Dealing with Non-Metric Dissimilarities in Fuzzy Central Clustering Algorithms

Maurizio Filippone[1]

Department of Computer Science, University of Sheffield,
Regent Court, 211 Portobello Street, Sheffield, S1 4DP. United Kingdom

Abstract

Clustering is the problem of grouping objects on the basis of a similarity measure among them. Relational clustering methods can be employed when a feature-based representation of the objects is not available, and their description is given in terms of pairwise (dis)similarities. This paper focuses on the relational duals of fuzzy central clustering algorithms, and their application in situations when patterns are represented by means of non-metric pairwise dissimilarities. Symmetrization and shift operations have been proposed to transform the dissimilarities among patterns from non-metric to metric. In this paper, we analyze how four popular fuzzy central clustering algorithms are affected by such transformations. The main contributions include the lack of invariance to shift operations, as well as the invariance to symmetrization. Moreover, we highlight the connections between relational duals of central clustering algorithms and central clustering algorithms in kernel-induced spaces. One among the presented algorithms has never been proposed for non-metric relational clustering, and turns out to be very robust to shift operations.

Key words: fuzzy clustering, relational clustering, kernel clustering methods

1 Introduction

Clustering is the problem of grouping objects on the basis of a similarity measure among them. It occurs very often in different disciplines and research areas; this is the reason why several approaches have been proposed. In some clustering applications, it is not possible to have a feature-based representation

Email address: `filippone@dcs.shef.ac.uk` (Maurizio Filippone).

[1] The author wishes to express deep gratitude to professors Carlotta Domeniconi, Daniel Barbará and Zoran Duric for their support and influence.

of the objects, and the description is given in terms of pairwise (dis)similarities. Some approaches have been proposed to cluster objects represented in this way, and are referred to as *relational clustering methods*.

Popular crisp relational clustering algorithms form hierarchical structures agglomerating patterns on the basis of the given dissimilarities; they are the so called Sequential Agglomerative Hierarchical Non-Overlapping SAHN approaches [37,18,39]. The result is a hierarchical structure of groups known as *dendrogram*. Other approaches to the relational clustering are the Partitions Around Medoids (PAM) method [19], Clustering LARge Applications (CLARA) [19], and Clustering Large Applications based upon RANdomized Search (CLARANS) [29]. A relational clustering algorithm, called EVCLUS, has been developed in the framework of belief functions [7]. Some fuzzy relational clustering algorithms can be found in literature, for instance those proposed by Ruspini [33], Diday [8], Roubens [32], the Relational Fuzzy c-means (RFCM) [14], the Relational Possibilistic c-means (RPCM) [6], Fuzzy Analysis (FANNY) [19], and the Windham association prototypes [40]. In fuzzy clustering, a pattern can belong to more than one cluster with different degrees. This allows to obtain a more sound description of the clusters in situations where some patterns can belong to more than one cluster, or some patterns do not belong to any cluster, since they are outliers. All these scenarios can be efficiently handled by means of the generalization of the concept of membership from crisp to fuzzy.

RFCM is based on the optimization of a proper objective function similar to that of Fuzzy c-means (FCM) [4]. Also the optimization procedure follows the scheme used by FCM. In fact, RFCM turns out to be the relational dual of the FCM; in other words, the RFCM with the squared Euclidean distances as dissimilarities, gives the FCM. This duality can be found between the RPCM [6] and the Possibilistic c-means (PCM) [21] too. In general, the central clustering algorithms are based on the concept of memberships and centroids, and are asked to find the clusters in the input space that is usually Euclidean. In the dual versions, since the patterns are not described in terms of features, the concept of centroid as a weighted mean of the patterns looses its meaning. Moreover, if the dissimilarities are not metric, the convergence of the algorithms is not guaranteed [13,30,31]. This problem arises mainly because the distances between patterns and centroids can assume negative values, thus leading to numerical problems. For this reason, some solutions have been proposed. In Ref. [20], the authors propose a fuzzy relational algorithm that selects the centroids among the objects composing the data set. A fuzzy clustering dealing with non-Euclidean dissimilarities is the Non-Euclidean Relational Fuzzy c-means (NERF c-means) [13]. FANNY optimizes the same objective function as RFCM with $m = 2$, but employing the Lagrange multiplier technique; this gives an elegant way to handle non-metric dissimilarities.

Another approach proposes to transform the dissimilarities among patterns from non-metric to metric [30,13] and forms the basis of the modification allowing NERF c-means to deal with non-metric dissimilarities. Non-metric dissimilarities are characterized by the fact that at least one of the following conditions is not met: symmetry and obeying to the triangular inequality. The transformations needed to let them become metric are symmetrization and shift operations. The symmetrization operation makes the dissimilarities symmetric. Shift means that a constant value is added to the pairwise dissimilarities, to let them satisfy the triangular inequality [2] . The point is how these transformations influence the behavior of the clustering algorithms. It has been shown that they do not influence the K-means optimization procedure [30,31], since they change the objective function by a constant. Once the dissimilarities are metric, they can be considered as pairwise squared Euclidean distances between vectors representing the objects. These are called embedding vectors, and are not computed explicitly. This is the link with the theory of central clustering in the space induced by positive semidefinite kernels [10]. Such kernels can be obtained by the dissimilarity matrix, and each entry is a scalar product between vectors representing the original objects. The pairwise scalar products contain enough information to let to apply the central clustering algorithms on the embedding vectors. Popular unsupervised learning algorithms making use of kernels are the Kernel PCA [36], K-Means in feature space [12,36], and One Class SVM [16,17,38]. For a survey on kernel clustering methods see [10].

This paper considers the approaches belonging to the K-means [25,27] family, in particular those based on fuzzy memberships [3,4,21,22]. The literature lacks of an explicit analysis on what happens to central fuzzy clustering algorithms when the dissimilarities are transformed. This paper explicitly shows how the objective functions of four clustering algorithms based on fuzzy memberships change, due to symmetrization and shift operations. The considered clustering algorithms are: Fuzzy c-means I (FCM I) [4], Fuzzy c-means II (FCM II) [3] (also known as soft K-means [26]), Possibilistic c-means I (PCM I) [21], and Possibilistic c-means II (PCM II) [22]. The main contributions include the lack of invariance to shift operations, as well as the invariance to symmetrization. As a byproduct, the kernel versions of FCM I, FCM II, PCM I and PCM II are obtained, that can be viewed as relational duals of the four algorithms. FCM I and PCM II in feature space have been proposed in Refs. [41] and [11]. The relational duals of FCM I and PCM I have been proposed in Refs. [14] and [6]; the non-Euclidean case is studied in Ref. [13] for FCM I. To the best of our knowledge, FCM II and PCM I in feature space have never been proposed so far, as well as the non-Euclidean relational dual of FCM II and PCM II; this represent another novelty of this paper.

[2] In fact, we require the stronger condition that the dissimilarities become squared Euclidean distances.

The relational dual of FCM II, in particular, turns out to be very robust to shift operations. For the sake of presentation, however, we prefer to show a general formulation of the relational duals of central fuzzy clustering algorithms, introducing the four fuzzy clustering algorithms as special cases.

In the experimental tests on synthetic data sets, we analyze the behavior of the presented algorithms during and at the end of the optimization. We also study if there is a chance to cope with the effect of the shift, by tuning the parameters, by using a score based on the Kullback-Leibler divergence. On one of the two synthetic data sets, we study the performances in terms of correct assignments of cluster labels, when adding noise to the relational matrix; this could simulate a real scenario, where the measures of relations between pairs of patterns are noisy. The experimental part ends showing the performances of the algorithms in a real application.

This Section discusses how to embed in Euclidean spaces sets of patterns described by pairwise dissimilarities, along with some basic concepts on positive semidefinite kernels. Then the paper is organized as follows: Section 2 shows how the objective functions of four fuzzy central clustering algorithms change, due to distance transformations; Section 3 provides an experimental analysis on synthetic and real data sets, and then the conclusions are drawn. Many technical details concerning the derivations of the proposed algorithms and theoretical aspects can be found in the appendices.

1.1 Embedding Objects Described by Pairwise Dissimilarities in Euclidean Spaces

Let $Y = \{y_1, \ldots, y_n\}$ be a set of objects and $r : Y \times Y \to \mathbb{R}$ a function between pairs of its elements. The conditions that r must satisfy to be a distance are:

- $r(y_i, y_j) \geq 0 \quad \forall i, j = 1, \ldots, n$ and $r(y_i, y_i) = 0 \quad \forall i = 1, \ldots, n$ (Positivity);
- $r(y_i, y_j) = r(y_j, y_i) \quad \forall i, j = 1, \ldots, n$ (Symmetry) ;
- $r(y_i, y_j) + r(y_j, y_k) \geq r(y_i, y_k) \quad \forall i, j, k = 1, \ldots, n$ (Triangular inequality).

Let's assume that r satisfies only the first condition. In this case, r can be interpreted as a dissimilarity measure between the elements of the set Y. Clearly, it is not possible to embed the objects according to r in a Euclidean space, as long as it does not satisfy also the other two conditions. The only way to cope with this problem is to apply some transformations to let r become a distance function. Regarding the symmetry, the following, for instance, could represent a solution:

$$\hat{r}(y_i, y_j) = \max(r(y_i, y_j), r(y_j, y_i)) \qquad \forall i, j \tag{1}$$

or:

$$\hat{r}(y_i, y_j) = \frac{1}{2}(r(y_i, y_j) + r(y_j, y_i)) \qquad \forall i, j \tag{2}$$

Depending on the application, one can choose the most suitable solution to fix the symmetry.

Once the symmetry is fixed, to make r satisfy the triangular inequality, a constant shift 2α can be added to all the pairwise distances, excluding the dissimilarity between a pattern and itself:

$$\tilde{r}(y_i, y_j) = r(y_i, y_j) + 2\alpha \qquad \forall i \neq j \tag{3}$$

Let's introduce R as the $n \times n$ matrix with entries $r_{ij} = r(y_i, y_j)$. Let $e = \{1, 1, \ldots, 1\}^{\mathrm{T}}$ and I the $n \times n$ identity matrix. Eq. 3 is equivalent to:

$$\tilde{R} = R + 2\alpha(ee^{\mathrm{T}} - I) \tag{4}$$

The natural question arises: how can we choose α to guarantee that $\tilde{R}$ is a squared Euclidean distance matrix? The answer is in a theorem that can be found in Refs. [23,31]. In this Section the theorem is reported, while the proof can be found in App. B.

Before showing the theorem, some preliminary definitions are needed. Let's decompose R by means of a matrix S:

$$r_{ij} = s_{ii} + s_{jj} - 2s_{ij} \tag{5}$$

Let $Q = I - \frac{1}{n}ee^{\mathrm{T}}$. The centralized version P^c of a generic matrix P is defined as:

$$P^c = QPQ \tag{6}$$

It's clear from Eq. 5 that S is not uniquely determined by R. All the matrices $S + \alpha ee^{\mathrm{T}}$, for instance, lead to the same matrix R $\forall \alpha \in \mathbb{R}$. It can be proved, however, that the centralized version of S is uniquely determined by R (see App. A):

$$S^c = -\frac{R^c}{2} \tag{7}$$

Now we have all the elements to claim that:

Theorem 1.1 *R is a squared Euclidean distance matrix if and only if $S^c \succeq 0$.*

The proof can be found in App. B or in Refs. [23,31]. The theorem states that S^c must be positive semidefinite to ensure that R is a squared Euclidean distance matrix. It is well known that the eigenvalues λ_i of positive semidefinite matrices satisfy $\lambda_i \geq 0 \quad \forall i = 1, \ldots, n$ [1]. If at least one eigenvalue of S^c is negative, R is not a squared Euclidean distance matrix. Let λ_1 be the smallest eigenvalue of S^c. Simple concepts of linear algebra ensure that the following diagonal shift to S^c:

$$\tilde{S^c} = S^c - \lambda_1 I \tag{8}$$

makes $\tilde{S^c}$ positive semidefinite. The diagonal shift of S^c transforms R in a matrix representing squared Euclidean distances. The resulting transformation on R is the following:

$$\tilde{R} = R - 2\lambda_1(ee^{\mathrm{T}} - I) \tag{9}$$

Since $\tilde{S^c}$ is positive semidefinite, it can be thought as representing a scalar product. Thus, it exists a matrix X for which:

$$\tilde{S^c} = XX^{\mathrm{T}} \tag{10}$$

The rows of X are the realization of the embedding vectors $\mathbf{x}_i$. In other words each element y_i of the set Y has been embedded in a Euclidean space and is represented by $\mathbf{x}_i$. The entries of $\tilde{S^c}$ are the scalar product between the vectors $\mathbf{x}_i$.

Resuming, if the only thing known about the data to analyze are the pairwise dissimilarities, the matrix S^c can be checked for positive semidefiniteness. If it is, S^c can be kept as is, otherwise the diagonal shift to S^c has to be applied. Either way, S^c or $\tilde{S^c}$ is the product of two unknown matrices X. This is the link between the theory of embedding a set of objects in Euclidean spaces and the theory of kernel methods. $\tilde{S^c}$ can be interpreted as the Gram matrix that is used in kernel algorithms. In Ref. [23,24] the authors give an interpretation of the negative eigenvalues of S^c.

1.2 Mercer Kernels

A kernel function $K : X \times X \to \mathbb{R}$ is called a *positive definite kernel* (or *Mercer kernel*) if and only if K is symmetric and positive semidefinite [2,34].

Each Mercer kernel can be expressed as follows:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i)^{\mathrm{T}} \Phi(\mathbf{x}_j) \tag{11}$$

where $\Phi : X \rightarrow \mathcal{F}$ performs a mapping from the input space X to $\mathcal{F}$ which is called *feature space*. In order to simplify the notation, we introduce $k_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$. The elements k_{ij} are the entries of the Gram matrix containing the kernel function evaluated for all the pairs of objects belonging to X. It is worth noting that the choice of K induces an implicit map Φ, that can be unknown in general. Despite that, a well known result shows that it is not necessary to know Φ to compute the distances in feature space:

$$\begin{aligned} \|\Phi(\mathbf{x}_i) - \Phi(\mathbf{x}_j)\|^2 &= (\Phi(\mathbf{x}_i) - \Phi(\mathbf{x}_j))^{\mathrm{T}} (\Phi(\mathbf{x}_i) - \Phi(\mathbf{x}_j)) \\ &= k_{ii} + k_{jj} - 2k_{ij} \end{aligned} \tag{12}$$

This is the so called *distance kernel trick* [28,36].

Kernels have been used in many supervised and unsupervised algorithms. In fact, every algorithm where input vectors appear only in dot products with other input vectors can be kernelized [35]. In Support Vector Machines [5], one takes advantage of this mapping to solve a classification problem in a high dimensional feature spaces. In clustering methods, the goal is to identify groups in data; the kernel function, that implicitly map the input space into another space, should be chosen in such a way so as to highlight such structures.

From the previous analysis, we know that starting from the pairwise dissimilarities between patterns, it is possible to construct the matrix $\tilde{S}^c$ having all the properties of Mercer kernels K. Here the dissimilarities in R imply $K = \tilde{S}^c$, that implies Φ. The next Section shows how it is possible to obtain a formulation of the central clustering algorithms, knowing just K. Since K induces an implicit map Φ, it will not be possible to know the prototypes of the clusters, that will be points in the space $\mathcal{F}$.

1.3 Pre-Shift and Post-Shift

Before closing this Section, it is worth noting that in general there are two options when shifting R to obtain $\tilde{S}^c$. The first is to shift the dissimilarities R obtaining $\tilde{R}$, and then compute $\tilde{S}^c$ associated to $\tilde{R}$. Let's call this procedure *pre-shift*:

$$\tilde{S}^c = -\frac{1}{2}(Q\tilde{R}Q) \tag{13}$$

The second choice, the *post-shift*, is to compute S^c associated to R, and then shift its diagonal elements:

$$S^c + \alpha I \tag{14}$$

Both the methods allow to compute a matrix corresponding to the same shift, but:

$$S^c + \alpha I \neq -\frac{1}{2}(Q\tilde{R}Q) \tag{15}$$

App. C shows that the choice between pre-shift and post-shift does not affect the studied clustering algorithms.

2 Central Clustering Algorithms Objective Functions

The central clustering algorithms are based on the concept of centroids and memberships. In this family, we can find the fuzzy versions of the K-means with the probabilistic and possibilistic description of the memberships: Fuzzy c-means [4] and Possibilistic c-means [21]. Given a set of patterns X, the set of centroids $V = \{\mathbf{v}_1, \dots, \mathbf{v}_c\}$ and the membership matrix U are defined. The set V contains the prototypes/representatives of the c clusters. The element $\mathbf{v}_i$ are also referred to as codevectors or centroids. U is a $c \times n$ matrix having entries u_{ih} representing the membership of the pattern h to the cluster i. Both Fuzzy and Possibilistic c-means are fuzzy, since $u_{ih} \in [0, 1]$ while $u_{ih} \in \{0, 1\}$ for K-means. In K-means and FCM algorithms the memberships of a pattern to all the c clusters are constrained to sum up to one:

$$\sum_{i=1}^{c} u_{ih} = 1 \quad \forall k = 1, \dots, n \tag{16}$$

This is the so called *probabilistic constraint*, that is relaxed in the possibilistic paradigm; in the latter case, the memberships can be interpreted as a degree of typicality.

In general, the clustering solution is obtained by minimizing a functional composed by two terms:

$$J(U, V) = G(U, V) + H(U) \tag{17}$$

The first term is a measure of the distortion and the second is an entropic score on the memberships. The distortion can be written as the following sum:

$$G(U,V) = 2\sum_{i=1}^{c}\sum_{h=1}^{n} u_{ih}^{\theta}\|\mathbf{x}_h - \mathbf{v}_i\|^2 \tag{18}$$

with $\theta \geq 1$. The aim of the entropy term $H(U)$ is to avoid trivial solutions where all the memberships are zero or equally shared among the clusters.

For the algorithms having a constraint on U, the Lagrange multipliers technique has to be followed in order to perform the optimization. This means that a further term, depending only on U, must be added to $J(U,V)$. The Lagrangian associated to the optimization problem reads:

$$L(U,V) = G(U,V) + H(U) + W(U) \tag{19}$$

The technique used by these methods to perform the minimization is the so called Picard iteration technique. The Lagrangian $L(U,V)$ depends on two groups of variables U and V related to each other, namely $U = U(V)$ and $V = V(U)$. In each iteration one of the two groups of variables is kept fixed, and the minimization is performed with respect to the other group. In other words:

$$\frac{\partial L(U,V)}{\partial \mathbf{v}_i} = 0 \tag{20}$$

with U fixed gives a formula for the update of the centroids $\mathbf{v}_i$, and:

$$\frac{\partial L(U,V)}{\partial u_{ih}} = 0 \tag{21}$$

with V fixed gives a formula for the update of the memberships u_{ih}. The algorithms start by randomly choosing U or V, and iteratively update U and V by means of the previous two equations. It can be proved that the value of L does not increase after each iteration [15]. The algorithms stop when a convergence criterion is satisfied on the U, V or G; usually the following is considered:

$$\|U - U'\|_p < \varepsilon \tag{22}$$

where U' is the updated version of the memberships and $\|\|_p$ is a p-norm.

Since $L(U, V)$ depends on V only because of $G(U, V)$, the update of the $\mathbf{v}_i$ is the same for all the considered algorithms. From Eq. 20:

$$\mathbf{v}_i = \frac{\sum_{h=1}^{n} u_{ih}^{\theta} \mathbf{x}_h}{\sum_{h=1}^{n} u_{ih}^{\theta}} \tag{23}$$

By substituting Eq. 23 into Eq. 18, it is easy to verify that the following functional is equivalent to $G(U, V)$:

$$G(U) = \sum_{i=1}^{c} \frac{\displaystyle\sum_{r=1}^{n} \sum_{s=1}^{n} u_{ir}^{\theta} u_{is}^{\theta} d_{rs}^{2}}{\displaystyle\sum_{r=1}^{n} u_{ir}^{\theta}} \tag{24}$$

Here d_{rs}^2 is the squared Euclidean distance between patterns r and s. This allows to write the objective function only in terms of U, when the description of the data set is in terms of pairwise distances.

In the non-metric case, it is not possible to identify d_{rs}^2 as the squared Euclidean distance between patterns r and s. Anyway, it is still possible to think that the objective function of the clustering is:

$$G(U) = \sum_{i=1}^{c} \frac{\displaystyle\sum_{h=1}^{n} \sum_{k=1}^{n} u_{ih}^{\theta} u_{ik}^{\theta} r_{hk}}{\displaystyle\sum_{h=1}^{n} u_{ih}^{\theta}} \tag{25}$$

In the following, this way of writing $G(U)$ will be useful to show how the objective functions change with respect to dissimilarities transformations.

2.1 Analysis of Four Clustering Algorithms

In this Section, we analyze four central clustering algorithms based on fuzzy memberships: Fuzzy c-means I (FCM I) [4], Fuzzy c-means II (FCM II) [3], Possibilistic c-means I (PCM I) [21], and Possibilistic c-means II (PCM II) [22] (see App. D for the complete derivation of these four algorithms). Tab. 1 resumes the terms of the Lagrangian in Eq. 19 for the mentioned clustering algorithms. Since the sum of the memberships of a point to all the clusters is constrained to be one in fuzzy clustering, the term $W(U) \neq 0$. For the possibilistic algorithms $W(U) = 0$, since that constrain is relaxed. In fact, for these algorithms the minimization of $L(U)$ should be done in the hypercube

Table 1
Resuming table of the entropy functions, θ value, and constraints, for the considered clustering algorithms.

Method	θ	$H(U)$	$W(U)$
FCM I	m	0	$\sum_{h=1}^{n} \beta_h \left(1 - \sum_{i=1}^{c} u_{ih}\right)$
FCM II	1	$\lambda \sum_{i=1}^{c} \sum_{h=1}^{n} u_{ih} \ln(u_{ih})$	$\sum_{h=1}^{n} \beta_h \left(1 - \sum_{i=1}^{c} u_{ih}\right)$
PCM I	m	$\sum_{i=1}^{c} \eta_i \sum_{h=1}^{n} (1 - u_{ih})^m$	0
PCM II	1	$\sum_{i=1}^{c} \eta_i \sum_{h=1}^{n} (u_{ih} \ln(u_{ih}) - u_{ih})$	0

defined by $u_{ih} \in [0, 1] \, \forall i, h$. Since the form assumed by the update equations, however, this constrain is automatically satisfied. In FCM I and PCM I, the exponent of the memberships θ is usually called m, while $\theta = 1$ in FCM II and PCM II.

App. D shows the derivation of the four clustering algorithms in the case of a feature-based representation of the patterns. Now we show how to obtain a clustering solution starting from a relational matrix R. From the analysis in Section 1.1, it is possible to choose α big enough to guarantee that $\tilde{R}$ represents a squared Euclidean distance matrix. This allows to represent each pattern in a Euclidean space $\mathcal{F}$, where the discussed clustering algorithms can be applied. In fact, the positions of the patterns in $\mathcal{F}$ is still encoded in $\tilde{R}$, and thus is unknown. Nevertheless, using the fact that $K = \tilde{S}^c$ contains the scalar products between patterns, an update formula for the memberships can be explicitly found. Each pattern is represented by a vector $\mathbf{x}_i \in \mathcal{F}$ and the set of centroids V is composed by prototypes in $\mathcal{F}$. As an example, let's analyze the update equations of $\mathbf{v}_i$ and u_{ih} for the FCM II:

$$u_{ih} = \frac{\exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2}{\lambda}\right)}{\sum_{j=1}^{c} \exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_j\|^2}{\lambda}\right)} \tag{26}$$

$$\mathbf{v}_i = \frac{\sum_{h=1}^{n} u_{ih} \mathbf{x}_h}{\sum_{h=1}^{n} u_{ih}} \tag{27}$$

Since we don't know explicitly the vectors $\mathbf{x}_i$, it would not be possible to compute $\mathbf{v}_i$ explicitly. Substituting Eq. 27 in Eq. 26, however, we obtain:

$$\|\mathbf{x}_h - \mathbf{v}_i\|^2 = \left\| \mathbf{x}_h - \frac{\sum_{r=1}^{n} u_{ir} \mathbf{x}_r}{\sum_{r=1}^{n} u_{ir}} \right\|^2$$

Table 2
Resuming table of the memberships update equations for the considered clustering algorithms.

FCM I $u_{ih}^{-1} = \sum_{j=1}^{c} \left(\frac{z_h^{(0)} - 2a_i z_{ih}^{(1)} + a_i^2 z_i^{(2)}}{z_h^{(0)} - 2a_j z_{jh}^{(1)} + a_j^2 z_j^{(2)}} \right)^{\frac{1}{m-1}}$
FCM II $u_{ih} = \frac{\exp \left(- \frac{z_h^{(0)} - 2a_i z_{ih}^{(1)} + a_i^2 z_i^{(2)}}{\lambda} \right)}{\sum_{j=1}^{c} \exp \left(- \frac{z_h^{(0)} - 2a_j z_{jh}^{(1)} + a_j^2 z_i^{(2)}}{\lambda} \right)}$
PCM I $u_{ih}^{-1} = \left(\frac{z_h^{(0)} - 2a_i z_{ih}^{(1)} + a_i^2 z_i^{(2)}}{\eta_i} \right)^{\frac{1}{m-1}} + 1$
PCM II $u_{ih} = \exp \left(- \frac{z_h^{(0)} - 2a_i z_{ih}^{(1)} + a_i^2 z_i^{(2)}}{\eta_i} \right)$

$$= k_{hh} - 2 \frac{\sum_{r=1}^{n} u_{ir} k_{rh}}{\sum_{r=1}^{n} u_{ir}} + \frac{\sum_{r=1}^{n} \sum_{s=1}^{n} u_{ir} u_{is} k_{rs}}{\left(\sum_{r=1}^{n} u_{ir}\right)^2} \tag{28}$$

This allows to obtain an update equation for the memberships for the considered clustering algorithms.

To obtain a more convenient way of writing the update equations, let U_θ be the $c \times n$ matrix having u_{ih}^θ as elements, and let:

$$a_i = \left(\sum_{h=1}^{n} u_{ih}^\theta \right)^{-1} \tag{29}$$

$$\mathbf{z}^{(0)} = \mathrm{diag}(K) \tag{30}$$

$$Z^{(1)} = U_\theta K \tag{31}$$

$$\mathbf{z}^{(2)} = \mathrm{diag}(U_\theta K U_\theta^{\mathrm{T}}) \tag{32}$$

Then, Eq. 28 becomes:

$$\|\mathbf{x}_h - \mathbf{v}_i\|^2 = z_h^{(0)} - 2a_i z_{ih}^{(1)} + a_i^2 z_i^{(2)} \tag{33}$$

Tabs. 2 and 3 show the update equations of the memberships and the steps composing the considered clustering algorithms.

Table 3
Pseudocode of the presented clustering algorithms

```
(1) if $R$ is not symmetric, then symmetrize it using Eq. 34;
(2) Compute $S^c$ using Eq. 7;
(3) if $S^c \succeq 0$ then $K = S^c$;
(4) else $K = S^c - \lambda_1 I$;
(5) Initialize parameters: $c$, $m$ (FCM I, PCM I), $\lambda$ (FCM II), $\eta_i$ (PCM I, PCM II);
(6) Initialize $U$;
(7) Update $U$ using the update equation in Tab. 2 corresponding to the chosen method;
(8) if the convergence criterion is not satisfied then go to step 7;
(9) else stop.
```

2.2 Effect of Symmetrization and Shifts on the Lagrangian

Up to now, we have seen how the symmetrization and the shift operation allowed to cope with the non-metricity of the dissimilarity matrix. The crucial aspect that has to be considered is the impact of these transformations on the behavior of the clustering algorithms. In the following, we show the effect of the transformation on the Lagrangian of the central clustering algorithms.

2.3 Invariance of $G(U)$ to Symmetrization of R

Let's analyze what happens to the Lagrangian L when R is transformed in the following way:

$$\hat{r}_{ij} = \frac{r_{ij} + r_{ji}}{2} \tag{34}$$

which is equivalent to:

$$\hat{R} = \frac{R + R^{\mathrm{T}}}{2} \tag{35}$$

It's clear that the only term of the functional affected by the distance transformation is $G(U)$. Showing that:

$$\begin{aligned}\sum_{h=1}^{n}\sum_{k=1}^{n} u_{ih}^{\theta} u_{ik}^{\theta} \hat{r}_{hk} &= \frac{1}{2}\sum_{h=1}^{n}\sum_{k=1}^{n} u_{ih}^{\theta} u_{ik}^{\theta} r_{hk} + \frac{1}{2}\sum_{h=1}^{n}\sum_{k=1}^{n} u_{ih}^{\theta} u_{ik}^{\theta} r_{kh} \\ &= \sum_{h=1}^{n}\sum_{k=1}^{n} u_{ih}^{\theta} u_{ik}^{\theta} r_{hk} \qquad (36)\end{aligned}$$

the invariance of the Lagrangian $L(U)$ to the symmetrization of R is proved. In other words, in presence of a non-symmetric R, the symmetrization in Eq. 34 does not change the clustering objective function. In force of this result, R will be considered symmetric in the rest of this paper.

2.4 Transformation of $G(U)$ to Shift Operations

The shift operation on the dissimilarities reads:

$$\tilde{r}_{hk} = r_{hk} + 2\alpha \quad \forall\, h \neq k \qquad (37)$$

which is equivalent to Eq. 4:

The only term in the Lagrangian $L(U)$ changing due the dissimilarities shift is $G(U)$:

$$\begin{aligned}G_{\alpha}(U) &= \sum_{i=1}^{c} \frac{\sum_{h=1}^{n}\sum_{k=1}^{n} u_{ih}^{\theta} u_{ik}^{\theta} \tilde{r}_{hk}}{\sum_{h=1}^{n} u_{ih}^{\theta}} \\ &= G(U) + 2\alpha \sum_{i=1}^{c}\sum_{h=1}^{n} u_{ih}^{\theta} - 2\alpha \sum_{i=1}^{c} \frac{\sum_{h=1}^{n} u_{ih}^{2\theta}}{\sum_{h=1}^{n} u_{ih}^{\theta}} \qquad (38)\end{aligned}$$

The Lagrangian will result in:

$$L_{\alpha}(U) = G(U) + H(U) + W(U) + 2\alpha\,(A(U) - B(U)) \qquad (39)$$

This result shows that in general the Lagrangian for the central clustering algorithms is not invariant to such transformations. Only for K-means $A(U) - B(U) = n - c$, which means that the K-means objective function is invariant to distance shifts [30,31]. Besides, for fuzzy clustering algorithms for which $\theta = 1$, $A(U)$ reduces to n.

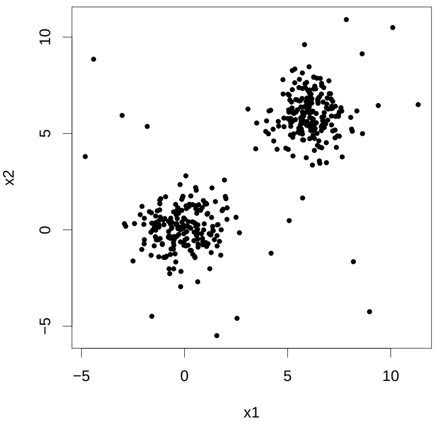

Figure 1. Plot of the synthetic data set composed by two clusters and some outliers.

In general, since $\theta \geq 1$ and $u_{ih} \in [0,1]$, the following two inequalities are satisfied:

$$A(U) = \sum_{i=1}^{c} \sum_{h=1}^{n} u_{ih}^{\theta} < n \tag{40}$$

$$B(U) = \sum_{i=1}^{c} \frac{\sum_{h=1}^{n} u_{ih}^{2\theta}}{\sum_{h=1}^{n} u_{ih}^{\theta}} < c \tag{41}$$

The contributions of $A(U)$ and $B(U)$ to $L_\alpha(U)$ are weighted by 2α. This means that $L_\alpha(U)$ can be strongly affected by large shift values. The next Section provides an experimental analysis showing the effect of the shift operation on the behavior of the presented clustering algorithms.

Tab. 4 resumes the Lagrangian $L_\alpha(U)$ of the discussed clustering algorithms, considering also the effect of the shift. In FCM II and PCM II, $A(U) = n$; in FCM I and PCM I, both $A(U)$ and $B(U)$ are not constant.

3 Experimental Analysis

3.1 Synthetic Data Set 1

The presented clustering algorithms have been tested on a synthetic data set composed by two clusters in two dimensions (Fig. 1). Each cluster is composed by 200 points sampled from a Gaussian distribution. The position of

Table 4
Resuming table of the objective functions, after the shift operation, for the considered clustering algorithms.

FCM I	$\sum_{i=1}^{c} \frac{\sum_{h=1}^{n}\sum_{k=1}^{n} u_{ih}^{m} u_{ik}^{m} r_{hk}}{\sum_{h=1}^{n} u_{ih}^{m}} + \sum_{h=1}^{n} \beta_h \left(1 - \sum_{i=1}^{c} u_{ih}\right) + 2\alpha \sum_{i=1}^{c}\sum_{h=1}^{n} u_{ih}^{m} - 2\alpha \sum_{i=1}^{c} \frac{\sum_{h=1}^{n} u_{ih}^{2m}}{\sum_{h=1}^{n} u_{ih}^{m}}$
FCM II	$\sum_{i=1}^{c} \frac{\sum_{h=1}^{n}\sum_{k=1}^{n} u_{ih} u_{ik} r_{hk}}{\sum_{h=1}^{n} u_{ih}} + \lambda \sum_{h=1}^{n}\sum_{i=1}^{c} u_{ih} \ln(u_{ih}) + \sum_{h=1}^{n} \beta_h \left(1 - \sum_{i=1}^{c} u_{ih}\right) + 2\alpha n - 2\alpha \sum_{i=1}^{c} \frac{\sum_{h=1}^{n} u_{ih}^{2}}{\sum_{h=1}^{n} u_{ih}}$
PCM I	$\sum_{i=1}^{c} \frac{\sum_{h=1}^{n}\sum_{k=1}^{n} u_{ih}^{m} u_{ik}^{m} r_{hk}}{\sum_{h=1}^{n} u_{ih}^{m}} + \sum_{i=1}^{c} \eta_i \sum_{h=1}^{n} (1 - u_{ih})^{m} + 2\alpha \sum_{h=1}^{n}\sum_{i=1}^{c} u_{ih}^{m} - 2\alpha \sum_{i=1}^{c} \frac{\sum_{h=1}^{n} u_{ih}^{2m}}{\sum_{h=1}^{n} u_{ih}^{m}}$
PCM II	$\sum_{i=1}^{c} \frac{\sum_{h=1}^{n}\sum_{k=1}^{n} u_{ih} u_{ik} r_{hk}}{\sum_{h=1}^{n} u_{ih}} + \sum_{i=1}^{c} \eta_i \sum_{h=1}^{n} (u_{ih} \ln(u_{ih}) - u_{ih}) + 2\alpha n - 2\alpha \sum_{i=1}^{c} \frac{\sum_{h=1}^{n} u_{ih}^{2}}{\sum_{h=1}^{n} u_{ih}}$

their center are respectively $(0, 0)$ and $(6, 6)$, and the standard deviations are equal to one for both the features and the clusters. Twenty outliers have been added sampling points in the set $[-6, 12] \times [-6, 12]$ using a uniform probability distribution. The average of the squared distances is 43.4, the median is 34.4, and the maximum is 360.9.

3.1.1 Behavior of the memberships during the optimization

For all the tested algorithms, the behavior of the memberships have been analyzed during the optimization, for different values of α. In order to do that, the elements r_{ij} have been set to the squared Euclidean distance $\|\mathbf{x}_i - \mathbf{x}_j\|^2$, and have been shifted with different values of α. The proposed algorithms have been run on the modified data sets. During the optimization, the memberships have been recorded to see how the distance shifts affected their behavior. At each iteration, the difference between the matrix U when $\alpha = 0$ and U' for an $\alpha \neq 0$ has been measured. The analysis has been made on the basis of the

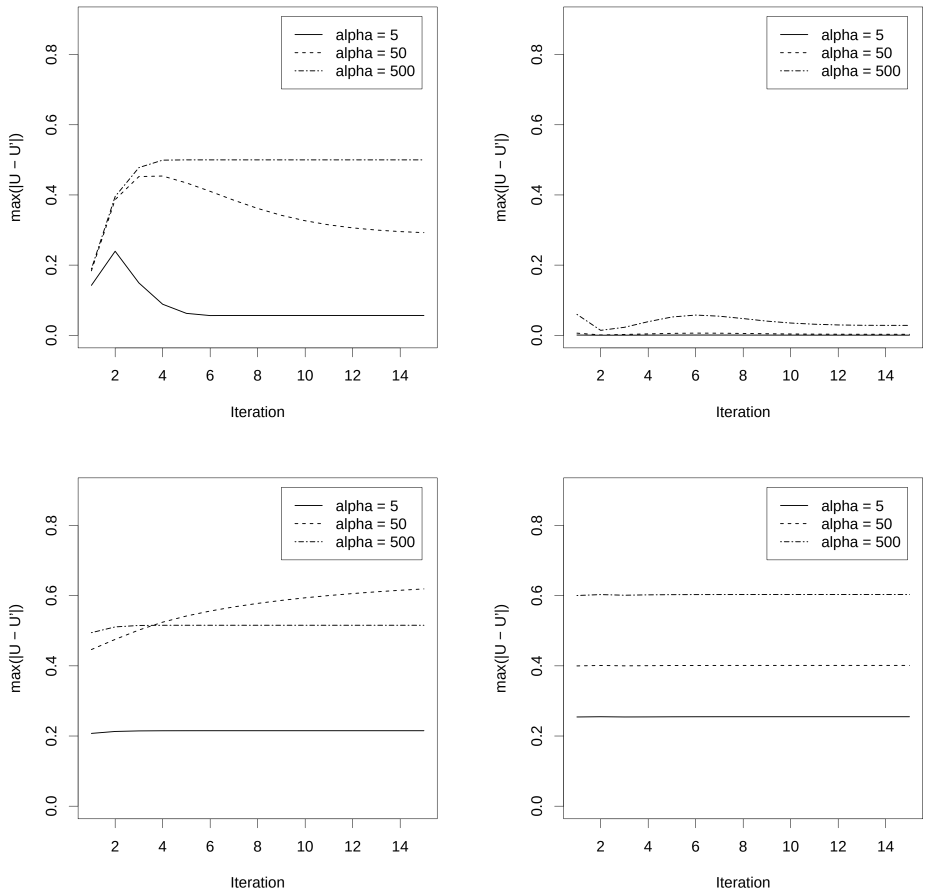

Figure 2. Behavior of the memberships during the optimization for different values of α. First row FCM I $m = 1.5$ and FCM II $\lambda = 20$; second row PCM I $m = 1.5$ and PCM II $\gamma = 0.5$. Results are averaged over 100 repetitions with different initialization of U.

following score:

$$\max(|U - U'|) = \max_{i,h} \left(|u_{ih} - u'_{ih}|\right) \tag{42}$$

averaged over 100 runs.

For the sake of brevity, we report in Fig. 2 the behavior of the memberships for selected values of α and the parameters of the clustering algorithms (see [9] for plots with other α and parameter values). In particular, we set the value of m in FCM I and λ in FCM II in order to obtain a similar distribution of the memberships at the end of the algorithms. For small α the results are almost invariant as expected (first row of Fig 2). For values of α of the order of the mean of the squared distances, the memberships in FCM I have a very different behavior with respect to those on the original set. FCM II seems to be less sensitive to shift operations, even for large values of α. At the end of the algorithm, the memberships can be defuzzified using a threshold of 0.5 to obtain the cluster labels. The cluster labels have been found to be identical for all the tested valued of α.

For PCM I we set $m = 1.5$ as in FCM I, and for PCM II there are no parameters to set up. In fact, in both the possibilistic algorithms, it is possible to set the value of γ for the computation of η. We set $\gamma = 0.5$ for PCM I and PCM II. The initialization of the memberships has been done using the result obtained by the FCM II, since it showed high robustness to distance shifts. This means that the values of η_i have been computed on the basis of the memberships obtained by the FCM II. It can be seen, in the second row of Fig. 2, that even for small values of α, the behavior of the memberships in PCM I and PCM II is strongly affected by the shift operation.

The difference of the memberships in FCM I, after dissimilarities shift, presents a peak around the first iterations. One possible explanation can be found by looking at the functional and at the values assumed by the memberships around those iterations. The terms $A(U)$ and $B(U)$ give a high contribution when the memberships are near $1/c$. In the first exploratory iterations, the values are more likely to be near $1/c$ than later, when the clusters are well identified. As we can see from Eq. D.10, as α increases, the memberships tend to be $1/c$, if $c \ll n$. The memberships for $\alpha \neq 0$ do not diverge from those of $\alpha = 0$; this effect can be noticed also for FCM II. A small peak in the difference of the memberships for different α can be seen also for FCM II, and is the effect of $B(U)$ in the Lagrangian.

3.1.2 Histogram of the memberships at the end of the optimization

Let's now analyze the histogram of the memberships, at the end of the optimization, for different values of the shift and the clustering parameters. Let's introduce the following entropy-based score:

$$O(U) = -\sum_{ij} u_{ij} \log(u_{ij}) \tag{43}$$

Fig. 3 shows a plot of $O(U)$ for different values of α and the parameters. It is possible to see how FCM II gives nearly the same results for different values of shift. In FCM I, it is evident that this happens only in an almost crisp set-up (m close to 1). In PCM I and PCM II, a small value of $O(U)$ is caused by several small values of the memberships. Indeed, for PCM II, where we set the width of the membership function, for small values of γ, PCM II will tend to give high membership to the more representative patterns only.

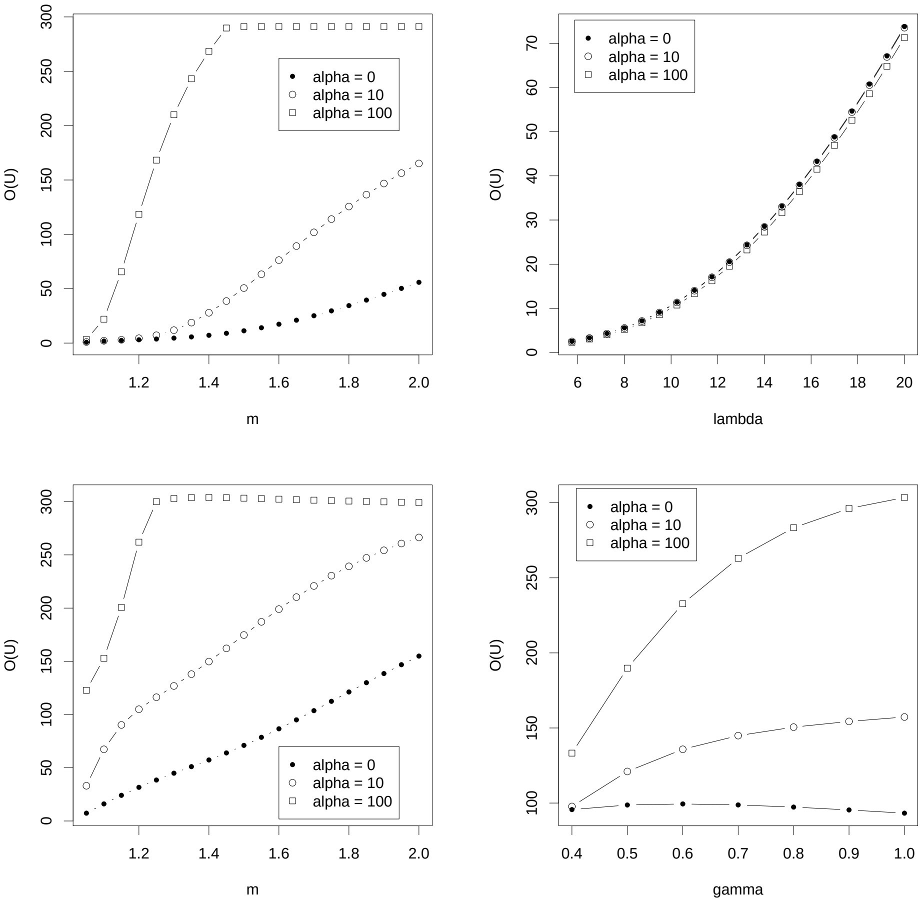

Figure 3. Entropy vs the clustering parameters for different values of α. First row FCM I and FCM II; second row PCM I and PCM II.

3.1.3 Coping with the shift by changing the parameters

Another interesting study that can be performed on a synthetic data set is to analyze if there is the chance to cope with the problems coming from the shift operation, by tuning the parameters in an appropriate way. In order to do that, let's denote with U and U' the memberships resulting from two clustering algorithms, and introduce the following score:

$$\mathrm{KL} = \sum_{ij} u_{ij} \log\left(\frac{u_{ij}}{u'_{ij}}\right) \tag{44}$$

KL is a Kullback-Leibler-based score on the memberships that measures the distance between the distributions of the memberships at the end of the two algorithms. Since we are in a synthetic set up, it is possible to do the following:

- run a fuzzy clustering algorithm with some parameters on the unshifted version of the data set;

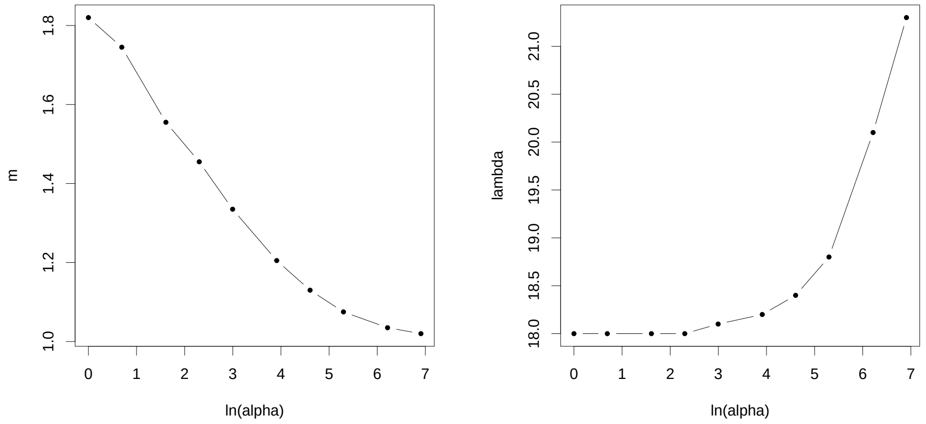

Figure 4. FCM I and FCM II - Parameter values needed to obtain the same solution on a shifted data set as in the unshifted one.

- search for the new parameters, on the shifted data set, that lead to the same distribution of the memberships as in the unshifted case.

We set the value of $m = 2$ for FCM I and a value of $\lambda = 18$ for FCM II; these two values give an almost identical distribution of the memberships at the end of the algorithms. Searching for the value of m and λ minimizing the value of KL, we obtained the plot in Fig. 4. It is possible to see that in a wide range of α values, FCM II is not affected by the shift. In FCM I, it is necessary to move toward a more crisp behavior of the algorithm (lowering m) to obtain the same distribution of the memberships as in the unshifted case. For the possibilistic clustering algorithms, we don't report this study for the sake of brevity, since the distribution of the memberships assume a very different form. This means that it was not possible to select the parameters in the shifted case giving a KL score close to 0.

3.2 Synthetic Data Set 2

The second data set is composed by 200-20 dimensional points divided in two clusters. The first ten features are normally distributed with means $(-2, -2, \ldots, -2)$ for the first 100 patterns, and $(2, 2, \ldots, 2)$ for the other 100. The standard deviation is set to 1 for all the features and both the distributions. The other 10 features are uniformly distributed in the interval $[-1, 1]$. The mean squared distance is 106.2, the median is 103.0, and the maximum is 334.

In Fig. 5, we report the entropy of the memberships at the end of the four clustering algorithm, for different values of α and the parameters. Again it is possible to repeat the considerations about the robustness of FCM II to shift operation, and the inability of the other algorithms to deal with that.

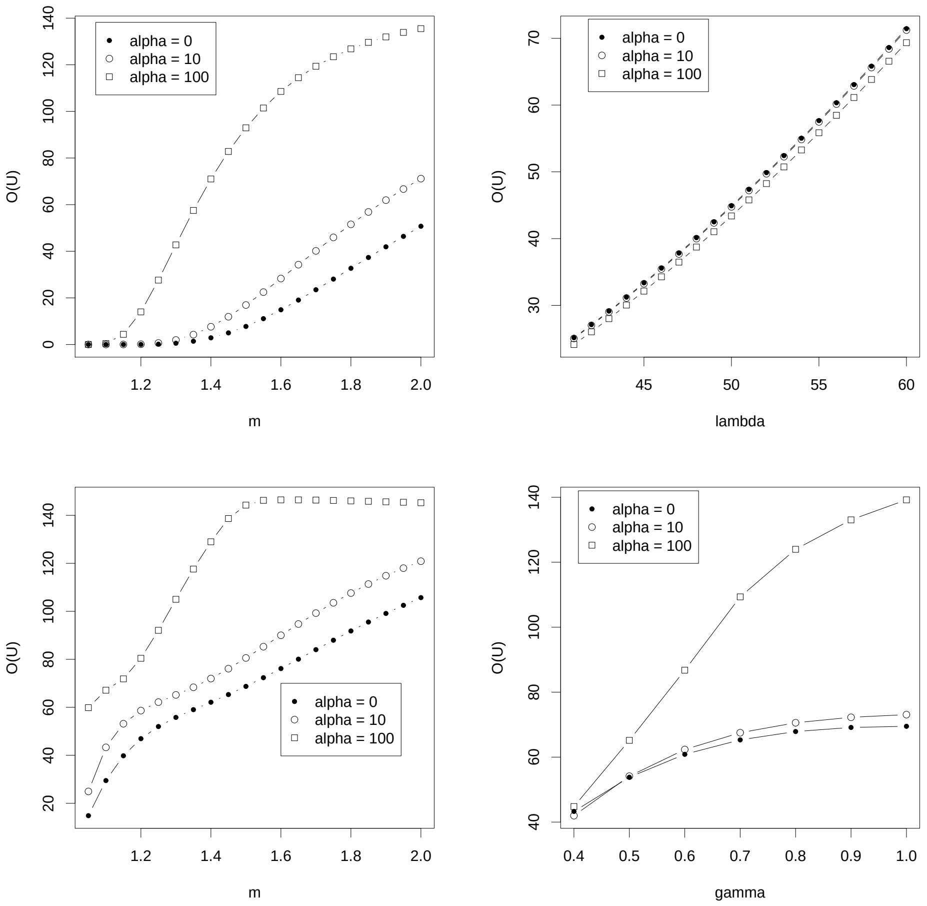

Figure 5. Entropy vs the clustering parameters for different values of α. First row FCM I and FCM II; second row PCM I and PCM II.

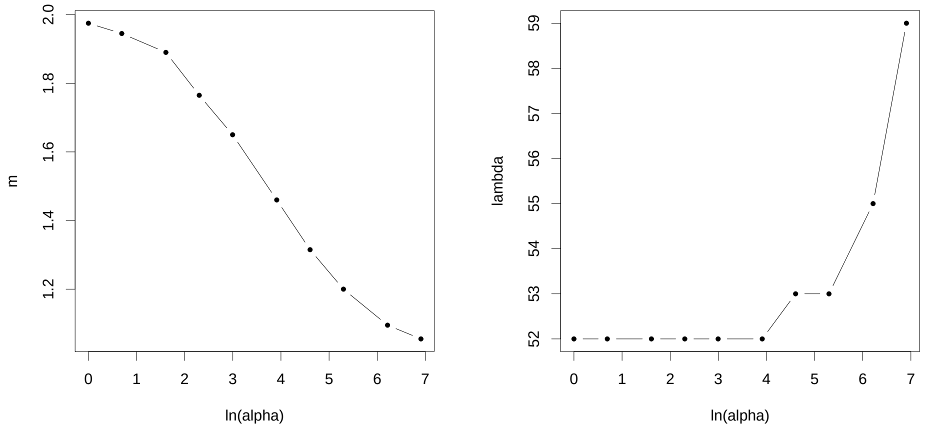

Figure 6. Parameter values needed to obtain the same solution on a shifted data set as the unshifted one (FCM I first plot, FCM II second plot).

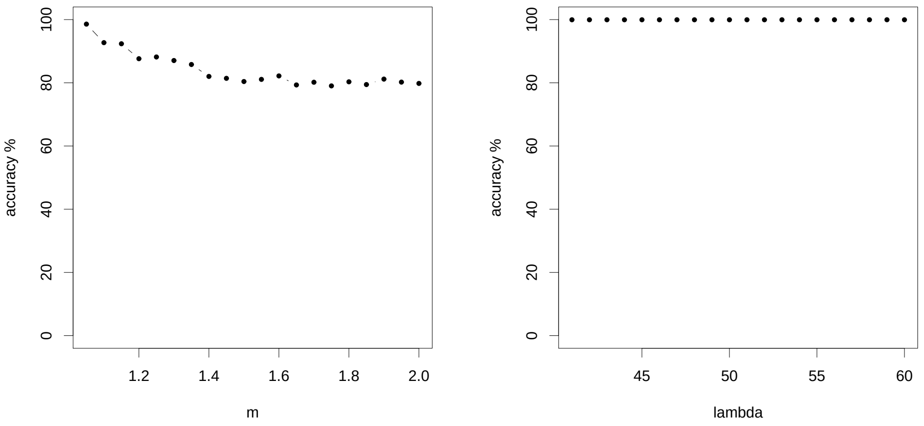

Figure 7. Accuracy vs the parameters (FCM I first plot, FCM II second plot).

We repeated the test based on the KL score (Fig. 6). Again, the possibilistic algorithms lead to very different distributions, and we do not report this analysis. For FCM I, it is necessary to set m to values close to 1 as the shift becomes large. FCM II does not need any tuning to λ in a broad range of values; when α is very large, it is necessary to increase it slightly.

In this synthetic data set we performed another test. We perturbed the relational matrix R with increasing levels of uniform distributed noise. This can simulate a real situation when the measures of the relations between patterns are noisy. In our case, we used the symmetrization on the noisy R, and we shifted it obtaining a positive semidefinite kernel matrix. We studied the match between cluster labels and true class labels, with respect to different values of the parameters. The cluster labels are obtained by assigning a pattern to the cluster for which the membership is the highest. Up to noise levels having maximum value of 500 both FCM I and FCM II are able to label the two clusters correctly. The situation for a noise uniformly distributed in $[0, 1000]$ is shown in Fig. 7. For the possibilistic clustering, the matching with the labels starts to fail with lower noise levels.

3.3 USPS Data Set

We tested the presented algorithms on the USPS data set [36,23]. It is composed by 9298 images acquired and processed from handwritten zip-codes appeared on real US mail. Each image is 16 × 16 pixels; the training set in composed by 7219 images and the test set by 2001 images. As in Ref. [23], only the characters in the training set labeled as “0” and “7” have been considered, obtaining a subset of 1839 images. The dissimilarity function used in Ref. [23] is based on the Simpson score, which is a matching function between binary images. Given two binary images, the following matrix can be constructed:

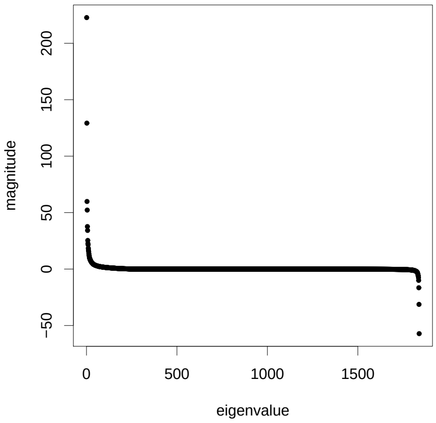

Figure 8. USPS data set - Eigenvalues of the matrix S^c sorted by decreasing magnitude.

		Img 1	
		0	1
Img 2	0	d	c
	1	b	a

where: a is the number of pixels that are white in both the images; b is the number of pixels that are white in Img 2 and black in Img 1; c is the number of pixels that are white in Img 1 and black in Img 2; d is the number of pixels that are black in both the images. The Simpson score of two binary images is defined as:

$$l = \frac{a}{\min(a+b, a+c)} \tag{45}$$

The images in the USPS data set are not binary; this has required a normalization between 0 and 1, and a thresholding at 0.5. The dissimilarity based on the Simpson score, is:

$$r_{ij} = 2 - 2l_{ij} \tag{46}$$

which is between 0 and 2. The mean value of R, in this data set, is 0.88, and the median is 0.92. The Simpson dissimilarity is symmetric, but does not obey to the triangular inequality. Indeed, as can be seen in Fig. 8, there are some negative eigenvalues of S^c. The smallest eigenvalue $\lambda_1 = -57.2$ is the value that added to the dissimilarities let $\tilde{R}$ become a squared Euclidean distance matrix. We applied the four clustering algorithms on the selected binary images, searching for two clusters.

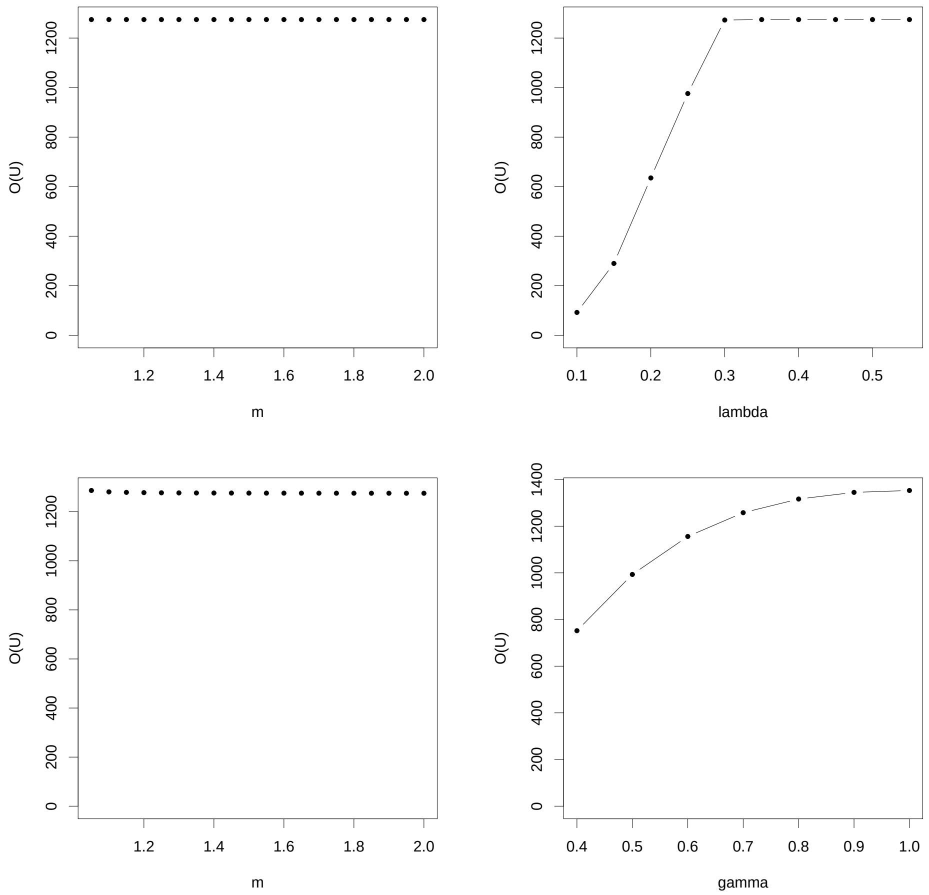

Figure 9. Entropy vs the clustering parameters. First row FCM I and FCM II; second row PCM I and PCM II.

In Fig. 9, we can see the plot of the entropy $O(U)$ of the memberships versus the parameters. Only FCM II, for particular values of λ, allows to obtain a meaningful distribution of the memberships Fig. 10 shows the accuracy obtained of the algorithms with respect to the parameters. The accuracy is measured as the match between cluster labels and class labels. Both the entropy and the accuracy are averaged over 50 trials with different initializations. In these experiments, we noticed that FCM I resulted to be strongly affected by different initializations.

FCM II resulted the best algorithm in terms of performances. The histogram of the membership allows to refine the results, identifying the patterns that are more representative of the two clusters, and those that are on the border between them. As an illustrative example, we show (Fig. 11) the histogram of the highest membership of the patterns to the clusters, obtained by FCM II with $\lambda = 0.15$, that is the setup giving the best results on average (accuracy of 98.2 %). We can set a threshold on such memberships to label the patterns as objects in the border between the two clusters. By looking at the histogram, we set this threshold to 0.9. Fig. 11 shows the group of border objects, and

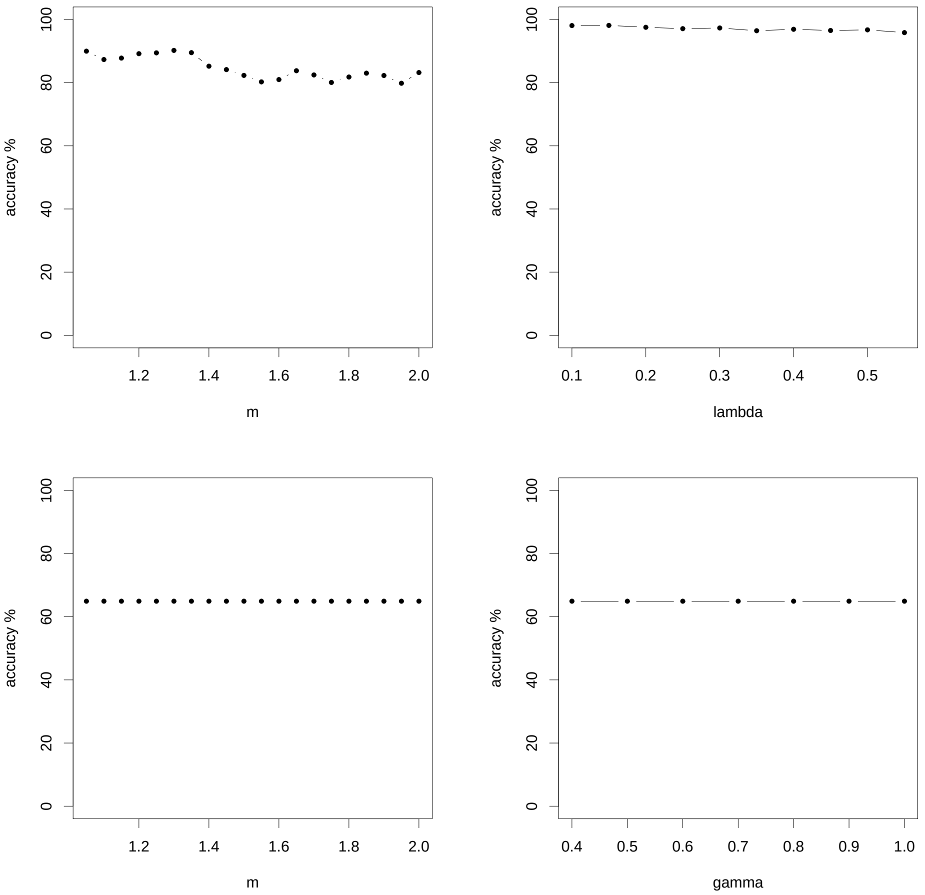

Figure 10. Accuracy vs the clustering parameters. First row FCM I and FCM II; second row PCM I and PCM II.

the two clusters found by the algorithm. The images have been sorted with decreasing values of memberships. The image in the top-left corner has the highest membership and moving to the right the memberships decrease.

4 Conclusions

In this paper, four central clustering algorithms based on fuzzy memberships have been studied: FCM I, FCM II, PCM I, and PCM II. In particular, it has been studied how the symmetrization and the shift operation on the dissimilarities affect their objective function. The main theoretical results include the proof of invariance of the objective function to symmetrization and the lack of invariance to shift operations. Moreover, the four considered clustering algorithms have been presented under a more general framework, highlighting the connections between the relational clustering and the clustering in the space induced by positive semidefinite kernels.

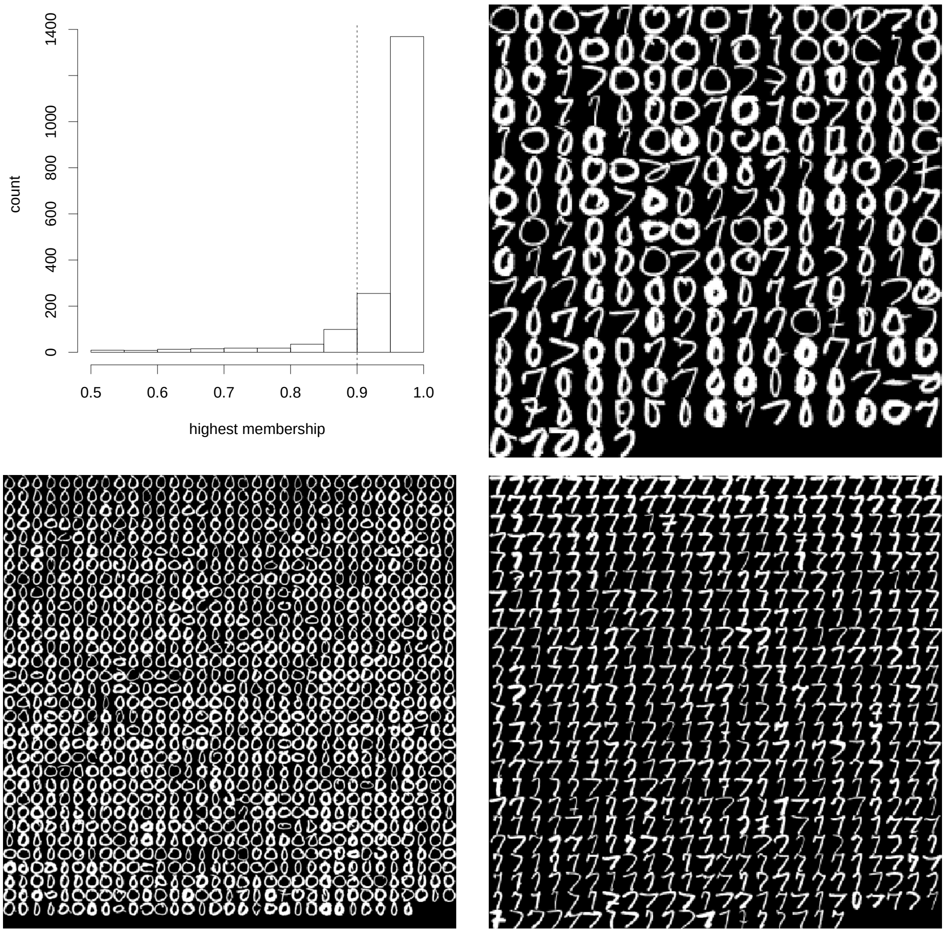

Figure 11. Analysis of the results obtained by FCM II with $\lambda = 0.15$. First row: histogram of the highest memberships of patterns to the two clusters and group of objects having membership below the threshold (border objects). Second row: the two clusters found by the algorithm

Both the theoretical analysis and the experiments conducted on synthetic and real data sets, show that FCM II is the least sensitive to shift operations. Indeed, its Lagrangian is not invariant only for a term that is bounded by the number of clusters c, while FCM I's contains also a term bounded by the number of patterns n. In a typical problem, the number of clusters is very small, compared to the number of patterns, and this gives to FCM II more robustness with respect to FCM I. The situation is the same for PCM II and PCM I, but the lack of competitivity between clusters, make them finding solution where centroids collapse into a single one, even for small shifts. Small distances are more affected by the sum of a constant than large distances, making the data set sparse; the possibilistic algorithms are not able to handle efficiently these situations.

In the experimental tests on synthetic data sets, we analyzed the behavior of the presented algorithms during and at the end of the optimization. In the first data set, we studied the behavior of the memberships during the optimization and the entropy of the memberships at the end of the algorithms. We also studied how it is possible to cope with the effect of the shift, by tuning the parameters, by analyzing a score based on the Kullback-Leibler divergence. On another synthetic data set, we studied the performances in terms of correct assignment of cluster labels, when adding noise to the relational matrix. In all these cases, FCM II showed more robustness in comparison to the other algorithms. Regarding the parameters, we saw that in FCM I, one needs to move toward a more crisp setup to cope with the shift. If the shift is very large, as in the case of the USPS data set, the values of m must be set to 1 plus very small fractions. In FCM II, the value of λ can be set on the basis of the values of the relational matrix. Values around half of the average of the dissimilarities, have been found to be a good starting point. For large values of the shift, the order of magnitude of λ to achieve the same result as in the unshifted case, does not change; λ requires only to be slightly increased. On USPS data set, we showed the performances of the algorithms in a real application. FCM II resulted the only algorithm, among those presented here, able to assign memberships to the patterns in a meaningful way. The analysis of the memberships in FCM II allows to identify the patterns that are close to the border between clusters, as well as those that are more representative of the clusters.

Based on the analysis conducted in this paper, we claim that FCM II is the algorithm, among those presented, that is less affected by shift transformations. This suggests that it is the preferable algorithm, among those presented, to be employed when patterns are represented by means of non-metric dissimilarities. The relational dual of FCM II has never been proposed before in the case of non-metric dissimilarities, and represents another novelty of this paper.

A Proof that S^c is Uniquely Determined by R^c

The centralized version of a generic matrix P is defined as:

$$P^c = QPQ \tag{A.1}$$

that is equivalent to:

$$p_{ij}^c = p_{ij} - \frac{1}{n}\sum_{h=1}^{n} p_{hj} - \frac{1}{n}\sum_{k=1}^{n} p_{ik} + \frac{1}{n^2}\sum_{h=1}^{n}\sum_{k=1}^{n} p_{hk} \tag{A.2}$$

Inverting Eq. 5, we can write:

$$s_{ij} = -\frac{1}{2}\left(r_{ij} - s_{ii} - s_{jj}\right) \tag{A.3}$$

The entries of the centralized version of S are:

$$\begin{aligned}
s^c_{ij} = &-\frac{1}{2}\Bigg[(r_{ij} - s_{ii} - s_{jj}) - \frac{1}{n}\sum_{h=1}^{n}(r_{hj} - s_{hh} - s_{jj}) - \frac{1}{n}\sum_{k=1}^{n}(r_{ik} - s_{ii} - s_{kk}) \\
&+\frac{1}{n^2}\sum_{h=1}^{n}\sum_{k=1}^{n}(r_{hk} - s_{hh} - s_{kk})\Bigg] && \text{(A.4)} \\
= &-\frac{1}{2}\left(r_{ij} - \frac{1}{n}\sum_{h=1}^{n} r_{hj} - \frac{1}{n}\sum_{k=1}^{n} r_{ik} + \frac{1}{n^2}\sum_{h=1}^{n}\sum_{k=1}^{n} r_{hk}\right) && \text{(A.5)}
\end{aligned}$$

This proves that the centralized version of S is uniquely determined by the centralized version of R:

$$S^c = -\frac{1}{2}R^c \tag{A.6}$$

B Proof of Theorem 1.1

In this section we report the proof that R is a squared Euclidean distance matrix $\iff S^c \succeq 0$ [23,31]. Let's start with $\Rightarrow$. The centralized version of R is:

$$R^c = QRQ = R - \frac{1}{n}ee^{\mathrm{T}}R - \frac{1}{n}Ree^{\mathrm{T}} + \frac{1}{n^2}ee^{\mathrm{T}}Ree^{\mathrm{T}} \tag{B.1}$$

Assuming that a set of vectors $\mathbf{x}$ exists, for which:

$$r_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|^2 \tag{B.2}$$

the entries of R^c can be written as:

$$\begin{aligned}
r^c_{ij} = &\|\mathbf{x}_i - \mathbf{x}_j\|^2 - \frac{1}{n}\sum_{h=1}^{n}\|\mathbf{x}_h - \mathbf{x}_j\|^2 - \frac{1}{n}\sum_{k=1}^{n}\|\mathbf{x}_i - \mathbf{x}_k\|^2 + \frac{1}{n^2}\sum_{h=1}^{n}\sum_{k=1}^{n}\|\mathbf{x}_h - \mathbf{x}_k\|^2 \\
= &\mathbf{x}_i^{\mathrm{T}}\mathbf{x}_i + \mathbf{x}_j^{\mathrm{T}}\mathbf{x}_j - 2\mathbf{x}_i^{\mathrm{T}}\mathbf{x}_j - \frac{1}{n}\left(\sum_{h=1}^{n}\mathbf{x}_h^{\mathrm{T}}\mathbf{x}_h + \mathbf{x}_j^{\mathrm{T}}\mathbf{x}_j - 2\mathbf{x}_h^{\mathrm{T}}\mathbf{x}_j\right) \\
&-\frac{1}{n}\left(\sum_{k=1}^{n}\mathbf{x}_i^{\mathrm{T}}\mathbf{x}_i + \mathbf{x}_k^{\mathrm{T}}\mathbf{x}_k - 2\mathbf{x}_i^{\mathrm{T}}\mathbf{x}_k\right) + \frac{1}{n^2}\left(\sum_{h=1}^{n}\sum_{k=1}^{n}\mathbf{x}_h^{\mathrm{T}}\mathbf{x}_h + \mathbf{x}_k^{\mathrm{T}}\mathbf{x}_k - 2\mathbf{x}_h^{\mathrm{T}}\mathbf{x}_k\right) \\
= &-2\left(\mathbf{x}_i^{\mathrm{T}}\mathbf{x}_j - \frac{1}{n}\sum_{h=1}^{n}\mathbf{x}_h^{\mathrm{T}}\mathbf{x}_j - \frac{1}{n}\sum_{k=1}^{n}\mathbf{x}_i^{\mathrm{T}}\mathbf{x}_k + \frac{1}{n^2}\sum_{h=1}^{n}\sum_{k=1}^{n}\mathbf{x}_h^{\mathrm{T}}\mathbf{x}_k\right) && \text{(B.3)}
\end{aligned}$$

Introducing the quantity:

$$\bar{\mathbf{x}} = \frac{1}{n} \sum_{h=1}^{n} \mathbf{x}_h \tag{B.4}$$

we can rewrite Eq. B.3 in a more compact way:

$$r_{ij}^{c} = -2(\mathbf{x}_i - \bar{\mathbf{x}})^{\mathrm{T}}(\mathbf{x}_j - \bar{\mathbf{x}}) = -2\breve{\mathbf{x}}_i^{\mathrm{T}}\breve{\mathbf{x}}_j \tag{B.5}$$

This is equivalent to say that:

$$S^{c} = \breve{X}\breve{X}^{\mathrm{T}} \tag{B.6}$$

which proves ⇒.

To prove ⇐, since S^c is positive semidefinite, we can write:

$$S^{c} = XX^{\mathrm{T}} \tag{B.7}$$

where the rows of X are vectors $\mathbf{x} \in \mathbb{R}^d$. From Eq. 5, we obtain:

$$\begin{aligned} r_{ij} &= s_{ii} + s_{jj} - 2s_{ij} \\ &= \mathbf{x}_i^{\mathrm{T}}\mathbf{x}_i + \mathbf{x}_j^{\mathrm{T}}\mathbf{x}_j - 2\mathbf{x}_i^{\mathrm{T}}\mathbf{x}_j \\ &= \|\mathbf{x}_i - \mathbf{x}_j\|^2 \end{aligned} \tag{B.8}$$

thus proving ⇐.

C Pre-Shift and Post-Shift

Let's analyze why:

$$S^{c} + \alpha I \neq -\frac{1}{2}(Q\tilde{R}Q) \tag{C.1}$$

and how this can influence the behavior of the studied clustering algorithms. First, let's see what is the difference between the resulting matrices. For the pre-shift we have:

$$-\frac{1}{2}(Q\tilde{R}Q) = -\frac{1}{2}(QRQ) - \alpha Q(ee^{\mathrm{T}} - I)Q = S^{c} - \alpha Q(ee^{\mathrm{T}} - I)Q \tag{C.2}$$

Now:

$$Q(ee^{\mathrm{T}} - I)Q = Qee^{\mathrm{T}}Q - QQ = -QQ = -Q \tag{C.3}$$

since:

$$Qe = (I - \frac{1}{n}ee^{\mathrm{T}})e = e - e = \mathbf{0} \tag{C.4}$$

and:

$$QQ = (I - \frac{1}{n}ee^{\mathrm{T}})(I - \frac{1}{n}ee^{\mathrm{T}}) = I - \frac{2}{n}ee^{\mathrm{T}} + \frac{1}{n^2}ee^{\mathrm{T}}ee^{\mathrm{T}} = I - \frac{1}{n}ee^{\mathrm{T}} = Q \tag{C.5}$$

Thus:

$$-\frac{1}{2}(Q\tilde{R}Q) = S^c + \alpha Q \tag{C.6}$$

The difference between the matrices associated to post-shift and pre-shift is:

$$\alpha(I - Q) = \frac{\alpha}{n}ee^{\mathrm{T}} \tag{C.7}$$

Now we prove that $\|\mathbf{x}_h - \mathbf{v}_j\|^2$ is independent from the choice of the pre-shift or post-shift:

$$\|\mathbf{x}_h - \mathbf{v}_j\|^2 = k'_{hh} - 2\frac{\sum_{r=1}^n u_{ir}^\theta k'_{rh}}{\sum_{r=1}^n u_{ir}^\theta} + \frac{\sum_{r=1}^n \sum_{s=1}^n u_{ir}^\theta u_{is}^\theta k'_{rs}}{(\sum_{r=1}^n u_{ir}^\theta)^2} \tag{C.8}$$

All the entries of the pre-shifted and post-shifted versions of K differ for a constant term:

$$k'_{ij} = k_{ij} + \frac{\alpha}{n} \qquad \forall i, j \tag{C.9}$$

Such difference cancels out in the computation of the distance between patterns and centroids:

$$\|\mathbf{x}_h - \mathbf{v}_j\|^2 = k_{hh} + \frac{\alpha}{n} - 2\frac{\sum_{r=1}^n u_{ir}^\theta k_{rh}}{\sum_{r=1}^n u_{ir}^\theta} - 2\frac{\alpha}{n} + \frac{\sum_{r=1}^n \sum_{s=1}^n u_{ir}^\theta u_{is}^\theta k_{rs}}{(\sum_{r=1}^n u_{ir}^\theta)^2} + \frac{\alpha}{n} \tag{C.10}$$

D Derivation of FCM I, FCM II, PCM I, and PCM II

This section shows the derivation of FCM I, FCM II, PCM I, and PCM II. At the end of each derivation, we discuss the influence of the distance shift on the update equations.

D.1 Fuzzy c-means I

The Lagrangian $L(U)$ is:

$$L(U, V) = \sum_{i=1}^{c} \sum_{h=1}^{n} u_{ih}^{m} \|\mathbf{x}_h - \mathbf{v}_i\|^2 + \sum_{h=1}^{n} \beta_h (1 - \sum_{i=1}^{c} u_{ih}) \tag{D.1}$$

The first term is the distortion $G(U, V)$ and the second is $W(U)$, which is not zero, since the memberships are subjected to the probabilistic constraint in Eq. 16. The parameter $m > 1$ works as a fuzzifier parameter; for high values of m the memberships tend to be equally distributed among clusters. Setting to zero the derivatives of $L(U, V)$ with respect to u_{ih}:

$$\frac{\partial L(U, V)}{\partial u_{ih}} = m u_{ih}^{m-1} \|\mathbf{x}_h - \mathbf{v}_i\|^2 - \beta_h = 0 \tag{D.2}$$

we obtain:

$$u_{ih} = \left(\frac{\beta_h}{m \|\mathbf{x}_h - \mathbf{v}_i\|^2} \right)^{\frac{1}{m-1}} \tag{D.3}$$

Substituting the expression of u_{ih} into the constraint equation:

$$\sum_{i=1}^{c} \left(\frac{\beta_h}{m \|\mathbf{x}_h - \mathbf{v}_i\|^2} \right)^{\frac{1}{m-1}} = 1 \tag{D.4}$$

we obtain the Lagrange multipliers:

$$\beta_h = \left[\sum_{i=1}^{c} \left(\frac{1}{m \|\mathbf{x}_h - \mathbf{v}_i\|^2} \right)^{\frac{1}{m-1}} \right]^{1-m} \tag{D.5}$$

Substituting Eq. D.5 into Eq. D.3, the equation for the update of the memberships u_{ih} can be obtained:

$$u_{ih}^{-1} = \sum_{j=1}^{c} \left(\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2}{\|\mathbf{x}_h - \mathbf{v}_j\|^2} \right)^{\frac{1}{m-1}} \tag{D.6}$$

To compute the equation for the update of the $\mathbf{v}_i$, we set to zero the derivatives of $L(U,V)$ with respect to $\mathbf{v}_i$:

$$\frac{\partial L(U,V)}{\partial \mathbf{v}_i} = -\sum_{h=1}^{n} u_{ih}^{m} \left(\mathbf{x}_h - \mathbf{v}_i\right) = 0 \tag{D.7}$$

obtaining:

$$\mathbf{v}_i = \frac{\sum_{h=1}^{n} u_{ih}^{m} \mathbf{x}_h}{\sum_{h=1}^{n} u_{ih}^{m}} \tag{D.8}$$

After a shift operation on the dissimilarities, the Lagrangian $L_\alpha(U,V)$ contains two more terms: $A(U)$ and $B(U)$. Since $A(U) < n$ and $B(U) < c$, if $c \ll n$, we can neglect the term $B(U)$:

$$L_\alpha(U,V) = \sum_{h=1}^{n}\sum_{i=1}^{c} u_{ih}^{m} \|\mathbf{x}_h - \mathbf{v}_i\|^2 + \alpha \sum_{h=1}^{n}\sum_{i=1}^{c} u_{ih}^{m} + \sum_{h=1}^{n} \beta_h (1 - \sum_{i=1}^{c} u_{ih}) \tag{D.9}$$

Following the same procedure, we obtain that the update of the $\mathbf{v}$ is the same as in Eq. D.8, but the update of the memberships is:

$$u_{ih}^{-1} = \sum_{j=1}^{c} \left(\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2 + \alpha}{\|\mathbf{x}_h - \mathbf{v}_j\|^2 + \alpha} \right)^{\frac{1}{m-1}} \tag{D.10}$$

This shows that for large values of α and $c \ll n$ the membership tend to be equally distributed among clusters.

D.2 Fuzzy c-means II

The Lagrangian $L(U,V)$ for FCM II is:

$$L(U,V) = \sum_{h=1}^{n}\sum_{i=1}^{c} u_{ih} \|\mathbf{x}_h - \mathbf{v}_i\|^2 + \lambda \sum_{h=1}^{n}\sum_{i=1}^{c} u_{ih} \ln(u_{ih}) + \sum_{h=1}^{n} \beta_h (1 - \sum_{i=1}^{c} u_{ih}) \tag{D.11}$$

The entropic term favors values of the memberships near zero or one (Fig. D.1). Let's compute the derivative of $L(U, V)$ with respect to u_{ih}:

$$\frac{\partial L(U,V)}{\partial u_{ih}} = \|\mathbf{x}_h - \mathbf{v}_i\|^2 + \lambda(\ln(u_{ih}) + 1) - \beta_h = 0 \tag{D.12}$$

This leads to:

$$u_{ih} = \frac{1}{e} \exp\left(\frac{\beta_h}{\lambda}\right) \exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2}{\lambda}\right) \tag{D.13}$$

Substituting the last equation into the probabilistic constraint, we obtain:

$$\sum_{i=1}^{c} \frac{1}{e} \exp\left(\frac{\beta_h}{\lambda}\right) \exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2}{\lambda}\right) = 1 \tag{D.14}$$

This allows to compute the Lagrange multipliers:

$$\beta_h = \lambda - \lambda \ln\left(\sum_{j=1}^{c} \exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_j\|^2}{\lambda}\right)\right) \tag{D.15}$$

Substituting Eq. D.15 into Eq. D.13, we obtain the equation for the update of u_{ih}:

$$u_{ih} = \frac{\exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2}{\lambda}\right)}{\sum_{j=1}^{c} \exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_j\|^2}{\lambda}\right)} \tag{D.16}$$

Setting to zero the derivatives of $L(U, V)$ with respect to $\mathbf{v}_i$:

$$\frac{\partial L(U,V)}{\partial \mathbf{v}_i} = -\sum_{h=1}^{n} u_{ih} (\mathbf{x}_h - \mathbf{v}_i) = 0 \tag{D.17}$$

the following update formula for the centroids $\mathbf{v}_i$ is obtained:

$$\mathbf{v}_i = \frac{\sum_{h=1}^{n} u_{ih} \mathbf{x}_h}{\sum_{h=1}^{n} u_{ih}} \tag{D.18}$$

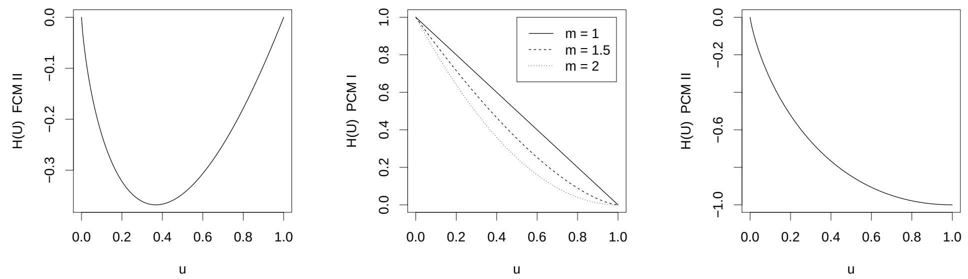

Figure D.1. Plot of the FCM II entropy $H(u_{ih}) = u_{ih} \ln(u_{ih})$, PCM I entropy $H(u_{ih}) = (1 - u_{ih})^m$ for increasing values of m, and PCM II entropy $H(u_{ih}) = u_{ih} \ln(u_{ih}) - u_{ih}$.

D.3 Possibilistic c-means I

The PCM I Lagrangian $L(U, V)$ does not have the $W(U)$ term coming from the probabilistic constraint on the memberships:

$$L(U, V) = \sum_{h=1}^{n} \sum_{i=1}^{c} u_{ih}^m \|\mathbf{x}_h - \mathbf{v}_i\|^2 + \sum_{i=1}^{c} \eta_i \sum_{h=1}^{n} (1 - u_{ih})^m \tag{D.19}$$

The entropic term penalizes small values of the memberships.

Setting to zero the derivatives of $L(U, V)$ with respect to the memberships u_{ih}:

$$\frac{\partial L(U, V)}{\partial u_{ik}} = m u_{ih}^{m-1} (\|\mathbf{x}_h - \mathbf{v}_i\|^2) - \eta_i m (1 - u_{ih})^{m-1} = 0 \tag{D.20}$$

We obtain directly the update equation:

$$u_{ih}^{-1} = \left(\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2}{\eta_i} \right)^{\frac{1}{m-1}} + 1 \tag{D.21}$$

The following derivative of $L(U, V)$:

$$\frac{\partial L(U, V)}{\partial \mathbf{v}_i} = - \sum_{h=1}^{n} u_{ih}^m (\mathbf{x}_h - \mathbf{v}_i) = 0 \tag{D.22}$$

gives the update equation for the centroids $\mathbf{v}_i$:

$$\mathbf{v}_i = \frac{\sum_{h=1}^{n} u_{ih}^m \mathbf{x}_h}{\sum_{h=1}^{n} u_{ih}^m} \tag{D.23}$$

The following criterion is suggested to estimate the value of η_i:

$$\eta_i = \gamma \frac{\sum_{h=1}^n u_{ih}^m \|\mathbf{x}_h - \mathbf{v}_i\|^2}{\sum_{h=1}^n u_{ih}^m} \tag{D.24}$$

where γ is usually set to one.

In presence of a shift operation on the dissimilarities, the Lagrangian is not invariant. Following the same considerations made for FCM I about $A(U)$ and $B(U)$, it is possible to neglect $B(U)$, if $c \ll n$:

$$L_\alpha(U,V) = \sum_{h=1}^n \sum_{i=1}^c u_{ih}^m \|\mathbf{x}_h - \mathbf{v}_i\|^2 + \sum_{i=1}^c \eta_i \sum_{h=1}^n (1 - u_{ih})^m + \alpha \sum_{h=1}^n \sum_{i=1}^c u_{ih}^m \tag{D.25}$$

Following the same procedure, we derive the equations for the update of U:

$$u_{ih}^{-1} = \left(\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2 + \alpha}{\eta_i} \right)^{\frac{1}{m-1}} + 1 \tag{D.26}$$

For large values of α, the memberships tend to become small.

D.4 Possibilistic c-means II

The PCM II Lagrangian $L(U,V)$ is:

$$L(U,V) = \sum_{h=1}^n \sum_{i=1}^c u_{ih} \|\mathbf{x}_h - \mathbf{v}_i\|^2 + \sum_{i=1}^c \eta_i \sum_{h=1}^n (u_{ih} \ln(u_{ih}) - u_{ih}) \tag{D.27}$$

The entropic term penalizes small values of the memberships.

Setting to zero the derivatives of $L(U,V)$ with respect to the memberships u_{ih}:

$$\frac{\partial L(U,V)}{\partial u_{ih}} = \|\mathbf{x}_h - \mathbf{v}_i\|^2 + \eta_i \ln(u_{ih}) = 0 \tag{D.28}$$

we obtain:

$$u_{ih} = \exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2}{\eta_i} \right) \tag{D.29}$$

Setting to zero the derivatives of $L(U, V)$ with respect to $\mathbf{v}_i$:

$$\frac{\partial L(U, V)}{\partial \mathbf{v}_i} = -\sum_{h=1}^{n} u_{ih} \left(\mathbf{x}_h - \mathbf{v}_i \right) = 0 \tag{D.30}$$

we obtain the update formula for the centroids $\mathbf{v}_i$:

$$\mathbf{v}_i = \frac{\sum_{h=1}^{n} u_{ih} \mathbf{x}_h}{\sum_{h=1}^{n} u_{ih}} \tag{D.31}$$

References

[1] T. M. Apostol. *Calculus, 2 vols.* Wiley, 2 edition, 1967.

[2] N. Aronszajn. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68(3):337–404, 1950.

[3] G. Beni and X. Liu. A least biased fuzzy clustering method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(9):954–960, 1994.

[4] J. C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms.* Kluwer Academic Publishers, Norwell, MA, USA, 1981.

[5] C. Cortes and V. Vapnik. Support vector networks. *Machine Learning*, 20:273–297, 1995.

[6] M. de Cáceres, F. Oliva, and X. Font. On relational possibilistic clustering. *Pattern Recognition*, 39(11):2010–2024, 2006.

[7] T. Denoeux and M. H. Masson. Evclus: evidential clustering of proximity data. *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, 34(1):95–109, 2004.

[8] E. Diday. La méthode des nuées dynamiques. *Revue de Stat Appliquée*, 19(2):19–34, 1971.

[9] M. Filippone. Fuzzy clustering of patterns represented by pairwise dissimilarities. Technical Report ISE-TR-07-05, Department of Information and Software Engineering, George Mason University, October 2007.

[10] M. Filippone, F. Camastra, F. Masulli, and S. Rovetta. A survey of kernel and spectral methods for clustering. *Pattern Recognition*, 41(1):176–190, January 2008.

[11] M. Filippone, F. Masulli, and S. Rovetta. Possibilistic clustering in feature space. In *WILF*, Lecture Notes in Computer Science. Springer, 2007.

[12] M. Girolami. Mercer kernel based clustering in feature space. *IEEE Transactions on Neural Networks*, 13(3):780–784, 2002.

[13] R. J. Hathaway and J. C. Bezdek. Nerf c-means: Non-euclidean relational fuzzy clustering. *Pattern Recognition*, 27(3):429–437, 1994.

[14] R. J. Hathaway, J. W. Davenport, and J. C. Bezdek. Relational duals of the c-means clustering algorithms. *Pattern Recognition*, 22(2):205–212, 1989.

[15] F. Höppner and F. Klawonn. A contribution to convergence theory of fuzzy c-means and derivatives. *IEEE Transactions on Fuzzy Systems*, 11(5):682–694, 2003.

[16] A. B. Hur, D. Horn, H. T. Siegelmann, and V. Vapnik. A support vector method for clustering. In Todd, editor, *NIPS*, pages 367–373, 2000.

[17] A. B. Hur, D. Horn, H. T. Siegelmann, and V. Vapnik. Support vector clustering. *Journal of Machine Learning Research*, 2:125–137, 2001.

[18] A. K. Jain and R. C. Dubes. *Algorithms for clustering data*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988.

[19] L. Kaufman and P. J. Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley, 1990.

[20] R. Krishnapuram, A. Joshi, O. Nasraoui, and L. Yi. Low-complexity fuzzy relational clustering algorithms for web mining. *IEEE Transactions on Fuzzy Systems*, 9(4):595–607, 2001.

[21] R. Krishnapuram and J. M. Keller. A possibilistic approach to clustering. *IEEE Transactions on Fuzzy Systems*, 1(2):98–110, 1993.

[22] R. Krishnapuram and J. M. Keller. The possibilistic c-means algorithm: insights and recommendations. *IEEE Transactions on Fuzzy Systems*, 4(3):385–393, 1996.

[23] J. Laub and K. R. Müller. Feature discovery in non-metric pairwise data. *Journal of Machine Learning Research*, 5:801–818, 2004.

[24] J. Laub, V. Roth, J. M. Buhmann, and K. R. Müller. On the information and representation of non-euclidean pairwise data. *Pattern Recognition*, 39(10):1815–1826, 2006.

[25] S. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28:129–137, 1982.

[26] D. J. C. Mackay. *Information Theory, Inference & Learning Algorithms*. Cambridge University Press, June 2002.

[27] J. B. Macqueen. Some methods of classification and analysis of multivariate observations. In *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297, 1967.

[28] K. R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf. An introduction to kernel-based learning algorithms. *IEEE Transactions on Neural Networks*, 12(2):181–202, 2001.

[29] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In T. G. Dietterich, S. Becker, and Z. Ghahramani, editors, *Advances in Neural Information Processing Systems 14*, Cambridge, MA, 2002. MIT Press.

[30] V. Roth, J. Laub, J. M. Buhmann, and K. R. Müller. Going metric: Denoising pairwise data. In *NIPS*, pages 817–824, 2002.

[31] V. Roth, J. Laub, M. Kawanabe, and J. M. Buhmann. Optimal cluster preserving embedding of nonmetric proximity data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(12):1540–1551, 2003.

[32] M. Roubens. Pattern classification problems and fuzzy sets. *Fuzzy Sets and Systems*, 1(4):239–253, October 1978.

[33] E. H. Ruspini. Numerical methods for fuzzy clustering. In D. Dubois, H. Prade, and R. R. Yager, editors, *Readings in Fuzzy Sets for Intelligent Systems*, pages 599–614. Kaufmann, San Mateo, CA, 1993.

[34] S. Saitoh. *Theory of Reproducing Kernels and its Applications.* Longman Scientific & Technical, Harlow, England, 1988.

[35] B. Schölkopf and A. J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond.* MIT Press, Cambridge, MA, USA, 2001.

[36] B. Schölkopf, A. J. Smola, and K. R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, 1998.

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

[38] D. M. J. Tax and R. P. W. Duin. Support vector domain description. *Pattern Recognition Letters*, 20(11-13):1191–1199, 1999.

[39] J. H. Ward. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58:236–244, 1963.

[40] M. Windham. Numerical classification of proximity data with assignment measures. *Journal of Classification*, 2(1):157–172, December 1985.

[41] D. Q. Zhang and S. C. Chen. Fuzzy clustering using kernel method. In *The 2002 International Conference on Control and Automation, 2002. ICCA*, pages 162–163, 2002.