CHALMERS, GÖTEBORGS UNIVERSITET

EXAM for
ARTIFICIAL NEURAL NETWORKS

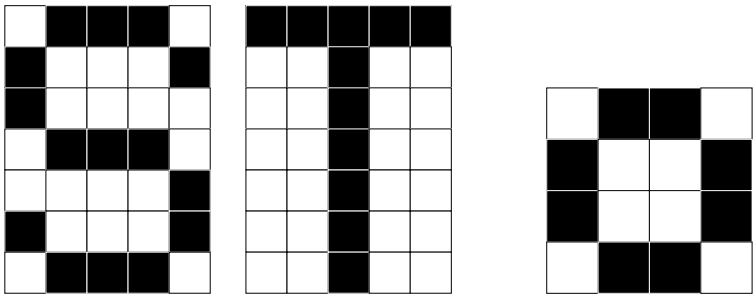
COURSE CODES: **FFR 135, FIM 720 GU, PhD**

Time:	October 26, 2020, at $8^{30} - 12^{30}$
Place:	Zoom
Teachers:	Bernhard Mehlig, 073-420 0988 (mobile)
	Johan Fries, 070-370 1272 (mobile)
Allowed material:	Mathematics Handbook for Science and Engineering
Not allowed:	Any other written material, calculator

Maximum score on this exam: 12 points.
Maximum score for homework problems: 12 points.
To pass the course it is necessary to score at least 5 points on this written exam.
CTH $\geq$14 passed; $\geq$17.5 grade 4; $\geq$22 grade 5,
GU $\geq$14 grade G; $\geq$ 20 grade VG.

---

1. Feature map. The two patterns $\boldsymbol{x}^{(1)}$ and $\boldsymbol{x}^{(2)}$ shown in Figure 1(a) are processed by a very simple convolutional network that has one convolution layer with one single 4×4 kernel with ReLU units, zero threshold, weights w_{ij} as given in Figure 1(b), and stride (1,1). The resulting feature map is fed into a 2×2 max-pooling layer with stride (1,1). Finally there is a fully connected output layer with one output unit $O^{(\mu)}$ with Heaviside activation function. For both patterns determine the resulting feature map and the output of the max-pooling layer. Determine weights W_k and a threshold Θ so that the network output is $O^{(1)} = 0$ for input pattern $\boldsymbol{x}^{(1)}$, and $O^{(2)} = 1$ for input pattern $\boldsymbol{x}^{(2)}$.



(a) (b).

Figure 1: (a) Input patterns $\boldsymbol{x}^{(1)}$ and $\boldsymbol{x}^{(2)}$ with 0/1 bits (□ corresponds to x_i=0 and ■ to x_i=1). (b) Weights w_{ij} of a 4×4 kernel of a feature map. The weights are either 0 or 1 (□ corresponds to $w_{ij} = 0$ and ■ to $w_{ij} = 1$). (Question 1).

2. Hopfield network with hidden units. A Hopfield network with hidden neurons can be used to learn a distribution of input patterns. Consider a Hopfield network with N visible neurons v_j and M hidden neurons h_i. The neurons are binary, with values -1 or $+1$. The network learns by updating the visible neurons according to

$$v_j \leftarrow \text{sgn}\left[b_j^{(v)}\right] \quad \text{with} \quad b_j^{(v)} = \sum_{i=1}^{M} h_i w_{ij}, \tag{1}$$

and by updating the hidden neurons according to

$$h_i \leftarrow \text{sgn}\left[b_i^{(h)}\right] \quad \text{with} \quad b_i^{(h)} = \sum_{j=1}^{N} w_{ij} v_j. \tag{2}$$

In Equations (1) and (2), w_{ij} are the elements of a $M \times N$ weight matrix. Furthermore, sgn$[b]$ is the signum function, sgn$[b] = -1$ if $b < 0$ and $+1$ otherwise. Show that the energy function

$$H = -\sum_{i=1}^{M} \sum_{j=1}^{N} w_{ij} h_i v_j \tag{3}$$

can not increase upon updating one of the hidden neurons according to eq. (2).

3. Backpropagation

Assuming the energy function

$$H = \frac{1}{2}\sum_{i,\mu}(y_i^{(\mu)} - O_i^{(\mu)})^2, \tag{4}$$

derive the update rule for the weights $w_{ij}^{(\ell)}$ for $\ell = 1, 2$ and 3 for the network shown in Figure 2.

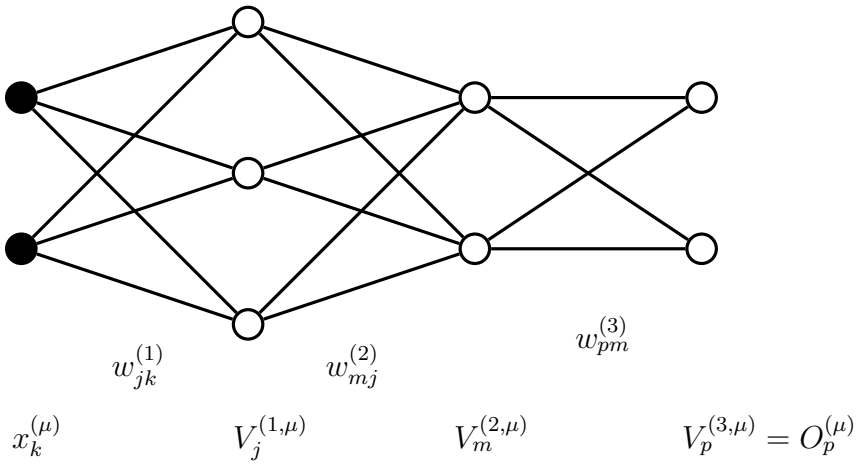


Figure 2: Network for Question 3.

4. XNOR function. The Boolean XNOR function takes two binary inputs. For the inputs $[-1,-1]$ and $[1,1]$ the function evaluates to $+1$, for the other two to -1. Encode the XNOR function as weights w_{ij} in a Hopfield net with three neurons by storing the patterns $\boldsymbol{x}^{(1)} = [-1,-1,1]$, $\boldsymbol{x}^{(2)} = [1,1,1]$, $\boldsymbol{x}^{(3)} = [-1,1,-1]$, and $\boldsymbol{x}^{(4)} = [1,-1,-1]$ using Hebb's rule:

$$w_{ij} = \frac{1}{3}\sum_{\mu=1}^{4} x_i^{(\mu)} x_j^{(\mu)} \quad \text{where} \quad i,j = 1,\ldots,3. \tag{5}$$

The update rule for bit S_i is

$$S_i \leftarrow \operatorname{sgn}\left[\sum_{j=1}^{3} w_{ij} S_j\right], \tag{6}$$

where $\operatorname{sgn}[b]$ is the signum function, $\operatorname{sgn}[b] = -1$ if $b < 0$ and $+1$ otherwise.

(a) What is the weight matrix that you obtain? Feed the stored patterns to the net, and test whether they are stable under synchronous updating.

(b) Use the weight matrix to compute the energy function,

$$H = -\frac{1}{2}\sum_{ij} w_{ij} s_i s_j. \tag{7}$$

Use the fact that the elements s_i only take values ± 1.

(c) Based on your answers to the previous parts, conclude with one or two sentences whether the network is useful for recognising the XNOR function.

(d) What would be the difference if one tried to store just patterns $1, 2$ and 3, and not all 4 patterns?

5. Gradient descent and momentum. Consider the given energy function H as a function of a single weight w as shown in Figure 3. Use the following gradient-descent update rule:

$$\delta w_{n+1} = -\eta \frac{\partial H}{\partial w} + \alpha\, \delta w_n. \tag{8}$$

Here η is the learning rate, and α is the momentum parameter. The weight at time step $n+1$ is then given by $w_{n+1} = w_n + \delta w_n$. Assume that the system is initially at point A. The slope of the segment AB in Figure 3 is $-s$ and the slope of the segment BC is 0. The slope at point A is defined to be $-s$ and that at point B to be 0. The system starts at time step 1, and assume that $\delta w_0 = 0$. Assume that $\eta s = 1/2$.

(a) At which time step n does the system reach point B for $\alpha = 0$?

(b) Repeat the previous calculation for the case $\alpha = 1/2$. You should find that the final equation you obtain for the number of time steps n involves a linear term in n, and an exponential term in n. Plot the linear and exponential functions schematically with n on the x-axis. In this plot, mark the value of n where the two functions intersect, thus obtaining the value of n at which the system reaches point B.

(c) Which of the two cases: $\alpha = 0$ and $\alpha = 1/2$ reaches point B faster? Use the results of the previous two parts to justify your answer.

(d) What is the fate of the two systems $\alpha = 0$ and $\alpha = 1/2$ once they cross point B?

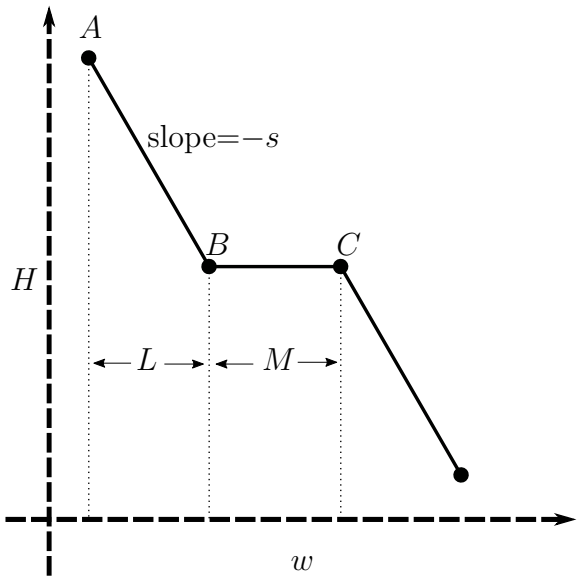


Figure 3: Energy as a function of weight for Question 5.

6. Linear activation function Consider using a linear activation function $g(b) = b$ in a fully connected simple perceptron with one output unit. Fed with a training pattern $\boldsymbol{x}^{(\mu)}$, the output $O^{(\mu)}$ is given by

$$O^{(\mu)} = \boldsymbol{w}^{\mathsf{T}} \boldsymbol{x}^{(\mu)} - \theta. \tag{9}$$

Here $\boldsymbol{w}$ is a column vector of weights, and θ is a scalar threshold. There are p training patterns, $\mu = 1, \ldots, p$. Their target outputs are denoted by $t^{(\mu)}$. For the perceptron considered, the energy function

$$H = \frac{1}{2} \sum_{\mu=1}^{p} \left(O^{(\mu)} - t^{(\mu)} \right)^2 \tag{10}$$

has only one minimum, and it can be found analytically. In the following, you will derive the threshold θ at the minimum.

a) Start by showing that the minimum implies

$$\mathbb{G} \boldsymbol{w} = \boldsymbol{\alpha} + \theta \boldsymbol{\beta} \tag{11a}$$

$$\boldsymbol{\beta}^{\mathsf{T}} \boldsymbol{w} = \theta + \gamma \tag{11b}$$

with

$$\mathbb{G} = \left\langle \boldsymbol{x} \boldsymbol{x}^{\mathsf{T}} \right\rangle, \quad \boldsymbol{\alpha} = \left\langle t \boldsymbol{x} \right\rangle, \quad \boldsymbol{\beta} = \left\langle \boldsymbol{x} \right\rangle \quad \text{and} \quad \gamma = \left\langle t \right\rangle, \tag{12}$$

where $\langle \ldots \rangle$ denotes an average over the training patterns.

b) Assume that $\mathbb{G}$ is invertible, with inverse $\mathbb{G}^{-1}$. Furthermore, assume that $\boldsymbol{\beta}^{\mathsf{T}} \mathbb{G}^{-1} \boldsymbol{\beta} \neq 1$ and solve eqs. (11) for θ.

c) If, in a fully connected multi-layer perceptron, one uses a linear activation function $g(b) = b$, it holds that

$$\begin{aligned} \boldsymbol{V}^{(\mu,\ell)} &= \boldsymbol{w}^{(\ell)} \boldsymbol{V}^{(\mu,\ell-1)} - \boldsymbol{\theta}^{(\ell)} \\ &= \left[\boldsymbol{w}^{(\ell)} \boldsymbol{w}^{(\ell-1)} \right] \boldsymbol{V}^{(\mu,l-2)} - \left[\boldsymbol{w}^{(l)} \boldsymbol{\theta}^{(\ell-1)} + \boldsymbol{\theta}^{(\ell)} \right]. \end{aligned} \tag{13}$$

Here, $\boldsymbol{V}^{(\mu,\ell)}$ is the μ^{th} neuron in the ℓ^{th} hidden layer. Furthermore, $\boldsymbol{w}^{(\ell)}$ and $\boldsymbol{\theta}^{(\ell)}$ are the weight matrix and theshold vector for the neurons in the ℓ^{th} hidden layer. Write at most three sentences where you, based on eq. (13), argue that a non-linear activation function is essential for a multi-layer perceptron.