Tik-61.261 Principles of Neural Computing

Raivio, Venna

Exercise 12 21.4.2003

1. It is sometimes said that the SOM algorithm preserves the topological relationships that exist in the input space. Strictly speaking, this property can be guaranteed only for an input space of equal or lower dimensionality than that of the neural lattice. Discuss the validity of this statement. (Haykin, Problem 9.3)

2. It is said that the SOM algorithm based on competitive learning lacks any tolerance against hardware failure, yet the algorithm is error tolerant in that a small perturbation applied to the input vector causes the output to jump from the winning neuron to a neighboring one. Discuss the implications of these two statements. (Haykin, Problem 9.4)

3. In this problem we consider the optimized form of the learning vector quantization algorithm (see Section 9.7, Haykin) developed by Kohonen. We wish to arrange for the effects of the corrections to the Voronoi vectors, made at different times, to have equal influence when referring to the end of the learning period.

   (a) First, show that Equation (9.30)

   $$\mathbf{w}_c(n+1) = \mathbf{w}_c(n) + \alpha_n[\mathbf{x}_i - \mathbf{w}_c(n)]$$

   and Equation (9.31)

   $$\mathbf{w}_c(n+1) = \mathbf{w}_c(n) - \alpha_n[\mathbf{x}_i - \mathbf{w}_c(n)]$$

   may be integrated into a single equation, as follows:

   $$\mathbf{w}_c(n+1) = (1 - s_n\alpha_n)\mathbf{w}_c(n) + s_n\alpha_n\mathbf{x}(n).$$

   In the above equations, $\mathbf{w}_c$ is the Voronoi vector closest to the input vector $\mathbf{x}_i$, $0 < \alpha_n < 1$ is a learning constant, and s_n is a sign function depending on the classification result of the nth input vector $\mathbf{x}(n)$: $s_n = +1$ if classification is correct, otherwise $s_n = -1$.

   (b) Hence, show that the optimization criterion described at the beginning of the problem is satisfied if

   $$\alpha_n = (1 - s_n\alpha_n)\alpha_{n-1}$$

   which yields the optimized value of the learning constant α_n as

   $$\alpha_n^{\text{opt}} = \frac{\alpha_{n-1}^{\text{opt}}}{1 + s_n\alpha_{n-1}^{\text{opt}}}.$$

   (Haykin, Problem 9.6)

4. The following algorithm introduced by J. Friedman can be used for speeding up winner search in the SOM algorithm: 1) Evaluate the Euclidean distance between the input vector and weight vector whose projections on some coordinate axis are least apart, 2) Examine the weight vectors in the increasing order of their projected distances to the input vector. Continue this until a weight vector whose projected distance to the input vector is greater than the smallest Euclidean distance calculated so far is found. At this point of the algorithm, the winning neuron has been found.

   Apply the algorithm described above for the problem illustrated in Figure 1.

   (a) Find the winning neuron when the weight vectors and the input vector are projected onto x_1-axis. Show that the weight vector found by the algorithm is indeed the winning one. How many Euclidean distances one must evaluate for finding it?

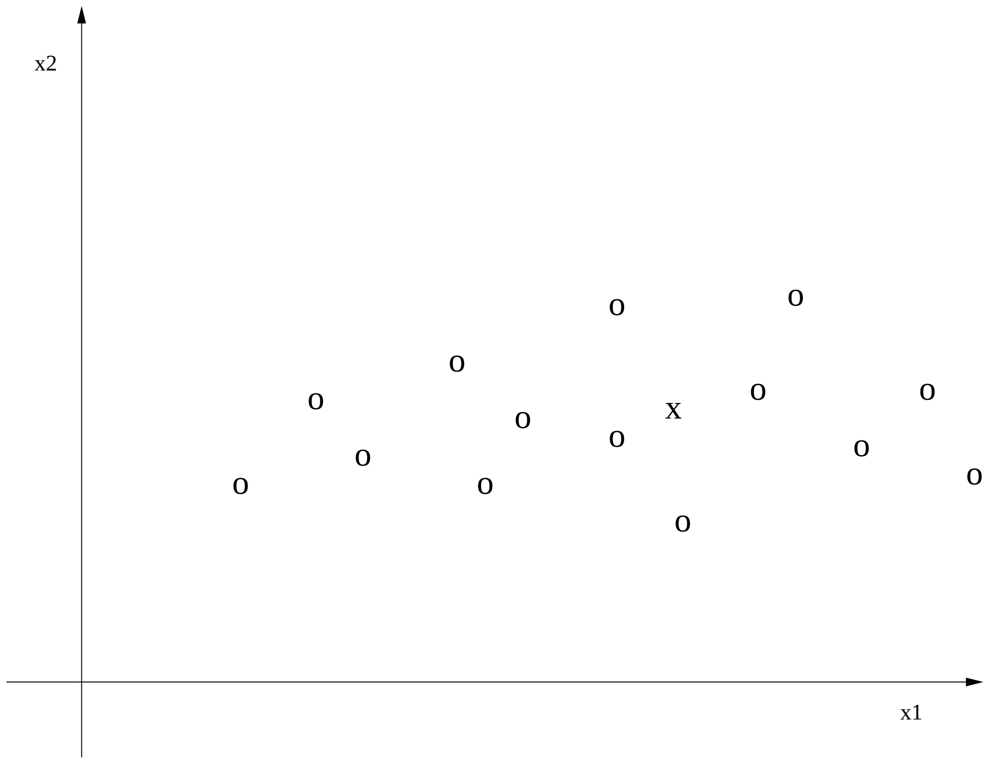

Figure 1: The weight vectors (o) among which the nearest one to the input vector (x) has to be found.

(b) Repeat the search but project the vectors onto x_2-axis this time.

(c) Which one of the two searches was the fastest? Are there some general rules on how the projection axis should be chosen?