Connor Twohey
cpt7282

Assignment 3

1. Included in tensor.py. Lines 2-19 are the code included from the TensorFlow tutorial, and have been commented accordingly. Comments/code on any other line are my own.
2. The weights for the layers in the model can be seen using the get_weights function. I added code at the end of the original tutorial code to create 10 plots (one for each hidden node/feature) from the weights generated between the first input layer (the flattening layer) and the first hidden layer (the first dense layer). Each plot was a 28 x 28 2D array of weight values. These values did not stay the same from run-to-run, so I have included results from only one run. Since there are ten 28 x 28 plots, I have included them in a separate file for viewing (plots.txt).
3. Number of hidden nodes is changed at line 10.

i. N = 1
Accuracy = 0.2885
ii. N = 10
Accuracy = 0.9265
iii. N = 50
Accuracy = 0.9691
iv. N = 100
Accuracy = 0.9767

Unsurprisingly, the accuracy shoots up at a great rate as soon as more than a single hidden node is used. However, the increases in accuracy get smaller and smaller as the number of hidden nodes continues to increase. It can be assumed that, using a large number of hidden nodes, the accuracy of the given model can be brought to nearly 100%. With a large enough number of nodes, the model has a statistically negligible/insignificant amount of inaccuracy. However, larger numbers cause the program to slow down considerably, so a good balance must be struck between desired accuracy and program speed/efficiency.