**Assignment No 4**

**Code:**

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

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

from sklearn.linear\_model import LinearRegression

from sklearn.metrics import mean\_absolute\_error,mean\_squared\_error,r2\_score

print("\*\*\*\*\* Housing dataset\*\*\*\*\*")

df=pd.read\_csv('Boston.csv')

print("------Data-------")

print(df)

print("--------NUll value counts----")

print(df.isnull().sum())

df['crim'].fillna(int(df['crim'].mean()))

df['zn'].fillna(int(df['zn'].mean()))

df['indus'].fillna(int(df['indus'].mean()))

df['chas'].fillna(int(df['chas'].mean()))

df['age'].fillna(int(df['age'].mean()))

df['lstat'].fillna(int(df['lstat'].mean()))

print("--------NUll value count After filling null values----")

print(df.isnull().sum())

# feature for prediction(you can modify based on you requirements )

X= df[['rm','lstat','crim']]

Y= df['medv']

# splits data into training and testing sets

X\_train,X\_test,Y\_train,Y\_test= train\_test\_split(X,Y,test\_size=0.2,random\_state=42)

# create a linear regresion model

model=LinearRegression()

#Train the model on training set

model.fit(X\_train,Y\_train)

#Make prediction on the test set

Y\_pred=model.predict(X\_test)

print(Y\_pred)

#Evalute the model

mse=mean\_squared\_error(Y\_test,Y\_pred)

r2=r2\_score(Y\_test,Y\_pred)

print(f'Mean Squared Error:{mse}')

print(f'R-squared:{r2}')

#plot the  regression line

plt.scatter(Y\_test,Y\_pred)

plt.plot([min(Y\_test),max(Y\_test)],(min(Y\_test),max(Y\_test)),linestyle="--",color="red",linewidth=2)

plt.title("linear regression model for home prices")

plt.xlabel("actual Prices")

plt.ylabel("predicted prices")

plt.show()

**Output:**

\*\*\*\*\* Housing dataset\*\*\*\*\*

------Data-------

     Unnamed: 0     crim    zn  indus  chas  ...  tax  ptratio   black  lstat  medv

0             1  0.00632  18.0   2.31     0  ...  296     15.3  396.90   4.98  24.0

1             2  0.02731   0.0   7.07     0  ...  242     17.8  396.90   9.14  21.6

2             3  0.02729   0.0   7.07     0  ...  242     17.8  392.83   4.03  34.7

3             4  0.03237   0.0   2.18     0  ...  222     18.7  394.63   2.94  33.4

4             5  0.06905   0.0   2.18     0  ...  222     18.7  396.90   5.33  36.2

..          ...      ...   ...    ...   ...  ...  ...      ...     ...    ...   ...

501         502  0.06263   0.0  11.93     0  ...  273     21.0  391.99   9.67  22.4

502         503  0.04527   0.0  11.93     0  ...  273     21.0  396.90   9.08  20.6

503         504  0.06076   0.0  11.93     0  ...  273     21.0  396.90   5.64  23.9

504         505  0.10959   0.0  11.93     0  ...  273     21.0  393.45   6.48  22.0

505         506  0.04741   0.0  11.93     0  ...  273     21.0  396.90   7.88  11.9

[506 rows x 15 columns]

--------NUll value counts----

Unnamed: 0    0

crim          0

zn            0

indus         0

chas          0

nox           0

rm            0

age           0

dis           0

rad           0

tax           0

ptratio       0

black         0

lstat         0

medv          0

dtype: int64

--------NUll value count After filling null values----

Unnamed: 0    0

crim          0

zn            0

indus         0

chas          0

nox           0

rm            0

age           0

dis           0

rad           0

tax           0

ptratio       0

black         0

lstat         0

medv          0

dtype: int64

[25.70806039 30.71362566 18.22383291 25.71515062 19.81083607 23.29855211

 17.66554117 15.85839515 22.02566447 20.55056533 18.12811793 18.87680049

 -6.23166075 22.83377476 20.20838777 26.83193493 17.91705738  3.25229357

 37.2854162  18.17578857 26.08660543 27.11538265 14.04742314 26.34956916

 18.82144947 14.08512233 22.64140268 20.86758352 18.5347124  19.36965278

 17.81246012 26.78539779 25.84110666 18.71840524 15.97917786 17.71356896

 32.82742004 22.22743005 20.58380803 25.57826988 13.16166461 28.98310191

 38.2251108  18.8323839  25.84808473 16.39890175 16.35851449 27.01376564

 19.72416762 28.88315054 21.0568934  31.29776404 18.60985948 28.51595004

 34.84645848 23.81843629 19.68360014 31.40490298 25.23711546 15.87874437

 27.38290782 32.7504841  29.38377978 18.76132866 28.61075227 11.79362736

 20.41860447 26.37642986 29.36219518 17.09149229 19.30679544 27.60865461

 12.72091418 24.96316327 23.47459281  3.49451016 22.4651945  36.68905007

 17.72572902 11.72382874 22.87114258 10.60237835 22.57457618  7.23512501

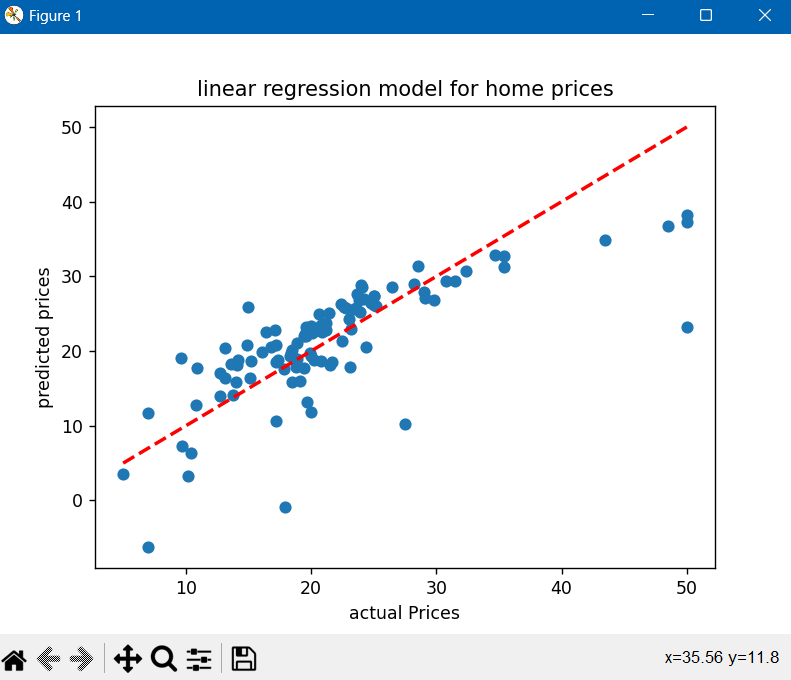
 21.98567063 27.95377255 22.58305673 27.30729957 26.15370352 22.77930243

 22.89457694  6.41424032 23.26367933 20.77858961 10.24755184 24.23466736

 23.17908278 -0.81503491 19.13340778 18.50069374 21.32830737 25.03594715]

Mean Squared Error:32.282339265462326

R-squared:0.5597889830000125

****