Pattern Recognition 43 (2010) 1964–1974

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/pr

Fuzzy clustering with weighted medoids for relational data

Jian-Ping Mei, Lihui Chen *

Division of Information Engineering, School of EEE, Nanyang Technological University, Republic of Singapore

ARTICLE INFO

Article history:
Received 1 June 2009
Received in revised form
2 December 2009
Accepted 6 December 2009

Keywords:
Fuzzy clustering
(Dis)similarity-based
k-Medoids
Prototype weight

ABSTRACT

The well known k-medoids clustering approach groups objects through finding *k* representative objects based on the pairwise (dis)similarities of objects in the data set. In real applications, using only one object to capture or interpret each cluster may not be sufficient enough which in turn could affect the accuracy of the data analysis. In this paper, we propose a new fuzzy clustering approach called PFC for (dis)similarity-based data or relational data analysis. In PFC, objects in each fuzzy cluster carry various weights called prototype weights to represent their degrees of representativeness in that cluster. This mechanism enables each cluster to be represented by more than one objects. Compared with existing clustering approaches for relational data, PFC is able to capture the underlying structures of the data more accurately and provide richer information for the description of the resulting clusters. We introduce the detailed formulation of PFC and provide the analytical as well as experimental studies to demonstrate the merits of the proposed approach.

1. Introduction

Clustering is a technique that tries to find underlying substructures in a set of unlabeled objects. It has been widely used for data analysis and been an active subject in several research fields such as statistics, pattern recognition and machine learning. More recently, clustering plays an important role in data mining applications, information retrieval, bioinformatics and many others. In the context of machine learning, clustering is an unsupervised learning method that groups data into subgroups called clusters based on a well defined measure of similarity between two objects. Such kind of cluster-represented data provides a simpler description of the original data set but without loss of much information. A variety of clustering approaches have been developed for different goals and applications in specific areas. More comprehensive reviews of clustering approaches and clustering related issues can be found in [1–5].

Despite the diversity of the data to be clustered, here we concern two typical representations of the available data sets, based on which the clustering algorithm is applied. One is the vector representation, where the data set is associated with a set of features or variables. Each object in the data set is then treated as a vector in a feature space with each of its elements being the weight of the corresponding feature. This kind of data is referred as direct data or object data [6]. The well known k-means clustering algorithm clusters data of this representation with numerical features. For k-means, the center of each cluster is defined as the mean of objects in that cluster, and this definition is only meaningful in a feature space. In some situations, the direct data may be not available. Instead, the available data set may exist in the form of pairwise relationships, which describe the similarity or dissimilarity between each pair of two objects. This type of data is called relational data. Normally, relational data can be easily derived from the object data with proper proximity measures. However, it is very difficult or infeasible to derive object data from the relational data. This indicates that clustering method, such as k-means cannot be directly applied when the data is given in the relational form. That arises the necessity to develop clustering algorithms for relational data.

Two well studied (dis)similarity-based clustering approaches are Sequential Agglomerative Hierarchical Non-overlapping Model (SAHN) [7] and Partitioning Around Medoids (PAM) [8]. SAHN also called Hierarchical Agglomerative Clustering (HAC) is originally used for taxonomy. It generates a series of nested clusters by successively merging two closest subsets. Different criteria, also called linkage metric, such as single link, complete link and group average link, can be used to measure the similarity between two sets of objects. Although HAC performs well in many situations, it is computational consuming and the termination criterium for a proper granularity of the data partition is not clear. Other than generating a hierarchy of clusters, many other clustering approaches only produce one partition of the data, such as PAM, which is based on choosing *k* representative objects or medoids from the data set as the prototypes of *k* clusters. The recently studied spectral methods for graph partitioning problems [9] and kernel approaches, such as kernel k-means [10], can also be regarded as clustering of relational data if the affinity matrix of

* Corresponding author. Tel.: +65 67904484; fax: +65 67933318.
E-mail addresses: meij0002@ntu.edu.sg (J.-P. Mei), elhchen@ntu.edu.sg (L. Chen).

0031-3203/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2009.12.007

the graph in spectral clustering or the kernel matrix in kernel clustering is treated as the similarity matrix.

Those aforementioned approaches are typically crisp or hard, where each object is only allowed to belong to one and only one of the *k* clusters. In many real applications, one object may belong to multiple clusters with different degrees of belonging. This means the "clear cut" assignment in hard clustering may not be able to reflect the true structure of the data set with overlaps. Compared with hard clusters, clusters generated by soft clustering, such as Expectation-Maximization (EM) [11] and fuzzy clustering [12], are more natural and informative for real world applications in data analysis. Fuzzy clustering incorporates concepts from fuzzy set theory into clustering, i.e. using the membership to reflect the degree of belonging of an object to a specific cluster. As a fuzzy extension of k-means, fuzzy c-means (FCM) [12] is one of the most extensively studied fuzzy clustering approaches. Other than using the fuzzy membership to represent the object-to-cluster soft assignment, some clustering approaches started exploring two types of memberships for various purposes, such as the combination of possibilistic and fuzzy memberships to enhance the robustness [13], and co-clustering or fuzzy clustering with feature weighting to handle high-dimensional data, such as [14,15] and [16]. Recently, a dual fuzzy-possibilistic co-clustering approach is proposed as a robust clustering approach for document categorization [17]. The results of those approaches are quite promising. However, most of these approaches are k-means type formulated based on the vector space representation. In the literature, the study on fuzzy extensions of k-medoids is much less than that of k-means. One existing fuzzy k-medoids is called FCMdd [18]. FCMdd is formulated similarly as FCM but it is designed for dissimilarity-based data. By introducing fuzzy membership, FCMdd produces soft clusters with each cluster represented by one representative object.

In this paper, we propose a new clustering approach, called PFC, which not only produces fuzzy clusters of the data set but also provides more sufficient description for each individual cluster via a new mechanism. More specifically, PFC partitions objects into fuzzy clusters based on the fuzzy membership and characterize each cluster by multiple medoids with the help of prototype weight. The prototype weight of an object with respect to a cluster describes the degree of representativeness of the related object in that cluster. It is introduced to enable PFC to capture the data set more precisely and gain more detailed information on the internal structure of each cluster, which we believe is also very important and useful in data analysis. The prototype weight in PFC corresponds to the soft medoid selection in each cluster and plays a critical role in the clustering procedure.

The rest of the paper is arranged as follow: in Section 2, we briefly review some medoid-based approaches. After that, we present the details of the formulation of our method PFC in Section 3. In Section 4, experiments are set to illustrate the performance of the new approach compared with other existing approaches. Finally, conclusions of this study are drawn in Section 5.

2. Partitioning clustering for relational data

2.1. PAM: a k-medoids approach

The k-medoids approach is similar to the k-means approach since both of them produce a flat partition of the data set with *k* non-overlapped clusters that minimizes the total intra-cluster distance or dissimilarity. One significant difference between k-medoids and k-means is the selection of the center of a cluster which is used to represent that cluster. In k-medoids, it has to be one of the real objects in the data set, while in k-means, the center of a cluster, called centroid, is computed as the mean of objects belonging to that cluster. This difference makes k-means approach always be applied to object data, but k-medoids for relational data.

Given a data set of *n* objects denoted as $X=\{x_i\}_{i=1,2,\ldots,n}$, the objective of k-medoids is to find *k* representative objects called medoids, such that the total dissimilarity of other objects to their closest medoid is minimized. The objective function of k-medoids can be written as

$$J_{\text{k-medoids}}=\sum_{c=1}^{k}\sum_{x_i\in\chi_c}Dis(x_i,\delta_c), \tag{1}$$

where $Dis(x_i,\delta_c)$ denotes the dissimilarity between object x_i and the medoid of cluster *c* denoted as δ_c, and χ_c denotes the set of objects that belong to cluster *c*. In k-medoids, x_i is only assigned to the cluster, the medoid of which is most similar to this object, i.e.

$$x_i\in\chi_f \quad \text{where } f=\arg\min_{c\in\{1,2,\ldots,k\}}Dis(x_i,\delta_c). \tag{2}$$

Since the solution of medoids are located in the sample space, which is discrete, the finding of medoids typically needs to be accomplished with heuristic search algorithms. This is different from the case of k-means, in which the centroid of each cluster can be updated iteratively because the centroid is a variable in a continues space. Due to the fact that exhaustive search of medoids is an NP hard problem, Kaufman et al. proposed one approximate search algorithm called PAM [8]. The main task of PAM is to find *k* objects as medoids. Clusters are then formed by assigning each of other objects to the nearest medoid. Different scalable extensions of PAM, such as CLARA [8] and CLARANS [19] are proposed to deal with large data sets.

It is easy to see that PAM is a hard clustering approach since each object can only belong to one of the *k* clusters. Next, we review an existing fuzzy k-medoids approach which produces fuzzy clusters, i.e. each object may be assigned to all of the *k* clusters with certain degrees of memberships.

2.2. FCMdd: a fuzzy relative of k-medoids

A recent work of k-medoids based fuzzy clustering called FCMdd has been proposed in [18] with the application to Web mining tasks. The objective function of FCMdd is given as

$$J_{\text{FCMdd}}=\sum_{c=1}^{k}\sum_{i=1}^{n}u_{ci}^{m}Dis(x_i,\delta_c) \tag{3}$$

subject to

$$\sum_{c=1}^{k}u_{ci}=1,\quad i=1,2,\ldots,n, \tag{4}$$

$$u_{ci}\geq 0,\quad c=1,2\ldots,k;\quad i=1,2,\ldots,n, \tag{5}$$

where u_{ci} is the fuzzy assignment measure, and $Dis(x_i,\delta_c)$ is defined the same way as in Eq. (1). In fact, FCMdd is formulated in the same way as that of FCM in [12] except that δ_c in Eq. (3) is defined as the medoid (one of the real object in the sample space) not the centroid (a point in a continues space). This is a natural step towards fuzzy k-medoids. Compared to k-medoids, where solutions of cluster assignments and medoids are both discrete, the solutions of cluster assignments in FCMdd now are continues variables. For a current set of medoids, u_{ci} in FCMdd can be updated using membership updating formulas of several existing methods, such as

$$u_{ci} = \frac{Dis(x_i, \delta_c)^{-1/(m-1)}}{\sum_{f=1}^{k} Dis(x_i, \delta_f)^{-1/(m-1)}}. \tag{6}$$

However, the k medoids still need to be found with search algorithms due to the discrete nature of the solution. In FCMdd, given the current memberships, each medoid is decided using a heuristic algorithm to select object x_q as the medoid of cluster c, i.e. $\delta_c = x_q$, where

$$q = \arg \min_{j \in \{1,2,\ldots,n\}} \sum_{i=1}^{n} u_{ci}^{m} Dis(x_i, x_j). \tag{7}$$

This way used in FCMdd to search for the medoids is simpler than that in PAM, hence makes FCMdd run faster in real time. However, the efficiency of a simplified heuristic search has the tradeoff in effectiveness. FCMdd is much easier to be trapped into a local minimum than PAM.

2.3. Other dissimilarity-based fuzzy clustering

Other than formulating the fuzzy clustering based on k-medoids, there are some other fuzzy clustering approaches for relational data in the literature including RFCM [6], NERFCM [20], FANNY [8], FRC [21] and their recent extensions such as scalable versions for handling large relational data [22,23] and the CARD approach based on aggregation of multiple relation matrixes [24]. Different from k-medoids based approach, the only unknown variable in these approaches is the one for fuzzy memberships. In other words, all these approaches only focus on the fuzzy partition of objects. In another approach called "assignment-prototype " (A-P) [25], two variables are used to get the fuzzy partition and the soft prototype. However, A-P has some defect in formulation which makes it infeasible for handling various data sets as been observed and reported in [26].

As discussed above, in order to capture overlapping among clusters, fuzzy membership is introduced in FCMdd to allow an object to belong to more than one clusters. In both PAM and FCMdd, each cluster is represented by one medoid, which is the most central object in the cluster. We believe that other than the partition of the objects, the characteristic on the prototype of each cluster is also important in data analysis. However, the way of using one object to describe a cluster may not be sufficient enough in some cases.

To illustrate the limitation of one-medoid cluster representation, we use a simple 2-D data set shown in Fig. 1. The data set contains two well separated clusters, of which the left cluster consists of four objects and the right cluster consists of six objects. It can be seen that in the left cluster, it is insufficient to describe the cluster structure if only one of the four objects alone is chosen to represent that cluster, since no one of the four objects could be viewed as a more proper representative than the other three in that cluster. In the right cluster in Fig. 1, two objects out of the six are almost equal reasonable to be selected as the representative of the cluster. This means choosing one of them, although may not distort the cluster representation much, may fail to discover the complete set of all the candidate representative objects. From this example, we see that for some data sets, in order to capture various aspects of the cluster structure, we may need more objects rather than one to be referred as the medoids or representative objects of each cluster. Motivated by this idea, we now propose our new fuzzy clustering approach for (dis)similarity-based data analysis.

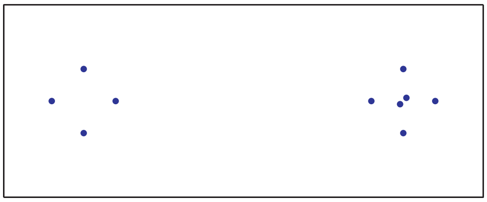

Fig. 1. A data set with two clusters.

3. The proposed approach: PFC

3.1. Problem formulation

To present the new approach, we rewrite Eq. (1) into another form as below

$$J_{k\text{-}medoids} = \sum_{c=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{n} u_{ci} v_{cj} Dis(x_i, x_j), \tag{8}$$

where

$$u_{ci} = \begin{cases} 1, & x_i \in \chi_c, \\ 0 & \text{otherwise}, \end{cases} \tag{9}$$

$$v_{cj} = \begin{cases} 1, & x_j = \delta_c, \\ 0 & \text{otherwise}. \end{cases} \tag{10}$$

Let $r_{ij} = Dis(x_i, x_j)$, then Eq. (8) can be rewritten as

$$J_{k\text{-}medoids} = \sum_{c=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{n} u_{ci} v_{cj} r_{ij}. \tag{11}$$

In Eq.(11), the value of u_{ci} and v_{cj} are still binary, i.e. each object only belongs to one of the k clusters and each cluster only has one of the n objects to be selected as the medoid or prototype. One way to obtain fuzzy memberships is by introducing an exponent m of u_{ci}. This technique is used in FCM and FCMdd. By using this parameter, called fuzzifier, together with the membership constraints, the cluster assignment becomes fuzzy rather than crisp or hard. Some researchers call such a way of FCM formulation a kind of regularization process of the crisp k-means [27]. From this point of view, other regulation methods can also be applied to obtain corresponding fuzzy versions of a crisp clustering, such as the entropy maximization approach [28]. In the entropy maximization approach, additional entropy term of u is incorporated into the objective function of k-means, therefore, the objective of this approach becomes to minimize the total dissimilarity, and at the same time maximize the entropy. The optimal result turns out to try to satisfy both criteria. Since the fuzzier the u is, the larger of the entropy term, the resulting u becomes fuzzy instead of crisp. Compared with the way of introducing an exponent m, combining individual regularization terms into the objective function corresponds to a more clear physical meaning. As the entropy regularization may cause overflow problem in computing for large scaled data [14], here we use the quadratic term instead of the entropy term as the regularization in the objective function [14,27].

As illustrated previously, in some situations, the conventional way of selecting k representative objects with each to totally represent one cluster may be insufficient to fully characterize a cluster. This in turn indicates that multiple objects may be used in order to capture each cluster more accurately. Making it more general, each object in a cluster can be referred as the representative or prototype of that cluster to some extent. The prototype weight reflects the degree of representativeness of each object in a cluster is then defined and treated as a generalization form of the one-medoid representation in Eq. (10). Therefore, we apply the quadratic regulation on both u and v in Eq. (12) to get the fuzzy membership as well as the prototype weight, respectively. The proposed method PFC is now formulated as follows:

$$J_{\text{PFC}} = \sum_{c=1}^{k}\sum_{i=1}^{n}\sum_{j=1}^{n} u_{ci}v_{cj}r_{ij} + \frac{T_u}{2}\sum_{c=1}^{k}\sum_{i=1}^{n} u_{ci}^2 + \frac{T_v}{2}\sum_{c=1}^{k}\sum_{j=1}^{n} v_{cj}^2, \tag{12}$$

with U is subject to constraints the same as in Eqs. (4) and (5), and V is subject to:

$$\sum_{j=1}^{n} v_{cj} = 1 \quad \text{for } c = 1,2,\ldots,k; \tag{13}$$

$$v_{cj} \geq 0 \quad \text{for } c = 1,2,\ldots,k, \quad j = 1,2,\ldots,n. \tag{14}$$

In Eq. (12), u_{ci} is the fuzzy membership representing the soft assignment of object i to cluster c, and v_{vj} is the prototype weight measuring how object j is weighted as a medoid in cluster c. Formulas in some other approaches may subject to the similar form of constraints like Eqs. (13) and (14), such as the normalized possibility in FPCM [29]. However, the prototype weight in PFC corresponds to the soft medoid selection. It is different from the possibility in FPCM, which measures how possible an object should be assigned to a cluster represented by an estimated centroid. This makes PFC a fuzzy clustering around weighted medoids for relational data while approaches in [29] or [13] are k-means based soft (fuzzy and possibilistic) clustering for object data.

The objective function in Eq. (12) consists of three terms. The first term is the total within cluster dissimilarity that we want to minimize. The last two terms are the quadratic terms of u and v. Two parameters T_u (>0) and T_v (>0) control the tradeoff between the first term and two regularization terms. The larger the T_u and T_v, the more the regularization terms contribute in the optimization process and the "smoother" of the resulting U and V are. Here U is said to be "smoother" or fuzzier when the u values of objects in different clusters tend to be evenly assigned and V becomes "smoother" when the v values of objects in each cluster tend to be evenly distributed. From this point of view, T_u and T_v play a similar role as parameter m in Eq. (3). Mathematically, the dissimilarity matrix R in Eq. (12) can be any form of a matrix that even may not be symmetric. But here we still consider R as a non-negative and symmetric matrix, i.e. for $i,j = 1,2,\ldots,n$

$$r_{ij} \geq 0, \quad r_{ii} = 0 \quad \text{and} \quad r_{ij} = r_{ji}. \tag{15}$$

From the above we see that the clustering problem of finding optimal cluster assignments of objects and representatives of clusters is now formulated as a constrained optimization problem, i.e. to find the optimal values of U and V subject to a set of constraints. Since U and V are continuous variables, we can use the method of Lagrange multipliers with the first order necessary condition to derive the solutions of them. The Lagrangian function is first constructed as

$$L_{\text{PFC}} = J_{\text{PFC}} + \sum_{i=1}^{N}\gamma_i\left(\sum_{c=1}^{K} u_{ci} - 1\right) + \sum_{c=1}^{K}\beta_c\left(\sum_{j=1}^{N} v_{cj} - 1\right), \tag{16}$$

where γ_i and β_c are the Lagrange multipliers. By calculating $\partial L_{\text{PFC}}/\partial u_{ci} = 0$, $\partial L_{\text{PFC}}/\partial v_{cj} = 0$, $\partial L_{\text{PFC}}/\partial \gamma_i = 0$, and $\partial L_{\text{PFC}}/\partial \beta_c = 0$, the solutions of u_{ci} and v_{cj} for $c = 1,2,\ldots,k$, $j = 1,2,\ldots,n$ can be derived as below

$$u_{ci} = \frac{1}{k} - \frac{1}{T_u}\left[\sum_{j=1}^{n} v_{cj}r_{ij} - \frac{1}{k}\sum_{f=1}^{k}\sum_{j=1}^{n} v_{fj}r_{ij}\right], \tag{17}$$

$$v_{cj} = \frac{1}{n} - \frac{1}{T_v}\left[\sum_{i=1}^{n} u_{ci}r_{ij} - \frac{1}{n}\sum_{q=1}^{n}\sum_{i=1}^{n} u_{ci}r_{iq}\right]. \tag{18}$$

It is easy to see that when $T_u \to \infty$,

$$u_{ci} = \frac{1}{k},$$

$$v_{cj} = \frac{1}{n} - \frac{1}{T_v * k}\left[\sum_{i=1}^{n} r_{ij} - \frac{1}{n}\sum_{q=1}^{n}\sum_{i=1}^{n} r_{iq}\right] \tag{19}$$

and when $T_v \to \infty$

$$u_{ci} = \frac{1}{k}, \quad v_{cj} = \frac{1}{n}. \tag{20}$$

This indicates that when T_u is set to be large enough, each object tends to have equal degrees of belonging to all the clusters, while all objects tend to be equally weighted as a prototype in each cluster when T_v is set large enough. A very large T_u makes the second term $(T_u/2)\sum_{c=1}^{k}\ \sum_{i=1}^{n} u_{ci}^2$ as a dominant term in the optimization and the optimal solution of J_{PFC} becomes the one that minimizes this term. In fact, for $i = \{1,2,\ldots,n\}$, the value of $\sum_{c=1}^{k} u_{ci}^2$ is minimized at $u_{ci} = 1/k$, hence the second term is also minimized with this solution. Similarly, when T_v is very large, the third term dominates the optimization leading the solution to minimize J_{PFC} is found by minimizing the third term. For $c = \{1,2,\ldots,n\}$, $\sum_{j=1}^{n} v_{cj}^2$ is minimized when $v_{cj} = 1/n$, with which the third term is minimized. Obviously, these two extreme results are useless for data analysis. The quadratic terms are introduced to regularize the first term, i.e. the original criterium that needs to be optimized, from generating singular solutions. However, the regularization cannot be over emphasized to make it surpass the first term and dominate the optimization. In order to get information that closely captures the underlying data structure, T_u and T_v should not be set too large.

From Eqs. (17) and (18), it can be seen that negative values of u and v might occur during the successive optimization process, hence the non-negative constraints in Eqs. (5) and (14) should be incorporated into the Lagrangian as

$$L'_{\text{PFC}} = L_{\text{PFC}} + \sum_{c=1}^{k}\sum_{i=1}^{n}\psi_{ci}u_{ci} + \sum_{c=1}^{k}\sum_{j=1}^{n}\phi_{cj}v_{cj}, \tag{21}$$

where ψ_{ci} and ϕ_{cj} are Lagrange multipliers. With the Karush–Kuhn–Tucker conditions as below:

$$\frac{\partial L'_{\text{PFC}}}{\partial u_{ci}} = 0, \tag{22}$$

$$\psi_{ci} \geq 0, \tag{23}$$

$$\psi_{ci}u_{ci} = 0. \tag{24}$$

The solution of u_{ci} can finally be derived as

$$u_{ci} = \begin{cases} 0 & \text{for } c \in k^-, \\ \dfrac{1}{|k^+|} - \dfrac{1}{T_u}\left(a_{ci} - \dfrac{1}{|k^+|}\displaystyle\sum_{f \in k^+} a_{fi}\right) & \text{for } c \in k^+, \end{cases} \tag{25}$$

where

$$a_{ci} = \sum_{j=1}^{n} v_{cj}r_{ij} \tag{26}$$

and

$$k^- = \{c : u_{ci} = 0\},$$

$$k^+ = \{c : u_{ci} > 0\}. \tag{27}$$

To a specific object x_i, k^+ represents a set of clusters where x_i belongs to or has positive values of memberships, and k^- represents a set of clusters this object does not belong to. We use $|k^+|$ and $|k^-|$ to denote the size of set k^+ and k^-, respectively. Based on the discussion in [27], we can decide k^+ and k^- for each object x_i where $i \in \{1,2,\ldots,n\}$ with procedure-K given in the next section.

Similarly, the solution of v_{cj} for each cluster $c = \{1,2,\ldots,k\}$ can be obtained as

$$v_{cj} = \begin{cases} 0 & \text{for } j \in n^-, \\ \dfrac{1}{|n^+|} - \dfrac{1}{T_v}\left(b_{cj} - \dfrac{1}{|n^+|}\sum_{q \in n^+} b_{cq}\right) & \text{for } j \in n^+, \end{cases} \tag{28}$$

where

$$b_{cj} = \sum_{i=1}^{n} u_{ci} r_{ij} \tag{29}$$

and

$$n^- = \{j : v_{cj} = 0\},$$

$$n^+ = \{j : v_{cj} > 0\}. \tag{30}$$

In a specific cluster c, the object set is split into n^+ and n^- according to whether an object has a positive value of v, and $|n^+|$, $|n^-|$ are the size of n^+ and n^-, respectively. For a cluster c, n^+ and n^- can be decided in a similar way as that of k^+ and k^-. Next, we give the details of the proposed algorithm.

3.2. The PFC algorithm

The PFC algorithm as well as the procedure-K and procedure-N are now given in Figs. 2 and 3, respectively.

First of all, different initialization schemes may be used in PFC, such as random initialization, which is the simplest one but always not an effective one. The optimization process starts by alternatively updating U and V in an iterative way until successive estimations of U is close enough. The alternative updating of U and V corresponds successively improvement in the quality of clusters with reassignment of objects to clusters and re-selection of cluster medoids. It is proved (in Appendix B) that the PFC algorithm can converge after a finite number of iterations. Based on the two final matrixes U and V, we obtain the fuzzy partition of the whole data set as well as the soft weighted medoids in each generated cluster. This cluster-based information would be very useful for users to gain more detailed description on the internal structure of each individual cluster.

```
Algorithm: PFC

Input: Dissimilarity matrix $R_{n\times n}$, the number of clusters $k$, the values of
    parameter $T_u$ and $T_v$, and the stopping criterium $\epsilon$.

Output: Fuzzy membership $U_{k\times n}$, prototype weight $V_{k\times n}$, such that the
    objective function $J_{\mathrm{PFC}}(U,V)$ in Eq. (12) is (locally) minimized.

Method:
    1. Initialize the fuzzy membership $U^{(0)}$, set iteration counter $l \leftarrow 0$;
    2. Repeat the following procedure until $\|U^{(l+1)} - U^{(l)}\| < \epsilon$:
        (a) for each $c = \{1,2,\ldots,k\}$
            i. calculate $b_{cj}^{(l)}$ from Eq. (29) using $u_{ci}^{(l)}$;
            ii. decide $n^+$ and $n^-$ using Procedure-N;
            iii. for $j = \{1,2,\ldots,n\}$
                 update the prototype weight $v_{cj}^{(l)}$ by Eq. (28) with $n^+$,
                 $n^-$ and $b_{cj}^{(l)}$;
        (b) for each $j = \{1,2,\ldots,n\}$,
            i. calculate $a_{ci}^{(l+1)}$ from Eq. (26) using $v_{cj}^{(l)}$;
            ii. decide $k^+$ and $k^-$ using Procedure-K;
            iii. for each $c = \{1,2,\ldots,k\}$
                 update the assignment measure $u_{ci}^{(l+1)}$ by Eq. (25); with
                 $k^+$, $k^-$ and $a_{ci}^{(l+1)}$;
        (c) $l \leftarrow l+1$.
```

Fig. 2. Algorithm of PFC.

```
Procedure-K:
    1. Initialize $k_0^+ = \varnothing$, $k_0^- = \{1,2,\ldots,k\}$, h=0;
    2. $h \leftarrow h+1$, $k_h^+ = k_{h-1}^+ + \{p\}$, $k_h^- = k_{h-1}^- - \{p\}$; where $p =$
       $\arg\min_{c \in k_{h-1}^-}\{a_{ci}\}$;
    3. Check whether $u_{fi} > 0$ computed by Eq. (25), where $f =$
       $\arg\max_{c \in k_h^+}\{a_{ci}\}$. If yes, go to step 2, else set $k^+ = k_{h-1}^+$,
       $k^- = k_{h-1}^-$ and terminate.

Procedure-N:
    1. Initialize $n_0^+ = \varnothing$, $n_0^- = \{1,2,\ldots,n\}$, h=0;
    2. $h \leftarrow h+1$, $n_h^+ = n_{h-1}^+ + \{q\}$, $n_h^- = n_{h-1}^- - \{q\}$; where $q =$
       $\arg\min_{j \in n_{h-1}^-}\{b_{cj}\}$;
    3. Check whether $v_{cg} > 0$ computed by Eq. (28), where $g =$
       $\arg\max_{j \in n_h^+}\{b_{cj}\}$. If yes, go to step 2, else set $n^+ = n_{h-1}^+$,
       $n^- = n_{h-1}^-$ and terminate.
```

Fig. 3. Procedure-K and procedure-N.

For a given dissimilarity matrix, the time complexity of PFC is $O(N^2)$ for one iteration, which is at the same level of complexity as other dissimilarity-based fuzzy clustering approaches like RFCM and FCMdd, but lower than that of hierarchical clustering, which is typically $O(N^2 \log N)$. In object data clustering, like k-means and FCM, the distances between objects and cluster centers need to be recalculated in every iteration. This increases the computation when the dimension of the data grows. However, in relational data clustering, the relation matrix is always calculated in advance and it does not need to be updated during the optimization process. To speed up the computation as well as reduce the storage requirement, the relation matrix can be sparsified by setting small entries to zero. For very large relational data, sampling technique, such as the framework reported in [22] may be adopted and integrated into PFC to enhance its overall scalability.

As discussed early, regularization weights T_u and T_v in PFC should be set properly to get reasonable results on different data sets. In practice, such kind of parameters always need to be tuned empirically due to the difficulty in developing generic parameter estimation approaches theoretically. We generally follow the guideline discussed in [17] of setting T_u or T_v by starting with a relative large value, and than gradually reducing it until a reasonable clustering result can be found. Based on our experimental study, reasonable results are always obtained when T_u and T_v are set to make T_u/k and T_v/n close to each other or be in the same order. This is because the summation constraint on u is over k clusters while on v is over n objects and the value of n is typically much larger than k. Therefore, the final clustering result is always more sensitive to T_u than T_v when $n \gg k$.

4. Experimental results

In this section, four sets of experiments are set up to evaluate the performance of PFC in various aspects. The first experiment is to show that multiple objects is able to capture the internal structure of each cluster more accurately than a single object. In the second experiment, the difference between prototype weight and fuzzy membership to interpret the structure of the data set is visualized with contour plots, which confirms that prototype weight is a better choice to naturally reflect the internal characteristic of each cluster. In the third experiment, we show the overall performance of PFC compared with single-medoid based approaches PAM and FCMdd as well as other two dissimilarity based fuzzy clustering approaches A-P and FRC. Finally, we show the clustering accuracy of PFC compared with other state-of-the-art dissimilarity-based clustering approaches on UCI benchmark data sets.

In our experiments, we follow the same initialization procedure as the one used in [18] to generate a set of k initial medoids $\Delta^0=\{\delta_1,\delta_2,\ldots,\delta_k\}$ for FCMdd. More specifically, k initial medoids are obtained one by one by successively selecting an object which is most dissimilar from those selected ones. The first medoid can be sampled randomly or chosen as the object with the smallest total dissimilarity to all other objects, i.e. $\delta_1=x_p$, where $p=\mathrm{argmin}_{i\in\{1,2,\ldots,n\}}\sum_{j=1}^{n} r_{ij}$. Medoids selected in such a way are well separated from each other and works better than random initialization as reported in [18]. For PFC, A-P and FRC, we initialize fuzzy membership U^0 calculated based on the selected Δ^0, i.e. for $i=1,2,\ldots,n,c=1,2,\ldots,k$, $u_{ci}=s_{ci}/\sum_{f=1}^{k} s_{fi}$, where $s_{ci}=\max_{i,j}\{r_{ij}\}-r(\delta_c,x_i)$.

4.1. X_{11} data set

In this experiment, we compare the results of PFC with FCMdd on the data set X_{11} which is shown in Fig. 4. The coordinates of X_{11} are included in Table 1. It can be seen that X_{11} data is similar to the data shown in Fig. 1 except including one more object 11 which locates slightly biased to the cluster on the right hand side. The dissimilarity matrix is calculated by using Euclidean distance and then scaled by $r_{ij}=d_{ij}/\max_{i,j}\{d_{ij}\}$. We set $k=2$, $\varepsilon=1.0\times10^{-5}$, $m=2$ for FCMdd and $T_u=1$, $T_v=1$ for PFC. The results of FCMdd and PFC are shown in Table 1. The medoids found by FCMdd are denoted with "*".

From Table 1, it can be seen that generally, the partition results obtained by two methods are similar, which indicates that object 1–4 are in cluster 1 with object 4 the most representative object and object 5–10 are in cluster 2 with object 7 the most representative object. However, when take a close look, it can be found that in FCMdd, the u values of medoids in the cluster they belong to are always one, e.g. $u_{1,4}=u_{2,7}=1$. This is not happen in PFC, since the objects with the largest v values are not necessary the same as those that have the largest u values, e.g. in cluster 1, object 4 has the largest v but its u value is the smallest among objects in that cluster. The reason that in FCMdd, the medoid objects always have $u=1$ in the cluster it represents is because the distance between a medoid to itself is 0! Therefore, according to Eq. (6), the fuzzy membership of the medoid in the cluster it represents is 1. In other words, in FCMdd, each object selected as the medoid of a cluster is not allowed to belong to other clusters at all. In PFC, for each cluster, no object alone is served as the single medoid, but combines together with other objects with different weights to characterize the cluster. Therefore, each individual object still can belong to other clusters to a certain extent. It also can be observed the difference of two approaches in labeling object 11. Although object 11 is closer to the center of cluster 2, in FCMdd, the crisp assignment of object 11 is cluster 1. This is because the distance from object 11 to the medoid of cluster 1, which is object 4, is smaller than that to the medoid of cluster 2 which is object 7, hence based on Eq. (6), the u of object 11 in cluster 1 is larger than that in the other cluster. This illustrates the limitation of using a single object to characterize a cluster, especially in cases where no object is near enough to the ideal center of the cluster. Nevertheless, this situation has been handled to a large extent in PFC, which assigns a larger u to object 11 in cluster 2. This demonstrates that using multiple objects with various prototype weights is able to characterize the structure of a cluster more precisely than using only one object, which in turn helps to produce a partition with a better quality.

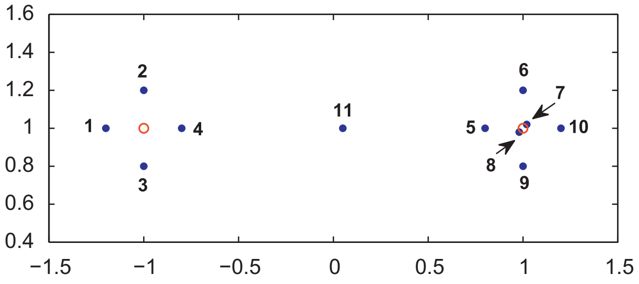

Fig. 4. The scatter plot of X_{11} data. The red circle "◦" represents the ideal center of a cluster. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

4.2. X_{600} data set

To show the different roles u and v playing in interpreting the data structure, in this experiment, the distributions of u and v are visualized through the contour plots. The data set used in this experiment is plot in Fig. 5a. It is created by generating two groups of 300 random points with Gaussian distributions of the same variance 0.1, but one is centered at (0, 1) while the other at (1.2, 0.7). The dissimilarity data is obtained with Euclidean distance. Other parameters are set as $k=2$, $m=2$ for FCMdd, $T_u=0.3$, $T_v=200$ for PFC. Truncated clusters are obtained by assigning each of the objects to the cluster associated with the largest u values.

Table 1
Results of FCMdd and PFC on X_{11}.

ID	Coordinates		FCMdd ($m=1.9$)				PFC ($T_u=2.5,\ T_v=3.0$)				
	x	y	u_1	u_2	Label	Medoid	u_1	u_2	Label	v_1	v_2
1	−1.20	1.00	0.8681	0.1319	1	–	0.8798	0.1202	1	0.1687	0.0000
2	−1.00	1.20	0.8974	0.1026	1	–	0.8484	0.1516	1	0.2471	0.0000
3	−1.00	0.80	0.8972	0.1028	1	–	0.8484	0.1516	1	0.2471	0.0000
4	−0.80	1.00	1.0000	0.0000	1	*	0.8134	0.1866	1	0.3370	0.0000
5	0.80	1.00	0.0816	0.9184	2	–	0.1838	0.8162	2	0.0000	0.1904
6	1.00	1.20	0.0880	0.9120	2	–	0.1478	0.8522	2	0.0000	0.1142
7	0.98	0.98	0.0000	1.0000	2	*	0.1309	0.8691	2	0.0000	0.2731
8	1.02	1.02	0.0207	0.9793	2	–	0.1242	0.8758	2	0.0000	0.2591
9	1.00	0.80	0.0719	0.9281	2	–	0.1477	0.8523	2	0.0000	0.1142
10	1.20	1.00	0.0796	0.9204	2	–	0.1154	0.8846	2	0.0000	0.0491
11	0.05	1.00	**0.5250**	**0.4750**	1	–	**0.4801**	**0.5199**	2	0.0000	0.0000

Fig. 5 b and c, respectively, show the contour plot of u in FCMdd and v in PFC of objects with respect to the cluster they belong to. It can be seen in Fig. 5 b that with respect to the ideal center of a cluster, contours related to that cluster are asymmetric. This means that for objects in a cluster which are equally distant to the cluster center, the closer of an object to the other cluster, the smaller value of u is assigned to this object. This indicates that the u value of each object in a cluster is influenced by other clusters. Therefore, u may not be a proper or accurate measurement to discover the internal structure of individual cluster, although it may be plausible to obtain a partition of the data by labeling each of the objects based on u. From Fig. 5 c, it can be seen that all contours related to each cluster are concentrate circles with the center located very near the ideal center of the cluster. This indicates that each object in a cluster is weighted only based on its closeness to the center of that cluster, i.e. the closer an object is to the ideal center, the larger value of v is assigned to this object. With respect to this, prototype weight is a good description of the internal structure of individual cluster.

4.3. Countries data

Here, we test PFC on a direct relational data, referred as the benchmark data set Countries Data [8]. The task is to group twelve countries into clusters based on pairwise dissimilarities as given in Table 2. Different from the dissimilarity data in the previous experiments which is calculated from the object data with Euclidean distance, the Countries Data in Table 2 is provided subjectively by students in a political science class. Generally, these countries are classified into three categories: Western, Developing and Communist. The PFC is compared with three popular dissimilarity-based fuzzy clustering in the literature, namely FCMdd, A-P and FRC, and the well known k-medoids based hard clustering PAM. In this experiment, the original

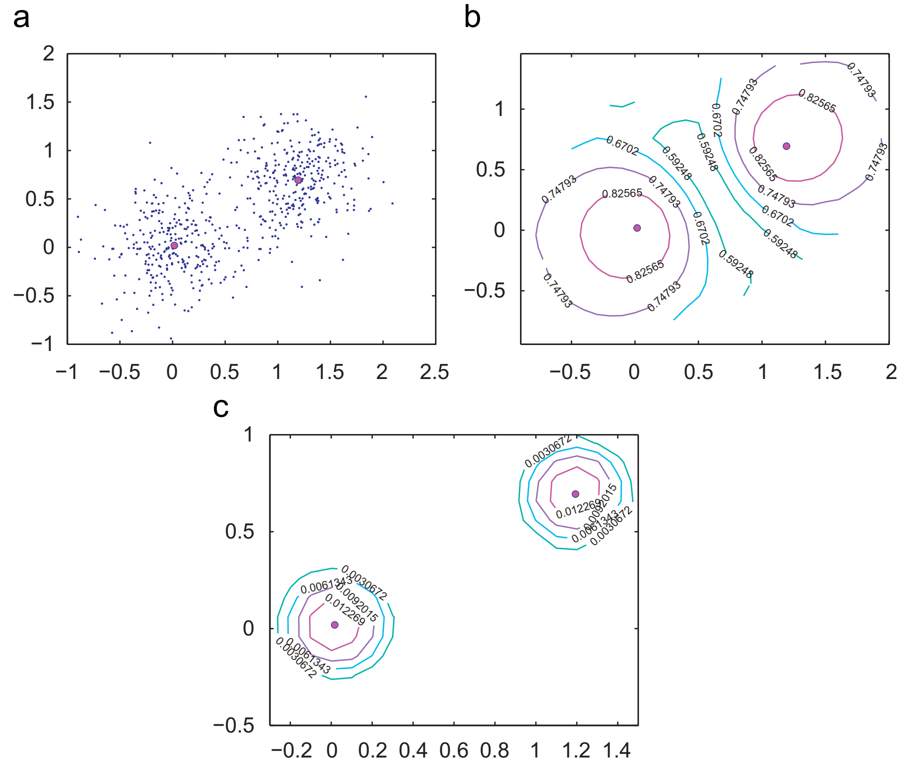

Fig. 5. Resulting fuzzy membership (u) and prototype weight (v) of X_{600} data: (a) the scatter plot of X_{600} data, (b) the contour plot of u of objects in the cluster they belong to with FCMdd, and (c) the contour plot of v of objects in the cluster they belong to with PFC. The red dot "•" represents the ideal center of a cluster. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 2
Countries data [8]: dissimilarity matrix ($d_{ij} \in D$).

Countries	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12
C1: Belgium	0.00	5.58	7.00	7.08	4.83	2.17	6.42	3.42	2.50	6.08	5.25	4.75
C2: Brazil	5.58	0.00	6.50	7.00	5.08	5.75	5.00	5.50	4.92	6.67	6.83	3.00
C3: China	7.00	6.50	0.00	3.83	8.17	6.67	5.58	6.42	6.25	4.25	4.50	6.08
C4: Cuba	7.08	7.00	3.83	0.00	5.83	6.92	6.00	6.42	7.33	2.67	3.75	6.67
C5: Egypt	4.83	5.08	8.17	5.83	0.00	4.92	4.67	5.00	4.50	6.00	5.75	5.00
C6: France	2.17	5.75	6.67	6.92	4.92	0.00	6.42	3.92	2.25	6.17	5.42	5.58
C7: India	6.42	5.00	5.58	6.00	4.67	6.42	0.00	6.17	6.33	6.17	6.08	4.83
C8: Israel	3.42	5.50	6.42	6.42	5.00	3.92	6.17	0.00	2.75	6.92	5.83	6.17
C9: USA	2.50	4.92	6.25	7.33	4.50	2.25	6.33	2.75	0.00	6.17	6.67	5.67
C10: USSR	6.08	6.67	4.25	2.67	6.00	6.17	6.17	6.92	6.17	0.00	3.67	6.50
C11:Yugoslavia	5.25	6.83	4.50	3.75	5.75	5.42	6.08	5.83	6.67	3.67	0.00	6.92
C12: Zaire	4.75	3.00	6.08	6.67	5.00	5.58	4.83	6.17	5.67	6.50	6.92	0.00

dissimilarity is scaled to give $r_{ij} = d_{ij}/\max_{i,j}\{d_{ij}\}$. The number of clusters is set to $k=3$, and other parameters are set as $m=1.5$ for FCMdd and FRC, $T_u = 0.5$, $T_v = 3$ for PFC. We test the performances of the four fuzzy approaches with two different sets of initial representative countries which are

$$\Delta_1^0 = \{USSR, Israel, India\},$$

$$\Delta_2^0 = \{France, Cuba, Belgium\}. \tag{31}$$

The three countries in Δ_1^0 are well separated while in Δ_2^0, *Belgium* is similar to *France*, which makes two initial medoids of three are not well separated.

The results of FCMdd, A-P as well as FRC and PFC are given in Table 3 to Table 5, respectively. To make comparison, the results of PAM reported in [8] are also listed in Table 5. The final medoids found by FCMdd and PAM are labeled with "*". The crisp labels of each of the countries are also included in these tables.

From Table 3, it can be seen that FCMdd is very sensitive to initializations. When Δ_1^0 is used, two of the three final representative countries are the same as the initial two. In other words, only one of the medoids has been updated before reaching a local minimum. Things become worse when the initialization is changed to Δ_2^0. Although the initial set of medoids are far from optimal, the updating ceased before any improvement in the medoids, which cause the mixture up of Western and Developing. This example illustrates that FCMdd is quite easy to be stuck in a local minimum.

For other three approaches, the results are the same even with different initializations. However, the performance of A-P is not as good as FRC and PFC. As shown in Table 4, in A-P, all u values are very similar and the v values of each object in all clusters are also nearly the same. The corresponding crisp partition based on the u values only consists of two categories and the country *Belgium* has the largest v in all three clusters. The resulting U and corresponding crisp labels of each object of FRC are similar as PFC, which accurately forms three categories of countries. It can be seen that the final crisp label of *Egypt* in these two approach is different from that in PAM. In FRC and PFC, *Egypt* is labeled as a Developing country while in PAM as a Western country. However, if we take a close look at the u values of *Egypt* in FRC and PFC, it can be found that the u values of *Egypt* in these two clusters are close. This indicates that *Egypt* is not so well belongs to Developing or Western alone, but belongs to both categories. This shows the advantage of fuzzy clustering, since compared with the crisp labels of objects, more detailed information can be derived from the fuzzy memberships.

Further more, based on the v values of PFC in Table 5, it can be seen that *USA*, *Cuba* and *Zaire* are the most representative countries in their groups. This is consistent with the medoids found by PAM. Moreover, based on V, we can obtain the rank of countries in each category and other detailed information on the internal cluster structure, e.g. *Israel* is much less representative than other three countries in the category of Western, and in the Developing category, *Egypt* and *India* are much less representative than other two countries. This example confirms that PFC is a new

Table 3
Results of FCMdd on countries data ($K=3$).

Country	FCMdd ($m=1.5$) with Δ_1^0					FCMdd ($m=1.5$) with Δ_2^0				
	u_1	u_2	u_3	Label	Medoid	u_1	u_2	u_3	Label	Medoid
Belgium	0.7572	0.1280	0.1148	1	–	0.0000	0.0000	1.0000	3	*
France	0.7963	0.1059	0.0978	1	–	1.0000	0.0000	0.0000	1	*
Israel	0.7371	0.1164	0.1464	1	–	0.3722	0.1388	0.4890	3	–
USA	1.0000	0.0000	0.0000	1	*	0.5251	0.0495	0.4254	1	–
China	0.2264	0.4896	0.2840	2	–	0.2024	0.6138	0.1838	2	–
Cuba	0.0997	0.7515	0.1488	2	–	0.0000	1.0000	0.0000	2	*
USSR	0.0000	1.0000	0.0000	2	*	0.1357	0.7246	0.1397	2	–
Yugoslavia	0.1816	0.5998	0.2186	2	–	0.2407	0.5028	0.2565	2	–
Brazil	0.3980	0.2166	0.3854	1	–	0.3654	0.2466	0.3880	3	–
Egypt	0.4014	0.2258	0.3727	1	–	0.3637	0.2590	0.3773	3	–
India	0.0000	0.0000	1.0000	3	*	0.3180	0.3640	0.3180	2	–
Zaire	0.3186	0.2424	0.4390	3	–	0.3247	0.2272	0.4481	3	–

Table 4
Results of A-P and FRC on countries data ($K=3$).

Country	A-P with Δ_1^0 and Δ_2^0							FRC ($m=1.5$) with Δ_1^0 and Δ_2^0			
	u_1	u_2	u_3	Label	v_1	v_2	v_3	u_1	u_2	u_3	Label
Belgium	0.3335	0.3332	0.3333	1	0.0908	0.0907	0.0908	0.8972	0.0372	0.0656	1
France	0.3334	0.3332	0.3333	1	0.0891	0.0890	0.0890	0.8984	0.0413	0.0603	1
Israel	0.3334	0.3332	0.3333	1	0.0855	0.0854	0.0855	0.7798	0.0886	0.1316	1
USA	0.3335	0.3332	0.3333	1	0.0904	0.0903	0.0904	0.9267	0.0251	0.0482	1
China	0.3332	0.3334	0.3333	2	0.0766	0.0767	0.0766	0.1007	0.7612	0.1381	2
Cuba	0.3332	0.3335	0.3333	2	0.0787	0.0788	0.0787	0.0341	0.9175	0.0484	2
USSR	0.3332	0.3335	0.3333	2	0.0816	0.0817	0.0816	0.0501	0.8910	0.0589	2
Yugoslavia	0.3333	0.3334	0.3333	2	0.0824	0.0825	0.0824	0.109	0.7904	0.1005	2
Brazil	0.3334	0.3333	0.3333	1	0.0809	0.0809	0.0809	0.0818	0.0526	0.8656	3
Egypt	0.3334	0.3333	0.3333	1	0.0837	0.0837	0.0837	0.3175	0.1617	0.5208	3
India	0.3333	0.3334	0.3333	2	0.0785	0.0786	0.0785	0.1382	0.1792	0.6826	3
Zaire	0.3333	0.3333	0.3333	**?**	0.0818	0.0818	0.0818	0.0723	0.0521	0.8756	3

Table 5
Results of PFC and PAM on countries data ($K=3$).

Country	PFC with Δ_1^0 and Δ_2^0							PAM	
	u_1	u_2	u_3	Label	v_1	v_1	v_3	Label	Medoid
Belgium	0.8018	0.0000	0.1982	1	0.2726	0.0000	0.0856	1	-
France	0.8343	0.0000	0.1657	1	0.2559	0.0000	0.0483	1	–
Israel	0.7707	0.0000	0.2293	1	0.1825	0.0000	0.0384	1	–
USA	0.8147	0.0000	0.1853	1	**0.2891**	0.0000	0.0923	1	*
China	0.0662	0.8623	0.0716	2	0.0000	0.1964	0.0000	2	–
Cuba	0.0000	1.0000	0.0000	2	0.0000	**0.3040**	0.0000	2	*
USSR	0.0247	0.9681	0.0072	2	0.0000	0.2865	0.0000	2	–
Yugoslavia	0.1530	0.8224	0.0246	2	0.0000	0.2131	0.0000	2	–
Brazil	0.2674	0.0000	0.7326	3	0.0000	0.0000	0.2261	3	–
Egypt	0.4056	0.0304	0.5640	3	0.0000	0.0000	0.1542	1	–
India	0.1646	0.2540	0.5815	3	0.0000	0.0000	0.1245	3	–
Zaire	0.2551	0.0000	0.7449	3	0.0000	0.0000	**0.2306**	3	*

Table 6
A summary of six UCI data sets.

Data set	No. objects	No. variables	No. clusters
Wine	178	13	3
Segment	2310	19	7
Breast-cancer	569	30	2
Iris	150	4	3
Digits	1797	64	10
Glass	214	10	6

Table 7
Parameter setting.

Data set	FRC (m)	FCMdd (m)	PFC (T_u, T_v)
Wine	1.3	1.5	$\{0.1, 200\}$
Segment	1.5	1.01	$\{0.3, 100\}$
Breast-cancer	1.5	1.1	$\{0.01, 10\}$
Iris	2.5	2	$\{0.3, 50\}$
Digits	1.05	1.05	$\{0.01, 500\}$
Glass	1.05	1.01	$\{0.01, 50\}$

approach with high potential since it is able to perform well on direct relational data by producing both good clusters and prototypes, meanwhile more detailed information on the data structure can be obtained throughout the clustering process.

4.4. *UCI data sets*

Finally, we compare the clustering accuracy of PFC with state-of the-art approaches including spectral clustering NCut [9], fuzzy relational clustering FRC [21], fuzzy k-medoid FCMdd [18] and the well-known k-medoid approach PAM [8] on six benchmark UCI data sets [30] summarized in Table 6. Euclidean distance is used as the dissimilarity measure for all data sets. For all the data sets except the *iris* data, the values associated to every attribute is linearly scaled to $[0,1]$ before calculating the dissimilarities. We evaluate the quality of cluster produced by each approach in terms of clustering accuracy, i.e. n_c/n, where n_c is the number of objects that correctly labeled, n is the total number of objects in the data set. With parameter setting listed in Table 7, we obtain the clustering accuracy of five approaches on each data set in Table 8.

Table 8
Accuracy of clustering results.

Data set	NCut	PAM	FCMdd	FRC	PFC
Wine	0.9719	0.9045	0.8371	0.9663	0.9607
Segment	0.7052	0.6675	0.6866	0.7108	**0.7113**
Breast-cancer	0.9262	0.9350	0.9350	0.9315	**0.9385**
Iris	0.8867	0.8933	0.9333	0.9267	**0.9333**
Digits	0.7941	0.7958	0.6255	0.8002	**0.8525**
Glass	0.5374	0.5280	0.5935	0.5841	**0.6308**

It can be seen from Table 8 that PFC achieves the best accuracy on five out of six data sets. The improvement of PFC compared to other approaches is especially significant on the last two data sets, namely *digits* and *glass*. On the *wine* data, the accuracy of PFC is slightly lower than NCut and FRC, but is still much better than "one-medoid for one cluster" based PAM and FCMdd.

5. Conclusions

We have proposed a new type of dissimilarity-based fuzzy clustering called PFC, which finds not only fuzzy clusters but also weighted medoids or representative objects for each cluster. We believe that in real world (dis)similarity-based data analysis, information on both cluster labeling and internal structure of each of the resulting clusters are important. One distinctive characteristic of the proposed method is that we apply the quadratic regularization technique to obtain both the fuzzy memberships and prototype weights of objects in each cluster for relational data analysis. Compared with existing (dis)similarity-based fuzzy clustering, where no medoid is defined or each cluster is represented by one object, PFC approach has several merits. Rather than using heuristic search methods, the solution of medoids can be obtained in terms of the prototype weights which are found in a continues space with the formula derived from the objective function. Moreover, the way to use prototype weights enables PFC to capture the internal cluster structure more precisely and completely hence improves the quality of the resulting clusters. Meanwhile, more detailed information on the discovered clusters can be obtained with the prototype weights. Our experiment results on different types of relational data show that PFC can be a very promising tool for dissimilarity-based data analysis.

Appendix A. The derivation of update equations

Here we give the detailed derivation of the update equations of U and V from the Lagrangian in Eq. (21) and Karush–Kuhn–Tucker conditions in Eqs. (22)–(24). Combining Eq. (21) and (22) it gives

$$u_{ci} = \frac{1}{k} - \frac{1}{T_u}\left[(\alpha_{ci}+\psi_{ci}) - \frac{1}{k}\sum_{f=1}^{k}(\alpha_{fi}+\psi_{fi})\right], \tag{32}$$

where

$$\alpha_{ci} = \sum_{j=1}^{n} v_{cj} r_{ij}. \tag{33}$$

Now we consider the condition Eq. (23) from two cases, respectively.

1. $\psi_{ci} = 0$, $\forall c \in \{1, 2, \dots k\}$, so Eq. (32) becomes

$$u_{ci} = \frac{1}{k} - \frac{1}{T_u}\left[\alpha_{ci} - \frac{1}{k}\sum_{f=1}^{k}\alpha_{fi}\right], \tag{34}$$

which is the same as Eq. (17). For each object, this set of solution is only valid if

$$\frac{1}{k} - \frac{1}{T_u}\left[\alpha_{ci} - \frac{1}{k}\sum_{f=1}^{k}\alpha_{fi}\right] \geq 0 \quad \text{for } c = 1, 2, \dots, k, \tag{35}$$

otherwise, have to consider the second case.

2. $\psi_{ci} > 0$ for at least one c. According to Eq. (24), when $\psi_{ci} > 0$, $u_{ci} = 0$, therefore

$$0 = \frac{1}{k} - \frac{1}{T_u}\left[(\alpha_{ci}+\psi_{ci}) - \frac{1}{k}\sum_{f=1}^{k}(\alpha_{fi}+\psi_{fi})\right]. \tag{36}$$

From the above equation, it can be obtained that

$$\alpha_{ci}+\psi_{ci} = \frac{1}{k}\left[\sum_{f=1}^{k}(\alpha_{fi}+\psi_{fi}) + T_u\right]. \tag{37}$$

Constrained by Eq. (4), it is known that not all u_{ci} can be 0, which means for some c, $u_{ci} > 0$. So the set of clusters is split into two subsets denoted as k^- and $k+$, with

$$k^- = \{c : u_{ci} = 0\},$$

$$k^+ = \{c : u_{ci} > 0\} \neq \emptyset. \tag{38}$$

From the right hand side of Eq. (37), it shows that the value of $(\alpha_{ci}+\psi_{ci})$ is independent on c, which means $\forall c \in k^-$, these values are the same. Therefore Eq. (37) can further be written as

$$\begin{aligned}\alpha_{ci}+\psi_{ci} &= \frac{1}{k}\left[\sum_{f\in k^+}\alpha_{fi} + \sum_{f\in k^-}(\alpha_{fi}+\psi_{fi}) + T_u\right] \\ &= \frac{1}{k}\left[\sum_{f\in k^+}\alpha_{fi} + |k^-|(\alpha_{ci}+\psi_{ci}) + T_u\right]\end{aligned}$$

$$\Rightarrow \alpha_{ci}+\psi_{ci} = \frac{1}{|k^+|}\left(\sum_{f\in k^+}\alpha_{fi} + T_u\right) \quad \text{for } c \in k^-. \tag{39}$$

With the above relation, for $c \in k^+$, Eq. (32) is rewritten as

$$\begin{aligned}u_{ci} &= \frac{1}{k} - \frac{1}{T_u}\alpha_{ci} + \frac{1}{T_u}\frac{1}{k}\left(\sum_{f\in k^+}\alpha_{fi} + \sum_{f\in k^-}(\alpha_{fi}+\psi_{fi})\right) \\ &= \frac{1}{k} - \frac{1}{T_u}\alpha_{ci} + \frac{1}{T_u}\frac{1}{k}\sum_{f\in k^+}\alpha_{fi} + \frac{1}{T_u}\frac{1}{k}\frac{|k^-|}{|k^+|}\left(\sum_{f\in k^+}\alpha_{fi} + T_u\right).\end{aligned} \tag{40}$$

After some transformation and with the relation $1/k(1+|k^-|/|k^+|) = \frac{1}{|k^+|}$, we finally obtain the final form as in Eq. (25).

Follow similar derivation, the update equation of V can also be derived as in Eq. (28).

Appendix B. The proof of convergence

Next we show the algorithm of PFC will converge to a local minimum after finite iterations. We first prove that

Theorem 1. *The value of the objective function J_{PFC} never increase during the alternating updating of U and V, i.e.* $J_{\text{PFC}}(U^{(l+1)}, V^{(l)}) \leq J_{\text{PFC}}(U^{(l)}, V^{(l)})$ *and* $J_{\text{PFC}}(U^{(l)}, V^{(l+1)}) \leq J_{\text{PFC}}(U^{(l)}, V^{(l)})$.

Proof.

$$\nabla^2 L_{\text{PFC}}(U) = \begin{bmatrix} \frac{\partial^2 L_{\text{PFC}}(U)}{\partial u_{11}\partial u_{11}} & \cdots & \frac{\partial^2 L_{\text{PFC}}(U)}{\partial u_{11}\partial u_{KN}} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 L_{\text{PFC}}(U)}{\partial u_{KN}\partial u_{11}} & \cdots & \frac{\partial^2 L_{\text{PFC}}(U)}{\partial u_{KN}\partial u_{KN}} \end{bmatrix} = \begin{bmatrix} T_u & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & T_u \end{bmatrix}. \tag{41}$$

We always set $T_u, T_v > 0$, thus

$$\mathbf{x}^T \nabla^2 L_{\text{PFC}}(U)\mathbf{x} > 0 \quad \text{for any } \mathbf{x} \neq \mathbf{0}, \tag{42}$$

which means $\nabla^2 L_{\text{PFC}}(U)$ is positive define. According to the second order conditions for constrained optimization, for fixed V, the U computed by Eq. (25) is a local minimum of J_{PFC} subject to constraints in Eqs. (4) and (5). Therefore, $J_{\text{PFC}}(U^{(l+1)}, V^{(l)}) \leq J_{\text{PFC}}(U^{(l)}, V^{(l)})$. Similarly, it can be shown that for a fixed U, $J_{\text{PFC}}(U^{(l)}, V^{(l+1)}) \leq J_{\text{PFC}}(U^{(l)}, V^{(l)})$. It follows that the value of J_{PFC} never increase during the alternating updating of U and V. □

According to Theorem 1 and the fact that $J_{\text{PFC}} \geq 0$, it yields that the algorithm of PFC will converge after a finite number of iterations.

References

[1] A.K. Jain, M.N. Murty, P.J. Flynn, Data clustering: a review, ACM Computing Surveys 31 (3) (1999) 264–323.

[2] J. Han, M. Kamber, J. Pei, Data Mining: Concepts and Techniques, second ed., Morgan Kaufmann, California, 2005.

[3] P. Berkhin, Survey of clustering data mining techniques, Technical Report, Accrue Software, Inc., 2002.

[4] R. Xu, D. Wunsch, Survey of clustering algorithms, IEEE Transactions on Neural Networks 16 (3) (2005) 645–678.

[5] M. Filippone, F. Camastra, F. Masulli, S. Rovetta, A survey of kernel and spectral methods for clustering, Pattern Recognition 41 (2008) 176–190.

[6] R.J. Hathaway, J.W. Devenport, J.C. Bezdek, Relational duals of the c-means clustering algorithms, Pattern Recognition 22 (1988) 205–212.

[7] P.H.A. Sneath, R.R. Sokal, Numerical Taxonomy—The Principles and Practice of Numerical Classification, W. H. Freeman, San Francisco, 1973.

[8] L. Kaufman, P.J. Rousseeuw, Finding Groups in Data: An Introduction to Cluster Analysis, Wiley, New York, 1990.

[9] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8) (2000) 888–905.

[10] B. Schölkopf, A.J. Smola, K.R. Müller, Non-linear component analysis as a kernel eigenvalue problem, Neural Computing 10 (1998) 1299–1319.

[11] T. Mitchell, Machine Learning, McGraw-Hill, New York, 1997.

[12] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Plenum, New York, 1981.

[13] N.R. Pal, K. Pal, J.M. Keller, J.C. Bezdek, A possibilistic fuzzy c-means clustering algorithm, IEEE Transactions on Fuzzy Systems 13 (2005) 508–516.

[14] K. Kummamuru, A. Dhawale, R. Krishnapuram, Fuzzy co-clustering of documents and keywords, in: Proceedings of the 12th IEEE International Conference on Fuzzy Systems, 2003, pp. 772–777.

[15] C.-H. Oh, K. Honda, H. Ichihashi, Fuzzy clustering for categorical multivariate data, in: Proceedings of the Joint Ninth IFSA World Congress and 20th NAFIPS International Conference, 2001, pp. 2154–2159.

[16] K. Honda, H. Ichihashi, F. Masulli, S. Rovetta, Linear fuzzy clustering with selection of variables using graded possibilistic approach, IEEE Transactions on Fuzzy Systems 15 (2007) 878–889.
[17] W.-C. Tjhi, L. Chen, Dual fuzzy-possibilistic coclustering for categorization of documents, IEEE Transactions on Fuzzy Systems 17 (2009) 532–543.
[18] R. Krishnapuram, A. Joshi, O. Nasraoui, L. Yi, Low-complexity fuzzy relational clustering algorithms for web mining, IEEE Transactions on Fuzzy Systems 9 (2001) 595–607.
[19] R.T. Ng, J. Han, CLARANS: a method for clustering objects for spatial data mining, IEEE Transactions on Knowledge-based Data Engineering 14 (2002) 1003–1016.
[20] R.J. Hathaway, J.C. Bezdek, NERF c-means: non-Euclidean relational fuzzy clustering, Pattern Recognition 27 (1994) 429–437.
[21] R.N. Davé, S. Sen, Robust fuzzy clustering of relational data, IEEE Transactions on Fuzzy Systems 10 (2002) 713–727.
[22] J.C. Bezdek, R.J. Hathaway, J.M. Huband, C. Leckie, R. Kotagiri, Approximate clustering in very large relational data, International Journal of Intelligent Systems 21 (2006) 817–841.
[23] L. Wang, J.C. Bezdek, C. Leckie, R. Kotagiri, Selective sampling for approximate clustering of very large data sets, International Journal of Intelligence Systems 23 (2008) 313–331.
[24] H. Frigui, C. Hwang, F.C.-H. Rhee, Clustering and aggregation of relational data with application to image database categorization, Pattern Recognition 40 (2007) 3053–3068.
[25] M.P. Windham, Numerical classification of proximity data with assignment measures, Journal of Classification 2 (1985) 157–172.
[26] J.C. Bezdek, R.J. Hathaway, M.P. Windham, Numerical comparison of the RFCM and AP algorithm for clustering relational data, Pattern Recognition 24 (1991) 783–791.
[27] S. Miyamoto, K. Umayahara, Fuzzy clustering by quadratic regularization, in: Fuzzy Systems Proceedings, IEEE World Congress on Computational Intelligence, 1998.
[28] R.-P. Li, M. Mukaidono, A maximum-entropy approach to fuzzy clustering, in: Proceedings of the Fourth IEEE International Conference on Fuzzy Systems, 1995.
[29] N.R. Pal, K. Pal, J.C. Bezdek, A mixed c-means clustering model, in: Proceedings of the Sixth IEEE International Conference on Fuzzy Systems, 1997.
[30] University of California at Irving, Machine Learning Repository ⟨http://www.ics.uci.edu/ ~ mlearn/MLRepository.html⟩.

About the Author—JIAN-PING MEI received her B.Eng. degree from School of Electronic and Information Engineering at Ningbo University, China, in 2005 and M.Eng. Degree from School of Information Science and Electronic Engineering at Zhejiang University, China, in 2007. She is currently working towards Ph.D. degree at Nanyang Technological University in Singapore. Her research interests include the effectiveness and interpretability aspects of clustering and applications to Web Documents.

About the Author—LIHUI CHEN holds the B.Eng. degree in Computer Science and Engineering (Zhejiang University, China) and the Ph.D. degree in Computational Science (University of St. Andrews, UK). She is currently an Associate Professor in the Division of Information Engineering at Nanyang Technological University (NTU) in Singapore. Her research interests include machine learning algorithms and applications, data mining and web intelligence.