A Modification to Improve Possibilistic Fuzzy Cluster Analysis

Heiko Timm and Rudolf Kruse

Dept. of Knowledge Processing and Language Engineering
Otto-von-Guericke-University of Magdeburg
Universitätsplatz 2, D-39106 Magdeburg, Germany

e-mail: {timm,kruse}@iws.cs.uni-magdeburg.de

Abstract - We explore an approach to possibilistic fuzzy clustering that avoids a severe drawback of the conventional approach, namely that the objective function is truly minimized only if all cluster centers are identical. Our approach is based on the idea that this undesired property can be avoided if we introduce a mutual repulsion of the clusters, so that they are forced away from each other. We develop this approach for the possibilistic fuzzy c-means algorithm and the Gustafson–Kessel algorithm.

Keywords: **fuzzy clustering, possibilistic membership degrees**

I. Introduction

Cluster analysis is a technique for classifying data, i.e., to divide a given dataset into a set of classes or *clusters*. The goal is to divide the dataset in such a way that two cases from the same cluster are as similar as possible and two cases from different clusters are as dissimilar as possible. Thus one tries to model the human ability to group similar objects or cases into classes and categories. In classical cluster analysis each datum must be assigned to exactly one cluster. Fuzzy cluster analysis relaxes this requirement by allowing gradual memberships, thus offering the opportunity to deal with data that belong to more than one cluster at the same time.

Most fuzzy clustering algorithms are objective function based: They determine an optimal classification by minimizing an objective function. In objective function based clustering usually each cluster is represented by a *cluster prototype*. This prototype consists of a *cluster center* (whose name already indicates its meaning) and maybe some additional information about the size and the shape of the cluster. The cluster center is an instantiation of the attributes used to describe the domain, just as the data points in the dataset to divide. However, the cluster center is computed by the clustering algorithm and may or may not appear in the dataset. The size and shape parameters determine the extension of the cluster in different directions of the underlying domain.

The degrees of membership to which a given data point belongs to the different clusters are computed from the distances of the data point to the cluster centers. These distances depend on the size and the shape of the cluster as stated by the additional prototype information. The closer a data point lies to the center of a cluster (w.r.t. size and shape), the higher is its degree of membership to this cluster. Therefore the problem to divide a dataset $X = \{\vec{x}_1, \ldots, \vec{x}_n\} \subseteq \mathbb{R}^p$ into c clusters can be stated as the task to minimize the distances of the data points to the cluster centers, since, of course, we want to maximize the degrees of membership.

Several fuzzy clustering algorithms can be distinguished depending on the additional size and shape information contained in the cluster prototypes, the way in which the distances are determined, and the restrictions that are placed on the membership degrees [4], [3], [8]. Here we focus on the fuzzy c-means algorithm [2], which uses only cluster centers and a Euclidean distance function, and the Gustafson–Kessel algorithm [7], which uses cluster centers, covariance matrices and a Mahalanobis distance function.

Probabilistic Fuzzy Clustering

In probabilistic fuzzy clustering the task is to minimize the objective function

$$J(\mathbf{X}, \mathbf{U}, \mathbf{B}) = \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^m d^2(\vec{\beta}_i, \vec{x}_j) \qquad (1)$$

subject to

$$\sum_{j=1}^{n} u_{ij} > 0, \qquad \text{for all } i \in \{1, \ldots, c\}, \qquad \text{and} \qquad (2)$$

$$\sum_{i=1}^{c} u_{ij} = 1, \qquad \text{for all } j \in \{1, \ldots, n\}, \tag{3}$$

where $u_{ij} \in [0, 1]$ is the membership degree of datum $\vec{x}_j$ to cluster c_i, $\vec{\beta}_i$ is the prototype of cluster c_i, and $d(\vec{\beta}_i, \vec{x}_j)$ is the distance between datum $\vec{x}_j$ and prototype $\vec{\beta}_i$. $\mathbf{B}$ is the set of all c cluster prototypes $\vec{\beta}_1, \ldots, \vec{\beta}_c$. The $c \times n$ matrix $\mathbf{U} = [u_{ij}]$ is called the fuzzy partition matrix and the parameter m is called the fuzzifier. This parameter determines the "fuzziness" of the classification. With higher values for m the boundaries between the clusters become softer, with lower values they get harder. Usually $m = 2$ is chosen.

Constraint (2) guarantees that no cluster is empty and constraint (3) ensures that the sum of the membership degrees for each datum equals 1. Because of the second constraint, this approach is called *probabilistic clustering*, since with it the membership degrees for a given datum formally resemble the probabilities of its being a member of the corresponding cluster.

Unfortunately, the objective function J cannot be minimized directly. Therefore an iterative algorithm is used, which alternately optimizes the cluster prototypes and the membership degrees. That is, first the cluster prototypes are optimized for fixed membership degrees, then the membership degrees are optimized for fixed prototypes. The main advantage of this scheme is that in each of the two steps the optimum can be computed directly. By iterating the two steps the joint optimum is approached. The update formulae are derived by simply setting the derivative of the objective function (extended by Lagrange multipliers to incorporate the constraints) w.r.t. the parameter to optimize equal to zero. For the membership degrees we thus obtain the following formula if $d^2(x_j, \beta_k) > 0$ holds with $k \in 1, \ldots, c$

$$u_{ij} = \frac{1}{\sum_{k=1}^{c} \left(\frac{d^2(x_j, \beta_i)}{d^2(x_j, \beta_k)} \right)^{\frac{1}{m-1}}}. \tag{4}$$

Equation (4) shows that the membership degree of a datum to a cluster depends not only on the distance between the datum and that cluster, but also on the distances between the datum and other clusters. The partitioning property of a probabilistic clustering algorithm, which "distributes" the weight of a datum on the different clusters, is due to this equation.

Although often desirable, the "relative" character of the membership degrees in a probabilistic clustering approach can lead to counterintuitive results. Consider, for example, the simple case of two clusters shown in figure 1. Datum $\vec{x}_1$ has the same distance to both clusters

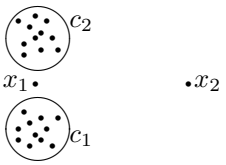

Fig. 1. A situation in which the probabilistic assignment of membership degrees is counterintuitive for datum x_2.

and thus it is assigned a degree of membership of about 0.5. This is plausible. However, the same degrees of membership are assigned to datum $\vec{x}_2$. Since this datum is far away from both clusters, it would be more intuitive if it had a low degree of membership to both of them.

Possibilistic Fuzzy Clustering

In possibilistic fuzzy clustering one tries to achieve a more intuitive assignment of degrees of membership by dropping constraint (3), which is responsible for the undesirable effect discussed above. However, this leads to the mathematical problem that the objective function is now minimized by assigning $u_{ij} = 0$ for all $i \in \{1, \ldots, c\}$ and $j \in \{1, \ldots, n\}$. In order to avoid this trivial solution, a penalty term is introduced, which forces the membership degrees away from zero. That is, the objective function J is modified to

$$\begin{aligned} J(\mathbf{X}, \mathbf{U}, \mathbf{B}) &= \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^m d^2(\vec{\beta}_i, \vec{x}_j) \\ &\quad + \sum_{i=1}^{c} \eta_i \sum_{j=1}^{n} (1 - u_{ij})^m, \end{aligned} \tag{5}$$

where $\eta_i > 0$. The first term leads to a minimization of the weighted distances while the second term suppresses the trivial solution. This approach is called *possibilistic clustering*, because the membership degrees for one datum resemble the possibility (in the sense of possibility theory [6]) of its being a member of the corresponding cluster [9], [5]. The formula for updating the membership degrees that is derived from this objective function is [9]

$$u_{ij} = \frac{1}{1 + \left(\frac{d^2(\vec{x}_j, \vec{\beta}_i)}{\eta_i} \right)^{\frac{1}{m-1}}}. \tag{6}$$

From this equation it becomes obvious that η_i is a parameter that determines the distance at which the membership degree equals 0.5. η_i is chosen for each cluster separately and can be determined, for example, by computing the fuzzy intra cluster distance [9]

$$\eta_i = \frac{K}{N_i} \sum_{j=1}^{n} u_{ij}^m d^2(\vec{x}_j, \vec{\beta}_i), \tag{7}$$

where $N_i = \sum_{j=1}^{n} u_{ij}^m$. Usually $K = 1$ is chosen.

At first sight this approach looks very promising. However, if we take a closer look, we discover that the objective function J defined above is, in general, truly minimized only if all cluster centers are identical. The reason is that formula (6) for the membership degree of a datum to a cluster depends only on the distance of the datum to that cluster, but not on its distance to other clusters. Hence, if there is a single optimal point for a cluster center (as it will usually be the case, since multiple optimal points would require a high symmetry in the data), all cluster centers will converge to this point. More formally, consider two cluster centers $\vec{\beta}_1$ and $\vec{\beta}_2$, which are not identical, and let

$$z_i = \sum_{j=1}^{n} u_{ij}^m d^2(\vec{\beta}_i, \vec{x}_j) + \eta_i \sum_{j=1}^{n} (1 - u_{ij})^m, \quad i = 1, 2,$$

i.e., let z_i be the amount that cluster β_i contributes to the value of the objective function. Except in very rare cases of high data symmetry, it will then either be $z_1 > z_2$ or $z_2 > z_1$. That is, we can improve the value of the objective function by setting both cluster centers to the same value, namely the one which yields the smaller z-value, because the two z-values do not interact.

Note that this behavior is specific to the possibilistic approach. In the probabilistic approach the cluster centers are driven apart, because a cluster, in a way, "consumes" part of the weight of a datum and thus leaves less that may attract other cluster centers. Hence sharing a datum between clusters is disadvantageous. In the possibilistic approach there is nothing equivalent to this effect.

Nevertheless, possibilistic fuzzy clustering usually leads to acceptable results, although it suffers from stability problems if it is not initialized with the corresponding probabilistic algorithm. We assume that other results than all cluster centers being identical are achieved only, because the algorithm gets stuck in a local minimum of the objective function. This, of course, is not a desirable situation. Hence we tried to improve the algorithm by modifying the objective function in such a way that the problematic property examined above is removed.

II. Cluster Repulsion

The idea of our approach is to combine an attraction of data to clusters with a repulsion between different clusters. In contrast to a probabilistic clustering algorithm this is not done implicitly using restriction (3), but explicitly by adding a cluster repulsion term to the objective function [10].

To arrive at a suitable objective function, we started from the following set of requirements:

- The distance between clusters and the data points assigned to them should be minimized.
- The distance between clusters should be maximized.
- There should be no empty clusters, i.e., for each cluster there must be datum with non-vanishing membership degree.
- Membership degrees should be close to one or close to zero and, of course, the trivial solution of all membership degrees being zero should be suppressed.

These requirements are very close to standard possibilistic cluster analysis. The attraction between data and clusters is modeled (as described above) by a term $\sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^m d^2(\vec{\beta}_i, \vec{x}_j)$. A term $\sum_{i=1}^{c} \eta_i \sum_{j=1}^{n} (1 - u_{ij})^m$ is used to suppress the trivial solution. The objective that no cluster should be empty leads to constraint (2). The repulsion between clusters can be described in analogy to the attraction between data and clusters. That is, we are using a term that is minimized if the sum of the distances between clusters are maximized.

This could be achieved by simply subtracting the sum of squared distances between clusters from the objective function. However, this straightforward approach does not work. The problem is that this kind of repulsion increases with the distance of the clusters and thus drives them ever farer apart. In the end, all data points would be assigned to one cluster and all other clusters would have been moved to infinity.

To avoid this undesired "explosion" of the cluster set, a repulsion term must be used that becomes smaller the farer the clusters are apart. Then the attraction of the data points can compensate the repulsion only if the clusters are sufficiently dispersed. This consideration lead us to the term $\sum_{i=1}^{c} \gamma_i \sum_{k=1, k \neq i}^{c} \frac{1}{d^2(\vec{\beta}_i, \vec{\beta}_k)}$ where γ_i is a weighting factor. This term is only relevant if the clusters are close together. With growing distance it becomes smaller, i.e., the repulsion is gradually decreased until it is compensated by the attraction of the data. The weighting factor γ_i should be cluster-specific to deal with the case that clusters have a highly varying number of data points assigned to them. γ_i can be defined as $\gamma_i = \gamma \sum_{j=1}^{n} u_{i,j}^m$. The repulsion is increased if cluster $\vec{\beta}_i$ is attracted by many data. An alternative approach to model the repulsion between clusters is to use the term $\sum_{i=1}^{c} \gamma_i \sum_{k=1, k \neq i}^{c} e^{-d^2(\vec{\beta}_i, \vec{\beta}_k)}$ instead of the quotient used above. The difference between both terms is how the repulsion between clusters decreases with growing distance. However, in this paper we only discuss the first one.

The classification problem is then described as the task to minimize

$$J(\mathbf{X},\mathbf{U},\mathbf{B}) = \sum_{i=1}^{c}\sum_{j=1}^{n} u_{ij}^m d^2(\vec{\beta}_i,\vec{x}_j) + \sum_{i=1}^{c}\eta_i \sum_{j=1}^{n}(1-u_{ij})^m + \sum_{i=1}^{c}\gamma_i \sum_{k=1,k\neq i}^{c} \frac{1}{d^2(\vec{\beta}_i,\vec{\beta}_k)} \quad (8)$$

w.r.t. the constraint $\sum_{j=1}^{n} u_{ij} > 0$ for all $i \in \{1,\ldots,c\}$. γ_i is used to weight two objectives against each other: the objective that the distance to the clusters should be minimized and the objective that the distance between clusters should be maximized. Using $\frac{1}{d^2(\vec{\beta}_i,\vec{\beta}_k)}$ means that only clusters with a small distance are relevant for minimizing the objective function, while clusters with a large distance are only slightly repelling each other.

Minimization of (8) w.r.t. the membership degrees leads to (6). That is, the membership degrees have the same meaning as in possibilistic cluster analysis.

For the variant of the fuzzy c-means algorithm (only cluster centers $\vec{c}_i$, Euclidean distance, and therefore spherical clusters) a minimization of (8) with respect to the cluster prototypes leads to

$$\sum_{j=1}^{n} u_{ij}^m(\vec{x}_j - \vec{c}_i) - \gamma_i \sum_{k=1,k\neq i}^{c} \frac{1}{d^4(\vec{c}_k,\vec{c}_i)}(\vec{c}_k - \vec{c}_i) = 0.$$

For reasons of simplicity, we interpret $\frac{1}{d^4(\vec{c}_k,\vec{c}_i)}$ as a repulsion degree between cluster $\vec{c}_k$ and cluster $\vec{c}_i$. With this interpretation we can compute the cluster centers by

$$\vec{c}_i = \frac{\sum_{j=1}^{n} u_{ij}^m \vec{x}_j - \gamma_i \sum_{k=1,k\neq i}^{c} \frac{1}{d^4(\vec{c}_k,\vec{c}_i)}\vec{c}_k}{\sum_{j=1}^{n} u_{i,j}^m - \gamma_i \sum_{k=1,k\neq i}^{c} \frac{1}{d^4(\vec{c}_k,\vec{c}_i)}} \quad (9)$$

Alternatively we can also solve this equation iteratively.

Next we turn to the Gustafson–Kessel algorithm [7]. Here we face the problem that we cannot transfer the computation of distances and data points and cluster centers to the computation of distances between cluster centers. The reason is that the distance between a data point and a cluster depends on the covariance matrix associated with the cluster. But for distances between clusters two covariance matrices have to be taken into account. A simple way to cope with this problem is to average the distances that result if the other cluster center is treated as a data point, i.e.

$$d^2(\vec{\beta}_i,\vec{\beta}_k) = \frac{1}{2}\left((\vec{c}_i - \vec{c}_k)^\top \mathbf{A}_i(\vec{c}_i - \vec{c}_k) + (\vec{c}_i - \vec{c}_k)^\top \mathbf{A}_k(\vec{c}_i - \vec{c}_k)\right). \quad (10)$$

$\mathbf{A}_i$ is the norm matrix that describes the shape of cluster $\vec{\beta}_i$.

A minimization of (8) with respect to the cluster prototypes leads to

$$\sum_{j=1}^{n} u_{i,j}^m(\vec{x}_j - \vec{c}_i) - \gamma_i \sum_{k=1,k\neq i}^{c} \frac{1}{d^4(\vec{c}_k,\vec{c}_i)}\frac{1}{2}\left((\vec{c}_k - \vec{c}_i)^\top \mathbf{A}_i + (\vec{c}_k - \vec{c}_i)^\top \mathbf{A}_k\right) = 0.$$

For reasons of simplicity we compute the cluster centers by

$$\vec{c}_i = \left(\sum_{j=1}^{n} u_{i,j}^m \mathbb{1} - \gamma_i \sum_{k=1,k\neq i}^{c} \frac{\left(\mathbf{A}_i^\top + \mathbf{A}_k^\top\right)}{2d^4(\vec{c}_k,\vec{c}_i)}\right)^{-1} \cdot \left(\sum_{j=1}^{n} u_{i,j}^m \vec{x}_j - \gamma_i \sum_{k=1,k\neq i}^{c} \frac{\left(\vec{c}_k^\top \mathbf{A}_i + \vec{c}_k^\top \mathbf{A}_k\right)}{2d^4(\vec{c}_k,\vec{c}_i)}\right)^\top. \quad (11)$$

The term $\frac{1}{d^4(\vec{c}_k,\vec{c}_i)}$ can be interpreted as the repulsion degree between cluster $\vec{\beta}_i$ and cluster $\vec{\beta}_k$.

A minimization of (8) with respect to $\mathbf{A}_i$ leads to

$$\mathbf{A}_i = \sqrt[p]{\det(\mathbf{S}_i)}\mathbf{S}_i^{-1}. \quad (12)$$

where $\mathbf{S}_i$ is defined as

$$\mathbf{S}_i = \sum_{j=1}^{n} u_{i,j}^m(\vec{x}_j - \vec{c}_i)(\vec{x}_j - \vec{c}_i)^\top - \gamma_i \sum_{k=1}^{c}(\vec{c}_k - \vec{c}_i)(\vec{c}_k - \vec{c}_i)^\top \frac{1}{2d^4(\vec{\beta}_i,\vec{\beta}_k)}. \quad (13)$$

The expressions to compute the cluster centers $\vec{c}_i$ and the norm matrices $\mathbf{A}_i$ demonstrate the effect of cluster repulsion. Clusters are attracted by data assigned to them and repelled by other clusters. The effect of the repulsion is roughly the same as if data points close to the repelling cluster center were neglected. If the repelling part of the expressions extends the attracting part, that is for the fuzzy c-means $\sum_{j=1}^{n} u_{i,j}^m < \gamma_i \sum_{k=1,k\neq i}^{c} \frac{1}{d^4(\vec{c}_k,\vec{c}_i)}$, one cluster should be removed and initialized at a different position.

III. Experimental Results

We used the wine dataset [1] to test our approach. The wine dataset has three clusters with 59, 71, and 48 data points, respectively. For cluster analysis we used the attributes 7, 10, 13. We scaled the dataset to $[0, 10]$ in each dimension.

Fig. 5 shows the classification obtained with the probabilistic Gustafson–Kessel algorithm. The grey scale indicates the membership degree to clusters. Attribute 7 and 10 are shown. This result clearly demonstrates the

partitioning property of the probabilistic algorithm. The dataset is divided into three clusters. Fig. 6 shows the classification obtained with the possibilistic Gustafson–Kessel algorithm. All clusters are identical. The global optimum of the possibilistic objective function is found. Fig. 7 shows the result of our approach with $\gamma = 1$. To show the classification borders we used a light grey. The classification of the data space is similar to the probabilistic approach. However the clusters differ. Because the data is not well separated all three clusters are close together, see fig. 2, 3, and 4. The data points in the middle attract all clusters. If we increase the weight of the repulsion to $\gamma = 6$ the cluster on the left is based mainly on the data on the left. The attraction of the data points in the middle is canceled by the repulsion of other clusters. Compared with the classification result with $\gamma = 1$ the data points on the left are better represented by the cluster on the left. The cluster on the right does not change its position or its shape because it fits the data points.

IV. Conclusion

In this paper we presented an approach for possibilistic fuzzy cluster analysis that is based on data attracting cluster centers as well as cluster centers repelling each other. This approach combines the more intuitive membership degrees of possibilistic fuzzy cluster analysis (since they can be interpreted as similarities) with the property of probabilistic fuzzy cluster analysis to detect distinct clusters. The attraction between clusters and data points assigned to them and the repulsion between clusters is modeled separately. In contrast to a probabilistic clustering algorithm the membership degree can be interpreted as a measure of similarity to a cluster. The repulsion between clusters avoids the problems of possibilistic cluster analysis as described above. γ_i is used to weight the two opposite objectives, i.e., that the distance between clusters and data assigned to them should be minimized and that the distance between clusters should be maximized. The modeling of the repulsion avoids the problem of the "'explosion"' of clusters.

However, if we compare possibilistic fuzzy cluster analysis with probabilistic fuzzy cluster analysis we always have to keep in mind that in possibilistic fuzzy cluster analysis data can have a high membership degree to several clusters while a probabilistic fuzzy clustering algorithm partitions the data.

References

[1] Aeberhard, S., Coomans, D. and de Vel, O.: Comparison of Classifiers in High Dimensional Settings. Tech Rep. 92-02, Dept. of Computer Science and Dept. of Mathematics and Statistics, James Cook University of North Queensland, 1992.

[2] Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press, New York, NY, USA 1981.

[3] Bezdek, J.C., Keller, J., Krishnapuram R., and Pal, N.R.: Fuzzy Models and Algorithms for Pattern Recognition and Image Processing. Kluwer, Boston, London, 1999.

[4] Bezdek, J.C. and Pal S.K.: Fuzzy Models for Pattern Recognition — Methods that Search for Structures in Data. IEEE Press, Piscataway, NJ, USA 1992.

[5] Davé, R.N. and Krishnapuram, R.: Robust Clustering Methods: A Unified View, IEEE Transactions on Fuzzy Systems, pp. 270-293, (5) 1997.

[6] D. Dubois and H. Prade. *Possibility Theory.* Plenum Press, New York, NY, USA 1988

[7] Gustafson, E.E. and Kessel, W.C. Fuzzy Clustering with a Fuzzy Covariance Matrix. IEEE CDC, San Diego, California, pp. 761-766, 1979.

[8] Höppner, F., Klawonn, F., Kruse, R., and Runkler, T.: Fuzzy Cluster Analysis. J. Wiley & Sons, Chichester, England 1999.

[9] Krishnapuram, R. and Keller, J.: A Possibilistic Approach to Clustering, IEEE Transactions on Fuzzy Systems, pp. 98-110, (1) 1993.

[10] Timm, H., Borgelt, C., Döring, C., and Kruse, R.: Fuzzy Cluster Analysis with Cluster repulsion. Proc. European Symposium on Intelligent Technologies (EUNITE), Tenerife, Spain, 2001.

Fig. 2. Wine dataset with attribute 7 and 10.

Fig. 3. Wine dataset with attribute 10 and 13.

Fig. 4. Wine dataset with attribute 7 and 13.

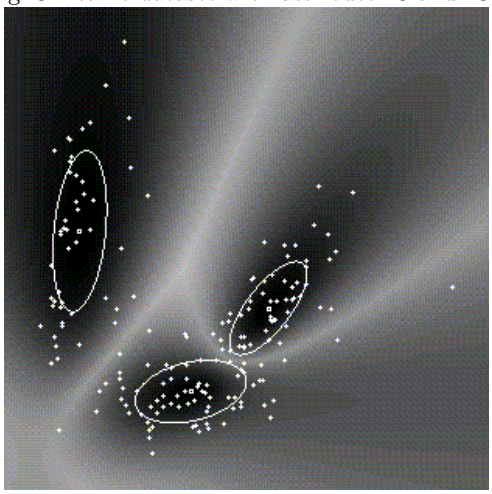

Fig. 5. Wine dataset classified with probabilistic Gustafson–Kessel algorithm. Attributes 7, 10.

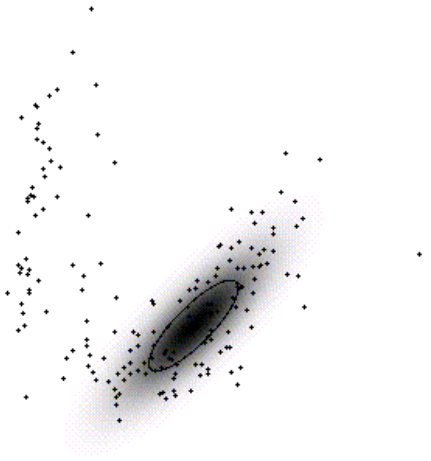

Fig. 6. Wine dataset classified with possibilistic Gustafson–Kessel algorithm. Attributes 7, 10.

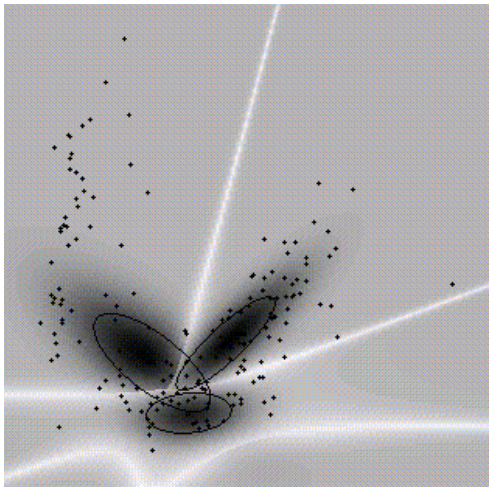

Fig. 7. Wine dataset classified with possibilistic Gustafson–Kessel algorithm. Attributes 7, 10, $\gamma = 1$.

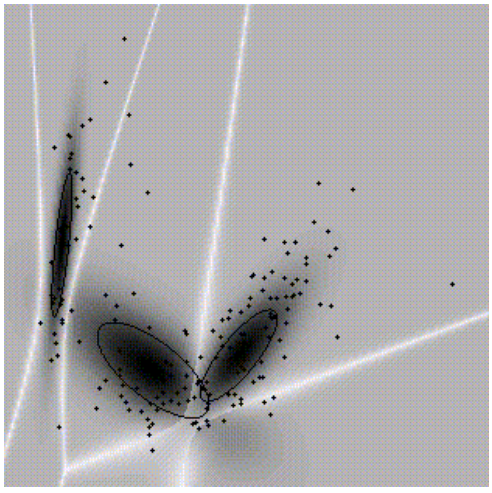

Fig. 8. Wine dataset classified with possibilistic Gustafson–Kessel algorithm. Attributes 7, 10, $\gamma = 6$.

0-7803-7280-8/02/$10.00 ©2002 IEEE