called a Hamming memory network.

5.3 Kohonen Self-Organizing Feature Maps

5.3.1 Theory

Feature mapping is a process which converts the patterns of arbitrary dimensionality into a response of one- or two-dimensional arrays of neurons, i.e., it converts a wide pattern space into a typical feature space. The network performing such a mapping is called feature map. Apart from its capability to reduce the higher dimensionality, it has to preserve the neighborhood relations of the input patterns, i.e., it has to obtain a topology preserving map. For obtaining such feature maps, it is required to find a self-organizing neural array which consists of neurons arranged in a one-dimensional array or a two-dimensional array. To depict this, a typical network structure where each component of the input vector x is connected to each of the nodes is shown in Figure 5-5.

On the other hand, if the input vector is two-dimensional, the inputs, say $x(a, b)$, can arrange themselves in a two-dimensional array defining the input space (a, b) as in Figure 5-6. Here, the two layers are fully connected.

The topological preserving property is observed in the brain, but not found in any other artificial neural network. Here, there are m output cluster units arranged in a one- or two-dimensional array and the input signals are n-tuples. The cluster (output) units' weight vector serves as an exemplar of the input pattern that is associated with that cluster. At the time

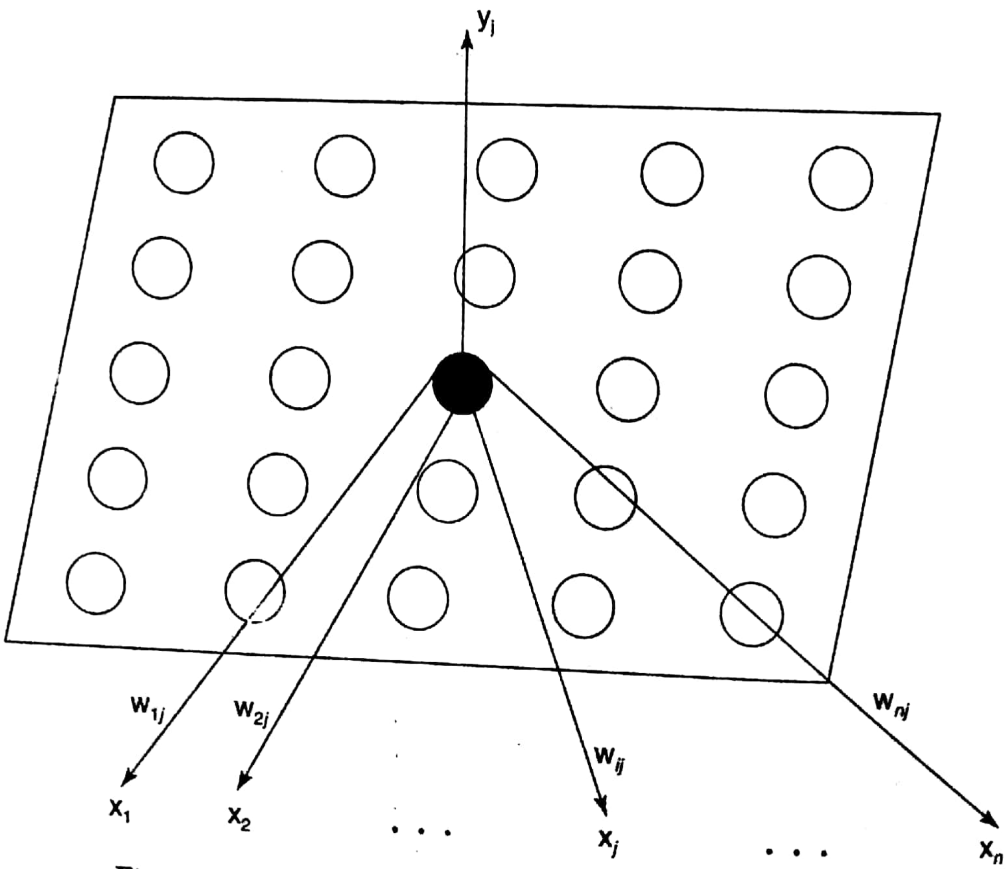

Figure 5-5 One-dimensional feature mapping network.

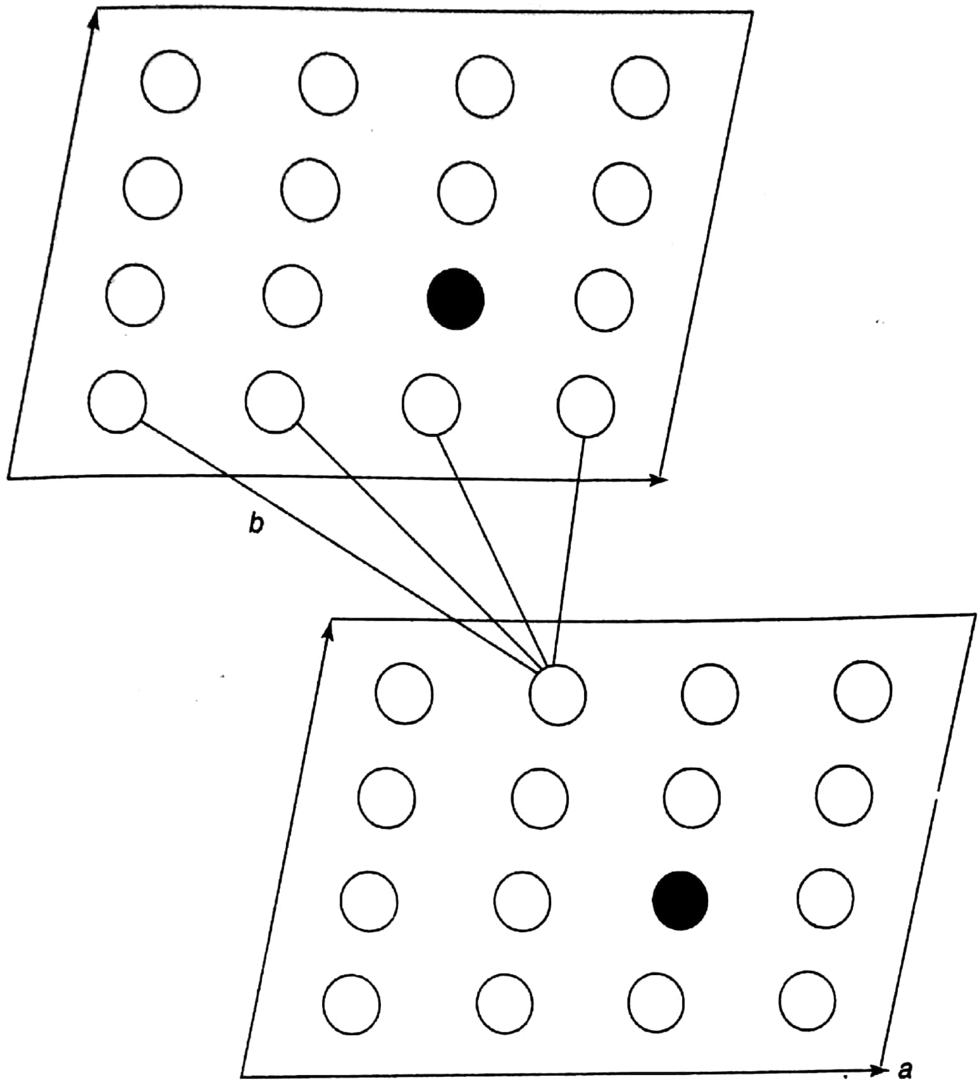

Figure 5-6 Two-dimensional feature mapping network.

of self-organization, the weight vector of the cluster unit which matches the input pattern very closely is chosen as the winner unit. The closeness of weight vector of cluster unit to the input pattern may be based on the square of the minimum Euclidean distance. The weights are updated for the winning unit and its neighboring units. It should be noted that the weight vectors of the neighboring units are not close to the input pattern and the connective weights do not multiply the signal sent from the input units to the cluster units until dot product measure of similarity is being used.

5.3.2 *Architecture*

Consider a linear array of cluster units as in Figure 5-7. The neighborhoods of the units designated by "o" of radii $N_i(k_1)$, $N_i(k_2)$ and $N_i(k_3)$, $k_1 > k_2 > k_3$, where $k_1 = 0, k_2 = 1, k_3 = 2$.

For a rectangular grid, a neighborhood (N_i) of radii k_1, k_2 and k_3 is shown in Figure 5-8 and for a hexagonal grid the neighborhood is shown in Figure 5-9. In all the three cases (Figures 5-7–5-9), the unit with "#" symbol is the winning unit and the other units are indicated by "o." In both rectangular and hexagonal grids, $k_1 > k_2 > k_3$, where $k_1 = 2, k_2 = 1, k_3 = 0$.

For rectangular grid, each unit has eight nearest neighbors but there are only six neighbors for each unit in the case of a hexagonal grid. Missing neighborhoods may just be ignored. A typical architecture of Kohonen self-organizing feature map (KSOFM) is shown in Figure 5-10.

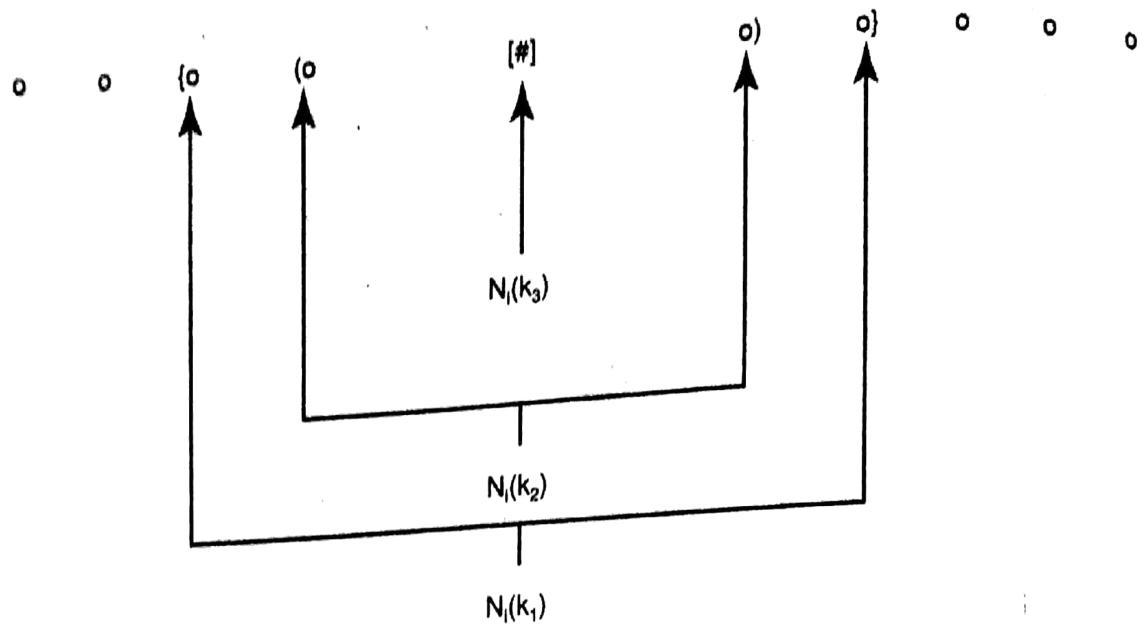

Figure 5-7 Linear array of cluster units.

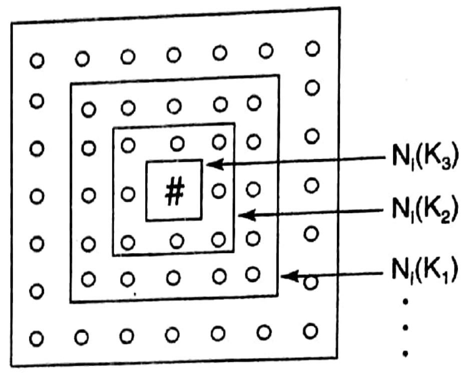

Figure 5-8 Rectangular gird.

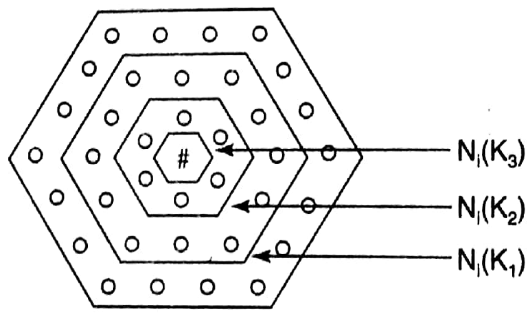

Figure 5-9 Hexagonal grid.

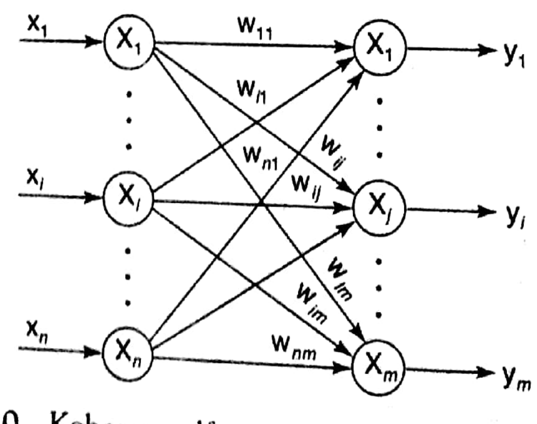

Figure 5-10 Kohonen self-organizing feature map architecture.

5.3.3 Flowchart

The flowchart for KSOFM is shown in Figure 5-11, which indicates the flow of training process. The process is continued for particular number of epochs or the learning rate reduces to a very small rate.

The architecture consists of two layers: input layer and output layer (cluster). There are "n" units in the input layer and "m" units in the output layer. Basically, here the winner unit is identified by using either dot product or Euclidean distance method and the weight updation using Kohonen learning rules is performed over the winning cluster unit.

5.3.4 Training Algorithm

The steps involved in the training algorithm are as shown below.

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,

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

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 times.

Step 8: Test for stopping condition of the network.

Thus using this training algorithm, an efficient training can be performed for an unsupervised learning network.

5.3.5 Kohonen Self-organizing Motor Map

The extension of Kohonen feature map for a multilayer network involves the addition of an association layer to the output of the self-organizing feature map layer. The output node is found to associate the desired output values with certain input vectors. This type of architecture

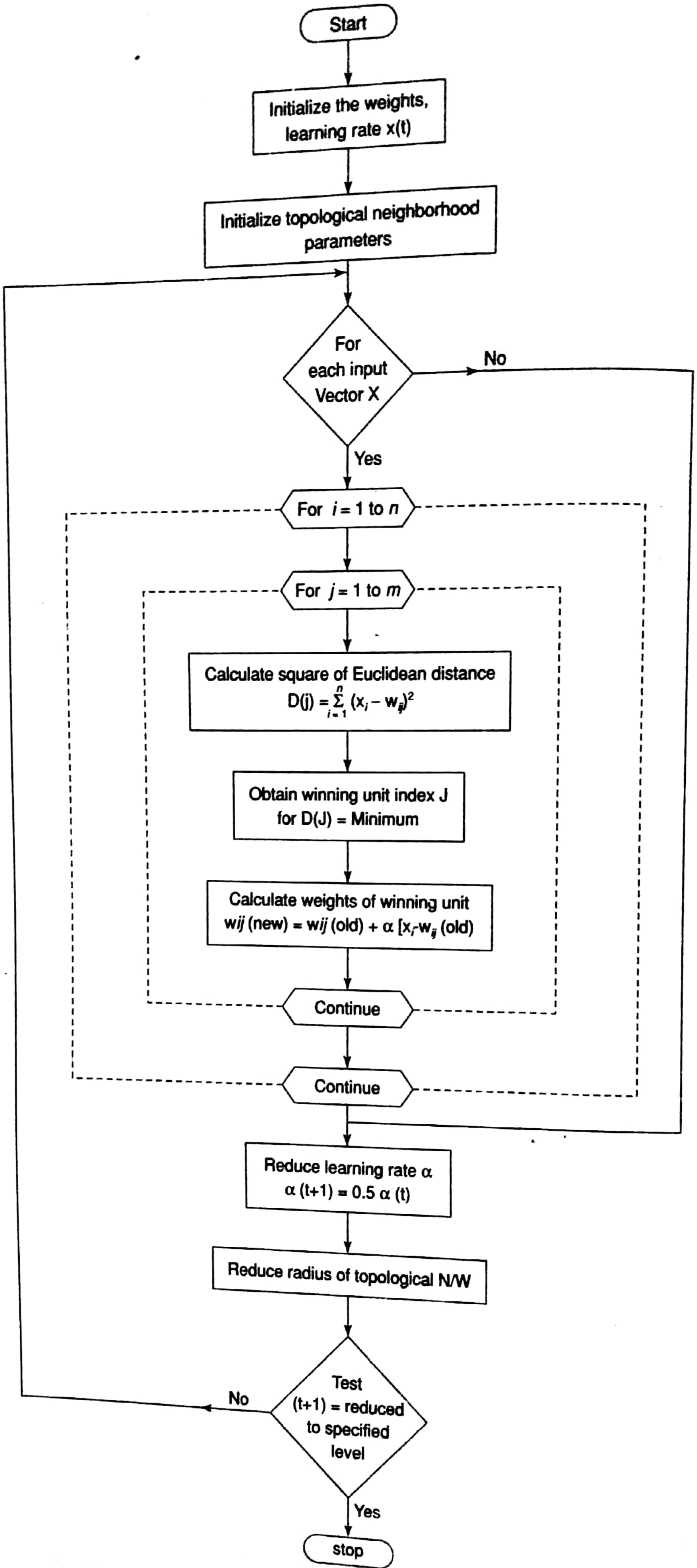

Figure 5-11 Flowchart for training process of KSOFM.

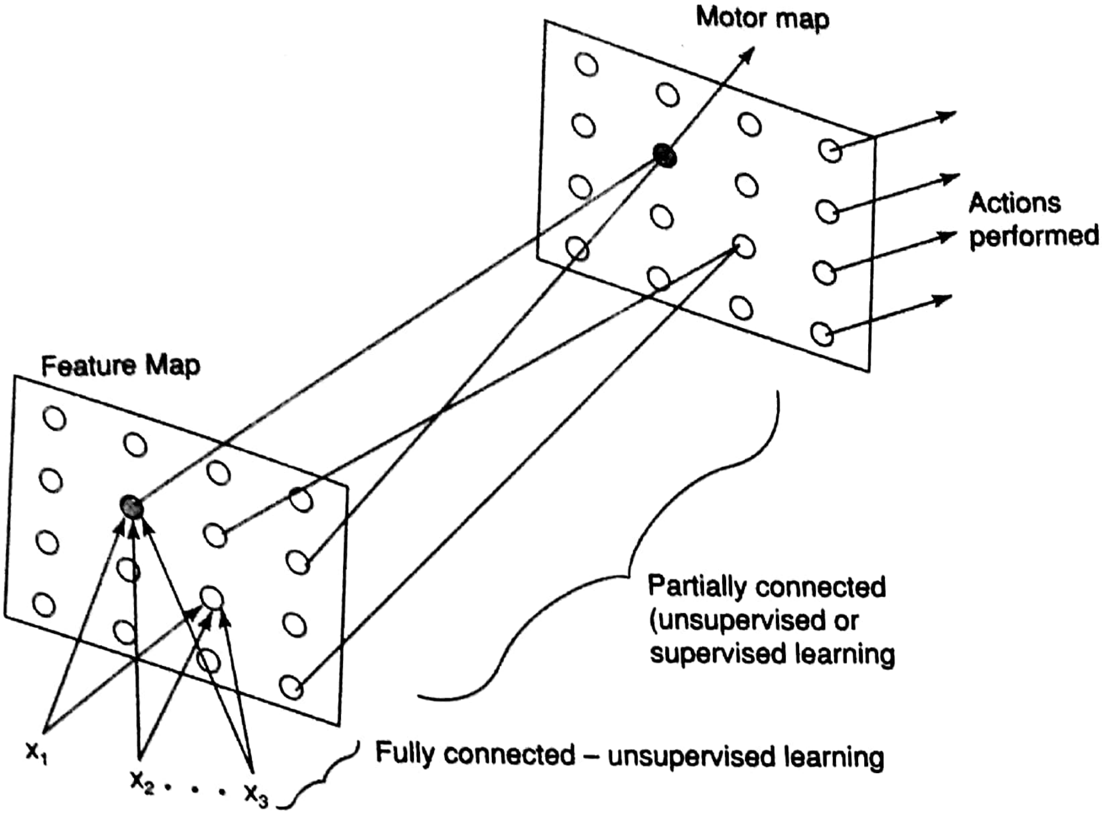

Figure 5-12 Architecture of Kohonen self-organizing motor map.

is called as Kohonen self-organizing motor map (KSOMM; Ritter, 1992) and layer that is added is called a motor map in which the movement commands are being mapped into two-dimensional locations of excitation. The architecture of KSOMM is shown in Figure 5-12. Here, the feature map is a hidden layer and this acts as a competitive network which classifies the input vectors. The feature map is trained as discussed in Section 5.3.3. The motor map formation is based on the learning of a control task. The motor map learning may be either supervised or unsupervised learning and can be performed by delta learning rule or outstar learning rule (to be discussed later). The motor map learning is an extension of Kohonen's original learning algorithm.