CHALMERS, GÖTEBORGS UNIVERSITET

RE-EXAM for
ARTIFICIAL NEURAL NETWORKS

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

Time:	January 20, 2018, at $8^{30} - 12^{30}$
Place:	SB Multisal
Teachers:	Bernhard Mehlig, 073-420 0988 (mobile)
	Johan Fries, 070-370 1272 (mobile), visits once at 9^{00}
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. Recognition of one pattern. In the deterministic Hopfield model, the state S_i of the i-th neuron is updated according to the Mc-Culloch Pitts rule

$$S_i \leftarrow \operatorname{sgn}(b_i), \tag{1}$$

$$\text{where } b_i = \sum_{j=1}^{N} w_{ij} S_j \,. \tag{2}$$

Here N is the number of neurons, w_{ij} are the weights. The weights depend on p patterns $\boldsymbol{\zeta}^{(\mu)} = (\zeta_1^{(\mu)}, \ldots, \zeta_N^{(\mu)})^{\mathsf{T}}$ stored in the network according to Hebb's rule:

$$w_{ij} = \frac{1}{N} \sum_{\mu=1}^{p} \zeta_i^{(\mu)} \zeta_j^{(\mu)} \quad \text{for } i, j = 1, \ldots, N\,. \tag{3}$$

Here $\zeta_i^{(\mu)}$ takes values 1 or -1.

a) Consider the patterns in Figure 1. Compute the quantity

$$\sum_{j=1}^{N} \zeta_j^{(\mu)} \zeta_j^{(\nu)} \tag{4}$$

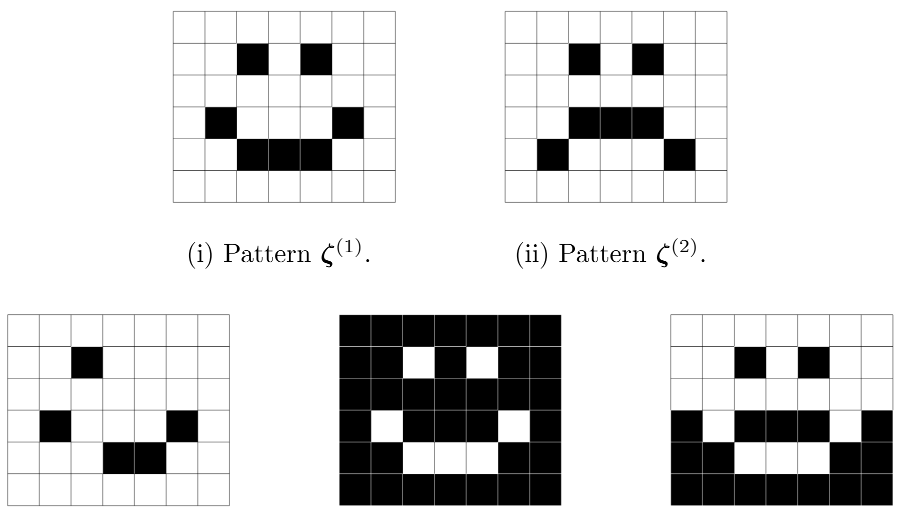
for the following ten combinations of μ and ν:

- $\mu = 1$ and $\nu = 1, \ldots, 5$,
- $\mu = 2$ and $\nu = 1, \ldots, 5$.

(Hint: The result can be read off from the Hamming distances between the patterns shown in Figure 1.)
(**0.5** p)

b) The patterns stored in the network are $\boldsymbol{\zeta}^{(1)}$ and $\boldsymbol{\zeta}^{(2)}$. Let $b_i^{(\nu)}$ denote b_i obtained by substituting S_j by $\zeta_j^{(\nu)}$ in eq. (2). Use your answer to a) to compute $b_i^{(\nu)}$ for $\nu = 1, \ldots, 5$. Express these results as linear combinations of $\zeta_i^{(1)}$ and $\zeta_i^{(2)}$. (**1** p)

c) Feed the patterns in Figure 1 to the network. Which ones are stable under synchronous updating? Show how you arrive to your results. (**0.5** p)



(i) Pattern $\boldsymbol{\zeta}^{(1)}$. (ii) Pattern $\boldsymbol{\zeta}^{(2)}$.

(iii) Pattern $\boldsymbol{\zeta}^{(3)}$. (iv) Pattern $\boldsymbol{\zeta}^{(4)}$. (v) Pattern $\boldsymbol{\zeta}^{(5)}$.

Figure 1: The patterns studied in Question 1. Each pattern consists of 42 bits $\zeta_i^{(\mu)}$ taking values $+1$ or -1. A black square for bit i in pattern μ stands for $\zeta_i^{(\mu)} = 1$, a white square stands for $\zeta_i^{(\mu)} = -1$.

2. Linearly inseparable problem. A classification problem is specified in Figure 2, where a grey triangle in input space is shown. The aim is to map input patterns $\boldsymbol{\xi}^{(\mu)}$ to outputs $O^{(\mu)}$ as follows: if a point with coordinate vector $\boldsymbol{\xi}^{(\mu)}$ lies inside the triangle it is mapped to $O^{(\mu)} = 1$, but if $\boldsymbol{\xi}^{(\mu)}$ is outside the triangle it is mapped to $O^{(\mu)} = 0$. How patterns on the boundary of the triangle are classified is not important.

a) This problem is not linearly separable. Show this by constructing a counter-example using four input patterns. (**0.5** p)

b) The problem can be solved by a perceptron with one hidden layer with three neurons v_i for $i = 1\,,\,2\,,\,3$, where neuron i is given the value

$$v_i^{(\mu)} = H\left[-\theta_i + \sum_{j=1}^{2} w_{ij}\xi_j^{(\mu)}\right] \tag{5}$$

and the output is given the value

$$O^{(\mu)} = H\left[-T + \sum_{i=1}^{3} W_i v_i^{(\mu)}\right]. \tag{6}$$

Here w_{ij} and W_i are weights and θ_i and T are thresholds. In eqs. (5) and (6),

$$H(b) = \begin{cases} 0 & \text{if} \quad b \leq 0 \\ 1 & \text{if} \quad b > 0 \end{cases}. \tag{7}$$

Find weights and thresholds that solve the classification problem. (**1** p)

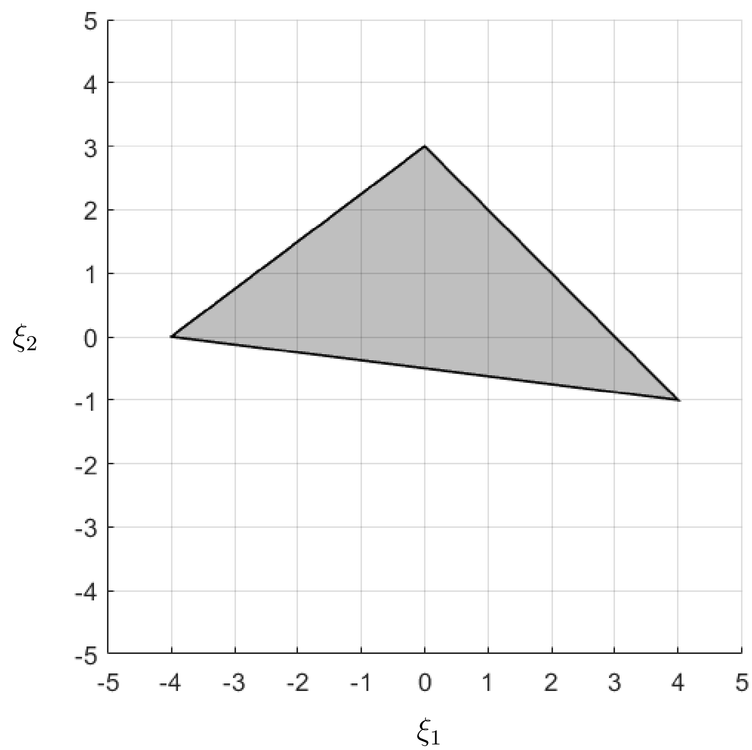


Figure 2: Specification of classification problem in Question 2 (see text). The input space is the (ξ_1,ξ_2)-plane.

3. Backpropagation. A multilayer perceptron has $L-1$ hidden layers, one input layer and one output layer. The state of a neuron $v_i^{(m,\mu)}$ in the m-th layer is given by

$$v_i^{(m,\mu)} = g\left(b_i^{(m,\mu)}\right), \tag{8}$$

where

$$b_i^{(m,\mu)} = -\theta_i^{(m)} + \sum_j w_{ij}^{(m)} v_j^{(m-1,\mu)} \tag{9}$$

for $m > 1$ and

$$b_i^{(m,\mu)} = -\theta_i^{(m)} + \sum_j w_{ij}^{(m)} \xi_j^{(\mu)} \tag{10}$$

for $m = 1$. In eqs. (9) and (10) $w_{ij}^{(m)}$ are weights and $\theta_i^{(m)}$ are thresholds. $g = g(b)$ in eq. (8) is the activation function and $\xi_j^{(\mu)}$ in eq. (10) is the input to the perceptron. An element $O_i^{(\mu)}$ in the output is given by

$$O_i^{(\mu)} = g\left(b_i^{(L,\mu)}\right), \tag{11}$$

with $b_i^{(L,\mu)}$ given by eq. (9).

a) Draw this network. Indicate where elements $\xi_i^{(\mu)}$, $b_i^{(m,\mu)}$, $v_i^{(m,\mu)}$, $O_i^{(\mu)}$, $w_{ij}^{(m)}$ and $\theta_i^{(m)}$ belong in your illustration. How does the total number of weights and thresholds depend on the network architecture? (**0.5** p)

b) Find the recursive rule for how the derivatives

$$\frac{\partial v_i^{(m,\mu)}}{\partial w_{qr}^{(p)}} \tag{12}$$

depend on the derivatives

$$\frac{\partial v_j^{(m-1,\mu)}}{\partial w_{qr}^{(p)}} \tag{13}$$

if $p < m$. (**1** p)

c) Evaluate the derivative in eq. (12) for $p = m$. (**0.5** p)

d) The update rule for the weights corresponds to gradient descent of the energy function

$$H = \frac{1}{2} \sum_{\mu} \sum_{i} \left(O_i^{(\mu)} - \zeta_i^{(\mu)} \right)^2 , \tag{14}$$

, where $\zeta_i^{(\mu)}$ is the target output of input pattern $\xi_i^{(\mu)}$. The update is done in batch mode using the learning rate η. Find the update rule for the weight $w_{ij}^{(L-2)}$. (**1** p)

4. True/False questions. Indicate whether the following statements are true or false. 13-14 correct answers give 2 points, 11-12 correct answers give 1.5 points, 9-10 correct answers gives 1 point and, 8 correct answers give 0.5 points and 0-7 correct answers give zero points. (**2** p)

1. You need access to the state of all neurons in a multilayer perceptron when updating all weights through backpropagation.
2. Consider the Hopfield network. If a pattern is stable it must be an eigenvector of the weight matrix.
3. If you store two orthogonal patterns in a Hopfield network, they will always turn out unstable.
4. Kohonens algorithm learns convex distributions better than concave ones.
5. The number of N-dimensional Boolean functions is 2^N.
6. The weight matrices in a perceptron are symmetric.
7. Using $g(b) = b$ as activation function and putting all thresholds to zero in a multilayer perceptron, allows you to solve some linearly inseparable problems.
8. You need at least four radial basis functions for the XOR-problem to be linearly separable in the space of the radial basis functions.
9. Consider $p > 2$ patterns uniformly distributed on a circle. None of the eigenvalues of the covariance matrix of the patterns is zero.
10. Even if the weight vector in Oja's rule equals its stable steady state at one iteration, it may change in the following iterations.
11. If your Kohonen network is supposed to learn the distribution $P(\boldsymbol{\xi})$, it is important to generate the patterns $\boldsymbol{\xi}^{(\mu)}$ before you start training the network.

12. All one-dimensional Boolean problems are linearly separable.

13. In Kohonen's algorithm, the neurons have fixed positions in the output space.

14. Some elements of the covariance matrix are variances.

5. Oja's rule.

a) The output of Oja's rule for the input pattern $\boldsymbol{\xi}^{(\mu)}$ is

$$\zeta^{(\mu)} = \sum_i w_i \xi_i^{(\mu)}, \tag{15}$$

and the update rule based on this pattern is $w_i \to w_i + \delta w_i^{(\mu)}$ with

$$\delta w_i^{(\mu)} = \eta \zeta^{(\mu)} \left(\xi_i^{(\mu)} - \zeta^{(\mu)} w_i \right). \tag{16}$$

Let $\langle \delta w_i \rangle$ denote the update of w_i averaged over the input patterns. Show that $\langle \delta w_i \rangle = 0$ implies

$$\sum_i w_i w_i = 1. \tag{17}$$

(**1** p)

b) Calculate the principal component of the patterns in Figure 3. (**1** p)

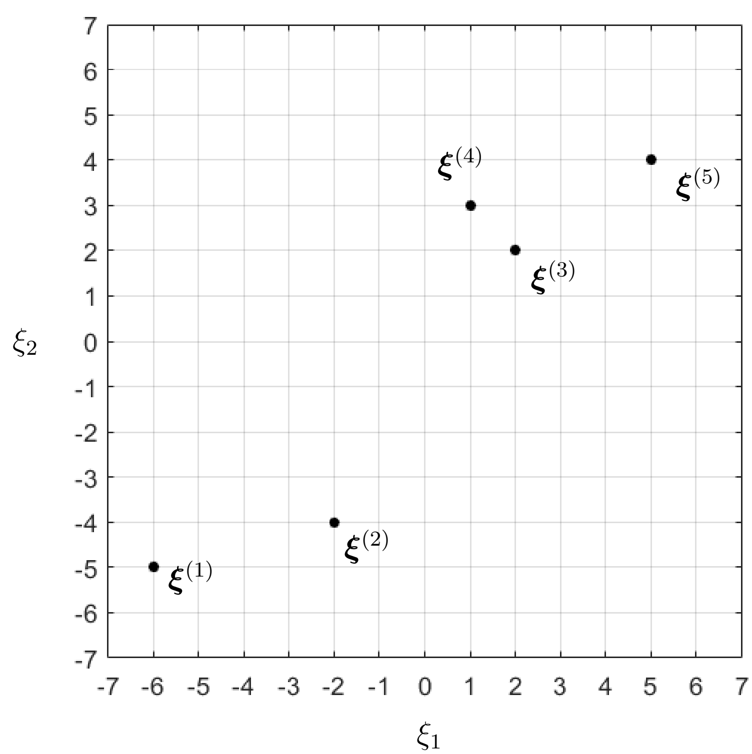


Figure 3: Patterns studied in Question 5b.

6. General Boolean problems. Any N dimensional Boolean problem can be solved using a perceptron with one hidden layer consisting of 2^N neurons. Here we consider $N = 3$ and $N = 2$.

a) The three-dimensional parity problem is specified in Figure 4. Here input bits $\xi_i^{(\mu)}$ for $i = 1\,, 2\,, 3$ are either +1 or -1. The problem is solved by that the output $O^{(\mu)}$ of the network is +1 if there is an odd number of positive bits in $\boldsymbol{\xi}^{(\mu)}$, and -1 if the number of positive bits are even. In one solution, a neuron $v_i^{(\mu)}$ for $i = 1\,, \ldots\,, 2^N$ in the hidden layer depends on the input $\boldsymbol{\xi}^{(\mu)}$ according to

$$v_i^{(\mu)} = \begin{cases} 1 & \text{if} \quad -\theta_i + \sum_j w_{ij}\xi_j^{(\mu)} > 0 \\ 0 & \text{if} \quad -\theta_i + \sum_j w_{ij}\xi_j^{(\mu)} \le 0 \end{cases} \quad , \tag{18}$$

where the weights and thresholds are given by

$$w_{ij} = \xi_j^{(i)} \quad \text{and} \quad \theta_i = 2. \tag{19}$$

The output is given by

$$O^{(\mu)} = \sum_j W_j v_j^{(\mu)}. \tag{20}$$

Determine the weights W_j. (**1** p)

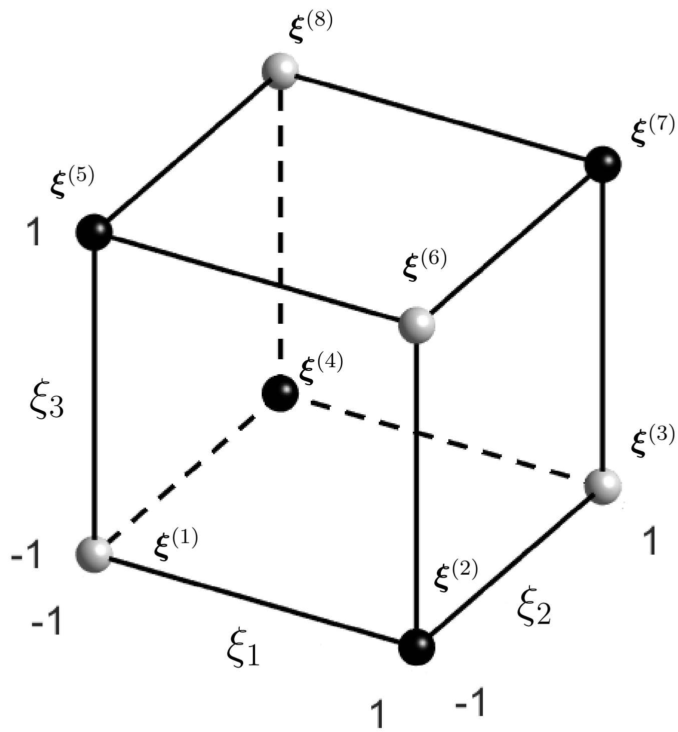


Figure 4: The three-dimensional parity problem (see text). A white ball indicates $O^{(\mu)} = -1$, and a black ball indicates $O^{(\mu)} = +1$.

b) The problem in two dimensions analogous to the problem in a) is the XOR-problem. Draw the decision boundaries of the hidden neurons in the solution to the XOR-problem that is analogous to the solution given in a). (**0.5** p)