Soft Comput (2008) 12:593–611
DOI 10.1007/s00500-007-0231-6

ORIGINAL PAPER

An enhanced possibilistic C-Means clustering algorithm EPCM

Zhenping Xie · Shitong Wang · F. L. Chung

Published online: 14 August 2007

Abstract The possibility based clustering algorithm PCM was first proposed by Krishnapuram and Keller to overcome the noise sensitivity of algorithm FCM (Fuzzy C-Means). However, PCM still suffers from the following weaknesses: (1) the clustering results are strongly dependent on parameter selection and/or initialization; (2) the clustering accuracy is often deteriorated due to its coincident clustering problem; (3) outliers can not be well labeled, which will weaken its clustering performances in real applications. In this study, in order to effectively avoid the above weaknesses, a novel enhanced PCM version (EPCM) is presented. Here, at first a novel strategy of flexible hyperspheric dichotomy is proposed which may partition a dataset into two parts: the main cluster and auxiliary cluster, and is then utilized to construct the objective function of EPCM with some novel constraints. Finally, EPCM is realized by using an alternative optimization approach. The main advantage of EPCM lies in the fact that it can not only avoid the coincident cluster problem by using the novel constraint in its objective function, but also has less noise sensitivity and higher clustering accuracy due to the introduction of the strategy of flexible hyperspheric dichotomy. Our experimental results about simulated and real datasets confirm the above conclusions.

Keywords Enhanced possibilistic c-Means clustering (EPCM) · Flexible hyperspheric dichotomy · Outliers · Image segmentation

Z. Xie · S. Wang (✉)
School of Information, Southern Yangtze University, Wuxi, China
e-mail: wxwangst@yahoo.com.cn

S. Wang · F. L. Chung
Department of computing, Hong Kong Polytechnic University, Hong Kong, China

1 Introduction

As an important data processing technique, clustering has been widely utilized in a variety of fields, such as data mining, pattern recognition, image processing and so on. By clustering the objects may be partitioned into different subgroups and in the same subgroup the objects are as compact as possible while the objects in different subgroups are as disperse as possible. Generally, clustering algorithms may be classified into the following categories: hierarchical clustering, partition-based clustering, density-based clustering and grid-based clustering (Zhang and Pal 2002, Chung et al. 2006, Jian 2005, Bezdek 1981, Sato et al. 1997, Dave and Krishnapuram 1997, Dave and Sen 2002, Wang et al. 2005, 2006, Deng and Wang 2005). Especially, the partition-based clustering algorithms which partition the objects with some membership matrices (fuzzy matrix) are most widely and deeply studied.

One of the most widely used fuzzy clustering algorithms is FCM (Fuzzy C-Means). FCM assigns the fuzzy memberships of data points (objects) to the clusters. In FCM, the fuzzy memberships of data points only represent the relative degrees of data points belonging to their clusters. So the fuzzy memberships of FCM cannot always represent the proper degrees of data points belonging to their clusters, especially in noise environment Krishnapuram and Keller (1993). To overcome this weakness, Krishnapuram and Keller (1993, 1996) proposed a new clustering algorithm named PCM (Possibilistic C-Means). PCM relaxes the column sum constraint of fuzzy membership matrix in FCM and introduces a possibilistic partition matrix, so that possibilistic memberships may reflect the typicalities of data points to their clusters well. Compared with FCM, PCM can effectively eliminate the influence of noise and outliers on clustering results. However, firstly the price PCM pays for its freedom to ignore noisy points is that PCM is very sensitive to initializations, and often results in the coincident cluster problem (Zhang and Yeung 2004, Barni et al. 1996). Secondly, typicalities, i.e., possibilistic memberships, are very sensitive to the choices of the additional parameters of PCM, which directly decide the clustering accuracy. Finally, the outliers cannot be labeled accurately which will reduce the clustering performance in real applications.

Nowadays, several improved PCM algorithms have been proposed to overcome the weaknesses of the original PCM algorithm (Gustafson and Kessel 1979, Timm et al. 2001, Timm and Kruse 2002, Timm et al. 2004, Zhang and Yeung 2004, Yang and Wu 2006, Pal et al. 2005). In Timm et al. (2001, 2004), Timm and Kruse (2002), proposed two possibilistic fuzzy clustering algorithms that can avoid the coincident cluster problem of PCM by adding an inverse function of the distances between cluster centers in PCM's objective function, which acts as a repulsive force and keeps the clusters separate (avoids coincident clusters). In Timm and Kruse (2002) and Timm et al. (2004), used the same concept to modify the objective function as used in Gustafson and Kessel's clustering algorithm Gustafson and Kessel (1979). In Pal et al. (2005), combined the objective functions of PCM and FCM into a new objective function and presented an improved version, called PFCM, which can be interpreted as PCM and FCM, respectively, in some special cases where some proper parameters are adopted. So, PFCM can inherit the merits of both clustering algorithms. In Zhang and Yeung (2004), introduced fuzzy membership of FCM into PCM's objective function and presented an improved PCM clustering algorithm to overcome the coincident cluster problem of PCM. In Barni et al. (1996), Yang et al. presented an unsupervised possibilistic clustering algorithm PCA and proved it to be more robust than PCM. Although these improved PCM clustering algorithms can partially overcome the drawbacks of PCM, particular attentions must be often paid to adjust some parameters and are not easy for real applications. Furthermore, they still have no ability to label the outliers accurately.

In this paper, we propose a novel improved PCM clustering algorithm called EPCM (Enhanced PCM). EPCM has the following two main features:

(1) It introduces the strategy of flexible hyperspheric dichotomy to avoid estimating the parameter η_i in PCM and its variants, which has an important influence on the clustering results In our method, two new variables are introduced to play similar roles as parameter η_i, while they can be adaptively updated with some learning rules such that the new clustering algorithm has fewer parameters and can easily be used in real applications. Moreover, this strategy will help EPCM label the outliers accurately and improve the clustering accuracy.

(2) It imposes a novel constraint on the objective function. Unlike other improved PCM algorithms Zhang and Yeung (2004), Yang and Wu (2006), Pal et al. (2005), which often introduce the constraints of FCM's objective function into their objective functions, our new algorithm introduces a novel constraint. In the adopted objective functions in Zhang and Yeung (2004), Pal et al. (2005), there are two partition matrices that are used to aim to avoid the coincident cluster problem. These two partition matrices play the roles of the fuzzy and possibilistic partitions, respectively. However, only a partition matrix is needed in our new objective function.

In summary, the proposed EPCM not only inherits the merits of PCM, but also weakens the coincident cluster and parameter sensitivity problem of PCM. Especially, EPCM has no any parameter needed to be adjusted by hand and can effectively label outliers of a dataset, which will result in higher clustering performance. So it is more available for real applications. Our experimental results demonstrate the above advantages of EPCM. The rest of this paper is organized as follows. Both FCM and PCM are briefly introduced and discussed in Sect. 2; Then, the detailed descriptions of EPCM is presented in Sect. 3; In Sect. 4, the clustering performances of EPCM on several experiments are reported by comparing it with several typical possibilistic clustering algorithms; Sect. 5 concludes this paper.

2 FCM and PCM algorithm

The most widely used fuzzy clustering algorithm is FCM (Fuzzy C-Means). Its objective function can be described as follows.

$$J_{FCM}(\mathbf{U},\mathbf{V};X)=\sum_{i=1}^{C}\sum_{j=1}^{N}u_{ij}^{m}*d^2(\mathbf{x}_j,\mathbf{v}_i)$$
$$0\le u_{ij}\le 1,\quad i=1,2,\ldots,C;\ j=1,2,\ldots,N$$
$$\sum_{i=1}^{C}u_{ij}=1,\quad j=1,2,\ldots,N \tag{1}$$
$$\sum_{j=1}^{N}u_{ij}>0,\quad i=1,2,\ldots,C$$

where $\mathbf{U}=[u_{ij}]_{C\times N}$ denotes the fuzzy partition matrix, u_{ij} denotes the fuzzy membership, $\mathbf{V}=\left[\mathbf{v}_1\ \mathbf{v}_2\ldots\mathbf{v}_C\right]$ denotes C cluster centers (prototypes), $X=\{\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_N\}$ denotes the dataset, $d(\mathbf{x}_j,\mathbf{v}_i)$ denotes the distance measure, e.g., the most commonly used Euclidean distance. Optimal partitions $\mathbf{U}^*$ of X are taken from pairs $(\mathbf{U}^*,\mathbf{V}^*)$ that are local minimum of J_{FCM}. In FCM, u_{ij}, $\mathbf{v}_i$ can be updated using the following (2):

$$u_{ij}=\left(\sum_{k=1}^{C}\left(\frac{d_{ij}}{d_{kj}}\right)^{2/(m-1)}\right)^{-1},$$
$$i=1,2,\ldots,C;\ j=1,2,\ldots,N \quad (2)$$
$$\mathbf{v}_i=\frac{\sum_{j=1}^{N}u_{ij}^{m}\mathbf{x}_j}{\sum_{j=1}^{N}u_{ij}^{m}},\quad i=1,2,\ldots,C$$

In FCM, the fuzzy membership of data point $\mathbf{x}_j$ is inversely proportional to the distance between $\mathbf{x}_j$ and the cluster center $\mathbf{v}_i$, and the sum of the memberships of $\mathbf{x}_j$ to all cluster centers is 1. Therefore, if data point $\mathbf{x}_j$ keeps the equal distance from two cluster centers, the membership of this data point in each cluster will be the same. This situation will give rise to the following problem: Given the cluster number $C = 2$, if noisy point $\mathbf{x}_j$ in the dataset is far but equidistant from two cluster centers, then FCM will give the same membership, i.e. 0.5, to each cluster. Obviously it seems far more natural that such a point should be given very low memberships belonging to these two clusters. To overcome this problem, Krishnapuram and Keller (1993, 1996) proposed an improved clustering algorithm called PCM (Possibilistic C-Means). The objective function of PCM reads

$$J_{PCM}(\mathbf{U},\mathbf{V};X)=\sum_{i=1}^{C}\sum_{j=1}^{N}u_{ij}^{m}d^{2}(\mathbf{x}_j,\mathbf{v}_i)+\sum_{i=1}^{C}\eta_i\sum_{j=1}^{N}(1-u_{ij})^{m} \quad (3)$$
$$0\le u_{ij}\le 1,\quad i=1,2,\ldots,C,\ j=1,2,\ldots,N$$
$$\sum_{j=1}^{N}u_{ij}>0,\quad i=1,2,\ldots C$$

where $\mathbf{U}=[u_{ij}]_{C\times N}$ denotes possibilistic partition matrix, u_{ij} denotes the possibilistic membership. u_{ij}, $\mathbf{v}_i$ can be updated using the following (4):

$$u_{ij}=\left(1+\left(\frac{d_{ij}^{2}}{\eta_i}\right)^{1/(m-1)}\right)^{-1},$$
$$i=1,2,\ldots,C;\ j=1,2,\ldots,N \quad (4)$$
$$\mathbf{v}_i=\frac{\sum_{j=1}^{N}u_{ij}^{m}\mathbf{x}_j}{\sum_{j=1}^{N}u_{ij}^{m}},\quad i=1,2,\ldots,C$$

where $\eta_i(i = 1,2,\ldots,C)$ is a scale parameter and in Krishnapuram and Keller (1996) it is suggested to be:

$$\eta i=K\frac{\sum_{j=1}^{N}u_{ij}^{m}*d^{2}(\mathbf{x}_j,\mathbf{v}_i)}{\sum_{j=1}^{N}u_{ij}^{m}} \quad (5)$$

where $K > 0$ and in general $K = 1$.

PCM relaxes the column sum constraint of the membership matrix in FCM, so that the sum of each column of PCM partition matrix satisfies the looser constraint. In other words, each element of one column of PCM partition matrix might be any number between 0 and 1, as long as at least one of them is positive. The element value of PCM partition matrix is often interpreted as the *typicality* of a data point associated with one cluster rather than its relative membership to the cluster in FCM. The advantage of PCM compared with FCM is its capability in identifying outliers in dataset and weakening the influence of outliers and noise on clustering results. However, as pointed out by Barni et al. (1996), the price PCM pays for its freedom to ignore noisy points is that PCM is very sensitive to initializations, and it sometimes generates the coincident clusters. Moreover, the typicalities, i.e. possibilistic memberships, are very sensitive to the choices of the additional parameters η_i needed by PCM. As analyzed in Pal et al. (2005), the above weakness of PCM are derived from the facts: (1) The relaxed column sum constraint of PCM makes different clusters be independent, so that PCM tends to generate identical clusters, i.e., the coincident cluster problem. (2) The strong sensitivity of possibilistic memberships to parameters η_i makes PCM very brittle to initialization and index parameter m. So these two aspects of PCM deserve further studying.

3 Enhanced possibilistic clustering algorithm EPCM

3.1 The strategy of hyperspheric dichotomy HD

In this subsection, we first present a strategy called hyperspheric dichotomy (HD), which can be utilized to partition a dataset into the main part and auxiliary part. Then in next subsection we further explore its adaptive version.

The strategy of hyperspheric dichotomy is based on the following objective function:

$$J_{HD}(\mathbf{U}_I,\mathbf{U}_O,\mathbf{v}_I;X)=\sum_{j=1}^{N}u_{I,j}^{m}d_{I,j}^{2}(\mathbf{v}_I,\mathbf{x}_j)+\sum_{j=1}^{N}u_{O,j}^{m}R^{2} \quad (6)$$
$$u_{I,j}+u_{O,j}=1$$

where $\mathbf{U}_I = [u_{I,j}]_{1\times N}$ and $\mathbf{U}_O = [u_{O,j}]_{1\times N}$ denote the hypersphere dichotomy matrices, whose elements represent the memberships of data point $\mathbf{x}_j$ to the main part and auxiliary part of data set X, respectively. $\mathbf{v}_I$ denotes the cluster center of the main part; parameter R is a constant which denotes the radius of the hypersphere. Optimal partitions $\mathbf{U}_I^*$ of X are taken from $(\mathbf{U}_I^*, \mathbf{U}_O^*, \mathbf{v}_I^*)$ that are local minimum of J_{HD}. If $u_{I,j}^* > 0.5$, then $\mathbf{x}_j$ belongs to the main part (cluster), otherwise, $u_{O,j}^* > 0.5$, $\mathbf{x}_j$ belongs to the auxiliary part (cluster). u_{ij}, $\mathbf{v}_I$ can be updated using the following (7) and (8):

$$\mathbf{v}_I = \frac{\sum_{j=1}^{N} u_{I,j}^m \mathbf{x}_j}{\sum_{j=1}^{N} u_{I,j}^m} \tag{7}$$

$$u_{I,j} = \frac{d_j^{-\frac{2}{m-1}}}{d_j^{-\frac{2}{m-1}} + R^{-\frac{2}{m-1}}}, u_{O,j} = 1 - u_{I,j} \tag{8}$$

where $d_{I,j} = d(\mathbf{x}_j, \mathbf{v}_I)$. Similar to FCM, we can easily derive the corresponding HD based clustering algorithm using (7) and (8).

For a dataset, we can partition it into the main part (cluster) and the auxiliary part (cluster) by the above strategy. So we may call the dichotomy procedure as hypersphere dichotomy clustering (HDC). Now, let us discuss the relationship between HDC and several existing clustering algorithms and then reveal the interesting phenomenon why these algorithms are sensitive to parameters. Here, we list two objective functions associated with the clustering algorithms NC Dave and Krishnapuram (1997) and PCM Krishnapuram and Keller (1993).

$$J_{NC} = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij}^m d_{ij}^2 + \sum_{j=1}^{N} \delta^2 \left(1 - \sum_{i=1}^{C} u_{ij}\right)^m \tag{9}$$

$$J_{PCM} = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij}^m d_{ij}^2 + \sum_{i=1}^{C} \sum_{j=1}^{N} \eta_i (1 - u_{ij})^m \tag{10}$$

From (6), (9) and (10), we can easily find that if set $C = 1$ and set $\delta^2 = R^2, \eta_i = R^2$ respectively, then (9) and (10) are equivalent to (1). So the parameters δ, η_i and R in different objective functions have similar influences on clustering results. From (8) it is easy to find that R is the partition boundary of a dataset. If $d_j < R$, then $u_{I,j} > 0.5$, and $\mathbf{x}_j$ is within the hypersphere and belongs to the main part (cluster) of the dataset, otherwise, $u_{O,j} > 0.5$, and $\mathbf{x}_j$ is beyond the hypersphere and belongs to the auxiliary part (cluster). So R has a serious influence on clustering results. Similarly, the influences of parameters δ, η_i on the clustering results of both NC and PCM can be revealed in a similar way.

From the above analysis we can see that it is not wise to set some parameters such as δ, η_i and R before clustering. A proper way should assume that we can obtain these parameters adaptively, i.e., these parameters may be automatically updated in the clustering approach. In the next subsection, we will present an improved strategy of hyperspheric dichotomy, called FHD (flexible HD), to achieve this goal.

3.2 The strategy of flexible hyperspheric dichotomy FHD

In order to avoid parameter R being inappropriately preset, here we introduce a new concept of flexible hyperspheric dichotomy (FHD). The FHD stems from the idea of the optimal hyperplane classifier SVM (Support Vector Machine).

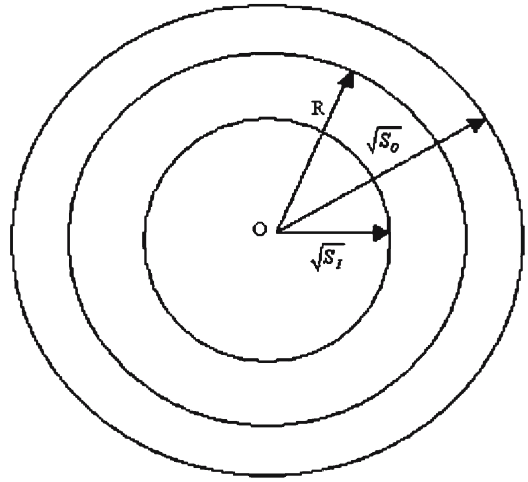

Fig. 1 The conceptual display of FHD

Figure 1 shows the principle of flexible hyperspheric dichotomy. In Fig. 1, three hyperspheres share the same center. The hypersphere with radius R corresponds the hypersphere in Subsect. 3.1 and is taken as a partition boundary. Another two hyperspheres with radius $\sqrt{S_I}$ and $\sqrt{S_O}$, called the inner/outer hyperspheres, respectively, give upper and lower bounds of the partition boundary. Now the data points between the inner and outer hyperspheres could be viewed as the data points near the hyperspheric boundary, which are expected to be very little for good clustering, while the data points within the inner hypersphere and beyond the outer should belong to the main cluster and auxiliary cluster with high memberships, respectively. So we propose a new objective function (11) based on the concept of FHD:

$$J_{FHD} = \sum_{j=1}^{N} u_{I,j}^m d_{I,j}^2 + \sum_{j=1}^{N} u_{O,j}^m S_I + (1-\alpha) \times \sum_{j, d_{I,j}^2 > S_I} u_{I,j}^m (d_{I,j}^2 - S_I) + \alpha \sum_{j, d_{I,j}^2 < S_O} u_{O,j}^m (S_o - d_{I,j}^2) \tag{11}$$

$$u_{I,j} + u_{O,j} = 1$$

where α is a free parameter, with range (0,1), S_I and S_O can be defined as (12) and (13), which may be interpreted as the scatter measures of both main cluster and auxiliary clusters to the hypersphere center.

$$S_I = \frac{\sum_{j=1}^{N} u_{I,j}^m * d_{I,j}^2}{\sum_{j=1}^{N} u_{I,j}^m} \tag{12}$$

$$S_O = \frac{\sum_{j=1}^{N} u_{O,j}^m * d_{I,j}^2}{\sum_{j=1}^{N} u_{O,j}^m} \tag{13}$$

For the above (11), all their terms can be explained as follows:

1. The first term makes all the data points in the main cluster as near its main cluster center as possible.
2. The second term attempts to make all the data points in the auxiliary cluster far from the inner hypersphere.
3. The third term assures $u_{I,j}$ intend to be 0 when data point $\mathbf{x}_j$ is beyond the inner hypersphere.
4. The fourth term assures $u_{O,j}$ intend to be 0 for all the data points in the outer hypersphere.

The optimal partition $\mathbf{U}_I^*$ of X can be obtained from ($\mathbf{U}_I^*$, $\mathbf{U}_O^*$, $\mathbf{v}_I^*$) that is local minimum of J_{FHD} by the following Theorem 1.

Theorem 1 *Given* $\mathbf{v}_I$, S_I *and* S_O, *the necessary conditions of minimizing* (11) *are*:

$$u_{I,j} = \frac{(d_{I,j}^2+(1-\alpha)\cdot SD_{I,j})^{\frac{-1}{m-1}}}{(d_{I,j}^2+(1-\alpha)\cdot SD_{I,j})^{\frac{-1}{m-1}}+(S_I+\alpha\cdot SD_{O,j})^{\frac{-1}{m-1}}} \tag{14}$$

$$u_{O,j} = 1 - u_{I,j} \tag{15}$$

where $SD_{I,j}$ *and* $SD_{O,j}$ *are defined as*

$$SD_{I,i} = St(d_{I,j}^2 - S_I) \tag{16}$$

$$SD_{O,i} = St(S_O - d_{I,j}^2) \tag{17}$$

$$St(x) = \begin{pmatrix} x & x > 0 \\ 0 & x \le 0 \end{pmatrix} \tag{18}$$

Proof When $\mathbf{v}_I$, S_I and S_O are given, we differentiate J_{FHD} with respect to $u_{I,j}$ and $u_{O,j}$, and set these derivatives to zero. Accordingly, (14) and (15) can be easily derived.

According to the above analysis, we give the FHD based clustering algorithm as follows:

FHD based clustering algorithm

1. Given m (in general, 2), initialize the partition matrices $\mathbf{U}_I$, $\mathbf{U}_O$;
2. Compute the main cluster center $\mathbf{v}_I$ using (7);
3. Estimate S_I and S_O using (12) and (13);
4. Update the partition matrices $\mathbf{U}_I$ and $\mathbf{U}_O$ using (14) and (15);
5. If the difference of $\mathbf{U}_I$ or $\mathbf{U}_O$ in successive iterations is less than a small threshold, then termination, otherwise go to step 2.

With the above FHD based clustering algorithm, we can easily get the optimal partition matrices $\mathbf{U}_I$ and $\mathbf{U}_O$. Furthermore, we give the following analysis to demonstrate the feasibility of the proposed algorithm.

Let $u_{I,j} = u_{O,j}$, with (14) and (15) we can obtain

$$d_{I,j}^2 + (1-\alpha)\cdot SD_{I,j} = S_I + \alpha \cdot SD_{O,j} \tag{19}$$

By integrating (16) and (17) together with (19), we get

$$d_{I,j} = \sqrt{S_I + \alpha \cdot (S_O - S_I)/2} \tag{20}$$

So, we can take the radius of the optimal partition hypersphere as

$$R_\alpha = \sqrt{S_I + \alpha \cdot (S_O - S_I)/2} \tag{21}$$

With (21), there exists an obvious relationship between $u_{I,j}$ and $u_{O,j}$:

$$u_{I,j} - u_{O,j} \begin{cases} > 0 & d_{I,j} < R_\alpha \\ = 0 & d_{I,j} = R_\alpha \\ < 0 & d_{I,j} > R_\alpha \end{cases} \tag{22}$$

Based on the above analysis, we can find that the FHD based clustering algorithm can really avoid the issue that R may be inappropriately preset. By introducing S_I and S_O, the proposed algorithm can effectively obtain the optimal radius R_α of the partition hypersphere, where R_α plays the same role as R in the concept of HD, i.e., the clustering strategy of PCM. So FHD can weaken the influence of initialization in this way.

From (21), we may see that the optimal R_α is dependent on parameter α, where $0 \le \alpha \le 1$. The selection of parameter α has some influence on the clustering results of the proposed algorithm. Here, we will first give an intuitive observation for the influence of parameter α on the clustering results by a simulation experiment. The simulation experiment is carried on a one-dimensional dataset (called dataset I here), which contains 4,000 data points and is generated from the probability distribution shown in Fig. 2. The probability distribution is the synthesis of two normal distributions with the same prior probability of 0.5, one of which is with mean of 0.2 and standard deviation of 0.08, and the other of which is with mean of 0.7 and standard deviation of 0.1. So the position of the valley between two peaks in Fig. 2 is at 0.43, i.e., the position of the ideal boundary point between two cluster data points.

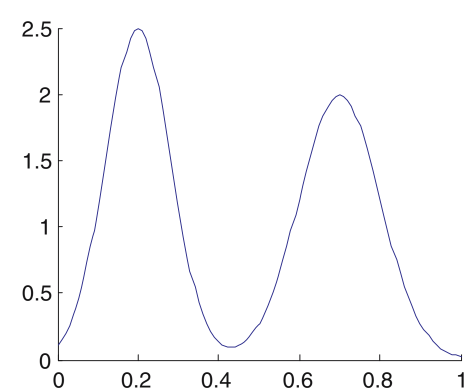

Fig. 2 The probabilistic distribution of artificial dataset I

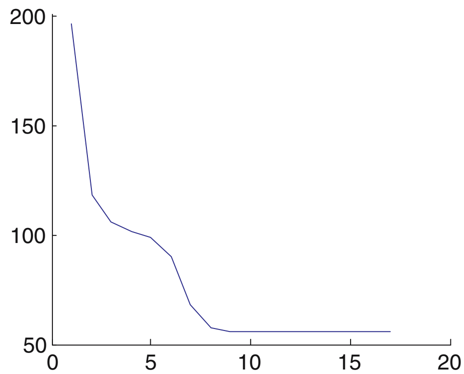

Fig. 3 The change curve of the objective function

Figure 3 shows the relationship between the objective function and the iteration number of the FHD based clustering algorithm when $\alpha = 0.4$. In fact, we will gain more similar curves for different α in interval (0, 1). Table 1 reveals the clustering result of dataset I with different α, where S_I^* and S_O^* are $\sqrt{S_I}$ and $\sqrt{S_O}$ respectively, and S_I^* may be approximately interpreted as the standard deviations of the main part of the dataset obtained by the proposed algorithm. In Table 1, P is the partition boundary point of two subgroups, which is obtained with $P = \mathbf{v}_I + R_\alpha$ when $\mathbf{v}_I$ reaches the left peak of Fig. 2, otherwise $P = \mathbf{v}_I - R_\alpha$ when $\mathbf{v}_I$ reaches the right peak of Fig. 2.

Table 1 shows that although R_α becomes bigger when α increases from 0.01 to 0.9, the corresponding tendency is very small. The obtained main cluster center $\mathbf{v}_I$ is near 0.2 or 0.7, which are very in accordance with the real cluster centers. From Table 1, we can find the best boundary partition point is obtained with $\alpha = 0.3$, the relative error of which is only 2%. Meanwhile, we can see that the relative error between S_I^* and the standard deviation of the main cluster is only 3.75%.

Therefore, the selection of parameter α has some influence on the clustering results. However, its selection is much easier than the determination of R in the HD based clustering algorithm where we must determine R appropriately before clustering. This is not a trivial task due to the big range of R. However, since $\alpha \in (0, 1)$ and even if α falls in a comparatively big interval, the obtained clustering results are still rational (see the clustering results in Table 1, where $\alpha \in [0.2, 0.6]$) such that we can easily determine α. In other words, an uneasy determination problem about R is successfully transformed into a comparatively easy determination problem about α in the above way. Furthermore, an adaptive estimation method about α will be introduced in the following subsection.

Table 1 The partition results of FHD on dataset I with different α

α	$\mathbf{v}_I$	R_α	S_I^*	S_O^*	P
0.01	0.7051	0.0410	0.0278	0.4256	0.6641
0.05	0.1977	0.0890	0.0476	0.4779	0.2867
0.1	0.1992	0.1238	0.0573	0.4940	0.3230
0.2	0.2014	0.1728	0.0685	0.5063	0.3742
0.3	0.2040	0.2104	0.0770	0.5115	0.4144
0.4	0.2074	0.2422	0.0853	0.5141	0.4496
0.5	0.6905	0.2676	0.0989	0.5070	0.4229
0.6	0.2171	0.2951	0.1038	0.5149	0.5122
0.7	0.2240	0.3174	0.1145	0.5133	0.5414
0.8	0.2328	0.3372	0.1256	0.5101	0.5700
0.9	0.2438	0.3543	0.1397	0.5051	0.5981

3.3 Enhanced possibilistic C-Means clustering algorithm EPCM

In this subsection, we propose a novel enhanced possibilistic c-means (EPCM) clustering algorithm by extending the strategy of FHD. EPCM may be viewed as a generalization of the above FHD based clustering algorithm. The objective function of EPCM is formulated as follows:

$$
\begin{aligned}
&J_{EPCM}(\mathbf{U}_I, \mathbf{U}_O, \mathbf{V}_I; X)\\
&= \sum_{i=1}^{C}\sum_{j=1}^{N} u_{I,ij}^m d_{I,ij}^2 + \sum_{i=1}^{C}\sum_{j=1}^{N} u_{O,ij}^m S_{I,I}\\
&\quad + (1-\alpha_i)\sum_{i=1}^{C}\sum_{j=1}^{N} u_{I,ij}^m St(d_{I,ij}^2 - S_{I,i})\\
&\quad + \alpha_i \sum_{i=1}^{C}\sum_{j=1}^{N} u_{O,ij}^m St(S_{O,i} - d_{I,ij}^2)\\
&s.t. \begin{cases} u_{I,ij} + u_{O,ij} = 1 & i = 1,2,\ldots,C;\ j = 1,2,\ldots,N \\ \sum_{i=1}^{C} u_{I,ij} \le 1 & j = 1,2,\ldots,N \end{cases}
\end{aligned} \tag{23}
$$

where α_i is free parameter; $\mathbf{U}_I, \mathbf{U}_O$ denote the partition matrices associated with the main and auxiliary clusters, respectively; $\mathbf{V}_I = [\mathbf{v}_{I,1}\, \mathbf{v}_{I,2} \ldots \mathbf{v}_{I,C}]$ are the centers of the main clusters, $S_{I,i}$ and $S_{O,i}$ are the extensions of $\mathrm{S_I}$ and $\mathrm{S_O}$ in (11) and can be defined respectively using (24) and (25), respectively.

$$
S_{I,i} = \frac{\sum_{j=1}^{N} u_{I,ij}^m d_{I,ij}^2}{\sum_{j=1}^{N} u_{I,ij}^m} \tag{24}
$$

$$
S_{O,i} = \frac{\sum_{j=1}^{N} u_{O,ij}^m d_{I,ij}^2}{\sum_{j=1}^{N} u_{O,ij}^m} \tag{25}
$$

In (23), a new constraint $\sum_{i=1}^{C} u_{I,ij} \le 1$ is introduced to avoid the coincident cluster problem of PCM. As revealed in Pal et al. (2005), the reason of the coincident cluster problem occurring in PCM is the independence of each cluster. To overcome this problem, the fuzzy partition matrix $\mathbf{U}^{fuzzy} =$

$[u_{ij}^{fuzzy}]$ is introduced into PCM Zhang and Yeung (2004), Pal et al. (2005), where the constraint $\sum_{i=1}^{C} u_{ij}^{fuzzy} = 1$ makes different clusters to exclude with each other. In summary, all the current PCM and its variants employ a fuzzy partition matrix to overcome the coincident cluster problem and a possibilistic matrix to weaken the influence of outliers and noises. However, by imposing a novel constraint $\sum_{i=1}^{C} u_{I,ij} \leq 1$, our algorithm can effectively solve these two problems simultaneously with only a partition matrix, because this new constraint not only makes different clusters to exclude with each other but can also weaken the influence of outliers and noises.

Now, let us discuss the implementation of EPCM. First, (23) can be simplified as follows:

$$\begin{aligned} &J_{EPCM}(\mathbf{U}_I, \mathbf{V}_I; X) \\ &\quad= \sum_{i=1}^{C}\sum_{j=1}^{N} u_{I,ij}^m d_{I,ij}^2 + \sum_{i=1}^{C}\sum_{j=1}^{N}(1-u_{I,ij})^m S_{I,i} \\ &\qquad+ (1-\alpha_i)\sum_{i=1}^{C}\sum_{j=1}^{N} u_{I,ij}^m St(d_{I,ij}^2 - S_{I,i}) \\ &\qquad+ \alpha_i \sum_{i=1}^{C}\sum_{j=1}^{N}(1-u_{I,ij})^m St(S_{O,i} - d_{I,ij}^2) \\ &s.t. \quad \sum_{i=1}^{C} u_{I,ij} \leq 1 \quad j = 1, 2, \ldots, N \end{aligned} \tag{26}$$

i.e.,

$$\begin{aligned} &J_{EPCM}(\mathbf{U}_I, \mathbf{V}_I; X) \\ &\quad= \sum_{i=1}^{C}\sum_{j=1}^{N} u_{I,ij}^m \left(d_{I,ij}^2 + (1-\alpha_i) St\left(d_{I,ij}^2 - S_{I,i}\right)\right) \\ &\qquad+ \sum_{i=1}^{C}\sum_{j=1}^{N}(1-u_{I,ij})^m \left(S_{I,i} + \alpha_i St\left(S_{O,i} - d_{I,ij}^2\right)\right) \\ &s.t. \quad \sum_{i=1}^{C} u_{I,ij} \leq 1 \quad j = 1, 2, \ldots, N \end{aligned} \tag{27}$$

Second, the following Theorem 2 can help us derive the optimization procedure of EPCM.

Theorem 2 *Given $\mathbf{V}_{I,i}$, $S_{I,i}$ and $S_{O,i}$, and set $m = 2$, the optimal $\mathbf{U}_I$ can be obtained by the following procedure of determining the optimal $\mathbf{U}_I$ (its Matlab implementation can be seen in Appendix* II).

Proof See the Appendix I.

Procedure of determining the optimal $\mathbf{U}_I$

1 Set

$$A_{ij} = d_{I,ij}^2 + (1-\alpha_i)\cdot St(d_{I,ij}^2 - S_{I,i}),$$
$$B_{ij} = S_{I,i} + \alpha_i \cdot St(S_{O,i} - d_{I,ij}^2)$$
$$U_{I,j} = \sum_{i=1}^{C} \frac{B_{ij}}{A_{ij} + B_{ij}};$$

2 If $U_{I,j} \leq 1$, then $u_{I,ij} = \frac{B_{ij}}{A_{ij}+B_{ij}}$ and terminate this procedure, otherwise, go to 3;

3 Let $L_j = 2\left(\sum_{i=1}^{C} \frac{1}{A_{ij}+B_{ij}}\right)^{-1}(U_{I,j} - 1)$;

4 If $L_j \leq 2 \times \min_i\{B_{ij}\}$, then go to 5, otherwise, go to 6;

5 Calculate $u_{I,ij} = \frac{B_{ij} - L_j/2}{A_{ij}+B_{ij}}$, then terminate this procedure;

6 Obtain the values of the variables that satisfy the following conditions using certain greedy algorithm,

$$L_j = 2\cdot\left(\sum_{i=1}^{C}\frac{1}{A_{ij}+B_{ij}} - \sum_{i\in(\prod j)}\frac{1}{A_{ij}+B_{ij}}\right)^{-1} \cdot \left(U_{I,j} - \sum_{i\in(\prod j)}\frac{B_{ij}}{A_{ij}+B_{ij}} - 1\right)$$
$$\prod j = \{i | U_{I,ij} = 0\}$$
$$u_{I,ij} = St\left(\frac{B_{ij} - L_j/2}{A_{ij}+B_{ij}}\right) \quad i = 1, 2, \ldots, C$$
$$s.t. \quad L_j > 0$$

Then terminate this procedure.

Based on the above Theorem 2, we can get the updating rules on $\mathbf{U}_I$ and $\mathbf{U}_O$. Moreover, similar to the above HD based clustering algorithm, we directly adopt the following update rule on cluster centers $\mathbf{V}_{I,i}$:

$$\mathbf{v}_{I,i} = \frac{\sum_{j=1}^{N} u_{I,ij}^m \mathbf{x}_j}{\sum_{j=1}^{N} u_{I,ij}^m} \tag{28}$$

Now, except for the cluster number C, only parameter $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_C]^T$ must be assigned by hand in EPCM. Here, we present a clustering validity index function on $\boldsymbol{\alpha}$ to estimate the optimal $\boldsymbol{\alpha}$. In general, an effective fuzzy clustering strategy attempts to assign the elements of the partition matrix $\mathbf{U}_I$ to 0 or 1. From the procedure of determining the optimal $\mathbf{U}_I$, we know that $u_{I,ij}$ may be taken as the function of $\boldsymbol{\alpha}$. So we can propose the following clustering validity index function on $\boldsymbol{\alpha}$ to estimate the optimal $\boldsymbol{\alpha}$.

$$J(\boldsymbol{\alpha}) = \sum_{i=1}^{C}\sum_{j=1}^{N} g(u_{I,ij}(\boldsymbol{\alpha})) \tag{29}$$

$$g(x) = (1 - |2x - 1|)^{\gamma} \tag{30}$$

where $g(\cdot)$ is the function of $u_{I,ij}$ and the parameter γ is usually set to be 2 . The shape of $g(x)$ in the interval $x \in [0, 1]$ with parameter $\gamma = 2$ is illustrated in Fig. 4. It can be easily found that when $u_{I,ij}$ approximates 1 or 0 the value of $g(u_{I,ij})$ approximates 0. Thus, the minimum of $J(\boldsymbol{\alpha})$ corresponds to an optimal $\boldsymbol{\alpha}$.

Based on the above discussion, we can get an optimal $\boldsymbol{\alpha}$ by minimizing (29). Furthermore, in order to avoid an inappropriate $\boldsymbol{\alpha}$, we introduce (31) and (32) as the constraints of (29). It is interesting that these two constraints may also help us avoid the coincident cluster problem in some sense.

$$S_{I,i} + (S_{O,i} - S_{I,i}) * \alpha_i/2 \le \min\{||v_{I,i} - v_{I,k}||^2 - S_{I,k}, k \ne i\} \quad \forall i \tag{31}$$

$$0.05 \le \alpha_i \le 0.6 \quad \forall i \tag{32}$$

where the upper and lower boundaries of $\boldsymbol{\alpha}$ in (32) can be slightly adjusted according to the practical requirements. The above upper and lower boundaries are obtained based on our experimental results.

In terms of the above analysis, we now give the complete description of EPCM as follows:

Algorithm EPCM

1. Use FCM to obtain an initial fuzzy partition $\mathbf{U}$; let $\mathbf{U}_I = \mathbf{U}$ and $\mathbf{U}_O = 1 - \mathbf{U}$;
2. Compute $\mathbf{V}_I$ using (28) and update $S_{I,i}$, $S_{O,i}$ using (24) and (25);
3. Minimize the index function (29) with the constraints (31) and (32) to obtain the optimal parameter $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \ldots, \alpha_C]^T$;
4. Repeat the following steps:
   a) Use the above procedure of determining the optimal $\mathbf{U}_I$ to update $\mathbf{U}_I$ and let $\mathbf{U}_O = 1 - \mathbf{U}_I$;
   b) Update the $\mathbf{V}_{I,i}$ using (28) for all i;
   c) Update the $S_{I,i}$ and $S_{O,i}$ using (24) and (25) respectively;
   Until some termination conditions are satisfied.

In the above algorithm, in order to reduce unnecessary computational burden only once optimal evaluation of $\boldsymbol{\alpha}$ is designed. In general, for EPCM and other PCM algorithms, FCM is firstly utilized to get an initial partition such that some parameters can be initially evaluated based on such a partition. In previous PCM algorithms parameters can only be evaluated one time because continuous parameter update will result in unreasonable solutions. However in EPCM, parameter $\boldsymbol{\alpha}$ can be updated at each iteration step. Even so, our experimental results about various applications demonstrate that it is enough for us to evaluate parameter $\boldsymbol{\alpha}$ only in the initialization step such that unnecessary computational burden can be reduced.

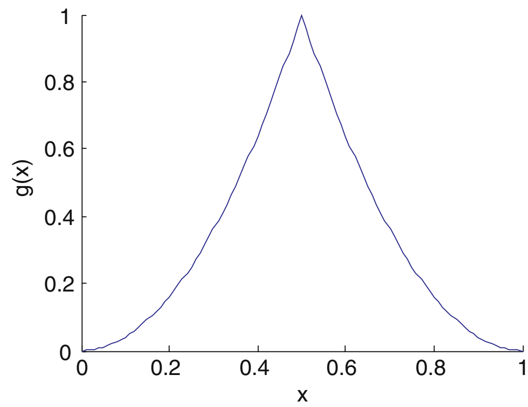

Fig. 4 The shape of g(x) at range [0,1]

Based on the above descriptions of EPCM, its computational complexity can be easily analyzed. It is obvious that the space complexity of EPCM is $O(N)$, which is equivalent to those of current possibilistic clustering algorithms. Intuitively, the time complexity of EPCM may be higher than those of other similar algorithms because of the need of performing a function minimization with constraints and a greedy algorithm at each iteration step. Based on the implementation of the adopted greedy algorithm " *ReguStep*" given in appendix II, we can deduce that its time complexity is not more than $O(L \times N \times C \times p)$, where C, N and L respectively represent the number of cluster centers, the number of data points and the step number of function " *ReguStep*'', and $p < 1$ is a factor because the adopted greedy algorithm is needed only for a few number of data points. For the procedure of the adopted function minimization with constraints, its time complexity depends on the number of the evaluations of the objective function and the time complexity of the objective function. In EPCM, the former is about several tens, and the latter is $O((L \times C \times p + M) \times N)$ (the time complexity of the procedure of determining the optimal $\mathbf{U}_I$ is little higher than the time complexity of the adopted greedy algorithm, and M is less than 5). So, the time complexity of function minimization is $O(K \times (L \times C \times p + M) \times N)$, where K is about several tens. Now, we can conclude that the total time complexity of EPCM is $O((K + T) \times (L \times C \times p + M) \times N)$, where T is the execution number of step 4 in *Algorithm EPCM*, about several tens too. Compared with other similar algorithms, whose time complexities are about $O(T \times N \times C)$, EPCM indeed needs a more burden in the running time (about several times than others'). However, it still keeps linearly proportional to the number of data points.

4 Experimental studies

To investigate the performance of EPCM, we compare its clustering results with FCM, PCM, UPCM Yang and Wu (2006) and IPCM Zhang and Yeung (2004) on several

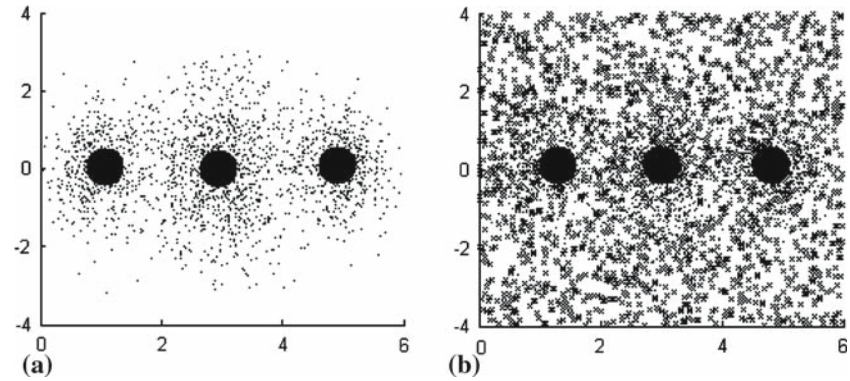

Fig. 5 The clusters obtained by EPCM on dataset II. **a** Original dataset II;**b** dataset II with noises

datasets in this section. Here, three clustering performance indexes are examined including the robustness of cluster centers to noises and outliers, the clustering accuracy and the ability to avoid the coincident cluster problem and to label the outliers accurately.

Experiment 1 In this experiment, we adopt the simulated dataset II to examine whether EPCM can avoid the coincident cluster problem or not, where the dataset is nearly the same as that in Zhang and Yeung (2004). This dataset contains three clusters and all the data points in each cluster are normally distributed over the two-dimensional space. Their respective means are (1, 0), (3, 0) and (5, 0) and the corresponding covariance matrices are $\begin{bmatrix} 0.16 & 0 \\ 0 & 0.64 \end{bmatrix}$, $\begin{bmatrix} 0.36 & 0 \\ 0 & 1.14 \end{bmatrix}$ and $\begin{bmatrix} 0.16 & 0 \\ 0 & 0.64 \end{bmatrix}$. There are 1,000 data points in the middle cluster, and 500 data points in each of the other two clusters. The cluster centers obtained by EPCM on dataset II can be seen in Fig. 5a. Figure 5b shows the cluster centers obtained from dataset II with 2,000 noisy points that are randomly and uniformly distributed over the region $[0, 6] \times [-4, 4]$.

It is indicated in Zhang and Yeung (2004) that for the dataset used in their experiments that has the same distribution as our dataset II, the clustering results of PCM generated the coincident clusters. Just like the IPCM, EPCM can effectively avoid this problem, as shown in Fig. 5.

Experiment 2 In this experiment, we use the same dataset (dataset II) as in experiment 1 to evaluate the robust capability of several clustering algorithms. To investigate the robustness of the cluster centers obtained by these five possibilistic clustering algorithms, we define the following robustness index to measure the deviation between the obtained cluster centers and the ideal cluster centers.

$$\Delta \mathbf{V} = \|\mathbf{V}_I - \mathbf{V}_{ideal}\|_F = \sqrt{\sum_{i=1}^{C}\sum_{j=1}^{N}(v_{ij} - v_{ideal,ij})^2} \tag{33}$$

where $\mathbf{V}_I$ denotes the cluster center matrix obtained by a clustering algorithm, $\mathbf{V}_{ideal}$ denotes the ideal cluster center matrix. The larger the value of $\Delta\mathbf{V}$ is, the worse the obtained clustering result is.

Table 2 shows the cluster centers and the corresponding robustness indexes obtained by five algorithms on dataset II. For this dataset II, only a valid cluster is obtained for PCM due to its coincident cluster problem, so the clustering results of PCM in Table 2 are omitted. It is clear that the cluster centers obtained by EPCM are slightly inferior to IPCM (see two bold values therein) and superior to other three algorithms. That is to say, both EPCM and IPCM have the comparable robust capability and are better than other three algorithms in robustness to noises and outliers

Experiment 3 In this experiment, the simulated dataset III that consists of an original dataset plus noises is presented. The original dataset is generated from the probability distribution in (34), whose shape is illustrated in Fig. 6, and the noisy data is added with the number of 10% original dataset points that are randomly and uniformly distributed on the range [−0.5,1.5]. In this experiment, 1,500 data points

Table 2 The cluster centers obtained by five algorithms on dataset II

	Cluster Centers	ΔV
EPCM	(1.0631, −0.0100) (4.9348, 0.0126) (2.9706, 0.0750)	**9.99E-2**
PCM	–	–
UPCM	(1.0607, −0.0071) (2.9952, 0.1212) (4.9464, 0.0099)	1.46E-1
IPCM	(1.0341, −0.0506) (2.9938, 0.0384) (4.9820, −0.0510)	**8.20E-2**
PFCM	(4.6835, −0.0068) (2.9986, 0.0486) (1.3218, 0.0133)	4.54E-1

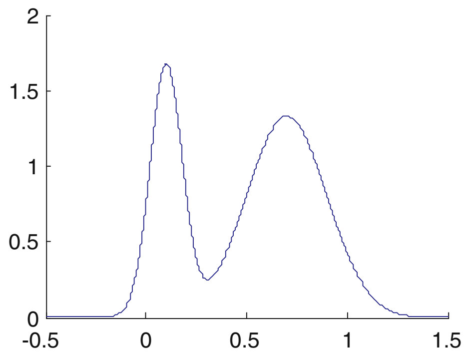

Fig. 6 The shape of the probability distribution used to generate the original dataset III

are generated for the original dataset and 150 noisy points is added to form dataset III.

$$p(x) = \frac{1}{3} \cdot \frac{1}{0.08\sqrt{2\pi}} \exp\left(-\frac{1}{2}\cdot\left(\frac{x-0.1}{0.08}\right)^2\right) + \frac{2}{3} \cdot \frac{1}{0.2\sqrt{2\pi}} \exp\left(-\frac{1}{2}\cdot\left(\frac{x-0.7}{0.2}\right)^2\right) \quad (34)$$

So, the prior probabilities and conditional probabilities of the two subgroups (represented with L cluster and R cluster respectively) and noise in the dataset III may be expressed as (35) and (36). Prior probabilities:

$$p(L) = 1/3.3,\ p(R) = 2/3.3,\ p(noise) = 0.1/1.1 \quad (35)$$

Conditional probabilities:

$$p(x|L) = \frac{1}{0.08\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-0.1}{0.08}\right)^2\right),$$
$$p(x|R) = \frac{1}{0.2\sqrt{2\pi}} \exp\left(-\frac{1}{2}\left(\frac{x-0.7}{0.2}\right)^2\right)$$
$$p(x|noise) = 1/2,\ x \in [-0.5, 1.5] \quad (36)$$

The cluster centers and the corresponding robustness indexes obtained by five algorithms on dataset III can be seen in Table 3 where the bold values mean the best results and the same meaning will be kept in the following tables.

Table 3 shows that for dataset III the cluster centers obtained by PFCM are the worst; the result of EPCM is comparable to those of PCM and UPCM, and inferior to that of IPCM.

In many situations, as the essential task in cluster analysis, the accuracy of cluster centers can be considered as a primary clustering performance index. Except for this, outliers should be labeled accurately if they appear in datasets. Although several popular possibilistic clustering algorithms have less sensitivity to noises and outliers, they cannot label them accurately in their original versions. For these possibilistic clustering algorithms, the possibilistic partition matrix is usually used to partition data points. Thus a new strategy of partitioning data points should be introduced if we want to label outliers well. In this paper, a reasonable strategy in (37) is proposed.

Table 3 The cluster centers obtained by five algorithms on dataset III

	EPCM	PCM	UPCM	IPCM	PFCM
Cluster	0.1048	0.1286	0.1105	0.1047	0.1256
Centers	0.6892	0.6852	0.7061	0.7043	0.7594
Δ**V**	1.18E−2	3.22E − 2	1.21E−2	**6.37E−3**	6.46E−2

$$\mathbf{x}_j \in \begin{cases} \text{outlier cluster} & \text{if } \sum_{i=1}^{C} u_{ij} \le \theta \\ \text{the } k\text{th cluster} & \text{if } \sum_{i=1}^{C} u_{ij} > \theta \text{ and } k = \arg\max_{1\le i\le C} u_{ij} \end{cases} \quad (37)$$

where θ is a threshold. Intuitively, θ should be different for different possibilistic clustering algorithms and different datasets, which will be detailedly examined in the following experiments.

In order to exactly compare the influence of θ on clustering performance, three clustering accuracy indexes in (38–40) are considered and the experimental results obtained by these five possibilistic clustering algorithms on dataset III are listed in Table 4–6 respectively.

$$\text{clustering accuracy of valid points} = \frac{\text{the number of accurate partitioning valid points}}{\text{the number of ideal valid points}} \quad (38)$$

$$\text{clustering accuracy of outlier points} = \frac{\text{the number of labeled outlier points}}{\text{the number of ideal outlier points}} \quad (39)$$

$$\text{total clustering accuracy} = \frac{\text{the number of total accurate partitioning points}}{\text{the number of all data points}} \quad (40)$$

To calculate the above three indexes, the ideal clustering partition should be first analyzed. Based on the Bayesian decision, assume that $p(x) = \sum_{i=1}^{C} \omega_i p_i(x) + \omega_n p_n(x)$, where ω_i, ω_n represent the prior probabilities of valid clusters and outlier cluster respectively, and $p_i(\cdot)$, $p_n(\cdot)$ represent the

Table 4 The clustering accuracy of valid points obtained by five algorithms on dataset III

	θ					
Algorithms	0.5	0.4	0.3	0.2	0.1	0
EPCM (%)	**93.57**	**95.38**	**97.35**	**98.53**	**99.11**	**99.25**
PCM (%)	81.45	85.85	89.81	93.18	96.21	96.21
UPCM (%)	73.90	79.14	85.01	88.63	90.95	95.33
IPCM (%)	40.21	47.66	57.35	69.79	88.72	97.08
PFCM (%)	78.85	88.35	90.97	93.13	94.59	94.59

Table 5 The clustering accuracy of outlier points obtained by five algorithms on dataset III

	θ					
Algorithms	0.5	0.4	0.3	0.2	0.1	0
EPCM (%)	100.0	100.0	100.0	100.0	83.24	0.00
PCM (%)	100.0	100.0	100.0	84.97	47.98	0.00
UPCM (%)	100.0	100.0	100.0	94.80	87.86	0.00
IPCM (%)	100.0	100.0	100.0	100.0	100.0	0.00
PFCM (%)	100.0	100.0	100.0	84.97	38.15	0.00

Table 6 The total clustering accuracy obtained by five algorithms on dataset III

	θ					
Algorithms	0.5	0.4	0.3	0.2	0.1	0
EPCM (%)	**94.15**	**95.80**	**97.59**	**98.66**	**97.66**	**90.23**
PCM (%)	83.14	87.14	90.74	92.43	91.82	87.46
UPCM (%)	76.27	81.04	86.37	89.19	90.94	86.67
IPCM (%)	45.64	52.52	61.23	72.54	89.74	88.26
PFCM (%)	80.77	89.41	91.79	92.39	89.46	85.99

conditional probabilities of valid clusters and outlier cluster respectively, then the ideal clustering partition of each data point can be decided according to the following formula.

$$\mathbf{x}_j \in \begin{cases} \text{kth cluster} & \omega_k p_k(\mathbf{x}_i) \geq \omega_n p_n(\mathbf{x}_i) \\ \text{outlier cluster} & \omega_k p_k(\mathbf{x}_i) < \omega_n p_n(\mathbf{x}_i) \end{cases}, \quad k = \arg\max_i \{\omega_i p_i(\mathbf{x}_j)\} \tag{41}$$

From Tables 4 and 6, we can easily find that EPCM always obtains the best clustering accuracy with different θ. From Table 5, it is clear that outliers may be labeled fully accurately at appropriate θ for all these possibilistic clustering algorithms. So how to determine the optimal θ should be studied in depth.

From the above three tables, we can also easily find that when θ decreases the clustering accuracy of valid points will increase while the clustering accuracy of outlier points will decrease. On the other hand, the total clustering accuracy will increase at first and then decrease when θ decreases. So there exists a peak of the total clustering accuracy within the range of θ.

Table 7 displays the highest accuracy and the corresponding θ obtained by these possibilistic clustering algorithms on dataset III, from which the best total clustering accuracy is still achieved by EPCM.

Table 7 The highest total clustering accuracy and the corresponding θ obtained by five algorithms on dataset III

	EPCM	PCM	UPCM	IPCM	PFCM
Accuracy (%)	**98.70**	95.00	93.98	96.08	93.82
θ	**0.1525**	0.1064	$6.54E{-4}$	0.0219	0.1245

In order to observe how different θ affects the clustering accuracy, the membership curves obtained by five algorithms on dataset III with noises are presented in Fig. 7, where the dashdotted curves denote the memberships of all the data points belonging to two clusters and the real curves denote the sum of memberships of all the data points to two clusters.

Now let us observe the possibilistic membership curves shown in Fig. 7. Compared with these real curves in Fig. 7a–e, i.e., the column sum curves of the possibilistic partition memberships of data points belonging to different clusters, we can easily find that the valley of EPCM is higher than others, so the clustering accuracy of EPCM is higher than those of other algorithms and insensitive to θ when it takes a small value (e.g., $\theta < 0.5$). Especially, we can see that the real curve of EPCM is very similar with the distribution curve of the dataset III, which really implies an appropriate partition. The above analysis explains the reason for its higher clustering accuracy.

Experiment 4 In this experiment, datasets X400 and X550 in Pal et al. (2005) are used to examine the clustering performance of EPCM. X400 consists of two clusters with two-dimensional standard normal distributions, whose centers are $[5.0\ 6.0]^{\mathrm{T}}$ and $[5.0\ 12.0]^{\mathrm{T}}$, and the number of data points of every cluster is 200. X550 is generated by X400 added into 150 noises that are randomly and uniformly distributed on $[0, 15] \times [0, 11]$. Table 8 displays the cluster centers obtained by five algorithms on X400 and X550, where $\Delta(\Delta\mathbf{V})$ is defined as the difference between the corresponding $\Delta\mathbf{V}$ obtained on X400 and X550. For these two datasets, EPCM wins the best clustering centers.

Three clustering accuracy indexes are listed in Tables 9, 10, 11, respectively, and Table 12 displays the highest total clustering accuracy and the corresponding θ obtained by five algorithms on X400 and X550. From Tables 9, 10, 11, we can find that the highest clustering accuracy is still achieved by EPCM. Moreover, UPCM has the comparable clustering accuracy with EPCM and is better than other algorithms. Table 12 displays that all these algorithms can achieve the highest total clustering accuracy with appropriate θ. In current possibilistic clustering algorithms, θ is set to be 0, that is to say, they only reduce the influence of noises and outliers on clustering centers i.e., the clustering accuracy of valid points are only involved such that the total clustering accuracy is not improved. When θ is set to be 0, the total clustering accuracies obtained by these possibilistic clustering algorithms will become lower. This poses such an interesting issue: "whether a universal θ is available for plenty of datasets for each possibilistic clustering algorithms?" The next experiment will give a negative answer and give a useful suggestion for the selection of θ in five clustering algorithms.

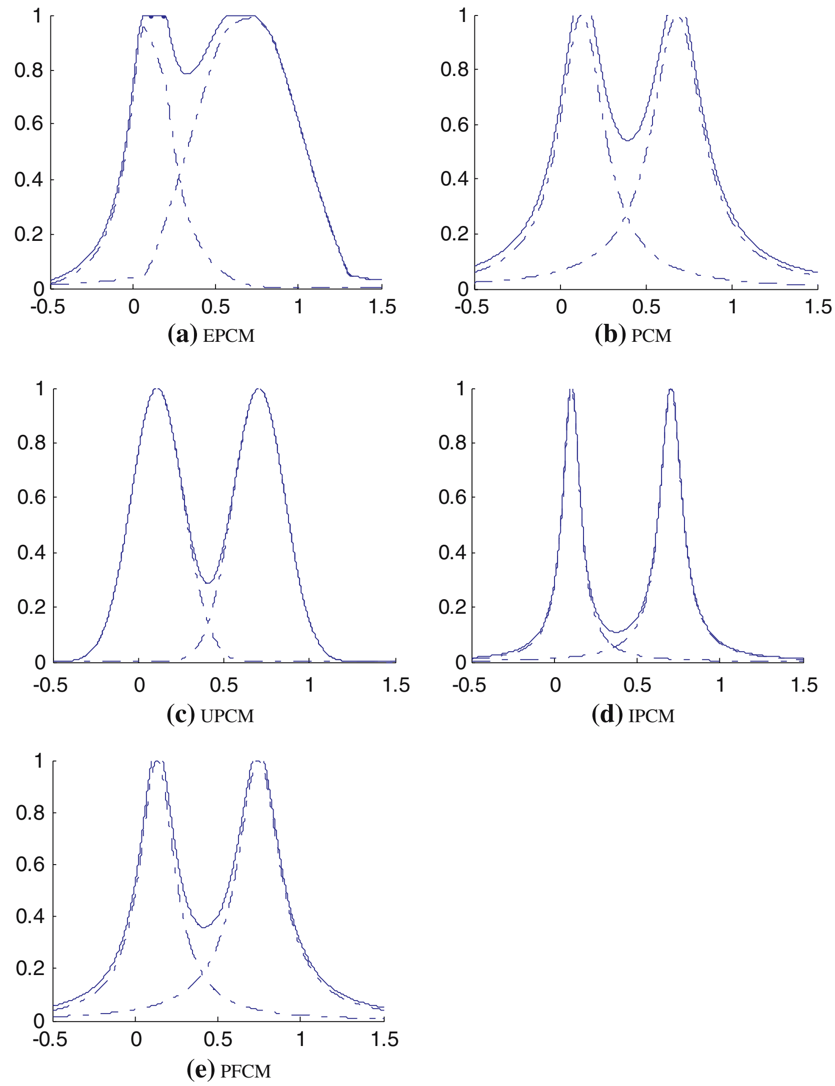

Fig. 7 The membership curves obtained by different algorithms on dataset III

Experiment 5 With the same datasets as above, we use the ranges of θ with different total clustering accuracies to investigate the clustering results obtained by five possibilistic clustering algorithms, which are shown in Table 13.

From Table 13, we can easily discover the fact that it is an impossible task to find a universal θ with which the highest total clustering accuracies (e.g., >95%) on three datasets are achieved simultaneously for five clustering algorithms. In terms of our experimental results for many datasets, we give the following suggestion about the range of θ for EPCM, PCM, UPCM, IPCM and PFCM: $\theta \in$ [0.4,0.5], [0.3,0.4], [0.1,0.2], [0.05,0.15], and [0.3,0.4] respectively, and the corresponding optimal values may be taken as 0.45, 0.35, 0.15, 0.1 and 0.35. Table 14 illustrates the total clustering accuracies obtained by five possibilistic clustering algorithms on three datasets, in which EPCM still win the highest clustering accuracy for all three datasets.

5 Application to image segmentation

In this section, we will apply EPCM to color peach shadow image segmentations. Image segmentation is a fundamental and important research topic in image processing field (Dong and Xie 2005, Cinque et al. 2004, Moghaddamzadeh and Bourbakis 1997, Tobias and Seara 2002). Clustering is one of the important image segmentation methods. The first task of color image segmentation is to choose a feature space to represent the pixels of an image. It is well known that color images are usually stored and processed by the RGB color space, but the RGB color space cannot often reflect the visual difference of eyes to identify different colors. Therefore, color images could usually be represented using some other feature spaces such as YIQ, YUV, CIE, HIS (HSL, HSB, HSV as mutants) and so on. In our experiments, the HSV feature space proposed by A. R. Smith Moghaddamzadeh and Bourbakis (1997) is adopted.

Table 8 The cluster centers obtained by five algorithms on X400 and X550

	EPCM	FCM	PCM	UPCM	IPCM	PFCM
X400	**(5.07,12.04) (5.15,5.95)**	(5.08,12.05) (5.15,5.91)	(5.08,11.99) (5.17,6.04)	(5.07,12.10) (5.17,5.94)	(5.31,12.09) (5.26,6.05)	(5.08,12.04) (5.16,5.94)
ΔV	**0.1775**	0.1987	0.1924	0.2131	0.4175	0.1929
X550	**(5.06,11.99) (5.18,5.98)**	(5.28,11.54) (5.74,5.40)	(5.05,11.37) (5.21,6.46)	**(5.05,12.00) (5.18,5.96)**	(5.17,12.08) (5.15,5.91)	(5.22,11.50) (5.54,5.74)
ΔV	**0.1910**	1.3629	0.8094	**0.1910**	0.2567	0.8109
$\Delta(\Delta V)$	**0.0135**	1.1642	0.6125	0.0221	0.1608	0.6180

Figure 8 shows original images and the corresponding segmentation results using five clustering algorithms with appropriate θ suggested in experiment 5. Let the cluster number C=3 and black pixels as outliers. Here, every peach shadow image is mainly composed of three parts: peach, background and shadow. Except for these, a lot of pixels indicate other colors. To get a good segmentation result, clustering algorithms should have robustness and high clustering accuracy because of the existence of outliers in original images. From Fig. 8, we can find that the segmentation results of EPCM for all four peach images are close to the results obtained by human vision. While for other algorithms, not all obtained results are good enough.

In summary, EPCM demonstrates higher clustering performance than all other four possibilistic clustering algorithms for this image segmentation application.

Table 15 show the average running time of five clustering algorithms on the image datasets, where the size of an image is 150×200. From this table, we can find that the time complexity of EPCM is higher than other clustering algorithms', which is in accordance with the conclusions given in the Sect. 3.3. However, we believe that such a time complexity of EPCM is still acceptable in the case where the clustering performance is more important.

6 Conclusions

In this study, to overcome the weaknesses of algorithm PCM, we propose a novel possibilistic clustering algorithm EPCM. The distinctive features can be concluded as follows.

(1) Due to the introduction of the strategy of flexible hyperspheric dichotomy, EPCM can use less parameter than PCM. Thus, EPCM actually avoid these parameters' influence on the clustering results.
(2) The objective function of EPCM with a novel constraint can effectively overcome the coincident cluster problem of PCM.
(3) EPCM can not only effectively weaken the influences of noise and outliers in datasets and obtain robust cluster centers, but can also take the effective strategy to label them, which results in the higher total clustering accuracy than other possibilistic clustering algorithms. As we may know well, to obtain better total clustering accuracy is usually more important for the clustering results than to obtain better cluster centers in real applications.

Although EPCM reveals better performances than PCM and its variants in the above, it has some problems to be further studied. For example, just like PCM, it is sensitive to initialization sensitivity problem and cannot obtain very satisfactory clustering results on non-ball distributed datasets. We will explore these open problems in near future.

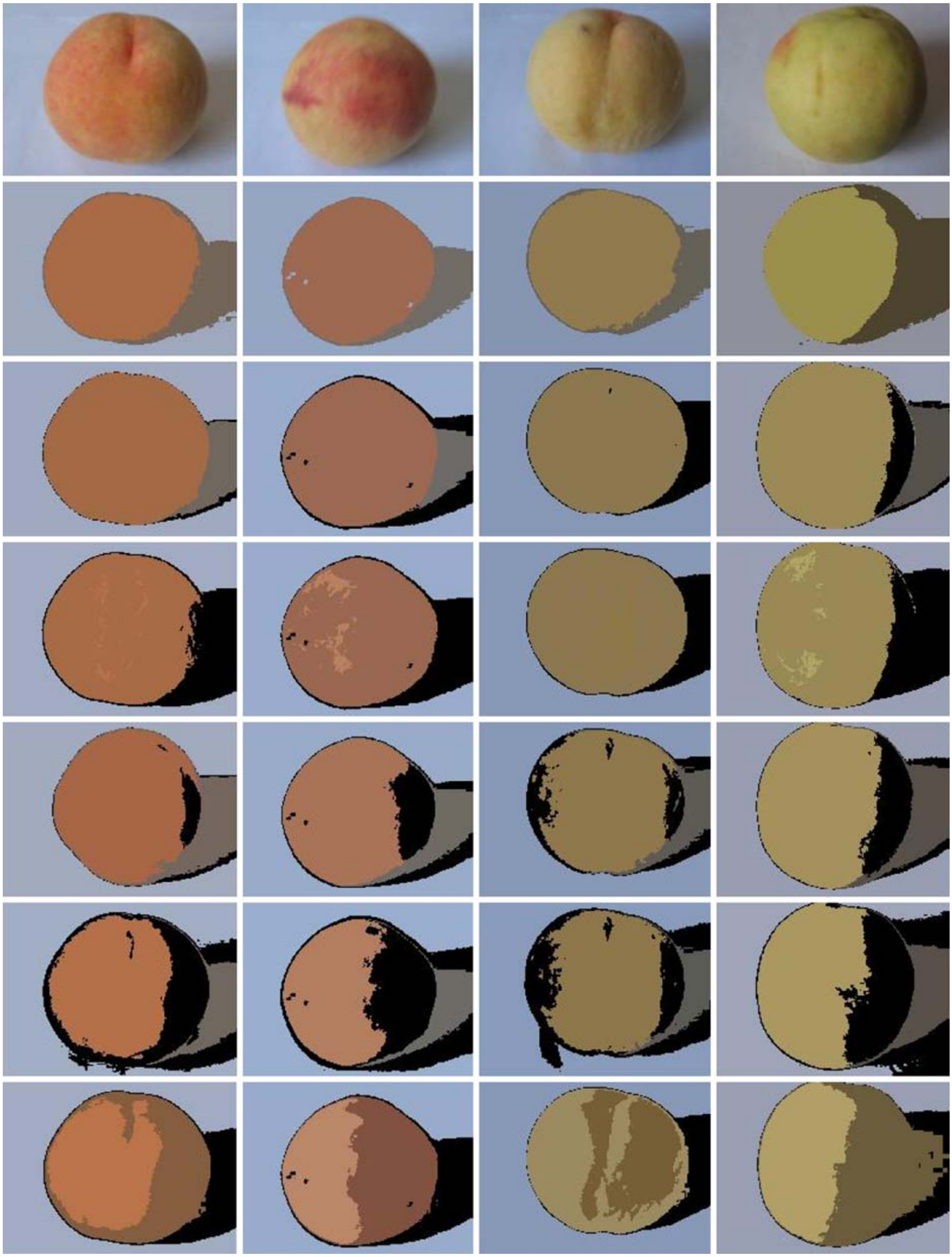

Fig. 8 The segmentation results for the peach images. The original images are arranged in the first row, and the second to seventh rows are the corresponding segmentation results obtained by FCM, EPCM, PCM, UPCM, IPCM and PFCM, respectively

Table 9 The clustering accuracy of valid points obtained by five algorithms on X400 and X550

Algorithms		θ 0.5	0.4	0.3	0.2	0.1	0
EPCM (%)	X400	100.0	100.0	100.0	100.0	100.0	100.0
	X550	91.54	98.53	100.0	100.0	100.0	100.0
PCM (%)	X400	72.00	86.02	94.50	99.50	100.0	100.0
	X550	100.0	100.0	100.0	100.0	100.0	100.0
UPCM (%)	X400	75.74	83.26	91.01	96.50	99.25	100.0
	X550	93.32	98.78	100.0	100.0	100.0	100.0
IPCM (%)	X400	33.25	46.75	61.25	83.26	99.25	100.0
	X550	60.20	76.01	92.53	100.0	100.0	100.0
PFCM (%)	X400	72.00	85.51	94.50	99.75	100.0	100.0
	X550	100.0	100.0	100.0	100.0	100.0	100.0

Table 10 The clustering accuracy of outliers labeled by five algorithms on X400 and X550

Algorithms		θ 0.5	0.4	0.3	0.2	0.1	0
EPCM (%)	X400	—	—	—	—	—	—
	X550	97.81	94.89	86.13	75.18	51.82	0.00
PCM (%)	X400	—	—	—	—	—	—
	X550	69.34	56.20	43.07	16.06	0.00	0.00
UPCM (%)	X400	—	—	—	—	—	—
	X550	99.27	97.08	89.05	80.29	68.61	0.00
IPCM (%)	X400	—	—	—	—	—	—
	X550	100.0	100.0	99.27	86.13	56.20	0.00
PFCM (%)	X400	—	—	—	—	—	—
	X550	70.07	51.82	39.42	15.33	0.00	0.00

Table 11 The total clustering accuracy obtained by five algorithms on X400 and X550

Algorithms		θ 0.5	0.4	0.3	0.2	0.1	0
EPCM (%)	X400	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
	X550	**93.09**	**97.63**	**96.54**	**93.81**	**88.00**	**75.09**
PCM (%)	X400	72.00	86.00	94.50	99.50	100.0	100.0
	X550	92.36	89.09	85.81	79.09	75.09	75.09
UPCM (%)	X400	75.77	83.25	91.00	96.50	99.25	100.0
	X550	**94.72**	**98.36**	**97.27**	**95.09**	**92.18**	75.09
IPCM (%)	X400	33.25	46.75	61.25	83.25	99.25	100.0
	X550	70.18	82.00	94.00	96.54	89.09	75.09
PFCM (%)	X400	72.00	85.50	94.50	99.75	100.0	100.0
	X550	92.54	88.00	84.90	78.90	75.09	75.09

Table 12 The highest total clustering accuracy and the corresponding θ obtained by five algorithms on X400 and X550

		EPCM	PCM	UPCM	IPCM	PFCM
X400	Accuracy (%)	100.0	100.0	100.0	100.0	100.0
	θ	[0,0.56]	[0,0.17]	[0,0.05]	[0,0.08]	[0,0.17]
X550	Accuracy (%)	97.64	95.09	98.73	97.45	94.73
	θ	0.3581	0.6168	0.3719	0.2324	0.6181

Table 13 The ranges of θ with different total clustering accuracies on three datasets

		Accuracy					
Algorithms		>95%	>90%	>85%	>80%	>70%	>60%
EPCM	Ds III	**[0,0.43]**	**[0,0.63]**	[0,0.70]	[0,0.76]	[0,0.85]	[0,0.91]
	X400	**[0,0.79]**	**[0,0.87]**	[0,0.91]	[0,0.95]	[0,0.98]	[0,0.99]
	X550	**[0,0.46]**	**[0,0.56]**	[0,0.61]	[0,0.68]	[0,0.79]	[0,0.86]
PCM	Ds III	[]	[0,0.30]	[0,0.43]	[0,0.54]	[0,0.63]	[0,0.74]
	X400	[0,0.29]	[0,0.36]	[0,0.41]	[0,0.44]	[0,0.51]	[0,0.56]
	X550	[0.61,0.63]	[0.43, 0.78]	[0.29, 0.85]	[0.21, 0.90]	[0,0.98]	[0,1.0]
UPCM	Ds III	[]	**[0,0.13]**	[0,0.30]	[0,0.39]	[0,0.58]	[0,0.72]
	X400	[0,0.22]	**[0,0.32]**	[0,0.36]	[0,0.44]	[0,0.53]	[0,0.63]
	X550	[0.19,0.49]	**[0.06, 0.58]**	[0.01, 0.64]	[0.01, 0.69]	[0,0.77]	[0,0.84]
IPCM	Ds III	[0.02,0.03]	[0,0.09]	[0,0.11]	[0,0.14]	[0,0.20]	[0,0.28]
	X400	[0,0.13]	[0,0.16]	[0,0.19]	[0,0.21]	[0,0.26]	[0,0.31]
	X550	[0.17,0.27]	[0.11, 0.34]	[0.07, 0.37]	[0.05, 0.41]	[0,0.50]	[0,0.60]
PFCM	Ds III	[]	[0,0.35]	[0,0.45]	[0,0.49]	[0,0.61]	[0,0.72]
	X400	[0,0.29]	[0,0.36]	[0,0.40]	[0,0.43]	[0,0.51]	[0,0.57]
	X550	[]	[0.45, 0.80]	[0.32, 0.87]	[0.22, 0.90]	[0,0.99]	[0,1.0]

Table 14 The total clustering accuracy obtained by five possibilistic clustering algorithms with appropriate θ on three datasets

	EPCM ($\theta = 0.45$)(%)	PCM ($\theta = 0.35$)(%)	UPCM ($\theta = 0.15$)(%)	IPCM ($\theta = 0.1$)(%)	PFCM ($\theta = 0.35$)(%)
Ds III	**94.64**	88.24	89.83	89.09	90.05
X400	**100.0**	90.50	98.25	99.25	91.25
X550	**95.82**	87.82	94.36	89.09	86.36

Table 15 The average running time of five clustering algorithms on the image datasets

EPCM	FCM	PCM	UPCM	IPCM	PFCM
13.04s	2.37s	3.52s	3.01s	6.84s	7.42s

Acknowledgments This work is supported by HongKong PolyU, JiangSu Natural Science Foundation, National 863 Grant, New_century Outstanding Young Scholar Grant of Ministry of Education of China, National KeySoft Lab. at Nanjing University, The Key Lab. of Computer Science at Institute of Software, CAS, China, The Key Lab. of Computer Information Technologies at JiangSu Province, and the 2004 key project of Ministry of Education of China.

Appendix I: The proof of Theorem 2

Proof Given $\mathbf{V}_I, \boldsymbol{\alpha}$, and $m = 2$, minimizing (27) can be transformed into the problem of minimizing (A-1).

$$J(\mathbf{U}_I; X) = \sum_{i=1}^{C}\sum_{j=1}^{N} u_{I,ij}^m * A_{ij} + \sum_{i=1}^{C}\sum_{j=1}^{N}(1-u_{I,ij})^m * B_{ij}$$
$$s.t. \quad \begin{cases} \sum_{i=1}^{C} u_{I,ij} \le 1 & \forall j \\ u_{I,ij} \ge 0 & \forall i, j \end{cases} \tag{A-1}$$

where $A_{ij} = d_{I,ij}^2 + (1-\alpha_i)\cdot St(d_{I,ij}^2 - S_{I,i})$, $B_{ij} = S_{I,i} + \alpha_i \cdot St(S_{O,i} - d_{I,ij}^2)$.

Furthermore, it can be transformed into the following N minimization problems:

$$J_j(\mathbf{U}_I; X) = \sum_{i=1}^{C} u_{I,ij}^m * A_{ij} + \sum_{i=1}^{C}(1-u_{I,ij})^m * B_{ij}$$
$$s.t. \quad \begin{cases} \sum_{i=1}^{C} u_{I,ij} \le 1 \\ u_{I,ij} \ge 0 & \forall i \end{cases} \tag{A-2}$$

To obtain the necessary conditions of (A-2), we first construct its Lagrangian function as follows.

$$J_{L,j}(U_I, L_j, \beta_{ij}; X) = \sum_{i=1}^{C} u_{I,ij}^m * A_{ij} + \sum_{i=1}^{C}(1-u_{I,ij})^m * B_{ij} - \sum_{i=1}^{C}\beta_{ij} * u_{I,ij} + L_j * \left(\sum_{i=1}^{C} u_{I,ij} - 1\right) \quad \text{(A-3)}$$

$$s.t. \begin{cases} \beta_{ij} \geq 0 & \forall i \\ L_j \geq 0 \end{cases}$$

Now the constraints on $u_{I,ij}$ are released. To differentiate (A-3) with respect to $u_{I,ij}$ (please note: $m = 2$), we have

$$u_{I,ij} = \frac{(B_{ij} + \beta_{ij}/2 - L_j/2)}{A_{ij} + B_{ij}}, \quad i = 1, 2, \ldots, C \quad \text{(A-4)}$$

By substituting (A-4) into (A-3), we obtain

$$\begin{aligned} J_{L,j}&(L_j, \beta_{ij}; X) \\ &= \sum_{i=1}^{C}\left(\frac{B_{ij} + (\beta_{ij} - L_j)/2}{A_{ij} + B_{ij}}\right)^2 \cdot A_{ij} \\ &+ \sum_{i=1}^{C}\left(\frac{A_{ij} + (L_j - \beta_{ij})/2}{A_{ij} + B_{ij}}\right)^2 \cdot B_{ij} \\ &- \sum_{i=1}^{C}\beta_{ij} \cdot \left(\frac{B_{ij} + (\beta_{ij} - L_j)/2}{A_{ij} + B_{ij}}\right) \\ &+ L_j \cdot \left(\sum_{i=1}^{C}\left(\frac{B_{ij} + (\beta_{ij} - L_j)/2}{A_{ij} + B_{ij}}\right) - 1\right) \end{aligned} \quad \text{(A-5)}$$

We differentiate (A-5) with respect to βij, L_j, and set their derivatives to be zero. Then we obtain

$$L_j = 2 * \left(\sum_{i=1}^{C}\frac{1}{A_{ij}+B_{ij}}\right)^{-1} * \left(\sum_{i=1}^{C}\frac{B_{ij}+\beta_{ij}/2}{A_{ij}+B_{ij}} - 1\right) \quad \text{(A-6)}$$

$$\beta_{ij} = L_j - 2 * B_{ij} \quad \forall i \quad \text{(A-7)}$$

To get the necessary condition on $u_{I,ij}$ of minimizing (A-2), we first introduce the following lemmas and corollaries that will help us prove this theorem.

Lemma A-1 *When* (A-2) *reaches its extremum, if* $L_j > 0$, *then* (A-6) *holds; if* $\beta_{ij} > 0$,*then* $\beta_{ij} = L_j - 2 * B_{ij}$.

Proof of Lemma A-1 If $L_j \neq 0$, we can easily obtain (A-6) by the Lagrangian method. Similarly, we can easily derive: $\beta_{ij} = L_j - 2 * B_{ij}$, if $\beta_{ij} > 0$.

Lemma A-2 *When* (A-2) *reaches its extremum, if* $\beta_{ij} > 0$, *then* $u_{I,ij} = 0$; *if* $L_j > 0$, *then* $\sum_{i=1}^{C} u_{I,ij} - 1 = 0$.

Proof of Lemma A-2 Assume that when (A-2) reaches its extremum, $\beta_{ij} > 0$ and $u_{I,ij} > 0$ holds, there must exist $\beta'_{ij} > \beta_{ij}$ or $\beta'_{ij} < \beta_{ij}$ in a small adjacent domain of β_{ij} which satisfies $J_j(\beta'_{ij}) < J_j(\beta_{ij})$ or $J_j(\beta'_{ij}) > J_j(\beta_{ij})$. This is contrary to the assumption. Therefore, if $\beta_{ij} > 0$ $u_{I,ij} = 0$ holds. Similarly, we can prove that if $L_j > 0$, then $\sum_{i=1}^{C} u_{I,ij} - 1 = 0$ when (A-2) reaches its extremum.

Lemma A-3 *When* (A-2) *reaches its extremum, if* $L_j = 0$, *then* $\beta_{ij} = 0, i = 1, 2, \ldots, C$.

Proof of Lemma A-3 Assume that when (A-2) reaches its extremum, if $L_j = 0$ then $\exists i$ $\beta_{ij} > 0$ holds. From (A-7) we know $\beta_{ij} = L_j - 2 * B_{ij}$ $\forall i$. Due to $L_j = 0$ and $B_{ij} > 0$, $\beta_{ij} = L_j - 2 * B_{ij} < 0$ $\forall i$. This is contrary to the assumption. Therefore if $L_j = 0$, there must be $\beta_{ij} = 0$.

Based on the above three lemmas, we discuss the necessary condition on $u_{I,ij}$ of minimizing (A-2) in different cases.

The necessary condition $u_{I,ij}$ of minimizing (A-2) without any constraints is $u_{I,ij} = \frac{B_{ij}}{A_{ij}+B_{ij}}$. Let $U_{I,j} = \sum_{i=1}^{C} \frac{B_{ij}}{A_{ij}+B_{ij}}$. Because there is $\sum_{i=1}^{C} u_{ij} = \sum_{i=1}^{C} \frac{B_{ij}}{A_{ij}+B_{ij}} = U_{I,j}$ at this situation, we can infer the following corollaries.

Corollary A-1 *if* $U_{I,j} \leq 1$, *the necessary condition on* $u_{I,ij}$ *of minimizing* (A-2) *can be written as*

$$u_{I,ij} = \frac{B_{ij}}{A_{ij} + B_{ij}}.$$

Now, let us discuss the case of $U_{I,j} > 1$. *When* $U_{I,j} > 1$, *we can derive the following corollaries.*

Corollary A-2 *When* (A-2) *reaches its extremum, if* $U_{I,j} > 1$, *then* $L_j > 0$.

Proof of corollary A-2 Assume that when (A-2) reaches its extremum, if $U_{I,j} > 1$ $L_j = 0$ holds. From lemma A-3, we obtain $\beta_{ij} = 0$. So we can have $u_{I,ij} = \frac{B_{ij}}{A_{ij}+B_{ij}}$ and $\sum_{i=1}^{C} u_{ij} = \sum_{i=1}^{C} \frac{B_{ij}}{A_{ij}+B_{ij}} = U_{I,j} > 1$, which is contrary to the constraint $\sum_{i=1}^{C} u_{ij} \leq 1$ in (A-2). Therefore the assumption is wrong and *corollary A-2* holds.

Corollary A-3 *Let* $\varsigma = 2 * \left(\sum_{i=1}^{C} \frac{1}{A_{ij}+B_{ij}}\right)^{-1} * (U_{I,j} - 1)$, *if* $U_{I,j} > 1$ *and* $\varsigma < 2B_{ij}$ $\forall i$, *the necessary condition of minimizing* (A-2) *is*

$$u_{I,ij} = \frac{(B_{ij} - \varsigma/2)}{A_{ij} + B_{ij}} \quad \forall i.$$

Proof of corollary A-3 From corollary A-2 and lemma A-1, we know that if $U_{I,j} > 1$ then $L_j > 0$ and $L_j = 2 * \left(\sum_{i=1}^{C} \frac{1}{A_{ij}+B_{ij}}\right)^{-1} * \left(\sum_{i=1}^{C} \frac{B_{ij}+\beta_{ij}/2}{A_{ij}+B_{ij}} - 1\right) = \varsigma + 2 * \left(\sum_{i=1}^{C} \frac{1}{A_{ij}+B_{ij}}\right)^{-1} * \left(\sum_{i=1}^{C} \frac{\beta_{ij}/2}{A_{ij}+B_{ij}}\right)$. Here, assume that there exists a $\beta_{kj} > 0$ $k \in K$, $K = \{1, 2 \ldots C\}$. Then from the Lemma A-1, we obtain $\beta_{kj} = L_j - 2 * B_{kj}$ $k \in K$. Due to $\sum_{i=1}^{C} a_i \cdot x_i / \sum_{i=1}^{C} a_i \leq \max\{x_i\}$, we have $L_j \leq \varsigma + 2 * \max\{\beta_{kj}/2\}, k \in K$.

Furthermore, we have

$$\beta_{kj} = L_j - 2B_{kj} \leq \varsigma + \max\{\beta_{kj}\} - 2B_{kj}, k \in K.$$

From the given condition $\varsigma < 2B_{ij}\ \forall i$, we have

$$\beta_{kj} < \max\{\beta_{kj}\}, k \in K.$$

This results in a contrary conclusion. So there must $\beta_{ij} = 0\ \ \forall i$ and $L_j = \varsigma$. Therefore, we obtain

$$u_{I,ij} = \frac{(B_{ij} - L_j/2)}{A_{ij} + B_{ij}} = \frac{(B_{ij} - \varsigma/2)}{A_{ij} + B_{ij}}.$$

From corollary A-2, we immediately have the following corollary:

Corollary A-4 *if $U_{I,j} > 1$ and $\varsigma < 2B_{ij}\ \forall i$, the necessary condition on $u_{I,ij}$ of minimizing (A-2) can be written as*

$$u_{I,ij} = \frac{(B_{ij} - \varsigma/2)}{A_{ij} + B_{ij}}.$$

Corollary A-5 *Let $\varsigma = 2 * \left(\sum_{i=1}^{C} \frac{1}{A_{ij}+B_{ij}}\right)^{-1} * (U_{I,j} - 1)$, if $U_{I,j} > 1$ and $\exists i\ \varsigma \geq 2 * B_{ij}$, the necessary condition on $u_{I,ij}$ of minimizing (A-2) is the solution of the following equations*:

$$\begin{cases} L_j = 2 * \left(\sum_{i=1}^{C} \frac{1}{A_{ij}+B_{ij}} - \sum_{i\in(\prod j)} \frac{1}{A_{ij}+B_{ij}}\right)^{-1} \\ \quad * \left(U_{I,j} - \sum_{i\in(\prod j)} \frac{B_{ij}}{A_{ij}+B_{ij}} - 1\right) \prod j = \{k | u_{I,kj} = 0\} \\ u_{I,ij} = St\left(\frac{B_{ij}-L_j/2}{A_{ij}+B_{ij}}\right) \quad \forall i \\ s.t. \quad L_j > 0 \end{cases}$$

$$Where\ St(x) = \begin{cases} x & x > 0 \\ 0 & x \leq 0 \end{cases}.$$

Proof of corollary A-5 For the condition $\exists i\ \varsigma \geq 2 * B_{ij}$, we know $\exists k\ \beta_{kj} > 0$. Form lemma A-2, we get $u_{I,kj} = 0$. By integrating (A-4) and (A-6) together with (A-7), we have

$$L_j = 2 * \left(\sum_{i=1}^{C} \frac{1}{A_{ij}+B_{ij}} - \sum_{i\in(\prod j)} \frac{1}{A_{ij}+B_{ij}}\right)^{-1} * \left(U_{I,j} - \sum_{i\in(\prod j)} \frac{B_{ij}}{A_{ij+B_{ij}}} - 1\right) \prod j = \{k | u_{I,kj} = 0\},$$

Meanwhile, in terms of Lemma A-2 and (A-4), we have

$$u_{I,ij} = St\left(\frac{B_{ij} - L_j/2}{A_{ij} + B_{ij}}\right) \quad \forall i.$$

So, Corollary A-5 is proved.

Based on the above Corollaries A-1, A-3 and A-5, the proof of Theorem 2 can be immediately derived.

Appendix II: The Matlab@ implementation of the procedure of determining the optimal $\mathbf{U}_I$

```
%% A Matlab@ implementation of the procedure of
%% determining the optimal U_I
%% A and B represent the A and B respectively
function U =UpdateU(A,B)
s=1./(A+B); U =B.*s;
lambda=(sum(U,1)-1)./(sum(s,1));
lambda=max(lambda,0);
U_I=U-s.*repmat(lambda,size(A,1),1);
ind=find((min(U_I,[],1))<0);
if ~isempty(ind)
  U_I(:,ind)=ReguStep(U_I(:,ind),U (:,ind),s(:,ind));
end
U =U_I;
return;
```

```
%% The greedy algorithm used to seek the solution to
%% equation listed in step 6 of %% Procedure of
determining the optimal U_I
function U_I=ReguStep(U_I,U,s)
sd=logical(U_I>0);
alpha=(sum(U.*sd,1)-1)./sum(s.*sd,1);
U_I=(U-s.*repmat(alpha,size(U,1),1)).*sd;
ind=find(min(U_I,[],1)<0);
if ~isempty(ind)
  U_new(:,ind)=ReguStep(U_I(:,ind),U(:,ind),s(:,ind));
end
return;
```

References

Zhang D, Pal SK (2002) A fuzzy clustering neural networks system design methodology. IEEE Trans Neural Netw 11(4):1174–1177

Gustafson EE, Kessel WC (1979) Fuzzy clustering with a fuzzy covariance matrix. In: Proceedings of IEEE conference on decision and control, San Diego, pp 761–776

Chung FL, Wang S, Deng Z, Shu C, Hu D (2006) Clustering analysis of gene expression data based on semi-supervised visual clustering algorithm. Soft Comput 10:981–983

Dong G, Xie M (2005) Color clustering and learning for image segmentation based on neural networks. IEEE Trans Neural Netw 16(4):925–936

Timm H, Borgelt C, Doring C, Kruse R (2001) Fuzzy cluster analysis with cluster repulsion. Presented at the Europian symposium intelligent technologies (EUNITE), Tenerife, Spain

Timm H, Kruse R (2002) A modification to improve possibilistic fuzzy cluster analysis. presented at the IEEE international conference on Fuzzy systems, FUZZ-IEEE' 2002, Honolulu

Timm H, Borgelt C, Doring C, Kruse R (2004) An extension to possibilistic fuzzy cluster analysis. Fuzzy Sets System, pp 3–16

Jian Yu (2005) General C-means Clustering Model. IEEE Trans Pattern Anal Mach Intell 27(8):1197–1211

Zhang JS, Yeung YW (2004) Improved possibilistic C-means clustering algorithms. IEEE Trans Fuzzy Syst 12(2):209–217

Bezdek JC (1981) Pattern recognition with fuzzy objective function algorithms. Plenum, New York

Cinque L, Foresti GL, Lombardi L (2004) A clustering fuzzy approach for image segmentation. Pattern Recognit 37(9):1797–1807

Yang M-S, Wu K-L (2006) Unsupervised possibilistic clustering. Pattern Recognit 39(1):5–21

Moghaddamzadeh, Bourbakis N (1997) A fuzzy region growing approach for segmentation of color images. Pattern Recognit 30(6):867–881

Barni M, Cappellini V, Mecocci A (1996) Comments on "A possibilistic approach to clustering". IEEE Trans Fuzzy Syst 4:393–396

Sato M, Sato Y, Jain LC (1997) Fuzzy clustering models and applications. Physica-Verlag, New York

Pal NR, Pal K, Keller JM, Bezdek JC (2005) A possibilistic fuzzy c-means clustering algorithm. IEEE Trans Fuzzy Syst 13(4): 517–530

Dave RN, Krishnapuram R (1997) Robust clustering methods: A unified view. IEEE Trans Fuzzy Syst 5(2):270–293

Dave RN, Sen S (2002) Robust fuzzy clustering of relational data. IEEE Trans Fuzzy Syst 10(6):713–727

Krishnapuram R, Keller JM (1993) A Possibilistic approach to clustering. IEEE Trans Fuzzy Syst 1(2):98–110

Krishnapuram R, Keller JM (1996) The possibilistic c-means algorithm: insights and recommendations. IEEE Trans Fuzzy Syst 4(3): 385–393

Wang S, Chung F-l, Xu M, Hu D, Qing L (2005) Possibility theoretic clustering. Lecture Notes in Computer Science, vol. 3644

Wang ST, Chung FL, Deng ZH (2006) Robust maximum entropy clustering algorithm with its labeling for outliers. Soft Comput 10: 555–563

Tobias OJ, Seara R (2002) Image segmentation by histogram thresholding using fuzzy sets. IEEE Trans Image Process 11(12): 1457–1465

Deng ZH, Wang ST (2005) Robust fuzzy clustering neural networks. Chin J Softw 16(8):1415–1422