Feature Selection with Self-Organizing Feature Map

Jukka Iivarinen, Kimmo Valkealahti, Ari Visa, Olli Simula
Helsinki University of Technology
Laboratory of Information and Computer Science
Rakentajanaukio 2 C, FIN-02150 Espoo
Finland

1 Introduction

Feature selection is an important phase in pattern recognition. It has also an important role in neural network methods even though it is often neglected. The fundamental function of feature selection is to find a set of features that will represent the pattern vector in a most optimal way. Only the information that is either redundant or irrelevant to the classification task is removed from the pattern vectors. The dimension of the feature vector is usually smaller than the dimension of the pattern vector so the computation time and the memory requirements are greatly reduced.

Feature selection provides also an indication about the discriminatory potential of the features. Thus it could be used as a tool to determine whether it is necessary to look after additional features that would allow the improvement of the performance of the classifier [Devijler et al. 1982].

It is not a trivial problem to find the best set of features or the best transformation. The features must be selected according to the given problem. For feature selection the principal component analysis (PCA), also known as the Karhunen-Loéve expansion, is often applied. However, it is not very illustrative. A covering collection of transformations that are used in image processing applications is found in reference [Jain 1989].

In this paper a new tool to feature selection is described. This approach is based on the visualization capability of the Self-Organizing Feature Map.

2 Feature Selection with Self-Organizing Feature Map

Some neural network methods can also be used in feature selection. The suggested method is based on the Self-Organizing Feature Map [Kohonen 1990].

SOM is used in a feature selection as follows. Take n features. Train the map with a large set of these features. Label the map with a set of preclassified samples. If the classes in the labeled map overlap add a new feature or replace a feature with a new one. If the feature does not improve the separability of the classes, the feature will be replaced. Train the map again with a set of new features and label the map. Repeat this procedure until you have a set of features that can reach the desired classification rate. This is a practical method to find a set of features. It is straightforward and easy to implement,

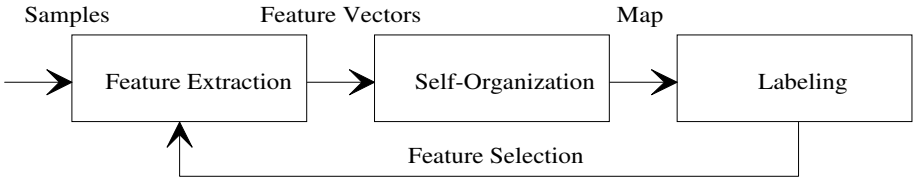

Figure 1: Block diagram of the feature selection with SOM.

and the need of preclassified samples is minimized. The method is illustrated in Figure 1. The features that are chosen to be tested, are found among the features that are suggested in literature.

SOM is a topology-preserving feature map that represents the distribution of features in the feature space. In SOM similar feature vectors are mapped to adjacent map elements. The map is trained in an unsupervised way. During the training phase the map is fed with a large number of unclassified feature vectors.

For a feature vector $\mathbf{x}$ the best matching map element $\mathbf{m}_c$ is presented as an output. The best matching element is determined by

$$\|\mathbf{x} - \mathbf{m}_c\| = \min_i \{\|\mathbf{x} - \mathbf{m}_i\|\} \tag{1}$$

During the training phase the best matching element and its neighborhood are updated. The neighborhood $N_c(t)$ is defined to include all the elements inside a certain distance from the central element. The training process can then be expressed as

$$\mathbf{m}_i(t+1) = \begin{cases} \mathbf{m}_i(t) + \alpha(t)[\mathbf{x}(t) - \mathbf{m}_i(t)] & \text{if } i \in N_c(t) \\ \mathbf{m}_i(t) & \text{if } i \notin N_c(t) \end{cases} \tag{2}$$

where $\alpha(t)$ is a monotonically decreasing scalar function, $0 < \alpha(t) < 1$, that determines the learning rate. The neighborhood $N_c(t)$ is dependent on time t. In the beginning of the learning $N_c(t)$ is quite wide but decays into $N_c(t) = \{\mathbf{m}_c\}$.

After the training phase the map is usually labeled. A set of preclassified feature vectors is fed into the map and the best matching element to each feature vector is calculated. Each map element can be identified to represent a class. The elements that have responded only to one class can be labeled directly to the respective class. If an element gives response to samples of several classes, the labeling can be done according to the majority decision. In this case reselection of features is necessary.

3 Case : Feature Selection for Cloud Classification from NOAA Satellite Images

The suggested method is suitable for problems where exist a lot of samples and a few preclassified samples. The cloud classification is a good example of that.

The cloud images are collected through the Advanced Very High Resolution Radiometer (AVHRR) on board of the NOAA-11 polar orbiting satellite

Cloud Genus	*Number of samples*	*Results with 9 features [%]*	*Results with 14 features [%]*
Cirrus over sea/land	13	84.6	100.0
Cirrus over low clouds	9	88.9	100.0
Thick Cirrostratus	15	66.7	73.3
Altostratus/Altocumulus	14	57.1	92.9
Stratus/Stratocumulus	12	100.0	91.7
Cumulus	9	100.0	100.0
Cumulonimbus	20	95.0	95.0
Nimbostratus	15	60.0	100.0
Total	107	80.4	93.5

Table 1: The test results based on preclassified cloud samples as a function of the number of features.

[Visa et al. 1991]. Experienced meteorologists from the Finnish Meteorological Institute have collected 218 preclassified samples. The samples were collected from 43 summer day images and they contained small well-defined regions of one cloud type. The samples were split into two separate sets, a learning set and a test set. The learning set was used in labeling the map. The same set was also used in testing the map in order to determine the discriminatory potential of the selected features. The goal for the classification rate of the learning samples was 90 per cent. If the goal were set to 100 per cent the map would become too tuned to the learning set.

The suggested features contain both texture and spectral features. Texture features are chosen among the Haralick features [Haralick et al. 1973] and spectral features among the features suggested in reference [Gu et al. 1991].

The size of the Self-Organizing Feature Map is crucial. If the map is too small, there does not exist enough elements to represent each class, and the classes will overlap. The map size was 11-by-14 elements which was obtained by experience.

The training of the map was done in two phases. First the map was topologically ordered with 60000 unclassified samples. Learning rate $\alpha(t)$ was 0.6 and neighborhood $N_c(t)$ was 10. During the second phase the map was fine-tuned with 200000 unclassified samples. Learning rate was 0.2 and neighborhood was 4.

Learning rate $\alpha(t)$ has a great impact during the training phase. If the initial value of $\alpha(t)$ is too small, the map will not be able to organize properly. The learning rate is smaller in the second phase.

The initial size of the neighborhood $N_c(t)$ is also important. If $N_c(t)$ is too small in the beginning, the map is not organized properly. In the first phase, $N_c(t)$ is usually taken almost equal to the diameter of the map. In the second phase, however, it is smaller.

The feature selection is made as in Figure 1. After the evaluation of the suggested texture features, the nine texture features were selected. The classification rate was 80 per cent for the learning samples. The suggested spectral features were evaluated next. Finally the nine texture features and the five

spectral features were selected. The classification rate was 94 per cent for the learning samples. This can be considered as a sufficient result. The test results are given in Table 1.

The map was also tested with the test samples. The test samples were classified 79 per cent right with all the features. This shows that the map is not too tuned to the learning samples and it can be used in a real classification task.

4 Conclusions

The suggested method is a practical tool for feature selection. It does not select features but merely tests the discriminatory potential of features. The features that are chosen to be tested should be found by experience. SOM is an illustrative tool in the sense that it shows the clustering effect of features.

Only a few preclassified samples are needed. However, thousands of unclassified samples are needed during the training phase. This is a minor drawback because unclassified samples are usually easy to obtain.

The size of the map, the learning rate $\alpha(t)$ and the neighborhood $N_c(t)$ must be chosen carefully according to empirical experiments.

The suggested method is general in the sense that it is not dependent on the features. It also offers an efficient way to visualize high-dimensional feature vectors.

References

[Devijler et al. 1982] Devijler P. and Kittler J. (1982), Pattern Recognition: A Statistical Approach. Prentice-Hall International, Inc.,London.

[Gu et al. 1991] Gu Z. Q., Duncan C. N., Grant P. M., Cowan C. F. N., Renshaw E., Mugglestone M. A. (1991), Textural and spectral features as an aid to cloud classification. Int. J. Remote Sensing, Vol. 12, No. 5, pp. 953-968.

[Haralick et al. 1973] Haralick R. M., Shanmugan K., Dinstein I. (1973), Textural Features for Image Classification. IEEE Transactions on Systems, Man and Cybernetics, Vol. SMC-3, No. 6, November.

[Jain 1989] Jain A. K. (1989), Fundamentals of Digital Image Processing. Prentice Hall, Englewood Cliffs.

[Kohonen 1990] Kohonen T. (1990), The Self-Organizing Map. Proceedings of the IEEE, Vol. 78, No.9, September.

[Visa et al. 1991] Visa A., Valkealahti K., Simula O. (1991), Cloud Detection Based on Texture Segmentation by Neural Network Methods. Proceedings of IEEE International Joint Conference on Neural Networks, Singapore, November 18-21, Vol. 2, pp. 1001-1006.