**U18ISI6204 – Machine Learning Techniques**

**LAB- EXPERIMENT 2**

**NAME:** Navina

**ROLL\_NO:** 20BIS028

# **OBJECTIVE OF THE EXERCISE/EXPERIMENT**

To perform Linear regression in single and multiple variables on the given dataset, using scikit library

**STEP-1:** Start the program.

**STEP-2:** import all the necessary libraries

i) Numpy – array manipulation

ii) Pandas – dataframe manipulation

iii) Matplotlib and seaborn – for data visualization

iv) Sklearn.model\_selection – train test data split

v) Sklearn.metrics – mean square error and r2 score.

vi) Sklearn,linear\_model – for linear regression

**STEP-3:** Loading the dataset using read\_csv method in pandas module.

**STEP-4:** Analyze the dataset using info method, which gives its data types and number of non- null values in each columns.

**STEP-5:** Perform basic statistic operation using describe() method.

**STEP-6:** Use heatmaps, correlation matrix, regression plots and pairplots in seaborn to find the relationship between features.

**STEP-7:** Implement Simple Linear regression(singleLR) with only one variable

(X3 distance to the nearest MRT station) and calculate the MSE and R2 score for the singleLR model.

**STEP-8:** Implement Multiple Linear regression(multiLR) with selected variable (refined cols) which are pick out by analyzing the relationship between features and calculate the MSE and R2 score for the multiLR model.

**STEP-9:** Stop the program.

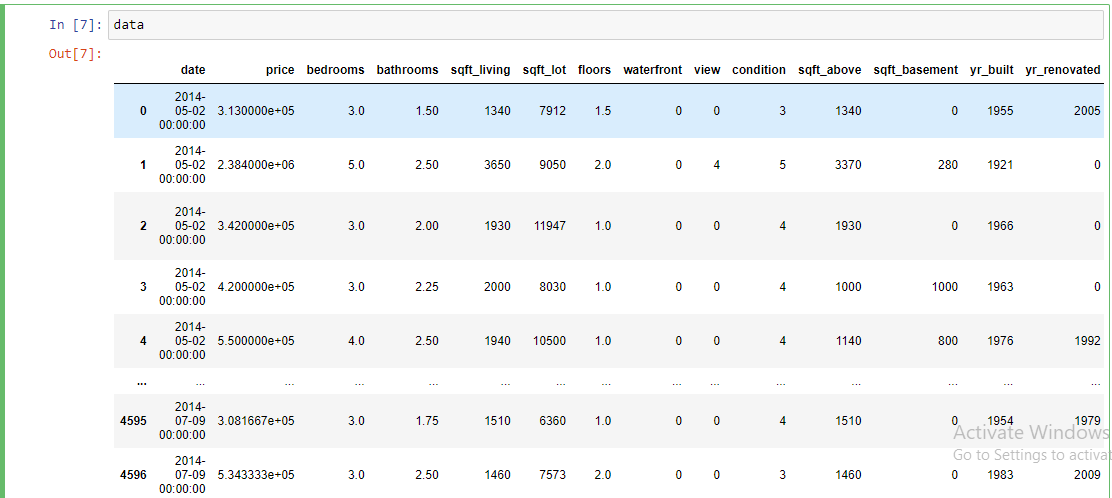
**PRICE PREDICTION DATASET**

**SYNTAX:**

import pandas as pd

data=pd.read\_csv("C:/Users/MADL22/Downloads/archive/data.csv")

Data

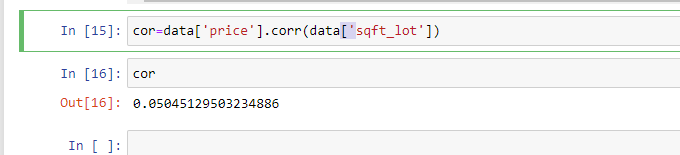


**CORRELATION:**

**SYNTAX:**

cor=data['price'].corr(data['sqft\_lot'])

Cor



**REGRESSION:**

**SYNTAX:**

import matplotlib.pyplot as plt

from scipy import stats

x=data['sqft\_living']

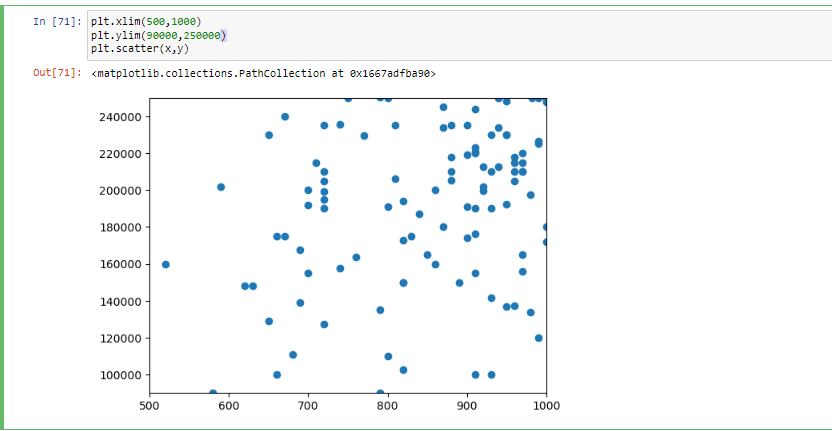
y=data['price']

slope, intercept, r, p, std\_err = stats.linregress(x, y)

plt.xlim(500,1000)

plt.ylim(90000,250000)

plt.scatter(x,y)



**SLOPE EQUATION:**

**SYNTAX:**

def myfunc(x):

return slope \* x + intercept

mymodel = list(map(myfunc, x))

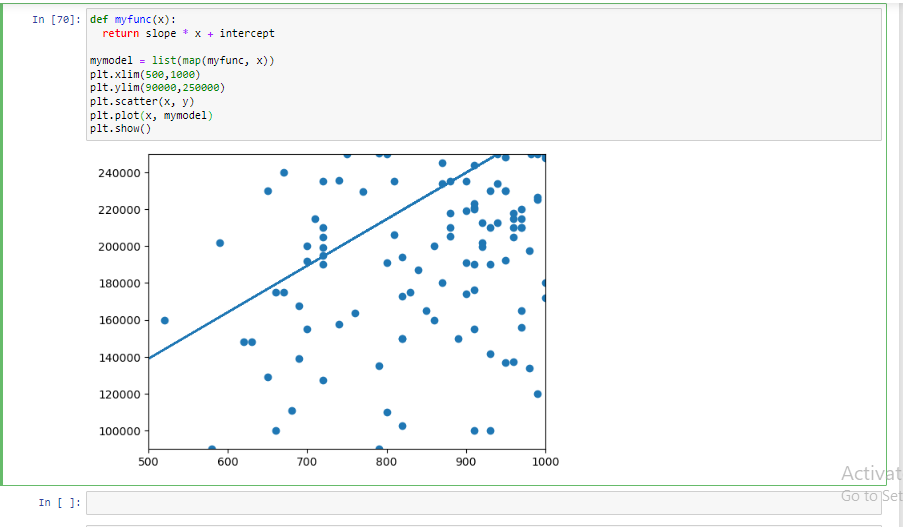
plt.xlim(500,1000)

plt.ylim(90000,250000)

plt.scatter(x, y)

plt.plot(x, mymodel)

plt.show()



**GRAPH 2:**

**SYNTAX:**

x=data['sqft\_lot']

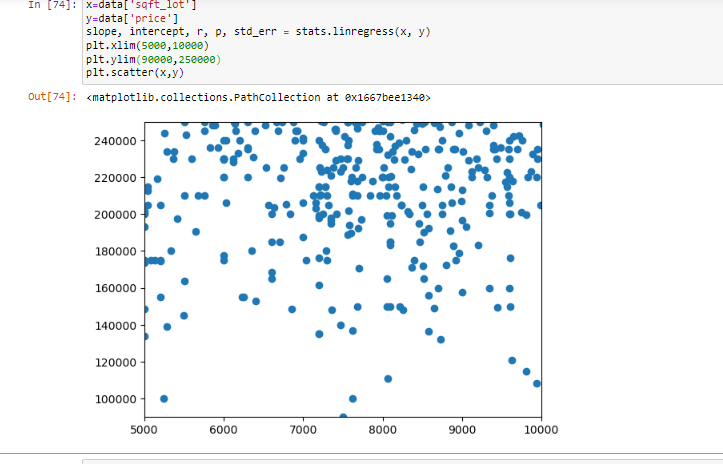
y=data['price']

slope, intercept, r, p, std\_err = stats.linregress(x, y)

plt.xlim(5000,10000)

plt.ylim(90000,250000)

plt.scatter(x,y)



**SLOPE:**

def myfunc(x):

return slope \* x + intercept

mymodel = list(map(myfunc, x))

plt.xlim(300000,100000)

plt.ylim(100000,2500000)

plt.scatter(x, y)

plt.plot(x, mymodel)

plt.show()

