Ludwig-Maximilians-Universitaet Muenchen
Institute for Informatics
Prof. Dr. Volker Tresp
Rajat Koner
Sebastian Schmoll

28.04.2020

Machine Learning

Summer 2021

Exercise Sheet 01

Exercise 01-1 Recap: Vector Calculus

Compute $\nabla g(\mathbf{x}) = \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}}$ for the functions below with $\mathbf{x} = (x_1, x_2, \dots x_n)^T$. *Hint:* Recall that for a function $g(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}$ with $\mathbf{x} \in \mathbb{R}^n$ holds:

$$\nabla g(\mathbf{x}) = \frac{\partial g(\mathbf{x})}{\partial \mathbf{x}} = \begin{pmatrix} \frac{\partial g(\mathbf{x})}{\partial x_1} \\ \frac{\partial g(\mathbf{x})}{\partial x_2} \\ \vdots \\ \frac{\partial g(\mathbf{x})}{\partial x_n} \end{pmatrix}$$

(a) $g(\mathbf{x}) = 2x_1 + 3x_2^2 + x_3$ for $n = 3$,

(b) $g(\mathbf{x}) = \sum_{i=1}^{n} x_i$,

(c) $g(\mathbf{x}) = \langle \mathbf{x}, \mathbf{x} \rangle = \mathbf{x}^T \mathbf{x}$, the standard scalar product of $\mathbf{x}$ with itself,

(d) $g(\mathbf{x}) = (\mathbf{x} - \boldsymbol{\mu})^2$ for $\boldsymbol{\mu} \in \mathbb{R}^n$.

Exercise 01-2 Boolean Function as Perceptron

Consider the boolean function *or* ($\vee$) for two binary inputs.

(a) Illustrate the different inputs as well as possible separating hyperplanes graphically.

(b) Given the illustration from (a), guess weights for a perceptron (with outputs 0 and 1) such that the perceptron is a classifier for the $\vee$ function. Instead of using the $sign$ function for getting the classification output, as in the lecture, use the Heaviside function f for classification:

$$f(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}$$

(c) Initialize the weight vector as $w = (0, 0, 0)$ and learn the right weights employing the algorithm of the lecture and a learning rate $\eta = 0.2$. Use the following learning rule:

$$w_j \leftarrow w_j + \eta \cdot (y_i - \hat{y}_i) x_{i,j}$$

Start training vector $p_3 = (1, 1)$ and proceed with increasing index (in contrast to the principle of random sampling). Use $p_0 = (0, 0), p_1 = (0, 1)$ and $p_2 = (1, 0)$.

Exercise 01-3 Applying the perceptron learning rule

Let A and B be two classes, both comprising two patterns:

$$A = \left\{ p_1 = \begin{pmatrix} 2 \\ 4 \end{pmatrix},\ p_2 = \begin{pmatrix} 1 \\ 0.5 \end{pmatrix} \right\}, \qquad B = \left\{ p_3 = \begin{pmatrix} 0.5 \\ 1.5 \end{pmatrix},\ p_4 = \begin{pmatrix} 0 \\ 0.5 \end{pmatrix} \right\}$$

Classes A and B are labeled with 1 and -1, respectively.

Solve the following exercises either using pen and paper or a programming language of your choice. Also, visualize the partial results.

(a) How many iterations are required by the pattern-based perceptron learning rule in order to seperate classes A and B correctly if the weight vector w is initialized as $(0, 1, -1)$ and step size η is set to 0.1?

(b) How many iterations are required if $\eta = 0.25$? Is the order of the considered patterns relevant? If so, give an example, otherwise, prove it.

(c) After how many iterations does the gradient-based learning rule terminate for both η? In this case: Is the order of the considered patterns relevant?

Hint: If you need more than 10 iterations, you miscalculated.