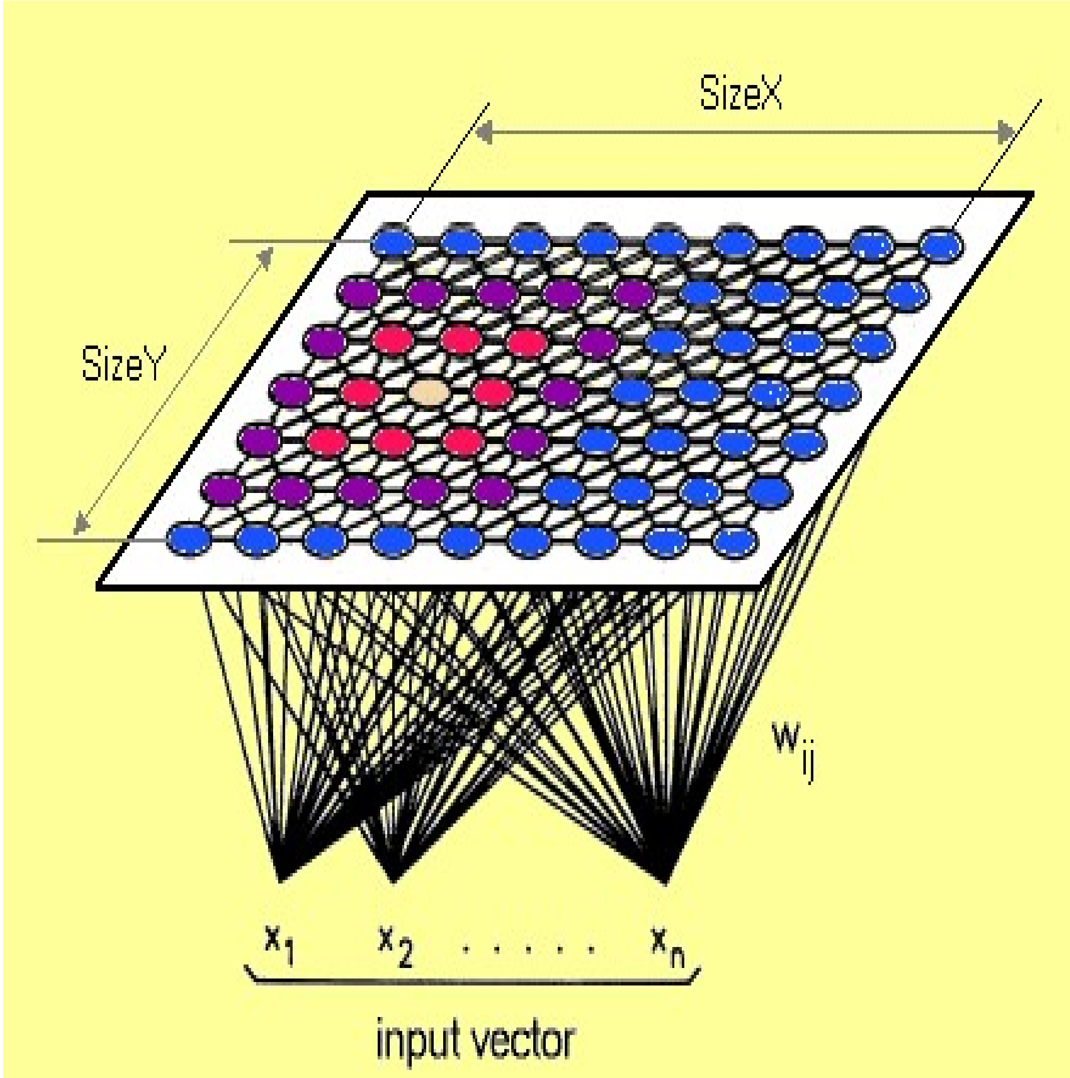
Self Organizing Map (or Kohonen Map or SOM)

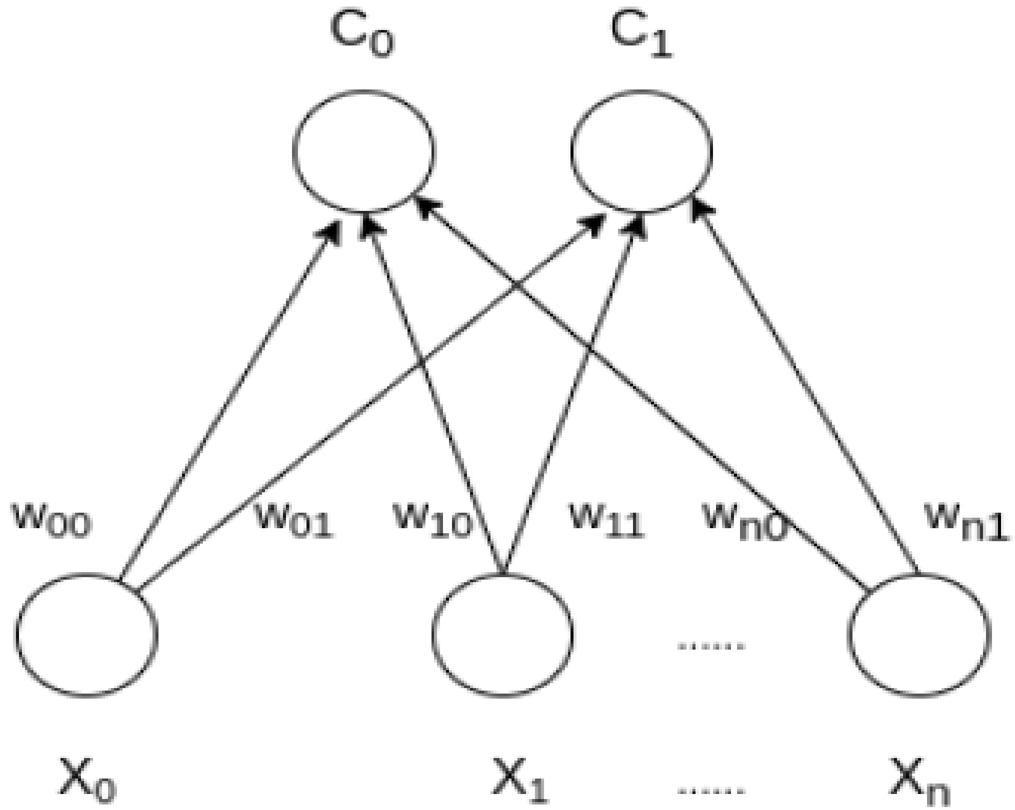
Self Organizing Map (or Kohonen Map or SOM)

- Type of ANN, inspired by biological models of neural systems from the 1970s.
- Follows an unsupervised learning approach and trained its network through a competitive learning algorithm.
- Used for clustering and mapping (or dimensionality reduction) techniques to map multidimensional data onto lower-dimensional for easy interpretation.
- Has two layers, one is the Input layer and the other one is the Output layer.



Architecture:

- Two clusters with 'n' input features for any sample:



How do SOM works?

- Let's say an input data of size (m, n) where m is the number of training examples and n is the number of features in each example.

- First, it initializes the weights of size (n, C) where C is the number of clusters.

- Then iterating over the input data, for each training example, it updates the winning vector (weight vector with the shortest distance (e.g Euclidean distance) from training example).

- Weight updation rule is given by:

  wij = wij(old) + alpha(t) * (xik - wij(old))

  - where alpha is a learning rate at time t,
  - j denotes the winning vector,
  - i denotes the ith feature of training example and
  - k denotes the kth training example from the input data.

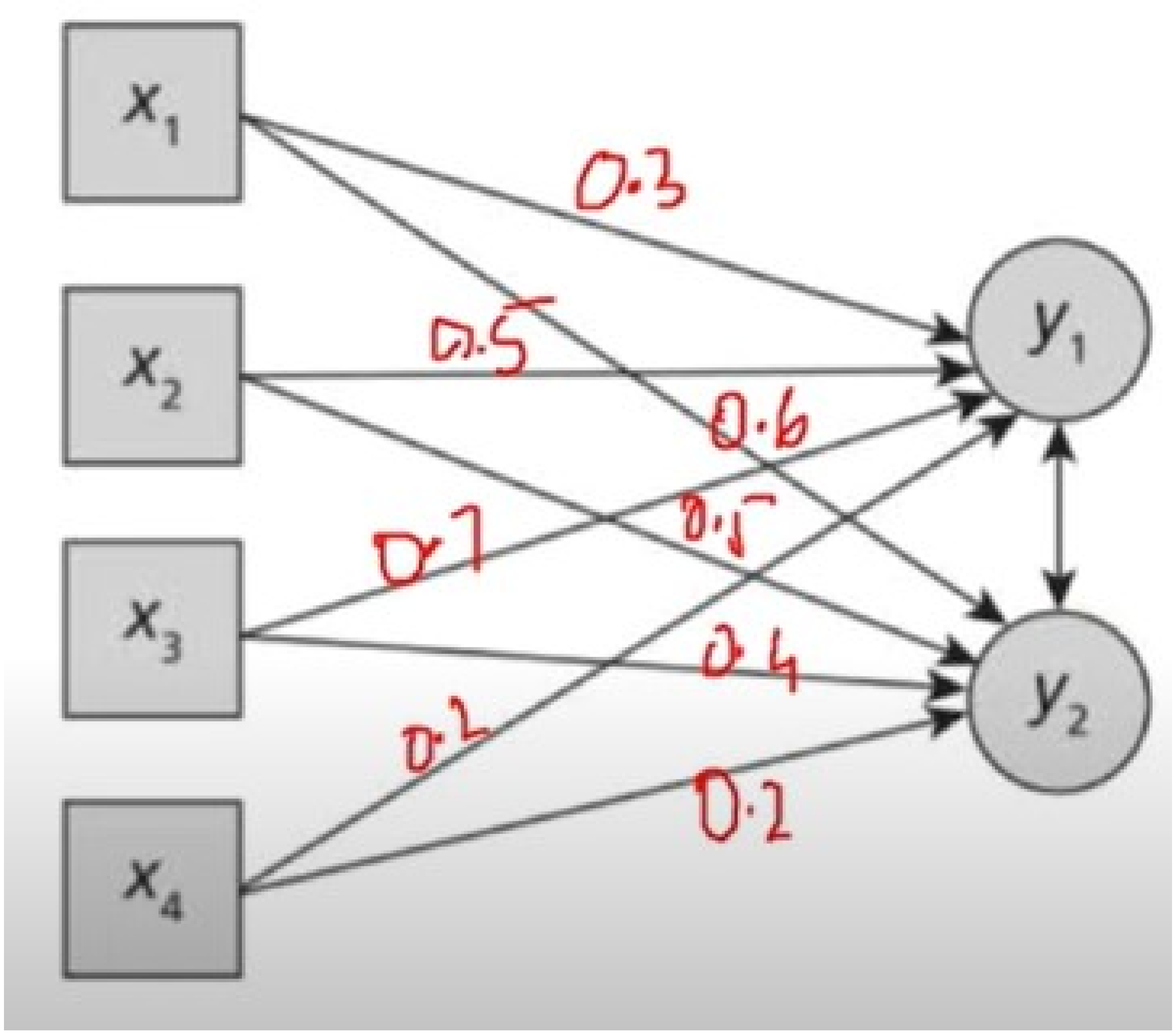
- After training, trained weights are used for clustering new examples. A new example falls in the cluster of winning vectors.

Example:

X1: (1, 0, 1, 0) X2: (1, 0, 0, 0).

X3: (1, 1, 1, 1) X4: (0, 1, 1, 0)

- Output Units: Unit 1, Unit 2
- Learning rate η(t) = 0.6
- Initial Weight matrix
- $\begin{bmatrix} Unit\ 1 \\ Unit\ 2 \end{bmatrix} = \begin{bmatrix} 0.3 & 0.5 & 0.7 & 0.2 \\ 0.6 & 0.5 & 0.4 & 0.2 \end{bmatrix}$



Example: Euclidean Distance: **d = √[(x2 – x1)2 + (y2 – y1)2].**

Iteration 1:

Training Sample x_1: (1, 0, 1, 0)

Weight matrix:

$$\begin{bmatrix} \text{Unit 1} \\ \text{Unit 2} \end{bmatrix} : \begin{bmatrix} 0.3 & 0.5 & 0.7 & 0.2 \\ 0.6 & 0.7 & 0.4 & 0.3 \end{bmatrix}$$

Compute Euclidean distance between x_1: (1, 0, 1, 0) and Unit 1 weights.

$$d^2 = (0.3 - 1)^2 + (0.5 - 0)^2 + (0.7 - 1)^2 + (0.2 - 0)^2 = 0.87$$

Compute Euclidean distance between x_1: (1, 0, 1, 0) and Unit 2 weights.

$$d^2 = (0.6 - 1)^2 + (0.7 - 0)^2 + (0.4 - 1)^2 + (0.3 - 0)^2 = 1.1$$

Unit 1 wins

Example:

$$w_j(t+1) = w_j(t) + \eta(t)(x_s - w_j(t))$$

Update the weights of the winning unit.

$$\text{New Unit 1 weights} = [0.3\ \ 0.5\ \ 0.7\ \ 0.2] + 0.6\,([1\ \ 0\ \ 1\ \ 0] - [0.3\ \ 0.5\ \ 0.7\ \ 0.2])$$

$$= [0.3\ \ 0.5\ \ 0.7\ \ 0.2] + 0.6\,[0.7\ \ -0.5\ \ 0.3\ \ -0.2]$$

$$= [0.3\ \ 0.5\ \ 0.7\ \ 0.2] + [0.42\ \ -0.30\ \ 0.18\ \ -0.12]$$

$$= [0.72\ \ 0.2\ \ 0.88\ \ 0.08]$$

$$\begin{bmatrix} \text{Unit 1} \\ \text{Unit 2} \end{bmatrix} : \begin{bmatrix} 0.72 & 0.2 & 0.88 & 0.08 \\ 0.6 & 0.7 & 0.4 & 0.3 \end{bmatrix}$$

Example:

Iteration 2:

Training Sample x_2: (1, 0, 0, 0)

Weight matrix:

$$\begin{bmatrix} \text{Unit 1} \\ \text{Unit 2} \end{bmatrix} : \begin{bmatrix} 0.72 & 0.2 & 0.88 & 0.08 \\ 0.6 & 0.7 & 0.4 & 0.3 \end{bmatrix}$$

Compute Euclidean distance between x_2: (1, 0, 0, 0) and Unit 1 weights.

$$d^2 = (0.72 - 1)^2 + (0.2 - 0)^2 + (0.88 - 0)^2 + (0.08 - 0)^2 = 0.74$$

Compute Euclidean distance between x_2: (1, 0, 0, 0) and Unit 2 weights.

$$d^2 = (0.6 - 1)^2 + (0.7 - 0)^2 + (0.4 - 0)^2 + (0.3 - 0)^2 = 0.9$$

Unit 1 wins

Example:

$$w_j(t+1) = w_j(t) + \eta(t)(x_s - w_j(t))$$

Update the weights of the winning unit:

$$\text{New Unit 1 weights} = [0.72\ \ 0.2\ \ 0.88\ \ 0.08] + 0.6\,([1\ \ 0\ \ 0\ \ 0] - [0.72\ \ 0.2\ \ 0.88\ \ 0.08])$$

$$= [0.72\ \ 0.2\ \ 0.88\ \ 0.08] + 0.6\,[0.28\ \ -0.2\ \ -0.88\ \ -0.08]$$

$$= [0.72\ \ 0.2\ \ 0.88\ \ 0.08] + [0.17\ \ -0.12\ \ -0.53\ \ -0.05]$$

$$= [0.89\ \ 0.08\ \ 0.35\ \ 0.03]$$

$$\begin{bmatrix} \text{Unit 1} \\ \text{Unit 2} \end{bmatrix} : \begin{bmatrix} 0.89 & 0.08 & 0.35 & 0.03 \\ 0.6 & 0.7 & 0.4 & 0.3 \end{bmatrix}$$

Example:

Iteration 3:

Training Sample x_3: (1, 1, 1, 1)

Weight matrix:

$$\begin{bmatrix} \text{Unit 1} \\ \text{Unit 2} \end{bmatrix} : \begin{bmatrix} 0.89 & 0.08 & 0.35 & 0.03 \\ 0.6 & 0.7 & 0.4 & 0.3 \end{bmatrix}$$

Compute Euclidean distance between x_3: (1, 1, 1, 1) and Unit 1 weights.

$$d^2 = (0.89 - 1)^2 + (0.08 - 1)^2 + (0.35 - 1)^2 + (0.03 - 1)^2$$
$$= 2.2$$

Compute Euclidean distance between x_3: (1, 1, 1, 1) and Unit 2 weights.

$$d^2 = (0.6 - 1)^2 + (0.7 - 1)^2 + (0.4 - 1)^2 + (0.3 - 1)^2$$
$$= 1.1$$

Unit 2 wins

Example:

$$w_j(t+1) = w_j(t) + \eta(t)(x_s - w_j(t))$$

Update the weights of the winning unit:

$$\begin{aligned}
\text{New Unit 2 weights} &= [0.6\ \ 0.7\ \ 0.4\ \ 0.3] + 0.6\,([1\ \ 1\ \ 1\ \ 1] - [0.6\ \ 0.7\ \ 0.4\ \ 0.3]) \\
&= [0.6\ \ 0.7\ \ 0.4\ \ 0.3] + 0.6\,[0.4\ \ 0.3\ \ 0.6\ \ 0.7] \\
&= [0.6\ \ 0.7\ \ 0.4\ \ 0.3] + [0.24\ \ 0.18\ \ 0.36\ \ 0.42] = [0.84\ \ 0.88\ \ 0.76\ \ 0.72]
\end{aligned}$$

$$\begin{bmatrix} \text{Unit 1} \\ \text{Unit 2} \end{bmatrix} : \begin{bmatrix} 0.89 & 0.08 & 0.35 & 0.03 \\ 0.84 & 0.88 & 0.76 & 0.72 \end{bmatrix}$$

Example:

Iteration 4:

Training Sample x_4: (0, 1, 1, 0)

Weight matrix:

$$\begin{bmatrix} \text{Unit 1} \\ \text{Unit 2} \end{bmatrix} : \begin{bmatrix} 0.89 & 0.08 & 0.35 & 0.03 \\ 0.84 & 0.88 & 0.76 & 0.72 \end{bmatrix}$$

Compute Euclidean distance between x_4: (0, 1, 1, 0) and Unit 1 weights.

$$d^2 = (0.89 - 0)^2 + (0.08 - 1)^2 + (0.35 - 1)^2 + (0.03 - 0)^2$$

$$= 2.06$$

Compute Euclidean distance between x_1: (0, 1, 1, 0) and Unit 2 weights.

$$d^2 = (0.84 - 0)^2 + (0.88 - 1)^2 + (0.76 - 1)^2 + (0.72 - 0)^2$$

$$= 1.3$$

Example:

$$w_j(t+1) = w_j(t) + \eta(t)(x_s - w_j(t))$$

Unit 2 wins

Update the weights of the winning unit:

New Unit 2 weights $= [0.84\ \ 0.88\ \ 0.76\ \ 0.72] + 0.6\,([0\ \ 1\ \ 1\ \ 0] - [0.84\ \ 0.88\ \ 0.76\ \ 0.72])$

$= [0.84\ \ 0.88\ \ 0.76\ \ 0.72] + 0.6\,[-0.84\ \ 0.12\ \ 0.24\ \ -0.72]$

$= [0.84\ \ 0.88\ \ 0.76\ \ 0.72] + [-0.5\ \ 0.07\ \ 0.14\ \ -0.43] = [0.34\ \ 0.95\ \ 0.9\ \ 0.29]$

$$\begin{bmatrix} \text{Unit 1} \\ \text{Unit 2} \end{bmatrix} : \begin{bmatrix} 0.89 & 0.08 & 0.35 & 0.03 \\ 0.34 & 0.95 & 0.9 & 0.29 \end{bmatrix}$$

Example:

Best mapping units for each of the sample taken are:

x_1: (1, 0, 1, 0) → Unit 1

x_2: (1, 0, 0, 0) → Unit 1

x_3: (1, 1, 1, 1) → Unit 2

x_4: (0, 1, 1, 0) → Unit 2

Epoch

This process is continued for many epochs until the feature map does not change.