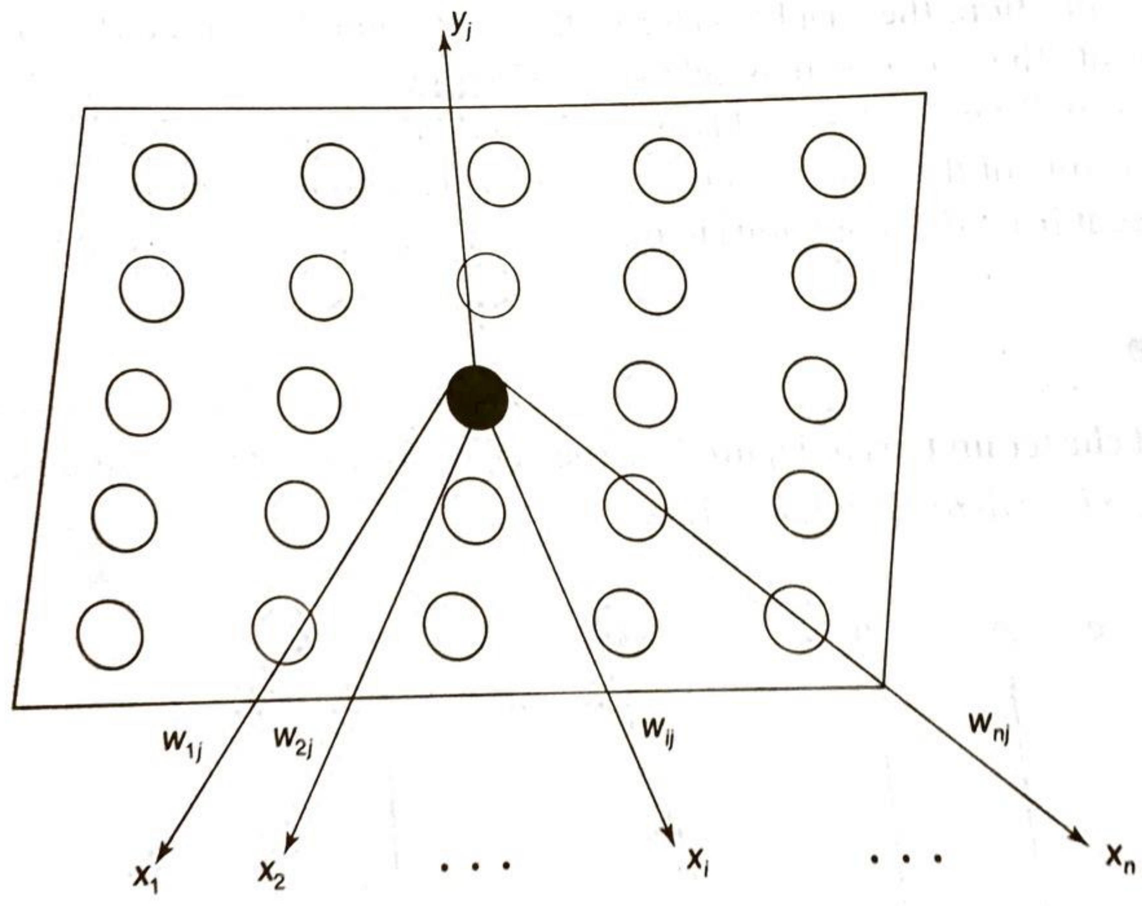
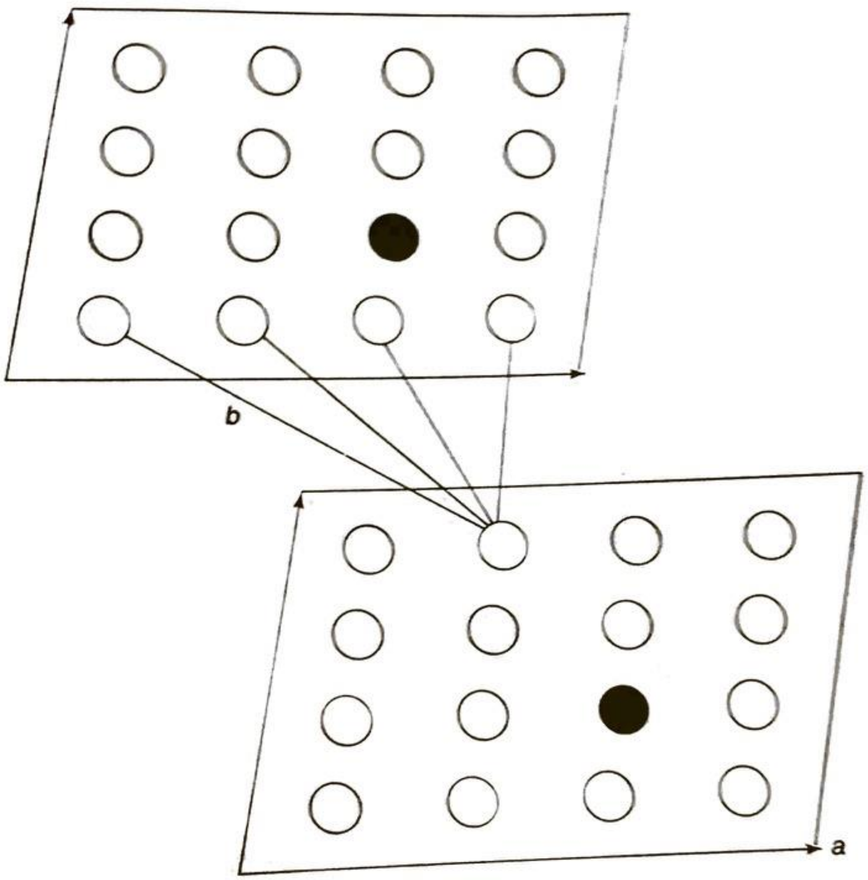
3.1 K means learning

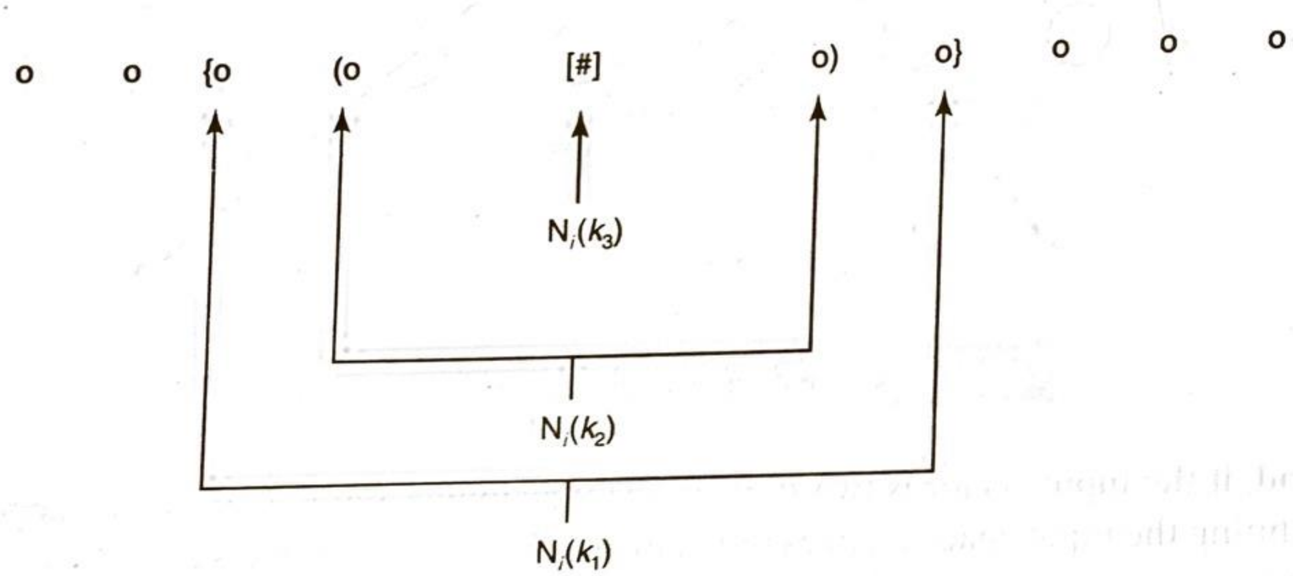
One dimensional feature mapping Network



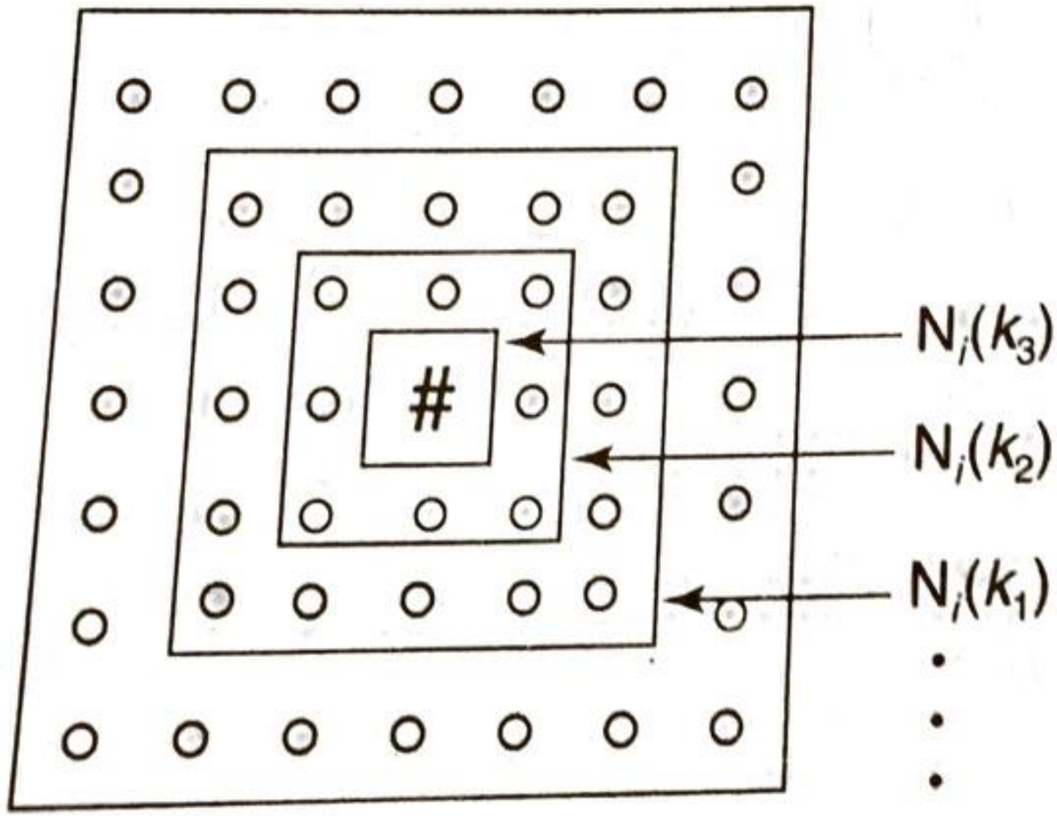
Two dimensional feature mapping Network



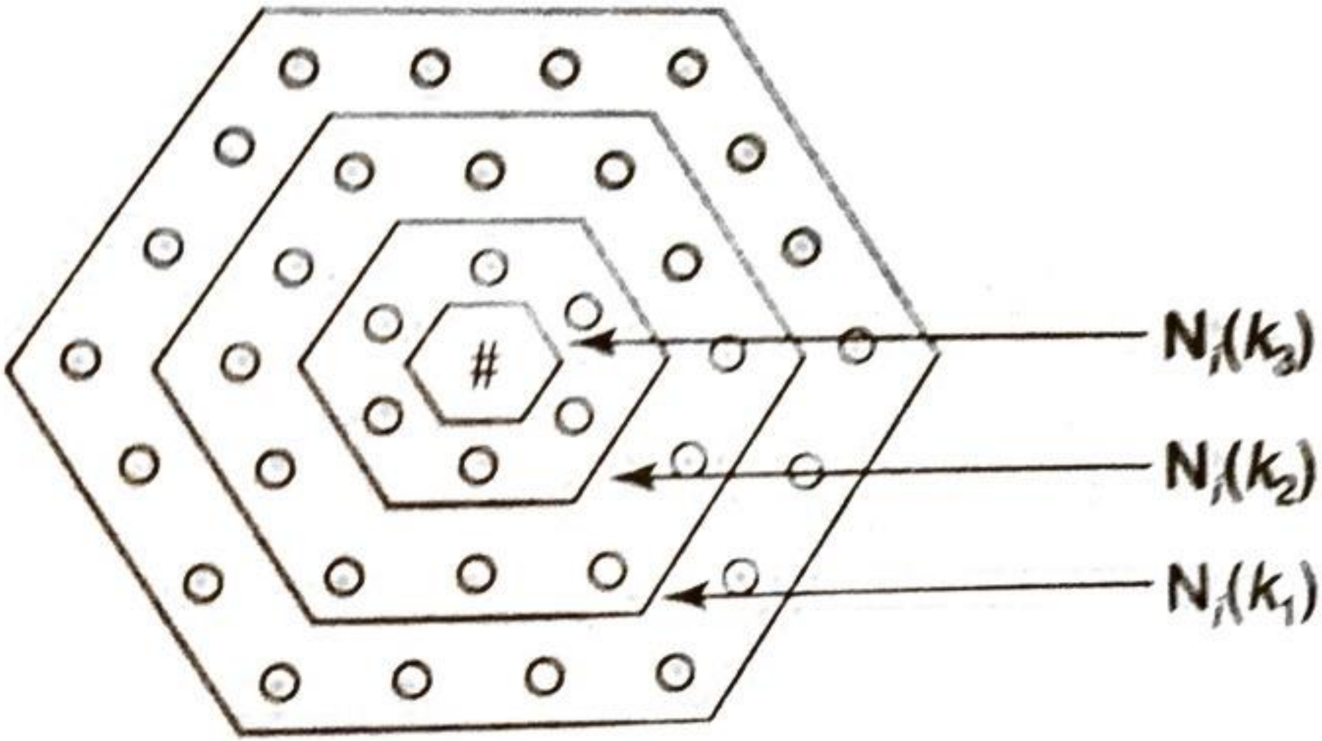
Linear Array of clustering units



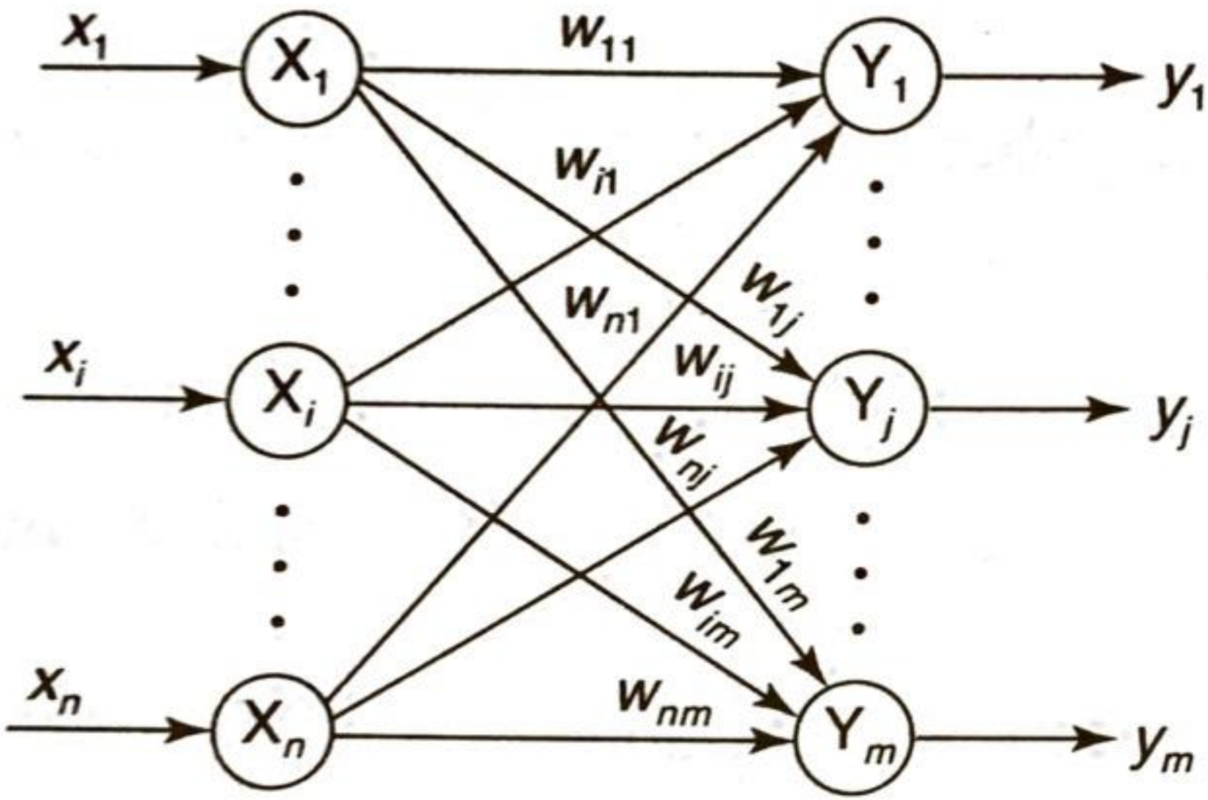
Rectangular Grid



Hexagonal Grid



Kohonen self organizing feature Map Archetecture



Training Algorithm

Step 0:
- Initialize the weights w_{ij}: Random values may be assumed. They can be chosen as the same range of values as the components of the input vector. If information related to distribution of clusters is known, the initial weights can be taken to reflect that prior knowledge.
- Set topological neighborhood parameters: As clustering progresses, the radius of the neighborhood decreases.
- Initialize the learning rate α: It should be a slowly decreasing function of time.

Step 1: Perform Steps 2–8 when stopping condition is false.

Step 2: Perform Steps 3–5 for each input vector x.

Step 3: Compute the square of the Euclidean distance, i.e., for each $j = 1$ to m,

$$D(j) = \sum_{i=1}^{n} \sum_{j=1}^{m} (x_i - w_{ij})^2$$

Step 4: Find the winning unit index J, so that $D(J)$ is minimum. (In Steps 3 and 4, dot product method can also be used to find the winner, which is basically the calculation of net input, and the winner will be the one with the largest dot product.)

Step 5: For all units j within a specific neighborhood of J and for all i, calculate the new weights:

$$w_{ij}(\text{new}) = w_{ij}(\text{old}) + \alpha[x_i - w_{ij}(\text{old})]$$

or

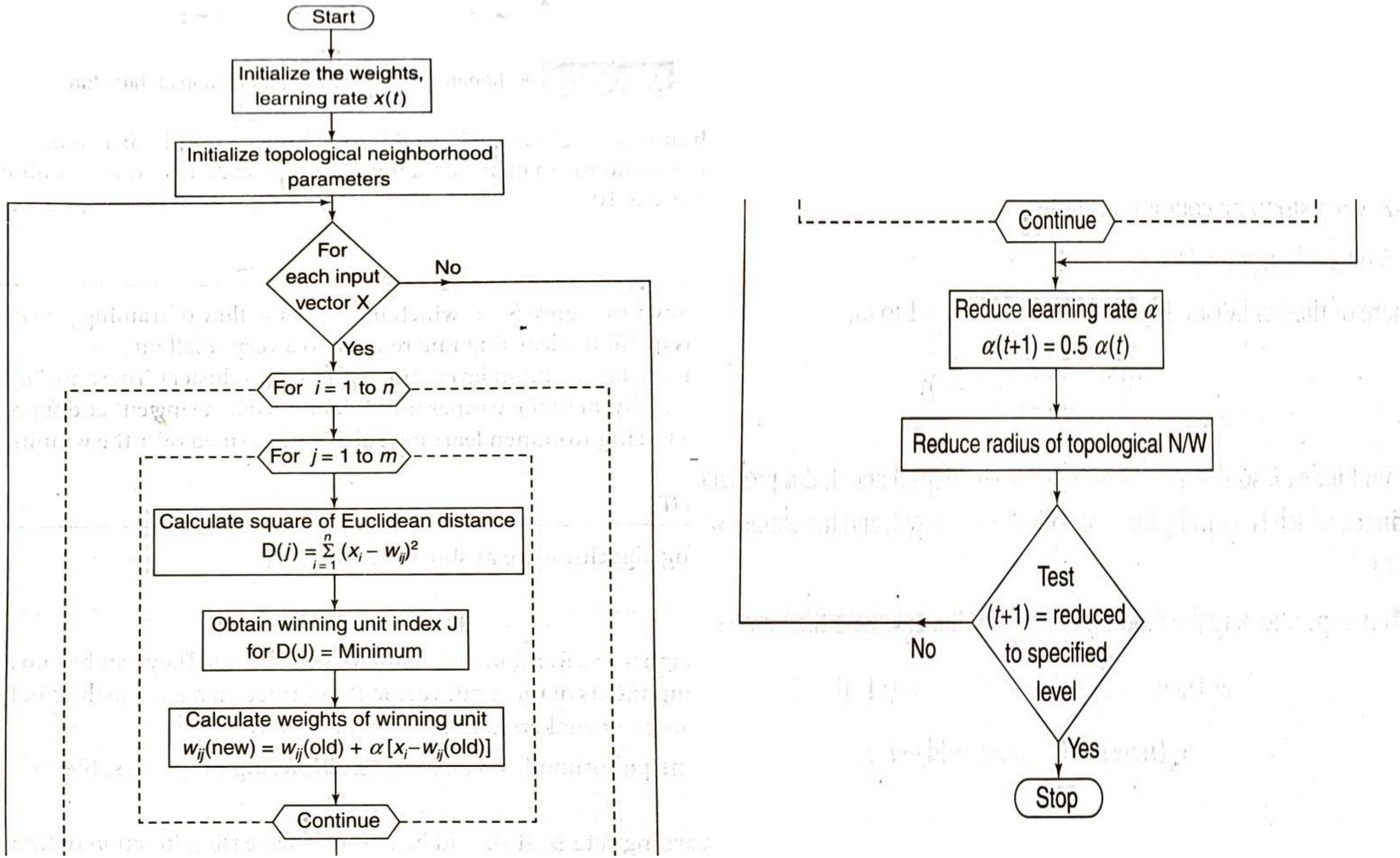
$$w_{ij}(\text{new}) = (1-\alpha)w_{ij}(\text{old}) + \alpha\, x_i$$

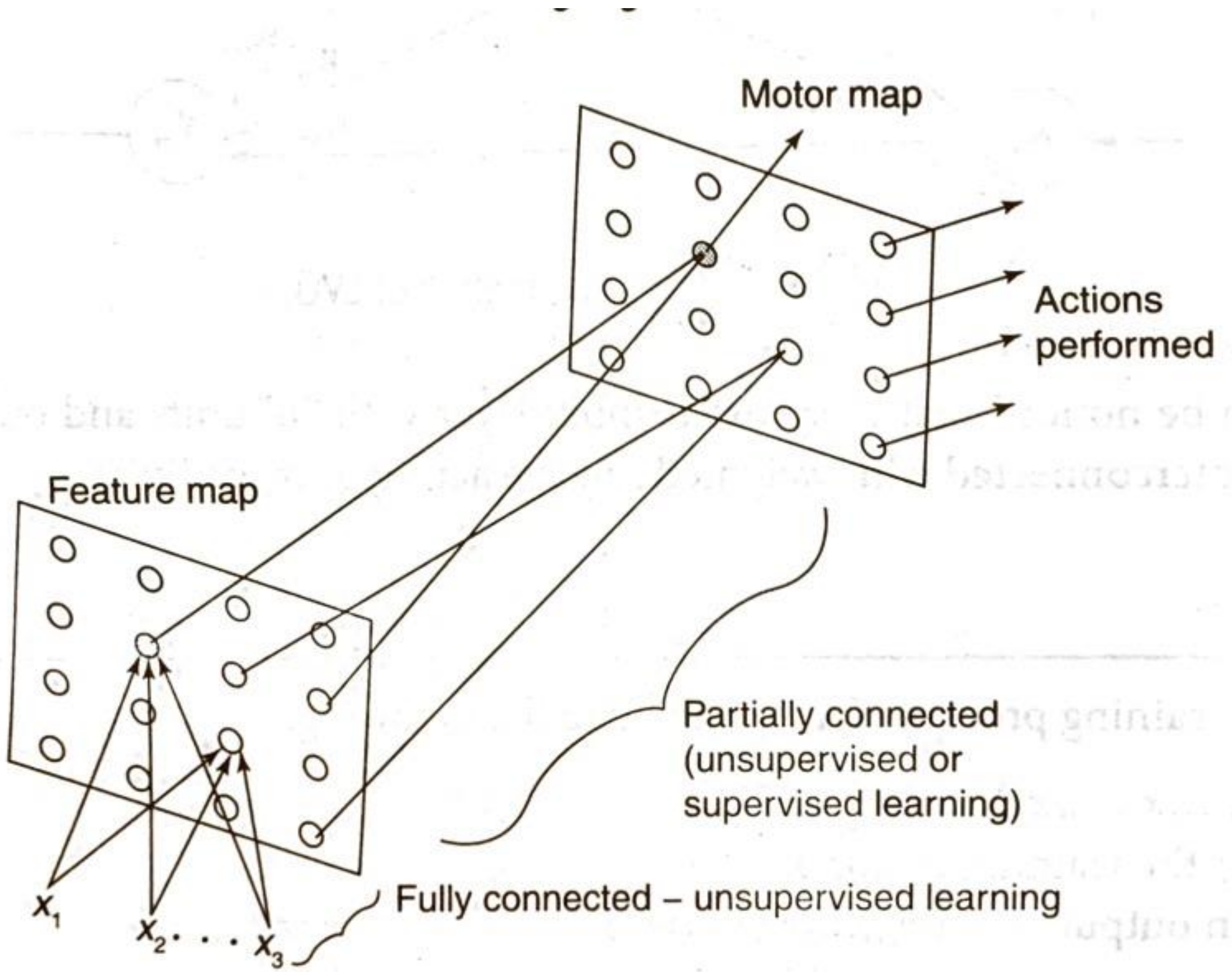
Step 6: Update the learning rate α using the formula $\alpha(t+1) = 0.5\alpha(t)$.

Step 7: Reduce radius of topological neighborhood at specified time intervals.

Step 8: Test for stopping condition of the network.

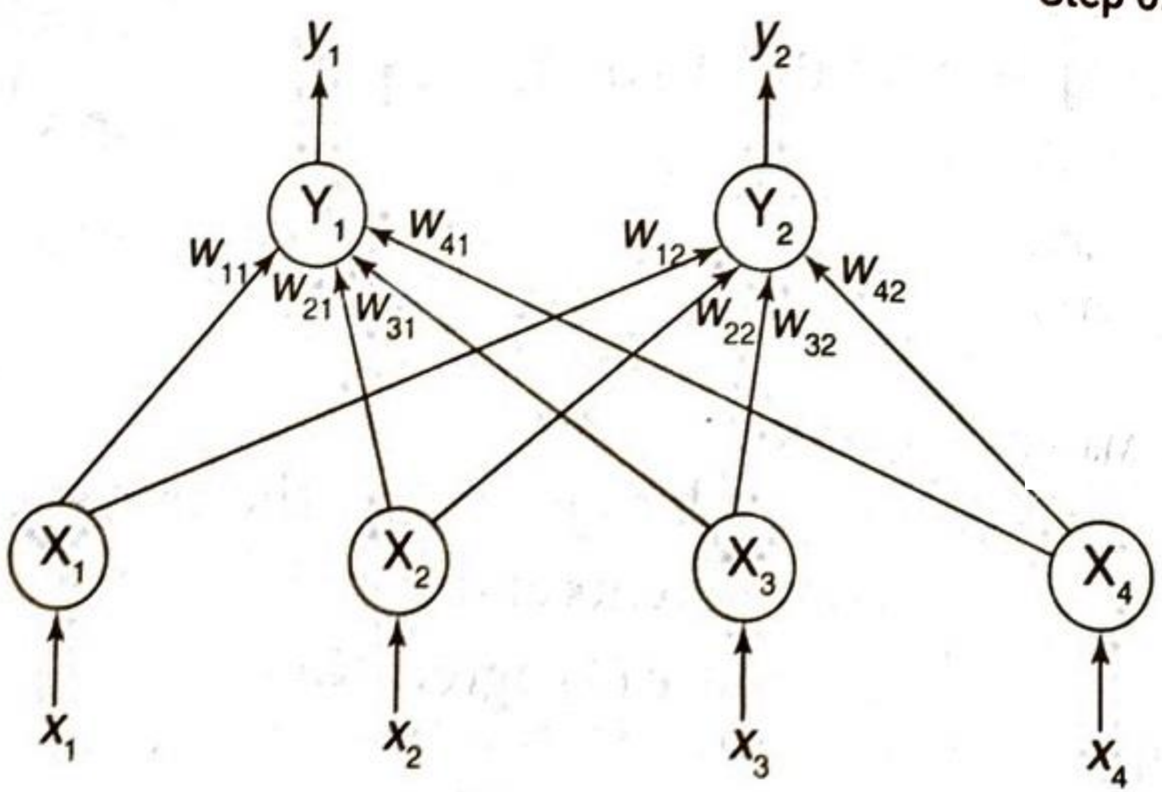
Flowchart of Training Algorithm




Motor map
Actions performed
Feature map
Partially connected (unsupervised or supervised learning)
Fully connected – unsupervised learning
x_1
$x_2 \cdots x_3$

Examples

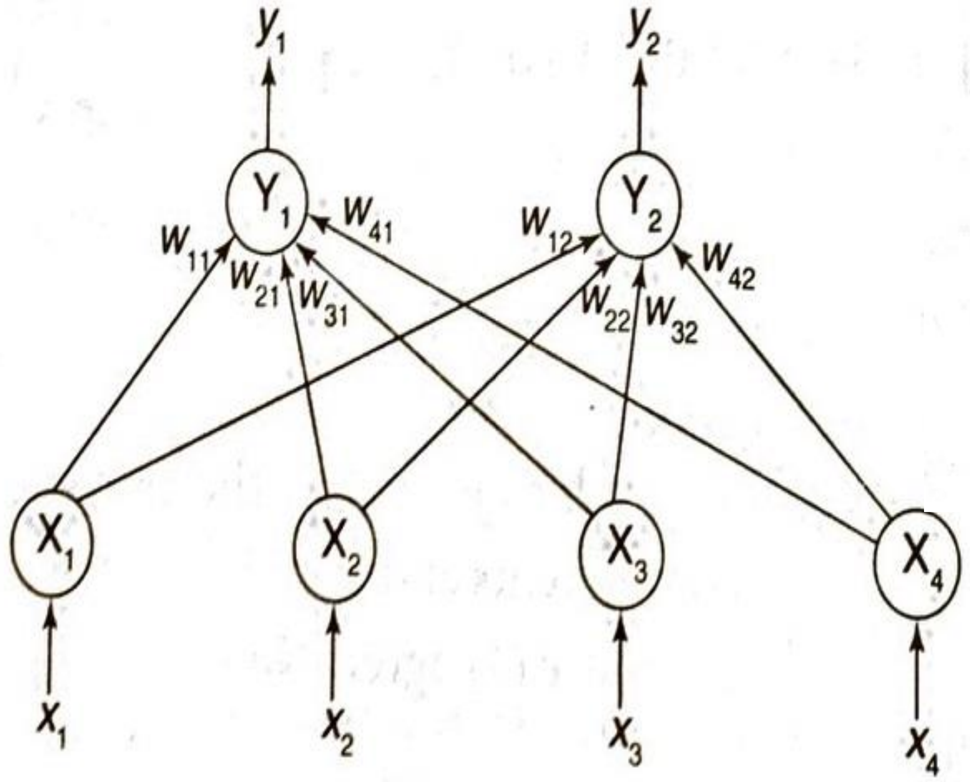
Construct a Kohonen self-organizing map to cluster the four given vectors, [0 0 1 1], [1 0 0 0], [0 1 1 0] and [0 0 0 1]. The number of clusters to be formed is two. Assume an initial learning rate of 0.5.



Step 0: Initialize the weights rand between 0 and 1.

$$w_{ij} = \begin{bmatrix} 0.2 & 0.9 \\ 0.4 & 0.7 \\ 0.6 & 0.5 \\ 0.8 & 0.3 \end{bmatrix}_{4\times 2} ;R=0;\alpha(0) =$$

Examples



Step 0: Initialize the weights randomly between 0 and 1.

$$w_{ij} = \begin{bmatrix} 0.2 & 0.9 \\ 0.4 & 0.7 \\ 0.6 & 0.5 \\ 0.8 & 0.3 \end{bmatrix}_{4\times 2} ; R = 0; \alpha(0) = 0.5$$

First Input Vector

Step 1: For $x = [0 \quad 0 \quad 1 \quad 1]$, perform Steps 2–4.

Step 2: Calculate the Euclidean distance:

$$D(j) = \sum_i (w_{ij} - x_i)^2$$

$$D(1) = \sum_{i=1}^{4} (w_{i1} - x_i)^2$$

$$= (0.2-0)^2 + (0.4-0)^2$$

$$+ (0.6-1)^2 + (0.8-1)^2$$

$$= 0.04 + 0.16 + 0.16 + 0.04$$

$$= 0.4$$

First Input Vector

$$D(2) = \sum_{i=1}^{4}(w_{i2} - x_i)^2$$
$$= (0.9 - 0)^2 + (0.7 - 0)^2$$
$$+ (0.5 - 1)^2 + (0.3 - 1)^2$$
$$= 0.81 + 0.49 + 0.25 + 0.49$$
$$= 2.04$$

Step 3: Since $D(1) < D(2)$, therefore $D(1)$ is minimum. Hence the winning cluster unit is Y_1, i.e., $J = 1$.

First Input Vector

Step 4: Update the weights on the winning cluster unit $J = 1$.

$$w_{iJ}(\text{new}) = w_{iJ}(\text{old}) + \alpha[x_i - w_{iJ}(\text{old})]$$

$$w_{i1}(\text{new}) = w_{i1}(\text{old}) + 0.5[x_i - w_{i1}(\text{old})]$$

$$w_{11}(n) = w_{11}(0) + 0.5[x_1 - w_{11}(0)]$$

$$= 0.2 + 0.5(0 - 0.2) = 0.1$$

$$w_{21}(n) = w_{21}(0) + 0.5[x_2 - w_{21}(0)]$$

$$= 0.4 + 0.5(0 - 0.4) = 0.2$$

$$w_{31}(n) = w_{31}(0) + 0.5[x_3 - w_{31}(0)]$$

$$= 0.6 + 0.5(1 - 0.6) = 0.8$$

$$w_{41}(n) = w_{41}(0) + 0.5[x_4 - w_{41}(0)]$$

$$= 0.8 + 0.5(1 - 0.8) = 0.9$$

First Input Vector

The updated weight matrix after presentation of first input pattern is

$$w_{ij} = \begin{bmatrix} 0.1 & 0.9 \\ 0.2 & 0.7 \\ 0.8 & 0.5 \\ 0.9 & 0.3 \end{bmatrix}$$

Second Input Vector

Step 1: For $x = [1\ 0\ 0\ 0]$, perform Steps 2–4.

Step 2: Calculate the Euclidean distance:

$$D(j) = \sum_i (w_{ij} - x_i)^2$$

$$D(1) = \sum_{i=1}^{4} (w_{i1} - x_i)^2$$

$$= (0.1 - 1)^2 + (0.2 - 0)^2$$

Second Input Vector

$$+(0.8-0)^2+(0.9-0)^2$$
$$=0.81+0.04+0.64+0.81$$
$$=2.3$$

$$D(2)=\sum_{i=1}^{4}(w_{i2}-x_i)^2$$
$$=(0.9-1)^2+(0.7-0)^2$$
$$+(0.5-0)^2+(0.3-0)^2$$
$$=0.01+0.49+0.25+0.09$$
$$=0.84$$

Step 3: Since $D(2)<D(1)$, therefore $D(2)$ is minimum. Hence the winning cluster unit is Y_2, i.e., $J=2$.

Second Input Vector

Step 4: Update the weights on the winning cluster unit $J = 2$:

$$w_{iJ}(\text{new}) = w_{iJ}(\text{old}) + \alpha[x_i - w_{iJ}(\text{old})]$$

$$w_{i2}(\text{new}) = w_{i2}(\text{old}) + 0.5[x_i - w_{i2}(\text{old})]$$

$$w_{12}(n) = w_{12}(0) + 0.5[x_1 - w_{12}(0)]$$

$$= 0.9 + 0.5(1 - 0.9) = 0.95$$

$$w_{22}(n) = w_{22}(0) + 0.5[x_2 - w_{22}(0)]$$

$$= 0.7 + 0.5(0 - 0.7) = 0.35$$

$$w_{32}(n) = w_{32}(0) + 0.5[x_3 - w_{32}(0)]$$

$$= 0.5 + 0.5(0 - 0.5) = 0.25$$

$$w_{42}(n) = w_{42}(0) + 0.5[x_4 - w_{42}(0)]$$

$$= 0.3 + 0.5(0 - 0.3) = 0.15$$

Second Input Vector

The updated weight matrix after presentation of second input pattern is

$$w_{ij} = \begin{bmatrix} 0.1 & 0.95 \\ 0.2 & 0.35 \\ 0.8 & 0.25 \\ 0.9 & 0.15 \end{bmatrix}$$

Third Input Vector

Step 1: For $x = [0\ 1\ 1\ 0]$, perform Steps 2–4.

Step 2: Calculate the Euclidean distance:

$$D(j) = \sum_i (w_{ij} - x_i)^2$$

$$D(1) = \sum_{i=1}^{4} (w_{i1} - x_i)^2$$

Third Input Vector

$$= (0.1-0)^2 + (0.2-1)^2$$

$$+ (0.8-1)^2 + (0.9-0)^2$$

$$= 0.01 + 0.64 + 0.04 + 0.81 = 1.5$$

$$D(2) = \sum_{i=1}^{4} (w_{i2} - x_i)^2$$

$$= (0.95-0)^2 + (0.35-1)^2$$

$$+ (0.25-1)^2 + (0.15-0)^2$$

$$= (0.9025) + (0.4225) + (0.5625)$$

$$+ (0.0225) = 1.91$$

Step 3: Since $D(1) < D(2)$, therefore $D(1)$ is minimum. Hence the winning cluster unit is Y_1, i.e., $J = 1$.

Third Input Vector

Step 4: Update the weights on the winning cluster unit $J = 1$:

$$w_{iJ}(\text{new}) = w_{iJ}(\text{old}) + \alpha[x_i - w_{iJ}(\text{old})]$$

$$w_{i1}(\text{new}) = w_{i1}(\text{old}) + 0.5[x_i - w_{i1}(\text{old})]$$

$$w_{11}(n) = w_{11}(0) + 0.5[x_1 - w_{11}(0)]$$

$$= 0.1 + 0.5(0 - 0.1) = 0.05$$

$$w_{21}(n) = w_{21}(0) + 0.5[x_2 - w_{21}(0)]$$

$$= 0.2 + 0.5(1 - 0.2) = 0.6$$

$$w_{31}(n) = w_{31}(0) + 0.5[x_3 - w_{31}(0)]$$

$$= 0.8 + 0.5(1 - 0.8) = 0.9$$

$$w_{41}(n) = w_{41}(0) + 0.5[x_4 - w_{41}(0)]$$

$$= 0.9 + 0.5(0 - 0.9) = 0.45$$

Third Input Vector

The weight update after presentation of third input pattern is

$$w_{ij} = \begin{bmatrix} 0.05 & 0.95 \\ 0.6 & 0.35 \\ 0.9 & 0.25 \\ 0.45 & 0.15 \end{bmatrix}$$

Fourth Input Vector

Step 1: For $x = [0\ 0\ 0\ 1]$, perform Steps 2–4.

Step 2: Compute the Euclidean distance:

$$D(j) = \sum_{i=1}^{4} (w_{ij} - x_i)^2$$

Fourth Input Vector

$$D(1) = \sum_{i=1}^{4} (w_{i1} - x_i)^2$$

$$= (0.05 - 0)^2 + (0.6 - 0)^2$$

$$+ (0.9 - 0)^2 + (0.45 - 1)^2$$

$$= 0.0025 + 0.36 + 0.81 + 0.3025$$

$$= 1.475$$

$$D(2) = \sum_{i=1}^{4} (w_{i2} - x_i)^2$$

$$= (0.95 - 0)^2 + (0.35 - 0)^2$$

$$+ (0.25 - 0)^2 + (0.15 - 1)^2$$

$$= (0.9025) + (0.1225) + (0.0625)$$

$$+ (0.7225)$$

$$= 1.81$$

Step 3: Since $D(1) < D(2)$, therefore $D(1)$ is minimum. Hence the winning cluster unit is Y_1, i.e., $J = 1$.

Fourth Input Vector

Step 4: Update the weights on the winning cluster unit $J = 1$:

$$w_{iJ}(\text{new}) = w_{iJ}(\text{old}) + \alpha[x_i - w_{iJ}(\text{old})]$$

$$w_{i1}(\text{new}) = w_{i1}(\text{old}) + 0.5[x_i - w_{i1}(\text{old})]$$

$$w_{11}(n) = w_{11}(0) + 0.5[x_1 - w_{11}(0)]$$
$$= 0.05 + 0.5(0 - 0.05) = 0.025$$

$$w_{21}(n) = w_{21}(0) + 0.5[x_2 - w_{21}(0)]$$
$$= 0.6 + 0.5(0 - 0.6) = 0.3$$

$$w_{31}(n) = w_{31}(0) + 0.5[x_3 - w_{31}(0)]$$
$$= 0.9 + 0.5(0 - 0.9) = 0.45$$

$$w_{41}(n) = w_{41}(0) + 0.5[x_4 - w_{41}(0)]$$
$$= 0.45 + 0.5(1 - 0.95) = 0.475$$

Fourth Input Vector

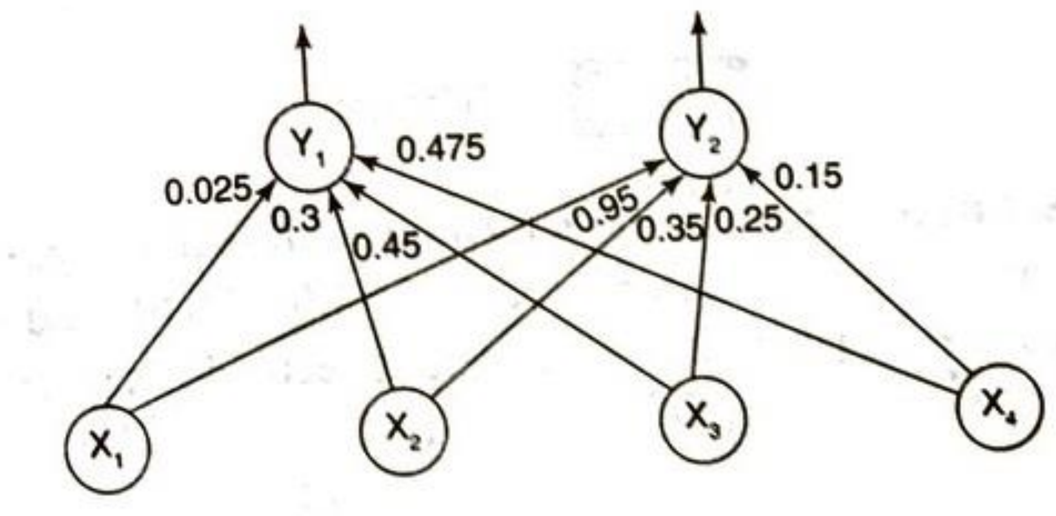
The final weight obtained after the presentation of fourth input pattern is

$$w_{ij} = \begin{bmatrix} 0.025 & 0.95 \\ 0.3 & 0.35 \\ 0.45 & 0.25 \\ 0.475 & 0.15 \end{bmatrix}$$

Since all the four given input patterns are presented, this is end of first iteration or 1-epoch. Now the learning rate can be updated as

$$\alpha(t+1) = 0.5\alpha(t)$$

$$\alpha(1) = 0.5\,\alpha(0) = 0.5 \times 0.5 = 0.25$$



Y1
Y2
0.475
0.025
0.3
0.45
0.95
0.35
0.25
0.15
X1
X2
X3
X4

$$+(0.8-0)^2+(0.9-0)^2$$
$$=0.81+0.04+0.64+0.81$$
$$=2.3$$

$$D(2)=\sum_{i=1}^{4}(w_{i2}-x_i)^2$$
$$=(0.9-1)^2+(0.7-0)^2$$
$$+(0.5-0)^2+(0.3-0)^2$$
$$=0.01+0.49+0.25+0.09$$
$$=0.84$$

Step 3: Since $D(2)<D(1)$, therefore $D(2)$ is minimum. Hence the winning cluster unit is Y_2, i.e., $J=2$.

Step 4: Update the weights on the winning cluster unit $J=2$:

$$w_{ij}(\text{new})=w_{ij}(\text{old})\ +\alpha[x_i-w_{ij}(\text{old})]$$
$$w_{i2}(\text{new})=w_{i2}(\text{old})+0.5[x_i-w_{i2}(\text{old})]$$
$$w_{12}(n)=w_{12}(0)+0.5[x_1-w_{12}(0)]$$
$$=0.9+0.5(1-0.9)=0.95$$
$$w_{22}(n)=w_{22}(0)+0.5[x_2-w_{22}(0)]$$
$$=0.7+0.5(0-0.7)=0.35$$
$$w_{32}(n)=w_{32}(0)+0.5[x_3-w_{32}(0)]$$
$$=0.5+0.5(0-0.5)=0.25$$
$$w_{42}(n)=w_{42}(0)+0.5[x_4-w_{42}(0)]$$
$$=0.3+0.5(0-0.3)=0.15$$

The updated weight matrix after presentation of second input pattern is

$$w_{ij}=\begin{bmatrix}0.1 & 0.95\\ 0.2 & 0.35\\ 0.8 & 0.25\\ 0.9 & 0.15\end{bmatrix}$$

Third input vector:

Step 1: For $x=[0\ 1\ 1\ 0]$, perform Steps 2–4.

Step 2: Calculate the Euclidean distance:

$$D(j)=\sum_i(w_{ij}-x_i)^2$$
$$D(1)=\sum_{i=1}^{4}(w_{i1}-x_i)^2$$

$$= (0.1-0)^2 + (0.2-1)^2 + (0.8-1)^2 + (0.9-0)^2$$

$$= 0.01 + 0.64 + 0.04 + 0.81 = 1.5$$

$$D(2) = \sum_{i=1}^{4} (w_{i2} - x_i)^2$$

$$= (0.95-0)^2 + (0.35-1)^2 + (0.25-1)^2 + (0.15-0)^2$$

$$= (0.9025) + (0.4225) + (0.5625) + (0.0225) = 1.91$$

Step 3: Since $D(1) < D(2)$, therefore $D(1)$ is minimum. Hence the winning cluster unit is Y_1, i.e., $J = 1$.

Step 4: Update the weights on the winning cluster unit $J = 1$:

$$w_{ij}(\text{new}) = w_{ij}(\text{old}) + \alpha[x_i - w_{ij}(\text{old})]$$

$$w_{i1}(\text{new}) = w_{i1}(\text{old}) + 0.5[x_i - w_{i1}(\text{old})]$$

$$w_{11}(n) = w_{11}(0) + 0.5[x_1 - w_{11}(0)]$$
$$= 0.1 + 0.5(0-0.1) = 0.05$$

$$w_{21}(n) = w_{21}(0) + 0.5[x_2 - w_{21}(0)]$$
$$= 0.2 + 0.5(1-0.2) = 0.6$$

$$w_{31}(n) = w_{31}(0) + 0.5[x_3 - w_{31}(0)]$$
$$= 0.8 + 0.5(1-0.8) = 0.9$$

$$w_{41}(n) = w_{41}(0) + 0.5[x_4 - w_{41}(0)]$$
$$= 0.9 + 0.5(0-0.9) = 0.45$$

The weight update after presentation of third input pattern is

$$w_{ij} = \begin{bmatrix} 0.05 & 0.95 \\ 0.6 & 0.35 \\ 0.9 & 0.25 \\ 0.45 & 0.15 \end{bmatrix}$$

Fourth input vector:

Step 1: For $x = [0\ 0\ 0\ 1]$, perform Steps 2–4.

Step 2: Compute the Euclidean distance:

$$D(j) = \sum_{i=1}^{4} (w_{ij} - x_i)^2$$

$$D(1) = \sum_{i=1}^{4}(w_{i1} - x_i)^2$$
$$= (0.05-0)^2 + (0.6-0)^2 + (0.9-0)^2 + (0.45-1)^2$$
$$= 0.0025 + 0.36 + 0.81 + 0.3025$$
$$= 1.475$$

$$D(2) = \sum_{i=1}^{4}(w_{i2} - x_i)^2$$
$$= (0.95-0)^2 + (0.35-0)^2 + (0.25-0)^2 + (0.15-1)^2$$
$$= (0.9025) + (0.1225) + (0.0625) + (0.7225)$$
$$= 1.81$$

Step 3: Since $D(1) < D(2)$, therefore $D(1)$ is minimum. Hence the winning cluster unit is Y_1, i.e., $J = 1$.

Step 4: Update the weights on the winning cluster unit $J = 1$:

$$w_{ij}(\text{new}) = w_{ij}(\text{old}) + \alpha[x_i - w_{ij}(\text{old})]$$
$$w_{i1}(\text{new}) = w_{i1}(\text{old}) + 0.5[x_i - w_{i1}(\text{old})]$$
$$w_{11}(n) = w_{11}(0) + 0.5[x_1 - w_{11}(0)]$$
$$= 0.05 + 0.5(0 - 0.05) = 0.025$$
$$w_{21}(n) = w_{21}(0) + 0.5[x_2 - w_{21}(0)]$$
$$= 0.6 + 0.5(0 - 0.6) = 0.3$$
$$w_{31}(n) = w_{31}(0) + 0.5[x_3 - w_{31}(0)]$$
$$= 0.9 + 0.5(0 - 0.9) = 0.45$$
$$w_{41}(n) = w_{41}(0) + 0.5[x_4 - w_{41}(0)]$$
$$= 0.45 + 0.5(1 - 0.95) = 0.475$$

The final weight obtained after the presentation of fourth input pattern is

$$w_{ij} = \begin{bmatrix} 0.025 & 0.95 \\ 0.3 & 0.35 \\ 0.45 & 0.25 \\ 0.475 & 0.15 \end{bmatrix}$$

Since all the four given input patterns are presented, this is end of first iteration or 1-epoch. Now the learning rate can be updated as

$$\alpha(t+1) = 0.5\alpha(t)$$
$$\alpha(1) = 0.5\,\alpha(0) = 0.5 \times 0.5 = 0.25$$

F

With this learning rate you can proceed further up to 100 iterations or till radius becomes zero or the weight matrix reduces to a very negligible value. The net with updated weights is shown by Figure 3.