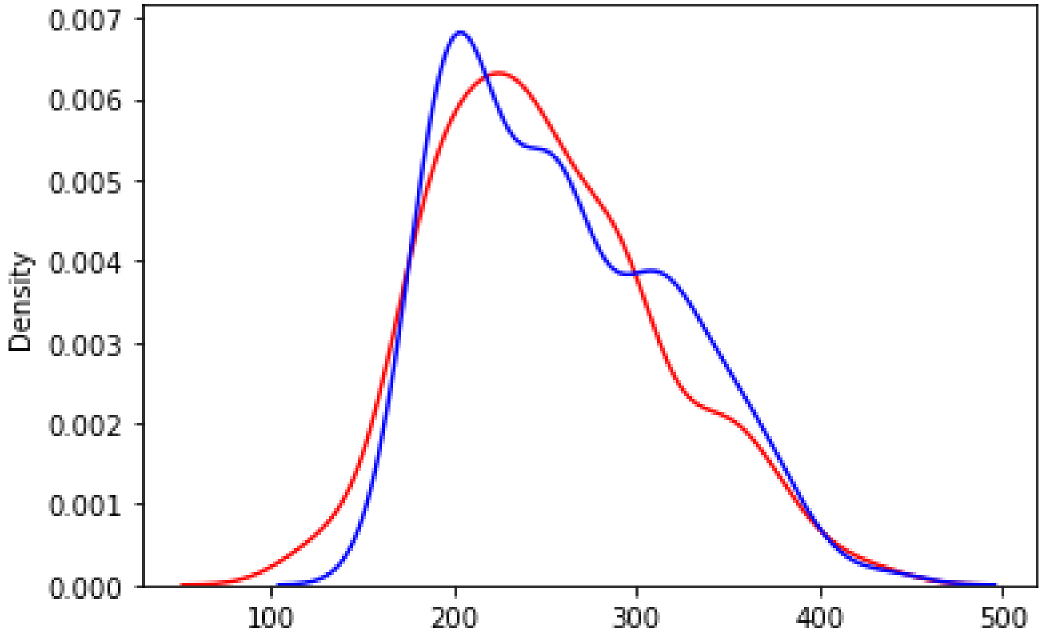
```
[ ]: print("Training : ",regressor.score(x_train,y_train))
```

```
Training :  0.8710231041815493
```

```
[ ]: ax1=sns.distplot(y_test,hist=False,color="r",label="Actual Value")
     sns.distplot(y_test_pred,hist=False,color="b",label="Fitted Values",ax=ax1)
```

```
/opt/anaconda3/lib/python3.9/site-packages/seaborn/distributions.py:2619:
FutureWarning: `distplot` is a deprecated function and will be removed in a
future version. Please adapt your code to use either `displot` (a figure-level
function with similar flexibility) or `kdeplot` (an axes-level function for
kernel density plots).
  warnings.warn(msg, FutureWarning)
/opt/anaconda3/lib/python3.9/site-packages/seaborn/distributions.py:2619:
FutureWarning: `distplot` is a deprecated function and will be removed in a
future version. Please adapt your code to use either `displot` (a figure-level
function with similar flexibility) or `kdeplot` (an axes-level function for
kernel density plots).
  warnings.warn(msg, FutureWarning)
```

```
[ ]: <AxesSubplot:ylabel='Density'>
```

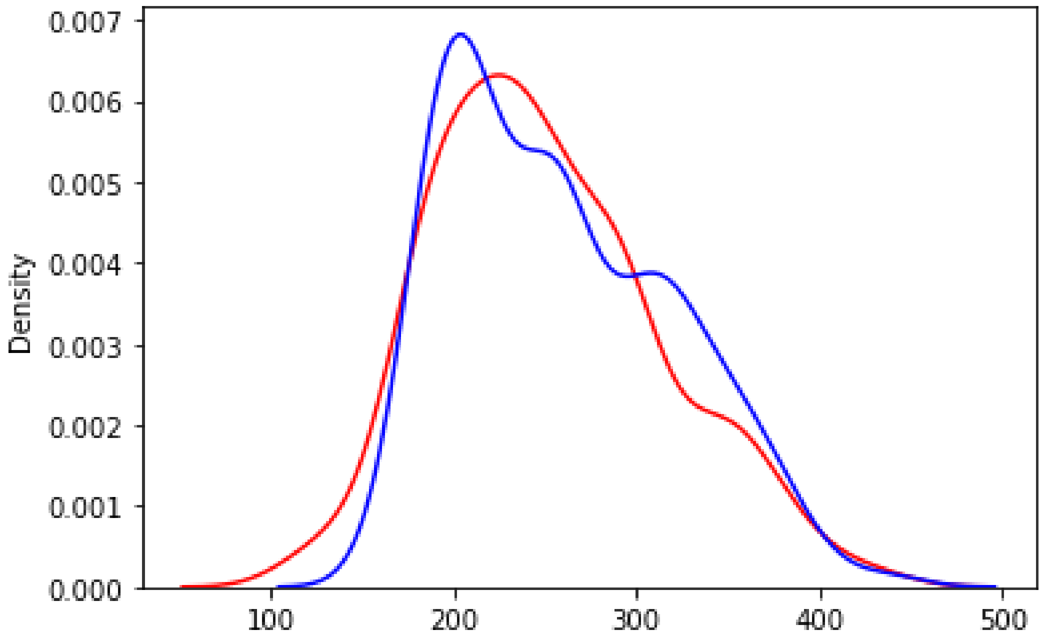


[ ]:
```
x= df[["FUELCONSUMPTION_CITY"]].values
y= df[["CO2EMISSIONS"]].values
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size= 1/
 ↪3,random_state=0)
regressor= LinearRegression()
regressor.fit(x_train, y_train)
y_pred= regressor.predict(x_test)
print("Testing : ",regressor.score(x_test,y_test))
print("Training : ",regressor.score(x_train,y_train))
ax1=sns.distplot(y_test,hist=False,color="r",label="Actual Value")
sns.distplot(y_test_pred,hist=False,color="b",label="Fitted Values",ax=ax1)
```

```
Testing :  0.7858531969002418
Training :  0.8143158005420086
```

```
/opt/anaconda3/lib/python3.9/site-packages/seaborn/distributions.py:2619:
FutureWarning: `distplot` is a deprecated function and will be removed in a
future version. Please adapt your code to use either `displot` (a figure-level
function with similar flexibility) or `kdeplot` (an axes-level function for
kernel density plots).
  warnings.warn(msg, FutureWarning)
/opt/anaconda3/lib/python3.9/site-packages/seaborn/distributions.py:2619:
FutureWarning: `distplot` is a deprecated function and will be removed in a
future version. Please adapt your code to use either `displot` (a figure-level
function with similar flexibility) or `kdeplot` (an axes-level function for
kernel density plots).
  warnings.warn(msg, FutureWarning)
```

[ ]:
```
<AxesSubplot:ylabel='Density'>
```



[ ]:

```
```