HUMBOLDT-UNIVERSITÄT ZU BERLIN
BERNSTEIN CENTER FOR COMPUTATIONAL NEUROSCIENCE

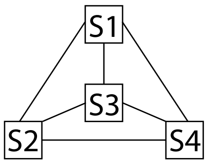
Humboldt-Universität zu Berlin
Philippstr. 13 House 4
Prof. Richard Kempter
Prof. Benjamin Lindner

Phone: 030/2093-9110
Fax: 030/2093-6771
Webpage: http://www.bccn-berlin.de/

Models of Neural Systems, WS 2020/21
Computer Practical 2
Solution due on November, 30th, 2020, at 10 am

Hopfield Network

Hopfield networks are an example of recurrent neural networks with binary or bipolar threshold units. They are used as a model for associative memory. In this exercise, you will investigate how given patterns can be memorized by a Hopfield network, by appropriately choosing its weights through a *learning rule*. As a toy example, consider the following Hopfield network with 4 bipolar units:



The state of neuron i is denoted with s_i. The connection weights between two units i and j are equal to $w_{i,j}$. The input to unit i is $z_i = \sum_j w_{ij} s_j$. The units have the sign function as activation function:

$$\Theta(z_i) = \begin{cases} 1 & \text{if } z_i \geq 0 \\ -1 & \text{if } z_i < 0 \end{cases} \tag{1}$$

Weight Initialization
Hebb's rule states that a set of patterns $\{S^K\}$ can be stored in the network by setting the weight between units i and j to

$$w_{ij} = w_{ji} = \sum_K s_i^K s_j^K. \tag{2}$$

No unit has a connection with itself (self-connection), so $w_{ii} = 0,\ \forall i$.

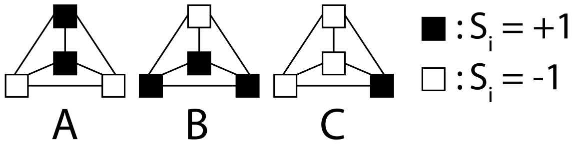
State update
Here we assume an asynchronous update of the units. At each time step we randomly select a unit i to be updated according to

$$s_i' = \Theta\left(\sum_j w_{ij} s_j\right) \tag{3}$$

Successfully stored patterns are stable states of the network, e.g., a network in such a state should not be changed by the update rule above.

Tasks:

1. Store the following three patterns A, B, and C in the Hopfield network.



   What are the resulting weights? Which of the patterns are stable states of the network dynamics? Which patterns do the unstable states converge to? (Hint: Apply the stored pattern as input and determine if they persist to the next iterations.)

2. Consider the following energy function for the network:

$$E = -\sum_i \sum_j w_{ij} s_i s_j. \tag{4}$$

   Calculate the energy after each update step and plot it as a function of time. (Hint: The energy should decrease after each update or remain the same in case of no update.)

3. Finally, you will reuse the code developed for the toy example above to store and recall image patterns in a larger Hopfield network.

   1. Load the numpy file 'images.npz' in ipython. (Hint: np.load() will return a 3D array with dimensions (k,v,h), where k is the number of patterns, v and h are the vertical and horizontal dimensions of the image.)

   2. Apply equation 2 to store the patterns into a weight matrix $W = \{w_{ij}\}$. (Hint(s): You will have to flatten the images from 2 dimensions into vectors. Applying matrix multiplication is always preferable than iteration over elements.)

   3. Which patterns are (un)stable?

   4. Start with a pattern consisting of random values ± 1 and update according to the rule in Equation 3 until reaching a stable state. Plot the initial pattern, the final stable state of the network, and the energy function over iterations. Repeat this step several times with different random initial patterns. (Hint: Before plotting you should reshape the patterns from 1D arrays into 2D arrays using the method reshape((v,h)). For plotting, you can use pyplot.imshow(M), where M is a 2D array.)

   5. As in the last task, create a random vector of length n. Copy a part of one memory (e.g., 25%) and fill it into the random vector. The created vector can be used as a cue to recall content-addressable memory. Use it as an initial pattern and update the network repeatedly until reaching a stable state. Plot the initial and final patterns and the energy function.

Contact	Location	Phone	Email
Gregory Knoll	Haus 2	2093 6247	gregory@bccn-berlin.de
Paula Kuokkanen	ITB, Haus 4, Room 107	2093 98407	paula.kuokkanen@hu-berlin.de
Eric Reifenstein	ITB, Haus 4, Room 013	2093 98413	eric.reifenstein@bccn-berlin.de
Natalie Schieferstein	ITB, Haus 4, Room 106	2093 98406	natalie.schieferstein@bccn-berlin.de