University of Stuttgart | ANALYTIC COMPUTING

Foundations of Machine Learning - Exercise (SS 25)

Assignment 10: Neural Networks

Arvindh Arunbabu
arvindh.arun@ki.uni-stuttgart.de

Nadeen Fathallah
Nadeen.Fathallah@ki.uni-stuttgart.de

Cosimo Gregucci
cosimo.gregucci@ki.uni-stuttgart.de

Akram Sadat Hosseini
Akram.Hosseini@ki.uni-stuttgart.de

Farane Jalali
farane.jalali-farahani@ki.uni-stuttgart.de

Osama Mohammed
osama.mohammed@ki.uni-stuttgart.de

Jiaxin Pan
jiaxin.pan@ki.uni-stuttgart.de

Tim Schneider
tim.schneider@ki.uni-stuttgart.de

Jingcheng Wu
jingcheng.wu@ki.uni-stuttgart.de

Daniel Frank
daniel.frank@ki.uni-stuttgart.de

Submit your theoretical solution in ILIAS as a single PDF file.[1] Make sure to list the full names of all participants, matriculation number, study program, and B.Sc. or M.Sc on the first page. Optionally, you can *additionally* upload source files (e.g., PPTX files). Submit your programming task in ILIAS as a single Jupyter notebook. If you have any questions, feel free to ask them in the exercise forum in ILIAS.

Submission is open until Monday, 7th of July, 12:00 noon.

[1] Your drawing software probably allows exporting as PDF. An alternative option is to use a PDF printer. If you create multiple PDF files, use a merging tool (like pdfarranger) to combine the PDFs into a single file.

Task 1: Activation Functions

In neural networks, both the logistic sigmoid function and the hyperbolic tangent (`tanh`) function are commonly used as activation functions. These functions are defined as:

$$\sigma(a) = \frac{1}{1 + e^{-a}} \quad \text{(logistic sigmoid)}$$
$$\tanh(a) = \frac{e^{a} - e^{-a}}{e^{a} + e^{-a}} \quad \text{(hyperbolic tangent)}$$

Consider a general linear combination of logistic sigmoid functions of the form:

$$y(x) = w_0 + \sum_{j=1}^{M} w_j \cdot \sigma\left(\frac{x - \mu_j}{s}\right),$$

where $x \in \mathbb{R}$ is the input, w_j are scalar weights, μ_j are center parameters, and $s > 0$ is a scaling factor.

1. **Task** Derive an expression that expresses the `tanh` function in terms of the sigmoid function.
2. **Task** Rewrite the function $y(x)$ so that it is expressed entirely in terms of the `tanh` function. Derive expressions that relate the new parameters to the original parameters $w_0, w_1, \ldots, w_M$.
3. **Task** Can we expect the same prediction results when we change the activation function from `sigmoid` to `tanh`? Do they behave identically during training deep neural networks? Justify your answer.

Task 2: Single Neuron

Follow the instructions of **Task 2** in the `10_nn.ipynb` notebook and add your implementation below the lines that are tagged with "`# TODO: ...`". Make sure to have the helper function `nn_helper.py` in the same directory.

Task 3: Multi-Layer Neural Networks

Follow the instructions of **Task 3** in the `10_nn.ipynb` notebook and add your implementation below the lines that are tagged with "`# TODO: ...`". Make sure to have the helper function `nn_helper.py` in the same directory.