Artificial Neural Networks

Self-Organizing Maps (Kohonen's map)

Ing. Tomas Borovicka

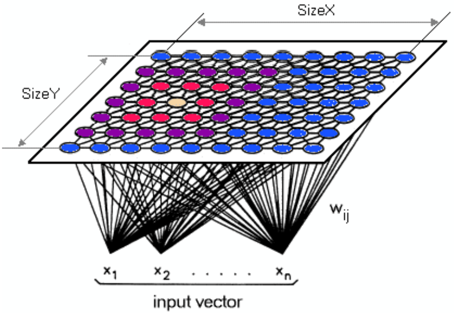
Department of Theoretical Computer Science (KTI), Faculty of Information Technology (FIT)
Czech Technical University in Prague (CVUT)

BIE-ZUM, LS 2013/14, 11. lecture

https://edux.fit.cvut.cz/courses/BIE-ZUM/

Kohonen map

- Neurobiologically inspired.
- Principle of topographic map formation of brain.
- Unsupervised technique.
- Reduces high dimensional data into 2D map.
- Neurons in lattice not connected together.
- Input layer is fully connected with the neuron lattice.



Learning process

1. **Initialization**

   Synaptic weights are initialized randomly in a given range.

2. **Competition**

   For the input vector, each neuron compute its matching criterion and the best-matching neuron is referred as a winner, also called Best-Matching-Unit (BMU).

3. **Cooperation**

   The best-matching neuron determines the radius of its neighborhood for cooperation. This radius is typically relatively large in early iterations and shrinks in each step.

4. **Adaptation**

   The synaptic weights of the BMU and cooperating neurons are adapted to become more similar to the input vector. The most adjusted are weights of the BMU, the further neurons are the less are the weights affected.

5. Steps 2-4 are repeated until stopping criterion is not fulfilled.

Initialization

Let w^k be the synaptic weight vector of the k-th neuron ($k = width * i + j$), formally

$$w^k = [w_1^k, \dots, w_J^k]$$

Apparently dimension of the vector w is the same as for the input vector x.
First of the learning process all weights have to be initialized. Weights are usually set randomly in a positive range

$$0 < w_j < \theta$$

where θ is typically equal to 1.

Competition

Having an input vector $x = [x_1, \dots, x_J]$ from the training set S. We want to determine the neuron k^* which minimizes the difference between input vector x and vector of synaptic weights w

$$k^* = \arg\min_k ||x - w^k||$$

Generally used similarity measure is Euclidean distance, then

Best-Matching-Unit

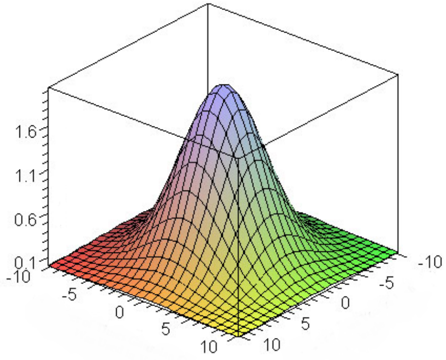
$$k^* = \arg\min_k \sqrt{\sum_j (x_j - w_j^k)^2}$$

Neuron that satisfied this condition is the winner (BMU) of the competition.

Cooperation

Neighborhood function, typically Gaussian

$$f(x) = \alpha e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$



Lateral distance function

$$\Delta(\nu_1, \nu_2)^2 = (i_{\nu 1} - i_{\nu 2})^2 + (j_{\nu 1} - j_{\nu 2})^2,$$

Time varying radius function σ

$$\sigma(t) = \sigma_0 e^{\left(\frac{t}{\tau_\sigma}\right)} \qquad t = 1, 2, 3 \ldots$$

Neighborhood function

$$N_{\nu_{BMU}, \nu_k}(t) = e^{\left(-\frac{\Delta(\nu_{BMU}, \nu_k)^2}{2\sigma(t)^2}\right)}$$

Adaptation

Synaptic weights of BMU and its neighborhood are adjusted to move closer towards the input vector. Synaptic weight vector of neuron ν_k in discrete time t is given as

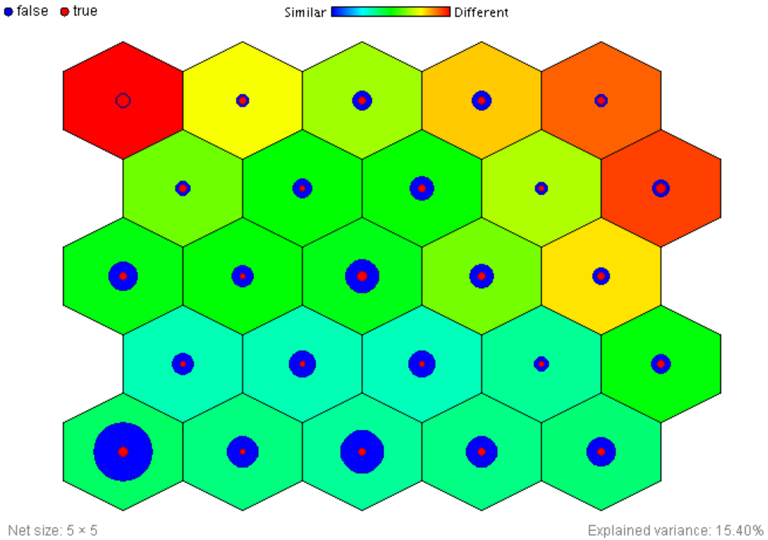
Weight function

$$w_k(t) = w_k(t-1) + \eta(t) N_{BMU}(t)[x - w_k(t-1)],$$

where $\eta(t)$ is time varying learning rate defined as

$$\eta(t) = \eta_0 e^{\left(-\frac{t}{\tau\eta}\right)}.$$

Experiments



Experiments