Pattern Recognition 47 (2014) 3920–3930

Contents lists available at ScienceDirect

Pattern Recognition

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

Improvements to the relational fuzzy *c*-means clustering algorithm

Mohammed A. Khalilia [a,*], James Bezdek [b], Mihail Popescu [c], James M. Keller [b]

[a] *Computer Science Department, University of Missouri, Columbia, MO 65211, USA*
[b] *Electrical and Computer Engineering Department, University of Missouri, Columbia, MO 65211, USA*
[c] *Health Management and Informatics Department, University of Missouri, Columbia, MO 65212, USA*

ARTICLE INFO

Article history:
Received 30 December 2013
Received in revised form
20 June 2014
Accepted 23 June 2014
Available online 8 July 2014

Keywords:
Fuzzy clustering
Relational *c*-means
Euclidean distance matrices

ABSTRACT

Relational fuzzy *c*-means (RFCM) is an algorithm for clustering objects represented in a pairwise dissimilarity values in a dissimilarity data matrix D. RFCM is dual to the fuzzy *c*-means (FCM) object data algorithm when D is a Euclidean matrix. When D is not Euclidean, RFCM can fail to execute if it encounters negative relational distances. To overcome this problem we can Euclideanize the relation D prior to clustering. There are different ways to Euclideanize D such as the β-spread transformation. In this article we compare five methods for Euclideanizing D to $\tilde{D}$. The quality of $\tilde{D}$ for our purpose is judged by the ability of RFCM to discover the apparent cluster structure of the objects underlying the data matrix D. The subdominant ultrametric transformation is a clear winner, producing much better partitions of $\tilde{D}$ than the other four methods. This leads to a new algorithm which we call the improved RFCM (iRFCM).

1. Introduction

Consider a set of objects $O=\{o_1,\cdots,o_n\}$, where the goal is to group them into c natural groups. Objects can be described by feature vectors $X=\{x_1,\ldots,x_n\}\in\mathbb{R}^p$ such that x_i is an attribute vector of dimension p representing object o_i. Alternatively, objects can be represented using a pairwise relationship. The relationships are stored in a relational matrix R, where $R=[r_{ij}]$ measures the relationship between o_i and o_j. If R is a dissimilarity relation denoted by $D=[d_{ij}]$, then it must satisfy the following three conditions:

$$d_{ii}=0 \quad \text{for } i=1,\cdots,n; \tag{1a}$$

$$d_{ij}\geq 0 \quad \text{for } i=1,\cdots,n \text{ and } j=1,\cdots,n; \text{ and} \tag{1b}$$

$$d_{ij}=d_{ji} \quad \text{for } i=1,\cdots,n \text{ and } j=1,\cdots,n, \tag{1c}$$

where condition (1a) is self-dissimilarity, (1b) is non-negativity and (1c) is symmetry. A well-known relational clustering algorithm that is suitable for clustering objects described by D is the relational fuzzy *c*-means (RFCM) proposed in [1] (Algorithm 1). RFCM, the relational dual of the FCM algorithm, takes an input dissimilarity matrix D and outputs a fuzzy partition matrix $U\in M_{fcn}$, where

$$M_{fcn}=\left\{U\in\mathbb{R}^{c\times n}\,|\,u_{ik}\in[0,1],\ \sum_{k=1}^{n}u_{ik}>0,\ \sum_{i=1}^{c}u_{ik}=1,\ \forall\ 1\leq i\leq c \text{ and } 1\leq k\leq n\right\} \tag{2}$$

Algorithm 1. Relational fuzzy *c*-means (RFCM) [1]

```
Input: $D$, $c$, fuzzifier $m>1$ (default $m=2$), $t_{max}$ (default $t_{max}=100$), $\varepsilon$ (default $\varepsilon=0.0001$)
Output: $U$, $V_R$
Initialize: step=$\varepsilon$, $t$=1
        Relational cluster centers $V_R^0=(v_{R,1}^0,v_{R,2}^0,\cdots,v_{R,c}^0)$, $v_{R,i}^0\in\mathbb{R}^n$
        Note: we use $c$ randomly chosen rows of $D$ as initial centers.
while $t\leq t_{\max}$ and step $\geq\varepsilon$
    $d_{R,ik}=(Dv_{R,i}^{t-1})_k-\frac{1}{2}(v_{R,i}^{t-1})^T Dv_{R,i}^{t-1}$   for $1\leq i\leq c$ and $1\leq k\leq n$   (3)
for $k$=1 to $n$
    if $d_{R,ik}\neq 0$ for all $i$
$u_{ik}=1/\left(\frac{d_{R,ik}}{\sum_{j=1}^{c}d_{R,ik}}\right)^{1/m-1}$ ; $\forall i$   (4)
else
    Set $u_{ik}>0$ for $d_{R,ik}=0$, $u_{ik}\in[0,\ 1]$ and $\sum_{j=1}^{c}u_{jk}=1$
    endif
```

[*] Corresponding author.
E-mail address: mohammed.khalilia@gmail.com (M.A. Khalilia).

http://dx.doi.org/10.1016/j.patcog.2014.06.021
0031-3203/© 2014 Elsevier Ltd. All rights reserved.

```
13 endfor
14 $v_{R,i}^t = (u_{i1}^m, \cdots, u_{in}^m) / \sum_{k=1}^{n} u_{ik}^m \quad \text{for } 1 \le i \le c \qquad (5)$
15     step $\leftarrow \max_{\substack{1 \le i \le c \\ 1 \le j \le n}} \{|V_R^{(t)} - V_R^{(t-1)}|\}$
16     $t \leftarrow t+1$
17 endwhile
```

The duality relationship between RFCM and FCM is based on the squared Euclidean distance or 2-norm that defines the dissimilarity d_{ij} between two feature vectors x_i and x_j describing o_i and o_j and the dissimilarity between the cluster center v_i and o_j. In other words, RFCM assumes that

$$D = [d_{ij}] = [||x_i - x_j||_2^2] \qquad (6)$$

The relation $D = [d_{ij}]$ is Euclidean if there exists feature vectors $X = \{x_1, \ldots, x_n\} \in \mathbb{R}^p$ with an embedding dimension $p < n$, such that for all i,j $d_{ij} = ||x_i - x_j||_2^2$. When D is Euclidean, it has a realization in some Euclidean space. In this case, RFCM and FCM will produce the same partition of relational and feature vector representation of the data. If D is not Euclidean, RFCM will still find clusters in any D whose entries satisfy (1) as long as it can execute, but in this case it is possible for RFCM to experience an execution failure. This happens when the relational distances between prototypes and objects $d_{R,ik}$ in Eq. (3) become negative for some i and k (Algorithm 1, line 6). Another important observation about RFCM is that it expects *squared* dissimilarities D. If the dissimilarities are not squared, meaning that we have $\sqrt{D}$ instead of D such that $\sqrt{D} = D^{1/2} = [\sqrt{d_{ij}}]$, then the dissimilarities must be squared before clustering using RFCM so that D is the Hadamard product $D = (\sqrt{D})^2$. Throughout this paper D is assumed to contain *squared* dissimilarities.

Non-Euclidean Relational Fuzzy c-Means (NERFCM), repairs RFCM "on the fly" with a self-healing property that automatically adjusts the values of $d_{R,ik}$ and the dissimilarities in D in case of failure [2]. The self-healing property is based on the β-spread, which works by adding a positive constant β to the off-diagonal elements of D. In fact, there exists β_0 such that the β-spread transformed matrix D_β is Euclidean for all $\beta \ge \beta_0$. The parameter β controls the amount spreading and must be as small as possible to minimize unnecessary dilation that distorts the original D, which in turn may result in the loss of cluster information. The exact value of β_0 is the largest positive eigenvalue of the matrix PDP, where $P = I - (1/n)(11^T)$ and I is $n \times n$ identity matrix. Eigenvalue computation is avoided by the self-healing module, which is invoked during execution only when needed. When activated, this module adjusts the current D by adding a minimal β-spread to its all off-diagonal elements.

An alternative to using NERFCM is to transform the matrix D by a mapping that converts it to Euclidean form (we call this operation "Euclideanizing D"), and then running RFCM on the Euclideanized matrix $\tilde{D}$. This approach guarantees that RFCM will not fail since $\tilde{D}$ is already Euclidean. There are at least five ways to Euclideanize D, including the β-spread transformation. In addition to the β-spread transformation, this paper will study the other four Euclideanization approaches indicated under option 1 in Fig. 1. As a result of this study, we will append an "i" (short for the word "improved") to RFCM, but not to NERFCM, which is NOT altered by these results. We hope to write a companion paper to this one that discusses improvements to NERFCM which would then become iNERFCM, but attempts to find an alternative to the current "self-healing" method described in [2] which is NERFCM have so far met stiff resistance.

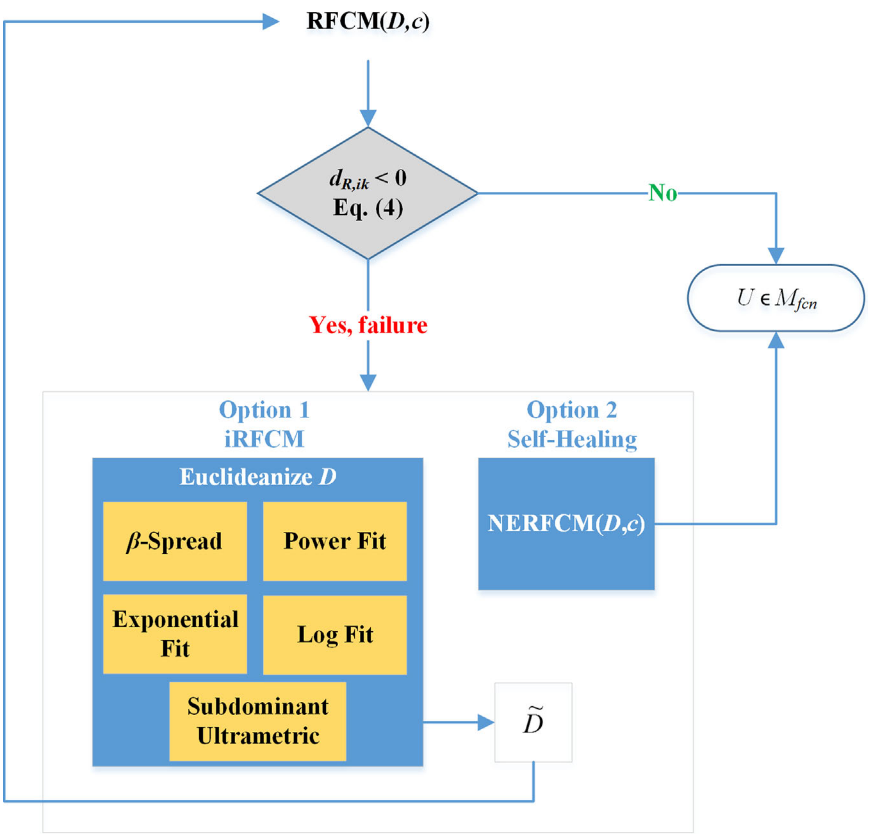

Fig. 1. Possible solutions RFCM can utilize when input D is non-Euclidean.