Variables for Controlling Cluster Sizes on Fuzzy c-Means

Yoshiyuki Komazaki[1] and Sadaaki Miyamoto[2]

[1] Master's Program in Risk Engineering,
University of Tsukuba
Ibaraki 305-8573, Japan
s1220596@u.tsukuba.ac.jp

[2] Department of Risk Engineering
University of Tsukuba
Ibaraki 305-8573, Japan
miyamoto@risk.tsukuba.ac.jp

Abstract. The fuzzy c-means proposed by Dunn and Bezdek is one of the most popular methods of fuzzy clustering. Clusters obtained by the fuzzy c-means are in the Voronoi sets when crisp reallocation rule is applied. This means that a part of a larger cluster may be assigned to a smaller one when there are clusters of different sizes. Therefore, some methods using variables for controlling cluster sizes have been proposed. In this paper, we study their theoretical properties and compare them using numerical examples.

1 Introduction

Fuzzy clustering means a method of clustering with fuzzy membership function for clusters. Fuzzy c-means proposed by Dunn [1] and Bezdek [2] is the most popular one, which we call here the standard fuzzy c-means (SFCM). SFCM has a simple objective function, and thus it has been studied by many authors and many different methods of fuzzy clustering have been proposed.

A major drawback to SFCM clustering is that it tends to make clusters of equal sizes. Namely, a part of a large cluster is misclassified as one of a smaller cluster if volumes of clusters are out of balance. Therefore some approaches using variables controlling cluster sizes have been proposed for tackling such a problem, and we discuss three methods here. One is derived from a modified entropy-based fuzzy c-means [3]. Another is a fuzzy extension of the maximum likelihood procedure [4], and the third is fuzzy c-means proposed by Ichihashi et al. [5], whose results are expected to be similar to those of the Gaussian mixture model.

All of these methods can solve the problem of cluster sizes. Nevertheless, there is no comparative study of these methods from theoretical viewpoint, and these methods are still open to discuss. The purpose of this paper is to study theoretical properties of these methods. We discuss them based on classifier functions [6] and thus our conclusions have generality.

V. Torra et al. (Eds.): MDAI 2013, LNAI 8234, pp. 192–203, 2013.

We first show SFCM as a basic algorithm of fuzzy c-means and three methods using variables for controlling cluster sizes in Section 2. Further, we show theoretical properties of these methods based on classifier functions in Section 3. We apply these methods to illustrative examples and show effectiveness and these properties with brief interpretations in Section 4. Finally, Section 5 concludes the paper.

2 Fuzzy c-Means with Cluster Sizes

In this section, we show the standard fuzzy c-means (SFCM) introduced by Dunn [1] and Bezdek [2] and algorithms with variables for cluster sizes [3,4,5].

2.1 Fuzzy c-Means

Let $X = \{x_1, \dots, x_n\}$ be a set of objects for clustering. They are points in the p-dimensional Euclidean space $\mathcal{R}^p$. Let $V = \{v_1, \dots, v_c\}$ be a set of centers of cluster i and let $U = (u_{ik})$ be an $c \times n$ matrix of fuzzy membership of x_k to cluster i. x_k and v_i are both p-dimensional vectors, i.e., $x_k = (x_k^1, \dots, x_k^p)^T$ and $v_i = (v_i^1, \dots, v_i^p)^T$.

SFCM is based on minimization of the following objective function:

$$J_{sfcm} = \sum_{i=1}^{c} \sum_{k=1}^{n} (u_{ik})^m d_{ik}, \tag{1}$$

where d_{ik} is dissimilarity between x_k and v_i; m is fuzzy parameter which is larger than 1. Note that the objective function is obviously equal to that of k-means if the fuzzy parameter m is 1. The constraint of U is

$$\mathcal{U} = \{(u_{ik}) : u_{ik} \in [0,1], \sum_{i=1}^{c} u_{ik} = 1, \forall k\}. \tag{2}$$

Unless noted otherwise, d_{ik} is the squared Euclidean norm:

$$d_{ik} = \|x_k - v_i\|^2 = \sum_{l=1}^{p} (x_k^l - v_i^l)^2. \tag{3}$$

The following iterative algorithm for minimizers J_{sfcm} is used.

Step 1. Generate c initial values for centroids V.
Step 2. Calculate optimal U that minimizes J_{sfcm} .
Step 3. Calculate optimal V that minimizes J_{sfcm} .
Step 4. If (U, V) is convergent, stop; else return to Step 2.

The optimal solutions of Step 2 and Step 3 are given by the Lagrangian multiplier method.

$$u_{ik} = \frac{\left(\frac{1}{d_{ik}}\right)^{\frac{1}{m-1}}}{\sum_{j=1}^{c}\left(\frac{1}{d_{jk}}\right)^{\frac{1}{m-1}}} \tag{4}$$

$$v_i = \frac{\sum_{k=1}^{n}(u_{ik})^m x_k}{\sum_{k=1}^{n}(u_{ik})^m} \tag{5}$$

Note that eq.(4) excludes the case when $d_{ik} = 0$ holds. If this is the case, then $u_{ik} = 1$ and $u_{jk} = 0$ $(\forall j \neq i)$.

In Step 4 we judge that the solution is convergent when U or V is unchanged.

2.2 Variables for Controlling Cluster Sizes

SFCM with crisp reallocation by the maximum membership rule may fail to divide accurately if there are unbalanced clusters like those in Fig.1 in Section 4. In the case of Fig.1, even if each centroids are at center of each circle, about 4.0 percent area of the left side of larger cluster must be assigned to the smaller one when crisp reallocation rule is applied. Therefore, three methods using variables for controlling cluster sizes have been proposed [3,4,5] for tackling such a problem.

The objective functions proposed in [3],[4] and [5], respectively, are as follows,

$$J_{fcma} = \sum_{i=1}^{c}\sum_{k=1}^{n}(\alpha_i)^{1-m}(u_{ik})^m d_{ik} \tag{6}$$

$$J_{pfcm} = \sum_{i=1}^{c}\sum_{k=1}^{n}(u_{ik})^m\{d_{ik} - \lambda\log(\alpha_i)\} \tag{7}$$

$$J_{efca} = \sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}\left\{d_{ik} + \lambda\log\left(\frac{u_{ik}}{\alpha_i}\right)\right\}, \tag{8}$$

where $A = (\alpha_1, \ldots, \alpha_c)$ is a variable for controlling cluster sizes, and λ is a positive parameter. The constraint for A is

$$\mathcal{A} = \left\{A = (\alpha_1, \ldots, \alpha_i) : \sum_{j=1}^{c}\alpha_j = 1; \alpha \geq 0, 1 \leq i \leq c\right\}. \tag{9}$$

Let us denote these three algorithms using the above objective functions as FCMA, PFCM and EFCA respectively. J_{fcma} has three variables U, V, and A, hence the following algorithm with three steps should be used.

Step1. Generate c initial values for V and A.
Step2. Calculate optimal U that minimizes J_{fcma} .
Step3. Calculate optimal V that minimizes J_{fcma} .
Step4. Calculate optimal A that minimizes J_{fcma} .
Step5. If (U, V, A) is convergent, stop; else return to Step2. .

PFCM and EFCA also use the same algorithm. The optimal solutions of each steps can be computed by the Lagrangian multiplier method.

Solutions for J_{fcma}

$$u_{ik} = \frac{\alpha_i \left(\frac{1}{d_{ik}}\right)^{\frac{1}{m-1}}}{\sum_{j=1}^{c} \alpha_j \left(\frac{1}{d_{jk}}\right)^{\frac{1}{m-1}}} \tag{10}$$

$$v_i = \frac{\sum_{k=1}^{n} (u_{ik})^m x_k}{\sum_{k=1}^{n} (u_{ik})^m} \tag{11}$$

$$\alpha_i = \frac{(\sum_{k=1}^{n} (u_{ik})^m d_{ik})^{\frac{1}{m}}}{\sum_{i=1}^{c} (\sum_{k=1}^{n} (u_{ik})^m d_{ik})^{\frac{1}{m}}} \tag{12}$$

Solutions for J_{pfcm}

$$u_{ik} = \frac{\left(\frac{1}{d_{ik} - \lambda \log \alpha_i}\right)^{\frac{1}{m-1}}}{\sum_{j=1}^{c} (\frac{1}{d_{jk} - \lambda \log \alpha_j})^{\frac{1}{m-1}}} \tag{13}$$

$$v_i = \frac{\sum_{k=1}^{n} (u_{ik})^m x_k}{\sum_{k=1}^{n} (u_{ik})^m} \tag{14}$$

$$\alpha_i = \frac{\sum_{k=1}^{n} (u_{ik})^m}{\sum_{i=1}^{c} \sum_{k=1}^{n} (u_{ik})^m} \tag{15}$$

Solutions for J_{efca}

$$u_{ik} = \frac{\alpha_i \exp(-\frac{d_{ik}}{\lambda})}{\sum_{j=1}^{c} \alpha_j \exp(-\frac{d_{jk}}{\lambda})} \tag{16}$$

$$v_i = \frac{\sum_{k=1}^{n} u_{ik} x_k}{\sum_{k=1}^{n} u_{ik}} \tag{17}$$

$$\alpha_i = \frac{\sum_{k=1}^{n} u_{ik}}{n} \tag{18}$$

3 Classifier Function

After finishing clustering, we are able to set a value of membership to a new object by classifier function. In the case of SFCM, the following is considered [6].

$$U_i^s(x) = \frac{\left(\frac{1}{d(v_i, x)}\right)^{\frac{1}{m-1}}}{\sum_{j=1}^{c} \left(\frac{1}{d(v_j, x)}\right)^{\frac{1}{m-1}}}. \tag{19}$$

This function is simply derived from the optimal solution of u_{ik}, where v_i ($i = 1, \ldots, c$) are the converged centroids. A classifier function helps us to consider the theoretical properties of clustering because it is defined in the whole space.

We can convert the result of fuzzy clustering to crisp clusters by regarding an object having the maximum value of membership to cluster i as a member of cluster i.

Now, a region of cluster i in SFCM is represented as the following.

$$U_i^s(x) > U_j^s(x) \tag{20}$$

$$\Leftrightarrow \left(\frac{1}{d(v_i,x)}\right)^{\frac{1}{m-1}} > \left(\frac{1}{d(v_j,x)}\right)^{\frac{1}{m-1}} \tag{21}$$

$$\Leftrightarrow d(v_i,x) < d(v_j,x) \tag{22}$$

Hence, the region of cluster i is

$$R_i = \{x \in \mathcal{R}^p \ : \ d(v_i,x) < d(v_j,x), \ j \neq i\} \tag{23}$$

It shows that the result of SFCM makes the Voronoi regions whose representative point is v_i. Now, as x approaches infinity in a region of cluster i, we obtain

$$\lim_{\|x\|\to\infty} U_i^s(x) = \frac{1}{c}. \tag{24}$$

In this way, we make characteristics of method clear by analyzing its classifier function. The classifier function of three methods using variables controlling size of clusters is the following.

$$U_i^a(x) = \frac{\alpha_i \left(\frac{1}{d(v_i,x)}\right)^{\frac{1}{m-1}}}{\sum_{j=1}^{c} \alpha_j \left(\frac{1}{d(v_j,x)}\right)^{\frac{1}{m-1}}} \tag{25}$$

$$U_i^p(x) = \frac{\left(\frac{1}{d(v_i,x)-\lambda\log\alpha_i}\right)^{\frac{1}{m-1}}}{\sum_{j=1}^{c} \left(\frac{1}{d(v_j,x)-\lambda\log\alpha_j}\right)^{\frac{1}{m-1}}} \tag{26}$$

$$U_i^e(x) = \frac{\alpha_i \exp\left(-\frac{d(v_i,x)}{\lambda}\right)}{\sum_{j=1}^{c} \alpha_j \exp\left(-\frac{d(v_j,x)}{\lambda}\right)} \tag{27}$$

The next propositions show theoretical properties of these classifier functions.

Proposition 1. *As x approaches infinity in an unbounded region R_i, then*

$$\lim_{\|x\|\to\infty} U_i^a(x) = \alpha_i \tag{28}$$

$$\lim_{\|x\|\to\infty} U_i^p(x) = \frac{1}{c} \tag{29}$$

$$\lim_{\|x\|\to\infty} U_i^e(x) = 1 \tag{30}$$

is obtained.

These can be confirmed visually by Fig.2 in Section 4.

Proposition 2. *As x approaches v_i, u_{ik} approaches unity in FCMA, however it doesn't approach unity in PFCM or EFCA, namely,*

$$\lim_{x \to v_i} U_i^a(x) = 1 \tag{31}$$

$$\lim_{x \to v_i} U_i^p(x) = \frac{1}{1 + C_p} < 1 \tag{32}$$

$$\lim_{x \to v_i} U_i^e(x) = \frac{1}{1 + C_e} < 1, \tag{33}$$

where

$$C_p = \sum_{j=1, j \neq i}^{c} \left(\frac{\lambda \log \alpha_i}{d(v_j, x) - \lambda \log \alpha_j} \right)^{\frac{1}{m-1}} \tag{34}$$

$$C_e = \alpha_i^{-1} \sum_{j=1, j \neq i}^{c} \alpha_j \exp\left(-\frac{d(v_j, x)}{\lambda} \right). \tag{35}$$

The proofs of Proposition 1 and 2 are obvious and thus the detail is omitted.

Proposition 3. *The region of cluster i is multiplicatively weighted Voronoi region[7] in FCMA, and locally additively weighted Voronoi in EFCA and PFCM. Each representative point of the regions is v_i $(i = 1, \ldots, c)$. Multiplicatively weighted Voronoi region i is defined as*

$$R_i = \left\{ x \in \mathcal{R}^p \ : \ \frac{d(v_i, x)}{w_i} < \frac{d(v_j, x)}{w_j}, \ j \neq i \right\}, \tag{36}$$

and additively weighted Voronoi region i is defined as

$$R_i = \{ x \in \mathcal{R}^p \ : \ d(v_i, x) - w_i < d(v_j, x) - w_j, \ j \neq i \}, \tag{37}$$

where $w_i > 0$ $(i = 1, \ldots, c)$ are weights of the region i.

Proof. Each boundary between cluster i and cluster j given by $U_i(x) = U_j(x)$ is as follows.

FCMA

$$\begin{aligned}
& U_i^a(x) = U_j^a(x) \\
\Leftrightarrow\ & \alpha_i \left(\frac{1}{d(v_i, x)} \right)^{\frac{1}{m-1}} = \alpha_j \left(\frac{1}{d(v_j, x)} \right)^{\frac{1}{m-1}} \\
\Leftrightarrow\ & \alpha_i^{m-1} \frac{1}{d(v_i, x)} = \alpha_j^{m-1} \frac{1}{d(v_j, x)} \\
\Leftrightarrow\ & \frac{d(v_j, x)}{\alpha_i^{m-1}} = \frac{d(v_j, x)}{\alpha_j^{m-1}}
\end{aligned} \tag{38}$$

PFCM

$$\begin{aligned}
& U_i^p(x) = U_j^p(x) \\
\Leftrightarrow & \left(\frac{1}{d(v_i, x) - \lambda \log \alpha_i}\right)^{\frac{1}{m-1}} = \left(\frac{1}{d(v_j, x) - \lambda \log \alpha_j}\right)^{\frac{1}{m-1}} \\
\Leftrightarrow & d(v_i, x) - \lambda \log \alpha_i = d(v_j, x) - \lambda \log \alpha_j \\
\Leftrightarrow & d(v_i, x) - \lambda \log \frac{1}{\alpha_j} = d(v_j, x) - \lambda \log \frac{1}{\alpha_i}
\end{aligned} \tag{39}$$

EFCA

$$\begin{aligned}
& U_i^e(x) = U_j^e(x) \\
\Leftrightarrow & \alpha_i \exp\left(-\frac{d(v_i, x)}{\lambda}\right) = \alpha_j \exp\left(-\frac{d(v_j, x)}{\lambda}\right) \\
\Leftrightarrow & \log \alpha_i - \frac{d(v_i, x)}{\lambda} = \log \alpha_j - \frac{d(v_j, x)}{\lambda} \\
\Leftrightarrow & d(v_i, x) - \lambda \log \frac{1}{\alpha_j} = d(v_j, x) - \lambda \log \frac{1}{\alpha_i}
\end{aligned} \tag{40}$$

The above indicates that FCMA makes multiplicatively weighted Voronoi region with weights α_i^{m-1} , while PFCM and EFCM make locally additively weighted Voronoi region with weights $\lambda \log(1/\alpha_i)$ (for cluster $j \neq i$). 'Locally' means that a weight of a region is dependent on a pair of clusters, in other words, the weight of a region between region i and j is different from the weight of the region considering between regions i and k.

Note that these propositions imply that the region of cluster i ($i = \arg\max_i \alpha_i$) is infinite while the region of cluster j ($j = 1, \ldots, c, j \neq i$) is finite in FCMA. Additionally, the boundary is locally linear (hyper-plane) when the dissimilarity function d is defined as the squared Euclidean norm, while boundary is locally hyperbolic when d is defined as the Euclidean norm in PFCM or EFCM.

4 Numerical Examples

The purpose of this paper is to give theoretical properties of methods with variables for controlling cluster sizes, hence we show only the result of simple illustrative examples in this section, and omit the result of applying to real examples.

4.1 First Data Set

Figure 1 is an artificially generated data set with two groups: one has 20 objects randomly in a circle with the radius of 1.0 , the other has 180 objects randomly

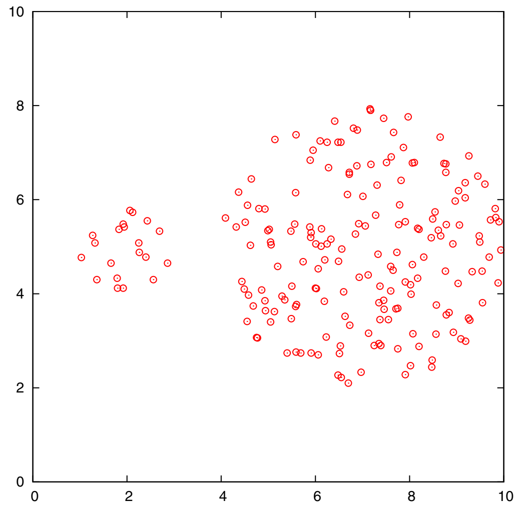

Fig. 1. Artificially generated data set with two groups: one has 20 objects in circle with the radius of 1.0, the other has 180 objects in circle with the radius of 3.0 and the distance between the centers of two circles is 5.0

in a circle with the radius of 3.0 and the distance between the centers of two circles is 5.0.

Figure 2 shows the results of clustering the data set as shown in Fig.1 ($c = 2$) with SFCM, FCMA, PFCM and EFCA, respectively, and with $\lambda = 5.0, m = 1.6$. In the figure, the objects of two clusters are displayed in small squares or small circles, and the centroids are cross marks. The contours denote the membership value, and increment is 0.1. Solid line in the contours, which shows the membership value is 0.5, indicates the boundary between two clusters. This data set has two clusters, which are small and large. SFCM makes a Voronoi diagram when the maximum membership rule is applied, thus a part of large cluster is misclassified as a part of smaller cluster as shown in Fig.2**(a)** while three methods consider these cluster sizes and succeed in having good clusters as shown in Fig.2**(b)-(d)**.

Centroid Inside and Outside of Its Region. PFCM and EFCA represent cluster sizes by additive weights, while FCMA represents them by multiplicative weights, whereby PFCM and EFCA may output an odd result: there is no centroid in its region. Such a result is shown when $d(v_i, v_j) < |\lambda \log(\alpha_i/\alpha_j)|$. Figure 3 shows the results of clustering data in Fig.1 by PFCM and EFCA,

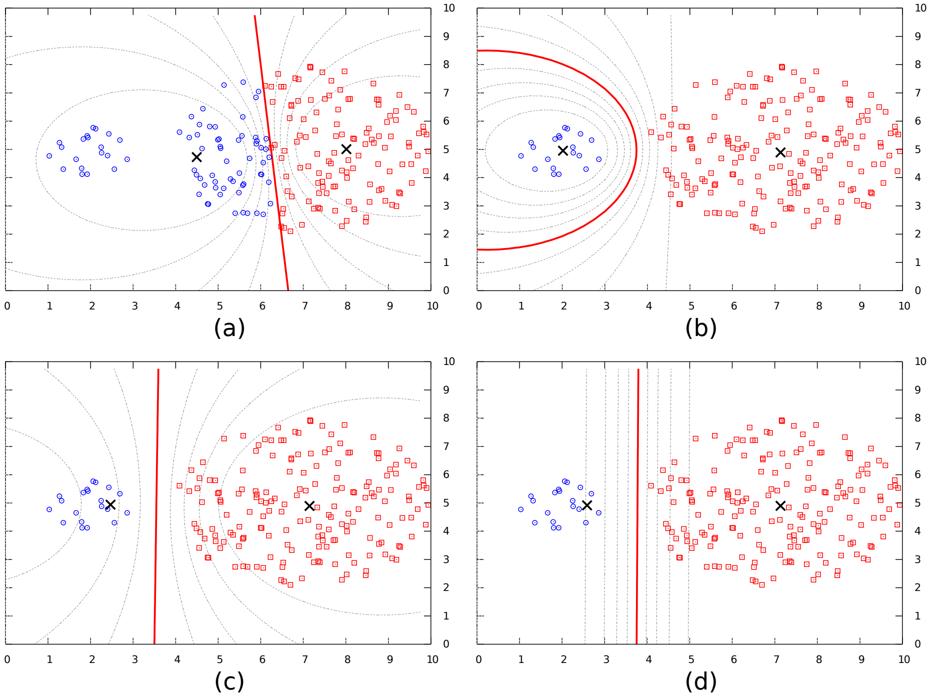

Fig. 2. Clusters when **(a)**SFCM, **(b)**FCMA, **(c)**PFCM and **(d)**EFCA were applied to the data set shown as Fig.1. A part of larger cluster is misclassified as a part of larger cluster in SFCM while FCMA, PFCM and EFCA succeed in having good clusters.

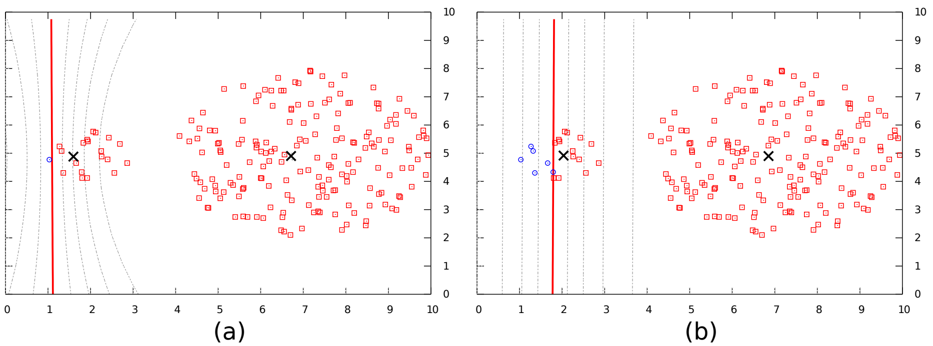

Fig. 3. Clusters when **(a)**PFCM and **(b)**EFCM with too large λ were applied to the data set shown as Fig.1. The centroid of smaller cluster is out of its region.

where $\lambda = 7.75$ and $\lambda = 8.20$ respectively. In these case, $|\lambda \log(\alpha_i/\alpha_j)| = 31.29$, $d(v_i, v_j) = 26.18$ in Fig.3**(a)**, and $|\lambda \log(\alpha_i/\alpha_j)| = 25.54$, $d(v_i, v_j) = 23.14$ in Fig.3**(b)**, therefore the centroid of a smaller cluster is in the region of a larger cluster. Note that FCMA doesn't output such results because multiplicative weights are used, however it is not flexible since it has only one parameter m.

4.2 Second Data Set

Figure 4 shows an artificially generated data set with three groups: one has 180 objects randomly in a circle with the radius of 3.0 , another has 80 objects randomly in a circle with the radius of 2.0 and the other has 20 objects randomly in a circle with the radius of 1.0.

Figure 5 shows the results of clustering the data set in Fig.4 ($c = 3$). The contours denote the membership value of the largest cluster. This data set has three clusters, which are small, medium and large. In this case, no matter what value of m or initial V, SFCM fails in a good classification: a centroid in small group side is pulled by larger cluster since SFCM tries to make clusters equally as shown in Fig.5**(a)**. On the other hand, the three methods are able to succeed as shown in Fig.5**(b)-(d)**. These results indicate that these methods may work well when there are three ore more clusters.

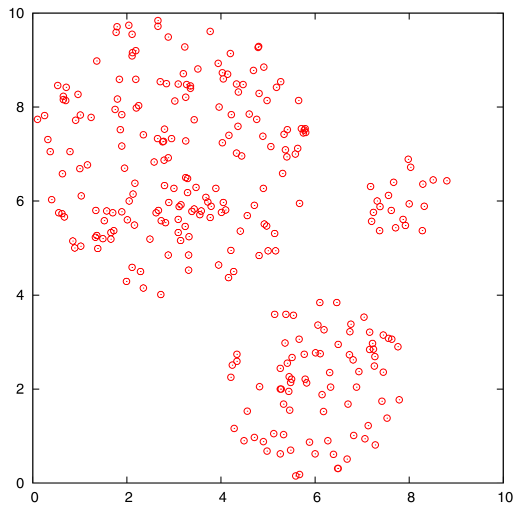

Fig. 4. Artificially generated data set with three groups: one has 180 objects in circle with the radius of 3.0, another has 80 objects in circle with the radius of 2 and the other has 20 objects in circle with the radius of 1.0

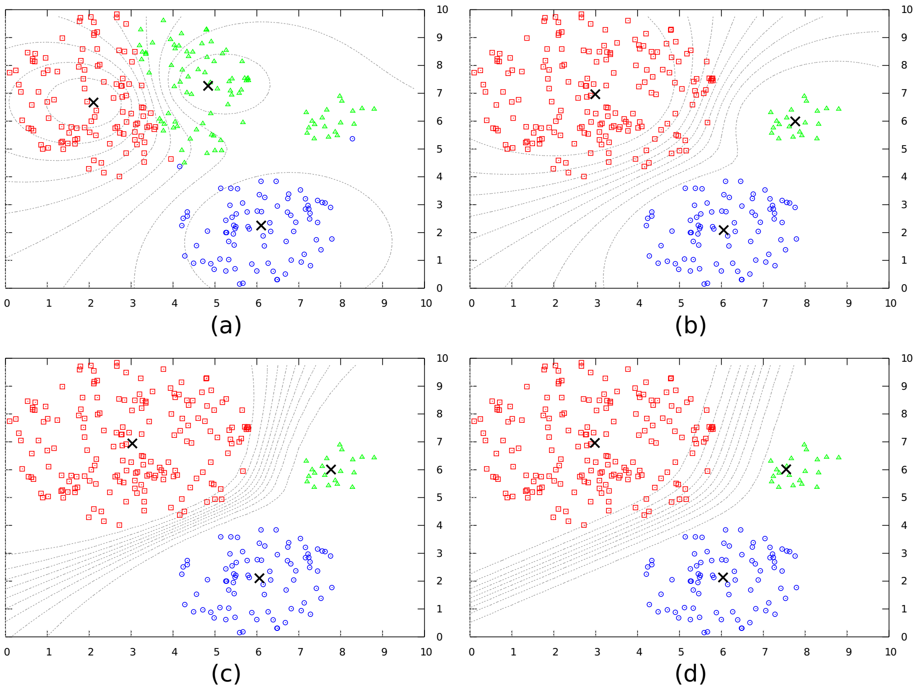

Fig. 5. Clusters when **(a)**SFCM, **(b)**FCMA, **(c)**PFCM and **(d)**EFCA were applied to the data set shown as Fig.4. The centroid of small cluster side is pulled by large cluster in SFCM while FCMA, PFCM and EFCA succeed in having good clusters.

5 Conclusion

In this paper, we described three methods with variables for controlling cluster sizes, and showed their theoretical properties using their classifier functions. Furthermore we applied these methods to illustrative examples and showed that these methods worked well. Each of the methods outputted different results though all of these methods were able to handle the cluster sizes.

From a practical viewpoint, the terms of covariance variables within clusters should also be used [6,5] with appropriate parameters. However we omitted discussion of this topic in this paper for simplicity. Besides, there are rooms for further discussion of the combination of the kernel method or the addition of constraints in semi-supervised clustering and comparison with other approaches, for example conditional FCM [8,9], whose constraint of membership is continuously updated during clustering, for our future works.

Acknowledgment. This work has partly been supported by the Grant-in-Aid for Scientific Research, Japan Society for the Promotion of Science, No. 23500269.

References

1. Dunn, J.C.: A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. Cybernetics and Systems 3(3), 32–57 (1973)
2. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algorithms. Kluwer Academic Publishers, Norwell (1981)
3. Miyamoto, S., Kurosawa, N.: Controlling cluster volume sizes in fuzzy c-means clustering. In: SCIS and ISIS, Yokohama, Japan, pp. 1–4 (September 2004)
4. Yang, M.S.: On a class of fuzzy classification maximum likelihood procedures. Fuzzy Sets Syst. 57(3), 365–375 (1993)
5. Ichihashi, H., Honda, K., Tani, N.: Gaussian mixture pdf approximation and fuzzy c-means clustering with entropy regularization. In: Proc. of the 4th Asian Fuzzy System Symposium, pp. 217–221 (2000)
6. Miyamoto, S., Ichihashi, H., Honda, K.: Algorithms for fuzzy clustering. Springer, Heidelberg (2008)
7. Boots, B.N.: Weighting thiessen polygons. Economic Geography, 248–259 (1980)
8. Noordam, J., Van Den Broek, W., Buydens, L.: Multivariate image segmentation with cluster size insensitive fuzzy c-means. Chemometrics and Intelligent Laboratory Systems 64(1), 65–78 (2002)
9. Lai, Y., Huang, P., Lin, P.: An integrity-based fuzzy c-means method resolving cluster size sensitivity problem. In: 2010 International Conference on Machine Learning and Cybernetics (ICMLC), vol. 5, pp. 2712–2717. IEEE (2010)