Exercise Sheet 7

Exercise 27 Activation Function and Classification

a) Suppose in a multi-layer perceptron the hyperbolic tangent (*tangens hyperbolicus*) is used as the activation function. Derive the factor in the backpropagation formula that results from this activation function! Try to find a similarly simple expression as for the logistic function (cf. the derivation in the lecture)!

b) Consider a three class problem, that is, a data set, in which each sample case is assigned to one class (and one class only). The Iris data discussed in the lecture may serve as an example. In order to solve this problem with a multi-layer perceptron, i.e., to create/train a neural network classifier, one has to choose the number of output neurons and the desired output values for these neurons. What number of output neurons would you choose and what desired output values would you assign if the activation function is (1) the logistic function or (2) the hyperbolic tangent (*tangens hyperbolicus*)? Is it a good idea to derive the desired values from the saturation values of these functions? Justify your answer!

Exercise 28 Convolutional Neural Networks

In image processing grayscale images are often seen as a binary function $f(x, y)$ (x and y: coordinates of a pixel, function value: gray value of the pixel) and represented as a matrix. Consider the following image $\mathbf{A}$:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 10 & 10 & 10 & 0 \\ 0 & 10 & 10 & 10 & 10 & 0 \\ 0 & 10 & 10 & 0 & 0 & 0 \\ 0 & 10 & 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Convolutional Neural Networks are able to efficiently process image data independent of the position and rotation of objects that are visible in the images. For this so-called kernel functions are used, which are *convolved* with the image matrix. The computed *convolved features* are then used to recognize objects.

A possible convolved feature is the existence of an edge. If one assumes that at an edge the brightness changes significantly, edges may be detected as maxima of the first derivative of the image function f. This is the idea underlying the so-called Sobel operator, which consists of two sub-operators or kernel functions, namely

$$\mathbf{S}_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad \text{and} \quad \mathbf{S}_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}.$$

a) Compute the convolved features $\mathbf{G}_x$ and $\mathbf{G}_y$ by sectionwise multiplication (convolution) of the operators $\mathbf{S}_x$ and $\mathbf{S}_y$ with the matrix $\mathbf{A}$, i.e. $\mathbf{G}_x = \mathbf{S}_x * \mathbf{A}$ and $\mathbf{G}_y = \mathbf{S}_y * \mathbf{A}$! (A spreadsheet program may be your friend.)

b) From the resulting direction-dependent matrices $\mathbf{G}_x$ and $\mathbf{G}_y$ a direction-independent matrix $\mathbf{G} = (g_{ij})_{i,j=1,\ldots,6}$ is to be computed. For this the entries $g_{x,ij}$ and $g_{y,ij}$ of the direction-dependent matrices $\mathbf{G}_x$ and $\mathbf{G}_y$, respectively, are squared, then summed, and finally the square root of the result is computed, that is, $g_{ij} = \sqrt{g_{x,ij}^2 + g_{y,ij}^2}$. Compute the matrix $\mathbf{G}$ with the help of the results obtained in part a) and describe the final result!

Exercise 29 Deep Learning: n-bit Parity

In the lecture it was shown how the n-bit parity function can be computed by a chain consisting of one *biimplication* network and $n-2$ *exclusive or* networks. Show how the n-bit parity function may also be computed by a binary tree of sub-networks, each of which computes the biimplication! How many layers does the resulting network have (as a function of n)? How many neurons does is contain in total (as a function of n)?

Exercise 30 Deep Learning: Dropout

a) Consider a 12-layer perceptron with 10 input neurons, 10 neurons in each hidden layer and 1 output neuron. How many parameters in total have to be trained in this network?

b) The network is to be trained with the dropout approach. How many parameters have to be considered per training step if due to the dropout (on average) 5 neurons are deactivated per layer?

c) How does the (expected) number of parameters to consider depend on the dropout rate, that is, the fraction of deactivated neurons per layer?