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

Dr. N. Goerke
Endenicher Allee 19a, 53115 Bonn, Tel: +49 228 73-4167
E-Mail: goerke_at_ais.uni-bonn.de
http://www.ais.uni-bonn.de/WS1920/4204_L_NN.html

New Lecture Location: Meckenheimer Allee 176, HS-IV

Exercises for module Technical Neural Networks (MA-INF 4204), WS1920

Exercises sheet 7, due: Monday 25.11.2019

18.11.2019

Group	Name	38	39	40	41	42	43	44	$\sum$ Sheet 7

Assignment 38 (2 Points)

Name and describe at least two advantages of the *Multi*-variants (Multi-SOM, Multi-N-GAS) compared to the classical approaches SOM, and Neural-Gas.

Assignment 39 (2 Points)

What could be called a "good" clustering ?
Define a criterium that should be met for a "good" clustering of a given set of data points.

Please argue using the *Silhouette Coefficient*, or the *Calinsky and Harabasz* criterium or the *gap statistics*.

Assignment 40 (4 Points)

The learning rules of the neural paradigms *SOM, Neural Gas, Multi-SOM, Multi Neural Gas, LVQ1, LVQ2.1* have a lot in common. Only the way how the neighborhood function is operating to calculate the change of the centers $\Delta \mathbf{C}_j$ is different. Describe these differences, and write down the respective learning rules as formulas.

Assignment 41 (1 Point)

Describe a **method** to find a quantization that can classify points $\mathbf{X} \in \mathcal{U}$ from the 8-dimensional unit cube $\mathcal{U}$ into 6561 uniform classes.

Assignment 42 (1 Point)

Derive a formula, that calulates the total number F of centers for a Multi-SOM with input dimension N, with M Partner-SOMs, each one having a g-dimensional, rectangular grid, with an edge-length $f_{m,1}, f_{m,2}, \ldots, f_{m,g}$.
Calculate F using your formula for $N = 17$, $M = 7$, $g = 2$ and $f_{m,1} = 2, f_{m,2} = 3$.

Assignment 43 (3 Points)

Re-write the adaptation rule of **k-means-clustering** in such a way, that it will fit into the general learning scheme proposed for SOM, Multi-SOM, Neural Gas, LVQ, ... within the lecture (see Assignment 40). A possible way could be to re-define the properties of h(i,j).

Assignment 44 (2 Points)

Please work on this assignment on your own. For the assignemnt 44 **no teamwork** is allowed. On the exercises website **http://www.ais.uni-bonn.de/WS1920/4204/4204_E_NN.html** you will find two files containing 20 coloured pictures each. **Pics_A.pdf** with 20 drawings, and **Pics_B.pdf** with 20 pictures of trees.

First you have to decicde what set of pictures you would like to work with. Then, please have a brief look at the pictures and try to classify the pictures spontaneously into exactly two classes. Describe the criterium you have used for classification.
What other possibilities to classify the pictures can you think of?
Please name at least two further ways to classify the pictures.

Remark: Since the classification into two classes is arbitrary, and only depends on you, there is no unique solution in classifying these pictures and thus, you can't do wrong.

Programming assignment PA-D (10 Points, Due date: Mon 2.12.2019)

Implement a Multi-SOM or a Multi-Neural-Gas (your coice) in C, C++, Java or Python, and use an example to demonstrate that your program is working correctly.

Network: M-SOM or M-Neural-Gas

The M-SOM, or M-Neural-Gas shall have a total of K Neurons, and M partner networks. For the M-SOM the dimension g of the grid $\mathcal{G}$ is resticted to $g \leq 5$, and shall be rectangular, with different edge lengths $f_1, f_2, \ldots, f_g$.
The input dimension N (restricted to $N < 7$), the structure, and size of the partner networks and the number of neurons K shall be adjustable (it is O.K. to do this in the source code)
To initialize the centers $\mathbf{C}_j$ either randomly from the unit cube, or as as random drawn subset of the training patterns ${}^p\mathbf{X}$. Take the Gaussian for the neighborhood function $h(dist(i,j),t)$ with adjustable but fixed size s; implement an exponentially decaying learning rate $\eta = \eta(t)$ decaying from $\eta_0 = \eta(t=0)$ to an end value η_{end}.

A program that trains the M-SOM, or M-Neural-Gas:

The parameters of the net M, N, g, K, the structure of the grid $(f_1, ..., f_g)$, the training patterns ${}^p\mathbf{X}$ the learning rate parameters η_0, η_{end} and the width of the gaussians shall be adjustable within the program (not necessarily at runtime). The program shall read in the training patterns either from the file `PA-D-train.dat` or shall be generated randomly from two circular, non overlapping areas within the unit cube.

Demonstrate that your program works properly by using a $M = 4$ network structure with $N = 2, g = 2$ to learn patterns drawn randomly, equally distributed from 3 non-overlapping areas within the unit cube. Please specify the boundaries of the areas you have used.

Write the found positions of the centers (one center per line, both components) of your network into a file `PA-D.net`.