Self-Organizing Maps (SOM)

Dr. Saed Sayad

University of Toronto

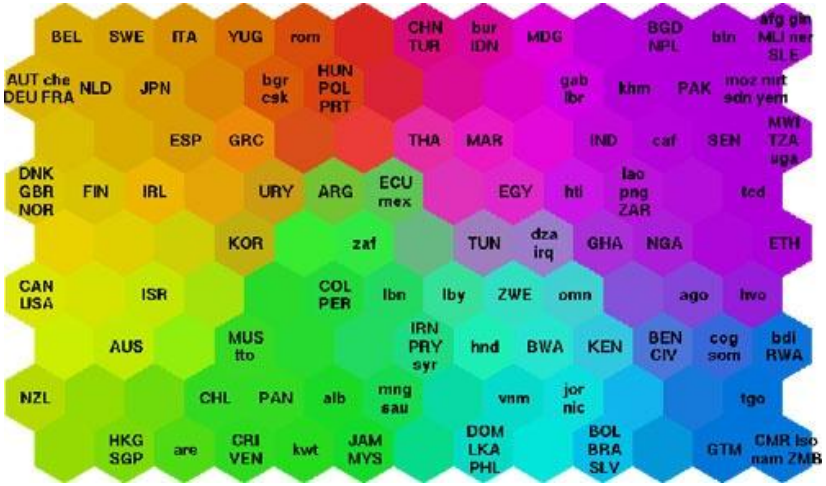
2010

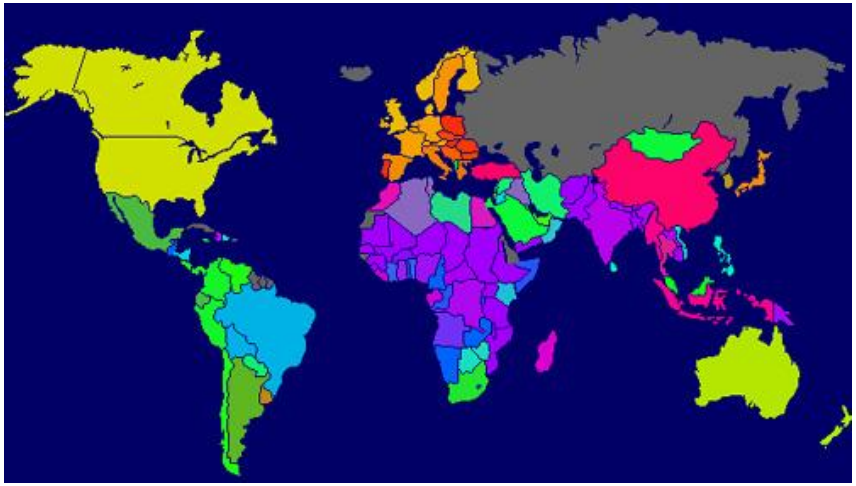
saed.sayad@utoronto.ca

Overview

- Self-Organizing Map (SOM) is an unsupervised learning algorithm.
- SOM is a visualization method to represent higher dimensional data in an usually 1-D, 2-D or 3-D manner.
- SOMs have two phases:
  - Learning phase: map is built, network organizes using a competitive process using training set.
  - Prediction phase: new vectors are quickly given a location on the converged map, easily classifying or categorizing the new data.

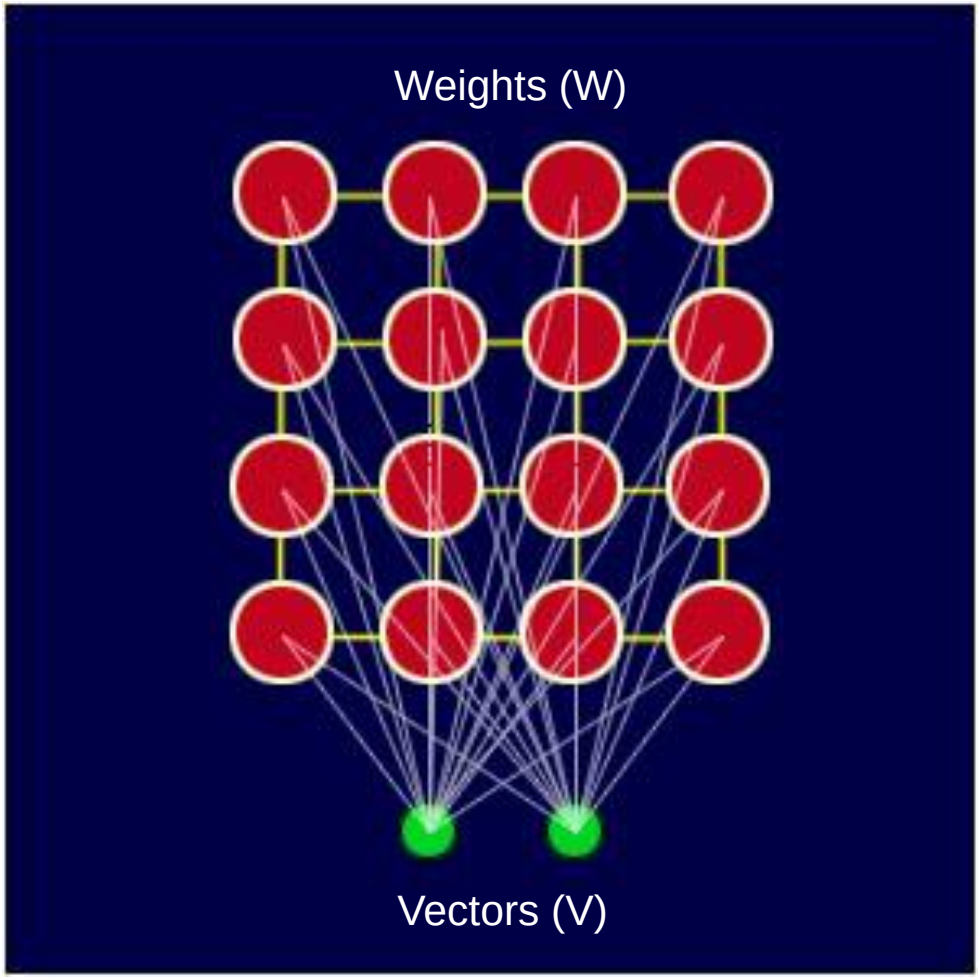
SOM: Example





- Example: Data sets for poverty levels in different countries.
  - Data sets have many different statistics for each country.
  - SOM does not show poverty levels, rather it shows how similar the poverty sets for different countries are to each other.

SOM Structure

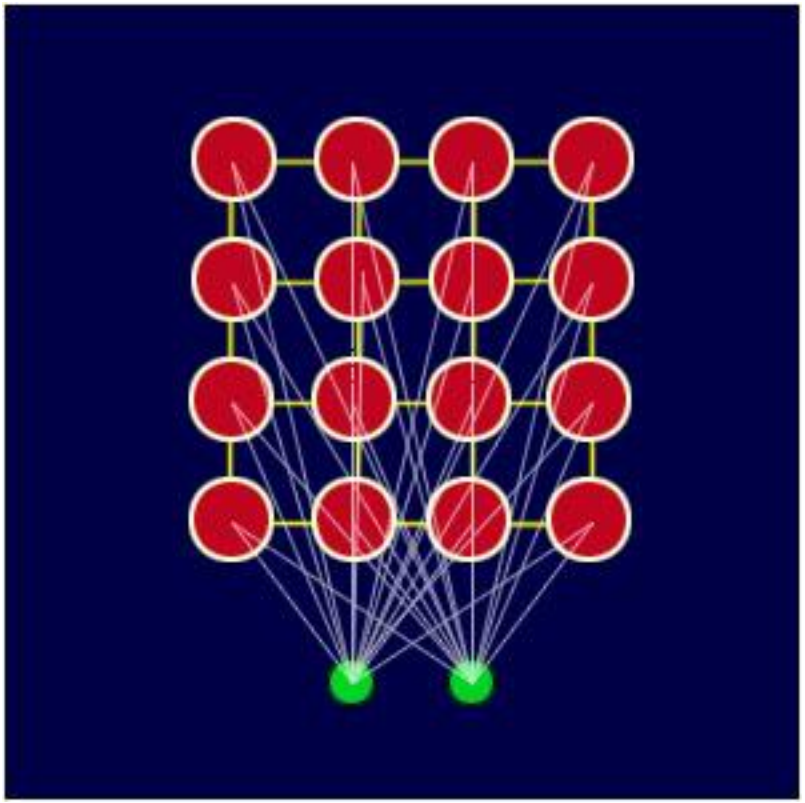


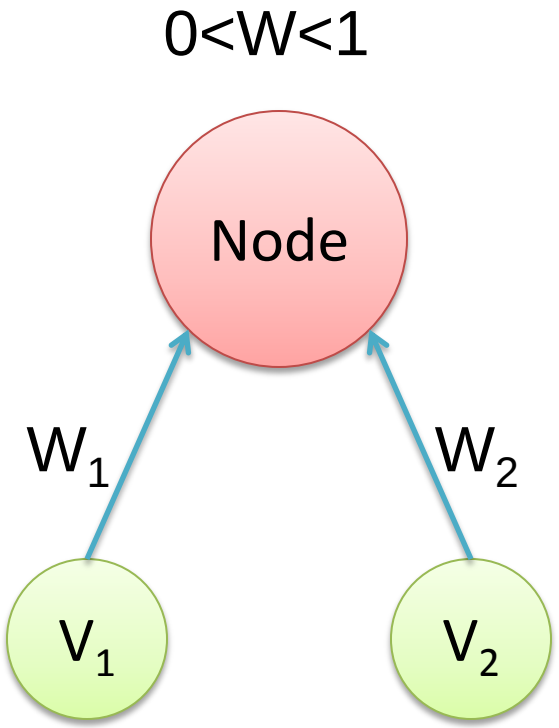
Every node is connected to the input the same way, and no nodes are connected to each other.

The Learning Process

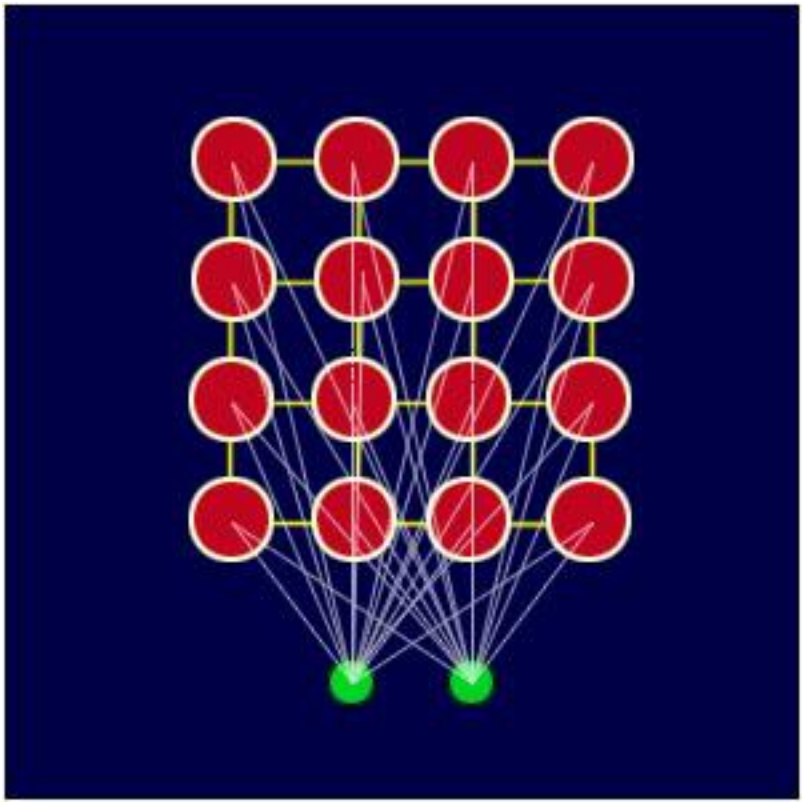
1) Initialize each node's weights.
2) Choose a random vector from training data and present it to the SOM.
3) Find the Best Matching Unit (BMU) by calculating the distance between the input vector and the weights of each node.
4) The radius of the neighborhood around the BMU is calculated. The size of the neighborhood decreases with each iteration.
5) Each node in the BMU's neighborhood has its weights adjusted to become more like the BMU. Nodes closest to the BMU are altered more than the nodes furthest away in the neighborhood.
6) Repeat from step 2 for enough iterations for convergence.

1- Initialize each node's weights





2- Choose a random vector



TEMP	HUMIDITY
85	85
80	90
83	78
70	96
68	80
65	70
64	65
72	95
69	70
75	80
75	70
72	90
81	75
…	…

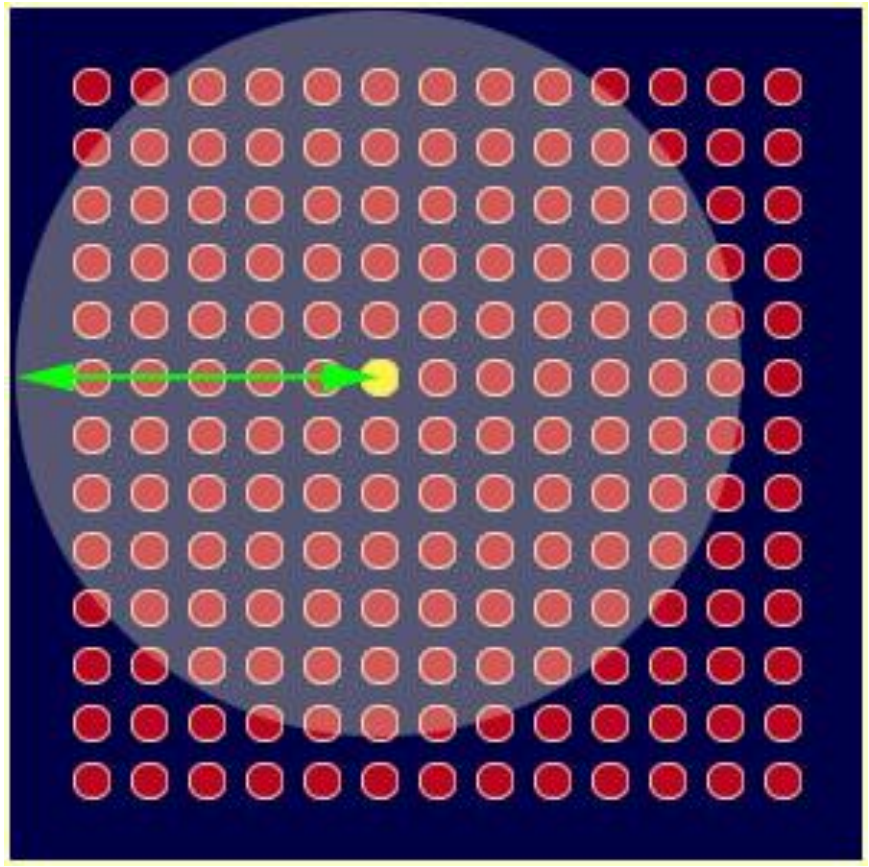
3- Calculating the Best Matching Unit

- Calculating the BMU is done according to the Euclidean distance among the node's weights (W_1, W_2, … , W_n) and the input vector's values (V_1, V_2, … , V_n).
- Euclidean distance is a measurement of similarity between two sets of data.

$$Dist = \sqrt{\sum_{i=0}^{i=n} (V_i - W_i)^2}$$

4- Determining the BMU Neighborhood

- Size of the neighborhood: an *exponential decay* function that shrinks on each iteration until eventually the neighborhood is just the BMU itself.



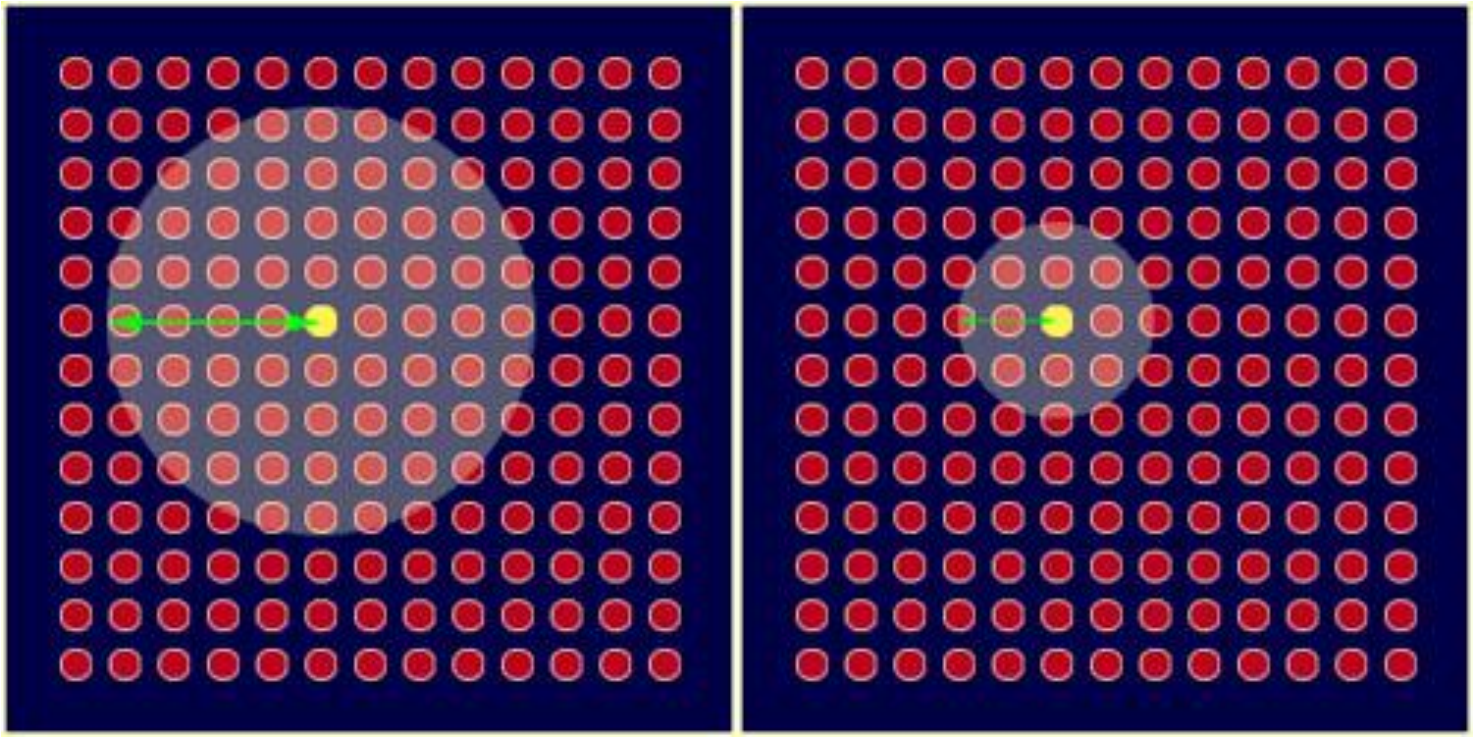
width of the lattice at time t_0

time (iteration of the loop)

$$\sigma(t) = \sigma_0 \, exp\left(-\frac{t}{\lambda}\right)$$

width of the lattice at time t

time constant



Exponential decay neighborhood function that shrinks on each iteration

5a- Modifying Nodes' Weights

- The new weight for a node is the old weight, plus a fraction (L) of the difference between the old weight and the input vector, adjusted (theta) based on distance from the BMU.

$$W(t+1) = W(t) + \Theta(t) L(t) \left(V(t) - W(t)\right)$$

- The learning rate, L, is also an exponential *decay* function.
  - This ensures that the SOM will converge.

$$L(t) = L_0 \, exp\left(-\frac{t}{\lambda}\right)$$

learning rate at time t

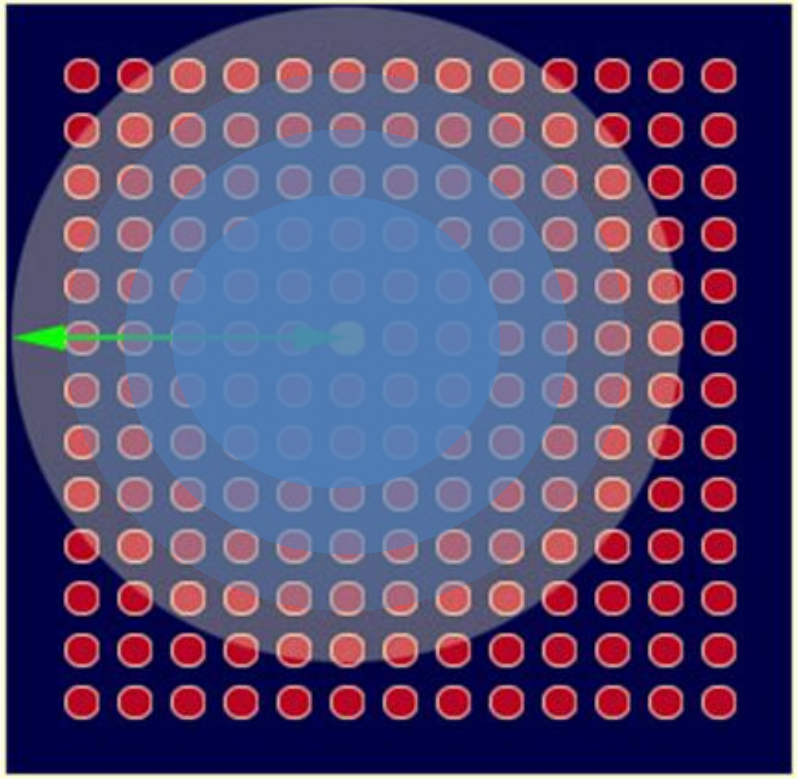
time (iteration of the loop)

learning rate at time t_0

time constant

5b- Modifying Nodes' Weights

- Effect of location within the neighborhood: The neighborhood is defined by a gaussian curve so that nodes that are closer are influenced more than farther nodes.

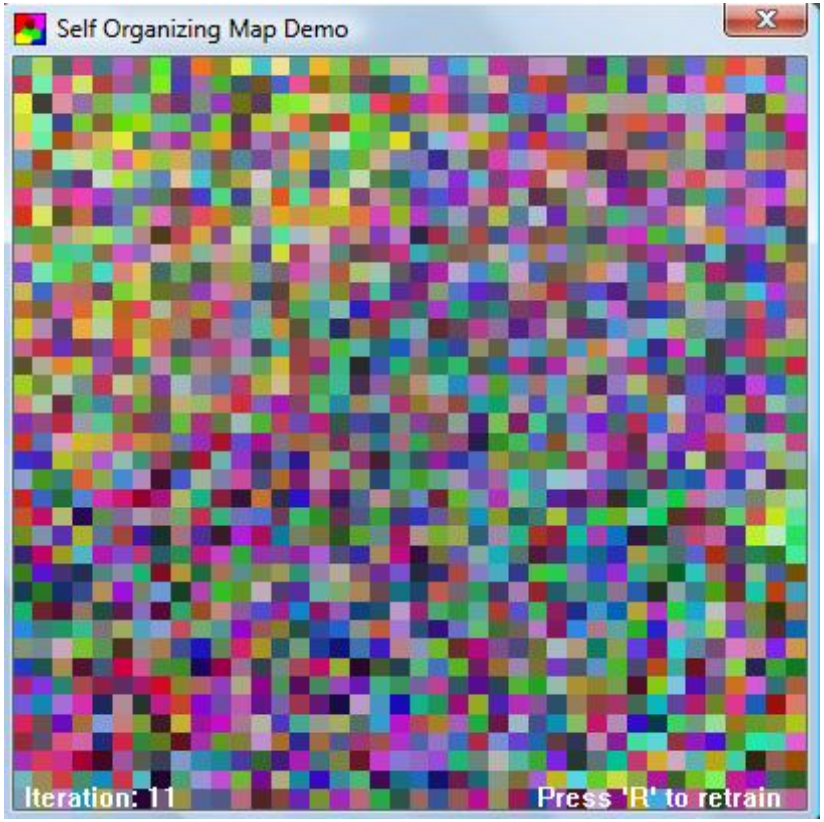


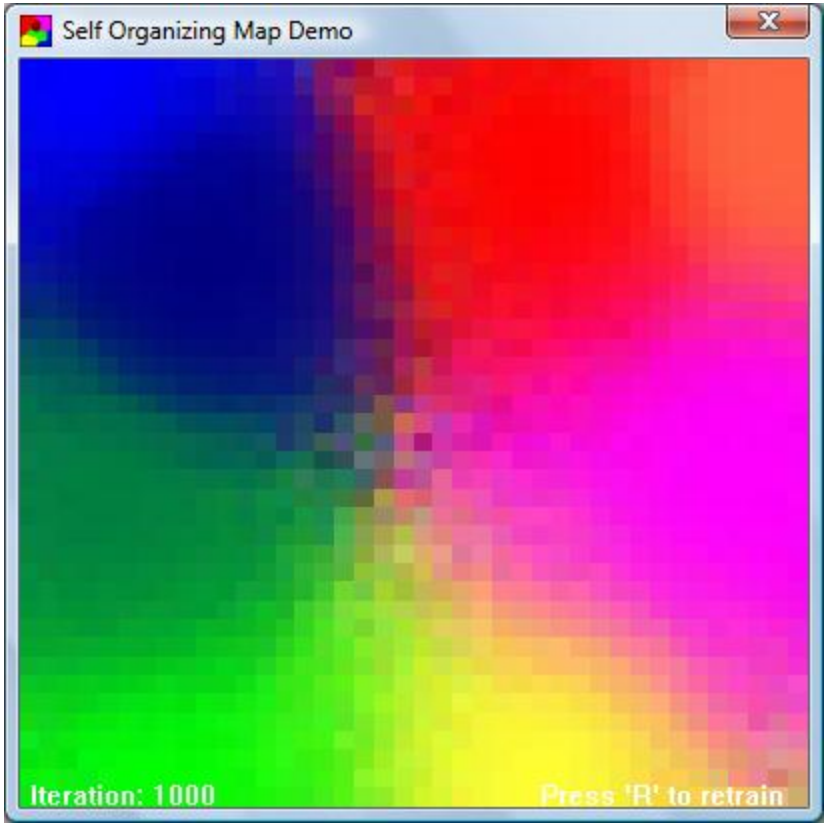
influence rate

$$\Theta(t) = exp\left(-\frac{dist^2}{2\sigma^2(t)}\right)$$

width of the lattice at time t

SOM: Demo





- 2-D square grid of nodes.
- Inputs are colors.
- SOM converges so that similar colors are grouped together.

References

- Wikipedia: “Self organizing map”.
  http://en.wikipedia.org/wiki/Self-organizing_map.
- AI-Junkie: “SOM tutorial”.
  http://www.ai-junkie.com/ann/som/som1.html.