Project 2 Part 1

Jason Van Tuinen, Bharadwaj Satyanarayana, Anthony Sainez, Karthik Gnanakumar

1 Reference Network

By varying batch size, learning rate, gamma, step size, and momentum, we were able to find a set of parameters that produced a model with 0.97230% validation accuracy. The model used a batch size of 256, learning rate of 0.2, gamma of 0.9, step size of 2, and momentum of 0.9. On the test set, the model achieved 1.33% accuracy. Figures 1 and 2 show the train and validation accuracy per SGD step, respectively. Figure 3 shows the confusion matrix of the model on the test set.

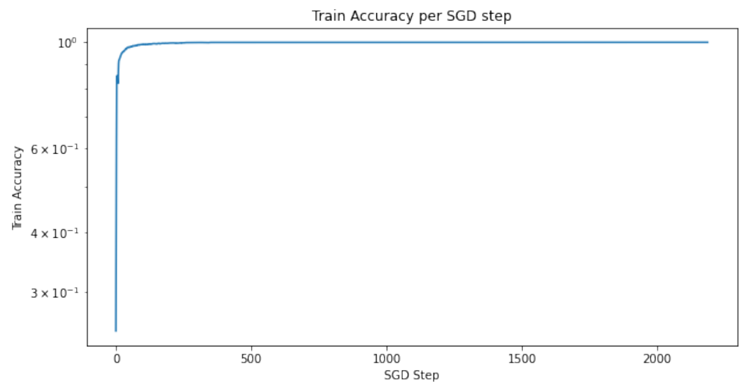

Figure 1: Train accuracy per SGD step for a model with batch size: 256, learning rate: 0.2, gamma: 0.9, step size: 2, and momentum: 0.9

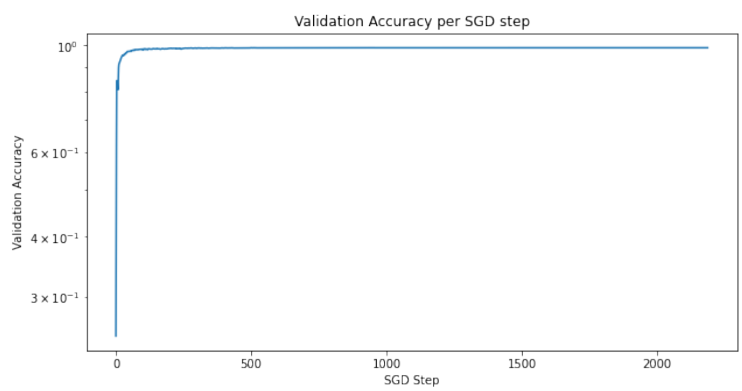

Figure 2: Validation accuracy per SGD step for a model with batch size: 256, learning rate: 0.2, gamma: 0.9, step size: 2, and momentum: 0.9

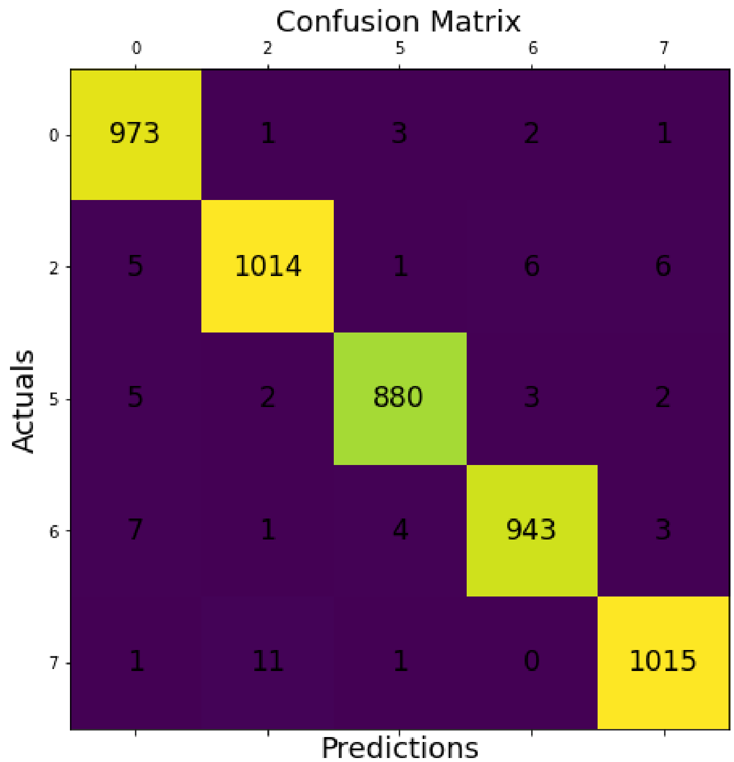

Figure 3: Confusion Matrix

2 Compression

We started by seeing if the parameters for the l-step worked as well on our model as they did for the demo model. They did, so we decided to keep the l-step parameters as they were and focus on the μ schedule parameters. The best results that we got for the μ schedule were with the same starting μ and rate of increase given in the demo code (9e-5 and 1.1, respectively) and 40 iterations (although we tried all values from 9e-2 to 9e-8 and 1.1 to 1.01 respectively). The best results gave a test error of 1.19% for pruning, 1.43% for quantization, and 1.37% for low-rank. The test errors for pruning and quantization are slightly lower than our test error for the reference network, and only slightly higher for low-rank compression.

The compression ratio and number of compressed parameters/codebook size vs train and test error plots are below, in Figures 4-9. We made sure to try all the parameters & combinations to get the best results, and stopped where we thought our test errors were getting worse.

2.1 Compression procedure & Analysis

We used extensive manual Gradient Decent to find the lowest test error after gathering our data, we were able to make some inferences from the data we collected and also found the best suitable parameters.

Through the data we collected, we found that Quantization seemed most dramatically affected by changes to the number of iterations, and we even noticed that for some runs the error for quantization would go up and down in later iterations. Pruning and low-rank generally decreased as we increased the number of iterations, but quantization oscillated more, so this led us to continue decreasing the error for pruning and low-rank and simultaneously find a good spot for quantization by increasing the number of iterations.

2.2 Best Result Plot of Compressed Network

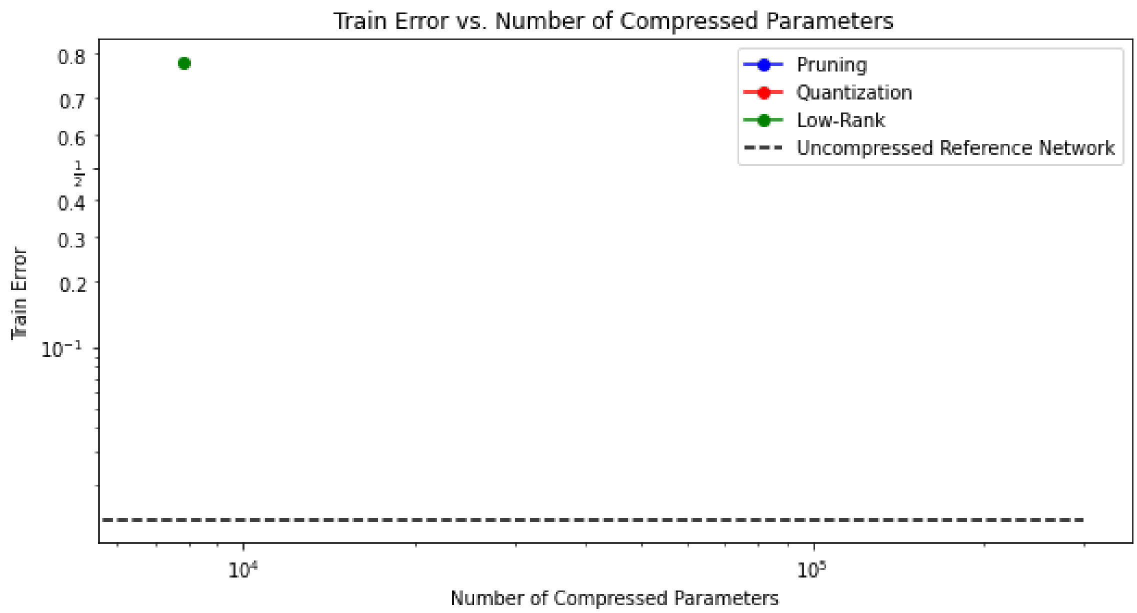

Figure 4: μ schedule: 9e-5 * (1.1** step) for 40 steps. Train error read 0.00% for all compression results besides the highest compression ratio that we applied with Low-Rank compression.

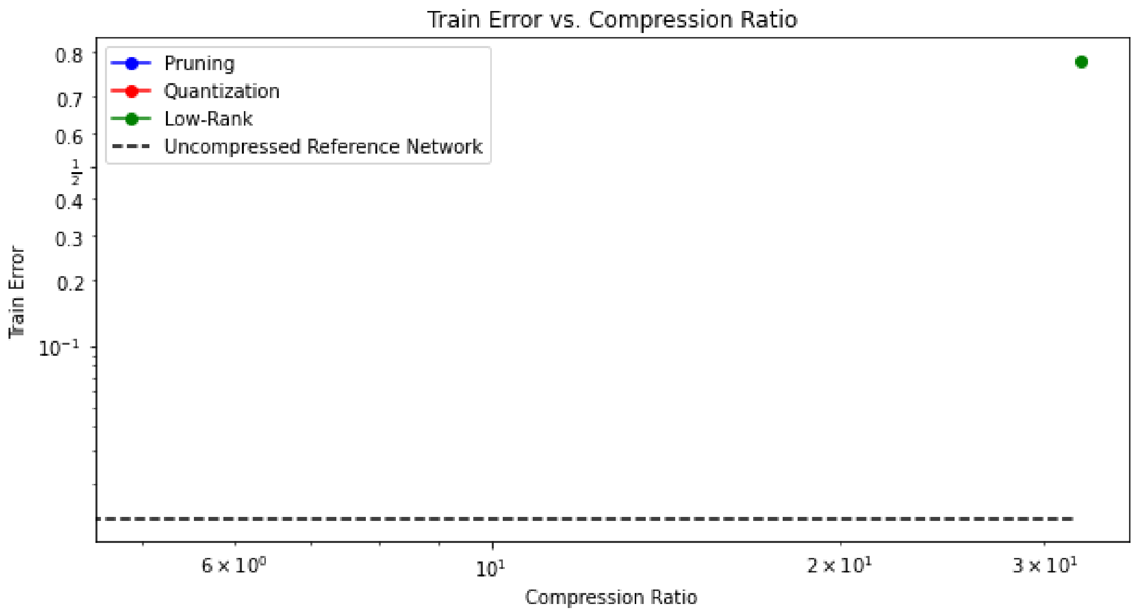

Figure 5: μ schedule: 9e-5 * (1.1** step) for 40 steps. Train error read 0.00% for all compression results besides the highest compression ratio that we applied with Low-Rank compression.

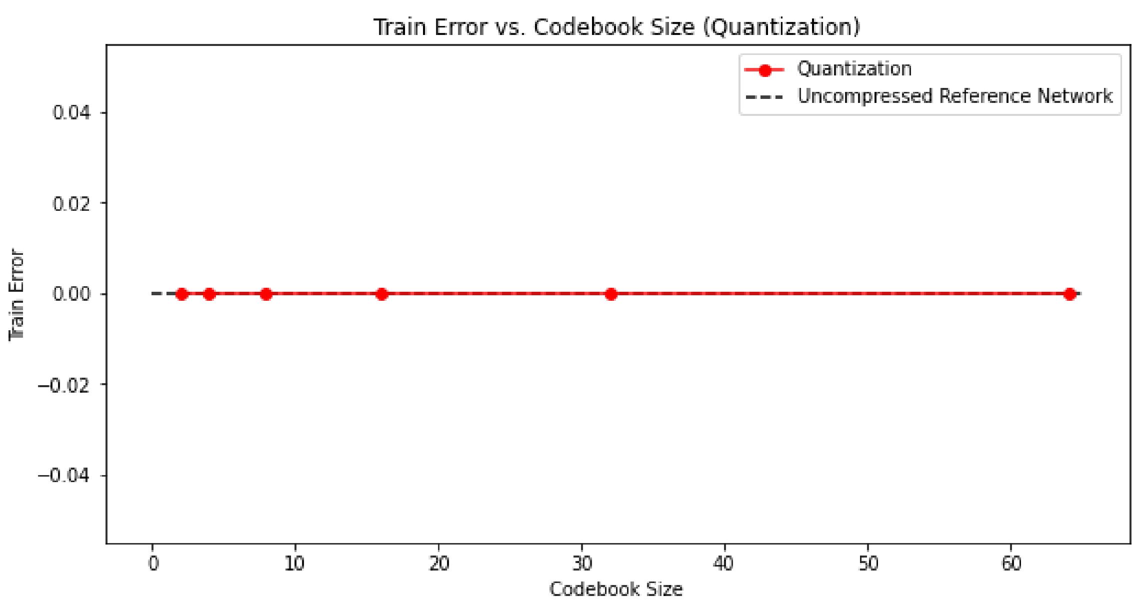

Figure 6: μ schedule: 9e-5 * (1.1** step) for 40 steps. Train error read 0.00% for all compression results.

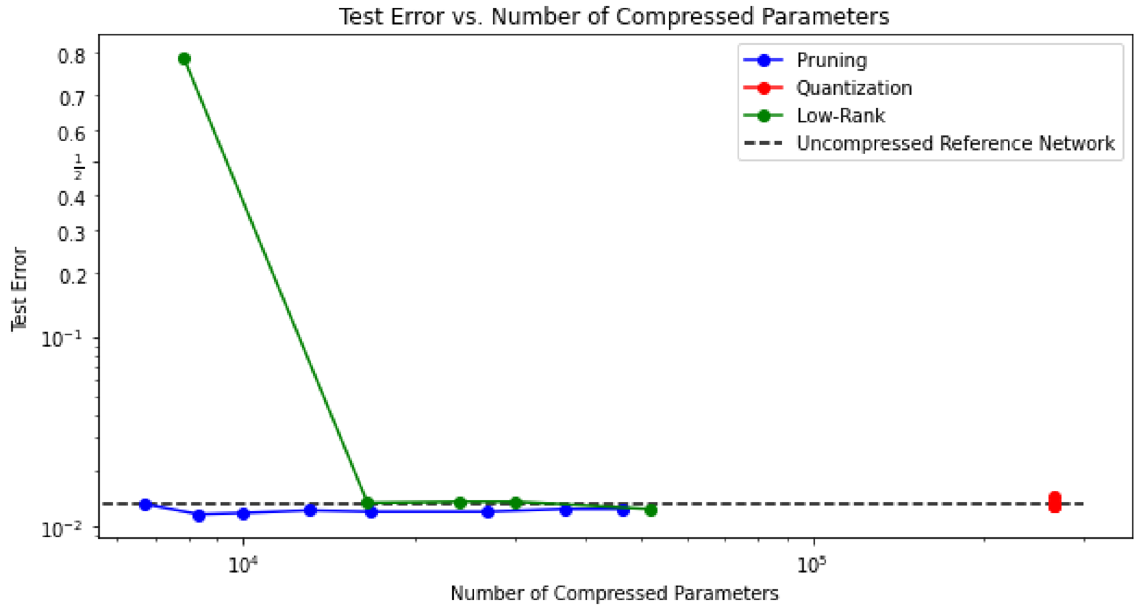

Figure 7: μ schedule: 9e-5 * (1.1** step) for 40 steps. Quantization reported very similar numbers of parameters for each run.

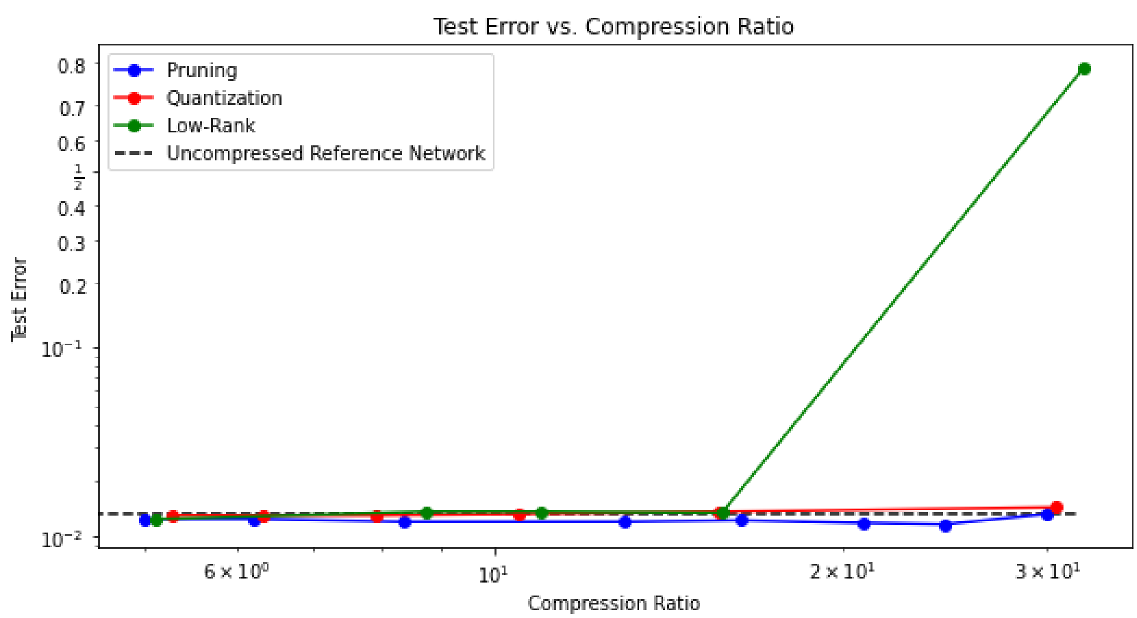

Figure 8: μ schedule: 9e-5 * (1.1** step) for 40 steps

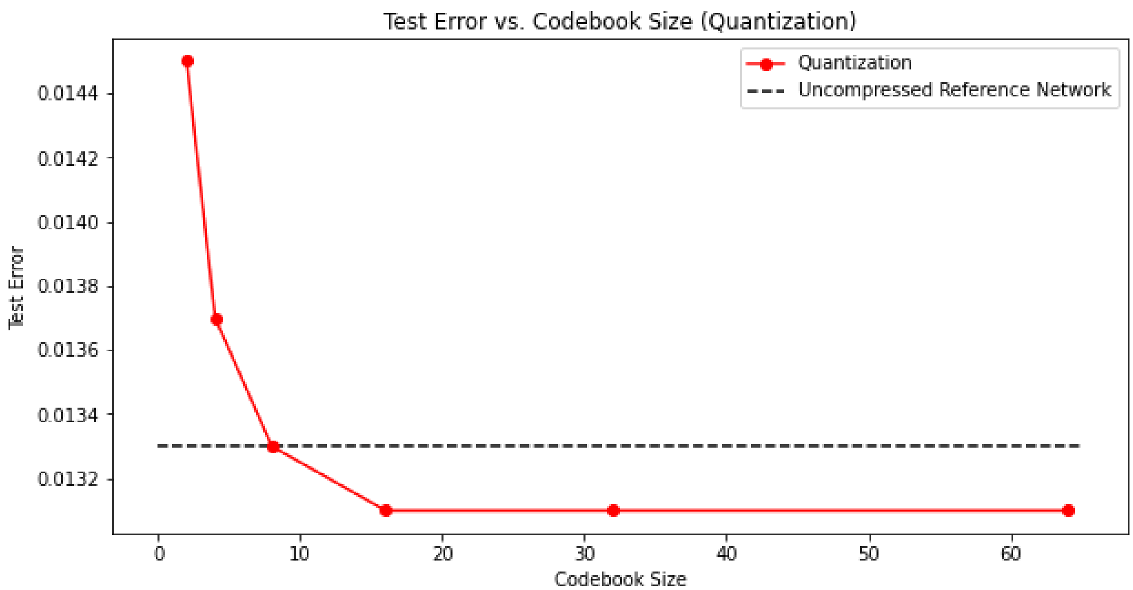

Figure 9: μ schedule: 9e-5 * (1.1** step) for 40 steps