Computer Science Department
Neuro-Cognitive Modeling

Summer Term 2022

Recurrent and Generative Neural Networks

Exercise Sheet 01

Release: April 29, 2022 Deadline: May 12, 2022 (23:59 pm)

General remarks:

- Download the file `exercisesheet01.zip` from the lecture site (ILIAS). This archive contains files (Java classes), which are required for the exercises.
- All relevant equations can be found in the corresponding lecture slides. Ideally, the exercises should be completed in teams of two students. Larger teams are not allowed.
- **Add a brief documentation (pdf) to your submission**. The documentation should contain protocols of your experiments, parameter choices, and a discussion of the results.
- When questions arise start a forum discussion in ILIAS or contact us via email:
  Sebastian Otte (`sebastian.otte@uni-tuebingen.de`)

Exercise 1 Multilayer Perceptrons [50 points]

(a) Implementation Back-Propagation [30 points]

The mentioned archive contains the class `MultiLayerPerceptron`. This class is an incomplete implementation for multilayer perceptrons but already provides important data structures and such. Your task is to implement the method `backwardPass`, which computes the gradient based on a previously computed network activation. First, compute the δ_j for each neuron

$$\delta_j := \frac{\partial E}{\partial net_j} \tag{1}$$

and, afterward, the partial derivatives for all weights

$$\frac{\partial E}{\partial w_{ij}} = x_i \delta_j. \tag{2}$$

Further, complete the methode `trainStochastic`. This method should realize a stochastic gradient descent with momentum term. Follow the source code remarks.

(b) XOR [10 points]

Test your implementation using the class `MLPXOR`. Find suitable parameters (number hidden layer, hidden layer size, learning rate, momentum rate, bias on/off). The epoch error (the average Root Mean Square Error (RMSE) over all training examples) should be smaller than 0.01 within 10 000 epochs of training (an epoch refers to the one time processing of all training samples). Detail and briefly discuss the results of your investigations.

(c) Geometry [10 points]

Experiment with the class `MLPGeometry`. Again, find suitable parameters, such that the foreground points (two bubbles) are visually sufficiently separated from the background points. Detail and briefly discuss the results of your investigations and add a screenshot of a successful training run. Is this classification problem from a theoretical perspective solvable by a one-layered network (with input- and output layer only)? Give an explanation.