DL-HW1

Shai Vaknin (034658492), Ev Zisselman (200479483), Yochai Zur (03050991)

December 5, 2016

Github link: https://github.com/yochaiz/DL-hw

Architecture description

Our model is composed of the flowing layers:
Spatial Convolution (1 input channel, 32 kernels, kernel size 5x5)
2x2 Max pooling
RelU
Batch normalization
Spatial Convolution (32 input channel, 64 kernels, kernel size 3x3)
2x2 Max pooling
RelU
Batch normalization
Spatial Convolution (64 input channel, 32 kernels, kernel size 3x3)
2x2 Max pooling
RelU
Batch normalization
Linear (32*3*3 inputs, 90 outputs)
Dropout (p=0.5)
Linear (90 inputs, 10 outputs)
LogSoftMax

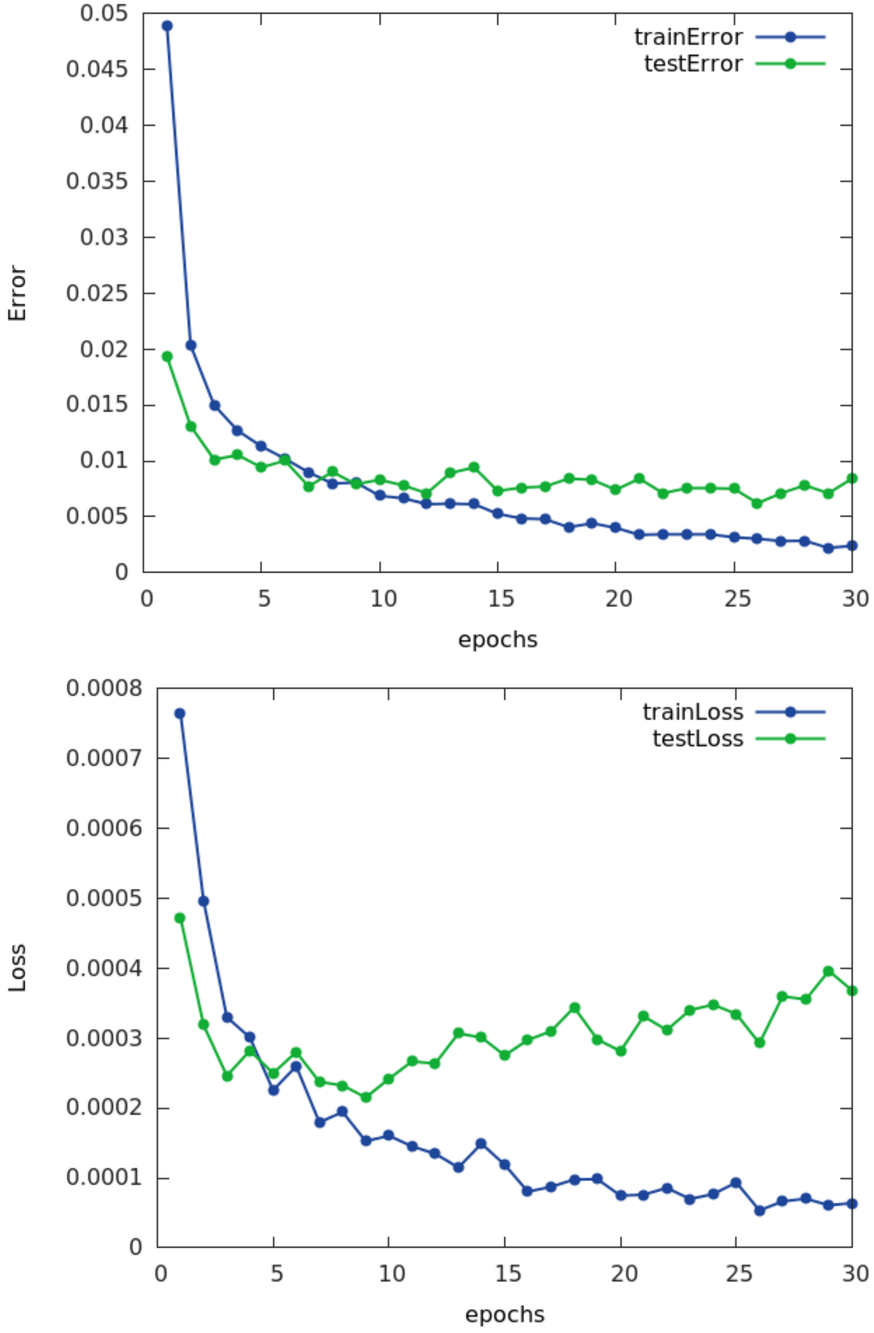
Training procedure

For regularization, we used dropout and batch normalization layers. No weight decay.

We have tried to use Adagrad and Adadelta to train the network. Adagrad reached 99.10% accuracy after 100 epochs, where Adadelta reached 99.379%.

Figure 1 contain the loss and error plots of Adadelta run.

Figure 1: Adadelta (Final accuracy 99.379%)



Summary:

Adadelta achieve better results than Adagrad in a fix time frame. Even though a single epoch of Adagrad (~4.2 seconds) is about ~0.1 second faster than Adadelta (~4.3 seconds), Adadelta require significantly less epoch to converge. Moreover Adadelta achieve error rates Adagrad can only dream of.

We also tried using only fully connected network. Best accuracy we got was ~98.5%.