Jin Woo Ahn
Exercise 1
Deep Learning Lab
WS 2018

Exercise 1 Report

The neural network is implemented. The implementation allows users to add arbitrary number of layers and the number of nodes of each of them. There are 4 options for activation: sigmoid, tanh, relu, and linear (None in implementation). Training can be done with either batch or stochastic gradient descent, where user can specify the learning rate and batch size. Additionally, I implemented a plotting function in the Neural network class which plots the train/valid loss and error achieved after every epoch.

For MNIST dataset classification, I used four fully connected layers with softmax output. The first three layers had 100 nodes, while the last layer had 10 (number of classes) nodes. All layers used Relu activation. With learning rate of 0.1, 30 epochs, batch size of 64 and stochastic gradient descent, I could reach the validation and test error of ~2.5%. Under such setting, the train and validation loss diverges after around 5 epochs. After 30 epochs, the training error reaches zero, while the validation error fluctuates around 2.7% The validation error, however, continues decreasing as training proceeds, because the cross entropy loss (our surrogate loss) continues to try to maximize the margin of the decision boundary. Overfitting does not seem to happen under current setting, as the validation and test error are consistent with train error.