YOUR ID HERE! **Homework Report 1** CS1420 – Spring 2019 February 17, 2022

Report Questions

1-a. When batch size = 1, accuracy = 92.9 and epochs = 31. When batch size = 5, accuracy = 85.8 and epochs = 32. When batch size = 10, accuracy = 82.6 and epochs = 33. When batch size = 75, accuracy = 70.5 and epochs = 27. When the batch size is 1, the program conducts gradient descent and updates weight matrix after calculating loss for one training example. Since out epochs are pretty similar in our observation, we can conclude that the number of gradient descent was the largest when batch size = 1 and accuracy was the highest of our observation. On the other hand, when batch size = 75, the number of gradient descent was smallest and the accuracy was also the lowest. Therefore, in our implementation, we observed that quicker converge (assuming smaller number of gradient descent = quicker convergence) generally suggests lower accuracy.

1-b. The reason why setting a smaller batch size resulted in slower convergence could be because each training example had more impact on weight updates and therefore estimated a noisier loss after each iteration, which ultimately lead the program to take more time to converge. However, since the weights are updated on smaller batches and therefore they reflect each training example more than with larger batches, it also produced higher accuracy.

2. It seems that all 'workclass', 'marital-status', 'occupation', 'relationship', 'race', 'native-country' are the categories that are one-hot eoncoded in the program. I am guessing they did this so that when they learn ERM model, the weights can be updated by calculating partial derivative of the loss function with respect to each weight. This operation is extremely difficult if categories were represented as an enumeration of possible values for the feature as they are not numerically represented.

3. Normalization is implemented to keep the scale of each attribute the same and avoid distorting differences in the ranges of values. Looking at the result run on the unnormalized data, I observed that the accuracy is lower than when I run the model on the normalized data. This is supposedly happened because not normalizing features resulted in over/under weighing particular features in the dataset. In our dataset, different features had different ranges, the effect of normalizing was evident.

4. I experimented running my model on two different files, one which includes sensitive data and the other without those data using different batch size and different learning rate. Then, I found out that the inclusion of sensitive data did not necessarily improve the accuracy of our prediction model. Specifically, depending on batch size and learning rate, our model sometimes performed better on file over the other file and other times performed better in opposite ways, and the difference in accuracy was always slim. Therefore, I conclude that there is no correlation between sex/race and education level.