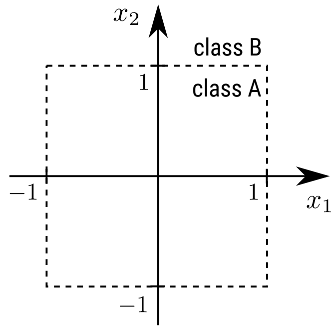
Abteilung Maschinelles Lernen
Institut für Softwaretechnik und theoretische Informatik
Fakultät IV, Technische Universität Berlin
Prof. Dr. Klaus-Robert Müller
Email: klaus-robert.mueller@tu-berlin.de

Exercise Sheet 11

Exercise 1: Designing a Neural Network (20 P)

We would like to implement a neural network that classifies data points in $\mathbb{R}^2$ according to decision boundary given in the figure below.



We consider as an elementary computation the *threshold neuron* whose relation between inputs $(a_i)_i$ and output a_j is given by

$$z_j = \textstyle\sum_i a_i w_{ij} + b_j \qquad\qquad a_j = 1_{z_j > 0}.$$

(a) *Design* at hand a neural network that takes x_1 and x_2 as input and produces the output "1" if the input belongs to class A, and "0" if the input belongs to class B. *Draw* the neural network model and *write down* the weights w_{ij} and bias b_j of each neuron.

Exercise 2: Backward Propagation (5 + 15 P)

We consider a neural network that takes two inputs x_1 and x_2 and produces an output y based on the following set of computations:

$$\begin{aligned}
z_3 &= x_1 \cdot w_{13} + x_2 \cdot w_{23} & z_5 &= a_3 \cdot w_{35} + a_4 \cdot w_{45} & y &= a_5 + a_6 \\
a_3 &= \tanh(z_3) & a_5 &= \tanh(z_5) & & \\
z_4 &= x_1 \cdot w_{14} + x_2 \cdot w_{24} & z_6 &= a_3 \cdot w_{36} + a_4 \cdot w_{46} & & \\
a_4 &= \tanh(z_4) & a_6 &= \tanh(z_6) & &
\end{aligned}$$

(a) *Draw* the neural network graph associated to this set of computations.

(b) *Write* the set of backward computations that leads to the evaluation of the partial derivative $\partial y / \partial w_{13}$. Your answer should avoid redundant computations. Hint: $\tanh'(t) = 1 - (\tanh(t))^2$.

Exercise 3: Neural Network Optimization (10 + 10 + 10 P)

Consider the one-layer neural network

$$f(\boldsymbol{x}) = \boldsymbol{w}^\top \boldsymbol{x}$$

applied to data points $\boldsymbol{x} \in \mathbb{R}^d$, and where $\boldsymbol{w} \in \mathbb{R}^d$ is the parameter of the model. We would like to optimize the mean square error objective:

$$J(\boldsymbol{w}) = \mathbb{E}_{\hat{p}}\Big[\frac{1}{2}(\boldsymbol{w}^\top \boldsymbol{x} - t)^2\Big],$$

where the expectation is computed over an empirical approximation $\hat{p}$ of the true joint distribution $p(\boldsymbol{x}, t)$. The ground truth is known to be of type: $t|\boldsymbol{x} = \boldsymbol{v}^\top \boldsymbol{x} + \varepsilon$, with the parameter $\boldsymbol{v}$ unknown, and where ε is some small i.i.d. Gaussian noise. The input data follows the distribution $\boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\mu}, \sigma^2 I)$ where $\boldsymbol{\mu}$ and σ^2 are the mean and variance.

(a) *Compute* the Hessian of the objective function J at the current location $\boldsymbol{w}$ in the parameter space, and as a function of the parameters $\boldsymbol{\mu}$ and σ of the data.

(b) *Show* that the condition number of the Hessian is given by: $\dfrac{\lambda_1}{\lambda_d} = 1 + \dfrac{\|\boldsymbol{\mu}\|^2}{\sigma^2}$.

(c) *Explain* for this particular problem what would be the advantages and disadvantages of centering the data before training. Your answer should include the following aspects: (1) condition number and speed of convergence, (2) ability to reach a low prediction error.

Exercise 4: Programming (30 P)

Download the programming files on ISIS and follow the instructions.