CHALMERS, GÖTEBORGS UNIVERSITET

EXAM for
ARTIFICIAL NEURAL NETWORKS

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

Time:	October 25, 2021, at $08^{30} - 12^{30}$
Place:	Lindholmen-salar
Teachers:	Bernhard Mehlig, 073-420 0988 (mobile)
	Anshuman Dubey, 072-190 6469 (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 >13.5 passed; >17 grade 4; >21.5 grade 5,
GU >13.5 grade G; > 19.5 grade VG.

---

1. Convolutional network. Construct a convolutional neural network with one convolution layer with a single 2×2 kernel with ReLU neurons, stride (1,1), and padding (0,0). This is followed by a 2×3 max-pooling layer with stride (1,1), and a fully connected classification layer with two output neurons and a signum (sgn) activation function to classify the patterns shown in Figure 1. Specify the weights of the kernel as well as weights and thresholds of the classification layer. **2p**.

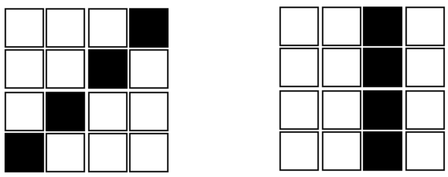

Figure 1: Patterns to be classified by convolutional network. Question 1.

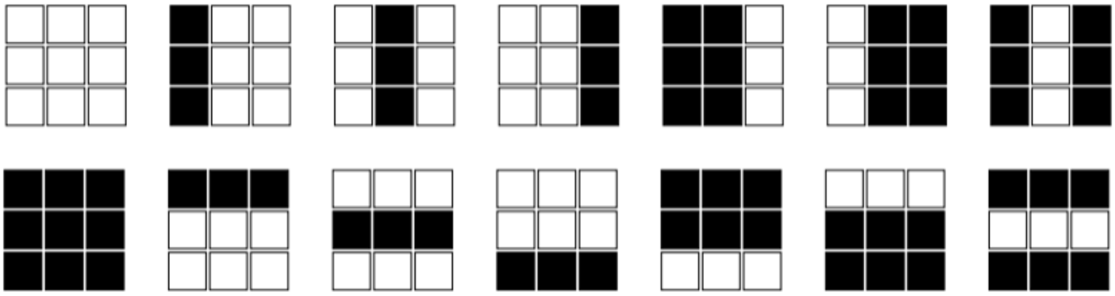

Figure 2: Bars-and-stripes ensemble, ■ corresponds to $x = 1$, and □ to $x = 0$. Question 2.

2. Boltzmann machine. Boltzmann machines approximate a binary data distribution $P_{\rm data}(\boldsymbol{x})$ in terms a model distribution, the Boltzmann distribution.
(a) Without hidden units, the Boltzmann distribution reads $P_{\rm B}(\boldsymbol{s}) = Z^{-1}\exp(-\beta H)$ with energy function $H = -\frac{1}{2}\sum_{i\neq j} w_{ij}s_i s_j$. A measure for how well $P_{\rm B}$ approximates $P_{\rm data}$ is the Kullback-Leibler divergence

$$D_{\rm KL} = \sum_{\mu=1}^{p} P_{\rm data}(\boldsymbol{x}^{(\mu)})\log[P_{\rm data}(\boldsymbol{x}^{(\mu)})/P_{\rm B}(\boldsymbol{s}=\boldsymbol{x}^{(\mu)})]\,. \tag{1}$$

In the sum over μ, terms with $P_{\rm data}(\boldsymbol{x}^{(\mu)})$= 0 are set to zero. Show that $D_{\rm KL}$ is non-negative, and that it assumes its global minimum $D_{\rm KL} = 0$ for $P_{\rm data}(\boldsymbol{x}^{(\mu)}) = P_{\rm B}(\boldsymbol{s} = \boldsymbol{x}^{(\mu)})$.
(b) Explain why one needs hidden units to approximate the bars-and-stripes distribution, where $P_{\rm data} = 1/14$ for the patterns shown in Figure 2, and equal to zero otherwise. **2p**.

3. Linearly inseparable classification problem. A classification problem is given in Figure 3. Inputs $\boldsymbol{x}^{(\mu)}$ inside the gray triangle have targets $t^{(\mu)} = 1$, inputs outside the triangle $t^{(\mu)} = -1$. The problem can be solved by a perceptron with one hidden layer with three neurons $V_j^{(\mu)} = \operatorname{sgn}\big(-\theta_j + \sum_{k=1}^{2} w_{jk}x_k^{(\mu)}\big)$, for j = 1,2,3. The network output is computed as $O^{(\mu)} = \operatorname{sgn}(-\Theta+\sum_{j=1}^{3} W_j V_j^{(\mu)})$. Find weights w_{jk}, W_j and thresholds θ_j, Θ that solve the classification problem. **2p**.

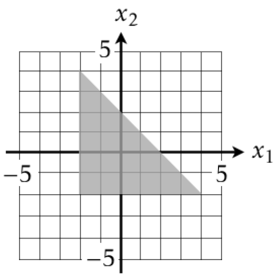

Figure 3: Classification problem. Question 3.

4. Backpropagation. Figure 4 shows a chain of neurons with residual connections. (a) Using the energy function $H = \frac{1}{2}(t - V^{(L)})^2$, show that the learning rule for $w^{(L,L-1)}$ is

$$\delta w^{(L,L-1)} \equiv -\eta \frac{\partial H}{\partial w^{(L,L-1)}} = \eta\,(t - V^{(L)})g'(b^{(L)})V^{(L-1)}. \tag{2}$$

Here $b^{(\ell)}$ is the local field of neuron $V^{(\ell)}$, $g(b)$ is its activation function, and $g'(b)$ is the derivative of g with respect to b. (b) Compute the learning rules for $w^{(L-1,L-2)}$ and $w^{(L-2,L-3)}$. **2p.**

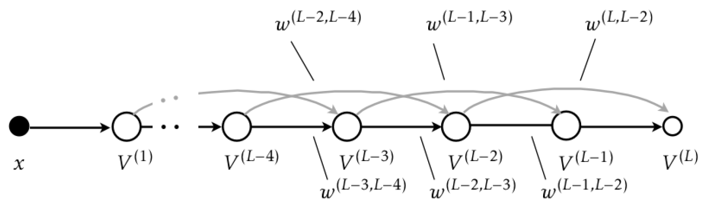

Figure 4: Chain of neurons with residual connections. Question 4.

5. Binary stochastic neurons have the asynchronous update rule

$$s'_m = \begin{cases} +1 & \text{with probability} \quad p(b_m)\,, \\ -1 & \text{with probability} \quad 1 - p(b_m)\,. \end{cases} \tag{3}$$

Here, $b_m = \sum_j w_{mj}s_j - \theta_m$ is the local field, and $p(b) = \frac{1}{1+\mathrm{e}^{-2\beta b}}$. Under certain conditions, Eq. (3) is equivalent to the following rule. *Change* s_m to s'_m with probability

$$\text{Prob}(s_m \to s'_m) = \frac{1}{1 + e^{\beta \Delta H_m}}\,, \tag{4a}$$

with

$$\Delta H_m = H(\ldots, s'_m, \ldots) - H(\ldots, s_m, \ldots)\,. \tag{4b}$$

with energy function $H = -\frac{1}{2}\sum_{ij} w_{ij}s_i s_j + \sum_i \theta_i s_i$.
(a) Assuming that the weight matrix is symmetric and that its diagonal elements are zero, show that:

$$\Delta H_m = -b_m(s'_m - s_m). \tag{5}$$

(b) Using Eq. (5), derive Eq. (4) from Eq. (3). **2p.**

6. Oja's rule for a linear neuron with weight vector $\boldsymbol{w}$, input $\boldsymbol{x}$, and output $y = \boldsymbol{w}^{\mathsf{T}}\boldsymbol{x}$ reads $\delta\boldsymbol{w} = \eta y(\boldsymbol{x} - y\boldsymbol{w})$. Show that for zero-mean data, $\langle \boldsymbol{x} \rangle = 0$, this learning rule has a steady state $\boldsymbol{w}^*$ equal to the leading normalised eigenvector of the matrix $\langle \boldsymbol{x}\boldsymbol{x}^{\mathsf{T}} \rangle$. The leading eigenvector is the one corresponding to the largest eigenvalue, and the average $\langle \cdots \rangle$ is over the data distribution of inputs $\boldsymbol{x}$. **2p.**