Machine Learning for Applications in Computer Vision: Week 2

Date: Friday, 24. April 2015

Please hand in your **printed(!!)** reports in two weeks (08.05.2015) before the lecture.
Please don't forget to put your names and the GitHub link on the report.
Exercises with an asterisk (*) require written answers. You don't have to write down anything about the other exercises.

SSH access to the lab computers:
`ssh  -XY YOURLOGIN@atcremersNR.informatik.tu-muenchen.de`

1 Reading: Deep Learning

Please read `http://neuralnetworksanddeeplearning.com/chap1.html` and `chap2.html`. You don't have to implement the network yourself nor do the exercises.

2 Problem: Backpropagation with a single modified neuron *

Please first complete the reading above. We are going to follow Michael Nielsen's notation.
Suppose we modify a single neuron in a feedforward network so that the output from the neuron is given by $f(\sum_j w_j x_j + b)$, where f is some function other than the sigmoid.
How should we modify the backpropagation algorithm (from chapter 2 of the above reading) in this case?

3 Caffe

3.1 Installation (not required if you use student lab computers)

Render yourself on `http://caffe.berkeleyvision.org`. Follow the installation instructions carefully and get caffe running on your system. Make sure you also install all Python dependencies and `make pycaffe` Alternatively, if you're planning to use the student lab computers, you can skip this step.

3.2 Example project

Caffe is a powerful, yet highly complex framework. There are many things to know and since it's work in progress, not everything is documented for non-experts. So let's make you experts, starting with one of the examples.
Go on `http://caffe.berkeleyvision.org/gathered/examples/imagenet.html` and run the example on your machine. It should work without any problems, errors or severe delays in runtime.

3.3 Your first DCNN classification *

Download the set of sample images from
`https://vision.in.tum.de/teaching/ss2015/mlpractice_ss2015/slides`.
Use the Imagenet network you set up in the previous problem to classify each image. Please provide for each image:

1. the class prediction (class name)
2. the probability for that class (softmax)
3. the entropy of the prediction
4. a discussion of these three quantities

as well as the code of course.

You might have to preprocess the images somehow (please document what you do and why). The examples from the examples folder in caffe can be helpful. We recommend using the Python wrapper.

Print out your report and hand it in before the next lecture.