Prof. Dr. Carsten Meyer
Faculty of Engineering
Christian-Albrechts-Universität Kiel

Neural Networks and Deep Learning – Summer Term 2020

Exercise sheet 1

Submission due: Wednesday, April 22, 13:15 sharp

Note:

Some of the exercises of this lab (starting with sheet 3) are based on the Keras and TensorFlow libraries. TensorFlow is an open source platform for machine learning, supporting matrix (tensor) multiplications on graphical processing units (GPUs) and therefore ideally suited for (deep) neural networks; for more information see https://www.tensorflow.org/. Keras is a high-level neural networks application programming interface (API) running on top of TensorFlow or other libraries like Theano or CNTK with the aim to facilitate using those libraries; see https://keras.io/.

Python code to configure neural networks using Keras and Tensorflow is provided in form of a Jupyter notebook, **`Sheet_1.ipynb`**; for documentation on Jupyter and Jupyter notebooks see https://jupyter.org/. In order to prepare for the coming lab sheets, also the first two lab sheets are provided in the form of a Jupyter notebook (although Keras and Tensorflow are not used yet).

Some lab sheets (e.g. **`Sheet_3.ipynb`**) need additional resources like images etc.; all necessary files are contained in a git repository in the corresponding folder (e.g. **`Lab3`**). In order to copy those files to your computer, you have to install **`git`** on your computer; see e.g. https://git-scm.com/download/win.

One way to copy the required lab files to your computer is to use a git bash shell: On Windows, open an explorer and navigate to a folder where you want to copy all relevant lab files. Then (after installing an appropriate git package), right click the mouse in that directory and select the option “Git Bash Here”. A git bash shell opens; in that shell, enter the following command:

```
git clone https://gitlab+deploy-token-26:XBza882znMmexaQSpjad@git.informatik.uni-kiel.de/las/nndl.git <your target directory>
```

where **`<your target directory>`** has to be replaced by the path of the directory on your local computer that you want to generate and which should contain the downloaded lab files. After cloning, this directory then should contain the lab folders (e.g. **`Lab1`**) with the file **`Sheet_1.ipynb`** and potentially other files.

For later use (starting with lab sheet 3), for python code involving Keras and TensorFlow to work, there are two possibilities (which are also recommended for the first two lab sheets, although the following description explicitly addresses lab sheet 1):

a) (recommended) Execute the code in Google Colab (https://colab.research.google.com/notebooks/intro.ipynb). To this end, you need to create a Google account. Log in and open https://colab.research.google.com/. In the upper right corner, select the "Upload" tab and then "Browse". Locate the cloned **`Lab3`** directory and select the file **`Sheet_3.ipynb`**. Note that you have to run the first cell (containing **`! git clone`** ...) before running any other cell; otherwise you may get an error message from subsequent cells like
   `OSError: nndl/Lab3/exercise3b_input.txt not found.`

   Note that with this option external images are not displayed correctly; instead, you only see e.g. the following line:
   `IMAGE: perceptron`
   For external images to be displayed correctly, option b) has to be invoked (or refer to the pdf exercise sheet).

b) Run the notebook locally. In order for the Jupyter notebook to work properly, also the files and the subdirectory of **`Lab3`** have to be loaded into Jupyter notebook. A possibility for that is to invoke Jupyter notebook from an Anaconda installation (see https://www.anaconda.com/) via an Anaconda Prompt (NOT directly selecting the Jupyter Notebook option): Open an Anaconda prompt, navigate to the **`Lab3`** directory just cloned, and then enter
   `jupyter-notebook`
   into the Anaconda prompt. This opens Jupyter notebook *including* all necessary files. Then, click on **`Sheet_3.ipynb`**; now you can start to work with the Jupyter notebook (and should see external images, which are contained in the **`images`** subdirectory, properly). If you want to execute the cells involving Keras and TensorFlow locally on your computer, you have to install Keras and TensorFlow locally and to configure your environment appropriately.

Run one cell after the other; this should finally provide you with some plots being generated.

Note that changes in **`Sheet_3.ipynb`** which are saved in the colab environment are *not* saved to **`Lab3/Sheet_3.ipynb`**!

Further documentation how to install Jupyter notebooks can be found at:

https://jupyterlab.readthedocs.io/en/stable/getting_started/installation.html
https://jupyter.readthedocs.io/en/latest/install.html#install

For submission, upload a pdf based on the jupyter notebook. The recommended way is opening the Print Preview (File -> Print Preview) and using a PDF printer. Avoid to generate too large cells: If a cell contains more content than which fits on a normal DIN A4 page, the content will be cut during printing. A solution to this is to split up the cell into several smaller cells.

Exercise 1 (Structure of a neuron):

Name the basic elements of a biological neuron in a neural network and briefly summarize the functional role of each element.

Exercise 2 (Type of signal transmission in neuronal components):

Name the type of signal transmission (electrical, chemical, wireless, …) at the axon, the synapses and the dendrites. Indicate whether it is a binary or an analog event. Explain your choice.

Exercise 3 (Neural codes):

What is the basic neuronal "event" of a neuron to "communicate" to other neurons? What are the basic neural codes to represent "meaningful information"? Give a brief explanation of the neural codes.

Exercise 4 (Neuron models and neuron properties):

a) Name the neuron models mentioned in the lecture in the order of descending model complexity (from complex to simple). Briefly summarize their main characteristics.

b) Explain the following terms characterizing the behavior of a neuron:
   - Absolute refractory period
   - Relative refractory period
   - Gain function
   - Interspike interval distribution