Jonathan Gregory

CS 461 Introduction to Artificial Intelligence

Brian Hare

8 December 2017

Neural Networks Project 4 Report

Upon completing the first tutorial, I was able to test the Gradient Descent optimizer algorithm by running it many times using mnist\_softmax.py in TensorFlow. The initial stochastic model resulted in an accuracy of 91.8% which was close to the expected accuracy of about 92%. I was able to make a modification by changing the optimizer to several other options. First, I tested Adadelta optimizer which resulted in an average accuracy of 91.6%. The Adam optimizer had an average accuracy of 88.8%. The RMS Prop Optimizer averaged an accuracy of 87.2% Finally, the Sync Replicas Optimizer had an accuracy averaging 91.5%. Overall, the entire set of different optimizers were all with in a few percent of each other with the Gradient resulting in the highest accuracy and the RMS Pro organizer scoring the lowest accuracy.

Concluding the second tutorial regarding the Deep MNIST, I was able to experiment with the Adam Optimizer again. I found that using the standard ReLU neurons in the Adam optimizer, it was able to reach a training accuracy nearing 1 by step 700. When the script was modified to utilize a Sigmoid instead, it took till 1,500 steps before it was able to reach a training accuracy nearing 1. Both versions of the file resulted in running out of memory.

Modifying this conventional model even further, I reduced the number of iterations to 5,000. This too resulted in running out of memory. However, it was able to conclude the experiment of running only 5,000 iterations significantly faster that before.