Institute for Computer Science VI, Autonomous Intelligent Systems, University of Bonn

Dr. N. Goerke
Friedrich-Hirzebruch-Allee 8, 53115 Bonn, Tel: +49 228 73-4167
E-Mail: goerke (at) ais.uni-bonn.de
http://www.ais.uni-bonn.de/WS2223/4204_L_NN.html

Exercises for module
Technical Neural Networks (MA-INF 4204), WS22/23

Assignments Sheet 11, due: Monday 16.01.2023

9.1.2023

Group	Name	67	68	69	70	71	72	73	74	75	$\sum$ Sheet 11

Assignment 67 (1 Point)

Count the points that you have gained so far in your assignments (including Programming Assignments) calculate the difference to the threshold necessary to be admitted to the exam, check if you have presented twice in the exercises, and tell what you still need to be admitted to the exam.

Assignment 68 (2 Points)

The method of unfolding in time yields different weight changes for different time steps.
Describe why this can be a problem when training the recurrent network.
Name (not describe) a method to cope with that problem.

Assignment 69 (1 Point)

Describe a case where an *Elman Network* is a *Jordan Network*.

Assignment 72 (2 Points)

Assignment 70 (2 Points)

Find an application for a recurrent MLP, and describe it in your own words.
Cite the publication you have found in a scientific way.

Assignment 71 (2 Points)

"Hopfield Networks can not distinguish between a pattern and it's inverse pattern".
Explain what the sentence is referring to.

Assignment 72 (2 Points)

Calculate the memory consumption of a Hopfield Network that consists of $K = 1000$ neurons, when the weights $w_{i,j}$ are stored as integer values.
Hint: please remind, that the maximal number of patterns that can be stored and recalled by a Hopfield Network is bounded, and thus, that the range of the weights should be reasonably bounded as well.

Assignment 73 (2 Points)

Explain the differences for the learning rule, when the Hopfield Network is trained in mode *heteroassociator* compared to a training in mode *autoassociator*.

Assignment 74 (3 Points)

Show that the energy E in a Hopfield Network (autoassociator) is never increasing.

$$E = -\frac{1}{2}\sum_{i,j} w_{i,j} x_i x_j + \sum_k x_k \Theta_k$$

Assume an asyncronous update, symmetric weights $w_{i,j} = w_{j,i}$, no self feedback $w_{i,i} = 0$, and the transfer function:

$$x_i(t+1) = \begin{cases} +1 & : \quad \text{if} \quad \sum_j w_{i,j} x_j > \Theta_i \\ -1 & : \quad \text{else} \end{cases}$$

Assignment 75 (2 Points)

Calculate the weights $w_{i,j}$ for a Hopfield Network that has been trained (as autoassociator) with the four patterns (I,II,III,IV).

I	=	(	-1,	-1,	-1,	-1,	+1,	+1,	+1,	+1	)
II	=	(	+1,	-1,	+1,	-1,	+1,	-1,	+1,	-1	)
III	=	(	+1,	+1,	-1,	-1,	+1,	+1,	-1,	-1	)
IV	=	(	+1,	+1,	+1,	-1,	-1,	-1,	-1,	+1	)

Programming assignment PA-H (5 Points, Due date: Mon 16.01.2023)

Implement a Hopfield Network in Python.

Hopfield Network:
Implement a Hopfield Network with up to $K = 1000$ neurons.
Use integer weights $w_{i,j}$. Set all thresholds Θ_k identical to a user defined value, or set all thresholds to be the starting pattern $\Theta_k = x_k(t = 0)$, or set all thresholds to be zero.
Implement two operating modes for the network: a learning mode (autoassociator), and a recall mode (asyncronous update).

For small networks ($K < 101$) the network state in every recall time step shall be depicted as *ASCII-art* console output, one ASCII character per neuron (e.g. {`O`,`#`} or {`+`,`-`}), one line per timestep.
For larger networks ($K > 100$) the energy value E shall be printed in every recall timestep and depicted in a diagram.

Take your own patterns to test your Hopfield network implementation. Describe or plot your patterns, and describe the obtained results.