CLUSTER ANALYSIS VIA THE DYNAMIC DATA ASSIGNING ASSESSMENT ALGORITHM

O. Georgieva[1], F. Klawonn[2]

[1] *Institute of Control and System Research - Bulgarian Academy of Sciences, P.O. Box 79, 1113 Sofia, Bulgaria*
phone: +359 2 979 2052, e-mail: ogeorgieva@icsr.bas.bg

[2] *Department of Computer Science, University of Applied Sciences Braunschweig/Wolfenbuettel, Germany*

Abstract. A new clustering algorithm based on the principles of noise clustering that finds good clusters step by step is presented and examined. The algorithm can be applied to finding just a few substructures, but also as an iterative method to data partition including the identification of the number of clusters and noise data. The algorithm is applicable in terms of both hard and fuzzy clustering techniques. Its capabilities are studied over different distance metrics of cluster calculation. The algorithm advantages and drawbacks are discussed with illustrative and real data examples. A certain parallel between the algorithm proposed here and other clustering algorithms that are based on the idea to search one cluster at a time is provided.

Key Words: Clustering, Noise clustering, K-means algorithm, Fuzzy C-means algorithm.

1. Introduction

Cluster analysis is an exploratory data analysis technique that aims at partitioning a given data set into clusters. A cluster is a group of data that can be considered as similar according to a suitable similarity or distance measure. Data objects assigned to different clusters should be

dissimilar.

Most clustering methods have to face the problem of characterizing good clusters among noise data. Rounding and grouping errors result from the inherent inaccuracy stemming from the collection and recording process of the data. The arbitrary noise points that just do not belong to any class being searched for are of a real concern. In fact, a single noise data can completely spoil the separation and thus the results of most least squired based clustering such as the K-means and fuzzy C-means algorithm (FCM) [1,2]. One promising approach to deal with noise data is noise clustering (NC) [7,8]. It maintains the principle of probabilistic clustering, but an additional noise cluster is introduced. NC was developed and investigated in the context of objective function-based clustering as K-means or FCM and it has demonstrated its reliable ability to detect clusters among noise data.

Although the original intention of cluster analysis is to partition a data set into "meaningful" substructures, clustering is often applied for other purposes. For instance, when fuzzy cluster analysis is applied in the context of generating fuzzy rules from data, it is very often used as a segmentation technique that simply partitions the data (in a fuzzy way), without putting a strong emphasis on well distinguished clusters. In other applications, like for example analysing gene expression data or astrophysics data, it is not necessary to partition the data into meaningful clusters, but to identify one or a few interesting clusters that might only cover a small portion of the data.

Following the idea to search for just one cluster at a time a prototype-based clustering algorithm named Dynamic Data Assigning Assessment (DDAA) is proposed. It is based on the NC technique and finds single clusters step by step. The method can be used for two purposes: either in the sense of standard cluster analysis to determine the number of clusters automatically or in order to identify one or a few clusters that might cover only a portion of the data set.

The paper is organised as follows. Second section briefly reviews the necessary background on the objective function-based clustering and the concept of noise clustering that we exploit in our approach. In third section we explain the underlying idea of the method and the algorithm itself. The capability of DDAA algorithm to deal with data spaces with different complexity is investigated by exploring the various distance measures to cluster calculation (Section 4). Fifth section shows how the algorithm is applied to a machine learning data set, whereas sixth section relates it to other clustering methods.

2. Basic concepts

Objective function-based clustering aims minimizing an objective function J that indicates a kind of fitting error of the clusters to the data. In this function, the number of clusters has to be fixed in advance. The underlying objective function for most of the clustering algorithms is [1]:

$$J = \sum_{i=1}^{c} \sum_{k=1}^{N} (u_{ik})^m d_{ik}^2 , \tag{1}$$

where N is the number of data points; c is the number of clusters; u_{ik} and d_{ik} denote correspondingly the membership degree and the distance of the k-th point $x_k = [x_{k1}, x_{k2}, ..., x_{kn}]$, k = 1, ..., N, to the i-th cluster prototype, i = 1, ..., c; $m \in [1,\infty)$ is the weighted exponent coefficient which determines how much clusters may overlap. In order to avoid the trivial solution assigning no data to any cluster, i.e. setting all u_{ik} to zero, and to avoid empty clusters, the following constraints are introduced:

$$u_{ik} \in [0,1], \qquad 1 \le i \le c, \qquad 1 \le k \le N \tag{2.a}$$

$$\sum_{i=1}^{c} u_{ik} = 1, \qquad 1 \le k \le N \tag{2.b}$$

$$0 < \sum_{i=1}^{N} u_{ik} < N, \qquad 1 \le i \le c \tag{2.c}$$

When we choose the fuzzifier m=1 we have $u_{ik} \in \{0,1\}$ at a minimum of the objective function (1) and the resulting partition will be crisp.

The parameters to be optimized are the membership degrees u_{ik} and the cluster parameters which finally determine the distance values d_{ik}. Each cluster is represented by a cluster prototype. In the simplest case, the cluster prototype is a single vector called cluster centre $v_i = [v_{i1}, v_{i2}, ..., v_{in}]$, $i = 1, ..., c$. The distance of a data point k to the i-th cluster is defined by a positive definite symmetric matrix A_i and the cluster centre as follows:

$$d_{ik}^2 = \|x_k - v_i\|_{A_i}^2 = (x_k - v_i) A_i (x_k - v_i)^T . \quad (3)$$

The minimization of the functional (1) represents a nonlinear optimization problem that is usually solved by means of Lagrange multipliers, applying an alternating optimization scheme [2]. This scheme alternatively considers one of the parameter sets, either the membership degrees or the cluster parameters as fixed, while the other parameter set is optimized, until the algorithm finally converges.

The arbitrary noise points that do not belong to any comprehensible class have to be taken into account. The successful solution to deal with the noise in the data set is to collect the noise points in one single cluster [7]. For this purpose a virtual noise prototype with no parameters to be adjusted is introduced that has always the same (large) distance δ to all points in the data set.

Let cluster number c be the noise cluster. Then, by definition we have

$$d_{ck} = \delta, \text{ for } \forall\, k. \quad (4)$$

The remaining c-1 clusters are assumed to be the good clusters in the data set. The objective function J_{noise} that considers the noise cluster is defined in the same manner as in the general scheme for the clustering minimization functional (1) i.e. $J_{noise} \equiv J$, but with some additional

specifications. The distances of every point $x_k,\ k=1,...,N$ are defined by (3) for all clusters $i,\ i=1,...,c-1$ and by

$$d_{ck}^2 = \delta^2 \text{ for } i = c\,. \tag{5}$$

The objective function J_{noise} has the global minimum for a fixed noise distance δ only if:

a) for hard noise clustering (i.e. $m=1$) the membership degrees are:

$$u_{ik} = 0 \text{ for } \forall\, i \neq j \text{ and} \tag{6}$$

$$u_{jk} = 1 \text{ for } j \text{ such that } d_{jk} = \min(d_{ik},\ i = 1,...,c)\,.$$

b) for fuzzy noise clustering ($m>1$) the membership degrees are

$$u_{ik} = \frac{1}{\sum_{j=1}^{c} (d_{ik} / d_{jk})^{2/(m-1)}} \tag{7}$$

and the cluster centres of the good clusters are defined by the weighted mean value:

$$v_i = \frac{\sum_{k=1}^{N} (u_{ik})^m x_k}{\sum_{k=1}^{N} (u_{ik})^m}, \quad \text{for } i = 1,...,c-1. \tag{8}$$

The specification of the noise distance δ is a matter of consideration for the particular data set. If δ is chosen too small, then most of the data points will be classified as noise, while for a large δ value even outliers will be assigned to good clusters.

3. Dynamic data assigning assessment method

Let us assume that the data set consists of only one good cluster among a certain number of noise data considered as a noise cluster. Thus, the two clusters could be separated in minimizing the objective function J_{noise}, which is simplified for the case c=2 to the following form:

$$J_{noise} = \sum_{k=1}^{N} (u_{1k})^m d_k^2 + \delta^2 \sum_{k=1}^{N} (u_{2k})^m\ . \tag{9}$$

The distance d_k denotes the distance between every point x_k and the centre v_1 of the single good cluster. The membership degrees are calculated for a fixed δ :

a) for hard noise clustering as

$u_{1k} = 1$ and $u_{2k} = 0$ if the k-th point belongs to the good cluster, i.e. $d_k \le \delta$ and (10)

$u_{1k} = 0$ and $u_{2k} = 1$ if the k-th point belongs to the noise cluster, i.e. $d_k > \delta$. (11)

b) for fuzzy noise clustering the membership degree to the good cluster is defined as

$$u_{1k} = \frac{1}{1 + \left(\frac{d_k}{\delta} \right)^{2/(m-1)}} \tag{12}$$

and the membership degree to the noise cluster is correspondingly defined as:

$$u_{2k} = \frac{1}{1 + \left(\frac{\delta}{d_k} \right)^{2/(m-1)}} . \tag{13}$$

Since we are still in the framework of probabilistic clustering the following statement is valid to both clustering variants:

$$u_{1k} = 1 - u_{2k} , \quad k = 1,...,N . \tag{14}$$

Even if the data set contains more than one cluster besides the noise data, we can still exploit the idea of having only a single proper cluster plus noise data. For that, we do not try to identify all proper clusters in parallel, but only one of them in each clustering step. Data belonging to other clusters should be assigned to the noise cluster along with the noise data. The proposed method separates one cluster at a time based on the concept of noise via dynamical decrease of the noise distance. Thus, by this approach, it is not necessary to seek for a proper noise distance.

The procedure starts by choosing a large noise distance, for instance the diameter of the

data set, so that all data points are assigned to the good cluster and no data are considered as noise. Then, decreasing the noise distance stepwise by a prescribed decrement $\Delta\delta$, for each $\delta = \delta_j$ value we can determine the number of data belonging to the good cluster according to the membership degrees (crisp or fuzzy) to the good cluster. The index *j* denotes the current step of the noise distance reduction. At every noise distance δ_j the distance d_k, $k = 1,...,N$ is calculated by (3). If the distance is less or equal to δ_j the current point x_k is assigned to the good cluster, if not – the point is separated to the noise cluster. Afterwards, the number of points belonging to the good cluster $N_{in}(j)$ is calculated. The unique cluster centre that defines the good cluster is calculated according to (8). It is obvious that by decreasing the distance δ a process of 'loosing' data, i.e. assigning them to the noise cluster will begin. Continuing to decrease the noise distance, we will start to separate points from the good cluster and add them to the noise cluster until the good cluster will be entirely empty as all data will be assigned to the noise cluster. By decreasing δ the cluster centre will also be shifted drastically, while data are removed from the good cluster.

The described dynamics of moving data from the good cluster to the noise cluster can be characterized by a curve showing the number of data points assigned to the good cluster over the varying noise distance. The velocity $\Delta N_{in}(j)$ via the noise distance alteration is also evaluated:

$$\Delta N_{in}(j) = N_{in}(j-1) - N_{in}(j). \tag{15}$$

Note that not the time is used for the δ-axis so these curves must be viewed from right to left, when we want to observe the behaviour over time.

It is clear that if we loose actual noise points (i.e. from a region of low data density) the curve will almost remain in a plateau, whereas a strong slope should be observed when data from an actual cluster (with higher density than the noise data) are moved to the noise cluster. Normally, a number of clusters with different shapes and densities are presented in the data set. In

this case, we will have a number of plateaus and a number of strong slopes in the curve. Plateaus indicate phases of loosing noise data, strong slopes characterise the situation when we loose one or more proper clusters to the noise cluster.

The peaks obtained of the curve ΔN_{in} correspond to the slopes in the N_{in} curve. The area of every peak is proportional to the number of points that are separated to the noise cluster within the current slope. If a small amount of data is separated to the noise cluster then the peak is small and vice versa – if the data amount is large, then the peak is large.

For each peak s two important noise distances are determined. The right base of the peak denoted by $\delta_{\max}(s)$ corresponds to the δ_j-value at which the data assignment dynamics (15) is bigger than zero. It remains like this until the left base of the peak marked by the second important noise distance $\delta_{\min}(s)$ occurs. It is located where the steep slope ends and the heavy loss of data to the noise cluster ceases. The area of each peak $S(s)$ is represented by the sum of the $\Delta N_{\text{in}}(j)$ values within the considered peak. Only significant peaks, whose area is larger than a predefined threshold *tol* are of a real concern:

$$S(s) = \sum_{j:\delta_{\min}(s)\le\delta_j\le\delta_{\max}(s)} \Delta N_{\text{in}}(j) > tol\,. \tag{16}$$

The last significant peak (the left-most one in $\Delta N_{\text{in}}(j)$ curve) occurs when the data points of the last data group are moved to the noise cluster. These data define a cluster that we remove from the whole data set. The other significant peaks also correspond to phases where at least one comprehensible cluster is shifted to the noise cluster. The data of the not significant peaks i.e. of the plateau phases should be considered as noise data. The whole procedure is applied again to the reduced data set and repeated until no more significant peaks are identified.

The algorithm can be also applied in the context of fuzzy noise clustering if the membership degree is calculated by (12). The point is assumed to belong to the good cluster if its

membership degree is bigger than a predefined value μ_{tol}. However, being stricter in the identification of proper clusters the prescribed membership value should be increased and for a more tolerant identification, it should be decreased.

The idea presented above was applied to identification of single clusters in large data sets [10] and shortly presented in [11]. Here it is summarized in the following Dynamic Data Assigning Assessment cluster identification algorithm:

Step 1: Specify the decrement step $\Delta\delta$ and threshold *tol*. If fuzzy DDAA clustering is applied specify μ_{tol}.

Step 2. Compute the curves N_{in} and ΔN_{in} decreasing δ by the prescribed decrement $\Delta\delta$.

Step 3: Find all peaks of ΔN_{in} curve and select the significant peaks according to (16).

Step 4: Separate one good cluster determined by the last significant peak.

Step 5: Subtract the separated points from the data set and repeat the procedure from ***Step 2*** for the remaining data points until no significant peaks could be found.

As in most clustering algorithms we normalize the data set in advance in order to let each feature has approximately the same influence on the distance used for clustering. Note that the proposed algorithm automatically determines the number of clusters, whereas in standard objective function-based clustering additional strategies have to be applied in order to define the number of clusters. The next section deals with the algorithm ability to partition data spaces comprising clusters with different density, shape and orientation.

4. Analysis of DDAA algorithm by numerical examples

The shape of the selected good clusters is determined by the choice of the matrix A_i that forms the distance metrics (3). For instance, if the identity matrix is incorporated we obtain the

standard Euclidean distance and thus, spherical clusters are determined. Clustering approaches using more complex cluster prototypes than only the cluster centres, leading to adaptive distance measures, are for instance the Gustafson-Kessel (GK) algorithm, volume adaptation strategy and Gath-Geva (GG) algorithm [3,4,5,6]. The latter one is not a proper objective function algorithm, but corresponds to a fuzzified expectation maximization strategy.

Applying the identification algorithm to a data set where the clusters are not well separated provides the comparative analysis of the hard and fuzzy DDAA clustering. The results of the first algorithm pass are shown in parallel for hard (*figure* 1) and fuzzy (*figure* 2) clustering both in Euclidean sense. In the first algorithm pass the hard clustering separates a cluster with highest density but which is a part of an enlarged cluster (*figure* 1.b). In that way the hard clustering splits the large cluster into two clusters. However, the enlarged cluster is recognised by the fuzzy DDAA algorithm (*figure* 2.b).

figure 1

figure 2

We also examine an extreme data set that consists of clusters with different size, shape and density among many noise data. Fuzzy DDAA clustering (*figure* 3.b) can deal better with the complex problems than hard DDAA (*figure* 3.a) due to the given relative degree of membership of a point to the good cluster. The lower oblong cluster and lower left round cluster have equal density and are equally close to the currently defined centre of the good cluster. The hard clustering is ‘hesitating’ to choose one of the custers zigzagging between the cluster centres. It finds the good cluster as a highly density group within the oblong cluster, whereas the fuzzy clustering separates directly the round cluster.

figure 3

In another extreme data set, in which the oblong clusters have highest density, we explore the fuzzy DDAA clustering over Euclidean and GK distance measure. The main feature of the GK distance is its local adaptation to the shape of the cluster as the matrix A_i of (3) is different for each good cluster [4]. For our particular case of consideration, where only one good cluster is searched for, the equation (3) is still valid however the only one product space matrix A_1 is calculated:

$$A_1 = det(F_1)^{1/n} F_1^{-1}, \tag{17}$$

where the fuzzy covariance matrix F_1 is:

$$F_1 = \frac{\sum_{k=1}^{N} (u_{1k})^m (x_k - v_1)^T (x_k - v_1)}{\sum_{k=1}^{N} (u_{1k})^m}. \tag{18}$$

The good cluster recognised in the first algorithm pass via Euclidean distance measure for $\mu_{tol} = 0.5$ does not fully cover the real data cluster (*figure* 4.a). The GK variant for $\mu_{tol} = 0.5$ fits much better the cluster (*figure* 4.b). Best cluster identification is established by acceptance level $\mu_{tol} = 0.2$ (*figure* 4.c). The decreased level of acceptance admits to select only those points that belong to the oblong cluster and thus, to cover the real cluster shape.

figure 4

5. Clustering of the glass identification data set

In order to demonstrate the capability of our algorithm to deal with complex data sets, we apply it to cluster the Glass Identification Database [13]. The data set contains 214 objects that are examples of the chemical analysis of different types of glass. Each object is characterized by values for nine attributes: Refractive index (RI); Sodium (Na); Magnesium (Mg); Aluminium (Al); Silicon (Si); Potassium (Kalium) (K); Calcium (Ca); Barium (Ba); Iron (Fe). The problem is

to forecast the type of glass on the basis of the chemical analysis. The study of classifying the types of glass was motivated by criminological investigations. At the scene of the crime, the glass left can be used as evidence. Although the glass data represents a supervised classification problem, it also suitable to study it in the context of unsupervised classification like cluster analysis.

The data set, represented by a 214x9 matrix, has a quite complex and irregular structure for clustering purposes. For instance, the third, eighth and ninth attributes have zero values for large amount of data that substantially decreases the clustering abilities.

A hard clustering variant of DDAA algorithm is applied to the normalized data set that forms a unit hyperbox with a diameter $\sqrt{9 \times 1^2} = 3$. However, we choose smaller initial noise distance $\delta = 2$ as there are no data assignment dynamics above this value. A cautious decrement $\Delta\delta = 0.005$ and proper threshold value *tol*=30 are also set in advance. Two strong slopes that correspond to the two significant peaks are obtained in the first algorithm pass (*figure* 5). The single good cluster separated by this pass is defined at $\delta = \delta_{\max}(2) = 0.105$. It consists of 44 data points. The second good cluster of 155 data points is selected in the second algorithm pass (*figure* 6) at $\delta = \delta_{\max}(1) = 0.825$. The left 15 data points are defined as noise data. These points are away from both clusters and thus they are not assigned to any of them. The centres of the two defined good clusters are presented in *table1*. In the same table the coordinates of the cluster centres obtained by subtractive clustering [14] are presented as well. It should be mentioned that the original data set consists of two main types of glasses: 51 non-window glass data and 163 window glass data. The mean values of both groups are also given in the *table* 1.

figure 5

figure 6

In case of reduced decrement value $\Delta\delta$ = 0.002 and threshold *tol*=5 more clusters are identified. They correspond to the sixth different types of glasses presented in the database. The cluster centres of this partition are given in *table* 2.

Table 1

Table 2

6. Comparison analysis

The main difference between our algorithm and other well known objective function clustering methods lies in the strategy of searching for good clusters. Most of the algorithms seek for all clusters at once are based on validity measures to assess the quality of the partition. They depend on the algorithm initialisation. The algorithms will usually converge to a local minimizer, which hopefully corresponds to the good clusters [1,8,12,15,16].

Another clustering approach is based on the evolving, distance-based partitioning method [14,17,18]. One of the quite frequently applied clustering methods in the last decade is the subtractive clustering proposed by Chiu [14].

The subtractive clustering method is an improved form of the mountain clustering method introduced by Yager and Filev [17]. In subtractive clustering each data point is considered as a possible cluster centre according to the estimated potential of that point. A data point with many neighbouring data points will have a high potential value. The data with the highest potential value is chosen as the first cluster centre. The key idea is that once the cluster centre is chosen the potential of each data point is revised according to its distance from the currently selected centre. Four clustering parameters need to be properly adjusted in advance in order to obtain a reliable data partition - cluster radius, squash factor, accepting and rejecting rate. Although the preferable values of the clustering parameters are mentioned in [14], experience shows that they should be

fine tuned according to the particular data set.

In contrast to subtractive clustering the adjusted parameters of our algorithm are only two. The more significant parameter is the decrement value $\Delta\delta$ that specifies the reduction rate of the noise distance. The second parameter is the threshold *tol* that determines the number of points above which a defined peak is selected as a significant one.

An important advantage of DDAA algorithm is the opportunity to visualise the dynamics of the accepted data points to the good cluster. Both curves – the accepted rate and its velocity, could be plotted independently of the data dimensionality. One can easily adjust the proper threshold value *tol* simply by looking at the significant steepness of the slopes. In this sense, we do not claim that DDAA algorithm performs better than others, but its few control parameters can be handled in a very intuitive way.

The key of subtractive and DDAA clustering methods is that they do not involve any iterative optimization and thus, the computation grows only linearly with the dimension of the problem (number of data as well as attributes).

It should be mentioned that DDAA algorithm incorporates relatively simple mathematical formulae. We are also free in the choice of the cluster prototype for the single cluster to be identified. We can also use more complicated prototypes for more flexible cluster shapes as they are very often found in objective function based clustering.

7. Conclusions

The cluster identification method presented here is based on the assessment of the dynamics of the number of points that are assigned to only one cluster through noise clustering of the data set by slightly changing the noise distance from a reasonable large to a sufficiently small

value. Two algorithm variants – hard and fuzzy, are presented. They are studied through the comparative analysis over different feature spaces by exploring different distance measures.

The proposed algorithm enables us to identify one good cluster at a time that has a specific data structure with variable shape in one pass of the algorithm and also to identify noise data. Further on, we can proceed with the same clustering procedure but with a reduced data set as the data belonging to the already identified cluster are subtracted from the entire data set. Thus, we can discover single interesting clusters, even if the majority of the data does not form any kind of cluster structures.

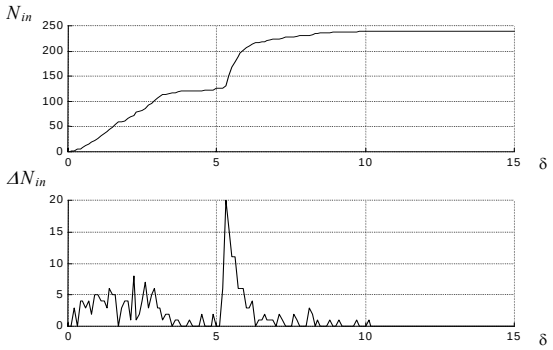
The method can be applied as a stand-alone clustering algorithm as it estimates the number of clusters and the clusters' mean values. However, it is also appropriate for initialising more sophisticated clustering algorithm as K-means or its fuzzy variants as FCM, GK and GG.

Acknowledgement: This work was partly supported by the German Research Society (DFG), under Grant 436 BUL 112/2/04.

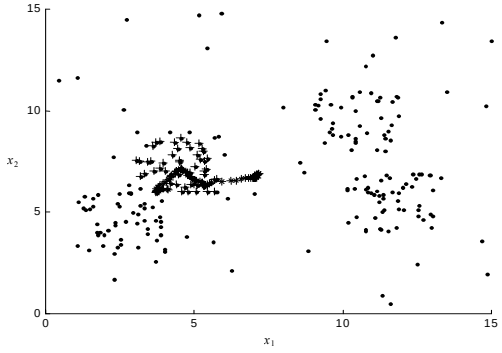
References

1. Bezdek, J.C., Pattern Recognition with Fuzzy Objective Function Algorithms Plenum Press, New York, 1981.
2. Bezdek, J.C., "A Convergence Theorem for the Fuzzy ISODATA Clustering Algorithms", *IEEE Trans. Pattern Analysis and Machine Intelligence*, 2(1), 1-8, 1980.
3. Gath, I., A.B. Geva, "Unsupervised optimal fuzzy clustering", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v.7, pp. 773-781, 1989.
4. Gustafson, D., W. Kessel, Fuzzy clustering with a fuzzy covariance matrix, *Advances in fuzzy set theory and applications*, North-Holland, 1979, 605-620.
5. Hoeppner, F., F. Klawonn, R. Kruse, T. Runkler, Fuzzy Cluster Analysis, John Wiley & Sons, Chichester, 1999.
6. Keller A., F. Klawonn, Adaptation of cluster sizes in objective function based fuzzy clustering, in: C.T. Leondes (ed.): *Intelligent Systems: Technology and Applications* vol. IV: Database and Learning Systems. CRC Press, Boca Raton, 2003, 181-199.
7. Dave, R. H., "Characterization and detection of noise in clustering", *Pattern Recognition Letters* 12, 657-664, 1991.

8. Dave, R. H., R. Krishnapuram, "Robust clustering methods: A unified view", *IEEE Trans. Fuzzy Systems*, vol. 5, 1997, 270-293.
10. Georgieva O., F. Klawonn, A clustering algorithm for identification of single clusters in large data sets, in Proc. *East West Fuzzy Colloquium*, Sept. 8-10, Zittau, Germany, Heft 81, 2004, 118-12.
11. Georgieva O., F. Klawonn, Dynamic data assigning assessment clustering via different distance metrics, Automatics and Informatics'05, Oct.3-5, Sofia, Bulgaria, 2005, 175-178.
12. Babuska, Fuzzy Modeling for Control, Kluwer Academic Publishers, Boston, USA, 1998.
13. Glass database, ftp://ftp.ics.uci.edu/pub/machine-learning-databases/glass/.
14. Chiu S.L., "Fuzzy model identification based on cluster estimation," *Journal of Intelligent and Fuzzy Systems*, vol. 2, 1994, 267-278.
15. Jolion J.-M., P. Meer, and S. Bataouche, "Robust clustering with applications in computer vision," *IEEE Trans. Pattern Anal. Machine Intell.,* vol. 13, Aug. 1991, 791–802.
16. Stewart C. V. , "MINPRAN: A new robust estimator for computer vision," *IEEE Trans. Pattern Anal. Machine Intell.,* vol. 17, no. 10, Oct. 1995, 925–938.
17. Yager R., D. Filev, *Essentials of Fuzzy Modeling and Control*, John Wiley & Sons, 1994.
18. Song Q., N. Kasabov, "Dynamic Evolving NeuroFuzzy Inference System (DENFIS): On-line Learning and Application for Time-Series Prediction," in *Proceedings of the 6 th International Conference on Soft Computing* , 2000, Iizuka, Japan.

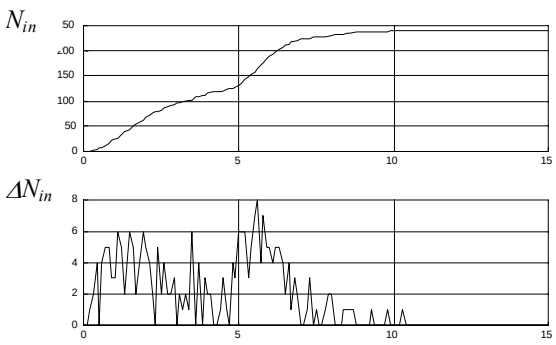


a) Data assignment dynamics of of the first algorithm pass.

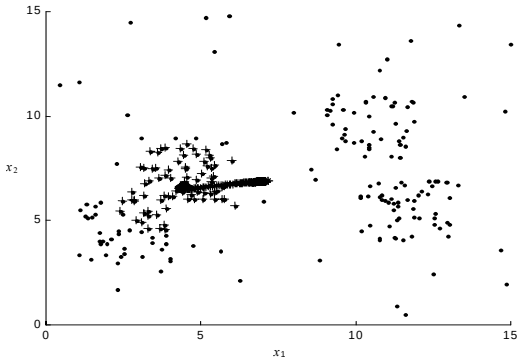


b) The last defined cluster (+) and the trace the cluster centre (∗).

Figure 1. Hard DDAA clustering, $tol = 30$.

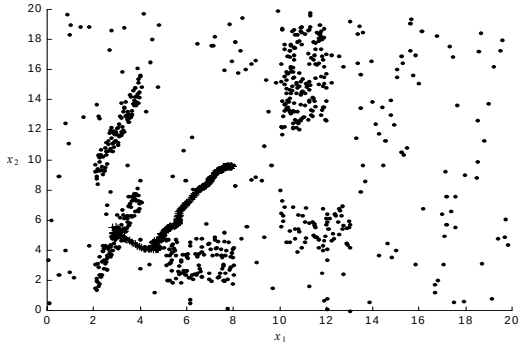


a) Data assignment dynamics of the first algorithm pass.

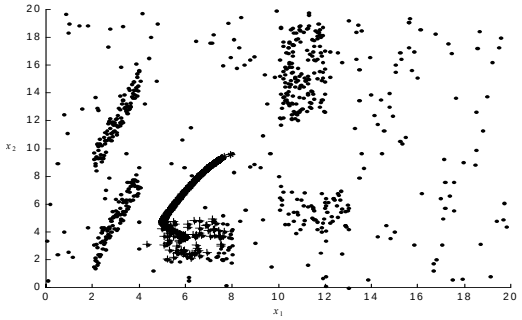


b) Defined cluster (+) in the first algorithm pass and the trace of the cluster centre (∗).

Figure 2. Fuzzy DDAA clustering, *tol*=30, μ_{tol} = 0.5.

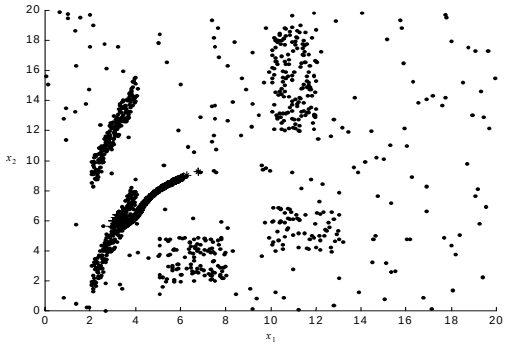


a) hard DDAA clustering

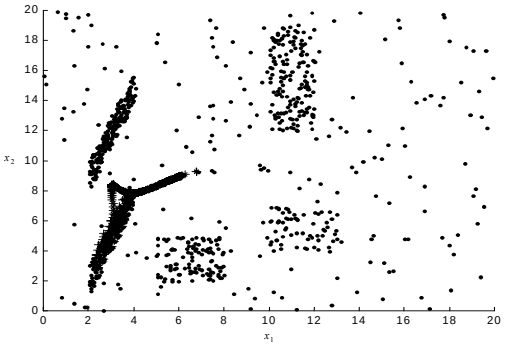


b) fuzzy DDAA clustering, μ_{tol} = 0.5

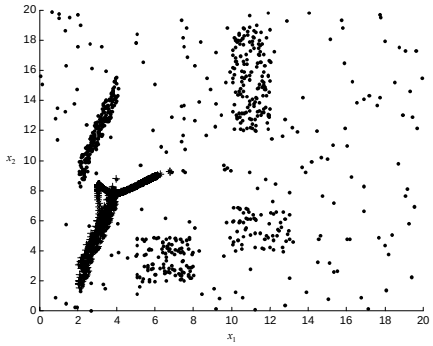
Figure 3. Selected cluster (+) and the trace (∗) of the cluster centre, *tol*=10 of the first algorithm pass.



a) via Euclidean distance measure, $\mu_{tol} = 0.5$.



b) via GK distance measure, μ_{tol}=0.5.



b) via GK distance measure, $\mu_{tol} = 0.2$

Figure 4. Selected cluster (+) and the trace (∗) of the cluster centre, $tol = 30$ of the first algorithm pass.

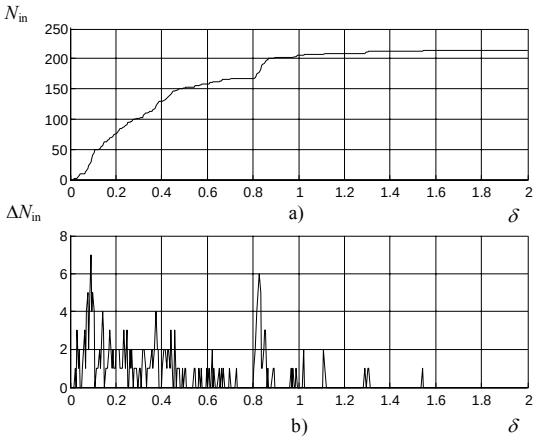


Figure 5. Data dynamics of the first DDAA algorithm pass applied to Glass database

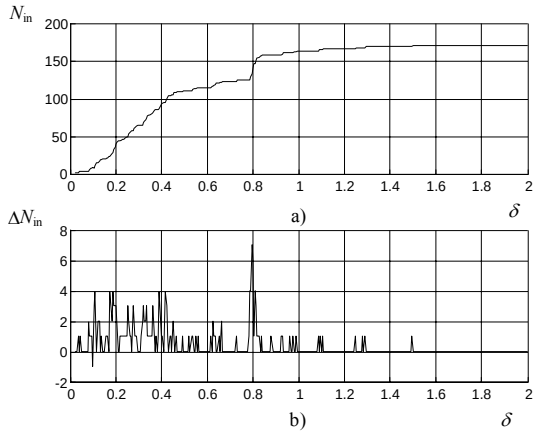


Figure 6. Data dynamics of the second DDAA algorithm pass applied to Glass database

Table 1. Cluster centres calculated by different partition techniques

	c	RI	Na	Mg	Al	Si	K	Ca	Ba	Fe
Our algorithm	1	1.5170	13.0448	3.5311	1.3336	72.9507	0.5789	8.3561	0	0
	2	1.5185	13.5538	2.6763	1.4512	72.6190	0.3882	8.9621	0.1760	0.0692
Subtractive clustering	1	1.5178	13.0800	3.4900	1.2800	72.8600	0.6000	8.4900	0	0
	2	1.5166	12.8500	3.5100	1.4400	73.0100	0.6800	8.2300	0.0600	0.2500
Mean of the two data types	1	1.5186	13.2017	3.2949	1.2817	72.5869	0.4775	8.9247	0.0298	0.0676
	2	1.5176	14.0667	0.7337	1.9667	72.8555	0.5596	9.0602	0.6392	0.0231

Table 2. Cluster centres of selected clusters by hard DDAA algorithm

RI	Na	Mg	Al	Si	K	Ca	Ba	Fe
1.5171	13.0147	3.5284	1.3142	72.9318	0.5818	8.4150	0	0
1.5174	13.2789	3.5237	1.4371	72.6908	0.5592	8.2963	0.0037	0.0042
1.5183	13.1161	3.4725	1.2935	72.7067	0.5071	8.6998	0.0120	0.1512
1.5166	14.6833	0	2.0875	73.2383	0.0117	8.6267	1.2833	0.0133
1.5204	13.7824	2.7489	1.2305	72.1392	0.2211	9.6932	0.0465	0.0227
1.5195	13.6330	0.3193	1.7926	72.9663	0.4170	10.2152	0.5185	0.0352