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**.

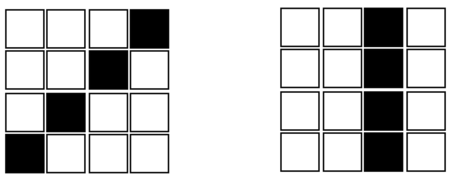


Figure 1: Patterns to be classified by convolutional network. Question 1.

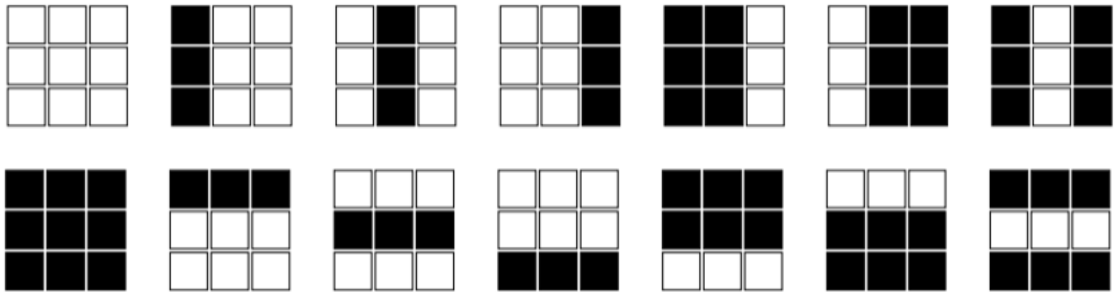


Figure 2: Bars-and-stripes ensemble, ■ corresponds to $x = 1$, and □ to $x = 0$. Question 2.

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

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

In the sum over μ, terms with $P_{\text{data}}(\boldsymbol{x}^{(\mu)})$= 0 are set to zero. Show that D_{KL} is non-negative, and that it assumes its global minimum $D_{\text{KL}} = 0$ for $P_{\text{data}}(\boldsymbol{x}^{(\mu)}) = P_{\text{B}}(\boldsymbol{s} = \boldsymbol{x}^{(\mu)})$.
(b) Explain why one needs hidden units to approximate the bars-and-stripes distribution, where $P_{\text{data}} = 1/14$ 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 $V_j^{(\mu)} = \text{sgn}\big(-\theta_j + \sum_{k=1}^{2} w_{jk}x_k^{(\mu)}\big)$, for j = 1,2,3. The network output is computed as $O^{(\mu)} = \text{sgn}(-\Theta + \sum_{j=1}^{3} W_j V_j^{(\mu)})$. Find weights w_{jk}, W_j and thresholds θ_j, Θ that solve the classification problem. **2p**.

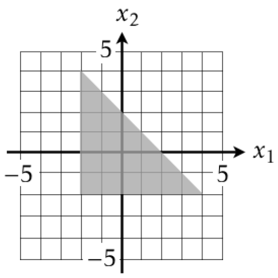


Figure 3: Classification problem. Question 3.

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**.

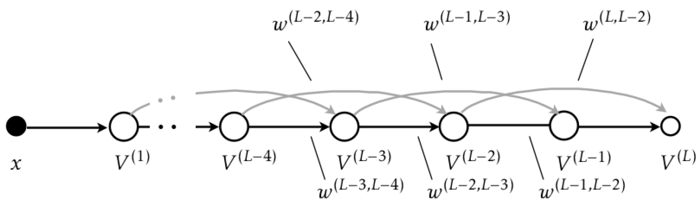


Figure 4: Chain of neurons with residual connections. Question 4.

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

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

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**.

CHALMERS, GÖTEBORGS UNIVERSITET

SOLUTIONS FOR EXAM for
ARTIFICIAL NEURAL NETWORKS
October 25, 2021

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

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.

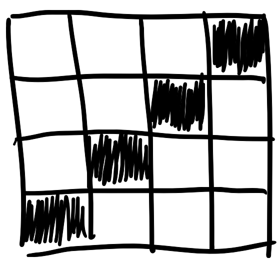
---

1. Convolutional network.

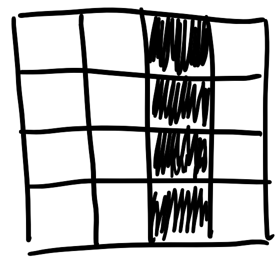
Convolutional network

Arbitrary choice

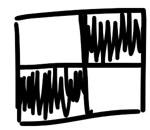
Pattern 1

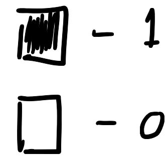


Pattern 2



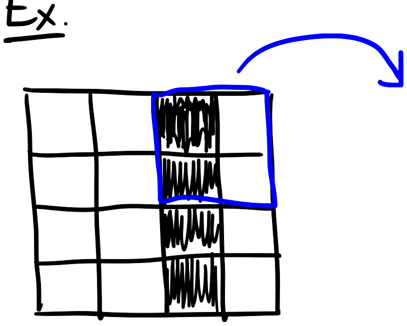
Kernel





- Apply kernel to patterns with stride (1,1) and padding (0,0,0,0), using a ReLU activation function

Ex.



$$\begin{pmatrix} 1\cdot 0 & 0\cdot 1 \\ 1\cdot 1 & 0\cdot 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}$$

Sum the entries of the resulting matrix and apply ReLU activation function: $g(0+0+1+0) = 1$

- Resulting convolution layers:

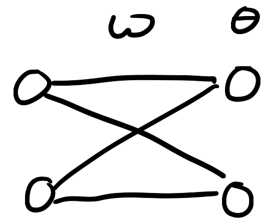
$$V^{(1)} = \begin{pmatrix} 0 & 0 & 2 \\ 0 & 2 & 0 \\ 2 & 0 & 0 \end{pmatrix}, \quad V^{(2)} = \begin{pmatrix} 0 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{pmatrix}$$

- Apply (2×3) max-pooling layer with stride $(1,1)$

$$M^{(1)} = \begin{pmatrix} 2 \\ 2 \end{pmatrix}, \quad M^{(2)} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

- Fully connected classification layer with signum activation function sgn:

Two inputs from max-pooling layer and two output neurons



w : (2×2) weight matrix

θ : (2×1) threshold vector

$$O_i^{(\mu)} = \mathrm{sgn}\Big(\sum_j^2 w_{ij} M_j^{(\mu)} - \theta_i\Big), \quad \mu = \text{pattern}$$

Pattern 1: $$\begin{pmatrix} O_1^{(1)} \\ O_2^{(1)} \end{pmatrix} = \begin{pmatrix} \mathrm{sgn}(2w_{11} + 2w_{12} - \theta_1) \\ \mathrm{sgn}(2w_{21} + 2w_{22} - \theta_2) \end{pmatrix}$$

Pattern 2: $$\begin{pmatrix} O_1^{(2)} \\ O_2^{(2)} \end{pmatrix} = \begin{pmatrix} \mathrm{sgn}(w_{11} + w_{12} - \theta_1) \\ \mathrm{sgn}(w_{21} + w_{22} - \theta_2) \end{pmatrix}$$

Choose: $$w = \begin{pmatrix} 1 & 1 \\ -1 & -1 \end{pmatrix} \quad \text{and} \quad \theta = \begin{pmatrix} 3 \\ -3 \end{pmatrix}$$

Pattern 1: $$\begin{pmatrix} O_1^{(1)} \\ O_2^{(1)} \end{pmatrix} = \begin{pmatrix} \mathrm{sgn}(4-3) \\ \mathrm{sgn}(-4+3) \end{pmatrix} = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

Pattern 2: $$\begin{pmatrix} O_1^{(2)} \\ O_2^{(2)} \end{pmatrix} = \begin{pmatrix} \mathrm{sgn}(2-3) \\ \mathrm{sgn}(-2+3) \end{pmatrix} = \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

The patterns can be classified using the parameters

$$w = \begin{pmatrix} 1 & 1 \\ -1 & -1 \end{pmatrix} \quad \text{and} \quad \theta = \begin{pmatrix} 3 \\ -3 \end{pmatrix}$$

2. Boltzmann machine (a) Start with the KL divergence,

$$D_{KL} = \sum_{\mu=1}^{p} P_{data}(x^\mu) \log \frac{P_{data}(x^\mu)}{P_B(s = x^\mu)} \tag{1}$$

$$= - \sum_{\mu=1}^{p} P_{data}(x^\mu) \log \frac{P_B(s = x^\mu)}{P_{data}(x^\mu)}. \tag{2}$$

Use the inequality $\log z \leq z - 1$, where the equality holds iff $z = 1$.

$$- \sum_{\mu=1}^{p} P_{data}(x^\mu) \log \frac{P_B(s = x^\mu)}{P_{data}(x^\mu)} \geq - \sum_{\mu=1}^{p} P_{data}(x^\mu) \left[\frac{P_B(s = x^\mu)}{P_{data}(x^\mu)} - 1 \right], \tag{3}$$

$$\geq - \sum_{\mu=1}^{p} \left[P_B(s = x^\mu) - P_{data}(x^\mu) \right], \tag{4}$$

Since the probabilities P_B, P_{data} must sum to 1,

$$- \sum_{\mu=1}^{p} P_{data}(x^\mu) \log \frac{P_B(s = x^\mu)}{P_{data}(x^\mu)} \geq - [1 - 1] \geq 0, \tag{5}$$

with the equality valid if and only if $P_B(s = x^\mu) = P_{data}(x^\mu)$.
(b) Hidden units are required because 3-point correlations must be considered to differentiate between bars and stripes.

3. Linearly inseparable classification problem The weights and thresholds for the three neurons can be inferred by writing the equations of the three decision boundaries:

$$f_1(x_1, x_2) = -x_1 - x_2 + 2 = 0 \tag{6}$$
$$f_2(x_1, x_2) = x_1 + 0\,x_2 + 2 = 0 \tag{7}$$
$$f_3(x_1, x_2) = 0\,x_1 + x_2 + 2 = 0. \tag{8}$$

For each decision boundary, $f_i(x_1, x_2) = 0$ on the boundary, $f_i(x_1, x_2) > 0$ on the side containing the origin, $(0, 0)$, and $f_i(x_1, x_2) < 0$ on the other side of the decision boundary. Since $f_i(0, 0) > 0$ for all i, the sign of the coefficients of x_1, x_2 are correct.
Thus,

$$w = \begin{bmatrix} -1 & -1 \\ 1 & 0 \\ 0 & 1 \end{bmatrix}, \theta = \begin{bmatrix} -2 \\ -2 \\ -2 \end{bmatrix} \tag{9}$$

Finally, choosing $W = [1, 1, 1]$ and $\Theta = 5/2$ maps the region enclosed by the three decision boundaries to $+1$ but the region outside to -1.

4. Backpropagation

Backpropagation

(a) with $H = \frac{1}{2}(t - V^{(L)})^2$ and $\delta w^{(L,L-1)} = -\eta \frac{\partial H}{\partial w^{(L,L-1)}}$

$$\frac{\partial H}{\partial w^{(L,L-1)}} = \frac{1}{2}\frac{\partial}{\partial w^{(L,L-1)}}\left(t - V^{(L)}\right)^2 = -\left(t - V^{(L)}\right)\frac{\partial V^{(L)}}{\partial w^{(L,L-1)}}$$

$$= -(t - V^{(L)})\frac{\partial}{\partial w^{(L,L-1)}} g(b^{(L)})$$

$$(*) = -(t - V^{(L)})\, g'(b^{(L)})\frac{\partial}{\partial w^{(L,L-1)}}\left(w^{(L,L-1)}V^{(L-1)} + w^{(L,L-2)}V^{(L-2)} - \theta^{(L)}\right)$$

$$= -(t - V^{(L)})\, g'(b^{(L)})\, V^{(L-1)}$$

$$\boxed{\therefore \quad \delta w^{(L,L-1)} = \eta (t - V^{(L)})\, g'(b^{(L)})\, V^{(L-1)}}$$

(b) Performing the same steps up until $(*)$ we have for $\delta w^{(L-1,L-2)}$:

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

$$= -(t - V^{(L)})\, g'(b^{(L)})\, w^{(L,L-1)}\, g'(b^{(L-1)})\, V^{(L-2)}$$

$$\boxed{\therefore \quad \delta w^{(L-1,L-2)} = \eta (t - V^{(L)})\, g'(b^{(L)})\, w^{(L,L-1)}\, g'(b^{(L-1)})\, V^{(L-2)}}$$

For $\delta w^{(L-2,L-3)}$ we have:

$$\frac{\partial H}{\partial w^{(L-2,L-3)}} = -(t - V^{(L)})\, g'(b^{(L)})\frac{\partial}{\partial w^{(L-2,L-3)}}\left(w^{(L,L-1)}V^{(L-1)} + w^{(L,L-2)}V^{(L-2)} - \theta^{(L)}\right)$$

$$= -(t - V^{(L)})\, g'(b^{(L)}) \left(w^{(L,L-1)} \frac{\partial V^{(L-1)}}{\partial w^{(L-2,L-3)}} + w^{(L,L-2)} \frac{\partial V^{(L-2)}}{\partial w^{(L-2,L-3)}} \right)$$

- $$\frac{\partial V^{(L-1)}}{\partial w^{(L-2,L-3)}} = g'(b^{(L-1)})\, w^{(L-1,L-2)} \frac{\partial V^{(L-2)}}{\partial w^{(L-2,L-3)}}$$

$$= g'(b^{(L-1)})\, w^{(L-1,L-2)}\, g'(b^{(L-2)})\, V^{(L-3)}$$

- $$\frac{\partial V^{(L-2)}}{\partial w^{(L-2,L-3)}} = g'(b^{(L-2)})\, V^{(L-3)}$$

Thus we have:

$$\boxed{\begin{aligned} &\therefore\ \delta w^{(L-2,L-3)} \\ &= -(t - V^{(L)})\, g'(b^{(L)}) \Big(w^{(L,L-1)} g'(b^{(L-1)})\, w^{(L-1,L-2)} g'(b^{(L-2)})\, V^{(L-3)} \\ &\qquad\qquad + w^{(L,L-2)} g'(b^{(L-2)})\, V^{(L-3)} \Big) \end{aligned}}$$

5. Binary stochastic neuron

(a) Assuming only neuron m was updated, $s_m \to s'_m$ while the other neurons remained in the same state: $s_i \to s'_i = s_i \forall i \neq m$, let us start by writing the energy H:

$$H = -\frac{1}{2}\Big(\sum_{i\neq m, j\neq m} w_{ij}s_i s_j + \sum_{i\neq m} w_{im}s_i s_m + \sum_{j\neq m} w_{mj}s_m s_j + w_{mm}s_m s_m\Big) + \sum_{i\neq m}\theta_i s_i + \theta_m s_m.$$

Now we use the symmetery of the weights, $w_{mj} = w_{jm}$, and that $w_{mm} = 0$,

$$H = -\frac{1}{2}\Big(\sum_{i\neq m, j\neq m} w_{ij}s_i s_j + 2\sum_{j\neq m} w_{mj}s_m s_j\Big) + \sum_{i\neq m}\theta_i s_i + \theta_m s_m. \tag{10}$$

Similarly, the updated energy H' is,

$$H' = -\frac{1}{2}\Big(\sum_{i\neq m, j\neq m} w_{ij}s_i s_j + \sum_{i\neq m} w_{im}s_i s'_m + \sum_{j\neq m} w_{mj}s'_m s_j + w_{mm}s'_m s'_m\Big) + \sum_{i\neq m}\theta_i s_i + \theta_m s'_m.$$

where we have used the fact that $s_i \to s'_i = s_i \forall i \neq m$. Now simpify using symmetry of weights and vanishing diagonals,

$$H' = -\frac{1}{2}\Big(\sum_{i\neq m, j\neq m} w_{ij}s_i s_j + 2\sum_{j\neq m} w_{mj}s'_m s_j\Big) + \sum_{i\neq m}\theta_i s_i + \theta_m s'_m. \tag{11}$$

Subtracting Eq. (10) from (11),

$$\Delta H = -(s'_m - s_m)\Big(\sum_{j\neq m} w_{mj}s_j - \theta_m\Big) = -b_m(s'_m - s_m). \tag{12}$$

where $w_{mm} = 0$ is used again in the last equality to write $\sum_{j\neq m} w_{mj}s_j - \theta_m = \sum_j w_{mj}s_j - \theta_m = b_m$.

(b) Here one needs to consider different cases and show that Equation (3) in the exam is always equivalent to Equation (4a) in the exam.

Case 1: $s'_m = 1, s_m = -1$

Equation (4a) gives:

$$P(-1 \to 1) = \frac{1}{1 + e^{\beta \Delta H_m}} = \frac{1}{1 + e^{-2\beta b_m}}$$

.

Equation (3) gives: $s'_m = 1$ with probability

$$p(b_m) = \frac{1}{1 + e^{-2\beta b_m}}$$

.
Case 2: $s'_m = -1, s_m = -1$.
Equation (4a): Use conservation of probability, $P(-1 \to 1) + P(-1 \to -1) = 1 \implies P(-1 \to -1) = 1 - P(-1 \to 1)$,

$$P(-1 \to -1) = 1 - \frac{1}{1 + e^{-2\beta b_m}} = \frac{1}{1 + e^{2\beta b_m}}$$

.
Equation (3) gives: $s'_m = -1$ with probability

$$1 - p(b_m) = 1 - \frac{1}{1 + e^{-2\beta b_m}} = \frac{1}{1 + e^{2\beta b_m}}$$

. **Case 3:** $s'_m = -1, s_m = 1$
Equation (4a) gives:

$$P(1 \to -1) = \frac{1}{1 + e^{\beta \Delta H_m}} = \frac{1}{1 + e^{2\beta b_m}}$$

.
Equation (3) gives: $s'_m = -1$ with probability

$$1 - p(b_m) = \frac{1}{1 + e^{2\beta b_m}}$$

. **Case 4:** $s'_m = 1, s_m = 1$ Equation (4a): Use conservation of probability, $P(1 \to -1) + P(1 \to 1) = 1 \implies P(1 \to 1) = 1 - P(1 \to -1)$,

$$P(1 \to 1) = 1 - \frac{1}{1 + e^{2\beta b_m}} = \frac{1}{1 + e^{-2\beta b_m}}$$

.
Equation (3) gives: $s'_m = 1$ with probability

$$p(b_m) = \frac{1}{1 + e^{-2\beta b_m}}$$

.
Thus, we have shown that in all 4 possible cases, the two update rules are equivalent.

6. Oja's rule

(a) We start with the given learning rule:

$$\begin{aligned}\delta \boldsymbol{w} &= \eta y(\boldsymbol{x} - y\boldsymbol{w}),\\ &= \eta(\boldsymbol{x}y - y^2\boldsymbol{w}),\\ &= \eta[\boldsymbol{x}\boldsymbol{x}^\mathsf{T}\boldsymbol{w} - (\boldsymbol{w}^\mathsf{T}\boldsymbol{x}\boldsymbol{x}^\mathsf{T}\boldsymbol{w})\boldsymbol{w}],\end{aligned}$$

Where for the first time we have written $y = \boldsymbol{w}^\mathsf{T}\boldsymbol{x} = \boldsymbol{x}^\mathsf{T}\boldsymbol{w}$, while for the second term: $y^2 = yy = \boldsymbol{w}^\mathsf{T}\boldsymbol{x}\boldsymbol{x}^\mathsf{T}\boldsymbol{w}$. Now avergaing $\delta\boldsymbol{w}$ over the data distribution,

$$\langle \delta \boldsymbol{w} \rangle = \eta[\langle \boldsymbol{x}\boldsymbol{x}^\mathsf{T}\rangle \boldsymbol{w} - (\boldsymbol{w}^\mathsf{T}\langle \boldsymbol{x}\boldsymbol{x}^\mathsf{T}\rangle\boldsymbol{w})\boldsymbol{w}].$$

Let $\mathbb{C} \equiv \langle \boldsymbol{x}\boldsymbol{x}^\mathsf{T}\rangle$, then the above equation reads,

$$\langle \delta \boldsymbol{w} \rangle = \eta[\mathbb{C}\boldsymbol{w} - (\boldsymbol{w}^\mathsf{T}\mathbb{C}\boldsymbol{w})\boldsymbol{w}].$$

Assume that $\boldsymbol{w} = \boldsymbol{w}^*$ is the normalized maximal eigenvector of the matrix $\mathbb{C}$. That is, $\mathbb{C}\boldsymbol{w}^* = \lambda_1\boldsymbol{w}^*$ where $\boldsymbol{w}^{*\mathsf{T}}\boldsymbol{w} = 1$ and λ_1 is the maximal eigenvalue. We obtain,

$$\begin{aligned}\langle \delta \boldsymbol{w} \rangle &= \eta[\mathbb{C}\boldsymbol{w}^* - (\boldsymbol{w}^{*\mathsf{T}}\mathbb{C}\boldsymbol{w}^*)\boldsymbol{w}^*],\\ &= \eta[\lambda_1\boldsymbol{w}^* - \lambda_1(\boldsymbol{w}^{*\mathsf{T}}\boldsymbol{w}^*)\boldsymbol{w}^*],\\ &= \eta[\lambda_1\boldsymbol{w}^* - \lambda_1\boldsymbol{w}^*],\\ &= 0.\end{aligned}$$

Thus we have shown that the normalized maximal eigenvector $\boldsymbol{w}^*$ of $\mathbb{C}$ is a steady state of the given learning rule.