Write a Python program that implements gradient descent for minimizing  
the least squares loss. As a stopping condition check for the objective (least squares error) between the current and previous iteration. If the objective improves  
by less than theta then you stop. The input and output should be the same  
as for Naive-Bayes. Note that you need to convert label 0 to -1 from training label file since in least squares model you have outputs of -1 or +1.  But you should display your predictions as 0 instead of -1.

Test your program with the data file [testLeastSquares.data](https://njit.instructure.com/courses/4463/files/164369/download?wrap=1) and training label file [testLeastSquares.trainlabels.0](https://njit.instructure.com/courses/4463/files/164376/download?wrap=1).

Use eta=.001 and stopping condition of .001.

Compute vector [w1, w2] and its distance to origin

Your final w for the test data would be close to

w = [0.087137,0.084443]

and distance of plane to origin would be about

abs(w0/||w||) = 6.754035

and your predictions for rows 8, 9 and 10 will be as follows:  
0 8  
1 9  
1 10