EBERHARD KARLS
UNIVERSITÄT
TÜBINGEN

MATH. - NATURWISS. FAKULTÄT
FACHBEREICH INFORMATIK
KOGNITIVE SYSTEME · PROF. A. ZELL

Deep Neural Networks

Assignment 7

Assignment due by: 26.06.2018, Discussions on: 03.07.2018

For this assignment, we provide a template (`nn.py`) based on last week's assignment. Because you will be performing many different experiments, we have wrapped the training and evaluation code in a function. You should add the parameters necessary to run the experiments to this function and your experiments should be reproducible by calling this function with the appropriate arguments.

Question 1 Tuning the network (2+2+4+5+5+2 points)

(a) Because you will be using networks of different size, you should automate the building of these networks. Instead of creating each layer separately, you can keep track of the current last layer and add new layers in a loop. Test this out by training a network with 5 hidden layers of size 50.

(b) To monitor how training of the network is progressing, you should add some code to collect both the test accuracy and the training accuracy on the current mini batch every 100 training steps. After training has finished, plot both of these on a graph to visualize how the network has trained.

(c) The learning rate is an important parameter for training deep neural networks. If it is chosen too small, the training will not make significant progress and take far too long, while a too large learning rate can prevent the model from really closing in on a minimum. Try out different learning rates over a wide range. How long do they take to converge to a reasonably steady value (if at all, training should not take more than ∼1 minute). For those that converge, examine the accuracy and comment on the results. For this experiment, you should use a network with 2 hidden layers of size 100 each.

(d) When we decide how large we want to make a neural network, there are two basic dimensions we can vary: depth and width. Depth is the number of hidden layers in our network, while width is the size of each of these layers. Try out networks of depth $\{2,3,5,10\}$ and with width of $\{10,100,500\}$ and investigate how long it takes for the network to fully train and what accuracy they achieve. Present your results (time and accuracy) in a table. See if you can detect overfitting in these networks.

(e) Tensorflow has several other activation functions besides the ReLU available. For the following activation functions, test how they affect the training performance and accuracy of the network.

- the logistic sigmoid
- the hyperbolic tangent
- the softplus function
- the leaky ReLU, which is not directly implemented in tensorflow for some reason, but can be simulated with: `lambda x: tf.maximum(0.1*x,x)`

Comment on how each of these activation functions performs as the network gets deeper.

(f) After training the networks, what are your conclusions about the best configuration of the network for the MNIST training task. Give a brief explanation why you think these parameters work well.