CSE 5526: Homework 4 Solution

Problem 1. For the first step,

$$y_1(1) = \varphi(4\cdot 0.2 – 1.2) = 0$$
$$y_2(1) = \varphi(4\cdot 0.2 – 1.2) = 0$$
$$y_3(1) = \varphi(4\cdot 0.3 – 1.1) = 0.1$$
$$y_4(1) = \varphi(4\cdot 0.4 – 1.0) = 0.6$$
$$y_5(1) = \varphi(4\cdot 0.3 – 1.1) = 0.1$$

Therefore,

$$\mathbf{y}^T(1) = [0, 0, 0.1, 0.6, 0.1]$$

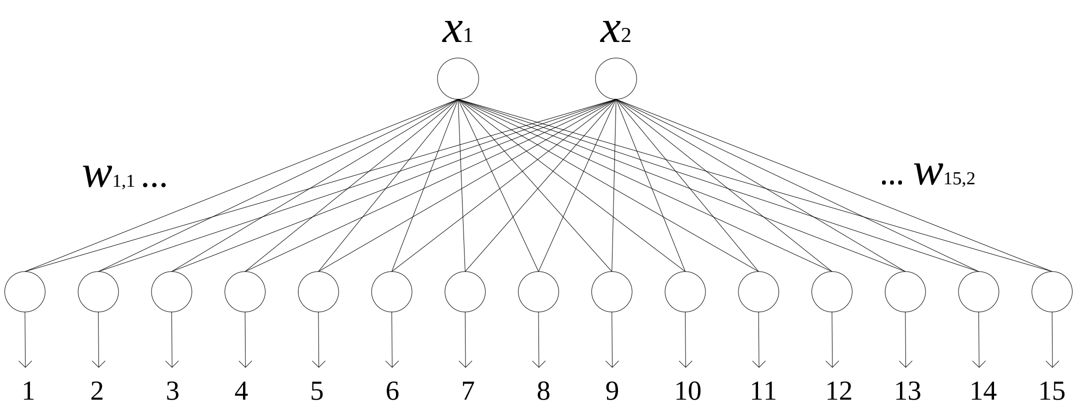
For the second step,

$$y_1(2) = \varphi(4\cdot 0 – 0.8) = 0$$
$$y_2(2) = \varphi(4\cdot 0 – 0.8) = 0$$
$$y_3(2) = \varphi(4\cdot 0.1 – 0.7) = 0$$
$$y_4(2) = \varphi(4\cdot 0.6 – 0.2) = 1$$
$$y_5(2) = \varphi(4\cdot 0.1 – 0.7) = 0$$

Therefore,

$$\mathbf{y}^T(2) = [0, 0, 0, 1, 0]$$

Problem 2. There are 15 neurons arranged as a 1-D (linear) layer. For more details see lecture notes.



Problem 3.

(a) We are given the fundamental memories:

$\xi_1 = [+1, +1, +1, +1, +1]^T$
$\xi_2 = [+1, -1, -1, +1, -1]^T$
$\xi_3 = [-1, +1, -1, +1, +1]^T$

The weight matrix of the Hopfield network (with N = 25 and p = 3) is therefore

$$\mathbf{W} = \frac{1}{N}\sum_{i=1}^{M} \boldsymbol{\xi}_i \boldsymbol{\xi}_i^T$$
$$= \frac{1}{5}\begin{bmatrix} +3 & -1 & +1 & +1 & -1 \\ -1 & +3 & +1 & +1 & +3 \\ +1 & +1 & +3 & -1 & +1 \\ +1 & +1 & -1 & +3 & +1 \\ -1 & +3 & +1 & +1 & +3 \end{bmatrix}$$

(b) According to the alignment condition, we write

$\xi_i = \varphi(\,\mathbf{W}\xi_i\,), \qquad i = 1, 2, 3$

Consider first $\boldsymbol{\xi}_1$, for which we have

$$\varphi(\mathbf{W}\boldsymbol{\xi}_1) = \varphi\left(\frac{1}{5}\begin{bmatrix} +3 & -1 & +1 & +1 & -1 \\ -1 & +3 & +1 & +1 & +3 \\ +1 & +1 & +3 & -1 & +1 \\ +1 & +1 & -1 & +3 & +1 \\ -1 & +3 & +1 & +1 & +3 \end{bmatrix}\begin{bmatrix} +1 \\ +1 \\ +1 \\ +1 \\ +1 \end{bmatrix}\right)$$
$$= \varphi\left(\frac{1}{5}\begin{bmatrix} 3 \\ 7 \\ 5 \\ 5 \\ 7 \end{bmatrix}\right) = \begin{bmatrix} +1 \\ +1 \\ +1 \\ +1 \\ +1 \end{bmatrix} = \boldsymbol{\xi}_1$$

$$\varphi(\mathbf{W}\boldsymbol{\xi}_2) = \varphi\left(\frac{1}{5}\begin{bmatrix} +3 & -1 & +1 & +1 & -1 \\ -1 & +3 & +1 & +1 & +3 \\ +1 & +1 & +3 & -1 & +1 \\ +1 & +1 & -1 & +3 & +1 \\ -1 & +3 & +1 & +1 & +3 \end{bmatrix}\begin{bmatrix} +1 \\ -1 \\ -1 \\ +1 \\ -1 \end{bmatrix}\right)$$

$$= \varphi\left(\frac{1}{5}\begin{bmatrix} 5 \\ -7 \\ -5 \\ 3 \\ -7 \end{bmatrix}\right) = \begin{bmatrix} +1 \\ -1 \\ -1 \\ +1 \\ -1 \end{bmatrix} = \boldsymbol{\xi}_2$$

$$\varphi(\mathbf{W}\boldsymbol{\xi}_3) = \varphi\left(\frac{1}{5}\begin{bmatrix} +3 & -1 & +1 & +1 & -1 \\ -1 & +3 & +1 & +1 & +3 \\ +1 & +1 & +3 & -1 & +1 \\ +1 & +1 & -1 & +3 & +1 \\ -1 & +3 & +1 & +1 & +3 \end{bmatrix}\begin{bmatrix} -1 \\ +1 \\ -1 \\ +1 \\ +1 \end{bmatrix}\right)$$

$$= \varphi\left(\frac{1}{5}\begin{bmatrix} -5 \\ 7 \\ -3 \\ 5 \\ 7 \end{bmatrix}\right) = \begin{bmatrix} -1 \\ +1 \\ -1 \\ +1 \\ +1 \end{bmatrix} = \boldsymbol{\xi}_3$$

Thus all three fundamental memories satisfy the alignment condition.

(c) Consider the input:

$\mathbf{x}$ = [+1, -1, +1, +1, +1]^T

which is the fundamental memory $\boldsymbol{\xi}_1$ with its second element reversed in polarity. We write

$$\mathbf{Wx} = \frac{1}{5}\begin{bmatrix} +3 & -1 & +1 & +1 & -1 \\ -1 & +3 & +1 & +1 & +3 \\ +1 & +1 & +3 & -1 & +1 \\ +1 & +1 & -1 & +3 & +1 \\ -1 & +3 & +1 & +1 & +3 \end{bmatrix}\begin{bmatrix} +1 \\ -1 \\ +1 \\ +1 \\ +1 \end{bmatrix}$$

$$= \frac{1}{5}\begin{bmatrix} +1 \\ -1 \\ +1 \\ +1 \\ +1 \end{bmatrix}$$

Therefore,

$$\varphi(\mathbf{Wx}) = \frac{1}{5}\begin{bmatrix} +1 \\ +1 \\ +1 \\ +1 \\ +1 \end{bmatrix}$$

Thus neuron 2 wants to change its state, which yields the result.

$\mathbf{x} = [+1, +1, +1, +1, +1]^T$

This vector is the fundamental memory $\boldsymbol{\xi}_1$, and the computation converges. Thus, when the noisy version of $\boldsymbol{\xi}_1$ is applied to the network, the original $\boldsymbol{\xi}_1$ is recovered after 1 iteration by the Hopfield network.

(d)

$$E = -\frac{1}{2}\sum_i \sum_j w_{ji} x_i x_j$$

$$= -\frac{1}{10}(3x_1^2 - 2x_1x_2 + 2x_1x_3 + 2x_1x_4 - 2x_1x_5 + 3x_2^2 + 2x_2x_3 + 2x_2x_4 + 6x_2x_5 + 3x_3^2 - 2x_3x_4 + 2x_3x_5 + 3x_4^2 + 2x_4x_5 + 3x_5^2)$$

$$= -\frac{3}{10}\sum_{i=1}^{5} x_i^2 + \frac{1}{5}(x_1x_2 - x_1x_3 - x_1x_4 + x_1x_5 - x_2x_3 - x_2x_4 - 3x_2x_5 + x_3x_4 - x_3x_5 - x_4x_5)$$