Gustafson-Kessel-like clustering algorithm based on typicality degrees

Marie-Jeanne Lesot
FIN, Otto-von-Guericke Universität,
Magdeburg, Germany
lesot@iws.cs.uni-magdeburg.de

Rudolf Kruse
FIN, Otto-von-Guericke Universität,
Magdeburg, Germany
kruse@iws.cs.uni-magdeburg.de

Abstract

Typicality degrees were defined in supervised learning as a tool to build characteristic representatives for data categories. In this paper, an extension of these typicality degrees to unsupervised learning is proposed to perform clustering. The proposed algorithm constitutes a Gustafson-Kessel variant and makes it possible to identify ellipsoidal clusters with robustness as regards outliers.

Keywords: clustering, typicality degrees, Gustafson-Kessel, outliers

1 Introduction

Typicality degrees [10, 6] were defined in a prototype building procedure, as a means to construct characteristic representatives of data categories: according to this approach, a point is typical of a category if it both resembles the other members of the category and differs from members of other categories. A prototype based on such typicality degrees then highlights the common features of the group members, but also their discriminative features compared to other categories. These properties make it a particularly appropriate data representative to characterise and summarise the category.

In this paper, typicality degrees are extended to the unsupervised learning framework, so as to perform clustering, i.e. to identify relevant subgroups in the data set. The underlying idea is that the characterisation of a data subgroup using both common and discriminative features corresponds to the aim of identifying clusters that are both homogeneous and distinct one from another: compactness is directly related to the common features and separability to the discriminative ones.

Therefore a typicality-based clustering algorithm, called TBC, is proposed. It relies on the Gustafson-Kessel principles [3], which makes it possible to identify ellipsoidal clusters and not only spherical ones, through the automatic extraction of the cluster covariance matrices. TBC replaces the membership degrees used in the original Gustafson-Kessel algorithm by typicality degrees and relies on a method to compute the latter in the unsupervised learning framework.

Section 2 recalls the principles of some fuzzy clustering algorithms and justifies the typicality based approach. Section 3 recalls the typicality degree definition in the supervised learning framework and section 4 extends it to the unsupervised case, describing the proposed clustering algorithm. Section 5 illustrates the obtained results on an artificial data set and section 6 concludes the paper.

2 Fuzzy clustering

This section briefly discusses properties of some classic fuzzy clustering algorithms. We first recall the Gustafson-Kessel algorithm and then comment on some of its variants based on different definitions for the data weighting scheme.

In the following, we denote $X = \{x_i, i = 1..n\}$ the data set containing n data points, and c the number of clusters.

Gustafson-Kessel clustering algorithm The Gustafson-Kessel algorithm [3] associates each cluster with both a point and a matrix, respectively representing the cluster centre and its covariance. Whereas the original fuzzy c-means make the implicit hypothesis that clusters are spherical, the Gustafson-Kessel algorithm is not subject to this constraint and can identify ellipsoidal clusters.

More precisely, denoting f_{ir} the influence of point i on cluster r (see below for some definitions), the cluster centre and covariance matrix are computed as

$$w_r = \frac{\sum_{i=1}^{n} f_{ir}^m x_i}{\sum_{i=1}^{n} f_{ir}^m} \quad (1)$$

$$A_r = \sqrt[p]{det(S_r)} S_r^{-1} \quad (2)$$

$$\text{with } S_r = \sum_{i=1}^{n} f_{ir}^m (x_i - w_r)(x_i - w_r)^T$$

m is a user-defined parameter called fuzzifier. The cluster centre is computed as a weighted mean of the data, the weights depending on the considered algorithm, as detailed in the following. The covariance matrix is defined as a fuzzy equivalent of classic covariance. Through eq. (2), a size constraint is imposed on the covariance matrix whose determinant must be 1. As a consequence, the Gustafson-Kessel algorithm can identify ellipsoidal clusters having approximately the same size.

This cluster parameter updating step is alternated with the update of the weighting coefficients until a convergence criterion is met. In the following, we discuss some classic choices for these weights. They are based on comparison between data and cluster centres, and rely on the distance defined as

$$d_{ir} = (x_i - w_r)^T A_r^{-1} (x_i - w_r) \quad (3)$$

Fuzzy c-means (FCM) In the FCM algorithm, the f_{ir} coefficients, usually denoted u_{ir}, are defined as membership degrees

$$u_{ir} = \left(\sum_{s=1}^{c} \left(\frac{d_{ir}}{d_{is}} \right)^{\frac{2}{m-1}} \right)^{-1} \quad (4)$$

where m is a user-defined parameter. These membership degrees indicate the extent to which a point belongs to a cluster, or more precisely the extent to which it is shared between the clusters: the quantities involved in the definition are relative distances that compare the distance to a cluster centre d_{ir} to the distance to other centres d_{is}.

Due to this relative definition, the influence of a point does not decrease with the absolute distance to the centres (see e.g. [8]). This implies that FCM is sensitive to outliers: the latter are considered as equally shared between the clusters and can highly influence the cluster parameters.

Possibilistic c-means (PCM) PCM [5] constitutes a more robust algorithm that relaxes the constraint causing the relative definition of membership degrees in FCM. The f_{ir} coefficients they are based on, usually denoted t_{ir}, measure the absolute resemblance between data points and cluster centres

$$t_{ir} = \left(1 + \left(\frac{d_{ir}^2}{\eta_r} \right)^{\frac{1}{m-1}} \right)^{-1} \quad (5)$$

where η_r is a parameter that evaluates the cluster diameter and can be defined *a priori* or defined from initialisations [4]. Outliers, that are far away from all clusters are then associated with small weights for all clusters and thus do not influence their parameters.

PCM suffers from a coincident cluster problem (see e.g. [4]): in some cases, clusters are confounded whereas natural subgroups in the data are overlooked. Moreover, it has been shown that the objective function global minimum is obtained when all clusters are coincident [12]. Satisfying results are obtained with PCM because the optimisation scheme leads to local minima and not the global minimum. This property is not satisfying from a theoretical point of view.

Possibilistic Fuzzy c-means To solve the PCM coincident cluster problem, Pal et al. [8, 9] propose to combine PCM and FCM: they argue that both possibilistic and membership coefficients are necessary to perform

clustering, respectively to reduce the outlier influence and to assign data points to clusters. Therefore, they take into account both relative and absolute resemblance to cluster centres. In the PFCM [9] algorithm, the combination is performed through a weighted sum, in the form

$$f_{ir} = au_{ir}^{m_1} + bt_{ir}^{m_2} \quad (6)$$

where u_{ir} are the membership degrees defined in eq. (4) and t_{ir} the possibilistic coefficients defined in eq. (5) with η_r replaced by η_r/b. a, b, m_1 and m_2 are user-defined parameters.

The algorithm proposed in this paper also takes into account two components to determine the importance of a data point: it considers the combination of other elements that provide more complementary information and considers a different aggregation scheme, as indicated in the following section, and thus leads to different results.

Other approaches There exist many other approaches to solve the merging cluster problem or that of the outlier sensitivity: the cluster repulsion method [12] e.g. includes in the objective function an additional term to impose repulsion between clusters and prevent their merging. The noise clustering algorithm [2] has a rejecting process for outliers or noisy data, McLachlan and Peel [7] use t-student distributions to better model outliers thanks to heavier tailed distributions.

In this paper, we examine the solution provided when considering the typicality degree framework, whose principles appear relevant for clustering, as described in the next section.

3 Typicality degrees

Principle Typicality degrees were first introduced to build fuzzy prototypes to characterise categories [10]: a prototype is an element chosen to represent a data set and summarise it. The method proposed by Rifqi [10] to construct fuzzy prototypes uses the notion of typicality defined by Rosch [11]: according to this approach, the typicality of a point depends on its resemblance to other members of the category (internal resemblance) and its dissimilarity to members of other categories (external dissimilarity). The prototype derived from such typicality degrees then underlines both the common points of the category members and their discriminative features as opposed to other categories.

The prototype construction method can be decomposed into three steps: computation of (i) the internal resemblance and external dissimilarity, (ii) typicality degrees and (iii) the prototype itself.

Internal resemblance and external dissimilarity For a given data point x belonging to a category C, its internal resemblance $R(x, C)$ and external dissimilarity $D(x, C)$ are respectively defined as its average resemblance to the other members of the category and its average dissimilarity to points belonging to other categories:

$$R(x, C) = avg(\rho(x, y), y \in C) \quad (7)$$
$$D(x, C) = avg(\delta(x, y), y \notin C) \quad (8)$$

ρ (resp. δ) is a resemblance (resp. dissimilarity) measure, i.e. a function that takes as input two data points and returns a value in the interval $[0, 1]$ that measures the similarity (resp. difference) between the two points [1].

Typicality degree The typicality degree of point x for category C is then defined as the aggregation of the internal resemblance and external dissimilarity, as

$$T(x, C) = \varphi(R(x, C), D(x, C)) \quad (9)$$

where φ denotes an aggregation operator such as the average mean or the symmetric sum for instance. It determines the semantics of the prototype, e.g. its being rather a central or discriminative element (see [6] for discussion).

Prototype computation Lastly, the prototype is computed as the aggregation of the most typical data, as

$$p_C = \psi(\{x, T(x, C) > \tau\}) \quad (10)$$

where τ is a user-defined threshold and ψ an aggregation operator: for fuzzy data, it is a

fuzzy aggregator that takes as input fuzzy sets and returns a fuzzy set [10]. For crisp data, it can be a weighted mean, or a more complex operator that aggregates crisp values into a fuzzy set, so as to build a prototype having an imprecise description [6].

4 Typicality degrees for clustering

4.1 Justification

The previous definition of typicality degrees implies that, for specific choices of the aggregator φ, two kinds of points can have low typicality: (i) outliers, that are far away from the core points of the category and thus have low internal resemblance, (ii) points located in overlapping areas between categories, as they are not distinct enough from other categories and thus have low external dissimilarity.

Now these two cases correspond to points that should have low influence on cluster parameter in a clustering task: clusters are expected to be compact and separable, which means they should be robust against outliers and not concentrated in overlapping areas where the distinction between clusters may be difficult. Typicality degrees are directly related to these two desired properties, thus it seems justified to adapt them to unsupervised learning to perform clustering.

4.2 Proposed algorithm architecture

The underlying idea of the typicality-based clustering algorithm, TBC, is to use typicality degrees as weighting coefficients to determine the cluster parameters. TBC is not based on the optimisation of a cost function, but directly on update functions to be alternated: it consists in alternatively computing typicality degrees for each data point, and updating the cluster parameters according to eq. (1) and (2) using these typicality degrees. These two steps are alternated until convergence of the centre positions.

The cluster parameter update process is then the same as in the Gustafson-Kessel algorithm (cf. eq. (1-2)). In the following, the typicality degree update process is described, as an adaptation of the previous methodology when the available information are cluster centres, covariance matrices and typicality degrees obtained from the previous step.

4.3 Assignment computation

Assignment computation role The computation of typicality degrees relies on a crisp partition of the data: the typicality degree of a point is non-zero only for the category it belongs to; moreover assignment is necessary to compute internal resemblance and external dissimilarity.

In the clustering case, clusters must be questioned, thus typicality is computed with respect to all clusters and not only the one a point is assigned to.

Thus the assignment is only used for the computation of internal resemblance and external dissimilarity: for a given point x, and a cluster C, the internal resemblance is defined as the average resemblance between x and points assigned to C. When, in turn, typicality degrees are computed for points assigned to C, they are computed for all clusters and not only with respect to C. The assignment remains only a hypothesis.

Assignment definition As seems natural, TBC assigns points to clusters according to their maximal typicality degree: a point is assigned to the cluster it is most typical of.

A special case is considered for points for which all typicality degrees are small (below 0.1 in our tests): such points, that correspond to outliers, should not be assigned to any cluster, as they are not typical of any. Indeed, if they were assigned to a cluster, they would arbitrarily lower the internal resemblance value for all points in the cluster: they would correspond to an especially low resemblance value and would thus distort the average value computation (see eq. (7)), disturbing the whole process. It is to be noted that these points are still involved in the cluster parameter estimation, with low influence due to their low typicality degrees. Their special handling only concerns the assignment step.

4.4 Comparison measure choice

Having defined a crisp partition of the data according to previously obtained typicality degrees, internal resemblance and external dissimilarity can be computed for all points and all clusters. To that aim, comparison measures must be defined, they involve the available cluster covariance matrices.

Resemblance measure Resemblance measures are normalised functions that indicate the extent to which two points are similar [1]. By analogy with PCM (see eq. (5)), the Cauchy function is used

$$\rho(x,y) = \frac{1}{1+\frac{d^2(x,y)}{\eta}}$$

The resemblance measure is applied to points belonging to the same cluster, therefore it should be adapted to each cluster: one resemblance measure per cluster is considered by using for each cluster the distance associated to its covariance matrix (see eq (3)). The normalising coefficient η is also determined locally: its square root corresponds to the distance from which the resemblance value is smaller than 0.5. Its value is chosen as being half the cluster diameter.

At the beginning of the process these cluster diameters are not known, as neither clusters nor their covariance matrices are known. As inappropriate normalisation factors could bias the resemblance measure and lead to inappropriate resemblance values, we apply the same process as for PCM: after convergence of the alternating scheme, the initial values for these parameters are updated and the alternating scheme is applied again with the new values.

Dissimilarity measure Dissimilarity measures are normalised functions that indicate the extent to which two points are different one from another [1]. A measure also based on a Cauchy function is used

$$\delta(x,y) = 1 - \frac{1}{1+\frac{d^2(x,y)}{\eta}}$$

with a different distance function d and another normalisation coefficient η: the dissimilarity measure is used to compute external dissimilarities, i.e. it has an inter-cluster meaning. Therefore, d is here chosen to be the Euclidian distance, and η is defined so that the dissimilarity is 0.9 for points such that their distance equals half the data diameter.

4.5 Aggregation operator choice

Typicality degrees are then deduced from internal resemblance and external dissimilarity by aggregation. In the supervised case, many choices are possible, depending on the desired semantics of the prototype [6].

In the clustering case, the aggregator should be a conjunctive operator, so that points are considered as typical only if they possess both high internal resemblance and external dissimilarity. Otherwise, outliers may have high typicality degrees for all clusters due to their high external dissimilarity (in the supervised case, this can be interesting if a discriminative prototype is desired). Therefore a t-norm is chosen (Lukasiewicz t-norm in the tests, $\varphi(a,b) = \max(a+b-1,0)$).

4.6 Overall algorithm

TBC can be summarised as follows. It only requires the user to set a single argument, the number of clusters. After an initialisation step through a few iterations of FCM, initial values for the data partition and the cluster diameters are estimated and used for the computation of an initial typicality degree matrix. The iterating loop is then applied a first time. Cluster diameters are then updated, and the loop is applied a second time.

The iterating loop consists in alternatively computing typicality degrees and cluster parameters (centres and covariance matrices) until convergence of the centre positions. Typicality degrees are computed as detailed above. The cluster parameter update equations are the same as in the Gustafson-Kessel algorithm (cf. eq. (1) and (2)), using as influence coefficients the typicality degrees.

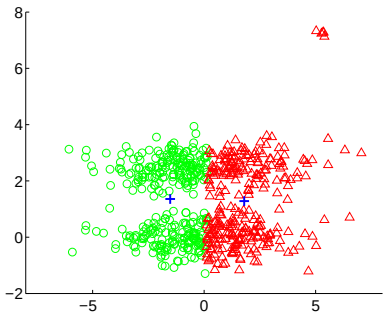


Figure 1: Results obtained with FCM.

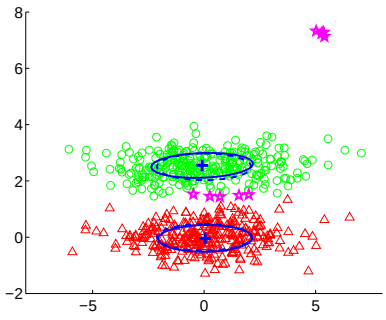


Figure 2: Results obtained with the proposed typicality-based clustering algorithm TBC.

5 Numerical experiments

5.1 Considered setup

Experiments were performed to compare the proposed TBC algorithm with the Gustafson-Kessel algorithm with fuzzy and possibilistic partitions (respectively denoted GKfcm and GKpcm) and the adaptation of the PFCM [9] algorithm to the detection of ellipsoidal clusters (denoted GKpfcm). The latter consists in applying the update equations for cluster parameters (eq. (1-2)), using as weighting coefficients the coefficients as defined in eq. (6).

The considered artificial dataset consists of two Gaussian distributed clusters and a small outlying group in the upper right corner (see figures). The lower and upper Gaussian clusters respectively have for centres and covariance matrices,

$$\begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 2.5 \end{bmatrix} \Sigma_1 = \Sigma_2 = \begin{bmatrix} 4.47 & 0 \\ 0 & 0.22 \end{bmatrix}$$

For the figures, points are assigned according to the maximal value of their coefficient (membership degree, possibilistic coefficient or typicality degree depending on the considered method). In the GKpfcm case, assignment is performed using the membership degrees. Each symbol depicts a different cluster, the plus sign represents the cluster centres, the ellipses represents the covariance matrix, the dashed one is the true covariance. In the case of GKpcm and TBC, stars represent points for which no assignment is relevant, i.e. points for which coefficients are smaller than 0.1 for both clusters.

Parameters were chosen as $c = 2$, $m = 2$ for GKfcm and GKpcm, $a = 1$, $b = 5$, $m = \eta = 1.5$ for GKpfcm, corresponding to values leading to the best results.

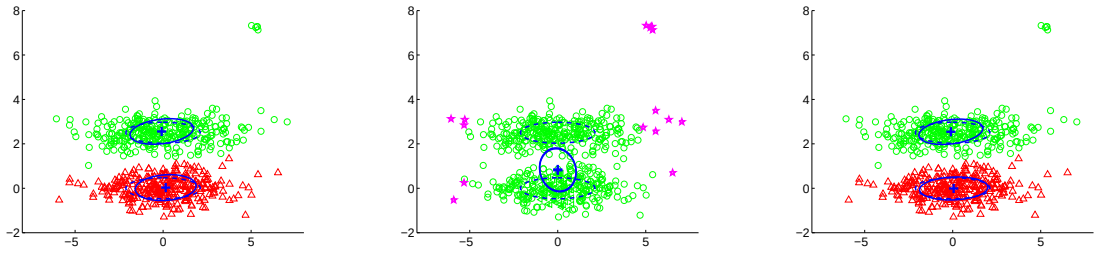
5.2 Obtained results

Figure 1 shows the results obtained using the fuzzy c-means to underline the necessity of extracting the covariance matrices to detect the expected subgroups: FCM cannot adapt to the elongated clusters and produces a counterintuitive result.

Figures 2 and 3 show the obtained partitions with the Gustafson-Kessel variants, table 1 the values of the cluster parameters. The indicated errors are computed as the square root sum of the square difference with the true parameters (w_i and Σ_i, $i = 1..2$).

Table 1 shows that TBC is indeed competitive, it produces the best estimates for the cluster parameters. In particular, it leads to a clearly smaller error value for the covariance matrices: the estimates are very close to the true values for w_1 and Σ_1 and better for the second cluster than the ones provided by the other algorithms.

GKpcm fails to identify the expected clusters (see also fig. 3b) because it produces two confounded clusters that do not reflect the dataset structure. Still, it can be seen that the outlying data are recognised as specific data: they are not assigned to any of cluster, as the coefficients are very small for both clusters. Likewise, the extreme points of the



(a) (b) (c)

Figure 3: Results obtained with (a) GKfcm, (b) GKpcm, (c) GKpfcm.

Algorithm	GKfcm	GKpcm	GKpfcm	TBC
Centres	$\begin{bmatrix} 0.15 & 0.03 \\ -0.08 & 2.56 \end{bmatrix}$	$\begin{bmatrix} 0.01 & 0.81 \\ 0.00 & 0.85 \end{bmatrix}$	$\begin{bmatrix} 0.06 & -0.01 \\ -0.11 & 2.54 \end{bmatrix}$	$\begin{bmatrix} 0.05 & -0.04 \\ -0.09 & 2.55 \end{bmatrix}$
Centre error	0.11	0.84	0.06	0.05
Covariance 1	$\begin{bmatrix} 3.00 & 0.14 \\ 0.14 & 0.34 \end{bmatrix}$	$\begin{bmatrix} 1.07 & -0.10 \\ -0.10 & 0.94 \end{bmatrix}$	$\begin{bmatrix} 3.77 & 0.10 \\ 0.10 & 0.27 \end{bmatrix}$	$\begin{bmatrix} 4.45 & 0.04 \\ 0.04 & 0.23 \end{bmatrix}$
Covariance 2	$\begin{bmatrix} 3.23 & 0.27 \\ 0.27 & 0.33 \end{bmatrix}$	$\begin{bmatrix} 1.08 & -0.10 \\ -0.10 & 0.94 \end{bmatrix}$	$\begin{bmatrix} 3.30 & 0.28 \\ 0.28 & 0.33 \end{bmatrix}$	$\begin{bmatrix} 5.19 & 0.13 \\ 0.13 & 0.20 \end{bmatrix}$
Cov. error	1.98	4.91	1.43	0.75

Table 1: Cluster parameters and error obtained by GKfcm, GKpcm, GKpfcm and TBC.

two elongated clusters are considered as special points and not assigned to the clusters.

It can be seen that GKfcm is influenced by the outliers, in particular, the covariance matrices are attracted by the outlying group (see fig. 3a and table 1). Its influence is especially noticeable in the estimation of the covariance between the two attributes of the upper Gaussian: the latter gets a high value, because the outlying group distorts the estimation. On the contrary, TBC is not biased towards these values, which explains its very low error value. For GKfcm, the error is due to the membership degree normalisation process: the latter cannot take simultaneously small values for both clusters. The outlying points have membership degrees around 0.6 and 0.4 for the upper and lower Gaussian cluster respectively.

In order to better interpret the results of the GKpfcm algorithm, figure 4a represents the weighting coefficient values for all data as a function of their position on the y-axis: it can be seen that, for GKpfcm, data in the outlying group have a weight comparable to the major part of the data in the bigger clusters. In TBC case (fig. 4b), their typicality is significantly lower, which explains their lower influence.

As regards the comparison between GKpfcm and TBC, it is moreover to be noted that the weighting coefficients can be exploited directly in the case of TBC, to characterise the data set further, whereas they do not have an intuitive interpretation in the case of GKpfcm. This is due to the fact that GKpfcm combines information about absolute and relative resemblance, whereas typicality degrees are based on more complementary components.

Figure 4 also shows that typicality degrees take into account both internal resemblance and external dissimilarity: the typicality degrees curves are not symmetric as compared to the mean of the clusters. Indeed, points located between the two clusters tend to have smaller typicality degrees, because they are not distinct enough from points belonging to other clusters. This is the reason why some points are not assigned (see fig. 2) in the area between the two Gaussian clusters. Globally unassigned points correspond to points having either a too low internal resemblance (outliers) or a too low external dissimilarity.

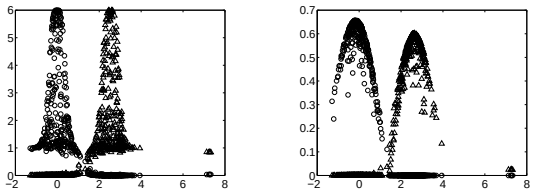


Figure 4: Values of the weighting coefficients used in GKpfcm (eq. (6)) and typicality degrees for all data points as a function of their position on the y-axis. Note that the scale differs from one graph to the other.

This property can lead to a cluster repulsion effect: data located in such a way between cluster only apply a small attraction on the cluster centres that are thus not attracted towards overlapping areas. This effect is similar to that introduced by [12] in the objective function: in TBC, it follows from the definition of the weighting coefficient and it is not expressed in the cluster centre definition.

6 Conclusion

This paper presented the extension of the typicality degree framework to unsupervised learning to perform clustering. First results indicate promising properties of the proposed algorithm and justify the proposed approach. A more comprehensive study of the algorithm is necessary to validate it.

One limitation of TBC comes from the fact that typicality degrees are based on a crisp partition of the data. This imposes an assignment step that could favourably be replaced by a more flexible definition of typicality degrees based on membership degrees. It must be noted that this step requires a precise study: membership degrees are related to the resemblance to the cluster centre, which is related to the notion of internal resemblance. It is thus necessary to examine the role of the latter in the computation of the internal resemblance involved in the process itself.

Acknowledgements

This research was supported by a Lavoisier grant from the French Ministère des Affaires Etrangères.

References

[1] B. Bouchon-Meunier, M. Rifqi, and S. Bothorel. Towards general measures of comparison of objects. *Fuzzy sets and systems*, 84(2):143–153, 1996.

[2] R. Davé. Characterization and detection of noise in clustering. *Pattern Recognition Letters*, 12:657–664, 1991.

[3] E. Gustafson and W. Kessel. Fuzzy clustering with a fuzzy covariance matrix. In *Proc. of IEEE CDC*, 1979.

[4] F. Höppner, F. Klawonn, R. Kruse, and T. Runkler. *Fuzzy Cluster Analysis, Methods for classification, data analysis and image recognition*. Wiley, 2000.

[5] R. Krishnapuram and J. Keller. A possibilistic approach to clustering. *IEEE Trans. on fuzzy systems*, 1:98–110, 1993.

[6] M.-J. Lesot, L. Mouillet, and B. Bouchon-Meunier. Fuzzy prototypes based on typicality degrees. In *8th Fuzzy Days*, pages 125–138, 2004.

[7] G. McLachlan and D. Peel. Robust cluster analysis via mixtures of multivariate t-distributions. *Lecture Notes in Computer Science*, 1451:658–666, 1998.

[8] N. Pal, K. Pal, and J. Bezdek. A mixed c-means clustering model. In *Fuzz-IEEE'97*, pages 11–21, 1997.

[9] N. Pal, K. Pal, J. Keller, and J. Bezdek. A new hybrid c-means clustering model. In *Fuzz-IEEE'04*, pages 179–184, 2004.

[10] M. Rifqi. Constructing prototypes from large databases. In *Proc. of IPMU'96*, 1996.

[11] E. Rosch. Principles of categorization. In E. Rosch and B. Lloyd, editors, *Cognition and categorization*, pages 27–48. Lawrence Erlbaum associates, 1978.

[12] H. Timm and R. Kruse. A modification to improve possibilistic fuzzy cluster analysis. In *Fuzz-IEEE'02*, pages 1460–1465, 2002.