HW 9 (Chaoran Lin)

1

The accuracy I obtained for the training set was around 0.93, about the same as the accuracy of 0.92 for the test set. My hypothesis is that the training set data had a high enough generality that represented other possible test sets well, resulting in very little difference when the learning model was actually put to test against test sets. The difference itself can be explained by the fact that the model trains on training data and thus obtains weights and bias originally suited for the training data, so there is bound to be some difference when test data is used.

2

After decreasing the number of steps allowed for training from 1000 to 10, I got an accuracy of about 0.76. In the case of 10000, I got an accuracy of 0.92. The relatively lower accuracy for 10 times compared to 1000 is evident of a neural network's ability to adjust its weights and bias to improve upon receiving more training. The small increase from 1000 to 10000, however, suggests that the room for significant increase in accuracy becomes smaller around this threshold, and that the learning model will approach and stay around an optimum after it.

3

After replacing the starting values for the optimization algorithm from 0's to 1's, I got around the same accuracy of 0.92, which is not significantly different. The reason is that by increasing the initial values, the optimization algorithm can potentially take less steps towards new values for W and b that perform better in recognizing digits, but the difference is very small considering that the change still allows the optimization algorithm to train many times, enough to reach an optimal result.