BULGARIAN ACADEMY OF SCIENCES

CYBERNETICS AND INFORMATION TECHNOLOGIES • Volume **6**, No 1

Sofia • 2006

A Self-Organizing Neural Network for Variable Convex Clusterization

Valeri Ilchev, Zlatoliliya Ilcheva***

** Institute of Information Technologies, 1113 Sofia,*
E-mail: ilch@iinf.bas.bg
*** Institute of Computer and Communication Systems, 1113 Sofia,*
E-mail: zlat@agatha.iac.bg

Abstract: *A method for more compact description of clusters with arbitrary form in comparison with Kohonen's or Kohonen-Hebb's methods is introduced in this paper.*

The neural network (NN), based on the described method, uses NN of Kohonen-Hebb for initial determination of the Mahalanobis distance between the points in the cluster, after which it find the biggest convex core inside the cluster. The difference between this core and the cluster, containing it, creates a group of sub-clasters to which the same method is applied. In this way, the arbitrary form of the cluster is divided into separate convex sub-areas – one central and several peripheral. Each one of them is fixed on the map of the self-organizing NN.

Keywords: *self-organizing NN, cluster analysis, Kohonen's NN with hyperelipsoidal clustering, pattern recognition, image analysis.*

1. Introduction

Considering the problems of clusterization by means of NN, the self-organizing NN of Kohonen and its modifications are most widely used. Since by these problems lack preliminary information for the number and situation of the clusters, the NN has to be trained in such a way that the set of similar vectors to activate one and the same output neural element, in order to solve the task for division of the initial clusters successfully [1]. For this purpose, having a well-trained NN, the scalar product of each input vector with the vectors of the coefficients of the separate clusters is calculated:

(1) $$Y_k = \mathbf{W}_k^{\mathrm{T}}\mathbf{X},$$

where $\mathbf{W}_k = (w_{1k}, \ldots, w_{nk})$ is the vector weight coefficients, corresponding to the K-cluster, $\mathbf{X} = (x_1, ..., x_n)$, $\mathbf{X} \in \boldsymbol{R}^n$ – input vector, defined in the n-dimensional Euclidean feature space, $\mathbf{Y}_k$ – output of the k-neuron.

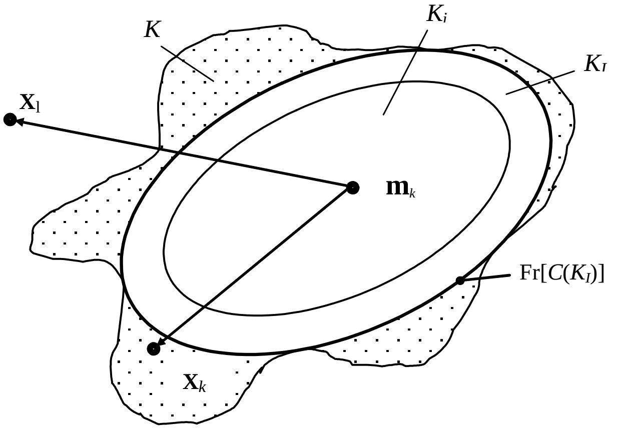

Fig. 1

At this type of NN, the scalar product (1) serves as a measure for closeness between the input vector and the centers of the clusters, already defined in the space $\boldsymbol{R}^n$, shown in Fig. 1. As is well known [2, 3], one of the basic ideas of the training method is that the scalar product (1) will satisfy the condition $Y_k(\mathbf{X}_t) > Y_l(\mathbf{X}_t)$ for $\mathbf{X}_t \in K$ and $\mathbf{X}_t \notin L$, where K and L are respectively the K- and the L-cluster. The output $Y_k(\mathbf{X}_t)$, satisfying the inequality, will be considered as a “winner” and will be activated, while the other outputs will be inactive. The correction of the weight vector $\mathbf{W}_k$ will be carried out only for the cluster K, whose output is active:

$$w_{ik}(q + 1) = w_{ik}(q) + \alpha[x_i - w_{ik}(q)],$$

where $i = 1, \ldots, n$, q is an iteration index.

By the well trained NN of this kind, the weight-vectors w_{ik} will be modified insignificantly [4]: $w_{ik}(q + 1) - w_{ik}(q) \to 0$, from where for the mathematical expectation we will have: $\mathbf{M}[X_i^k - w_{ik}(q)] \to 0$. This means, that $x_i^k - w_{ik}(q) \to \bar{x}_i^k \Rightarrow w_{ik}(q) \to m_{ik}$ for $q \to \infty$, where $\bar{x}_i^k$ is a component of the central input vector, belonging to the K-cluster; m_{ik} is a component of the vector of mathematical expectation: $\mathbf{m}_k = (m_{1k}, \ldots, m_{nk})$ of the cluster K. The Euclidean distance of each point $\mathbf{X}_t \in K$ from the center $\mathbf{m}_k$ of the cluster K will be: $\mathrm{D}_{kt}^E = (\mathbf{X}_t - \mathbf{m}_k)^{\mathrm{T}}(\mathbf{X}_t - \mathbf{m}_k)$. If we define the inequality for this distance: $\mathrm{D}_{kt}^E \leq Q_k$, we will obtain a hypersphere, containing, depending on the value of the threshold Q_k, all points of the cluster K or the most of them [5].

Obviously, for the clusters with a form, close to the spheroidal, the distance will define a relatively compact envelope of the corresponding cluster, but for clusters with non-regular form this distance will not be optimal.

In [6] is supposed comparatively more compact clasterization method by means of Mahalanobis distance: $\mathrm{D}_{kt}^M = (\mathbf{X}_t - \mathbf{m}_k)^{\mathrm{T}}(\mathbf{X}_t - \mathbf{m}_k)\mathbf{M}_k^{-1}$; where $\mathbf{M}_k^{-1}$ is an inverse matrix to the covaritional one $\mathbf{M}_k = \{\sigma_{ij}\}$, $i, j = 1, ..., n$, and $\sigma_{ij} = \sum_{t \in T}(x_{ti} - \bar{x}_i)(x_{tj} - \bar{x}_j)/T$;

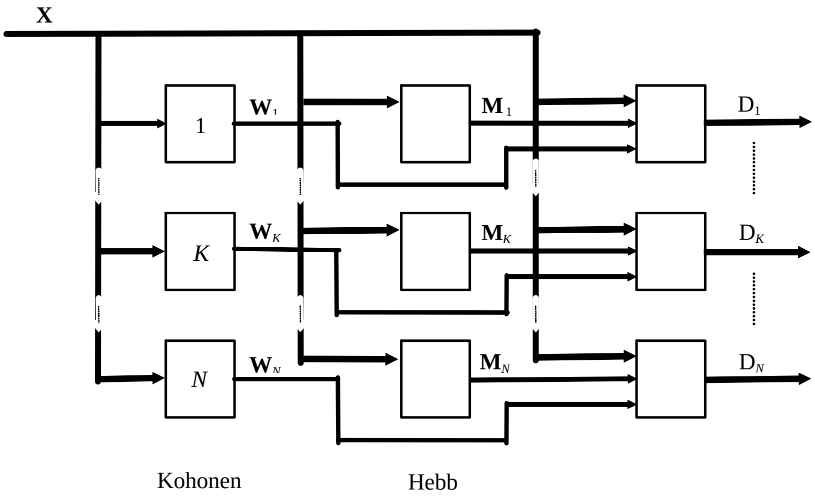

Fig. 2

$\overline{x}_i$, $\overline{x}_j \in \mathbf{X}_t \in K$ and $\overline{x}_i$ and $\overline{x}_j$ are the average values of the i and j coordinates of the current elements x_{ti} and x_{tj}, T is a number of the points in the cluster K [7].

The matrix $\mathbf{M}_k^{-1}$ is recurrently calculated by means of the method suggested in [6], which includes the method of principal component analysis (PCA) using the Hebbian rule [4]. The NN, which realizes the finding of the Mahalanobis distance D_{kt} is given in Fig. 2, where the blocks calculating the values of the $\mathbf{m}_k$-vectors and the $\mathbf{M}_k^{-1}$ matrices, are respectively NN of Kohonen and NN of Hebb [6].

In order to examine the basic properties of the Mahalanobis distance, we will first suppose that the cluster K contains a sufficient number of points, specified by their radius-vectors $\mathbf{X}_k$, with a clearly expressed linear dependence between them and a single-extremum function of the density of the probability distribution, such as the normal density of distribution or the cases of uniform distribution of the points in the cluster K.

2. Hyperellipsoidal clasterization

Let us specify for the distribution density function of the vectors $\mathbf{X}_k$ in the cluster K an intersection with a hyperplane, parallel to the hyperplane of the arguments $\mathbf{X}=(x_1, ..., x_n)$.

Then the points of this intersection will obviously have one and the same probability density and, as is well-known [8], the projection of these points onto the hyperplane $\mathbf{X}$ will be a hyperellipsoid, for which the square distance $D_{ks}^M = (\mathbf{X}_{ks} - \mathbf{m}_k)^T(\mathbf{X}_{ks} - \mathbf{m}_k)\mathbf{M}_k^{-1}$ will be constant for $\forall \mathbf{X}_{ks} \in \mathbf{S}_k$, where $\mathbf{S}_k$ is the set:

(2) $$\mathbf{S}_k = \{\mathbf{X}_{ks}: P(\mathbf{X}_{ks}) = \text{const}\}$$

and $P(\dots)$ is a function of the density of probability distribution.

For example, if the points $\mathbf{X}_{ks}$ of the cluster K have a normal density of probability distribution, so for $P(\mathbf{X}_{ks}) = P_s = \text{const}$ we will have the equation:

(3) $$P(\mathbf{X}_{ks}) = K\exp[-0.5(\mathbf{X}_{ks} - \mathbf{m}_k)^T (\mathbf{X}_{ks} - \mathbf{m}_k)\ \mathbf{M}_k^{-1}] = P_s$$

from where for $\mathbf{X}_{ks} \in \mathbf{S}_k$ we will obtain:

$$\ln(P_s) = -0.5(\mathbf{X}_{ks} - \mathbf{m}_k)^{\mathrm{T}}(\mathbf{X}_{ks} - \mathbf{m}_k)\mathbf{M}_k^{-1} + \ln K \Rightarrow (\mathbf{X}_{ks} - \mathbf{m}_k)^{\mathrm{T}}(\mathbf{X}_{ks} - \mathbf{m}_k)\mathbf{M}_k^{-1} =$$

$$2[\ln K - \ln(P_s)] = \ln(K/P_s)^2 = \mathbf{D}_{ks}^{M} = \text{const, where } K = \frac{1}{\sqrt{(2\pi)^n \,|\, M_k \,|}} = \text{const}.$$

In the cases of statistical independance and equal dispersion of the points, the covariation matrix will be diagonal and the Mahalanobis distance turns into an Euclidean one: $\mathrm{D}_k^M = \mathrm{D}_k^E$. This means that even if it is slower for calculating, the distance D_k^M will be more universal than the Euclidean one, because the distance D_k^M is determined by the average value of the vectors $\mathbf{m}_k$ and the covariation matrix $\mathbf{M}_{\mathrm{k}}$ for the chosen cluster K.

These two basic properties of the Mahalanobis distance can be also applied to clusters with various forms, containing an insufficiently large number of points, which often appear in the real problems. Because we can calculate for every cluster with random form the average value of its points $\mathbf{m}_k$ and its covariation matrix $\mathbf{M}_k$, so by means of these two generalized parameters we can calculate the distance D_k^M For this cluster which will define its hyperellipsoidal or in some special cases hyperspherical envelope. This greater universality of the Mahalanobis distance in comparison with the distance D_k^E is sufficient argument to use only D_k^E further in this paper, because of which we will mark it for shortness D_k .

3. Maximum convex clusterization

Definition. The maximum convex core of the cluster K (Fig. 1) will be called as inner core $K_I \subset K$, defined by the distance D_{kI} (or D_{kI}^E), which satisfies the conditions:

(4) $\quad P_{\mathrm{I}}(\mathbf{X}) - \mathrm{P}_{\mathrm{I}}(\mathbf{X}) \le \varepsilon$ and $M[K|C(K_{\mathrm{I}})] = M[K|C(K_i)]$

for $\forall \mathbf{X} \in \mathbf{S}_{\mathrm{I}} = \mathrm{Fr}[C(K_{\mathrm{I}})]$, where Fr[…] is the boundary of the convex envelope $C(K_{\mathrm{I}})$ and $P_{\mathrm{I}}(\mathbf{X})$ is an evaluation of the distribution density of the points on the $(n-1)$-dimensional convex hypersurface $\mathbf{S}_{\mathrm{I}}$; $P_{\mathrm{I}}(\mathbf{X}) = P_s(\mathbf{X})$; P_s is defined from condition (3), ε is a threshold constant, $C(K_{\mathrm{I}}) = \{\mathbf{Z}\colon \mathbf{Z} = \sum_{j=1}^{N} \lambda_j X_j \,,\ \sum_{j=1}^{N} \lambda_j = 1,\ \lambda_j \ge 0\,;\ \mathbf{X}_j \in K_{\mathrm{I}}$, $N = M(K_{\mathrm{I}})\}$, analogously is determined and $C(K_i)$, $K_i \subset K$, $i = 1, 2, \ldots$, and $M(\ldots)$ – power of the point set.

In the concrete example $P_{\mathrm{I}}(\mathbf{X})$ is specified by means of the Parzen window [8]:

$$(5) \qquad P_{\mathrm{I}}(\mathbf{X}) = \frac{1}{N}\sum_{k=1}^{N} \frac{1}{S}\,\varphi\!\left(\frac{\mathbf{X} - \mathbf{X}_k}{h}\right),$$

where $N = M(\mathbf{S}_{\mathrm{I}} \cap K)$ is the number of the points on the hypersurface $\mathbf{S}_{\mathrm{I}}$, $S = h^{n-1}$ is the Parzen window with a side-lehgtn h. The window function will be of the type:

$$\varphi(u) = \begin{cases} 1 \text{ for } |u_i| \le 1/2,\, i = 1, 2, \ldots, \\ 0, \text{ for the inverse case;} \end{cases}$$

for $\mathbf{X}_k \in S(\mathbf{X})$, $S(\mathbf{X})$ – an area with a center at the point $\mathbf{X}$, the function will be equal to 1 or to zero, if $\mathbf{X}_k \notin S(\mathbf{X})$.

In the real problems of clusterization in most cases we usually have not at our disposal the analytical form of the function, defining the distribution density $P(\mathbf{X})$, because of which we cannot use directly the evaluation a $P_{\mathrm{I}}(\mathbf{X}) - P_{\mathrm{I}}(\mathbf{X}) \leq \varepsilon$ while determining the kernel K_{I}. In order to avoid this lack of correspondence between the theoretical formulation of the definition of K_{I} and its real determination we will examine one of the properties of the distribution density $P_{\mathrm{I}}(\mathbf{X})$. If for the concrete cluster K_{I} we define the Mahalanobis distance $\mathbf{D}_k(\mathbf{X})$ = const, so from the conditions (2) and (3) above, it is clear, that $P_{\mathrm{I}}(\mathbf{X})$ = const for $\forall\mathbf{X}$, for which $\mathbf{D}_k(\mathbf{X})$ = const. Since this equation defines the n-dimensional hyperplane H_{I}, which is parallel to the hyperplane $\boldsymbol{R}^n$ and crosses the function $P(\mathbf{X})$, so $P_{\mathrm{I}}(\mathbf{X}) = P(\mathbf{X}) \cap H_{\mathrm{I}}$ and the intersection $P_{\mathrm{I}}(\mathbf{X})$ of the Gaus function $P(\mathbf{X})$ will obviously be a convex surface in H_{I}, which we will mark with $S_{\mathrm{I}}(\mathbf{X})$. Then its projection in the space $\boldsymbol{R}^n$ will be the translation $S_{\mathrm{I}}(\mathbf{X})$ in $\mathbf{R}^n$, which will also be a convex surface.

This property can be applied for an indirect determination of the convex kernel $K_{\mathrm{I.}}$. For this purpose we will divide the surface $S_{\mathrm{I}}(\mathbf{X})$ into sufficiently small equal size zones $S_t(\mathbf{X})$ such that $S_a \cap S_b = \varnothing$, $t = a, b$ and $\bigcup_t S_t(\mathbf{X}) = S_{\mathrm{I}}(\mathbf{X})$. As $P_{\mathrm{I}}(\mathbf{X})$ = const, so for $\forall\mathbf{X}_k \in S_t(\mathbf{X})$ we will have $P_{\mathrm{I}}(\mathbf{X}_k)$ = const. Then for each one of the zones S_t, the equality in property (4) can be written as:

$$P_{\mathrm{I}}(\mathbf{X}) - m \leq \varepsilon, \tag{6}$$

where $P_{\mathrm{I}}(\mathbf{X})$ is defined by formula (5), with a center of the area $S_t(\mathbf{X})$ at the point $\mathbf{X}_t$; m = const and ε = const.

Obviously, condition (6) is more acceptable for finding out the convex kernel K_{I} in the real problems of the clusterization analysis. In this case using (6) we can obtain an evaluation for the closeness of the separate areas $S_t(\mathbf{X})$:

$$|P_{\mathrm{I}}(\mathbf{X}_a) - P_{\mathrm{I}}(\mathbf{X}_b)| \leq \gamma \tag{7}$$

for $S_a(\mathbf{X})$, $S_b(\mathbf{X}) \subset S_{\mathrm{I}}(\mathbf{X})$ and γ = const – a sufficiently small number.

Since the elementary areas $S_t(\mathbf{X})$ form a cover of the $(n-1)$-dimensional surface $S_{\mathrm{I}}(\mathbf{X})$ in $\boldsymbol{R}^n$, so the inequality (7) is not very convenient for real applications in this form. One more convenient representation of this evaluation can be obtained by means of the theorem of Peano, according to which every limited closed set, defined in the n-dimensional space can be uniquely represented by a intercept of a straight line [9].

In the concrete case we have a real cluster K, which is a limited point set, so its convex envelope $C(K)$ will be a limited set. It is clear, that the covering of the hypersurface $S_{\mathrm{I}}(\mathbf{X})$ from the areas $S_t(\mathbf{X})$, which have a real size $V_t = h^{n-1} > 0$, will consist of a limited number of such areas. If we enumerate these areas in a certain way (but always the same), so, according to the theorem of Peano, the whole covering $\bigcup_t^T S_t(\mathbf{X})$ can be uniquely represented by a linear arrangement (where $T \ll \infty$):

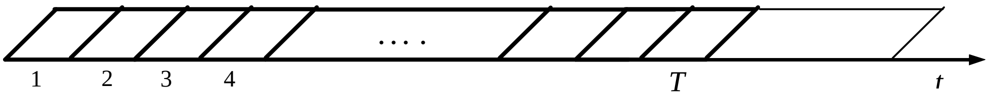

Fig. 3

Then, if we define a 2-dimensional space $[P(t), t]$, where on the ordinate axis we have set the evaluation of the distribution density $P_t(\mathbf{X})$ for each area $S_t(\mathbf{X})$ and t is a discrete value on the abscissa -axis (with a value equal to the number of the area S_t), so the function $P(t)$ will be the histogram representation of the points for the whole (n–1)-dimensional hypersurface $S_{\mathrm{I}}(\mathbf{X})$ in the 2-dimensional space, where $P(t) = P(\mathbf{X}_t)$, $\mathbf{X}_t$ – center of the area $S_t(\mathbf{X})$. In order to be satisfied condition (7), the variation of the function $P(t)$ has to satisfy the inequality: $\mathrm{Var}[P(t)] \leq \gamma \Rightarrow \max P(t) - \min P(t) \leq \gamma$. For this purpose we will view condition (6), which is equivalent to (7) and can be easily realized if we define the constants m and ε in the following way:

$$m = \frac{1}{T}\sum_{t=1}^{T} P(\mathbf{X}_t) \text{ and } \varepsilon = \alpha\left\{\frac{1}{T}\sum_{t=1}^{T}\sqrt{[m - P(\mathbf{X}_t)]^2}\right\}, \text{ where } 0 < \alpha \leq 1.$$

Then if for $\forall t \in [1, T]$ condition (6) is fulfilled, we will consider the current core $K_{\mathrm{I}} \subset K$ to be convex, because this would mean that condition (4) of the definition above is fulfilled. In order to satisfy the second condition of the same definition as well, have to find such a distance D_{I} for which $\mathrm{M}[C(K_{\mathrm{I}})] = \max\limits_{i} M[C(K_i)] = \min\limits_{i} M[K|C(K_i)]$, where $\forall C(K_i)$ satisfies the condition (6). This can be realized through the following algorithm.

Algorithm

A 0. We define a certain initial distance $\mathrm{D}_1 < \mathrm{D}_K$, where D_K is the distance of the whole cluster K. If $C(K_1)$ is a convex set, so go to A1. If $C(K_1)$ is not a convex set, so go to B1.

A1. If $C(K_i)$ is a convex set, so $\mathrm{D}_{i+1} = \mathrm{D}_i + \Delta\mathrm{D}$ go to A1, $i = 1, 2, \ldots$

A2. If $C(K_i)$ is not a convex set, so $C(K_{\mathrm{I}}) = C(K_{i-1})$ go to End.

B1. If $C(K_i)$ is not a convex set, so $\mathrm{D}_{i+1} = \mathrm{D}_i - \Delta\mathrm{D}$ go to B1.; $i = 1, 2, \ldots$

B2. If $C(K_i)$ is a convex set, so $C(K_{\mathrm{I}}) = C(K_{i-1})$ go to End.

End.

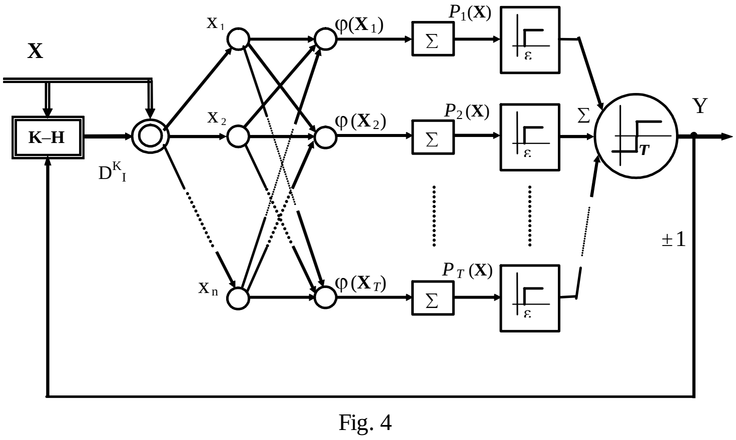

Fig. 4

The second part of the NN, determining the maximum convex core K_{I} of the cluster *K* (for the *K*-level of the NN from Fig. 2) is shown in more details in Fig. 4. For the sake of brevity *K–H* denotes the NN from Fig. 2, which defines the distance D_I^K.

4. Conclusion

The suggested method for finding of the maximum convex inner core of a given cluster *K* allows to approximate its parts by means of using separate convex cores.

This method presents the solution of the first part of the problem for a more compact description of clusters having a random, a priori unknown form. With this method, after the training of the NN, each part of the cluster *K* with a form close to a convex one, is represented by a separate level of the entire NN.

The unification of the separate cores of the given cluster is the second part of the general problem. The finding of a reliable criterion which binds them into a common set of points is a problem of no less complexity in comparison with the one discussed in the present paper. The methods which will solve this problem require additional research and their realization will obviously lead to the addition of new outer layers to the proposed NN.

References

1. Beale, R., T. Jackson. Neural Computing: An Introduction. Bristol, 1990.
2. Kohonen, T. An Introduction to Neural Computing. – Neural Networks, Vol. **1**, 1988, 3-16.
3. Kohonen, T. Self-Organized Formation of Topologically Correct Feature Maps. – Biological Cybernetics, **43**, 1982, 59-69.
4. Haykin, S. Neural Networks. Chapter No 10. Macmillan College Publishing Company, 1994.
5. Choi, D., S. Park. Self-Creating and Organizing Neural Networks. – IEEE Trans. Neural Networks, Vol. **5**, 1994, No 4, 561-575.
6. Mao, J., A. Jain. A Self-Organizing Network for HyperEllipsoidal Clustering (HEC). – IEEE Trans. Neural Networks, Vol. **7**, 1996, No 1, 16-29.
7. Jambu, M. Classifikation Automatique Pour L analyse des Donnees. Paris, Bordas, 1978 (Russian edition – Moscow, Finansi i Statistika, 1988).
8. Duda, R., P. Hart. Pattern Classiffication and Scene Analysis. Wiley-Inter Science Publication, 1973 (Russian edition – Moscow, Mir, 1976) .
9. Kuratowski, K. Introduction to Set Theory and Topology. Sofia, Nauka i Izkustvo, 1979 (Bulgarian edition).