Assignment 4: Problem 1

Please refer to the Tensor Flow program "Tensor Flow Examples 03b (MNIST Linear Model).ipynb" that is on the list of Camino Week 4 modules. This program implements the Logistic Regression algorithm, using the MNIST data set. Become familiar with the program by executing all its sub-modules, and then modify it in the following ways:

1. Add functionality to the program to be able to compute the Loss Function value based on the training data, at the end of each training epoch. This value should be averaged over all the samples in an epoch.
2. Add Tensor Board functionality to the program to be able to plot the following values, as a function of number of epochs:
   - Training Accuracy (over 1 batch of training data)
   - Test Accuracy (over the entire test data set)
   - Loss Function (as computed in Part 1).
3. Investigate the effect of the following parameters on the performance of the algorithm. The performance can be tracked by using one or more of the plots from Part 2:
   - The Learning Parameter eta.
     Hint: It is more productive to search for eta using the Logarithmic Scale. For example search for values of eta in the range 10^[Uniform(-5, 0)].
   - The Batch Size B.

(5 Points)

Assignment 4: Problem 2

Please refer to the Tensor Flow program “Tensor Flow Examples 04c (MNIST Fully Connected Model using dense module).ipynb” that is on the list of Camino Week 4 modules. This program implements a Deep Feed Forward Neural Network, using the MNIST data set. Become familiar with the program by executing all its sub-modules, and then modify it in the following ways :

1. Migrate the code that you developed as part of Problem 1, to be able to compute the Loss Function and also to plot the Accuracy and Loss Function values.
2. Using suitable values of the Learning Parameter and Batch Size (perhaps based on the results of Problem 1), investigate the effect of the following parameters on the performance of the algorithm. The performance can be tracked by using one or more of the plots from Part 1:
   - For the case of a single Hidden Layer, vary the number of Hidden Layer nodes.
   - Fix the number of nodes per Hidden Layer, and then vary the number of Hidden Layers.

(5 Points)