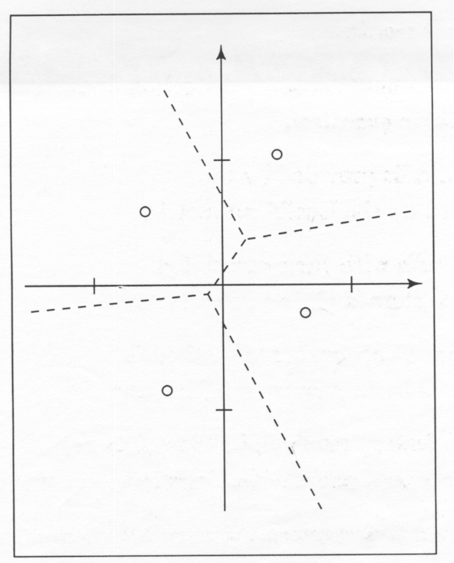
9.7 Learning Vector Quantization

- In vector quantization, the data (input) space is divided into a number of distinct regions.
- For each region, a reconstruction vector is defined.
- For a new incoming data vector, its region is determined at first.
- The data vector is then represented by using the reproduction vector for that region.
- Using an encoded version of this reproduction vector, considerable savings in storage or transmission bandwidth can be realized.
- The collection of possible reproduction vectors is called the *code book* of the vector quantizer.
- Its members are called *code words.*
- A vector quantizer with minimum encoding distortion is called a *Voronoi* or *nearest-neighbor quantizer*.

- *Voronoi cells* are partition cells provided by the nearest-neighbor rule based on the Euclidean metric.

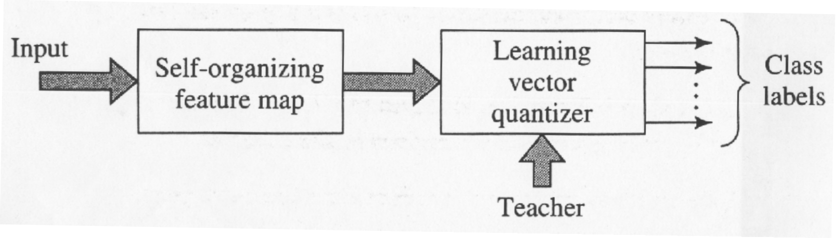


- An example of 4 Voronoi cells with their associated reconstruction (Voronoi) vectors.

- The SOM method provides an approximate method for computing the Voronoi cells in an unsupervised manner.

- Computation of the SOM feature map can be viewed as the first stage of adaptively solving a pattern classification problem.

- The second stage is learning vector quantization, which fine tunes the SOM feature map.



- *Learning vector quantization (LVQ)* is a supervised learning technique.
- Using class information, it moves the Voronoi vectors slightly for improving the decision regions of the classifier.
- Take an input vector $\mathbf{x}$ at random from the data space.
- If the class labels of $\mathbf{x}$ and a Voronoi vector $\mathbf{w}$ agree, $\mathbf{w}$ is moved in the direction of $\mathbf{x}$.
- If the class labels of $\mathbf{x}$ and $\mathbf{w}$ are different, $\mathbf{w}$ is moved away from the input vector $\mathbf{x}$.
- Assumption: there are many more input (data) vectors $\mathbf{x}_1, \ldots, \mathbf{x}_N$ than Voronoi vectors $\mathbf{w}_1, \ldots, \mathbf{w}_l$ ($N \gg l$).

The Learning Vector Quantization (LVQ) algorithm:

- Suppose that the Voronoi vector $\mathbf{w}_c$ is the closest to the input vector $\mathbf{x}_i$.
- Let $\mathcal{C}_{\mathbf{w}_c}$ denote the class of $\mathbf{w}_c$ and $\mathcal{C}_{\mathbf{x}_i}$ the class of $\mathbf{x}_i$.
- The Voronoi vector $\mathbf{w}_c$ is adjusted as follows:
  - If the classes are the same: $\mathcal{C}_{\mathbf{w}_c} = \mathcal{C}_{\mathbf{x}_i}$, then

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

  - If the classes are different: $\mathcal{C}_{\mathbf{w}_c} \neq \mathcal{C}_{\mathbf{x}_i}$, then

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

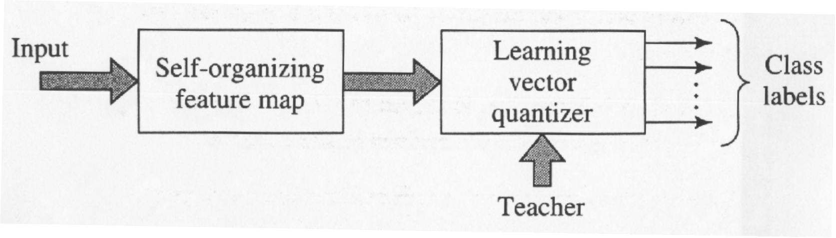
- The other (non-closest) Voronoi vectors are not changed.
- The learning parameter α_n usually decreases monotonically with the number of iterations n.

- For example, α_n may decrease linearly from its initial value 0.1.
- The Voronoi vectors typically converge after several epochs.

9.8 Computer Experiment: Adaptive Pattern Classification

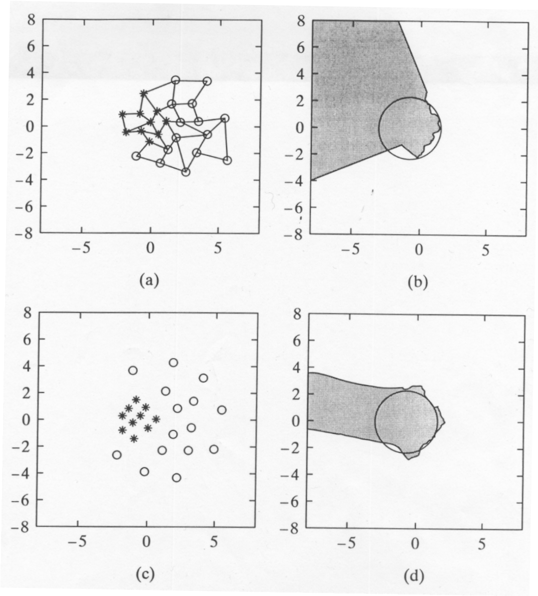
- A pattern classification task can be divided into feature selection (extraction) and actual class assignment steps.
- In *feature selection*, a reasonably small set of features containing the essential information needed for classification is sought.
- This important step is usually performed using some unsupervised method.
- The self-organizing map is well suited to feature selection.
- It can extract nonlinear features describing the data.
- After feature extraction, any suitable classification method can be used.
- Usually some supervised classifier trained using known prototype patterns is applied for achieving the best performance.

- An adaptive hybrid pattern classification approach: SOM + LVQ.



- Recall the classification test problem introduced in Section 4.8.

- Two two-dimensional overlapping Gaussian distributed classes.



(a) Labeled two-dimensional 5×5 SOM map after training.
(b) decision boundary given by SOM only.
(c) Labeled map after LVQ fine tuning.
(d) decision boundary given by combined use of SOM and LVQ.

- A comparison of Fig. a with c and Fig. b with d shows qualitatively the advantage of fine tuning using LVQ.

- Summary of the Classification Performances (percentage) for the Computer Experiment on Overlapping Two-Dimensional Gaussian Distribution Using 5×5 Lattice

Trial	SOM	Cascade combination of SOM and LVQ
1	79.05	80.18
2	79.79	80.56
3	79.41	81.17
4	79.38	79.84
5	80.30	80.43
6	79.55	80.36
7	79.79	80.86
8	78.48	80.21
9	80.00	80.51
10	80.32	81.06
Average	79.61	80.52

- The use of LVQ improves the performance in all the 10 trials.

- The average percentage of correct classification is:
  - 79.61% for SOM only
  - 80.52% for combined SOM and LVQ
  - 81.51% for the optimal Bayes classifier.

9.11 Summary and Discussion

- This section describes briefly some theoretical results derived for SOM.
- Generally, it is very difficult to analyze SOM rigorously.
- The results on convergence etc. are mainly for one-dimensional lattices only.