https://doi.org/10.20965/jaciii.2008.p0448

Paper:

Algorithms for Sequential Extraction of Clusters by Possibilistic Method and Comparison with Mountain Clustering

Sadaaki Miyamoto*, Youhei Kuroda, and Kenta Arai

*Department of Risk Engineering, School of Systems and Information Engineering, University of Tsukuba
Ibaraki 305-8573, Japan
E-mail: miyamoto@risk.tsukuba.ac.jp

[Received October 10, 2007; accepted February 15, 2008]

In addition to fuzzy c-means, possibilistic clustering is useful because it is robust against noise in data. The generated clusters are, however, strongly dependent on an initial value. We propose a family of algorithms for sequentially generating clusters "one cluster at a time," which includes possibilistic medoid clustering. These algorithms automatically determine the number of clusters. Due to possibilistic clustering's similarity to the mountain clustering by Yager and Filev, we compare their formulation and performance in numerical examples.

Keywords: possibilistic clustering, sequential extraction of clusters, mountain clustering

1. Introduction

Fuzzy *c*-means clustering currently plays the central role in clustering algorithms [1–3], but there is another useful tool of possibilistic clustering [4], which both resembles and differs from fuzzy *c*-means. Possibilistic clustering is robust against noise in data [5]. Results of possibilistic clustering, however, tend to strongly dependent on initial values. Although there is a proposal of an algorithm generating "one cluster at a time" [5] to mitigate this drawback, this proposal has not yet been put into a concrete algorithm.

We propose algorithms for sequentially extracting clusters one by one and present an objective function having the form $J(V)$ of a variable of cluster centers, instead of $J(U,V)$ of the two variables of memberships and centers in fuzzy *c*-means and possibilistic clustering in the original formulation.

We also consider how these algorithms are related to the mountain clustering algorithm [6], and present examples showing properties of the proposed algorithms and compare performance of the proposed method and the mountain clustering.

Our proposal determines the number of clusters automatically, and more efficient than the mountain clustering, which is also sequential and similar to our proposal.

2. Possibilistic Clustering Formulations

An object of clustering is assumed to be a vector in p-dimensional Euclidean space: $x_k = (x^1, \ldots, x_k^p) \in \boldsymbol{R}^p$, $(k = 1, \ldots, n)$, and the set of objects is denoted by $X = \{x_1, \ldots, x_n\}$. Cluster centers are denoted by $v_i = (v_i^1, \ldots, v_i^p)^T$, $i = 1, \ldots, c$, where c is the number of clusters. Simplified symbol $V = (v_1, \ldots, v_c)$ is used for the collection of c cluster centers, whereas cluster membership matrix $U = (u_{ki})$, $(i = 1, \ldots, c, k = 1, \ldots, n)$ is used as usual, where u_{ki} is the degree of belongingness of object x_k to cluster i. In fuzzy *c*-means clustering, the constraint for a fuzzy partition is given by

$$M = \{U = (u_{ki}) : \sum_{i=1}^{c} u_{ki} = 1, \forall k; u_{kj} > 0, \forall j, k\},$$

while in possibilistic clustering we set

$$M = \{U = (u_{ki}) : u_{kj} \geq 0, \forall j, k\}, \quad (1)$$

where we have omitted original constraint $0 < \sum_k u_{ki} \leq N$ proposed in [4], for simplicity.

The dissimilarity of clustering is the standard squared Euclidean distance between an individual and a cluster center:

$$D_{ki} = \|x_k - v_i\|^2.$$

We use the dissimilarity between generic elements $x, y \in \boldsymbol{R}^p$, denoted by

$$D(x,y) = \|x - y\|^2.$$

2.1. Fuzzy *c*-Means and Possibilistic Clustering

Fuzzy *c*-means [1] and possibilistic clustering are based on the optimization of an objective function. We consider the following two:

$$J_e(U,V) = \sum_{k=1}^{n} \sum_{i=1}^{c} \{u_{ki} D_{ki} + \lambda^{-1} u_{ki} (\log u_{ki} - 1)\} \quad (2)$$

$$J_2(U,V) = \sum_{k=1}^{n} \sum_{i=1}^{c} \{(u_{ki})^2 D_{ki} + \zeta^{-1} (1 - u_{ki})^2\} \quad (3)$$

J_e is an entropy-based objective function [7], and J_2 is a function for the possibilistic clustering [4] with a restriction to $m = 2$ and $\eta_i = \zeta^{-1}$ $(1 \leq i \leq c)$.

We focus on the possibilistic clustering, because the fuzzy c-means objective function by Bezdek [1] and Dunn [8] provides no help because it gives the trivial solution of $u_{ki} = 0$ in possibilistic clustering, while Eqs. (2) and (3) are useful for both possibilistic clustering and fuzzy c-means.

In the alternative optimization **PCM** of fuzzy c-means clustering, we use $J = J_e$ or J_2 and constraint (1) for possibilistic clustering.

Algorithm PCM
PCM0: Set an initial value $\bar{V}$.
PCM1: Find the optimal solution of J for U while V is fixed: put

$$\bar{U} = \arg\min_{U \in M} J(U, \bar{V}).$$

PCM2: Find the optimal solution of J for V while U is fixed: put

$$\bar{V} = \arg\min_{V} J(\bar{U}, V).$$

PCM3: If solution $(\bar{U}, \bar{V})$ is convergent, stop; else go to **PCM1**.
End PCM.

We show solutions for each step, writing u_{ki} instead of $\bar{u}_{ki}$, v_i instead of $\bar{v}_i$, without confusion, for simplicity.

$$u_{ki} = \exp(-\lambda D_{ki}) \quad \ldots\ldots\ldots\ldots \quad (4)$$

for J_e, or

$$u_{ki} = \frac{1}{1 + \zeta D_{ki}} \quad \ldots\ldots\ldots\ldots\ldots \quad (5)$$

for J_2, while

$$v_i = \frac{\sum_{k=1}^{n} (u_{ki})^m x_k}{\sum_{k=1}^{n} (u_{ki})^m}. \quad \ldots\ldots\ldots\ldots \quad (6)$$

in **PCM2** where $m = 1$ for J_e and $m = 2$ for J_2.

3. Sequential Extraction of Clusters

We define two functions related to Eqs. (4) and (5):

$$U_e(x_k, y) = \exp(-\lambda D(x_k, y)) \quad \ldots\ldots\ldots \quad (7)$$

$$U_2(x_k, y) = \frac{1}{1 + \zeta D(x_k, y)} \quad \ldots\ldots\ldots \quad (8)$$

Note $U_e(x_k, v_i) = u_{ki}$ for J_e; $U_2(x_k, v_i) = u_{ki}$ for J_2.

To see the properties of possibilistic clustering, we substitute $U(V) = (U_e(x_k, v_i))_{i=1,\ldots,c}$ into $J_e(U, V)$ where v_i is regarded as a variable, yielding

$$J_e(U(V), V) = -\lambda \sum_{i=1}^{c} \sum_{k=1}^{n} \exp(-\lambda D(x_k, v_i)).$$

If we put $J'_e(V) = J_e(U(V), V)$ and

$$j_e(y) = -\sum_{k=1}^{n} \exp(-\lambda D(x_k, y)),$$

we have

$$J'_e(V) = J_e(U(V), V) = \lambda \sum_{i=1}^{c} j_e(v_i).$$

Note that this substitution changes the original formulation of possibilistic clustering, so the optimization of $J'_e(V)$ is not identical to the alternative minimization of $J_e(U, V)$, so they are expected to have similar properties (cf. [9]). We therefore study the properties of $J'_e(V)$ as a function of V.

Note first that $J'_e(V)$ is the sum of $j_e(v_i)$. Since no constraint is imposed on v_i, every $j_e(v_i)$ can be minimized independently from other $j_e(v_j)$ $(j \neq i)$. If we assume the minimizing element is unique, minimization leads to $\bar{v} = v_1 = \cdots = v_c$ and therefore only one cluster center is obtained as the minimizing element of $J'_e(V)$. This means that if we want to have multiple clusters from this function, we must search for different minimizing solutions of a multimodal function, which is far more difficult than minimizing a unimodal function.

This leads us to the idea of extracting "one cluster at a time," discussed by Davé and Krishnapuram [5]. Note that this idea has not yet been fully developed, so we discuss their idea in greater detail in developing new algorithms.

We consider $J_2(U, V)$ in the same way. Substitute $U(V) = (U_e(x_k, v_i))_{i=1,\ldots,c}$ into $J_2(U, V)$ in which v_i is a variable. This yields

$$J'_2(V) = J_2(U(V), V) = \sum_{i=1}^{c} \sum_{k=1}^{n} \frac{D(x_k, v_i)}{1 + \zeta D(x_k, v_i)}.$$

We put

$$j_2(y) = \sum_{k=1}^{n} \frac{D(x_k, y)}{1 + \zeta D(x_k, y)},$$

and it follows that

$$J'_2(V) = \sum_{i=1}^{c} j_2(v_i).$$

Note again that $j_2(v_i)$ can be minimized independently from other $j_2(v_j)$. We thus have the unique solution $\hat{v} = v_1 = \cdots = v_c$ that minimizes $J'_2(V)$.

These observations justify the use of an algorithm to extract "one cluster at a time." generally done as follows.

SC: General Procedure for Sequential Clustering Algorithms
SC1. Let the initial set of objects be $X^{(0)} = X$ and $k = 0$. Let function $J(v; k) = j_e(v)$ (or $J(v; k) = j_2(v)$) with the set of objects $X^{(k)}$.
SC2. Search the minimizing element of $J(v; k)$:

$$v^{(k)} = \arg\min_{v} J(v; k)$$

SC3. Extract cluster $G^{(k)}$ that belongs to center $v^{(k)}$.
SC4. Let $X^{(k+1)} = X^{(k)} - G^{(k)}$. If $X^{(k+1)}$ does not have sufficient elements to extract one more cluster, stop; else $k := k + 1$ and go to step **SC2**.
End SC.

With this procedure, we need not specify the number of clusters beforehand.

Before considering a minimization algorithm which is not yet specified in this procedure, we compare the two objective functions and review the mountain clustering. Note that we call extracting one cluster at a time a sequential possibilistic clustering algorithm.

Note: The "one cluster at a time" concept has already been proposed [5], but the above analysis and algorithms below have not yet been studied.

3.1. Comparison of Objective Functions

To study properties of $J_e'(V)$ and $J_2'(V)$, we study $j_e(y)$, the sum of $-U_e(x_k,y)$, and $j_2(y)$, the sum of $U_2(x_k,y)$:

$$j_e(y) = -\sum_{k=1}^{n} U_e(x_k,y),$$
$$j_2(y) = \sum_{k=1}^{n} U_2(x_k,y).$$

$U_e(x_k,y)$ and $U_2(x_k,y)$, as the function of y, has properties similar in the minimization of $j_e(y)$ and $j_2(y)$. The next two propositions are easily obtained, and their proofs are omitted here.

Proposition 1: Function $U_e(x_k,y)$ satisfies

$$-U_e(x_k,x_k) = \min_y -U_e(x_k,y) = -1,$$
$$\lim_{\|y\|\to\infty} -U_e(x_k,y) = 0,$$

while $U_2(x_k,y)$ satisfies

$$U_2(x_k,x_k) = \min_y U_2(x_k,y) = 0, \quad \lim_{\|y\|\to\infty} U_2(x_k,y) = \frac{1}{\zeta}.$$

Proposition 2: Put $z = D(x_k,y)$. Then,

$$-U_e(x_k,y) = -\exp(-\lambda z), \quad U_2(x_k,y) = \frac{z}{1+\zeta z}.$$

If we consider

$$g_e(z) = -\exp(-\lambda z),$$
$$g_2(z) = \frac{z}{1+\zeta z}, \quad (0 \le z < +\infty),$$

as a function of real variable z, then both functions $g_e(z)$ and $g_2(z)$ are monotonically increasing.

3.2. Mountain Clustering

Note that the mountain clustering [6] is closely related to the proposed method. Indeed, the mountain clustering extracts clusters sequentially, i.e., one cluster at a time.

The mountain function is

$$M(y) = \sum_{k=1}^{n} \exp(-\alpha D(x_k,y)), \quad (\alpha > 0) \quad . \quad . \quad . \quad (9)$$

where $y \in \boldsymbol{R}^p$ is restricted to grid points. Let $y^{(1)}$ be the maximizing point of Eq. (9). The second mountain function is defined as follows:

$$M^{(2)}(y) = M(y) - M(y^{(1)}) \sum_{k=1}^{n} \exp(-\alpha D(y^{(1)},y)).$$

Calculation is then repeated:

$$M^{(\ell)}(y) = M(\ell-1)(y) - M(y^{(\ell-1)}) \sum_{k=1}^{n} \exp(-\alpha D(y^{(\ell-1)},y)).$$

. . . . . (10)

until no significant cluster remains. The stopping criterion is given by the ratio and a given parameter $\delta > 0$:

$$\frac{M(y^{(1)})}{M(y^{(\ell-1)})} < \delta. \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (11)$$

Note that

Proposition 3: $M(y)$ and $-j_e(y)$ have the identical form when $\alpha = \lambda$:

$$M(y) = -j_e(y) \sum_{k=1}^{n} U_e(x_k,y) = \sum_{k=1}^{n} \exp(-\alpha D(x_k,y)).$$

4. Sequential Clustering Algorithms

The main problem in the procedure **SC** is how to optimize the function $J(v;k) = j_e(v)$ (or $J(v;k) = j_2(v)$). Minimization is done in three ways to optimize $J(v;k)$.

The first way is simplest, minimizing the objective function on the finite set $\{y_1,\ldots,y_L\}$.

Procedure A

A1. Generate candidate points $y_1,\ldots,y_L \in \boldsymbol{R}^p$. $X^{(0)} = X$ and $k = 0$.

A2. Find minimizing element

$$\bar{y} = \arg \min_{v=y_1,\ldots,y_L} J(v;k).$$

A3. Find cluster $G^{(k)}$ with the center $\bar{y}$. Extract $G^{(k)}$: $X^{(k+1)} = X^{(k)} - G^{(k)}$. If $X^{(k+1)}$ does not have sufficient elements to extract one more cluster, stop; else $k := k+1$ and go to **A2**.

Points $y_1,\ldots,y_L$ take values at grid points, similar to the mountain clustering, or can be chosen randomly from X. Alternatively, we can take $\{y_1,\ldots,y_L\} = X$, then the method may be called a one-pass algorithm of *sequential possibilistic medoid* calculation, since the medoid is the cluster center that corresponds to an object (cf. Kaufman, Rousseeuw [10] for hard c-medoid).

The second procedure, similar to the ordinary alternative minimization of fuzzy c-means, is also useful, but requires more calculation than the first method.

Procedure B

B1. Generate candidate points $y_1,\ldots,y_L \in \boldsymbol{R}^p$ as initial cluster centers. $X^{(0)} = X$ and $k = 0$.

B2. Repeat the calculation of u_{ki} and v_i until convergence. Converged points are denoted by $z_1,\ldots z_\ell$. Find minimizing element

$$\bar{z} = \arg \min_{v=z_1,\ldots,z_\ell} J(v;k). \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad . \quad (12)$$

B3. Find cluster $G^{(k)}$ with center $\bar{z}$. Extract $G^{(k)}$: $X^{(k+1)} = X^{(k)} - G^{(k)}$. If $X^{(k+1)}$ does not have sufficient

elements to extract one more cluster, stop; else $k := k+1$ and go to **B2**.

Several cluster centers are simultaneously obtained if we do not impose condition (12), so this procedure is generally useful as an algorithm of ordinary possibilistic clustering (cf. [11]).

4.1. Possibilistic Medoid Calculation

The next procedure may be called generalized multi-pass possibilistic medoid clustering.

Procedure C
C1. Generate candidate points $y_1,\ldots,y_L \in X$ and choose initial cluster centers $z_1,\ldots,z_c$ from $Y = \{y_1,\ldots,y_L\}$. $X^{(0)} = X$ and $k = 0$.
C2. Repeat **C3** until convergence.
C3. Let $y_{i1},\ldots,y_{ki} \in Y$ be K-nearest elements to z_i $(i = 1,\ldots,c)$.
Find minimizing element

$$\bar{z}_i = \arg \min_{v=z_i,y_{i1},\ldots,y_{ki}} J(v;k).$$

Put $z_i = \bar{z}_i$.
C4. Let

$$\bar{z} = \arg \min_{v=z_i,y_{i1},\ldots,y_{ki}} J(v;k).$$

Find cluster $G^{(k)}$ with center $\bar{z}$. Extract $G^{(k)}$: $X^{(k+1)} = X^{(k)} - G^{(k)}$. If $X^{(k+1)}$ does not have sufficient elements to extract one more cluster, stop; else $k := k+1$ and go to **C2**.

This requires more calculation than procedures A and B, and seems less useful for sequential possibilistic clustering, but should be considered to find a medoid as a cluster center. With a slight modification, Procedure C is used for ordinary possibilistic medoid clustering. We thus have the next procedure.

Procedure C' (ordinary possibilistic medoid clustering)
C'1. Generate candidate points $y_1,\ldots,y_L \in X$ and choose initial cluster centers $z_1,\ldots,z_c$ from $Y = \{y_1,\ldots,y_L\}$.
C'2. Repeat **C'3** until convergence.
C'3. Let $y_{i1},\ldots,y_{ki} \in Y$ be K-nearest elements to z_i $(i = 1,\ldots,c)$.
Calculate

$$\bar{u}_{ki} = \exp(-\lambda D(x_k,z_i)), \quad \forall k,i,$$
$$\bar{V} = \arg \min_{v_i=z_i,y_{i1},\ldots,y_{ki}} J(\bar{U},V).$$

Put $z_i = \bar{v}_i$ $(i = 1,\ldots,c)$.

5. Examples

Figures 1 and **2** show clusters obtained from Procedures B and C for the same set of points on a plane. Objective function J'_e with $\lambda = 30.0$ has been used and the extraction of objects uses the crisp criterion of

$$G^{(k)} = \{x_\ell \in X : \exp(-\lambda D(x_\ell, v^{(k)})) \geq \beta\}.$$

with parameter $\beta = 0.2$.

Numbers $1,\ldots,5$ at lower right imply the number of clusters sequentially extracted with centers shown by small circles: number 1 is the first cluster, number 2 is the second, *etc*.

In **Fig. 1**, clusters 1 to 4 are extracted except for misclassified points shown by asterisks ($*$) and plusses ($+$). After the fourth cluster have been extracted, no object remains. Note that the correct number of four clusters

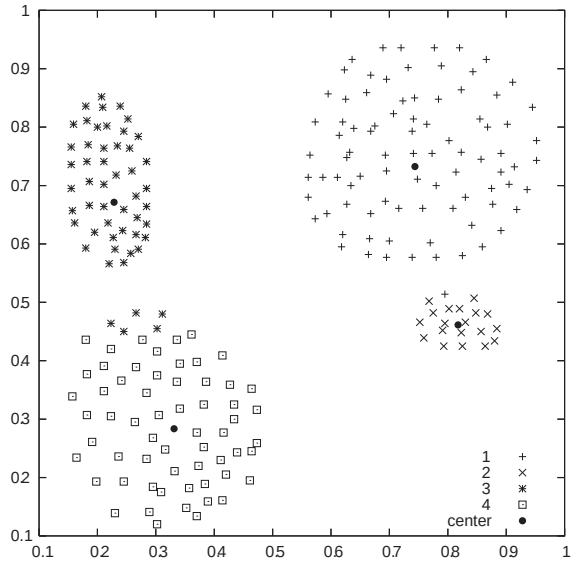

Fig. 1. Clusters $1-4$ sequentially extracted from a set of points on a plane using Procedure B.

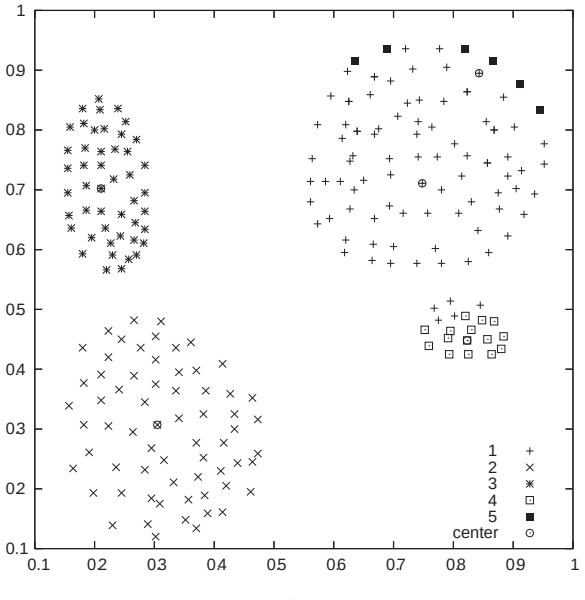

Fig. 2. Clusters $1-5$ sequentially extracted from a set of points on a plane using Procedure C.

was extracted even though the number of clusters was not given beforehand.

In **Fig. 2**, the medoid algorithm was used with all objects as $\{y_1, \ldots, y_L\}$ in Procedure C, with a result similar to that in **Fig. 1** except that a fifth cluster was detected, although not well-separated from other clusters. This algorithm thus fails to find the correct number of clusters, but provides acceptable overall performance.

The medoid method generally does not perform as well as *c*-means, although medoids may be preferred in some real applications.

Note: Misclassification in **Figs. 1** and **2** is reduced if we use fuzzy *c*-means using KL-information [12]. Using KL information requires a large calculation and the number of clusters must be given beforehand, whereas our proposal does not require that the number of clusters be known.

Note: We omit the result from Procedure A because it is similar to that for Procedure C, which generally requires more calculation than Procedure A, but yields more stable results, because an iterative algorithm is used.

6. Comparison to the Mountain Clustering

The number of grid points increases exponentially with dimension, so it is expected that the mountain clustering processing time also increases exponentially, which is a major drawback. Processing time for our proposal does not increase rapidly with the number of dimensions.

This was verified by a numerical experiment in which *p*-dimensional data sets, each having 200 points, have been generated randomly. For $p = 2, 3, 4, 5, 6$, 100 trials were done and average processing time was recorded. Algorithm **A** and the mountain clustering were applied to the same data sets, where five grid points in the mountain clustering for each coordinate were used (**Table 1**).We observe the rapid growth of processing time in the mountain clustering, while our proposal using algorithm **A** does not rapidly increase. The same numbers are shown in **Fig. 3** where the horizontal axis is the dimension and the vertical axis is the logarithmic scale of processing time. The efficiency of the sequential algorithm is observed.

Table 1. Comparison of processing time by the mountain clustering and the sequential clustering with the algorithm **A**. The hyphen (−) shows that calculation was stopped before convergence because of too much processing time.

Dimension p	Mountain (ms)	Sequential (ms)
2	24.84	30.46
3	772.96	35.62
4	20453.333	42.03
5	485758.59	60.47
6	−	74.68

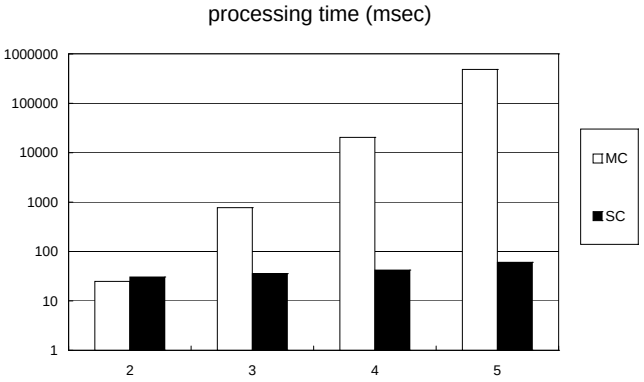

Fig. 3. Mountain (MC) versus sequential algorithm **A**(SC) processing time for dimensions $p = 2, 3, 4, 5$.

7. Conclusions

We have proposed algorithms for sequential extraction of clusters linking possibilistic clustering [4, 5] and the mountain clustering [6] by eliminating the membership matrix and considering the objective function of cluster centers alone. We also studied a medoid algorithm related to sequential extraction and the possibilistic clustering, and compared performance using numerical examples. We found that the our proposal is far more efficient than the mountain clustering when the data space dimension is large. Our proposal does not require the number of clusters, unlike ordinary fuzzy *c*-means.

We clarified possibilistic clustering and its variations, finding that the mountain clustering may have another option of taking random points instead of grid points [6], and $J_2'(V)$ can be used for the objective function in the mountain clustering. We also proposed the use of possibilistic medoids, but could demonstrate no advantage to them, although applications such as document retrieval may find a medoid useful as a representative object of a cluster than a centroid.

To summarize, possibilistic clustering is a useful sequential algorithm thanks to the automatic determination of the number of clusters. Many fuzzy *c*-means variations thus invite further investigation of both methodological features and applications.

Acknowledgements

This study was supported by Grants-in-Aid for Scientific Research, Japan Society for the Promotion of Science, Nos.16300065 and 19300074.

References:

[1] J. C. Bezdek, "Pattern Recognition with Fuzzy Objective Function Algorithms," Plenum, New York, 1981.
[2] J. C. Bezdek, J. Keller, R. Krishnapuram, and N. R. Pal, "Fuzzy Models and Algorithms for Pattern Recognition and Image," Proc. Kluwer, Boston, 1999.
[3] F. Höppner, F. Klawonn, R. Kruse, and T. Runkler, "Fuzzy Cluster Analysis," Wiley, Chichester, 1999.
[4] R. Krishnapuram and J. M. Keller, "A possibilistic approach to clustering," IEEE Trans. on Fuzzy Syst, Vol.1 , No.2, pp. 98-110, 1993.

[5] R. N. Davé and R. Krishnapuram, "Robust clustering methods: a unified view," IEEE Trans. Fuzzy Syst, Vol.5, No.2, pp. 270-293, 1997.
[6] R. R. Yager and D. Filev, "Approximate clustering via the mountain method," IEEE Trans., on Syst, Man, and Cybern, Vol.24, No.8, pp. 1279-1284, 1994.
[7] S. Miyamoto and M. Mukaidono, "Fuzzy *c*-means as a regularization and maximum entropy approach," Proc. of the 7th Int. Fuzzy Systems Association World Congress (IFSA'97), June 25-30, 1997, Prague, Czech, Vol.II, pp. 86-92, 1997.
[8] J. C. Dunn, "A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters," J. of Cybernetics, Vol.3, pp. 32-57, 1974.
[9] T. A. Runkler and C. Katz, "Fuzzy Clustering by Particle Swarm Optimization," 2006 IEEE Int. Conf. on Fuzzy Systems, Vancouver, BC, Canada, pp. 3065-3072, July 16-21, 2006.
[10] L. Kaufman and P. J. Rousseeuw, "Finding Groups in Data: An Introduction to Cluster Analysis," Wiley, 1990.
[11] S. Miyamoto, R. Inokuchi, and Y. Kuroda, "Possibilistic and Fuzzy c-Means Clustering with Weighted Objects," Proc. of 2006 IEEE Int. Conf. on Fuzzy Systems, Vancouver, BC, Canada, pp. 4260-4265, July 16-21, 2006.
[12] H. Ichihashi, K. Miyagishi, and K. Honda, "Fuzzy *c*-means clustering with regularization by K-L information," Proc. of 10th IEEE Int. Conf. on Fuzzy Systems, Vol.2, pp. 924-927, 2001.

Name:
Sadaaki Miyamoto

Affiliation:
Professor, Department of Risk Engineering, Faculty of Systems and Information Engineering, University of Tsukuba

Address:
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:
1978-1990 Joined University of Tsukuba
1990-1994 Professor, University of Tokushima
1994- Professor, University of Tsukuba

Main Works:
• "Fuzzy Sets in Information Retrieval and Cluster Analysis," Theory and Decision Library, Series D, Kluwer Academic Publishers, pp. 1-259, 1990.
• S. Miyamoto, H. Ichihashi, and K. Honda, "Algorithms for Fuzzy Clustering," Springer, Heidelberg, pp. 1-247, 2008.

Membership in Academic Societies:
• The Institute of Electrical and Electronics Engineers (IEEE)
• The Society of Instrument and Control Engineers of Japan (SICE)
• Japan Society for Fuzzy Theory and Systems (SOFT)

Name:
Youhei Kuroda

Affiliation:
Masters Program in Risk Engineering, Graduate School of Systems and Information Engineering, University of Tsukuba

Address:
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:
2006- Masters Program, Graduate School of Systems and Information Engineering, University of Tsukuba

Main Works:
• S. Miyamoto, Y. Kuroda, and K. Arai, "Algorithms for Sequential Extraction of Clusters by Possibilistic Clustering," Proc. of MDAI 2007, 2007.

Name:
Kenta Arai

Affiliation:
Masters Program in Risk Engineering, Graduate School of Systems and Information Engineering, University of Tsukuba

Address:
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573, Japan

Brief Biographical History:
2007 - Masters Program, Graduate School of Systems and Information Engineering, University of Tsukuba

Main Works:
• S. Miyamoto, Y. Kuroda, and K. Arai, "Algorithms for Sequential Extraction of Clusters by Possibilistic Clustering," Proc. of MDAI 2007, 2007.