**MACHINE LEARNING PROJECT**

***In this project ,I have used Linear Regression Machine Learning model for the simple one input feature.***

**PROBLEM STATEMENT:**

***This is the simple example with one input feature using LinearRegression model and SGDRegressor. Create the array into the reshape array.SGD is very sensitive to varying-sized feature values. so first do feature scaling.***

***And also using the fit transform method to SGDRegressor.***

**ML METHODOLAGY :**

***Linear Regression is the methodology used for shaping and reshaping the array.***

***Linear Regression is a machine learning algorithm based on supervised learning. Linear Regression performs the task to predict a dependent variable(y) based on a given independent variable(x). so this regression technique finds out a linear relationship between x(input) and y(output).***

**DATASET DESCRIPTION:**

***Some relevant columns in the dataset***

* ***sgdr coefficient ,weight and intercept values.***
* ***Predict model.***

***PRE\_PROCESSING:***

***Pre\_processing refers to the transformations applied to our data before feeding it to the algorithm.***

***[1] %matplotlib inline***

***import matplotlib.pyplot as plt***

***import numpy as np***

***from sklearn.linear\_model import LinearRegression***

***from sklearn.linear\_model import SGDRegressor***

***[2] x=np.array([1,2,4,3,5])***

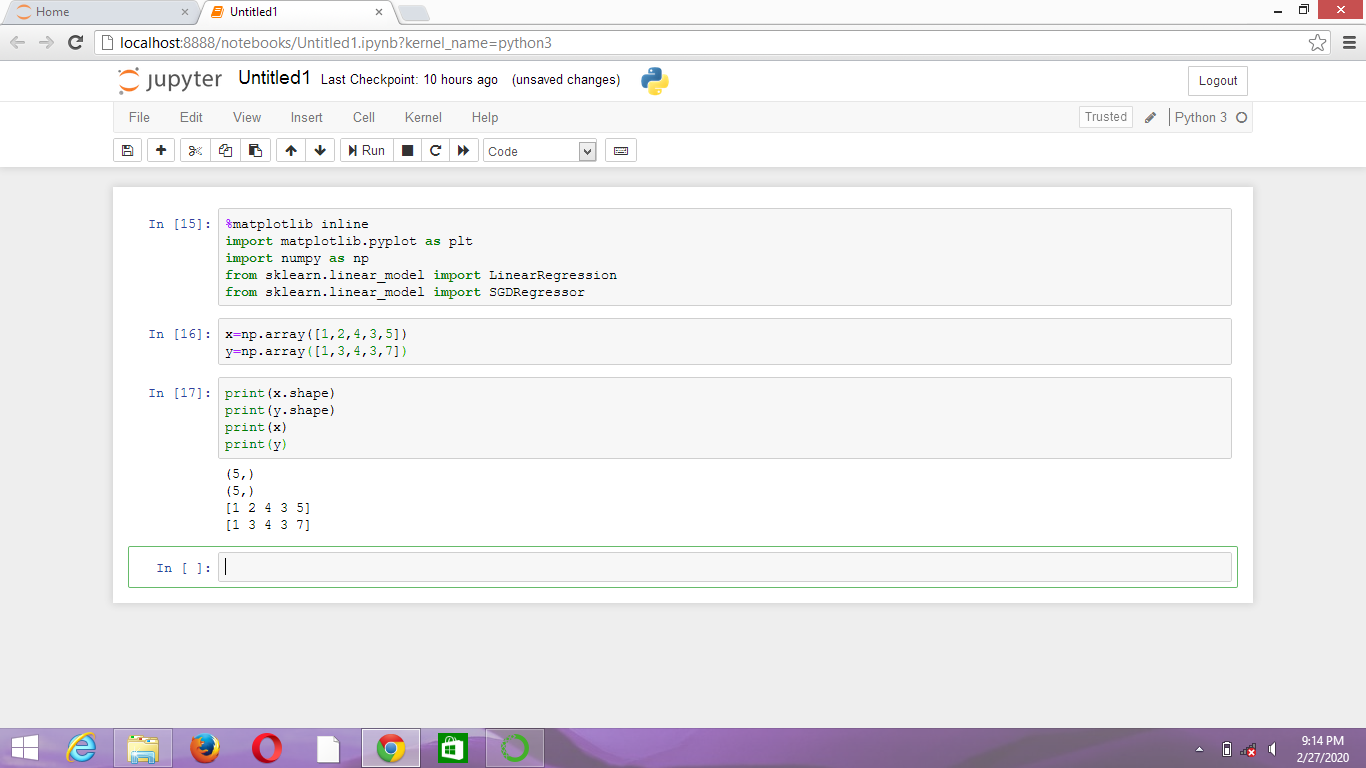
***Y=np.array([1,3,4,3,7])***

***[3] print(x.shape)***

***print(y.shape)***

***print(x)***

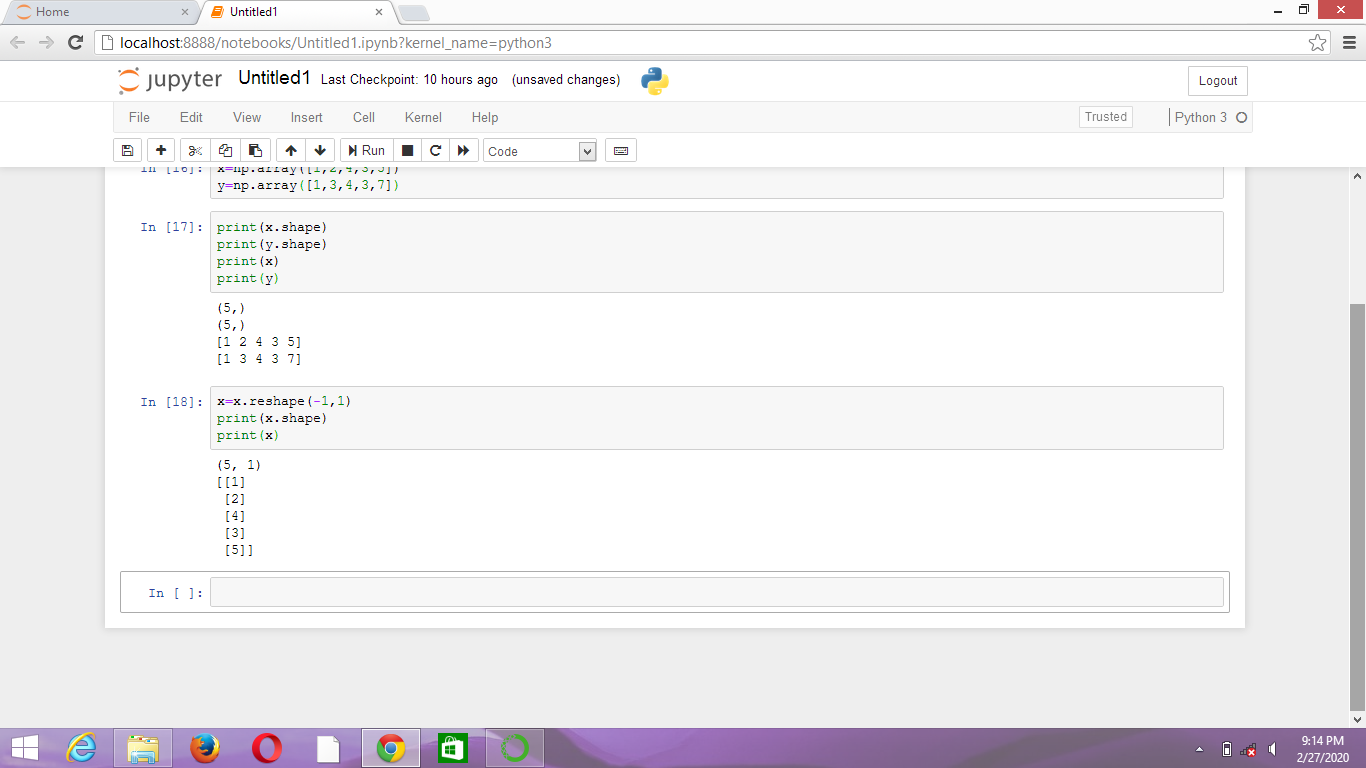
***print(y)***

******

***[4] x=x.reshape(-1,1)***

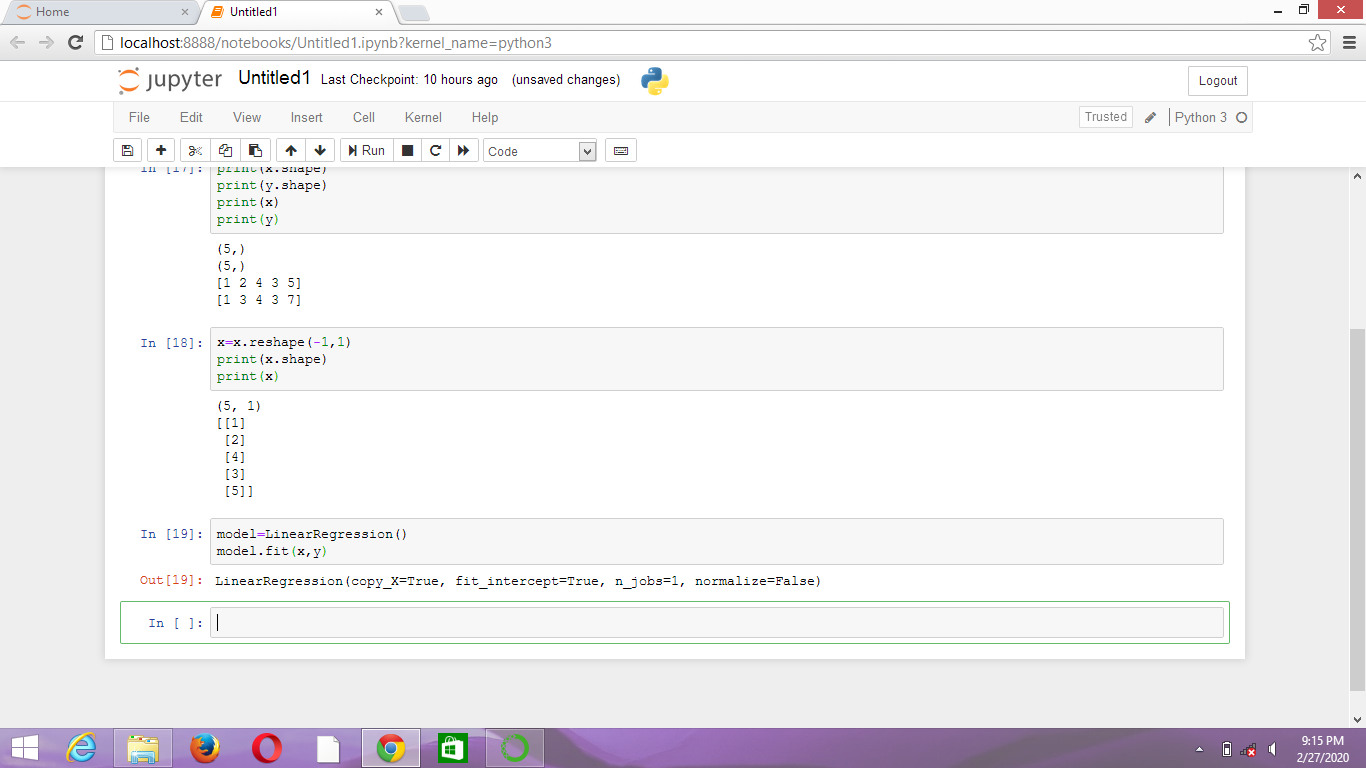
***print(x.shape)***

***print(x)***

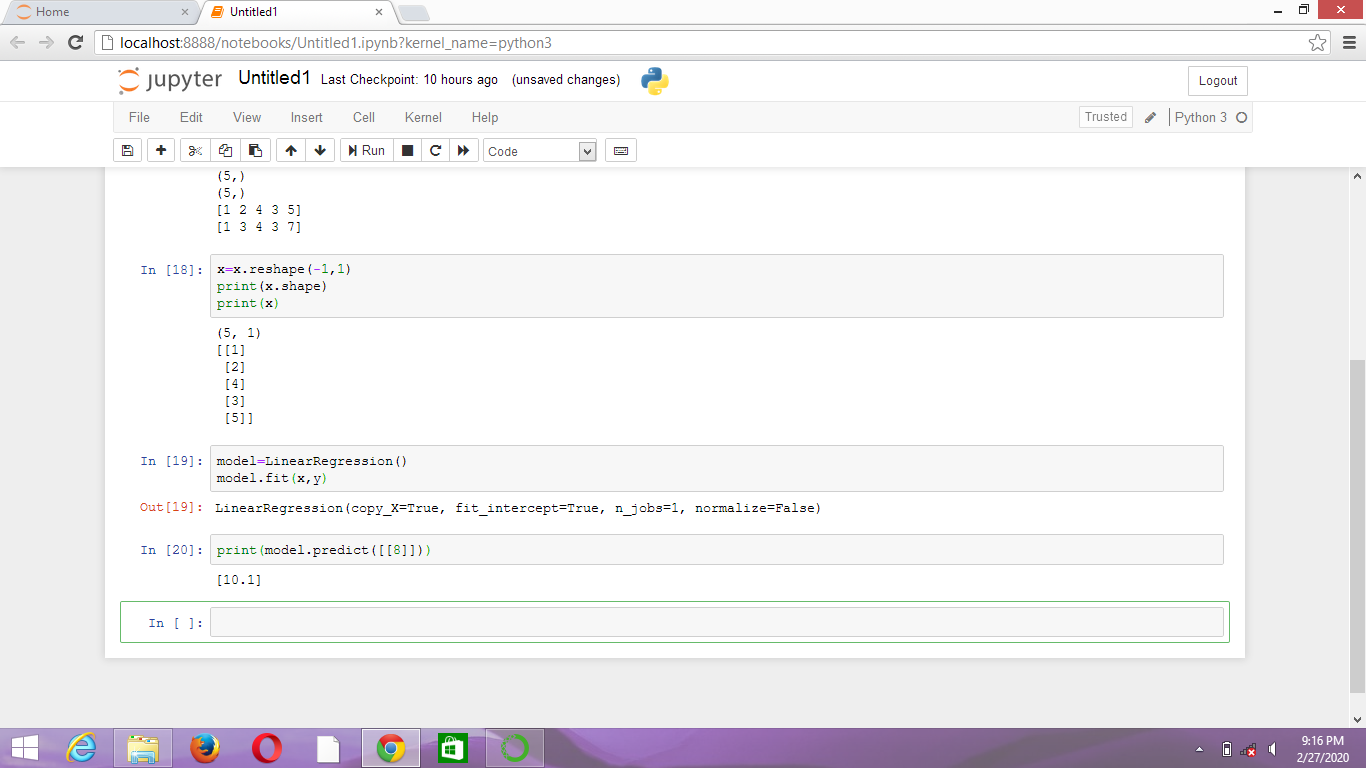
******

***[5] model=LinearRegression()***

***Model.fit(x,y)***

******

***[6] print(model.predict([[8]])***

******

***[7] y\_pred=model.predict(x)***

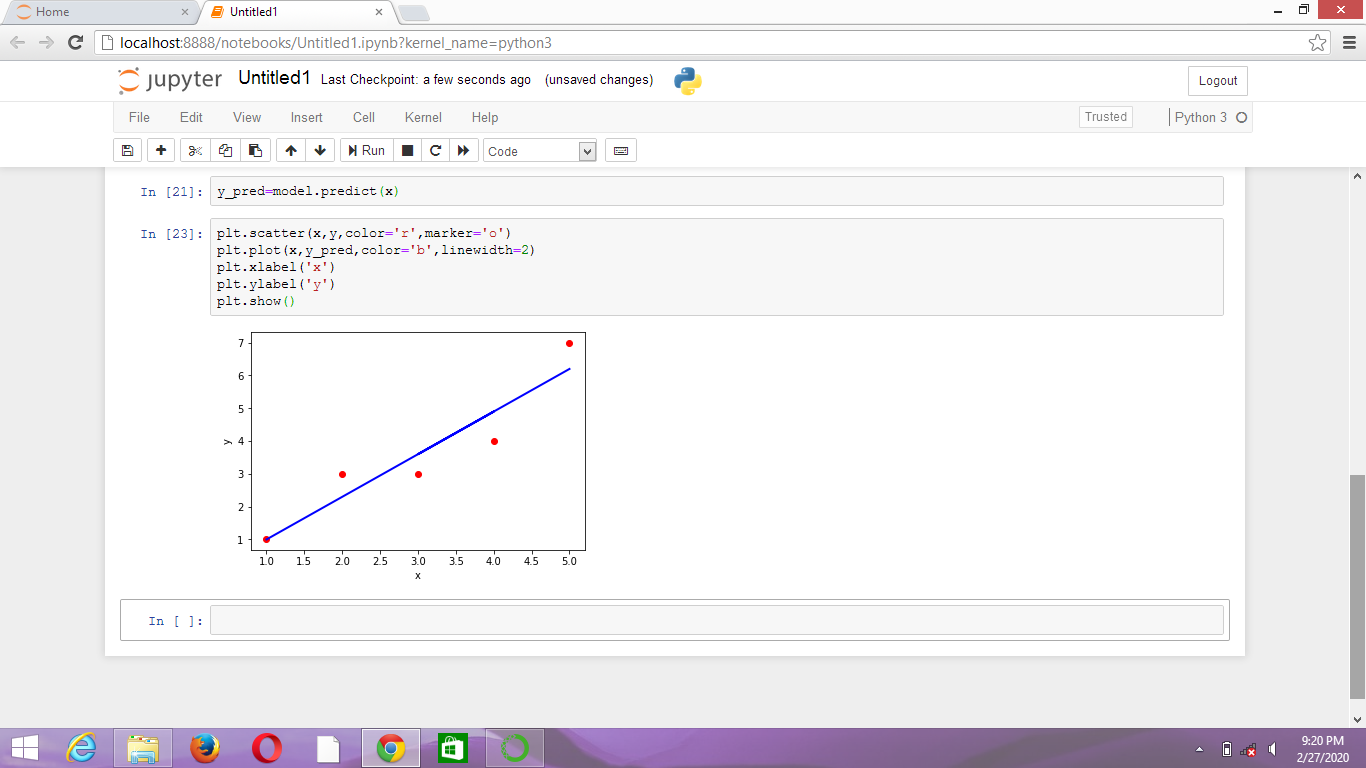
***[8] plt.scatter(x,y,color=’r’,marker=’o’)***

***plt.plot(x,y\_pred,color=’b’,linewidth=2)***

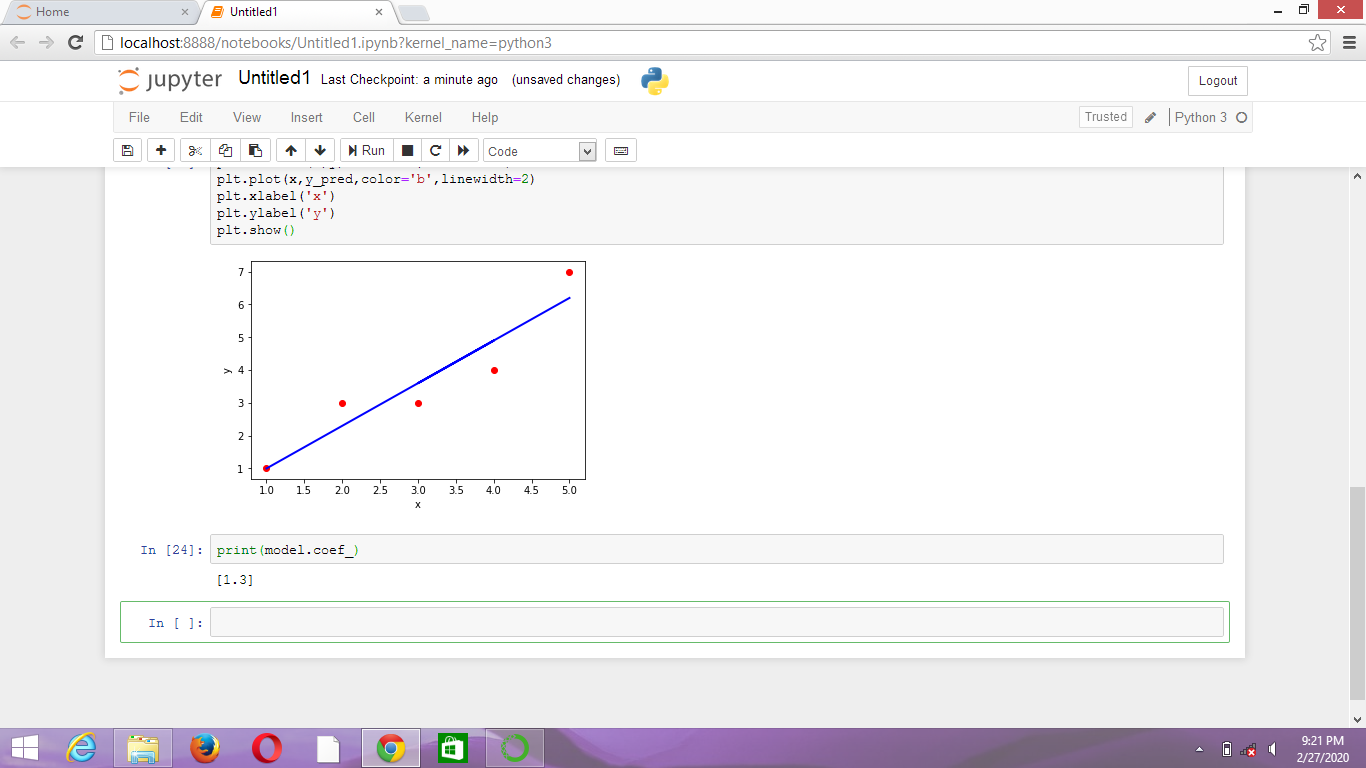
***plt.xlabel(‘x’)***

***plt.ylabel(‘y’)***

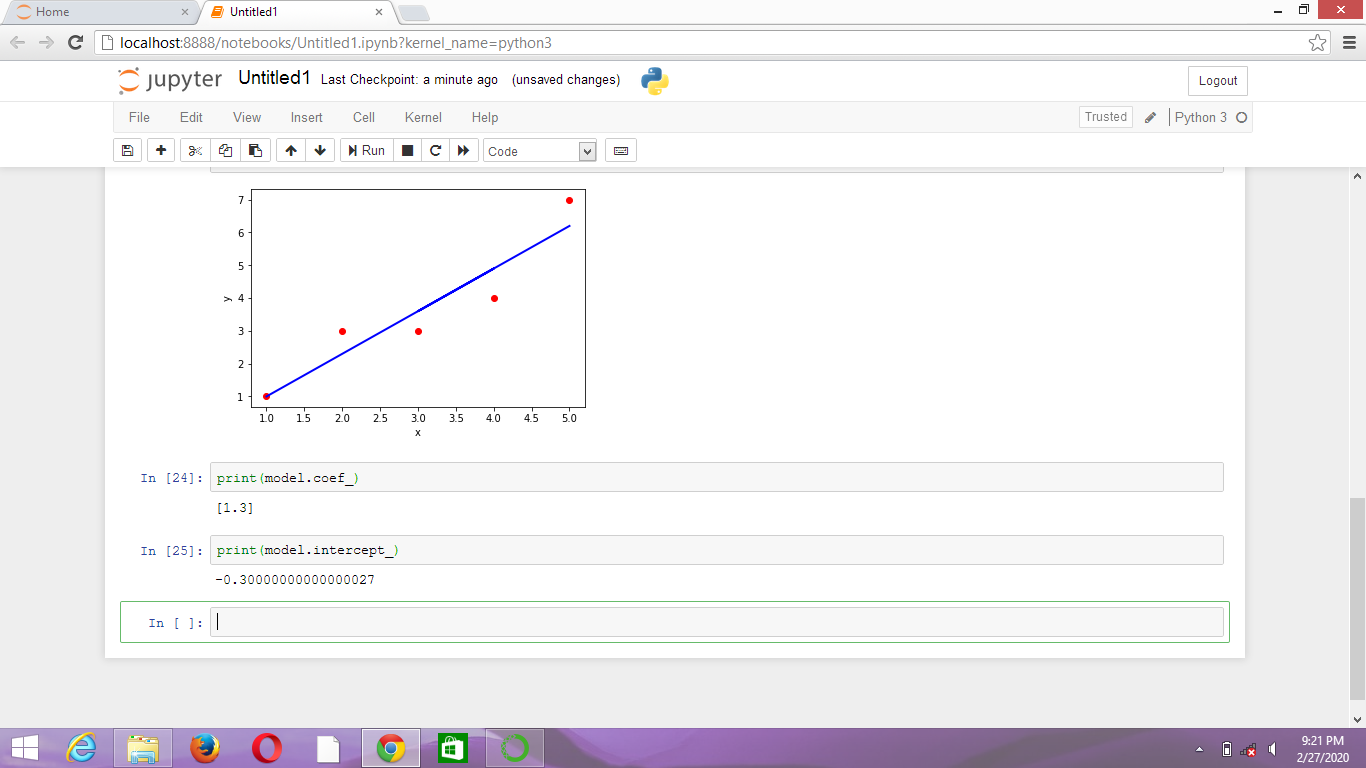
***plt.show()***

******

***[9] print(model.coef\_)***

******

***[10] print(model.intercept\_)***

******

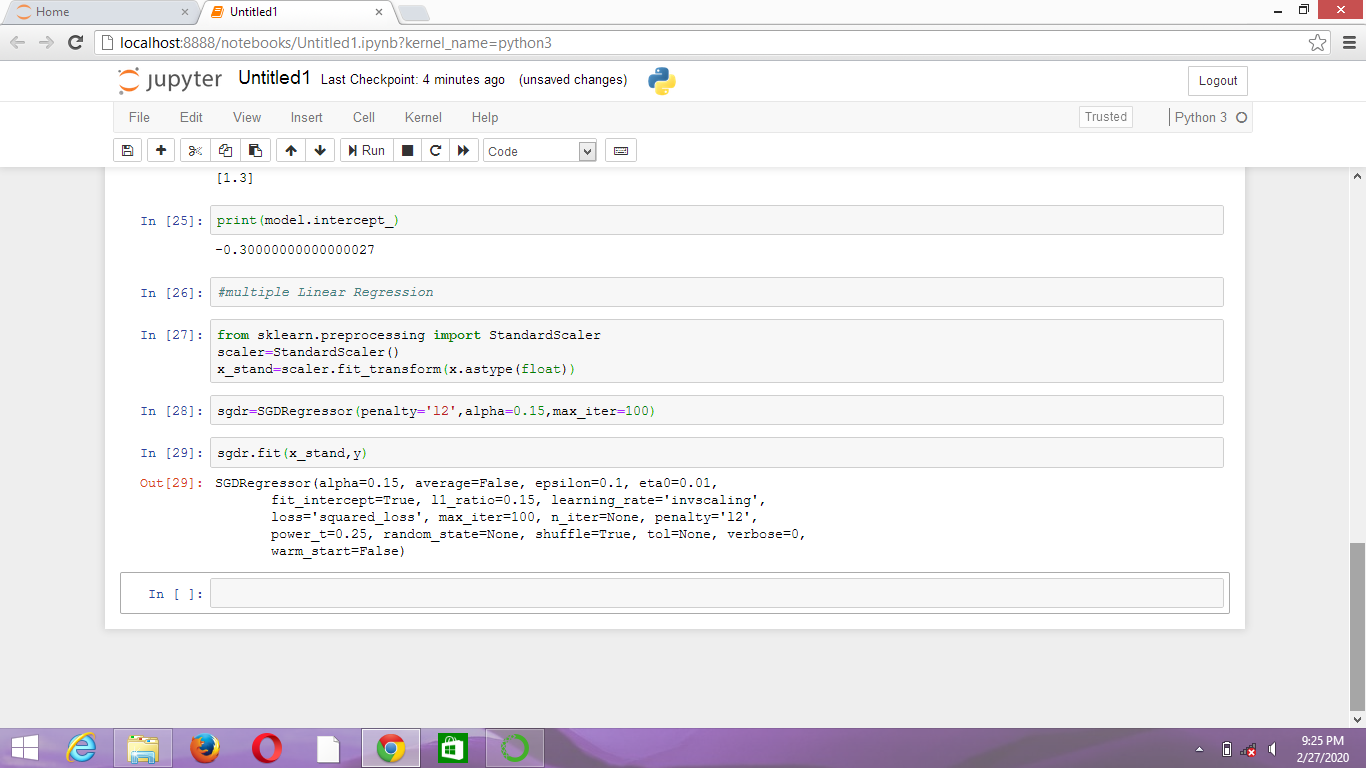
***[11] from sklearn.preprocessing import StandardScaler***

***Scaler=StandardScaler()***

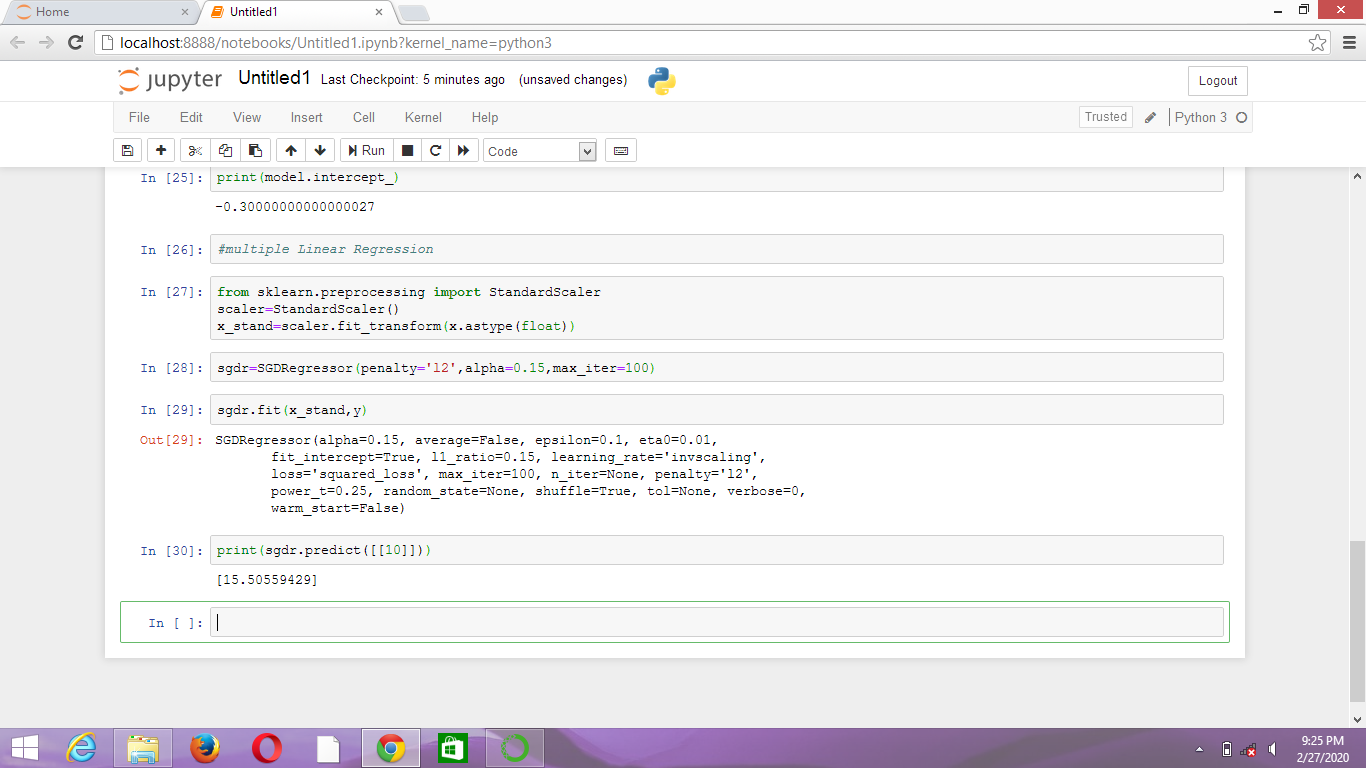
***x\_stand=scaler.fit\_transform(x.astype(float))***

***[12] sgdr=SGDRegressor(penalty=’l2’,alpha=0.15,max\_iter=100)***

***[13] sgdr.fit(x\_stand,y)***

******

***[14] print(sgdr.predict([[10]])***

******

***[15] y\_pred2=sgdr.predict(x\_stand)***

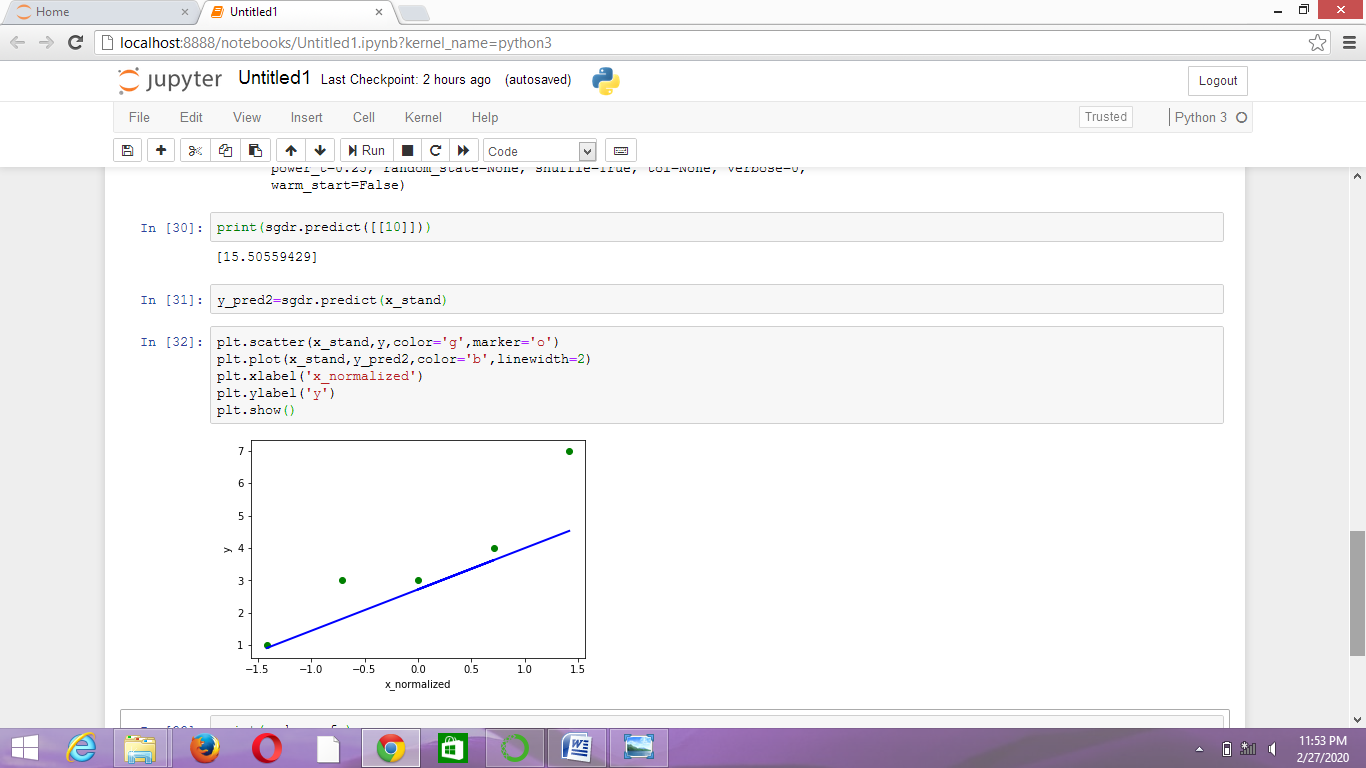
***[16] plt.scatter(x\_stand,y,color=’g’,marker=’o’)***

***plt.plot(x\_stand,y\_pred2,color=’b’,linewidth=2)***

***plt.xlabel(‘x\_normalized’)***

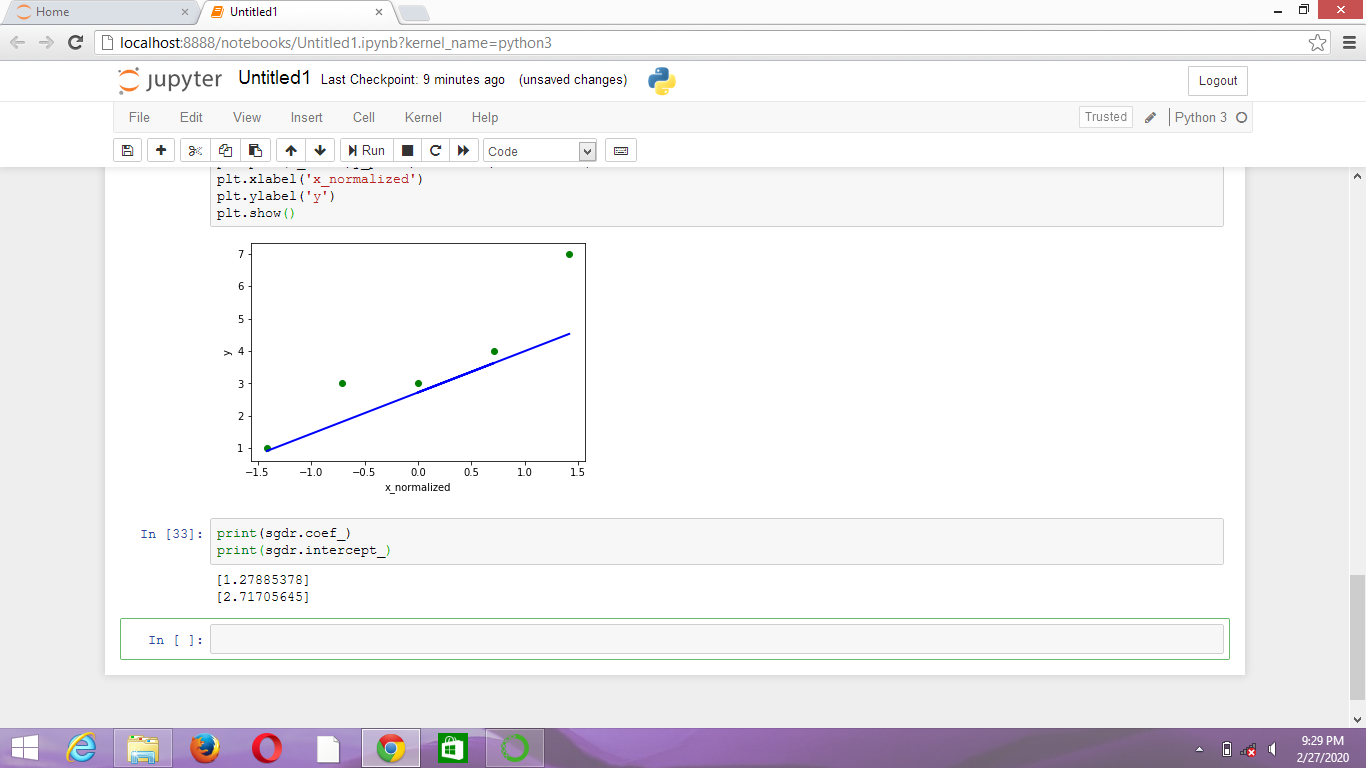
***plt.ylabel(‘y’)***

***plt.show()***

******

***[17] print(sgdr.coef\_)***

***print(sgdr.intercept\_)***

******

**CONCLUSION:**

***Here the output of weight and intercept of the LinearRegression model.***