WARSAW UNIVERSITY OF TECHNOLOGY
FACULTY OF MATHEMATICS
AND INFORMATION SCIENCE

Neural Networks

Lecture 7

B.Macukow

Kohonen Self - Organizing Networks

Kohonen Networks

In the previous part we discussed ***a supervised learning*** techniques based among others on the back propagation technique. Now we will speak about ***unsupervised learning methods***, and in particular Kohonen's self organizing maps.

The idea of self - organization was proposed in 1973 by von der Malsburg and was based on close studies of the topology of the brain's cortex region.

Kohonen Networks

It is well known that learning or adaptation is the chemical process changing the effectiveness of the synaptic connections at the cell input.

The self - organization network has two main assumptions:

- the input patterns that share common features belong to the same class,
- the network will be able to identify common features across the range of input patterns.

Kohonen Networks

Kohonen used the idea that the brain uses spatial mapping to model complex data structures internally.

It allows him to perform data compression on the vectors to be stored in the network, using a technique known as vector quantization.

Data compression means that multi - dimensional data can be represented in a much lower dimensional space. The implementation of Kohonen's is two - dimensional.

Kohonen Networks

The perceptron was the network model where neurons were acting independently. Now, we will speak about system performing *feature maps*), generalizing the self organizing process by means of geometrical organization mutually competing cells.

Kohonen Networks

Two types of a learning by competition:

winner-takes-all

or

winner-takes-most

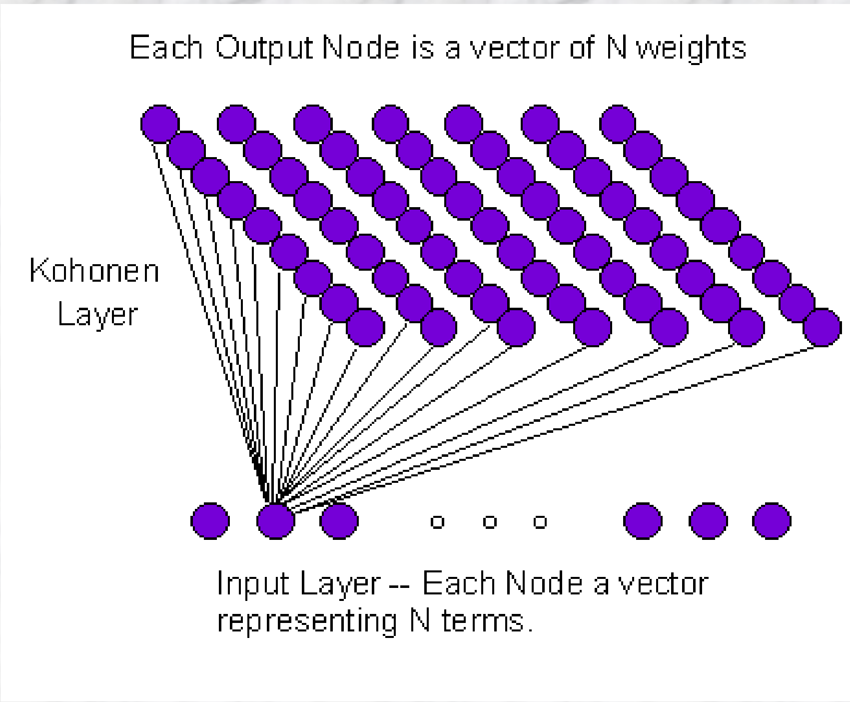
Kohonen Networks

Kohonen uses the idea that some regions in a brain are responsible for certain functions – i.e. used the von der Malsburg model
The part of a one-layer, two-dimensional Kohonen network with the connections of between only two input elements and the neurons are shown (in practice all input are connected to all nodes).

Kohonen Networks

The neurons are not arranged in layers (as in the multilayer perceptron) but in the flat grid. Feedback is restricted to lateral interconnections to intermediate neighboring nodes

Note also, that there is no special output layer - each of the elements is itself and output element.

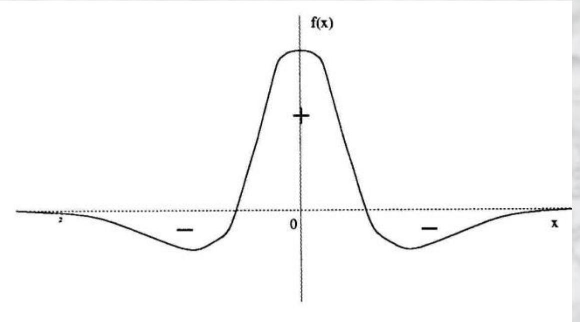


Kohonen Networks

We have seen from the previous lectures that activation in a nervous cell is propagated to other cells via axon links – which may have an inhibitory or excitatory effect at the input of another cell. However, we have not considered the question of how the axon links are affected by lateral distance from propagating neuron.

Kohonen Networks

A simplified model of the effect is illustrated by the *Mexican hat* function



Cells physically close to the active cell have strongest links. Those of certain distance even switch to inhibitory links. Kohonen modeled this effect by using only locally interconnected networks and restricting the adaptation the weights values to localized ”neighborhoods”.

Kohonen Networks

The other Kohonens' assumptions !!

- input signals are normalized (i.e. $|X| = 1$)
- the element to be learned is selected by the special rule (see algorithm),
- the weights of elements in the neighborhood of this selected neuron are also modified

Kohonen Networks

The idea of neighborhood is introduced and defined.

The examples of neighborhood:

- 1-D chain with numbered elements
- 2-D array with elements indexed by a(i,j).

The location (distance) of neighbor elements differs less then a certain value.

Kohonen Networks

For 1-D the function defining the distance between the *i*th and *j*th element

$$\boldsymbol{h(i,j)} = \begin{cases} 1 & \text{dla} \quad i = j \\ \frac{1}{2} & \text{dla} \quad |i - j| = 1 \\ 0 & \text{dla} \quad |i - j| > 1 \end{cases}$$

or $\quad \boldsymbol{h(i,j)} = 1/\rho(i,j)$

where $\rho(i,j)$ is the distance between elements

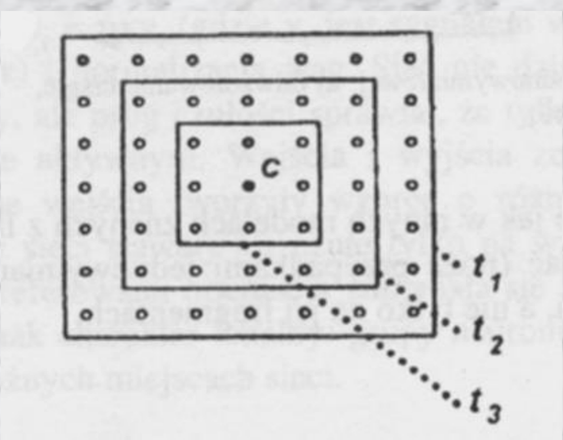
or $\quad \boldsymbol{h(i,j)} = \exp[-\rho^2(i,j)].$

Kohonen Networks

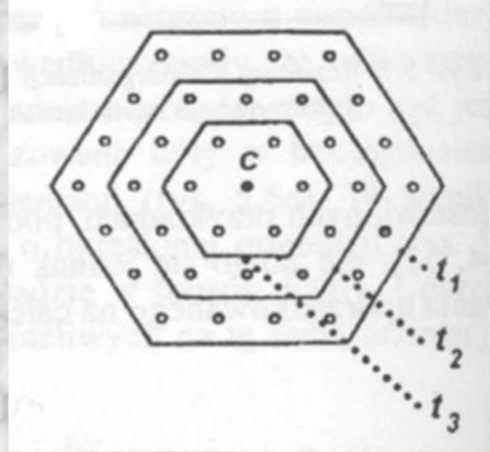
Two proposals of neighborhood:
a) rectangular grid
b) hexagonal grid

Kohonen introduce the idea that number of neurons surrounding the winning node and size of the neighborhood is reduced with time during the training sequence to its final size.

Kohonen Networks



rectangular grid



hexagonal grid

Kohonen Networks

Algorithm description

At the beginning all weights are set to be usually small random values. Each node have the unique weight vector, the dimensionality of which is defined by the number of components in the input vector. During the learning cycle, a set of training patterns is shown tom the network. Comparison is made between each input pattern and the weight vectors.

Kohonen Networks

The node with the weight vector closest to the input pattern is selected to as the „winner”. This node modifies its own weight vector to align with the input. The node has now become more sensitive to the particular training input and will provide maximum response from the network if this input is applied again.

Also the nodes in the neighborhood of the winning node are also modified.

Kohonen Networks

The network is trying to create regions that will respond to a spread of values around the training input. The nodes around are given similar alignment.

As the result, vectors that are close spatially to the training values will still be classified correctly. This demonstrates generalization properties of the network.

Kohonen Networks

The change of the connection weight between the j^{th} input element and the i^{th} Kohonen element (in the time t) is defined by

$$\boldsymbol{w_{ij}(t+1) = w_{ij}(t) + \eta(t)h(i,j)[x_i(t) - w_{ij}(t)]}$$

where $w_{ij}(t)$ is the connection weight of the j^{th} input with the i^{th} neuron

η is the learning rate coefficient

Kohonen Networks

The learning rate coefficient η (unit of proportionality) decreases the adaption rate with time (where „time” means the number of passes through the training set.

The training process attempts to cluster the nodes on the topological map to reflect the range of class types found in the training data. At the beginning the adaptation is kept high (>0.5) and is reduced as training progresses. Typically fine tuning stage will take between 100 and 1000 times as many steps as finding the coarse representation.

Kohonen Networks

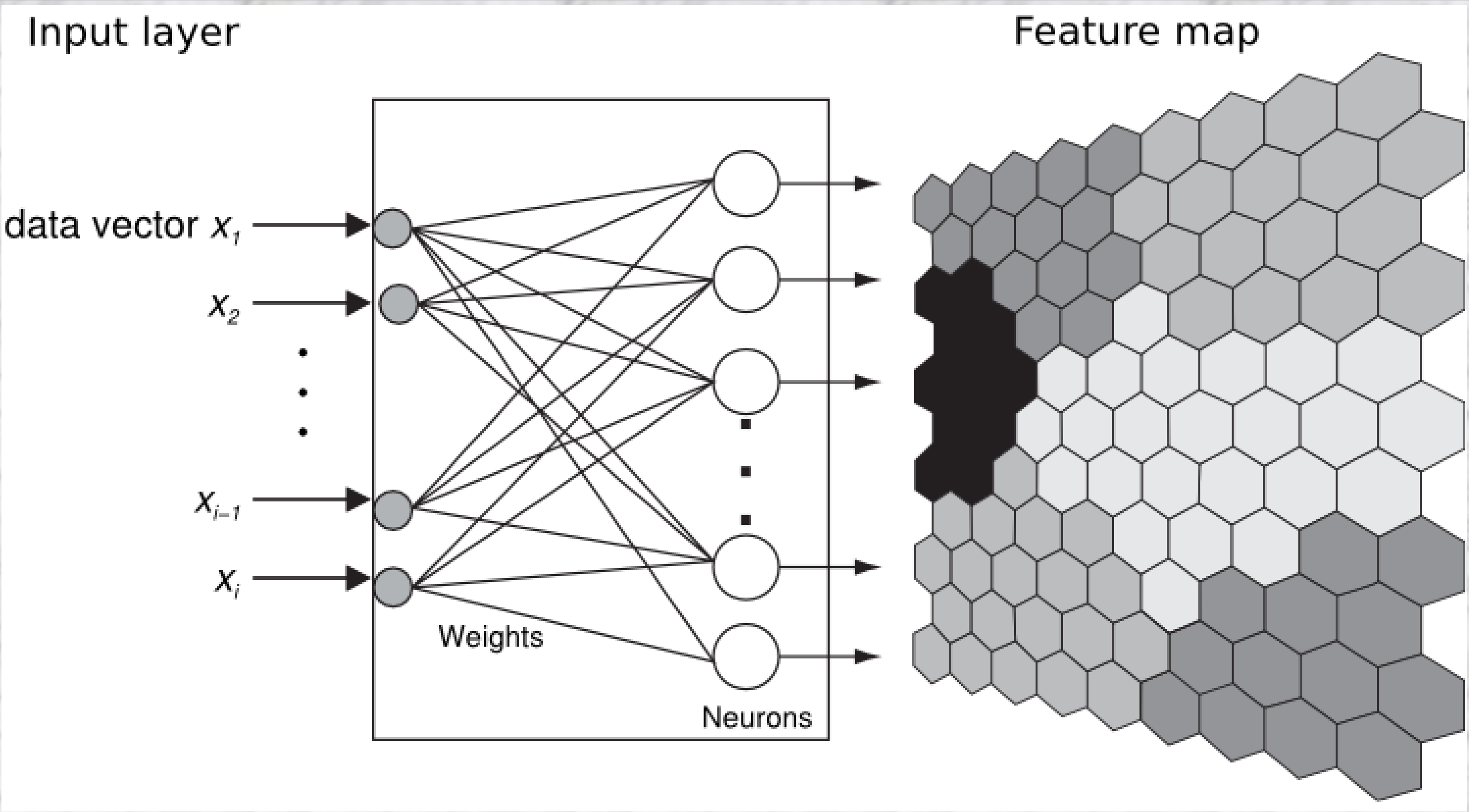
The training algorithm will produce clusters for all the class types found in the training data. The ordering of the clusters on the map and the convergence tomes for training are dependent on the way the training data are presented to the network.

Once the network has self-organized the internal organization the clusters can be labeled to indicate their class so that the network can be used to classify unknown inputs.

Kohonen Networks

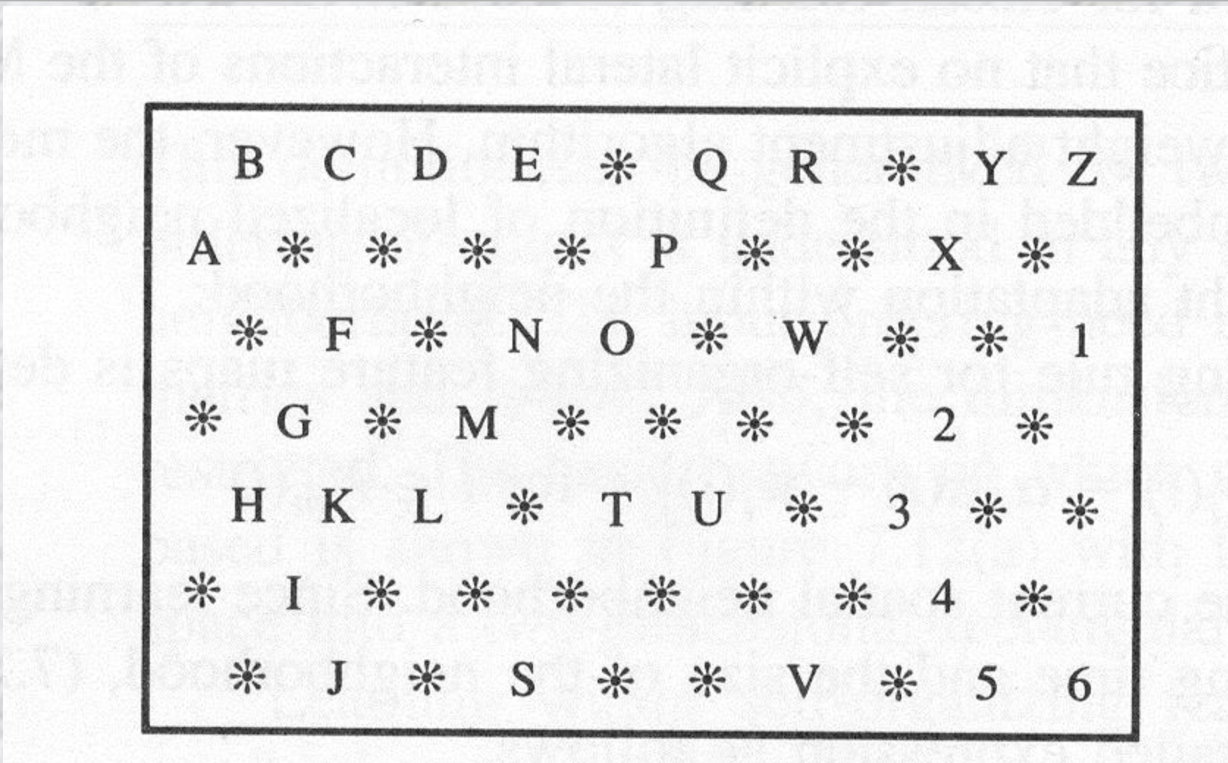
The network forms the internal features without supervision, but the classification labeling must be done by hand, once the network is fully trained.

Kohonen Networks



Feature map produced after training

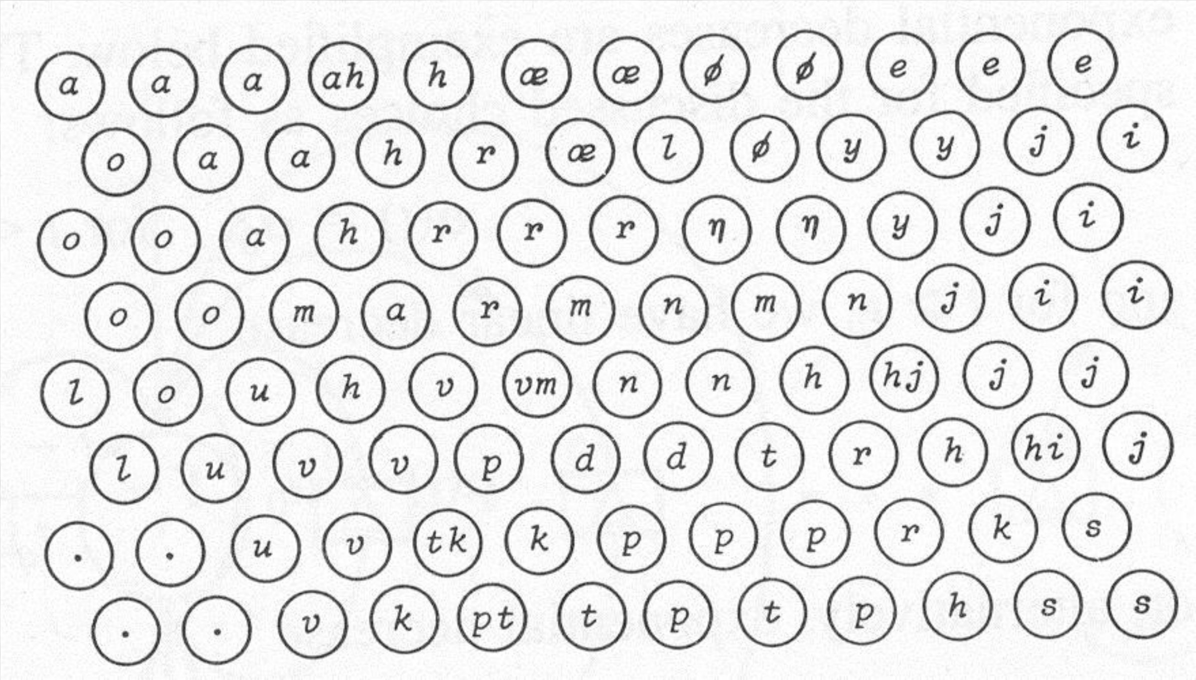
Kohonen Networks



Feature map produced after training

Kohonen Networks

A phoneme typological feature map. The network was trained on voice data (Finish language)



Kohonen Networks

A characters’ recognition typological feature map.