**Experiment 3**

**Student Name:** Sahil Kaundal **UID:** 21BCS8197

**Branch:** BE CSE (Lateral Entry) **Section/Group:** 616/A

**Semester:** 5th **Date of Performance:** 04/06/2022

**Subject Name:** ML Lab **Subject Code:** 20CSP-317

1. **Aim/Overview of the practical:**

To Implement Linear Regression on any data set.

1. **Task To Be Done:**

To Implement Linear Regression on any data set.

**3. Apparatus / Simulator Used:**

1. Windows 7 or above.
2. Google Collab.

Python provides various types of libraries that comes with different types of features which can support various types of graphs. These libraries are:

Matplotlib, Seaborn, Bokeh, Plotly

**Matplotlib:** It is easy to use low level library built on Numpy arrays. It consists of various plots like scatter plot, line plot, histogram etc. Matplotlib provides a lot of flexibility.

**Steps:**

Step 1: Importing Python libraries.

Step 2: Creating the dataset.

Step 3: Opening the dataset.

Step 4: Uploading the dataset.

Step 5: Feature Scaling and Normalization.

Step 6: Add a column of ones to the X vector.

Step 7: Plotting the dataset

Step 8: The Hypothesis (Linear Regression Model)

Step 9: Calculating the Cost Function

Step 10: Gradient Descent

Step 11: Predictions

Step 12: Intuitions concerning the learning constant α

Step 13: Contour plot of J and θ

Step 14: How to adapt the code for Multiple Variables

**4. Program / Commands:**

#Sahil Kaundal

#21BCS8197

import pandas as pd

import seaborn as sns

import matplotlib.pyplot as plt

import numpy as np

import sklearn

data = pd.read\_csv('/housing.csv')

data.head()

data.plot.scatter('RM','B')

x = data[['NOX',  'RM', 'DIS',  'RAD',  'TAX',  'PTRATIO',  'B']]

y = data['MEDV']

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

from sklearn.linear\_model import LinearRegression

x\_train, x\_test, y\_train, y\_test = train\_test\_split(x,y, test\_size=0.4)

data.isnull().sum()

lm = LinearRegression()

lm.fit(x\_train, y\_train)

predictions = lm.predict(x\_test)

plt.scatter(y\_test,predictions)

plt.xlabel('Y Test')

plt.ylabel('Predicted Y')

from sklearn import metrics

print('MAE:', metrics.mean\_absolute\_error(y\_test, predictions))

print('MSE:', metrics.mean\_squared\_error(y\_test, predictions))

print('RMSE:', np.sqrt(metrics.mean\_squared\_error(y\_test, predictions)))

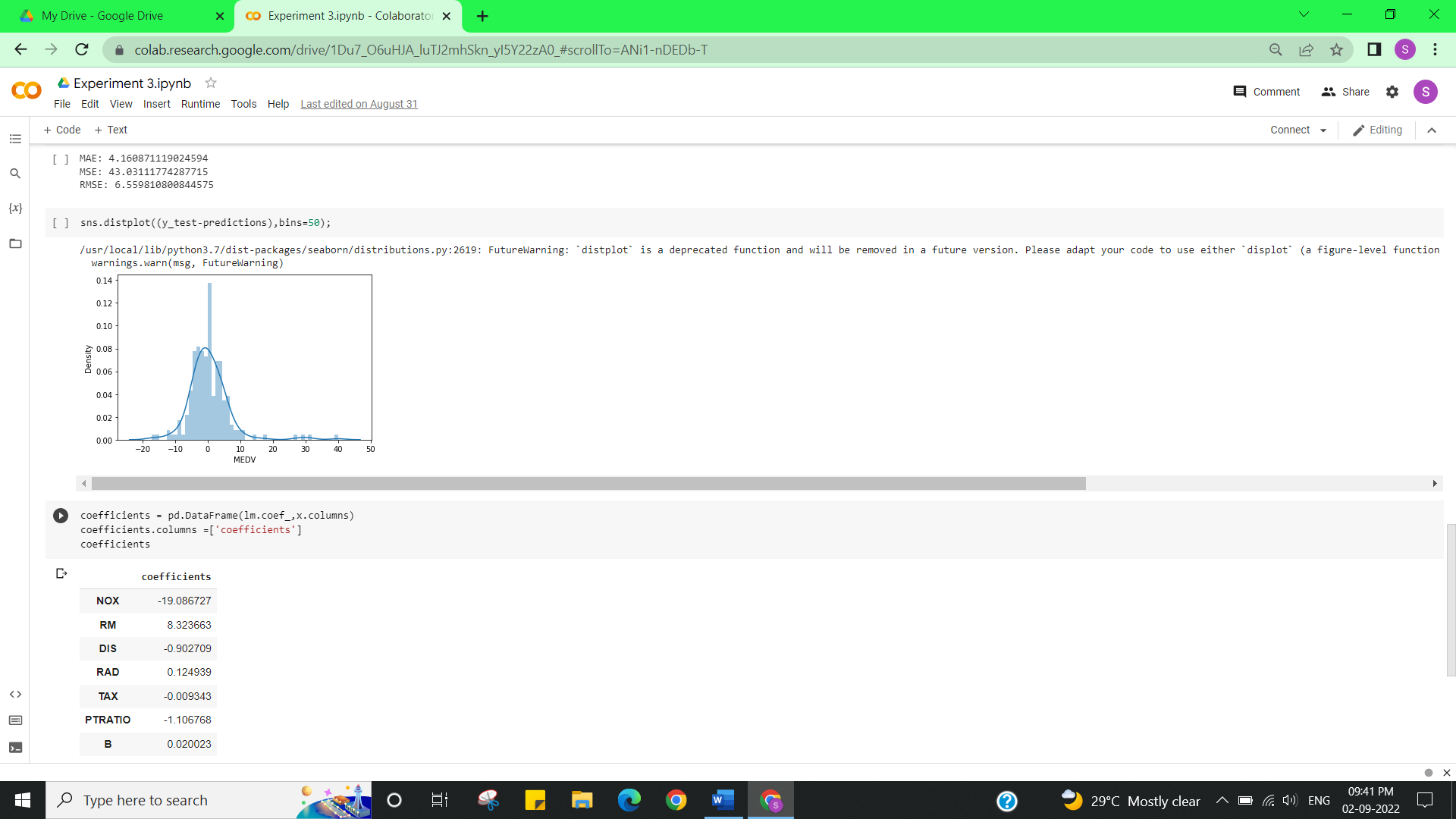
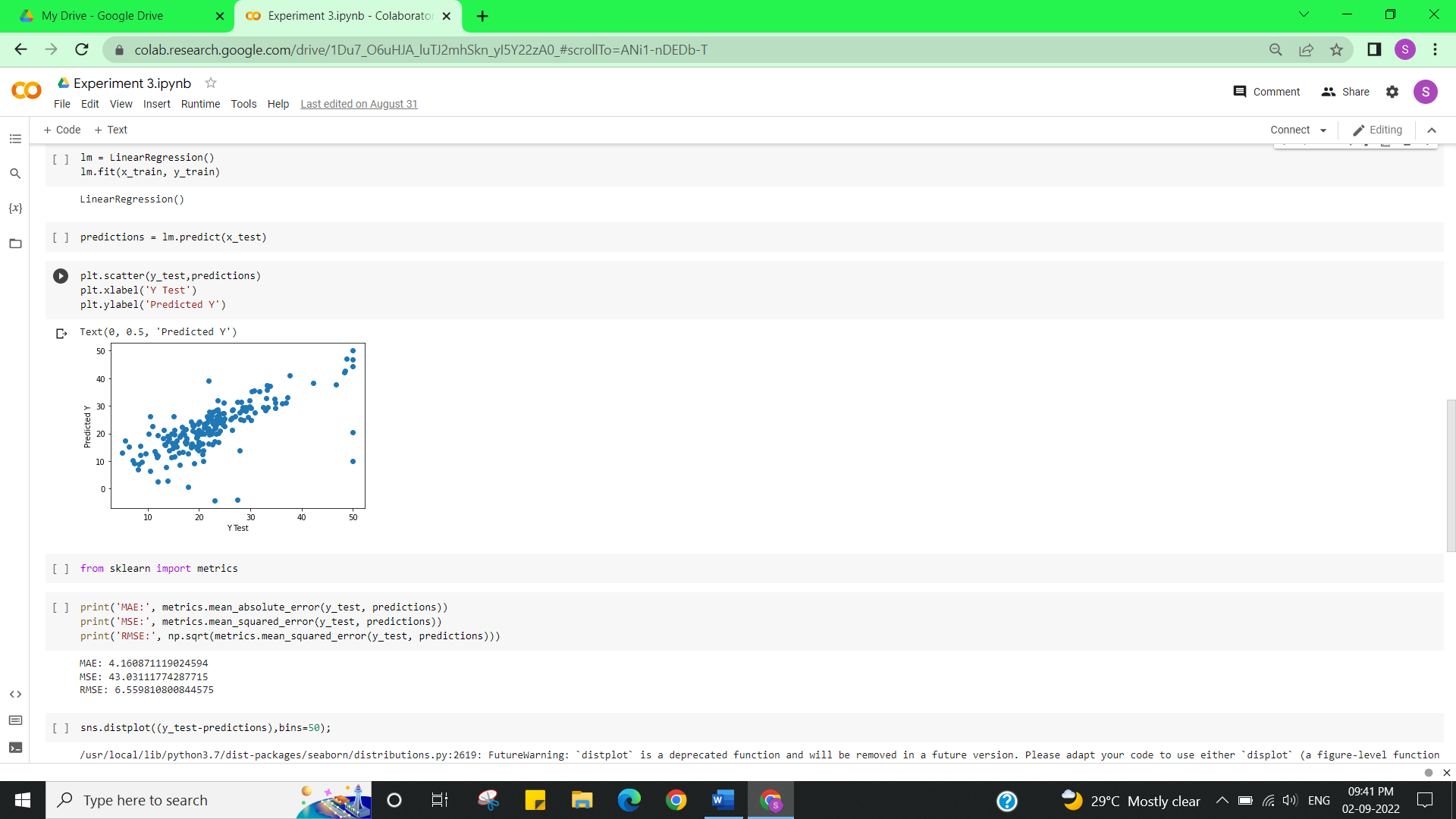
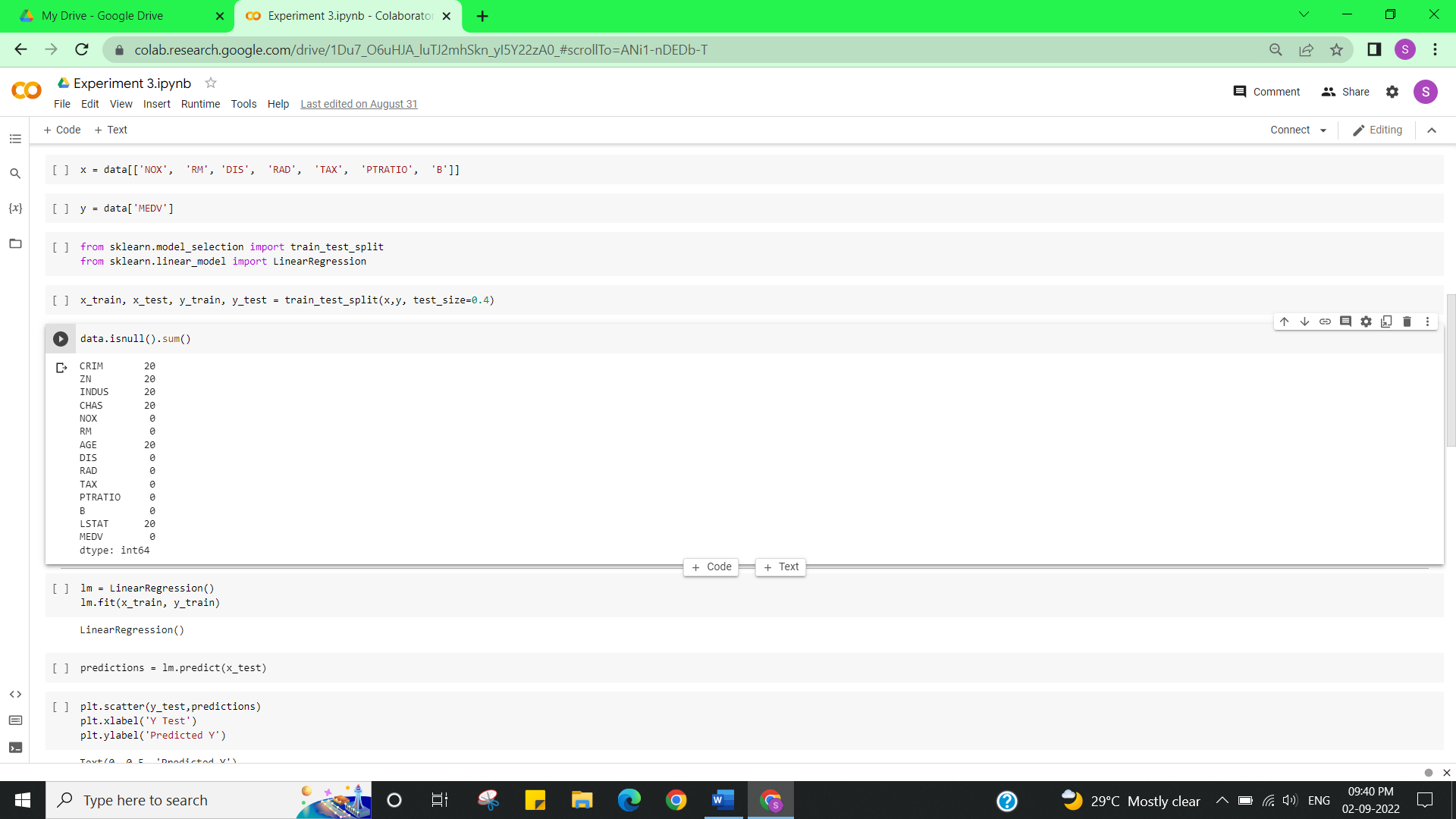
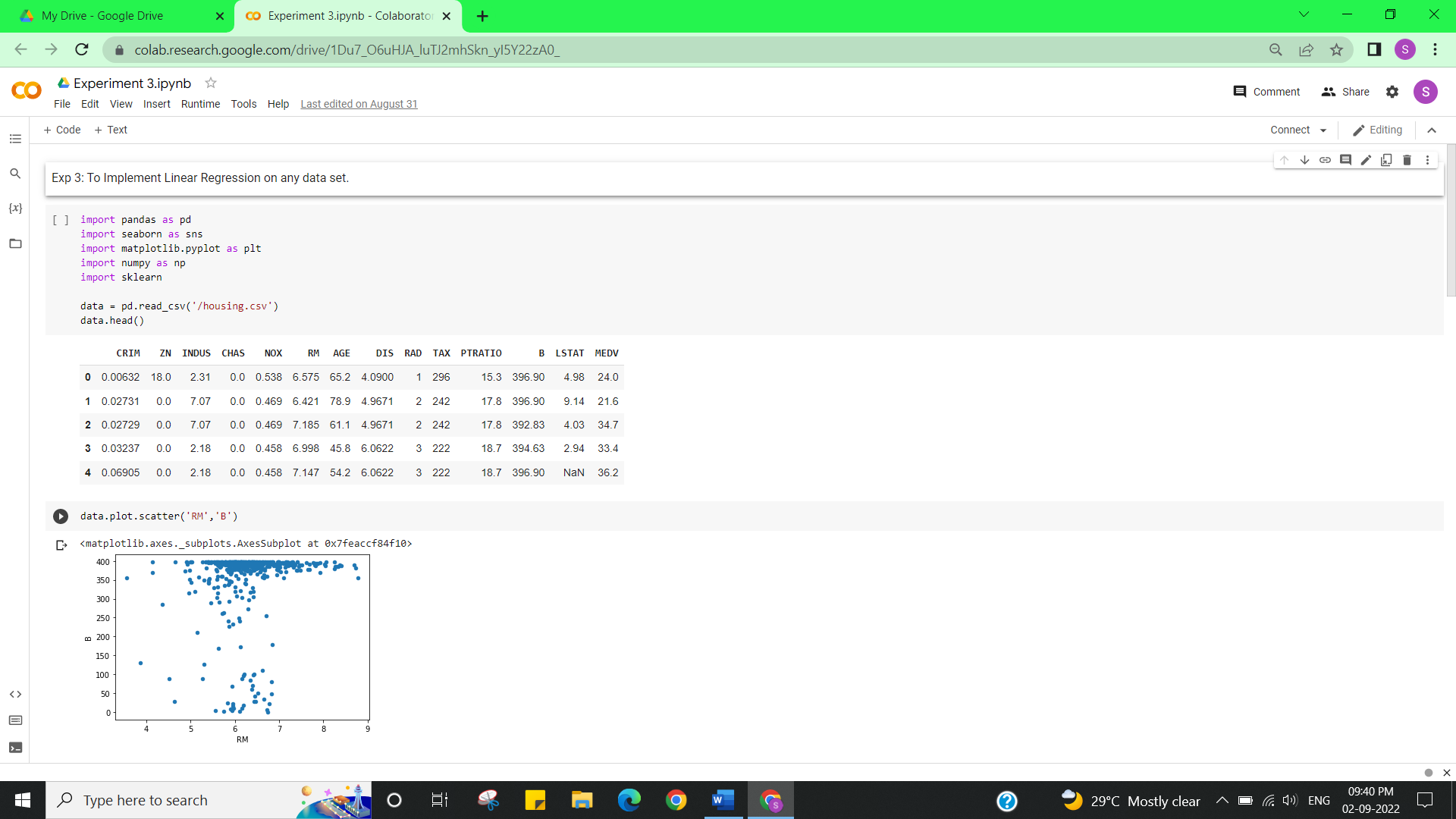
sns.distplot((y\_test-predictions),bins=50);

coefficients = pd.DataFrame(lm.coef\_,x.columns)

coefficients.columns =['coefficients']

coefficients

1. **Result/Output/Writing Summary:**



**Learning outcomes (What I have learnt):**

Implement Linear Regression.

**Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Parameters | Marks Obtained | Maximum Marks |
| 1. |  |  |  |
| 2. |  |  |  |
| 3. |  |  |  |
|  |  |  |  |