Deep Learning Lab Course
Exercise 2

Rolf-David Bergdoll
Matr.Nr. 3338345

13.11.2018

The objective of assignment was the implementation of a LeNet-like convolutional neural network and subsequently the optimization of the network's hyperparameters, both by student descent and by random search. This provided some first experiences with TensorFlow and reaffirmed the importance of the right choice of hyperparameter values.

Learning Rate

For this setup, larger learning rates generally delivered better results regarding the validation error, as can be seen in Figure 1.

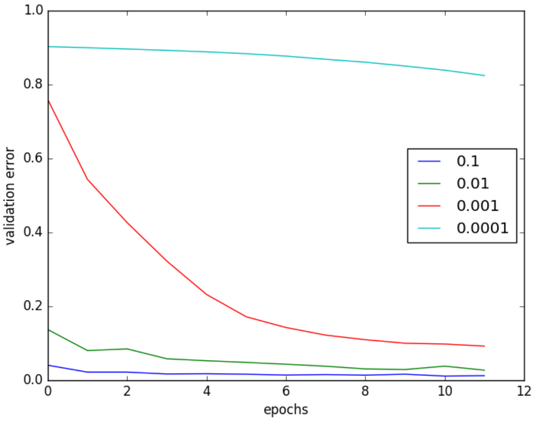

Figure 1: Learning curve for different learning rates.

Using after twelve epochs, the largest learning rate 0.1 had the smallest validation and test errors. However, for a larger number of epochs the smaller rates would most likely catch up to 0.1 at some point.
Generally, a smaller learning rates mean smaller but more precise steps towards the optimum. On the other hand, large learning rates approach the optimum faster, but will eventually start to overshoot the goal – in the worst case they might even land further away from it.

Convolution Type

For the filter size, it was also the largest parameter value that produced the smallest errors within the given horizon (Figure 2).

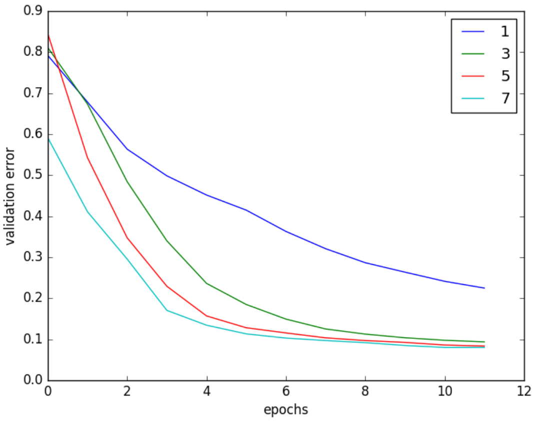

Figure 2: Learning curve for different filter sizes.

However, except for filter size 1, the final validation error were very close to each other. In addition to that, the training took significantly more time for the larger sizes. Thus, if the x-axis was measured by time instead of epochs, the graph for size 3 would probably have passed the one for size 7.

I assume that in general, the optimal filter size depends very much on the setting. Large filters are probably better suited to recognize larger shapes while smaller filter are better at identifying small details.

Random Search

The random search compilation produced the following hyperparameter assignment: Filter size = 3, batch size = 31, learning rate = 0.0019 and number of filters = 8. Interestingly, this goes somewhat against our previous observations regarding learning rate and filter size.

This parametrization produced a test error of 0.0777. Unfortunately, I ran into trouble with the plotting tool, which for some reason refused to compile. Therefore the respective visualizations are missing.