Fuzzy Clustering of Patterns Represented by Pairwise Dissimilarities

Maurizio Filippone
Department of Information and Software Engineering, George Mason University,
4400 University Drive, Fairfax, Virginia 22030, and
Department of Computer and Information Science, University of Genova,
Via Dodecaneso 35, I-16146 Genova, Italy
email - filippone@disi.unige.it

October 2007

Technical Report ISE-TR-07-05

Abstract

Clustering is the problem of grouping objects on the basis of a similarity measure between them. This paper considers the approaches belonging to the K-means family, in particular those based on fuzzy memberships. When patterns are represented by means of non-metric pairwise dissimilarities, these methods cannot be directly applied, since they are not guaranteed to converge. Symmetrization and shift operations have been proposed, to transform the dissimilarities between patterns from non-metric to metric. It has been shown that they modify the K-means objective function by a constant, that does not influence the optimization procedure. Some fuzzy clustering algorithms have been extended, in order to handle patterns described by means of pairwise dissimilarities. The literature, however, lacks of an explicit analysis on what happens to K-means style fuzzy clustering algorithms, when the dissimilarities are transformed to let them become metric. This paper shows how the objective functions of four clustering algorithms based on fuzzy memberships change, due to dissimilarities transformations. The experimental analysis conducted on a synthetic and a real data set shows the effect of the dissimilarities transformations for four clustering algorithms based on fuzzy memberships.

1 Introduction

Clustering is the problem of grouping objects on the basis of a similarity measure between them. It occurs very often in different disciplines and fields; this is the reason why several approaches have been proposed. Clustering algorithms can be roughly divided in two categories: hierarchical and partitional. Hierarchical clustering techniques [12, 26, 27] are able to find structures which can be further divided into substructures and so on recursively. The result is a hierarchical structure of groups known as *dendrogram*.

Partitioning clustering methods try to obtain a single partition of data and are often based on the optimization of an appropriate objective function. The result of such clustering algorithms is the creation of hypersurfaces separating groups of patterns. This paper considers the approaches belonging to the K-means [18, 19] family, in particular those based on fuzzy memberships [3, 4, 14, 15]. In this context, a pattern can belong to more than one cluster with different degrees. This allows to better describe 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.

All the K-means style 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. Some fuzzy clustering algorithms have been extended in order to handle patterns described by means of pairwise dissimilarities, in particular: Fuzzy relational clustering [10] and Possibilistic relational clustering [6]. For such clustering algorithms, the concept of centroids looses its meaning, since the patterns are not described in terms of features. Moreover, if the dissimilarities are not metric, the convergence of the algorithms is not guaranteed. In Ref. [13], the authors propose a fuzzy relational algorithm that looks for the centroids among the objects composing the data set.

Some approaches have been proposed to transform the dissimilarities between patterns from non-metric to metric, to cope with the convergence problem. Non-metric dissimilarities are not symmetric, and do not obey to the triangular inequality. The transformations needed to let the dissimilarities 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. 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 objective function [21, 22]. In other words, changing the dissimilarities with their transformed versions does not reflect any changes on the objective function. In fact, it changes by a constant that does not affect the optimization. Once the dissimilarities are metric, they can be considered as pairwise squared Euclidean distances between patterns. This is the link with clustering methods using positive semidefinite kernels. Such kernels can be obtained by the dissimilarity matrix, and each entry is a scalar product between vectors representing the original objects. These are called embedding vectors, and are not computed explicitly. The pairwise scalar products contain enough information to let to apply the K-means family algorithms on the embedding vectors. This corresponds to the clustering in feature space [7].

A fuzzy clustering dealing with non-Euclidean dissimilarities can be found in Ref. [9]. The literature, however, lacks of an explicit analysis on what happens to 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 dissimilarities transformations.

The considered clustering algorithms based on fuzzy memberships are: Fuzzy c-means I (FCM I) [4], Fuzzy c-means II (FCM II) [3], Possibilistic c-means I (PCM I) [14], and Possibilistic c-means II (PCM II) [15]. 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 dual of the four algorithms. FCM II and PCM I in feature space have never been proposed before, while FCM I and PCM II in feature space can be found in Refs. [28] and [8]. The relational duals of FCM I and PCM I have been proposed in Ref. [10] and [6]; the non-Euclidean case is studied in Ref. [9] for FCM I. The relational dual of FCM II and PCM II have never been proposed before. The experimental analysis conducted in this paper on a synthetic and a real data set shows the effect of the dissimilarities transformations on the four considered clustering algorithms.

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 K-Means style fuzzy 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 appendix.

1.1 How to Embed 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 $e = \{1, 1, \ldots, 1\}^T$ and I the $n \times n$ identity matrix. Eq. 3 is equivalent to:

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

The natural question arises: how can we choose α to guarantee that $\tilde{r}$ satisfies the triangular inequality? The answer is in a theorem that can be found in Refs. [16, 22]. In this Section the theorem is reported, while the proof can be found in App. A.2.

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

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

It's clear from Eq. 5 that S is not uniquely determined by R. All the matrices $S + \alpha ee^T$, for instance, lead to the same matrix R no matter what α is. It can be proved, however, that the centralized version of S is uniquely determined by R (see App. A.1):

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

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. A.2. 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 guaranteed to be 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:

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

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

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

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 distances, 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 and the theory of kernel methods. $\tilde{S^c}$ can be interpreted as the Gram matrix that is used in kernel algorithms. In Ref. [16, 17] the authors give an interpretation of the negative eigenvalues of S^c.

1.2 Mercer Kernels

A kernel function $K : X \times X \rightarrow \mathbb{R}$ is called a *positive definite kernel* (or *Mercer kernel*) if and only if K is symmetric and positive semidefinite [2, 23]. Each Mercer kernel can be expressed as follows:

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

where $\Phi : X \rightarrow \mathcal{F}$ performs a mapping from the input space X to $\mathcal{F}$ which is called *feature space*. A well known result shows that it is not necessary to know Φ to compute the distances in feature space:

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

This is the so called *distance kernel trick* [20, 24].

Kernels have been using 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 [25]. In Support Vector Machines [5], one takes advantage of this mapping to solve a classification problem in a high dimensional feature spaces. Clustering methods in feature space should take the advantage of the mapping to solve a clustering problem where the cluster structure is more evident. 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 hot it is possible to obtain a formulation of the K-means style fuzzy clustering algorithms, knowing just K. Since Φ is unknown, it will not be possible to know the prototypes of the clusters, that will be points in the space $\mathcal{F}$.

1.3 Preshift and Postshift

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 *preshift*:

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

The second choice, the *postshift*, 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 S corresponding to the same shift of the distances, but:

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

App. A.3 shows that the choice between preshift and postshift does not affect the studied clustering algorithms.

2 K-means Family Algorithms Objective Functions

The algorithms belonging to the K-means family 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 [14]. Given a set of patterns X, the set of centroids $V = \{\mathbf{v}_1, \ldots, \mathbf{v}_c\}$ 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 where each element u_{ih} represents 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 constraint to sum up to one:

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

In the possibilistic paradigm, the memberships are not subject to any constraint, and can be interpreted as a degree of typicality.

In general, all the K-means family algorithms are based on the minimization of a functional composed of 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 the U, the Lagrange multipliers technique has to be followed in order to perform the optimization. This means that a further term, depending only from the U, must be added to $J(U, V)$. The Lagrangian associated to the optimization problem can be introduced:

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

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 [11]. The algorithms stop when a convergence criterium 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, the update of the $\mathbf{v}_i$ is the same for all the considered algorithms. From Eq. 20:

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

Now it is possible to prove that the following functional is equivalent to $G(U, V)$ (see appendix A.4):

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

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:

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

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 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:

$$\hat{r}_{ij} = \frac{r_{ij} + r_{ji}}{2} \tag{26}$$

which is equivalent to:

$$\hat{R} = \frac{R + R^T}{2} \tag{27}$$

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

$$\begin{aligned}\sum_{h=1}^{n}\sum_{k=1}^{n} u_{ih}^{\theta}u_{ik}^{\theta}\hat{r}_{hk} &= \frac{1}{2}\sum_{h=1}^{n}\sum_{k=1}^{n} u_{ih}^{\theta}u_{ik}^{\theta}r_{hk} + \frac{1}{2}\sum_{h=1}^{n}\sum_{k=1}^{n} u_{ih}^{\theta}u_{ik}^{\theta}r_{kh} \\ &= \sum_{h=1}^{n}\sum_{k=1}^{n} u_{ih}^{\theta}u_{ik}^{\theta}r_{hk} \end{aligned} \tag{28}$$

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. 26 does not change the clustering objective function. In force of this result, R will be considered symmetric in the rest of this paper.

2.2 Transformation of $G(U)$ to Shifts Operations

This Section analyzes what happens to the Lagrangian L when transforming the distances in the following way:

$$\tilde{r}_{hk} = r_{hk} + 2\alpha \quad \forall\, h \neq k \tag{29}$$

which is equivalent to Eq. 4:

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

$$\begin{aligned} G_{\alpha}(U) &= \sum_{i=1}^{c} \frac{\sum_{h=1}^{n}\sum_{k=1}^{n} u_{ih}^{\theta}u_{ik}^{\theta}\tilde{r}_{hk}}{\sum_{h=1}^{n} u_{ih}^{\theta}} \\ &= G(U) + 2\alpha \sum_{i=1}^{c} \frac{\sum_{h=1}^{n}\sum_{k=1}^{n} u_{ih}^{\theta}u_{ik}^{\theta} - \sum_{h=1}^{n} u_{ik}^{2\theta}}{\sum_{h=1}^{n} u_{ih}^{\theta}} \\ &= G(U) + 2\alpha \sum_{i=1}^{c}\sum_{h=1}^{n} u_{ih}^{\theta} - 2\alpha \sum_{i=1}^{c} \frac{\sum_{h=1}^{n} u_{ih}^{2\theta}}{\sum_{h=1}^{n} u_{ih}^{\theta}} \end{aligned} \tag{30}$$

The Lagrangian will result in:

$$L_{\alpha}(U) = G(U) + H(U) + W(U) + 2\alpha\,(A(U) - B(U)) \tag{31}$$

This result shows that in general the Lagrangian for the K-means family 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. Besides, for fuzzy clustering algorithms for which $\theta = 1$, $A(U)$ reduces to n.

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

$$A(U) = \sum_{i=1}^{c} \sum_{h=1}^{n} u_{ih}^{\theta} < n \tag{32}$$

$$B(U) = \sum_{i=1}^{c} \frac{\sum_{h=1}^{n} u_{ih}^{2\theta}}{\sum_{h=1}^{n} u_{ih}^{\theta}} < c \tag{33}$$

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.

Given a clustering algorithm, in order to obtain the update of the memberships, the derivatives of the Lagrangian with respect to them have to be set to zero. From that, an update formula for the memberships has to be obtained (in presence of constraints, this implies to compute also the value of the Lagrange multipliers). Let's consider the term $B(U)$:

$$\frac{\partial B(U)}{\partial u_{ih}} = \frac{2\theta u_{ih}^{2\theta-1} \sum_{r=1}^{n} u_{ir}^{\theta} - \theta u_{ih}^{\theta-1} \sum_{h=1}^{n} u_{ih}^{2\theta}}{(\sum_{h=1}^{n} u_{ih}^{\theta})^2} \tag{34}$$

This is not easily invertible when summed to the derivative of the other terms to obtain zero. The next Section provides an experimental analysis showing the effect of the shift operation on the behavior of the memberships during the optimization.

2.3 Analysis of Four Clustering Algorithms

This Section shows the results just obtained to four 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) [14], and Possibilistic c-means II (PCM II) [15] (see App. A.5 for the complete derivation of these four algorithms). In Tab. 1, the terms of the Lagrangian in Eq. 19 for the mentioned clustering algorithms are resumed. Since the sum of the memberships of a point to all the clusters is constrained to be one in fuzzy clustering, the term $W(U)$ is introduced. For the possibilistic algorithms $W(U) = 0$, since the memberships are not constrained. In fact, for these algorithms the minimization of $L(U)$ should be done in the hypercube $u_{ih} \in [0,1]$. Since the form assumed by the update equations, 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.

Tab. 2 resumes the Lagrangian $L_\alpha(U)$ of the discussed clustering algorithms, considering also the effect of the shift. K-means is invariant to distance shifts since $A(U) = n$ and $B(U) = c$. In FCM II and PCM II, $A(U) = n$; in FCM I and PCM I, both $A(U)$ and $B(U)$ are not zero.

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

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

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

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

Table 2: Resuming table of the objective functions, for the considered clustering algorithms.

explicitly found. Each pattern is represented by a vector $\mathbf{x}_i \in \mathcal{F}$ and the set of centroids V is composed of prototypes in $\mathcal{F}$. Let's analyze, for instance, the update equations for $\mathbf{v}_i$ and u_{ih} for FCM II:

$$u_{ih} = \frac{\exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2}{\lambda}\right)}{\sum_{j=1}^{c} \exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_j\|^2}{\lambda}\right)} \tag{35}$$

$$\mathbf{v}_i = \frac{\sum_{h=1}^{n} u_{ih}\mathbf{x}_h}{\sum_{h=1}^{n} u_{ih}} \tag{36}$$

Since we don't know explicitly the vectors $\mathbf{x}_i$, it would not be possible to explicitly compute $\mathbf{v}_i$. Substituting Eq. 36 in Eq. 35, we obtain:

$$\begin{aligned}
\|\mathbf{x}_h - \mathbf{v}_i\|^2 &= \left\| \mathbf{x}_h - \frac{\sum_{r=1}^{n} u_{ir}\mathbf{x}_r}{\sum_{r=1}^{n} u_{ir}} \right\|^2 \\
&= \left(\mathbf{x}_h - \frac{\sum_{r=1}^{n} u_{ir}\mathbf{x}_r}{\sum_{r=1}^{n} u_{ir}}\right)\left(\mathbf{x}_h - \frac{\sum_{r=1}^{n} u_{ir}\mathbf{x}_r}{\sum_{r=1}^{n} u_{ir}}\right) \\
&= \mathbf{x}_h\mathbf{x}_h - 2\frac{\sum_{r=1}^{n} u_{ir}\mathbf{x}_r\mathbf{x}_h}{\sum_{r=1}^{n} u_{ir}} + \frac{\sum_{r=1}^{n}\sum_{s=1}^{n} u_{ir}u_{is}\mathbf{x}_r\mathbf{x}_s}{(\sum_{r=1}^{n} u_{ir})^2} \\
&= k_{hh} - 2\frac{\sum_{r=1}^{n} u_{ir}k_{rh}}{\sum_{r=1}^{n} u_{ir}} + \frac{\sum_{r=1}^{n}\sum_{s=1}^{n} u_{ir}u_{is}k_{rs}}{(\sum_{r=1}^{n} u_{ir})^2}
\end{aligned} \tag{37}$$

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:

$$a_i = \left(\sum_{h=1}^{n} u_{ih}^{\theta}\right)^{-1} \tag{38}$$

$$\mathbf{z}^{(0)} = \mathrm{diag}(K) \tag{39}$$

$$Z^{(1)} = U_\theta K \tag{40}$$

$$\mathbf{z}^{(2)} = \mathrm{diag}(U_\theta K U_\theta^T) \tag{41}$$

Eq. 37 becomes:

$$\|\mathbf{x}_h - \mathbf{v}_i\|^2 = z_h^{(0)} - 2a_i z_{ih}^{(1)} + a_i^2 z_i^{(2)} \tag{42}$$

Tab. 3 shows the update equations of the memberships for the considered clustering algorithms. Tab. 4 shows the steps composing the considered clustering algorithms.

3 Experimental Analysis

3.1 Synthetic Data Set

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

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 $u_{ih} = \exp\left(-\frac{z_h^{(0)} - 2a_i z_{ih}^{(1)} + a_i^2 z_i^{(2)}}{\eta_i} \right)$

Table 3: Resuming table of the memberships update equations, for the considered clustering algorithms.

Table 4: Pseudocode of the considered clustering algorithms

1. **if** R is not symmetric, **then** symmetrize it using Eq. 26;
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), λ (FCM II), η_i (PCM I, PCM II);
6. Initialize U;
7. Update U using the update equation in Tab. 3 corresponding to the chosen method;
8. **if** the convergence criteria is not satisfied **then** go to step 7;
9. **else** stop.

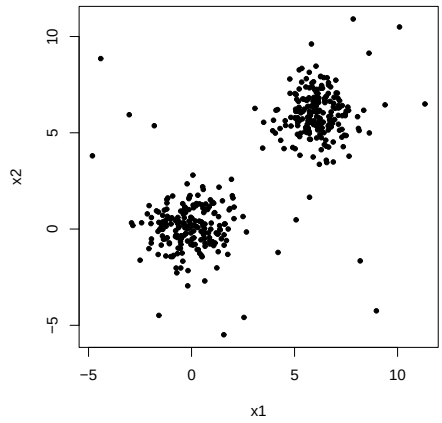


Figure 1: Plot of the synthetic data set composed of two clusters and some outliers.

bution. The position of their centers are respectively in $(0, 0)$ and $(6, 6)$, and the standard deviations are equal to one for both the features and clusters. Twenty outlier points have been added; they have been extracted with a uniform distribution in the set $[-6, 12] \times [-6, 12]$. The average of the squared distances is 43.4, the median is 34.4, and the maximum is 360.9.

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 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. This can be done in two equivalent ways, namely the preshift and the postshift (see App. A.3). 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 these two scores:

$$\mathrm{sd}(U - U') = \sqrt{\left(\frac{\sum_{h=1}^{n} \sum_{i=1}^{c} (u_{ih} - u'_{ih})^2}{c\, n} \right)} \tag{43}$$

$$\max(U - U') = \max_{i,h} \left(|u_{ih} - u'_{ih}| \right) \tag{44}$$

averaged over 100 runs.

FCM I has been tried three different values of m, in particular $m = 1.1$, 1.5, 2. Fig. 2 shows the behavior of the memberships during the optimization for different values of α and m. The first row in Fig. 2 corresponds to $m = 2$, the one in the middle to $m = 1.5$, and the one on the bottom to $m = 1.1$. For small α the results are almost invariant as expected. For values of α of the order of the average of the squared distances, the memberships have a very different behavior with respect to those on the original set. Reducing the fuzziness m it can be noticed that the results are better. This is not surprising since for m tending to 1, FCM I behaves like K-means which is invariant to shift

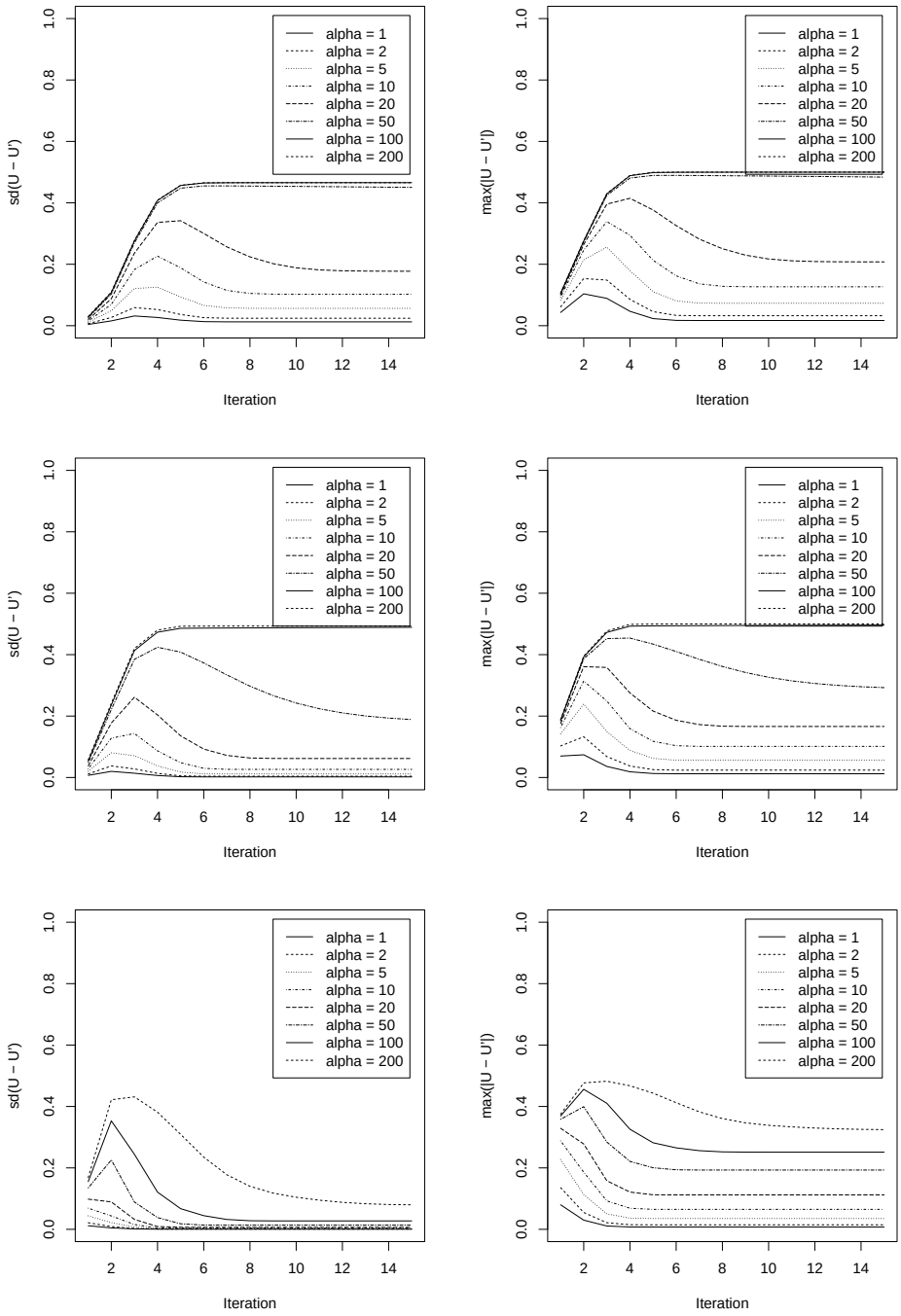


Figure 2: FCM I - Behavior of the memberships during the optimization for different values of α. First row $m = 2$, second row $m = 1.5$, third row $m = 1.1$. Results are averaged over 100 repetitions with different initialization of U.

transformations. 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 for different values of alpha have been found to be identical for all the tested valued of α.

FCM II has been tried with three different values of λ, in particular $\lambda = 10, 20, 30$. For such values of λ, the resulting memberships range from almost crisp to moderate fuzzy. For different fuzziness levels (higher λ leads to fuzzier solutions), the memberships are almost invariant, even for values of α higher than the maximum of the original squared distances (Fig. 3). The Lagrangian in FCM II is not invariant to shift transformations only because of the term $B(U)$. The fact that $A(U)$ is constant gives to FCM II more robustness to distance shifts.

PCM I Fig. 4 shows the behavior of the memberships during the optimization for the PCM I with different values of m, in particular $m = 1.1, 1.5, 2$. The initialization of the memberships has been done using the result obtained by the FCM II, since it showed high robustness to distance shifts. The values of η_i have been computed on the basis of the memberships obtained by the FCM II. It can be seen that even for small values of α, the behavior of the memberships is strongly affected by the shift operation.

PCM II The initialization of the memberships and the computation of the η_i have been done on the basis of the result obtained by the FCM II. In PCM II there are no further parameters to set up. Fig. 5 shows that also PCM II is strongly affected by dissimilarities shifts, even for small values of α.

3.2 USPS Data Set

The studied algorithms have been tested on the USPS data set, which has been studied also in Refs. [24, 16]. It is composed of 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. [16], 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. [16] is based on the Simpson score, which is a matching function between binary images. Given two binary images, the following matrix can be constructed:

		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:

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

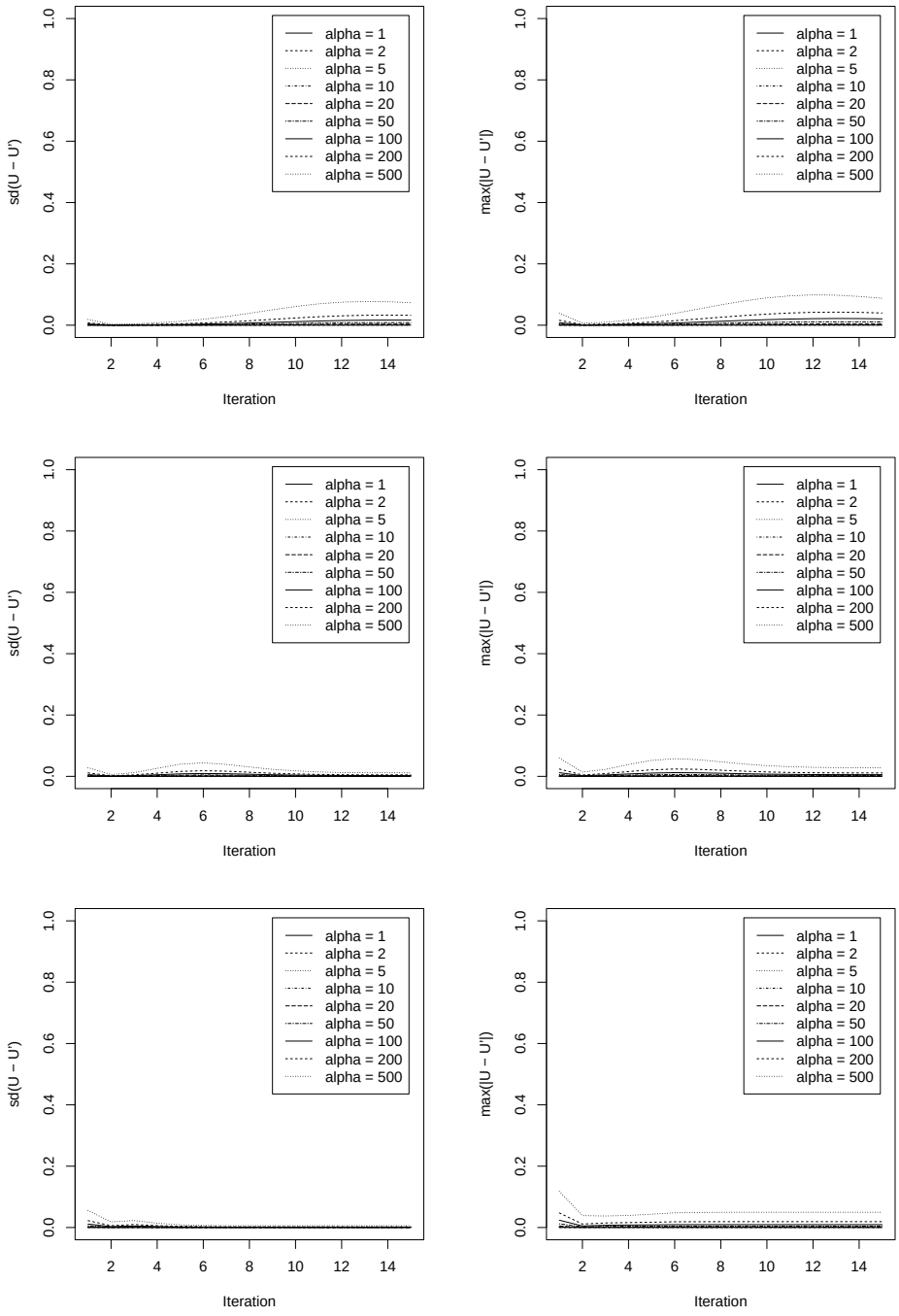


Figure 3: FCM II - Behavior of the memberships during the optimization for different values of α. First row $\lambda = 30$, second row $\lambda = 20$, third row $\lambda = 10$. Results are averaged over 100 repetitions with different initialization of U.

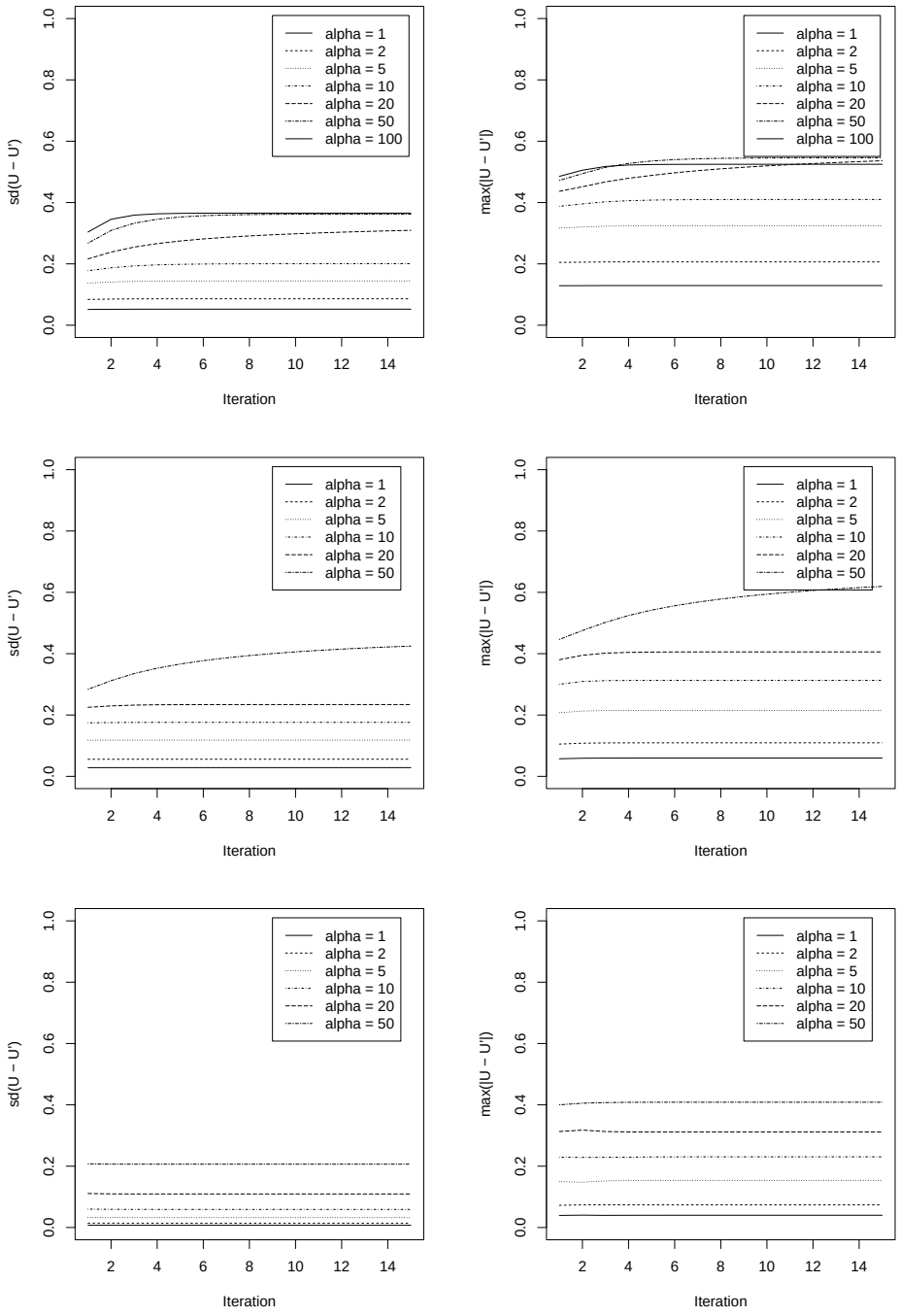


Figure 4: PCM I - Behavior of the memberships during the optimization for different values of α. First row $m = 2$, second row $m = 1.5$, third row $m = 1$. Results are averaged over 100 repetitions with different initialization of U.

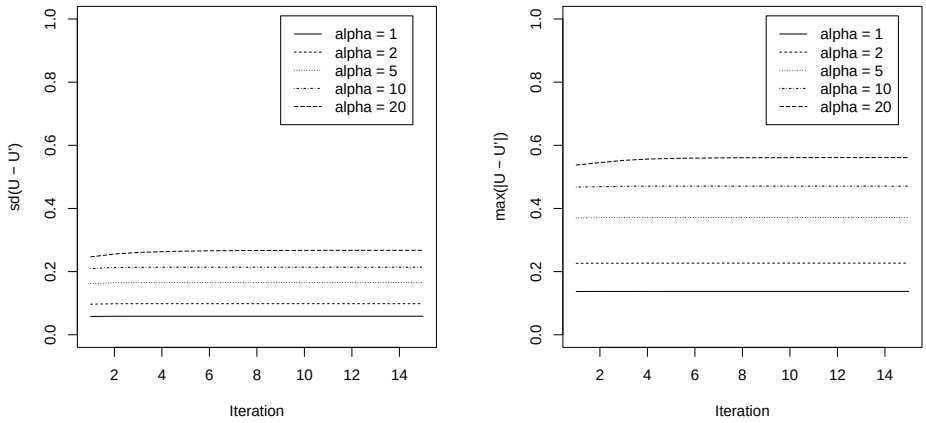


Figure 5: PCM II - Behavior of the memberships during the optimization for different values of α. Results are averaged over 100 repetitions with different initialization of U.

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 Simpson dissimilarity is symmetric, but does not obey to the triangular inequality. Indeed, as can be seen in Fig. 6, 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. FCM II has been applied on the selected binary images, with $\lambda = 0.2$ and looking for 2 clusters. Fig. 7 shows 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. In the area of the images where the memberships are low, some images are misclassified. A thorough analysis shows that the characters that have been assigned to the wrong clusters have, in fact, their membership shared almost equally between the two clusters.

4 Conclusions

In this paper, four 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 Lagrangian. The main results include the proof of the invariance of the Lagrangian to symmetrization and the lack of invariance to shift operations.

The tests conducted on a synthetic data set show that FCM II, among the studied algorithms, is the least sensitive to shift operations. 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

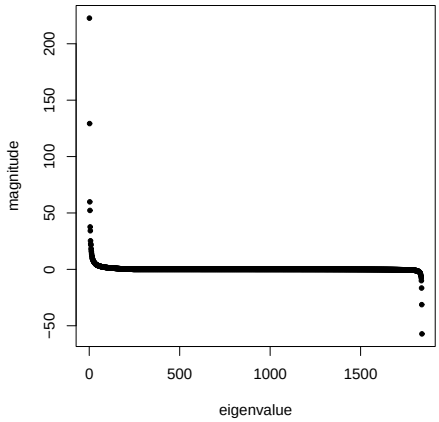


Figure 6: USPS data set - Eigenvalues of the matrix S^c sorted by decreasing magnitude.

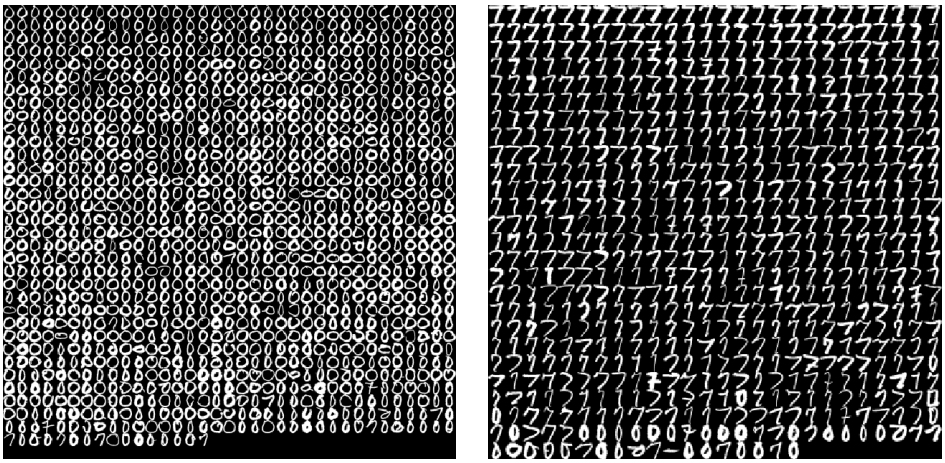


Figure 7: Clusters found by FCM II. The image in the top-left corner has the highest membership value. Moving right, the memberships decrease.

$A(U)$ 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. 84, 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. In both FCM I and FCM II, the defuzzification of the membership produced the same cluster labels, even for large shifts. These experimental results suggest that FCM I and FCM II could be useful to perform the optimization, to obtain the cluster labels; the value of the memberships are distorted by the shift, though. Shift operations affects mostly PCM I and PCM II. Even for moderate values of α, the memberships assume very different values with respect to the unshifted case. Small distances are more affected by the sum of a constant than large distances. The lack of a probabilistic constraint leads to the inability of the possibilistic algorithms to handle sparse data set [15]. The algorithm considers all the data set as a single cluster, and the centroids collapse into a single one. This could be the reason for this strong distortion after shift operations. From the results on handwritten character recognition problem, it is possible to see how FCM II performed in a real scenario. A simple analysis on the memberships can help to avoid a decision on the assignment of patterns having their membership almost equally shared among clusters.

Other interesting studies could involve the effect of the cardinality of the data set n and the number of clusters c. It would be also interesting to try different approaches for the estimation of η_i, as suggested in Ref. [6], or see what is the difference between the behavior of memberships associated to outlier and normal patterns. All these considerations could be the basis of new studies on the behavior of the studied clustering algorithms, for patterns described by non-metric pairwise dissimilarities.

Acknowledgments

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

A Appendix

A.1 Proof that S^c is Uniquely Determined by R^c

The centralized version of a generic matrix P is the following:

$$P^c = QPQ \tag{47}$$

This is equivalent to:

$$p^c_{ij} = p_{ij} - \frac{1}{n}\sum_{h=1}^{n} p_{hj} - \frac{1}{n}\sum_{k=1}^{n} p_{ik} + \frac{1}{n^2}\sum_{h=1}^{n}\sum_{k=1}^{n} p_{hk} \tag{48}$$

Inverting Eq. 5, we can write:

$$s_{ij} = -\frac{1}{2}\left(r_{ij} - s_{ii} - s_{jj}\right) \tag{49}$$

The centralized version of S is:

$$\begin{aligned} s_{ij}^c &= -\frac{1}{2}\left[(r_{ij} - s_{ii} - s_{jj}) - \frac{1}{n}\sum_{h=1}^{n}(r_{hj} - s_{hh} - s_{jj}) - \frac{1}{n}\sum_{k=1}^{n}(r_{ik} - s_{ii} - s_{kk})\right. \\ &\quad \left. + \frac{1}{n^2}\sum_{h=1}^{n}\sum_{k=1}^{n}(r_{hk} - s_{hh} - s_{kk})\right] \qquad (50) \\ &= -\frac{1}{2}\left(r_{ij} - \frac{1}{n}\sum_{h=1}^{n} r_{hj} - \frac{1}{n}\sum_{k=1}^{n} r_{ik} + \frac{1}{n^2}\sum_{h=1}^{n}\sum_{k=1}^{n} r_{hk}\right) \qquad (51) \end{aligned}$$

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

$$S^c = -\frac{1}{2}R^c \qquad (52)$$

A.2 Proof of Theorem 1.1

In this section we provide the proof that R is a squared Euclidean distance matrix $\Longleftrightarrow S^c \succeq 0$. Let's start with $\Rightarrow$. The centralized version of R is:

$$R^c = QRQ = R - \frac{1}{n}ee^T R - \frac{1}{n}Ree^T + \frac{1}{n^2}ee^T Ree^T \qquad (53)$$

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

$$r_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|^2 \qquad (54)$$

the elements of R^c can be written:

$$\begin{aligned} r_{ij}^c &= \|\mathbf{x}_i - \mathbf{x}_j\|^2 - \frac{1}{n}\sum_{h=1}^{n}\|\mathbf{x}_h - \mathbf{x}_j\|^2 - \frac{1}{n}\sum_{k=1}^{n}\|\mathbf{x}_i - \mathbf{x}_k\|^2 + \frac{1}{n^2}\sum_{h=1}^{n}\sum_{k=1}^{n}\|\mathbf{x}_h - \mathbf{x}_k\|^2 \\ &= \mathbf{x}_i\mathbf{x}_i + \mathbf{x}_j\mathbf{x}_j - 2\mathbf{x}_i\mathbf{x}_j - \frac{1}{n}\left(\sum_{h=1}^{n}\mathbf{x}_h\mathbf{x}_h + \mathbf{x}_j\mathbf{x}_j - 2\mathbf{x}_h\mathbf{x}_j\right) - \frac{1}{n}\left(\sum_{k=1}^{n}\mathbf{x}_i\mathbf{x}_i + \mathbf{x}_k\mathbf{x}_k - 2\mathbf{x}_i\mathbf{x}_k\right) \\ &\quad + \frac{1}{n^2}\left(\sum_{h=1}^{n}\sum_{k=1}^{n}\mathbf{x}_h\mathbf{x}_h + \mathbf{x}_k\mathbf{x}_k - 2\mathbf{x}_h\mathbf{x}_k\right) \\ &= -2\left(\mathbf{x}_i\mathbf{x}_j - \frac{1}{n}\sum_{h=1}^{n}\mathbf{x}_h\mathbf{x}_j - \frac{1}{n}\sum_{k=1}^{n}\mathbf{x}_i\mathbf{x}_k + \frac{1}{n^2}\sum_{h=1}^{n}\sum_{k=1}^{n}\mathbf{x}_h\mathbf{x}_k\right) \qquad (55) \end{aligned}$$

Introducing the quantity:

$$\bar{\mathbf{x}} = \frac{1}{n}\sum_{h=1}^{n}\mathbf{x}_h \qquad (56)$$

we can rewrite in a more compact way Eq. 55:

$$r_{ij}^c = -2(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_j - \bar{\mathbf{x}}) = -2\breve{\mathbf{x}}_i\breve{\mathbf{x}}_j \qquad (57)$$

This is equivalent to say that:

$$S^c = \breve{X}\breve{X}^T \tag{58}$$

which proves $\Rightarrow$.

To prove $\Leftarrow$, since S^c is positive semidefinite, we can write:

$$S^c = XX^T \tag{59}$$

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

$$\begin{aligned} r_{ij} &= s_{ii} + s_{jj} - 2s_{ij} \\ &= \mathbf{x}_i\mathbf{x}_i + \mathbf{x}_j\mathbf{x}_j - 2\mathbf{x}_i\mathbf{x}_j \\ &= \|\mathbf{x}_i - \mathbf{x}_j\|^2 \end{aligned} \tag{60}$$

This proves $\Leftarrow$.

A.3 Preshift and Postshift

Let's analyze why:

$$S^c + \alpha I \neq -\frac{1}{2}(Q\tilde{R}Q) \tag{61}$$

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 preshift we have:

$$-\frac{1}{2}(Q\tilde{R}Q) = -\frac{1}{2}(QRQ) - \alpha Q(ee^T - I)Q = S^c - \alpha Q(ee^T - I)Q \tag{62}$$

Now:

$$Q(ee^T - I)Q = Qee^TQ - QQ = -QQ = -Q \tag{63}$$

since:

$$Qe = (I - \frac{1}{n}ee^T)e = e - e = \mathbf{0} \tag{64}$$

and:

$$QQ = (I - \frac{1}{n}ee^T)(I - \frac{1}{n}ee^T) = I - \frac{2}{n}ee^T + \frac{1}{n^2}ee^Tee^T = I - \frac{1}{n}ee^T = Q \tag{65}$$

Thus:

$$-\frac{1}{2}(Q\tilde{R}Q) = S^c + \alpha Q \tag{66}$$

The difference between the matrices associated to postshift and preshift is:

$$\alpha(I - Q) = \frac{\alpha}{n}ee^T \tag{67}$$

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

$$\|\mathbf{x}_h - \mathbf{v}_j\|^2 = k'_{hh} - 2\frac{\sum_{r=1}^n u_{ir}^\theta k'_{rh}}{\sum_{r=1}^n u_{ir}^\theta} + \frac{\sum_{r=1}^n \sum_{s=1}^n u_{ir}^\theta u_{is}^\theta k'_{rs}}{(\sum_{r=1}^n u_{ir}^\theta)^2} \tag{68}$$

$$k' = k + \frac{\alpha}{n} \tag{69}$$

$$\|\mathbf{x}_h - \mathbf{v}_j\|^2 = k_{hh} + \frac{\alpha}{n} - 2\frac{\sum_{r=1}^n u_{ir}^\theta k_{rh}}{\sum_{r=1}^n u_{ir}^\theta} - 2\frac{\alpha}{n} + \frac{\sum_{r=1}^n \sum_{s=1}^n u_{ir}^\theta u_{is}^\theta k_{rs}}{(\sum_{r=1}^n u_{ir}^\theta)^2} + \frac{\alpha}{n} \tag{70}$$

A.4 Proof of Equivalence between $G(U,V)$ and $G(U)$

To prove the equivalence between the distortion functions in Eqs. 18 and 24, let's introduce the quantity:

$$b_i = \sum_{h=1}^{n} u_{ih}^{\theta} \tag{71}$$

Since:

$$\mathbf{v}_i = \frac{\sum_{h=1}^{n} u_{ih}^{\theta} \mathbf{x}_h}{\sum_{h=1}^{n} u_{ih}^{\theta}} \tag{72}$$

part of the sum in $G(U,V)$ can be rewritten in the following way:

$$\begin{aligned}
\sum_{h=1}^{n} u_{ih}^{\theta} \|\mathbf{x}_h - \mathbf{v}_i\|^2 &= \sum_{h=1}^{n} u_{ih}^{\theta} (\mathbf{x}_h - \mathbf{v}_i)(\mathbf{x}_h - \mathbf{v}_i) \\
&= \sum_{h=1}^{n} u_{ih}^{\theta} (\mathbf{x}_h \mathbf{x}_h + \mathbf{v}_i \mathbf{v}_i - 2\mathbf{x}_h \mathbf{v}_i) \\
&= \sum_{h=1}^{n} u_{ih}^{\theta} \mathbf{x}_h \mathbf{x}_h + \sum_{h=1}^{n} u_{ih}^{\theta} \mathbf{v}_i \mathbf{v}_i - 2 \sum_{h=1}^{n} u_{ih}^{\theta} \mathbf{x}_h \mathbf{v}_i \\
&= \sum_{h=1}^{n} u_{ih}^{\theta} \mathbf{x}_h \mathbf{x}_h + b_i \mathbf{v}_i \mathbf{v}_i - 2 b_i \mathbf{v}_i \mathbf{v}_i \\
&= \sum_{h=1}^{n} u_{ih}^{\theta} \mathbf{x}_h \mathbf{x}_h - b_i \mathbf{v}_i \mathbf{v}_i
\end{aligned} \tag{73}$$

Rewriting part of $G(U)$, we obtain:

$$\begin{aligned}
\sum_{r=1}^{n} \sum_{s=1}^{n} u_{ir}^{\theta} u_{is}^{\theta} \|\mathbf{x}_r - \mathbf{x}_s\|^2 &= \sum_{r=1}^{n} \sum_{s=1}^{n} u_{ir}^{\theta} u_{is}^{\theta} (\mathbf{x}_r - \mathbf{x}_s)(\mathbf{x}_r - \mathbf{x}_s) \\
&= \sum_{r=1}^{n} \sum_{s=1}^{n} u_{ir}^{\theta} u_{is}^{\theta} (\mathbf{x}_r \mathbf{x}_r + \mathbf{x}_s \mathbf{x}_s - 2\mathbf{x}_r \mathbf{x}_s) \\
&= \sum_{r=1}^{n} \sum_{s=1}^{n} u_{ir}^{\theta} u_{is}^{\theta} \mathbf{x}_r \mathbf{x}_r + \sum_{r=1}^{n} \sum_{s=1}^{n} u_{ir}^{\theta} u_{is}^{\theta} \mathbf{x}_s \mathbf{x}_s - 2 \sum_{r=1}^{n} \sum_{s=1}^{n} u_{ir}^{\theta} u_{is}^{\theta} \mathbf{x}_r \mathbf{x}_s \\
&= \sum_{r=1}^{n} u_{ir}^{\theta} \mathbf{x}_r \mathbf{x}_r \sum_{s=1}^{n} u_{is}^{\theta} + \sum_{s=1}^{n} u_{is}^{\theta} \mathbf{x}_s \mathbf{x}_s \sum_{r=1}^{n} u_{ir}^{\theta} - 2 \sum_{r=1}^{n} \mathbf{x}_r u_{ir}^{\theta} \sum_{s=1}^{n} u_{is}^{\theta} \mathbf{x}_s \\
&= 2 b_i \sum_{r=1}^{n} u_{ir}^{\theta} \mathbf{x}_r \mathbf{x}_r - 2 b_i^2 \mathbf{v}_i \mathbf{v}_i
\end{aligned} \tag{74}$$

This proves that $G(U,V) = G(U)$.

A.5 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.

A.5.1 Fuzzy c-means I

The Lagrangian $L(U)$ is introduced:

$$L(U,V) = \sum_{i=1}^{c}\sum_{h=1}^{n} u_{ih}^{m}\|\mathbf{x}_h - \mathbf{v}_i\|^2 + \sum_{h=1}^{n}\beta_h(1 - \sum_{i=1}^{c} u_{ih}) \tag{75}$$

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 the u_{ih}:

$$\frac{\partial L(U,V)}{\partial u_{ih}} = m u_{ih}^{m-1}\|\mathbf{x}_h - \mathbf{v}_i\|^2 - \beta_h = 0 \tag{76}$$

we obtain:

$$u_{ih} = \left(\frac{\beta_h}{m\|\mathbf{x}_h - \mathbf{v}_i\|^2}\right)^{\frac{1}{m-1}} \tag{77}$$

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

$$\sum_{i=1}^{c}\left(\frac{\beta_h}{m\|\mathbf{x}_h - \mathbf{v}_i\|^2}\right)^{\frac{1}{m-1}} = 1 \tag{78}$$

we can obtain the Lagrange multipliers:

$$\beta_h = \left[\sum_{i=1}^{c}\left(\frac{1}{m\|\mathbf{x}_h - \mathbf{v}_i\|^2}\right)^{\frac{1}{m-1}}\right]^{1-m} \tag{79}$$

Substituting Eq. 79 into Eq. 77, the equation for the update of the memberships u_{ih} can be obtained:

$$u_{ih}^{-1} = \sum_{j=1}^{c}\left(\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2}{\|\mathbf{x}_h - \mathbf{v}_j\|^2}\right)^{\frac{1}{m-1}} \tag{80}$$

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 $\mathbf{v}_i$:

$$\frac{\partial L(U,V)}{\partial \mathbf{v}_i} = -\sum_{h=1}^{n} u_{ih}^{m}\left(\mathbf{x}_h - \mathbf{v}_i\right) = 0 \tag{81}$$

obtaining:

$$\mathbf{v}_i = \frac{\sum_{h=1}^{n} u_{ih}^{m}\mathbf{x}_h}{\sum_{h=1}^{n} u_{ih}^{m}} \tag{82}$$

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 $B(U)$:

$$L_\alpha(U,V) = \sum_{h=1}^{n}\sum_{i=1}^{c} u_{ih}^m \|\mathbf{x}_h - \mathbf{v}_i\|^2 + \alpha \sum_{h=1}^{n}\sum_{i=1}^{c} u_{ih}^m + \sum_{h=1}^{n} \beta_h (1 - \sum_{i=1}^{c} u_{ih}) \tag{83}$$

Following the same procedure, we obtain that the update of the $\mathbf{v}$ is the same as in Eq. 82, but the update of the memberships is:

$$u_{ih}^{-1} = \sum_{j=1}^{c} \left(\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2 + \alpha}{\|\mathbf{x}_h - \mathbf{v}_j\|^2 + \alpha} \right)^{\frac{1}{m-1}} \tag{84}$$

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

A.5.2 Fuzzy c-means II

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

$$L(U,V) = \sum_{h=1}^{n}\sum_{i=1}^{c} u_{ih} \|\mathbf{x}_h - \mathbf{v}_i\|^2 + \lambda \sum_{h=1}^{n}\sum_{i=1}^{c} u_{ih} \ln(u_{ih}) + \sum_{h=1}^{n} \beta_h (1 - \sum_{i=1}^{c} u_{ih}) \tag{85}$$

The entropic term favors values of the memberships near zero or one (Fig. 8). Let's compute the derivative of $L(U,V)$ with respect to u_{ih}:

$$\frac{\partial L(U,V)}{\partial u_{ih}} = \|\mathbf{x}_h - \mathbf{v}_i\|^2 + \lambda(\ln(u_{ih}) + 1) - \beta_h = 0 \tag{86}$$

This leads to:

$$u_{ih} = \frac{1}{e} \exp\left(\frac{\beta_h}{\lambda}\right) \exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2}{\lambda}\right) \tag{87}$$

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

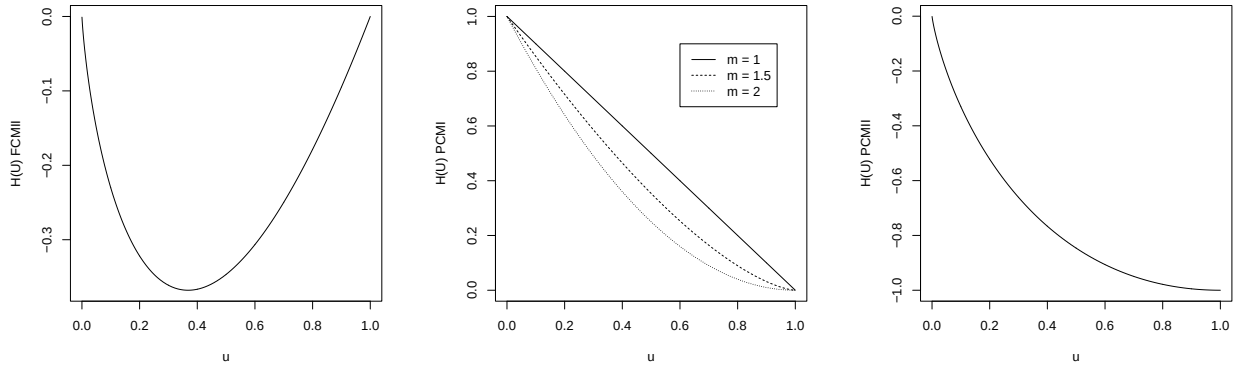
$$\sum_{i=1}^{c} \frac{1}{e} \exp\left(\frac{\beta_h}{\lambda}\right) \exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2}{\lambda}\right) = 1 \tag{88}$$

This allows to compute the Lagrange multipliers:

$$\beta_h = \lambda - \lambda \ln\left(\sum_{j=1}^{c} \exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_j\|^2}{\lambda}\right)\right) \tag{89}$$

Substituting Eq. 89 into Eq. 87, we obtain the equation for the update of the u_{ih}:

$$u_{ih} = \frac{\exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2}{\lambda}\right)}{\sum_{j=1}^{c} \exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_j\|^2}{\lambda}\right)} \tag{90}$$



(a) (b) (c)

Figure 8: (a) Plot of the FCM II entropy $H(u_{ih}) = u_{ih}\ln(u_{ih})$. (b) Plot of the PCM I entropy $H(u_{ih}) = (1-u_{ih})^m$ for increasing values of m. (c) Plot of the PCM II entropy $H(u_{ih}) = u_{ih}\ln(u_{ih}) - u_{ih}$.

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

$$\frac{\partial L(U,V)}{\partial \mathbf{v}_i} = -\sum_{h=1}^{n} u_{ih}\left(\mathbf{x}_h - \mathbf{v}_i\right) = 0 \tag{91}$$

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

$$\mathbf{v}_i = \frac{\sum_{h=1}^{n} u_{ih}\mathbf{x}_h}{\sum_{h=1}^{n} u_{ih}} \tag{92}$$

A.5.3 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:

$$L(U,V) = \sum_{h=1}^{n}\sum_{i=1}^{c} u_{ih}^m \|\mathbf{x}_h - \mathbf{v}_i\|^2 + \sum_{i=1}^{c}\eta_i \sum_{h=1}^{n}(1-u_{ih})^m \tag{93}$$

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}:

$$\frac{\partial L(U,V)}{\partial u_{ik}} = m u_{ih}^{m-1}(\|\mathbf{x}_h - \mathbf{v}_i\|^2) - \eta_i m(1-u_{ih})^{m-1} = 0 \tag{94}$$

We obtain directly the update equation:

$$u_{ih}^{-1} = \left(\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2}{\eta_i}\right)^{\frac{1}{m-1}} + 1 \tag{95}$$

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

$$\frac{\partial L(U,V)}{\partial \mathbf{v}_i} = -\sum_{h=1}^{n} u_{ih}^m \left(\mathbf{x}_h - \mathbf{v}_i\right) = 0 \tag{96}$$

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

$$\mathbf{v}_i = \frac{\sum_{h=1}^{n} u_{ih}^m \mathbf{x}_h}{\sum_{h=1}^{n} u_{ih}^m} \tag{97}$$

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

$$\eta_i = \gamma \frac{\sum_{h=1}^{n} \left(u_{ih}\right)^m \left\|\mathbf{x}_h - \mathbf{v}_i\right\|^2}{\sum_{h=1}^{n} \left(u_{ih}\right)^m} \tag{98}$$

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 $c \ll n$:

$$L_\alpha(U,V) = \sum_{h=1}^{n}\sum_{i=1}^{c} u_{ih}^m \|\mathbf{x}_h - \mathbf{v}_i\|^2 + \sum_{i=1}^{c} \eta_i \sum_{h=1}^{n} \left(1 - u_{ih}\right)^m + \alpha \sum_{h=1}^{n}\sum_{i=1}^{c} u_{ih}^m \tag{99}$$

Following the same procedure to derive the equations for the update of U:

$$u_{ih}^{-1} = \left(\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2 + \alpha}{\eta_i}\right)^{\frac{1}{m-1}} + 1 \tag{100}$$

For large values of α the memberships tend to be small.

A.5.4 Possibilistic c-means II

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

$$L(U,V) = \sum_{h=1}^{n}\sum_{i=1}^{c} u_{ih} \|\mathbf{x}_h - \mathbf{v}_i\|^2 + \sum_{i=1}^{c} \eta_i \sum_{h=1}^{n} \left(u_{ih} \ln(u_{ih}) - u_{ih}\right) \tag{101}$$

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}:

$$\frac{\partial L(U,V)}{\partial u_{ik}} = \|\mathbf{x}_h - \mathbf{v}_i\|^2 + \eta_i \ln(u_{ih}) = 0 \tag{102}$$

we obtain:

$$u_{ik} = \exp\left(-\frac{\|\mathbf{x}_h - \mathbf{v}_i\|^2}{\eta_i}\right) \tag{103}$$

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

$$\frac{\partial L(U,V)}{\partial \mathbf{v}_i} = -\sum_{h=1}^{n} u_{ih} \left(\mathbf{x}_h - \mathbf{v}_i\right) = 0 \tag{104}$$

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

$$\mathbf{v}_i = \frac{\sum_{h=1}^{n} u_{ih}\mathbf{x}_h}{\sum_{h=1}^{n} u_{ih}} \tag{105}$$

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] 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.

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

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

[10] 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.

[11] 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.

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

[13] 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.

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

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

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

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

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

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

[20] 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.

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

[22] 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.

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

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

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

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

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

[28] 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.