Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

Variable-Geometry Clustering and Its Optimization

Adam Pedrycz, Fangyan Dong, and Kaoru Hirota
Department of Computational Intelligence and Intelligent Informatics
Tokyo Institute of Technology
Yokohama, Japan

***Abstract*—Clustering is often viewed as a synonym of techniques used to reveal the structure in data. The inherent geometrical diversity of data is a strong motivating factor to search for *geometrically flexible* clusters design supported by the clustering algorithms. In this study, we introduce a concept of geometrically variable fuzzy clustering (making use of Fuzzy C-Means, FCM), in which the fuzzification coefficients are associated with individual clusters thus endowing them with significant geometric flexibility. We introduce a hybrid optimization environment in which both global and local optimization mechanisms are engaged. The global optimization is supported by evolutionary computing (and particle swarm optimization, PSO, in particular) whereas the local optimization is realized by adopting some modified iterative schemes encountered in FCM. We show that this hybrid vehicle of optimization is of interest when dealing with comprehensive fitness functions which quantify a general view at the results of clustering (such as e.g., the one expressed by cluster validity indexes or the one articulating the mapping- reconstruction capabilities of the clusters).**

***Keywords*—clustering, variable-geometry, optimization, Fuzzy C-Means (FCM), Particle Swarm Optimization (PSO)**

I. Introduction

Recent years saw a wealth of developments in the realm of clustering and fuzzy clustering. The concept of cluster or more generally, information granule, is a fundamental one that supports a remarkable variety of pursuits. In the landscape of fuzzy clustering, there are several interesting and far-fetching developments in terms of the underlying concepts, augmentations of the existing techniques and improvements of the optimization procedures. Not claiming completeness of the coverage, let us highlight the main directions which are quite apparent in the existing literature:

Conceptual enhancements of clustering—Those that are realized in terms of enhancing flexibility of clusters and a way in which the overall stricture becomes articulated. This pertains to ideas such as possibilistic clustering, clustering with noise cluster (which forms an interesting way of absorbing noise component), clustering with geometric variety (such as C-lines, C-varieties, shell clustering, etc.). Some representative studies are included in [2][7][11][12] [13].

Improvements of optimization strategies supporting the construction of clustering techniques—While the generic computing vehicles are well established and quite efficient (even though they do not guarantee global minimum of objective functions guiding the formation of clusters, e.g., Fuzzy C-Means), we note the prominence of optimization pursuits aimed at global optimization. The framework of Evolutionary Computing (genetic algorithms, evolutionary strategies, ant colonies, particle swarm optimization) is intensively researched in the context of clustering with intent to support more effective exploration of the search space and providing an ability to produce better clustering results. One can refer here to [1][3][5][6][8][9][14][15] which come as a testimony to the activities present in this domain.

Developments of mechanisms of collaborative clustering and incorporation of domain knowledge (which gives rise to so-called knowledge-based clustering)—The dichotomy of supervised-unsupervised learning is augmented by recognizing that there is a broad spectrum of situations in which there is some element of supervision, say a small subset of labeled patterns [13]. The other component of supervision comes in the form of proximity constraints which generalize the binary format of constraints that are given in the form of *must-link* or *should-not-link* relationships.

There are two fundamental issues that always call for more attention, namely (a) a way of endowing the clustering with more geometric flexibility which makes the clusters more flexible and yet highly interpretable, and (b) bringing more advanced optimization techniques.

Introduced is a concept of variable geometry fuzzy clustering in which each cluster exhibits its own easily interpretable geometry. This helps capture some essential local properties of data which otherwise would not have been taken into consideration when exercising optimization at the global level. We consider a class of objective function-based fuzzy clustering such as the well-known Fuzzy C-Means (FCM) [2] which comes with a wealth of numeric investigations, experiments, and conceptual enhancements. In this sense, the study contributes to the already well-documented assets of clustering techniques available in the FCM setting. The objective is to exploit the geometric diversity supplied by modifiable fuzzification coefficients whose values are adjusted locally to the characteristics of the individual clusters.

The study is structured into 6 sections. In Section 2, we start with the fundamental concept by showing how the clusters of variable geometry emerge, what they entail in terms of

978-1-4244-2794-9/09/$25.00

knowledge representation and how this concept can be realized within a suitable algorithmic setting. The detailed elaborations on the optimization procedure and their hybrid nature (combining global and local search techniques) are covered in Section 3. The evaluation criteria to be used to guide the mechanism of global search are presented in Section 4 where we emphasize the diversity of tasks in which the clustering results are used. Next, in Section 5 discussed are results of numeric experimentation.

Throughout the study, we adhere to the standard notation used in pattern recognition and system modeling. As usual, we consider a set of data (patterns) $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_N$ located in the n-dimensional feature space, $\boldsymbol{x}_k \in \boldsymbol{R}^n$, k=1,2,…, N. The distance used in this space is denoted by $\|\,.\,\|$ and its semantics will be clear from the context of the presentation.

II. Clusters of Variable Geometry – Their Concept and Realization

Before proceeding with the concept of clusters of variable geometry, it is instructive to recall some main ideas pertaining to cluster geometry as they have been realized in most clustering available in the literature.

The geometry of fuzzy clusters (and clusters) is implied by several essential parameters of the clustering algorithm. In the FCM clustering (which is commonly encountered in various studies both of conceptual and applied nature), the minimized objective function

$$Q = \sum_{i=1}^{c} \sum_{k=1}^{N} u_{ik}^{m} \|\boldsymbol{x}_k - \boldsymbol{v}_i\|^2 \qquad (1)$$

contains several adjustable components (parameters) which impact the geometry of clusters. (As usual $\boldsymbol{v}_i$, i = 1, 2, …, c denote the prototypes of the clusters whereas $U = [u_{ik}]$ is a partition matrix, k = 1, 2, …, N while "m" is the fuzzification coefficient). First, the distance function between patterns and prototypes plays a crucial role as its choice implies what structure in the data is preferred when searching for clusters. This aspect is somewhat overlooked and when talking about clustering one usually views it as a synonym of unsupervised learning. While this is a sound observation, at the same time one must be cognizant that there is an obvious component of implicit supervision injected into the clustering method upfront in the sense that the predefined distance function predisposes the clustering method to focus search on groups that conform to this particular geometry. Let us recall that the geometry of points which are equidistant from the origin depends on the distance: in case of the Hamming distance we encounter diamond-like shapes which in the Tchebyshev distance we end up with a collection of "hyperboxes". The Euclidean distance leads to the circles while the Mahalanobis distance implies some hyperellipsoidal shapes (which could be also rotated versus axes of the coordinate system).
The fuzzification coefficient (m) present in (1) substantially affects the shapes of the clusters. The commonly used case of m = 2 produces Gaussian-like shapes of the clusters (membership functions). For the values of "m" close to 1, the membership functions start to resemble characteristic functions with quite steep boundaries. Higher values of "m", say around 3-4, produces "spiky" clusters with rapidly declining membership values when moving away from the prototypes. The changes in the distance function and/or the fuzzification coefficient affects the geometry of all clusters in the same way meaning that the structure in data is searched for at the global level. As the feature space in which data are positioned could be quite diversified and locally there could be profound geometric differences when moving from cluster to cluster, it would be advantages to capture this diversity at a local level by admitting different values of the fuzzification coefficient for each cluster, that is m_1, m_2, …, and m_c. Thus the objective function which incorporates these fuzzification coefficients comes in the form

$$Q = \sum_{i=1}^{c} \sum_{k=1}^{N} u_{ik}^{m_i} \|\boldsymbol{x}_k - \boldsymbol{v}_i\|^2 \qquad (2)$$

The optimization of Q, in addition to the "classic" portion of the problem, which involves the determination of prototypes and the partition matrix, has to include the vector of the fuzzification coefficients, $\boldsymbol{m} = [m_1\ m_2 \ldots\ m_c]$. More formally, we express the problem as follows

$$\text{Min } Q \text{ with respect to } U \in \mathrm{U},\ v_1, v_2, \ldots v_c, m \qquad (3)$$

where along the "usual" constraints imposed on the partition matrix we require that all entries of $\boldsymbol{m}$ are greater than 1, $\boldsymbol{m} > \boldsymbol{1}$.

Leaving the detailed optimization aspects to be discussed in the next section, it is beneficial to gain a better insight what type of geometric variability the fuzzification coefficients bring into considerations and how this flexibility could impact/enhance the applied facet of clustering. To offer a convincing yet simple illustrative example, let us consider a one-dimensional case (n = 1) and three clusters only (c = 3). Depending upon the values of m, the membership functions exhibit quite diverse shapes.

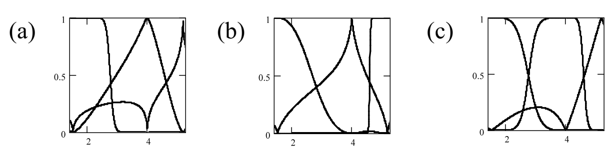


Figure 1. Membership functions for selected values of m: (a) $\boldsymbol{m}$ = [1.2 2.5 4.5], (b) $\boldsymbol{m}$ = [2.0 3.5 1.1], (c) $\boldsymbol{m}$ = [1.6 1.4 2.8]. In all cases the prototypes are the same, v_1 = 1.5, v_2 = 4.0 and v_3 = 5.2.

The boundary between the clusters (i.e., the collection of points where the membership degrees to the clusters are the same) is located somewhere in-between the prototypes of the clusters, say v_1 and v_2. The detailed relationship reads as follows

$$\frac{1}{1+\left(\frac{x-v_1}{x-v_2}\right)^{\frac{2}{m-1}}}=\frac{1}{1+\left(\frac{x-v_2}{x-v_1}\right)^{\frac{2}{m-1}}} \quad (4)$$

Simple algebraic manipulations lead to the expression $|x-v_1| = |x-v_2|$ and subsequently the boundary x_0 comes as the average of the prototypes, $x_0= (v_1+v_2)/2$. What is even more important is the fact that for the fixed prototypes, the value of the fuzzification coefficient does not influence the location of the boundary. In other words, we may not see too much value in changing the value of "*m*" with intent of shifting the position of the boundary. On the other hand, different values of the fuzzification coefficient coming with each cluster yield the expression

$$\frac{1}{1+\left(\frac{x-v_1}{x-v_2}\right)^{\frac{2}{m_1-1}}}=\frac{1}{1+\left(\frac{x-v_2}{x-v_1}\right)^{\frac{2}{m_2-1}}} \quad (5)$$

whose solution with respect to x_0 depends upon the values of m_1 and m_2. Results for selected values of m_1 and m_2 (see Table 1), show that these fuzzification coefficient directly impact the location of the boundary between the clusters. This demonstrates that the fuzzification coefficients offer flexibility which could be directly utilized in the design of classifiers.

TABLE I. VALUES OF THE BOUNDARY BETWEEN CLUSTERS

m_1/m_2	1.1	2.0	3.5
1.1	2.75	2.88	2.81
2.0	2.62	2.75	3.02
3.5	2.70	2.48	2.75

Boundary values for selected values of m_1 and m_2. The prototypes are fixed and equal to $v_1 = 1.5$ and $v_2 = 4.0$.

With regard to clusters of variable geometry, one could mention the well-known Gustafson-Kessel (GK) method [7] in which each cluster is characterized by its own Mahalanobis distance function of the form $||\boldsymbol{x}-\boldsymbol{v}_i||^2 = (\boldsymbol{x}-\boldsymbol{v}_i)^{\mathrm{T}} \sum^{-1}(\boldsymbol{x}-\boldsymbol{v}_i)$ where Σ_i is a fuzzy covariance matrix of the i^{th} cluster which controls its shape. In contrast to the method proposed here in which *m* comes with a straightforward interpretation (as it tells us immediately about the shape of membership functions), in the GK clustering method the visually appealing interpretation of the covariance matrix is not available. The determination of the entries of the covariance matrix itself is also more cumbersome and requires some assumptions to be made upfront (typically, we require that the determinant of the covariance matrix is fixed, that is for all clusters we require that $\det(\Sigma_i) = \rho > 0$).

III. OPTIMIZATION PROCEDURE

The optimization problem expressed by (2) is more complex than the one encountered in the standard FCM. Recall that in the latter case, for the fixed number of clusters and the fuzzification coefficient and the distance taken as the Euclidean one, there is an iterative optimization scheme in which we successively update the partition matrix and the prototypes following the formulas

- update of partition matrix $u_{ik}=\frac{1}{\sum_{j=1}^{c}\left(\frac{\|\boldsymbol{x}_k-\boldsymbol{v}_i\|}{\|\boldsymbol{x}_k-\boldsymbol{v}_j\|}\right)^{\frac{2}{m-1}}}$
- update of prototypes $\boldsymbol{v}_i=\frac{\sum_{k=1}^{N}u_{ik}^m\boldsymbol{x}_k}{\sum_{k=1}^{N}u_{ik}^m}$

(6)

The process iterates until there are no substantial changes in the entries of the successive partition matrices. This optimization scheme does not guarantee convergence to the global minimum of the objective function (and, as practice shows, the results produced in this way are meaningful). The problem encountered here involves another set of parameters contained in the vector of the fuzzification coefficients ***m***. Notably, the optimization of the entries of ***m*** cannot be directly realized through gradient-based optimization. Similarly, there is a strong monotonicity between the values of *m* and the resulting objective function, which could easily misguide techniques of local optimization preventing from a thorough exploration of the parameter space. Taking this into consideration, there is a strong motivation behind the use of evolutionary optimization whose population-based search strategy offers a possibility of carrying out global search. Furthermore evolutionary methods can cope with complex fitness functions and this could help directly use the results of clustering in the predefined application context. There is panoply of mechanisms of evolutionary optimization. When making a proper selection, one has to look for computational costs, effectiveness, and suitability to handle some category of the optimization problems. One of the alternatives, which has been recognized in the literature as an effective optimization method is Particle Swarm Optimization (PSO), see [10]. The reader may refer to the abundant literature with this regard including also a number of variants of the generic version of the PSO.

Let us consider a certain fitness function *V* (which is not restricted to the original objective function *Q* yet *Q* could be viewed as one of the realizations of *V*). The nature of the fitness function depends upon the way in which the results of clustering will be used or the quality of clustering be assessed– the details will be presented in the next section.

To summarize, all optimization components are put together and portrayed in a schematic format in Figure 2.

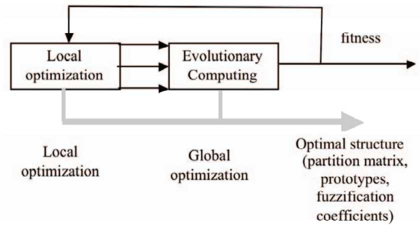


Figure 2. An overview of optimization realized at global and local level.

Here we envision a collaborative optimization that involves some mechanisms of gradient-based optimization or other local search–oriented optimization is embedded in the more general settings of the PSO. While PSO supports global optimization, the effectiveness of the method is higher when the size of the search space is limited which is particularly critical in case of complicated and computationally demanding fitness functions. Furthermore if there is a possibility to construct a solution in an analytical fashion, this hybrid optimization strategy is always highly desirable.

Here the following optimization scenario is realized. We introduce the expression

$$u_{ik} = \frac{1}{\sum_{j=1}^{c} \frac{\|\boldsymbol{x}_k - \boldsymbol{v}_i\|^{\frac{2}{m_i - 1}}}{\|\boldsymbol{x}_k - \boldsymbol{v}_i\|^{\frac{2}{m_j - 1}}}} \tag{7}$$

as a way of computing the partition matrix; note that we do not claim that these calculations realize some optimization procedure but rather we consider that this expression to be a meaningful way of determining the membership grades. The prototypes are determined in a way they minimize (2) with respect to $\boldsymbol{v}_i$; that is $\nabla_{v_i} Q = 0$, which when using the Euclidean distance produces a global minimum of Q. The detailed formula comes as the weighted sum of the form

$$\boldsymbol{v}_i = \frac{\sum_{k=1}^{N} u_{ik}^{m_i} \boldsymbol{x}_k}{\sum_{k=1}^{N} u_{ik}^{m_i}} \tag{8}$$

The vector of the fuzzification coefficients is subject to the PSO minimization of V. The flow of optimization consists of two phases: each individual located in the search space of $\boldsymbol{m}$'s triggers the search in the space of the partition matrices and the prototypes. At this inner optimization loop, in a series of iterations the prototypes are update following (7) while the calculations of the partition matrix are governed by (6). As there is no formal proof of convergence of these iterations, we also realize a stopping criterion for this inner loop by terminating calculations if the objective function starts to increase. The general overview of the overall optimization scheme is shown in Figure 3. The nested nature of the optimization is highlighted here.

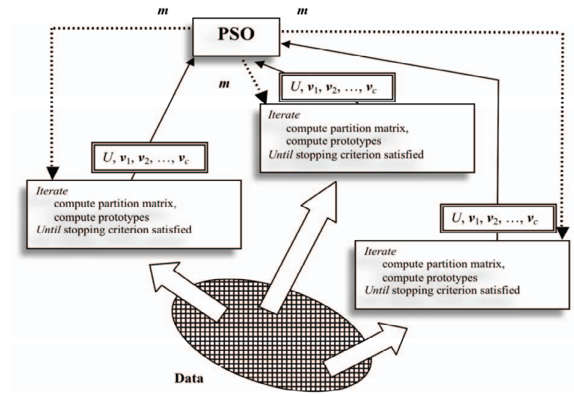


Figure 3. Details of the optimization setup; show are linkages between local (iterative scheme) and global (PSO).

IV. Evaluation Criteria

Clustering results can be used in many different ways; this naturally leads to the use of a certain fitness function. We consider three groups of the tasks whose goal is clearly delineated and thus the resulting criterion is equipped with a well-defined semantics.

A. Validity of Structure

Here we are focused on answering the fundamental question: how valid are obtained results of clustering? Is the structure being revealed (discovered) truly the structure which is present in the data set or the results of clustering are merely superimposed on the data at hand? There is also another fundamental question as to the number of clusters in the data. In those cases we encounter a great deal of cluster validity indexes and those can be used to serve as examples of fitness function.

B. Reconstruction Capabilities of the Structure

Here we are concerned with the ability of the clusters – information granules to represent (or approximate) original data. The idea links directly with the concept of quantization or data compression – as a matter of fact this is the essence of information granulation. Being more specific, we express the quality of reconstruction as the sum of distances between original data $\boldsymbol{x}_k$ and its reconstructed version $\hat{\boldsymbol{x}}_k$ that is $V = \sum_{k=1}^{N} \| \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k \|^2$. This sum is treated as the fitness function. We determine $\hat{\boldsymbol{x}}_k$ as a solution to the following optimization problem

$$\sum_{i=1}^{c} u_{ik}^{m_i} \| \boldsymbol{x}_k - \hat{\boldsymbol{x}}_k \|^2 \tag{9}$$

in which the reconstruction of $\boldsymbol{x}_k$ is realized on a basis of the collection of the prototypes where the corresponding elements of the partition matrix serve as the weights of the respective distances. By minimizing this sum of weighted distances from the corresponding prototypes (assuming the Euclidean distance in this expression), the straightforward calculations produce the following expression

$$\hat{\boldsymbol{x}}_k = \frac{\sum_{i=1}^{c} u_{ik}^{m_i} \boldsymbol{v}_i}{\sum_{i=1}^{c} u_{ik}^{m_i}} \qquad (10)$$

(note that $\hat{\boldsymbol{x}}_k$ obtained in this way assures global minimum of the reconstruction error).

C. Classification and/or Mapping Usefulness of the Clusters

Quite often clustering is regarded as a preliminary phase of system modeling where the revealed structure is used to construct the detailed model. In essence, the clusters, say prototypes, form a blueprint of the overall model. For instance, we use prototypes to position receptive fields in RBF neural networks. Likewise we can treat the membership functions as the receptive fields (which is advantages in the sense we do not require to determine and adjust the sizes of the receptive fields, say by modifying the spreads of the Gaussian functions). Typically, fuzzy clusters are used to form rules in fuzzy rule-based modeling. In all those cases the effectiveness of fuzzy clustering is measured implicitly by quantifying the quality of the resulting model. In this sense, the fitness function is the performance measure of the model and it becomes apparent that in this optimization framework there are no explicit expressions using which one could exploit mechanisms of gradient –based learning.

Any of these categories of the problems comes with its own specific fitness function which is next optimized by the PSO.

V. NUMERICAL ILLUSTRATION

The experiments presented in this section which were completed for several data sets coming from Machine Learning Repository (http://archive.ics.uci.edu/ml/) serve as an illustration of the performance of the clustering method and a nature of the results, especially in terms of the values of the fuzzification coefficient. For comparative reasons, we present the results when the fuzzification coefficient has been selected to be equal to 2 (a scenario which is commonly present in the literature). As to the evaluation criterion of the underlying optimization we consider the reconstruction abilities achieved by the clusters. Obviously, any other criterion could be also taken into consideration. The PSO optimization environment was realized with the following parameters: population size of 20 individuals, number of iterations of 10, *c1*, *c2* = random numbers in [0,2]. This specific numeric value of the inertial weight is the one commonly encountered in the literature (and we did not feel compelled to make any adjustments with this regard). In some other cases, the numeric values were selected on a basis of some experimental evidence gathered during experiments (and this aspect is concerned with the size of the population and the number of iterations used to run the optimization).

A. Boston Housing Data

In the experiments, the number of clusters was varied in-between 2 and 16 and the corresponding values of the reconstruction criterion V were recorded. The clustering was run for a fixed value of the fuzzification coefficient $m = 2$ and the optimized values of the vector $\boldsymbol{m}$. In all cases, there is an improvement in terms of the lower values of the reconstruction error, see Figure 4. Figure 5 shows the values of the optimized fuzzification coefficients for selected values of c. Interestingly, the optimal entries of $\boldsymbol{m}$ vary from cluster to cluster showing that the improvement comes as a result of allowing for variability in the geometry of the clusters.

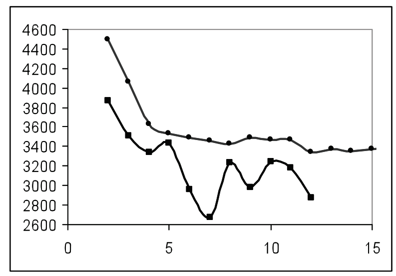
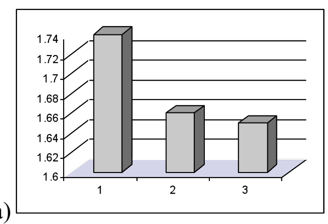
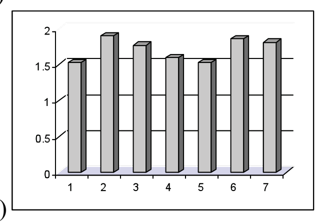


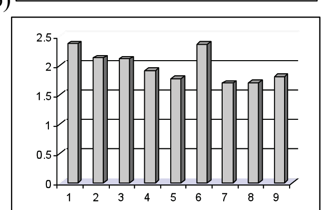
Figure 4. Reconstruction error V versus the number of clusters: upper curve - m=2, lower curve – variable fuzzification coefficient.



(a)



(b)



(c)

Figure 5. Bar plots of the fuzzification coefficients for selected numbers of clusters: (a) c=3, (b) c=8, and (c) c=9.

B. auto_mpg Data

The results for this data set are visualized in Figure 6 and 7, respectively. In all cases, we note a reduction in the values of the reconstruction error resulting from the use of variable geometry of the clusters. There is also a significant diversity in the values of the optimized fuzzification coefficients.

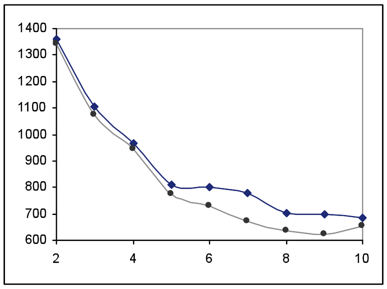


Figure 6. Reconstruction error V versus the number of clusters: upper curve - m=2, lower curve – variable fuzzification coefficient.

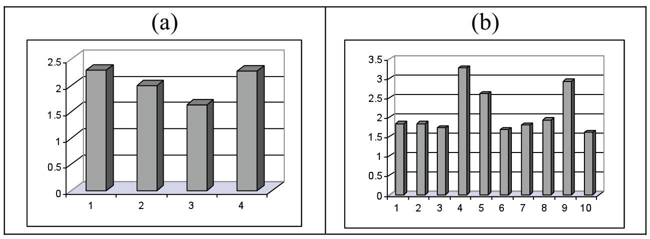


Figure 7. Bar plots of the fuzzification coefficients for selected numbers of clusters: (a) *c*=4, (b) *c*=10.

VI. CONCLUSIONS

The geometric diversity of fuzzy clusters implemented in this study through modifiable fuzzification coefficient endows the clustering method with additional flexibility. This augmentation of the method comes with high interpretability as the role of fuzzification coefficients is well understood. The optimization setup has been extended by bringing together the techniques of global optimization (PSO) and local optimization (realized through gradient-based learning). This hybrid optimization is highly beneficial when dealing with complex fitness functions (and in this way one can incorporate a variety of general design criteria ranging from cluster validity indices, reconstruction criterion, and performance indices pertinent to fuzzy models whose integral structural component are the clusters formed by grouping the data.

There is another interesting and promising direction which could augment the investigations presented in this study. The distance function itself could be allowed to vary from cluster to cluster and its optimization offers another component of flexibility one can exploit when searching for structure in data. Here the Minkowski distance between $\boldsymbol{x}$ and $\boldsymbol{y}$ expressed as $\|\boldsymbol{x}-\boldsymbol{y}\| = \left(\sum_{j=1}^{n} |x_j - y_j|^r\right)^{1/r}$ with $r \geq 1$ arises here as a viable alternative. In fact, we have an infinite family of distances indexed by some parameter "*r*" It is well-known that the Hamming, Euclidean and Tchebyshev distances are particular cases of the Minkowski distance for r = 1, 2, and ∞, respectively. As in case of the fuzzification coefficients, the optimization with respect to "*r*" can be completed through genetic optimization (and the PSO, in particular). The overall objective function in this case comes in the form

$$Q = \sum_{i=1}^{c} \sum_{k=1}^{N} u_{ik}^{m_i} \|\boldsymbol{x}_k - \boldsymbol{v}_i\|_{r_i}^2 \quad (11)$$

and the associated vector of parameters to be optimized consists of two parts, that is [***m r***] with ***r*** being a *c*-dimensional vector containing parameters of the Minkowski distances used individually for the clusters. The flexibility inbuilt into the clustering method could be also utilized in case of clustering with partial supervision where the available constraints could be accommodated through the proper adjustments of the parameters of the method.

ACKNOWLEDGMENT

The Monbukagakusho Scholarship from the Government of Japan is gratefully acknowledged.

REFERENCES

[1] Bandyopadhyay, S., and U. Maulik. "Nonparametric genetic clustering: comparison of validity indices." *IEEE Transactions on Systems, Man, and Cybernetics - Part C* 31, (2001) 120-125.

[2] Bezdek, J. C. *Pattern recognition with fuzzy objective function algorithms.* Plenum Press, N. York, 1981.

[3] Bezdek, J. C., and R. J. Hathaway. "Optimization of fuzzy clustering criteria using genetic algorithms." *Proc. IEEE World Congress on Computational Intelligence.* 1994. 589-594.

[4] EE Elbeltagi, TE Hegazy, DE Grierson, Comparison among five evolbtionaryEbaseb optimization algorithms, *Advanced Engineering Anvarc atics*, 1l , 2t t g, 4TEgTE

[5] Estivill-Castro, V., and A. T. Murray. "Spatial clustering for data mining with genetic algorithms." *Proc. Int. ICSC Symposium on Engineerign of Intelligent Systems.* 1997. 317-323.

[6] GE Gan, aE , b, ZE Yang, D genetis fbzzy pEE obes algorithm for slbstering sategorisal bata, *Edvert t stec s with Avvlicatians,* T1, 2, 111gE112t E

[7] EEEEGbstafson, , ECEy essel, Fbzzy slbstering 4 ith a fbzzy sovarianse matrie, *AracEAEEE canvEan Decisian and cantral*, 1l el , gan Diego, CD, e11Ee11E

[8] Hall, L. O., I. B. Özyurt, and J. C. Bezdek. "Clustering with a genetically optimized approach." *IEEE Trans. on Evolutionary Computation* 3 (1999): 103-112.

[9] Hruschka, Eduardo R., Ricardo J. G. B. Campello, Alex A. Freitas, and André C. P. L. F. Carvalho. "A Survey of Evolutionary Algorithms for Clustering." *IEEE Transactions on Systems, Man, and Cybernetics - Part C: Applications and Reviews*, 2008: to appear.

[10] Kennedy, J., and R. C. Eberhart. *Swam Intelligence.* Morgan Kaufmann, CA, 2001.

[11] Klawonn, F., and A. Keller. "Fuzzy clustering with evolutionary algorithms." *Proc. Seventh IFSA World Congress.* 1997. 312-323.

[12] Krishnapuram, R., O. Nasraoui, and H. Frigui. "The fuzzy c-spherical shells algorithm: A new approach." *IEEE Transactions on Neural Networks*, 1992: 663-671.

[13] aEl alente be Oliveira, , EGebrysz, *Advances in i uDD clustering and Ats Avvlicatians*, aE, iley, Hobopen, y a, 2t t eE

[14] Van Le, T. "Evolutionary fuzzy clustering." *Proceedings of IEEE International Conference on Evolutionary Computation.* 1995. 753-758.

[15] Zhao, L., Y. Tsujimura, and M. Gen. "Genetic algorithm for fuzzy clustering." *Proceedings of IEEE International Conference on Evolutionary Computation.* 1996. 716-719.