import numpy as np

import pandas as pd

import scipy.stats as stats

import matplotlib.pyplot as plt

import sklearn

from sklearn.datasets import load\_boston

boston = load\_boston()

bos =pd.DataFrame(boston.data,columns=boston.feature\_names)

X=bos.iloc[:,:].values

Y=boston.target

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

x\_train,x\_test,y\_train,y\_test=train\_test\_split(X,Y,test\_size=.35,random\_state=2)

from sklearn.linear\_model import LinearRegression

LR=LinearRegression()

LR.fit(x\_train,y\_train)

y\_pred=LR.predict(x\_test) from sklearn.metrics import mean\_squared\_error, r2\_score

model\_score = LR.score(x\_train,y\_train)

# Have a look at R sq to give an idea of the fit ,Explained variance score: 1 is perfect prediction

print('R sq: ',model\_score)

print('Coefficients: ', LR.coef\_)

# The mean squared error

print("Mean squared error: %.2f"

% mean\_squared\_error(y\_test, y\_pred))

# Explained variance score: 1 is perfect prediction

print('Variance score: %.2f' % r2\_score(y\_test, y\_pred))