**CODE REVIEW**

Congratulations! You just learned how a Logistic Regression model works and how to predict the probability for classification problems.

This will be a guided project where we will implement the concepts of Linear Regression.

Problem- We want to predict whether each student will pass their final exam. And the following step to making that prediction is to predict the probability of each student passing

***STEP 1- Import Libraries***

import numpy as np

[Cite your source here.]

***STEP2- Import the dataset***

By default, we are taking the student data with parameters-

1. use hours\_studied as the features
2. calculated\_coefficients as the coefficients
3. intercept as the intercept.

hours\_studied= [[0],[18],[19]]

calculated\_coefficients = [[0.20678491]]

intercept = [-1.76125712]

***STEP3- Define the Log-odds function***

Inside the log\_odds () function, use hours\_studied as the features, calculated\_coefficients as the coefficients and intercept as the intercept.

Here we multiply each of the feature coefficients by their respective feature value to get what is known as the log-odds

def log\_odds (hours\_studied, calculated\_coefficients, intercept):

return np.dot (hours\_studied, calculated\_coefficients) + intercept

***STEP4- Define the sigmoid function***

This enables our Logistic Regression model to output the probability of a sample belonging to the positive class, or in our case, a student passing the final exam.

Now place the log-odds into the sigmoid function to link the output to the range [0,1], giving us a probability

e^(-z) is the exponential function, which can be written in numpy as np.exp(-z)

def sigmoid(z):

denominator = 1 + np.exp(-z)

return 1/denominator

***STEP5- Define the prediction function***

Let’s use all the knowledge till now we’ve gathered to create the functions and performs thresholding and makes class predictions.

In predict\_class(), find the probabilities that the sample data belong to the positive class.It will return 1 for all values within probabilities equal to or above threshold, and 0 for all values below threshold.

def predict\_class(hours\_studied, calculated\_coefficients,intercept,threshold):

calculated\_log\_odds=log\_odds(hours\_studied, calculated\_coefficients,intercept)

probabilities=sigmoid(calculated\_log\_odds)

return np.where(probabilities >= threshold,1,0)

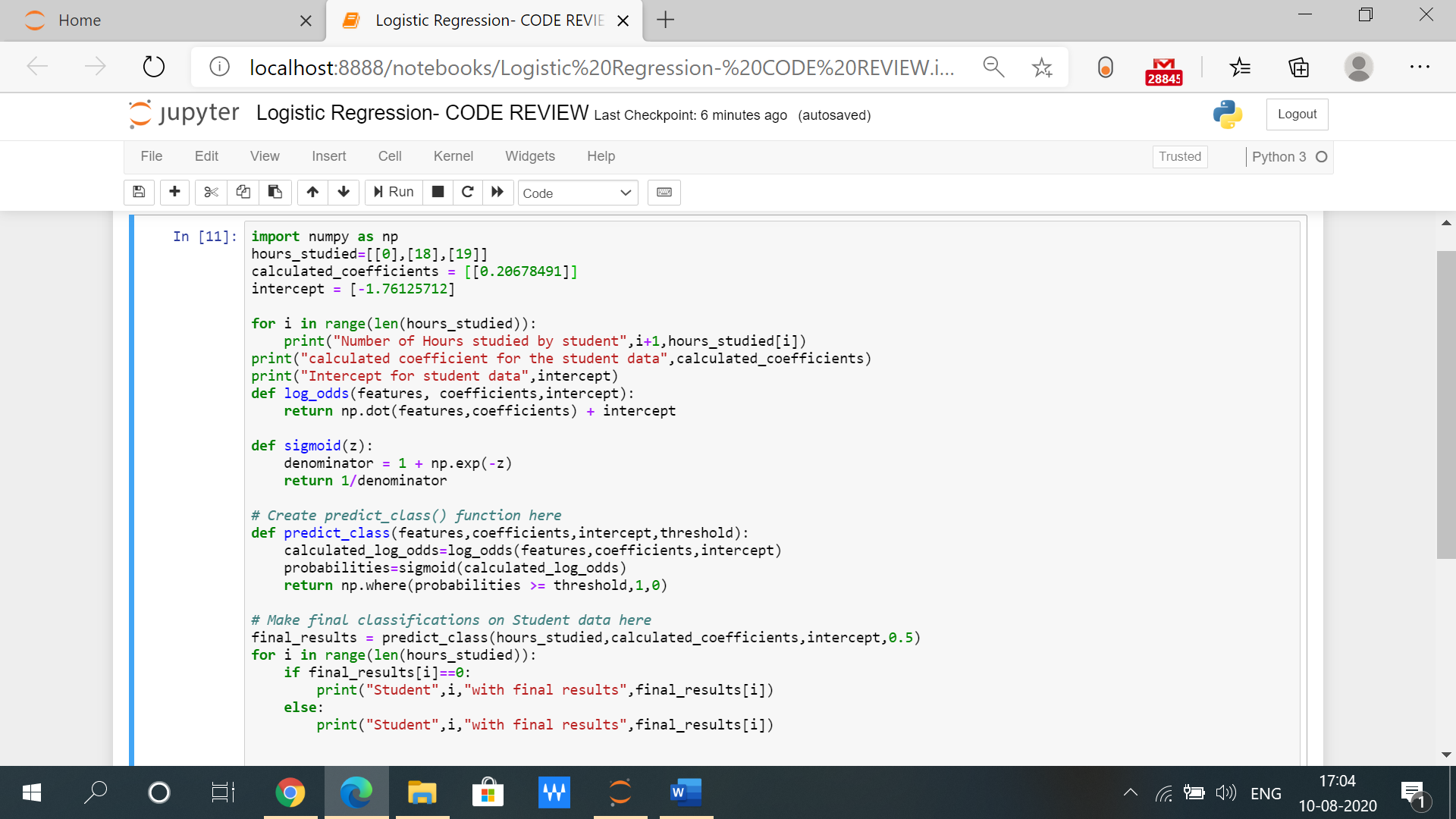
***STEP6- Make final classifications on data***

Let’s make final classifications on Student data to see which students passed the exam. Using the predict\_class() function with hours\_studied, calculated\_coefficients, intercept, and a threshold of 0.5 as parameters. Store the student results in a variable final\_results, and obtained the desired student’s results.

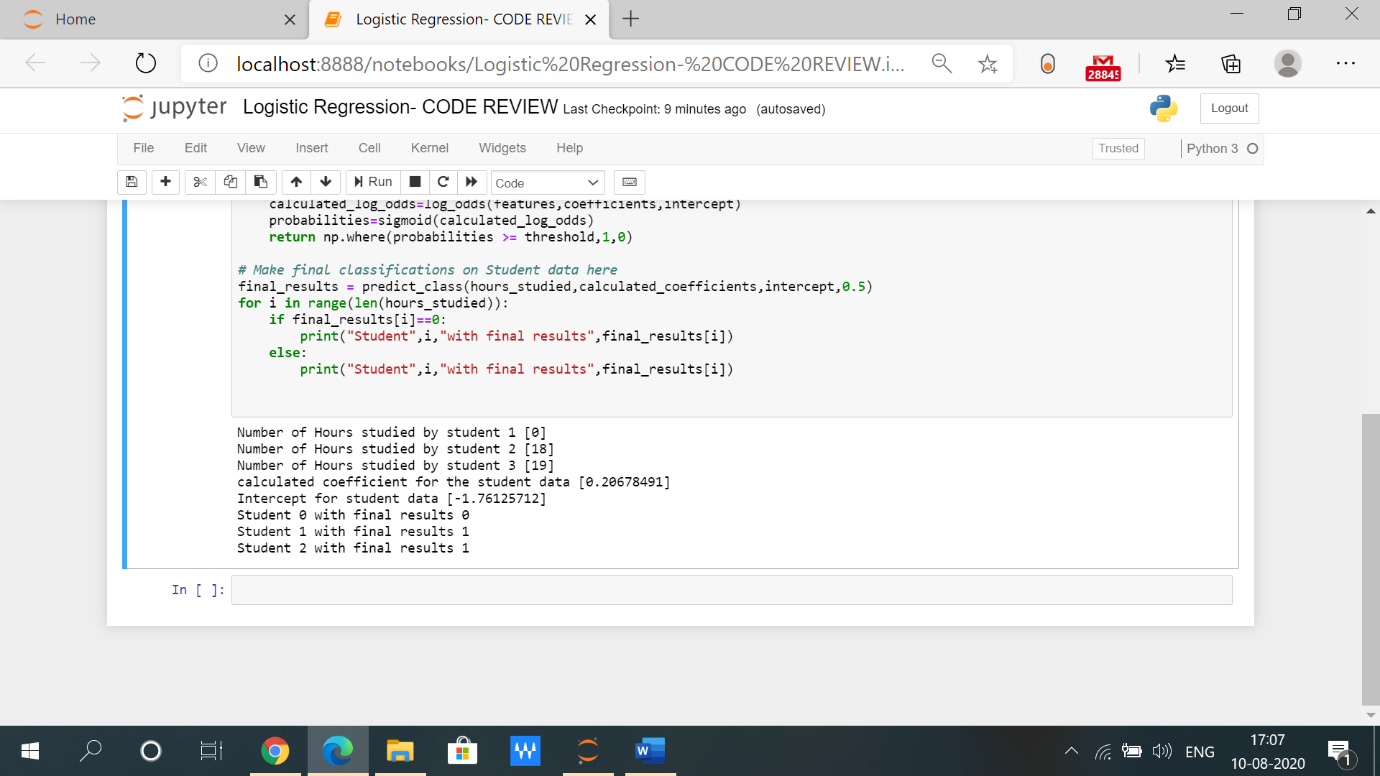
final\_results = predict\_class(hours\_studied,calculated\_coefficients,intercept,0.5)

print(final\_results)

**LOOK AT THE CODE**

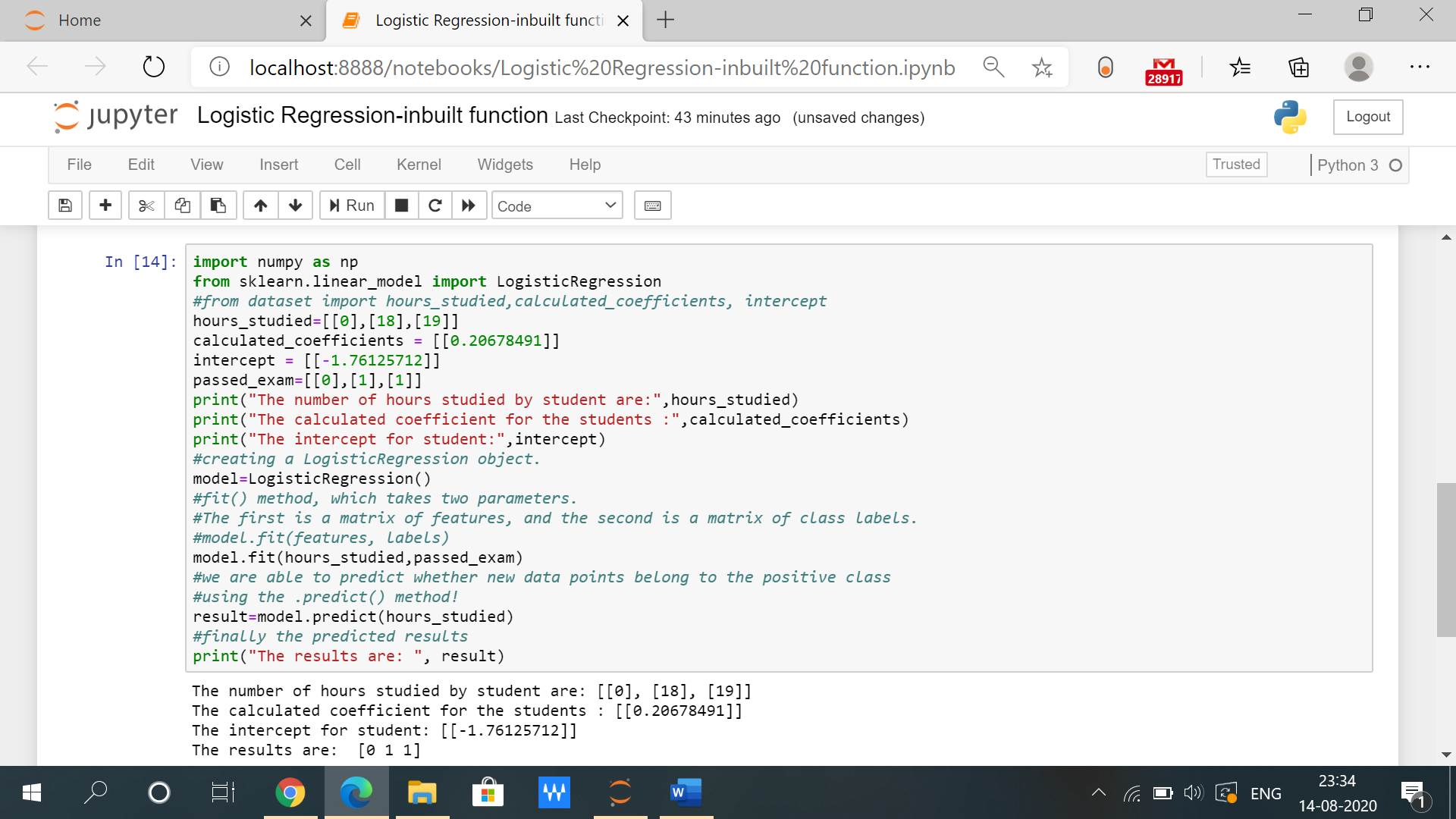


**OUTPUT**

****

**Logistic Regression with inbuilt function**

**Try these inbuilt functions using sklearn library and save your time!!**

****