Modeling Proportional Membership in Fuzzy Clustering

Susana Nascimento, Boris Mirkin, and Fernando Moura-Pires

***Abstract*—To provide feedback from a cluster structure to the data from which it has been determined, we propose a framework for mining typological structures based on a fuzzy clustering model of how the data are generated from a cluster structure. To relate data entities to cluster prototypes, we assume that the observed entities share parts of the prototypes in such a way that the membership of an entity to a cluster expresses the proportion of the cluster's prototype reflected in the entity (proportional membership). In the generic version of the model, any entity may independently relate to any prototype, which is similar to the assumption underlying the fuzzy c-means criterion. The model is referred to as fuzzy clustering with proportional membership (FCPM). Several versions of the model relaxing the generic assumptions are presented and alternating minimization techniques for them are developed. The results of experimental studies of FCPM versions and the fuzzy c-means algorithm are presented and discussed, especially addressing the issues of fitting the underlying clustering model. An example is given with data in the medical field in which our approach is shown to suit better than more conventional methods.**

***Index Terms*—Alternating minimization, fuzzy clustering, fuzzy model identification, least-squares, proportional membership, prototype, semi-soft clustering.**

I. Introduction

FUZZY clustering techniques have been applied effectively in image processing, pattern recognition and fuzzy modeling. The best known approach to fuzzy clustering is the method of fuzzy c-means (FCM), proposed by Bezdek [1] and Dunn [2], and generalized by other authors. A good survey of relevant works on the subject can be found in [3]. In FCM, membership functions are defined based on a distance function, and membership degrees express proximities of entities to cluster centers (i.e., prototypes). By choosing a suitable distance function different cluster shapes can be identified [4]–[9]. Another approach to fuzzy clustering due to Krishnapuram and Keller [10] is the possibilistic c-means (PCM) algorithm which eliminates one of the constraints imposed on the search for c partitions leading to possibilistic (absolute) fuzzy membership values instead of FCM probabilistic (relative) fuzzy memberships. All these methods show how a cluster structure is determined from the data, but they are not oriented to provide feedback on generation of the data from a cluster structure. Developing models with explicit mechanisms for data generation from cluster structures can be of interest, because such a model can provide a theoretical framework for cluster structures found in data. In [11], Hathaway and Bezdek propose a fuzzy clustering approach for switching regression models where data are assumed to be generated from c regression models in such a way that each data point fits several (or all) of the c models to varying degrees (i.e., membership values). This idea can be carried out toward traditional fuzzy clustering as well, if the data are assumed to come from a cluster structure model.

Especially appealing in this respect seems the so-called typological structure in which observed entities relate in various degrees to one or several "prototypes." Such structures are relevant in many areas such as medicine where any patient may adhere, in different degrees, to one or several prototype disorder or disease. Obviously, problems of revealing hidden prototypes and extent of the entities' adherence to them from a data set belong to the realm of data mining.

In this paper, we propose a framework for mining for typological structures based on a fuzzy clustering model of how the data are generated from a cluster structure to be identified. Some preliminary results are described in [12], [13], and [14]. In this approach, the underlying fuzzy c partition is supposed to be defined in such a way that the membership of an entity to a cluster expresses a part of the cluster's prototype reflected in the entity. This way, an entity may bear 60% of a prototype A and 40% of prototype B, which simultaneously express the entity's membership to the respective clusters. This type of a membership function will be referred to as a proportional membership function.

The idea of proportional membership was initially described by Mirkin and Satarov in the so-called ideal type fuzzy clustering model [15], in which observed entities are represented as convex combinations of the prototypes; the convex combination coefficients are considered as the entity membership values. However, this approach as is invokes the extremal rather than averaged properties of the data, which may lead to unrealistic solutions [16]. Moreover, these solutions typically have nothing to do with those found with the FCM method. The ultimate goal of this paper is to develop a proportional membership model whose solutions would be more similar to the FCM solutions. Although, for every entity, its memberships to clusters are related by the condition that they sum up to unity, the bottom line is that any entity may independently relate to any prototype, which is akin to the assumption in the fuzzy c-means criterion. Our approach takes the adherence to centroids from the

Manuscript received December 6, 2000; revised September 30, 2002. The work of S. Nascimento was supported by FCT-Portugal under a Ph.D. grant (PRAXIS XXI program). This work was supported in part by DIMACS, Rutgers University, New Brunswick, NJ.

S. Nascimento is with the Centro de Inteligência Artificial (CENTRIA) Ciências, Faculdade Ciências e Tecnologia-Universidade Nova de Lisboa, Lisbon 2825-114, Portugal (snt@di.fct.unl.pt).

B. Mirkin is with the School of Computer Science and Information Systems, Birkbeck College, University of London, London WC1E 7HX, U.K.

F. Moura-Pires is with the Computer Science Department, Universidade de Évora, 7000 Évora, Portugal.

Digital Object Identifier 10.1109/TFUZZ.2003.809889

fuzzy c-means, but the membership is treated as a multiplicative factor to the prototype in a manner similar to that of the ideal type fuzzy clustering. We refer to our model as *fuzzy clustering with proportional membership* (FCPM, (which slightly differs from the denotation "FCMP" used in [12] and [13]). It should be pointed out that the FCPM suggests a specific mechanism for data generation from a cluster structure, which does not necessarily fit any data set.

We begin Section II by introducing FCM. Then, the FCPM approach is introduced: a generic form of FCPM is described in Section II.C, and extensions of the model to the case in which only large membership values are taken into account in the criterion are presented in Section II.D. The alternating minimization approach of FCM is extended to FCPM criteria in Section III. In contrast to FCM, the fuzziness constraints are not automatically satisfied for the FCPM solutions. This requires the use of a convenient nonlinear constrained optimization method. As such, the so-called gradient projection method is utilized and adapted for all versions of the FCPM. A combination of FCPM and FCM is suggested in Section III.E following the approach outlined in [17]. In Section IV, the results of experimental studies with generated data are presented and discussed. In Section V, we give an example from the field of psychiatry at which mental disorder syndromes represent ideal rather than average cases. It appears that FCPM is quite suitable in such a situation; in contrast to FCM, FCPM does not change the mental disorder prototypes when patients with less severe symptoms are added to the data set. Section VI concludes with the main results and issues.

II. Fuzzy Clustering Model With Proportional Membership (FCPM)

A. Data-Driven Cluster Modeling

To provide feedback from a cluster structure to the data from which it has been determined, we employ a framework based on the assumption that the data are generated according to the cluster structure. The structure underlies the data in the format of a traditional statistical equation

$$\text{observed data} = \text{model data} + \text{noise}. \tag{1}$$

In statistics, such an equation is accompanied by a probabilistic model of the noise. In our case, however, the model is not prespecified but rather derived from the data. Thus we concentrate on the "*model data*" part and leave the *noise* to be considered as just the set of differences between the observed and model data. The differences are treated here as mere residuals; they just should be made as small as possible by fitting the model.

In our clustering model, we assume the existence of some prototypes which serve as "ideal" patterns to data entities. To relate the prototypes to observations, we assume that the observed entities share parts of the prototypes. It is these parts that constitute the model data. The underlying structure of this model can be described by a fuzzy c partition defined in such a way that the membership of an entity to a cluster expresses the proportion of the cluster's prototype reflected in the entity. This, to an extent, models the concept of typology in descriptive sciences. The typological structure may be absent from the data as, for instance, when the data are generated by a preference relation. Thus, our assumption does not necessarily apply to any data set.

The idea of proportional membership can be formalized differently. In the so-called ideal type model [15], any observed entity is a convex combination of the prototypes and the coefficients are the entity membership values. Accordingly, the prototypes found with the ideal type model are extremes or even outsiders with regard to the "cloud" of points constituting the data [16]. This makes the ideal type model much different from the other fuzzy clustering techniques: the prototypes found with the other methods tend to be centroids of the corresponding clusters rather than their extremes. The extremity/externality of prototypes may become an issue when the feature values must be nonnegative or belong to a scoring system with fixed boundaries. However, even if no prior constraints are imposed, interpretation of the prototypes may be difficult. Indeed, to define a conceptually meaningful ideal type, fundamental properties of the objects must be utilized; this may not be the case in a typical situation in which relations between the observed features and underlying conceptual properties of the phenomenon are either indirect or unclear or both.

To bring the model-based approach closer to traditional fuzzy clustering techniques, we consider here a different way of associating observed entities to the prototypes: any entity may independently relate to any prototype, up to the condition that the sums of memberships for any entity must sum up to unity, which is similar to the assumption in the fuzzy c-means criterion described next.

B. FCM

The FCM [1] is one of the most popular methods in fuzzy clustering. It involves the concept of fuzzy c partition proposed by Ruspini [18], summarized here as follows.

Let $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ be a set of given data points, where each data point $\mathbf{x}_k$ $(k = 1, \ldots, n)$ is a vector in $\Re^p$. Let us denote the set of all real $c \times n$ matrices by U_{cn}, where c is a prespecified integer $2 \leq c < n$. Then, the fuzzy c partition space for X is the set $M_{fcn} \subset U_{cn}$ such that $U \in M_{fcn}$ if and only if

$$u_{ik} \in [0,1], \qquad \text{for all } i=1, \ldots, c \tag{2a}$$

$$\sum_{i=1}^{c} u_{ik} = 1 \tag{2b}$$

where u_{ik} is interpreted as the membership of an entity $\mathbf{x}_k$ in cluster i $(i = 1, \ldots, c)$.

The aim of the FCM algorithm is to find a fuzzy c partition and corresponding prototypes minimizing the objective function

$$J_m(U,V) = \sum_{k=1}^{n} \sum_{i=1}^{c} (u_{ik})^m \, \|\mathbf{x}_k - \mathbf{v}_i\|^2 . \tag{3}$$

In (3), $V = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_c)$ is a matrix of unknown cluster centers (prototypes) $\mathbf{v}_i \in \Re^p$, $\|\cdot\|$ is an inner product norm, and the weighting exponent $m \in [1, \infty)$ is a constant which affects the membership values, determining the degree of fuzziness of the cluster partition.

Clustering criterion J_m belongs to the class of least-squares criteria. Since it may be difficult to globally minimize (3), Bezdek [1] proposed a version of the alternating minimization algorithm defined as follows. Specify integer c, m and ε, a small positive constant; then set iteration number $t = 0$ and initialize $U^{(0)} \in M_{fcn}$. Any iteration consists of two steps. First, given the membership values $u_{ik}^{(t)}$, calculate the cluster centers $\mathbf{v}_i^{(t)} = \left[v_h^{(t)}\right]_i$ $(i = 1, \ldots, c)$, with

$$v_{ih}^{(t)} = \frac{\left\langle \left(\mathbf{u}_i^{(t)}\right)^m, \mathbf{x}_h \right\rangle}{\left\langle \left(\mathbf{u}_i^{(t)}\right)^{m-1}, \mathbf{u}_i^{(t)} \right\rangle} \tag{4}$$

where $\langle \mathbf{x}, \mathbf{y} \rangle$ denotes the scalar product of vectors $\mathbf{x}$, $\mathbf{y}$ and, for $\mathbf{x} = [x_h]$, $(\mathbf{x})^m$ denotes vector $[x_h^m]$.

Second, given the new cluster centers $\mathbf{v}_i^{(t)}$, update membership values $u_{ik}^{(t)}$

$$u_{ik}^{(t+1)} = \left[\sum_{j=1}^{c} \left(\frac{\left\| \mathbf{x}_k - \mathbf{v}_i^{(t)} \right\|}{\left\| \mathbf{x}_k - \mathbf{v}_j^{(t)} \right\|} \right)^{2/m-1} \right]^{-1}. \tag{5}$$

The process stops when $\left| U^{(t+1)} - U^{(t)} \right| \leq \varepsilon$, or a predefined maximum number of iterations is reached.

The FCM alternating (4) and (5) follow from the first-order optimality conditions (cf. [1] for the derivations). Since criterion J_m is not convex, the stationary point of the process may fail to give the global solution. However, the generated solutions $\{\left(V^{(1)}, U^{(1)}\right), \left(V^{(2)}, U^{(2)}\right), \ldots\}$ always converge to local minima or saddle points of J_m [19].

The FCM clustering criterion (3) aims to minimize the total distance between entities and prototypes weighted by the corresponding membership values. To decide for what c a resulting fuzzy c partition better fits the data, the FCM algorithm has to be run for different values of c ($c \geq 2$). Then each c partition can be evaluated by an expert or, sometimes, with formal criteria such as the so-called *validation function* [1].

The FCM method can be applied in a wide range of applications, leading to hyperspherical cluster shapes due to the averaging nature of formulas (4) and (5). However, this clustering criterion does not follow the pattern of (1) and it may be difficult sometimes to explicitly express how to reconstruct the data from a cluster solution.

C. Generic Proportional Membership Model

Let the data matrix X be preprocessed into Y by shifting the origin to the gravity center of all the entities (rows) in Y and rescaling features (columns) by their ranges. Thus, $Y = [y_{kh}]$ is a $n \times p$ entity-to-feature data table where each entity, described by p features, is defined by the row-vector $\mathbf{y}_k = [y_{kh}] \in \Re^p$ $(k = 1, \ldots, n;\ h = 1, \ldots, p)$.

Let us assume that each entity $\mathbf{y}_k = [y_{kh}]$ of Y is related to each prototype $\mathbf{v}_i = [v_{ih}]$ $(i = 1, \ldots, c)$, as in the FCM. Moreover, we further assume that the membership value u_{ik} is not just a weight, but it expresses the proportion of $\mathbf{v}_i$ which is present in $\mathbf{y}_k$. That is, we assume that approximately $y_{kh} = u_{ik} v_{ih}$ for every feature h. More formally, we suppose that

$$y_{kh} = u_{ik} v_{ih} + \varepsilon_{ikh} \tag{6}$$

where the residual values ε_{ikh} are as small as possible.

A clustering criterion according to (6) can be defined as fitting of each data point to a share of each of the prototypes, represented by the degree of membership. This goal is expressed in the least-squares criterion

$$E_0(U, V) = \sum_{i=1}^{c} \sum_{k=1}^{n} \sum_{h=1}^{p} (y_{kh} - u_{ik} v_{ih})^2 \tag{7}$$

which is to be minimized over all v_{ih} and admissible u_{ik} satisfying the constraints (2a) and (2b).

Equation (6) along with the least-squares criterion (7) to be minimized by unknown parameters $U \in M_{fcn}$ and $V = (\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_c) \in \Re^{cp}$ for Y given, will be referred to as the generic *fuzzy clustering proportional membership model*, FCPM-0, for short. In this model, the principle of the least-squares criterion in the fuzzy c-means is extended to the framework of (1).

Let us point out some aspects of this approach.

1) Each prototype $\mathbf{v}_i$ according to (6) is a "model" or "ideal" point such that any entity $\mathbf{y}_k$ bears a proportion of it u_{ik} up to the residuals. The proportion u_{ik} is considered as the value of membership of $\mathbf{y}_k$ to the cluster i. This way, both the prototypes and memberships are reflected in the model of data generation.
2) Equation (6) can be considered as a device to reconstruct the data from the model. The clustering criterion follows the least-squares framework to warrant that the reconstruction is, on average, as exact as possible. Other scalarizations of the idea of minimization of the residuals can be considered as well.
3) The least-squares criterion (7) differs from other least-squares criteria, such as that of FCM, by the fact that the trivial structure in which each of the observed entities forms a prototype on its own is not its solution. The trivial structure obviously reduces the FCM criterion to its absolute minimum 0, but it does not bring (7) to the minimum.
4) The model (6) may be considered overspecified: any observed entity must share a proportion of each of the prototypes, which, ideally, may occur only if all the entities and prototypes belong to the same unidimensional space. Such a solution is obviously not realistic, especially when contradictory tendencies are present in the data. This property of the generic model may lead to some over-estimation effects which may require modification of the criterion to a more realistic form in which the entities pertain to not all but only a few or just one of the prototypes.
5) A property of the clustering criterion (7) is that it remains constant if vectors $\mathbf{v}_i$ and $\mathbf{u}_i$ are changed for $\mathbf{v}_i/\gamma$ and $\mathbf{u}_i\gamma$ for some i, where γ is an arbitrary real. In particular, tending γ to zero, the membership vector,

$\mathbf{u}_i\gamma$, tends to zero while the prototype $\mathbf{v}_i/\gamma$ to infinity, without any change in the corresponding differences in criterion (7). This way, the following phenomenon may occur in the process of adjusting solutions during alternating minimization of criterion (7): to decrease some of the differences in (7) the membership values involved can be increased while simultaneously decreasing other membership values to zero along with moving corresponding prototypes to infinity. Some prototypes tending to infinity is a specific pattern of nonconvergence of the alternating minimization, which may occur in the generic FCPM model.

6) The latter two properties may make the model sensitive to the number of clusters c to be identified in the data. When this number is greater than the number of prototypes fitting well in the model, some of the prototypes in a computation may be driven out to infinity in the process of alternating minimization of the criterion (7) as described in the previous comment. If such a phenomenon can be confirmed with simulation experiments, this model could be utilized as a device for attacking such difficult issues as: 1) *"what is the correct number of clusters?"*; and 2) *"does the found clustering structure correspond to the data or not?"*. These issues can be made meaningful only under an assumption of a rigid cluster structure the data may have come from, such as in FCPM. This is confirmed, to an extent, in our experiments (Section IV.B).

D. Modifying the FCPM Criterion

As has been pointed out in our comments to (6) and (7), the requirement of FCPM that each entity can be expressed as a part of each prototype may be too strong and unrealistic sometimes. The intuition leads us to consider that only meaningful proportions, those expressed by high membership values, should be taken into account in (6).

We consider two ways to implement this idea in the FCPM framework: in a "smooth" manner and in a "hard" one, as specified in the next two sections.

1) Smooth Version: In order to decrease the effect of the residual values ε_{ikh} corresponding to small memberships u_{ik}, let us weigh the squared residuals in (7) by a power m of corresponding u_{ik}

$$E_m(U,V)=\sum_{i=1}^{c}\sum_{k=1}^{n}\sum_{h=1}^{p}u_{ik}^m(y_{kh}-u_{ik}v_{ih})^2 \tag{8}$$

subject to the fuzziness constraints (2a) and (2b).

The models corresponding to these criteria will be denoted as FCPM-1, for $m=1$, and FCPM-2, for $m=2$; no other m will be considered here. Criterion (7) is a special case of (8) corresponding to $m=0$.

2) Hard Version: A "hard" version of the FCPM model should only involve those equations in (6) that contain large values of u_{ik}. By specifying a threshold, β between 0 and 1.0, only those differences ε_{ikh} are left in the criterion (7) that satisfy the inequality, $u_{ik}\geq\beta$. In such a model, FCPM_β introduced in [20], entities may relate to as few prototypes as we wish. In particular, $\beta>0.5$ leads to the exclusive relationship of any entity to one prototype only. Thus, in FCPM_β only differences ε_{ikh} in which $u_{ik}\geq\beta$ are left in criterion (7). This leads to the clustering criterion defined as follows:

$$E_\beta(U,V,I_k)=\sum_{k=1}^{n}\sum_{i\in I_k}\sum_{h=1}^{p}(y_{kh}-u_{ik}v_{ih})^2 \tag{9}$$

where

$$I_k=\{i:u_{ik}\geq\beta,i=1,\ldots,c\} \tag{10}$$

and such that for all $k=1,\ldots,n$

$$u_{ik}\in[\beta,1],\qquad i\in I_k \tag{11a}$$

$$u_{ik}=0,\qquad i\notin I_k \tag{11b}$$

$$\sum_{i=1}^{c}u_{ik}=1. \tag{11c}$$

The idea of removing all small interactions between prototypes and entities from the criterion has been considered in the context of FCM by Selim and Ismail [21] in several versions, one of which relates to directly thresholding the membership weights as in the FCPM_β approach. The authors of [21] refer to this approach as to the "soft clustering," an intermediate between crisp clustering and fuzzy clustering.

III. FCPM Method and its Modifications

A. Alternating Minimization: Major and Minor Iterations

Let us consider the aforementioned FCPM criteria in the general format of criterion $E: M_{fcn}\times\Re^{cp}\rightarrow\Re^{+}$, to be minimized

$$\min E(U,V), U\in M_{fcn},\qquad V\in\Re^{cp}. \tag{12}$$

The alternating minimization algorithm applied to this problem involves two iterating steps. First, given $\widehat{V}\in\Re^{cp}$, minimize $E(U,\widehat{V})$ with regard to $U\in M_{fcn}$. Second, given the solution from the first step, $\widehat{U}\in M_{fcn}$, minimize $E(\widehat{U},V)$ over $V\in\Re^{cp}$. Based on this, an alternating minimization algorithm can be defined as follows.

Initialize $V^{(0)}$;
Repeat
given $V^{(t-1)}$
set $U^{(t)}:=\arg\min_{U\in M_{fcn}}E(U,V^{(t-1)})$;
given $U^{(t)}$, *set* $V^{(t)}:=\arg\min_{V\in\Re^{cp}}E(U^{(t)},V)$;
until $V^{(t)}\approx V^{(t-1)}$.

Given $U^{(t)}$, minimization of $E(U^{(t)},V)$ with regard to $V\in\Re^{cp}$ can be done according to the first-order condition of optimality. This condition implies that

$$v_{ih}^{(t)}=\frac{\left\langle\left(\mathbf{u}_i^{(t)}\right)^{m+1},\mathbf{y}_h\right\rangle}{\left\langle\left(\mathbf{u}_i^{(t)}\right)^{m+1},\mathbf{u}_i^{(t)}\right\rangle} \tag{13}$$

where parameter m takes value $m=0,1,2$ for either version of FCPM-m. This equation resembles (3) in the FCM method,

which suggests that the FCPM does capture the averaging feature of FCM. However, there is a difference as well. In (13), the power $m+1$ of $\mathbf{u}$ in the numerator differs from the power $m+2$ of $\mathbf{u}$ in the denominator, while these powers coincide in (3). Thus, the FCM prototypes are convex combinations of the observed points, which is not the case for the FCPM prototypes.

The minimization of criterion $E(U, V^{(t)})$ with regard to $U \in M_{fcn}$ is not as straightforward as in FCM, because of the constraints (2a) and (2b). This distinguishes the FCPM criteria from those of FCM for which the first-order solutions automatically satisfy constraints (2a) and (2b). After preliminary experiments with several options, the gradient projection method [22], [23] has been selected for the latter problem. This method works especially well for criterion E_0 in (7) as will be shown in the next section.

The gradient projection method is iterative. To distinguish between the alternating minimization iterations and iterations of the gradient projection method, we will refer to the former ones as "major" iterations and to the latter as "minor" iterations. The complete cycle of "minor" iterations is performed at each "major" iteration.

B. Gradient Projection Method for FCPM

In this section, we introduce the gradient projection method (GPM) and explain how it can be applied to minimize $E(U, \widehat{V})$ over $U \in M_{fcn}$.

GPM: Let $f : \Re^c \to \Re$ be a function to be minimized over a subset $Q \subset \Re^c$. For any $\mathbf{y}$ in $\Re^c$, let us denote its projection in Q by $P_Q(\mathbf{y})$, so that $P_Q(\mathbf{y})$ minimizes $\|\mathbf{x}-\mathbf{y}\|$ over all $\mathbf{x} \in Q$.

The gradient projection method for solving this optimization problem, starts with arbitrary $\mathbf{x}^{(0)} \in Q$ and iteratively transforms it according to the following rule:

$$\mathbf{x}^{(t+1)} = P_Q(\mathbf{x}^{(t)} - \alpha \nabla f(\mathbf{x}^{(t)})) \tag{14}$$

where α is a positive constant and $\nabla f(\mathbf{x})$ the gradient of f at $\mathbf{x} \in Q$.

Let us introduce conditions of convergence of the gradient projection method [22], [23].

A vector function $g : \Re^c \to \Re^c$ is said to satisfy the Lipschitz continuity condition with constant L if

$$\|g(\mathbf{x}) - g(\mathbf{y})\| \le L \|\mathbf{x}-\mathbf{y}\| \quad \forall \mathbf{x}, \mathbf{y} \in \Re^c. \tag{15}$$

Let us refer to $f(x)$ as strictly convex with constant $l > 0$ if

$$f(\mathbf{x}+\mathbf{y}) - f(\mathbf{x}) > \langle \nabla f(\mathbf{x}), \mathbf{y} \rangle + \frac{l}{2} \|\mathbf{y}\|^2 \quad \forall \mathbf{x}, \mathbf{y} \in \Re^c. \tag{16}$$

For a twice differentiable function f, this is equivalent to its Hessian, $\nabla^2 f$, being bound over $\Re^c$, that is, $\nabla^2 f(\mathbf{x}) \succeq l \cdot I$, where I is the diagonal matrix, and $A \succeq B$ means that $A - B$ is a positive semidefinite matrix.

The next two theorems from [22] and [23] state some convergence results for the gradient projection method.

Theorem 1: Let Q be convex and closed. Let $f(\mathbf{x})$ be a convex differentiable function in $\Re^c$, whose gradient satisfies the Lipschitz condition over Q with constant L. Let α be a real such that $0 < \alpha < 2/L$. Then, $\mathbf{x}^{(t)}$ converges to a globally optimal point $\mathbf{x}^*$ when t tends to infinity.

Theorem 2: In the conditions of Theorem 1 , if, supplementarily, $f(\mathbf{x})$ is twice differentiable and $L \cdot I \succeq \nabla^2 f(\mathbf{x}) \succeq l \cdot I$ for all $\mathbf{x}$ in Q, then $\left\|\mathbf{x}^{(t)} - \mathbf{x}^*\right\| \le c \cdot q^t$ (geometric progression convergence) where $q = \max\{|1-\alpha l|, |1-\alpha L|\}$.

Applying the Gradient Projection Method to FCPM: Let us denote the set of membership vectors satisfying conditions 2(a) and 2(b) by Q, which is a convex set (cf. [1, Th. 6.2.]). With $V^{(t)}$ fixed, the function $E(U, \widehat{V})$ is to be minimized over such U whose columns, $\mathbf{u}_k$, belong to Q.

The gradient projection method (14) applied to minimize $E(U, \widehat{V})$ can be stated as

$$\mathbf{u}_k^{(t)} = P_Q(\mathbf{u}_k^{(t-1)} - \alpha \nabla E(\mathbf{u}_k^{(t-1)}, \widehat{V})), \qquad k = 1, \ldots, n. \tag{17}$$

The possibility of translating the problem defined over matrices in terms of separate membership vectors in (17) is due to the fact that for each $\mathbf{u}_k^{(t)}$ its components $u_{ik}^{(t)}$ only depend on $u_{ik}^{(t-1)}$.

To apply (17), one needs to specify the following three parts of it:

i) computation of $\nabla E(\mathbf{u}_k^{(t-1)}, \widehat{V})$;
ii) choice of stepsize length α;
iii) finding the projection $P_Q(\mathbf{d}_k)$ for $\mathbf{d}_k = \mathbf{u}_k^{(t-1)} - \alpha \nabla E(\mathbf{u}_k^{(t-1)}, \widehat{V}) \in \Re^c$ $(k = 1, \ldots, n)$.

To address the former two problems, and for the sake of simplicity, we start from the criterion of the generic model E_0 in (7) as the $E(U, \widehat{V})$. Then, we extend the analysis to the other criteria E_m (8).

The function $E_0(U, \widehat{V})$ is convex and twice differentiable over its variables u_{ik}. The elements of its gradient are

$$\nabla E_0\left([\mathbf{u}_k], \widehat{V}\right) = 2(\langle \mathbf{v}_i, \mathbf{v}_i \rangle u_{ik} - \langle \mathbf{y}_k, \mathbf{v}_i \rangle), \qquad i = 1, \ldots, c \tag{18}$$

and its Hessian is a $cn \times cn$ diagonal matrix whose $((i,k),(i,k))$th element is $2\langle \mathbf{v}_i, \mathbf{v}_i \rangle$. Let us denote $l = 2\min_i \langle \mathbf{v}_i, \mathbf{v}_i \rangle$ and $L = 2\max_i \langle \mathbf{v}_i, \mathbf{v}_i \rangle$. Then, $L \cdot I \succeq \nabla^2 E_0(U, \widehat{V}) \succeq l \cdot I$. We assume all $\mathbf{v}_i$ are nonzero which implies $L \ge l > 0$.

The gradient ∇E_0 satisfies the Lipschitz condition over Q with constant L thus defined. Indeed

$$\begin{aligned} \nabla E_0\left([\mathbf{u}_k], \widehat{V}\right) - \nabla E_0\left([\mathbf{z}_k], \widehat{V}\right) &= 2\langle \mathbf{v}_i, \mathbf{v}_i \rangle (u_{ik} - z_{ik}) \\ &\le L(u_{ik} - z_{ik}) \\ \forall \mathbf{u}_k, \mathbf{z}_k &\in Q. \end{aligned} \tag{19}$$

which implies the same inequality for the vector norms, that is, the Lipschitz condition (15).

We have proven the following.

Proposition 3: Given $\widehat{V}$, l and L defined above, function $E_0\left(U, \widehat{V}\right)$ is strictly convex with constant l in the space of membership vectors $\mathbf{u}_k$, and its gradient ∇E_0 satisfies the Lipschitz condition over $\Re^c$ with the constant L.

This shows that both of the Theorems 1 and 2 are applicable here so that the process (17) converges with any α between 0 and $2/L$. Let us consider $\alpha = 2/(1+\varepsilon)L$, so that α spans the interval between 0 and $2/L$ when ε changes from 0 to infinity. In such a case the speed of convergence is controlled by

$q = \max\{|1-2l/(1+\varepsilon)L|, |1-2/(1+\varepsilon)|\}$. A computational experiment has been conducted starting from $\varepsilon = 0.1$ and repeatedly incrementing it by 0.1, to watch the rate of convergence for function E_0. The value $\varepsilon = 0.5$ has been chosen as giving most stable convergence, thus leading to $\alpha = 1/0.75L$. This way, issues i) and ii) have been addressed for criterion E_0.

The situation for functions $E_m(U, \widehat{V})$ $(m = 1, 2)$ is different: neither is convex over Q, though each satisfies the Lipschitz condition.

The elements of the gradients ∇E_m are defined by

$$\begin{aligned}\nabla E_m\left([\mathbf{u}_k], \widehat{V}\right) =& (m+2)\langle \mathbf{v}_i, \mathbf{v}_i\rangle u_{ik}^{(m+1)} \\ & - 2(m+1)\langle \mathbf{v}_i, \mathbf{y}_k\rangle u_{ik}^{m} \\ & + m\langle \mathbf{y}_k, \mathbf{y}_k\rangle u_{ik}^{m-1}. \end{aligned} \tag{20}$$

Note that (18) is a particular case of (20) with $m = 0$. On the other hand

$$\left|\nabla E_m\left([\mathbf{u}_k], \widehat{V}\right) - \nabla E_m\left([\mathbf{z}_k], \widehat{V}\right)\right| \le L\,|(u_{ik} - z_{ik})| \tag{21}$$

with $|\cdot|$ the L_1 norm, and L a constant equal to

$$L = (m+2)(m+1)V + 2m(m+1)YV + m(m-1)Y \tag{22}$$

with $V = \max_i \langle \mathbf{v}_i, \mathbf{v}_i\rangle$, $YV = \max_{i,k} |\langle \mathbf{v}_i, \mathbf{y}_k\rangle|$ and $Y = \max_k \langle \mathbf{y}_k, \mathbf{y}_k\rangle$. This shows that ∇E_m (with $m \ge 1$) satisfies Lipschitz condition for the L_1-norm with constant L previously defined.

Although Lipschitz continuity condition (15) is defined for the L_2-norm, it is known that the condition holds or does not hold in both L_2 and L_1-norms simultaneously, though the constant L in (15) may change [22].

When a function is not convex and satisfies Lipschitz condition, the gradient projection method may converge to a local optimum only, adding this to the general local search nature of the method of alternating optimization.

In order to specify the stepsize length α, we used the same $\alpha = 2/(1+\epsilon)L$ as for E_0 with $\epsilon = 0.5$. Substituting L from (22) in the formula, this leads to

$$\alpha_m = \frac{1}{1.5(c_m^1 V + c_m^2 YV + c_m^3 Y)}, \qquad m = 0, 1, 2 \tag{23}$$

with V, YV and Y previously defined, and coefficients c_m^j defined by

m	0	1	2
c_m^1	1	3	6
c_m^2	0	2	6
c_m^3	0	0	1

Notice that (23) is compatible with the α defined above for criterion E_0, since it is defined by the same rule.

Now, we can turn to the problem iii) of projection of the difference vectors $\mathbf{d}_k = \mathbf{u}_k^{(t-1)} - \alpha\nabla E_m(\mathbf{u}_k^{(t-1)}, \widehat{V})$ onto the set Q of vectors satisfying conditions (2a) and (2b).

For each criterion E_m, vectors $\mathbf{d}_k = [d_{ik}]$ to be projected onto Q are defined by

$$\begin{aligned} d_{ik}^{(t)} =& u_{ik}^{(t-1)} - 2\alpha_m \Big[(m+2)\langle \mathbf{v}_i, \mathbf{v}_i\rangle \left(u_{ik}^{(t-1)}\right)^{m+1} \\ & -2(m+1)\langle \mathbf{v}_i, \mathbf{y}_k\rangle \left(u_{ik}^{(t-1)}\right)^{m} \\ & +m\langle \mathbf{y}_k, \mathbf{y}_k\rangle \left(u_{ik}^{(t-1)}\right)^{m-1}\Big] \end{aligned} \tag{24}$$

derived from (17) with ∇E_m in (20) substituted for ∇E.

Projecting a Vector on the Simplex of Membership Vectors: Let us consider the problem of finding a vector $\mathbf{u} = [u_i] \in Q$ $(i = 1, \ldots, c)$, which is at the minimum distance from a prespecified vector $\mathbf{d} = [d_i]$. This problem can be stated as follows:

$$\min_{\mathbf{u}} f(\mathbf{u}) = \|\mathbf{u} - \mathbf{d}\|^2 \tag{25}$$

subject to constraints (2a) and (2b).

In order to solve this problem, let us assume without any loss of generality that $d_1 \ge d_2 \ge \cdots \ge d_c$.

Proposition 4: The optimal $\mathbf{u}^*$ in (25) has the same order of components, that is, $u_1^* \ge u_2^* \ge \cdots \ge u_c^*$.

To prove the statement, let us assume that it is not true, thus, for instance, $u_1^* < u_2^*$. Then, we can further decrease criterion (25) by shifting a small amount of u_2^* to u_1^*. Indeed, let us take $\delta > 0$ such that $\delta < u_2^* - u_1^*$ and put $u_1^{**} = u_1^* + \delta$ and $u_2^{**} = u_2^* - \delta$ in $\mathbf{u}^*$ instead of u_1^* and u_2^*, respectively. Then, the value of $f(\mathbf{u})$ will change by $\Delta = (u_1^* + \delta - d_1)^2 - (u_2^* - \delta - d_2)^2 - (u_1^* - d_1)^2 + (u_2^* - d_2)^2$. With a little arithmetic, this can be reduced to $\Delta = 2\delta\,[u_1^* - u_2^* + \delta - d_1 + d_2]$ which is negative because, in our assumptions, $u_1^* - u_2^* + \delta < 0$ and $-d_1 + d_2 < 0$. Thus, criterion (25) has been further decreased, which contradicts the optimality of $\mathbf{u}^*$ and proves the proposition.

Thus, $u_1^* \ge u_2^* \ge \cdots \ge u_{c^+}^* > 0$ for some $c^+ \le c$ and the final $c - c^+$ components are zero. For the nonzero components, the following equations hold for an optimal $\mathbf{u}^*$:

$$u_1^* - d_1 = u_2^* - d_2 = \cdots = u_{c^+}^* - d_{c^+}.$$

Otherwise, we could transform $\mathbf{u}^*$ as above in the proof of Proposition 4 by redistribution of values among the positive $u_1^*, \ldots, u_{c^+}^*$ in such a way that its distance from $\mathbf{d}$ decreases, which would contradict the assumption that the distance had been minimized by $\mathbf{u}^*$. Thus, for the optimal $\mathbf{u}^*$, $u_1^* = d_1 - a_{c^+}$, $u_2^* = d_2 - a_{c^+}$, $\ldots$, $u_{c^+}^* = d_{c^+} - a_{c^+}$, where a_{c^+} is the common value of the differences; it can be determined as the result of summation of

$$a_{c^+} = \frac{1}{c^+}\sum_{i=1}^{c^+} d_i - \frac{1}{c^+}. \tag{26}$$

The value c^+ is not known beforehand. To find it, the following iterative process can be applied. Start with $c^+ = c$, and at each iteration compute a_{c^+} with formula (26) and take the difference $u_{c^+}^* = d_{c^+} - a_{c^+}$. If it is less than or equal to zero, decrease c^+ by 1 and repeat the process until the difference becomes positive. Then, define all the other u_i^* as follows: $u_i^* = d_i - a_{c^+}$ for $i = 1, \ldots, c^+$ and $u_i^* = 0$ for

$i = c^+ + 1, \ldots, c$. The process can be accelerated if, at each iteration, c^+ is decreased by the number of negative values in the set of differences $u_i^* = d_i - a_{c^+}$ $(i = 1, \ldots, c^+)$. This is reflected in algorithm A4.1, below.

C. FCPM Algorithm Reviewed

The FCPM algorithm is defined as an iterative alternating minimization algorithm in which each "major" iteration consists of two steps as follows. First, given prototype matrix V, the optimal membership values are found with (17). This requires an iterative process described in Section III-B. Second, given membership matrix U, the optimal prototypes are determined according to the first-degree optimality conditions (13).

Algorithm A4.1: $Projection_Q(\mathbf{d})$

Given $\mathbf{d} = [d_i]$ $(i = 1, \ldots, c)$
sort $\mathbf{d} = [d_i]$ in the descending order;
$c^+ := c$;
Repeat
  calculate a_{c^+} by **(26)**
  $zeros := false$; $i := 0$;
  Repeat
    $i := i + 1$;
    $u_i := d_i - a_{c^+}$;
    If $u_i \leq 0$ **then** $zeros := true$; **endIf**
  until $(i = c^+$ **.or.** $zeros)$;
  If $zeros$ **then**
    For $j = i, \ldots, c^+$ **do** $u_j := 0$; **endFor**
    $c^+ := i - 1$;
  endIf
until $(c^+ = 0$ **.or. not** $zeros)$;
return $\mathbf{u} = [u_1, \ldots, u_{c^+}, 0, \ldots, 0]$.

The algorithm starts with a set $V^{(0)}$ of c arbitrarily selected prototype points in $\Re^p$ and $U^{(0)}$ in M_{fcn}; it stops when the difference between successive prototype matrices becomes small (according to an appropriate matrix norm, $|\cdot|_{err}$).

The global convergence of the FCPM algorithm is not guaranteed. Moreover, with a "wrong" number of clusters prespecified, FCPM-0 may not converge at all since FCPM-0 may shift some prototypes to infinity as was observed in comments for the generic FCPM model. In our experiments, the number of major iterations in FCPM-0 algorithm when it converges is rather small, which is exploited as a stopping condition: when the number of major iterations in an FCPM-0 run goes over a large number (in our calculations, over 100), that means the process does not converge.

The FCPM-m $(m = 0, 1, 2)$ algorithm is defined in A4.2.

D. Hard Version of FCPM

The "hard" version of the FCPM model described in Section II-D.II can be implemented with corresponding adjustments of both major and minor iterations of FCPM.

The FCPM_β clustering criterion (9) is a nonconvex function (as are the other FCPM criteria). Constraints (11a)–(11c) also define nonconvex sets. This adds to the difficulty of constructing an algorithm to solve this problem. To minimize (9), the FCPM algorithm has been modified in two places: 1) the initial setting $(V^{(0)}, U^{(0)})$ has been set to be the result of running one FCPM-1 major iteration, and the initial I_k $(k = 1, \ldots, n)$ are calculated accordingly, and 2) in the projection algorithm, the boundary value defining null membership is to be taken as the threshold β $(0 < \beta < 1.0)$ rather than zero. The modified projection algorithm finds solutions in the set $Q_\beta = \{\mathbf{u} = [u_i] : u_i \geq \beta \text{ or } u_i = 0\}$ rather than in Q. The sets I_k are then adjusted at each iteration so that (10) holds. This version of FCPM algorithm will be referred to as FCPM_β.

Algorithm A4.2: FCPM-m Algorithm

Given $Y = [y_k]$
choose $c(2 \leq c < n)$, m $(m = 0, 1, 2)$, T_1, T_2, $\varepsilon > 0$;
initialize $V^{(0)}$, $U^{(0)}, U^{(0)} \in M_{fcn}$, $t_1 := 0$;
Repeat
  $t_2 := 0$;
  $U^{(t_2)} := U^{(t_1)}$;
  Repeat
    $t_2 := t_2 + 1$;
    For $k = 1, \ldots, n$ **do**
      calculate $\mathbf{d}_k^{(t_2)}$ with $V^{(t_1)}$, $\mathbf{u}_k^{(t_2-1)}$ by **(24)**;
      $\mathbf{u}_k^{(t_2)} := Projection_Q(\mathbf{d}_k^{(t_2)})$ $\quad \%(Alg.A4.1)$
    endFor
  until $(\left|U^{(t_2)} - U^{(t_2-1)}\right|_{err} < \varepsilon$ **.or.** $t_2 = T_2)$;
  $t_1 := t_1 + 1$;
  $U^{(t_1)} := U^{(t_2)}$
  calculate $V^{(t_1)}$ with $U^{(t_1)}$ by**(13)**;
until $(\left|V^{(t_1)} - V^{(t_1-1)}\right|_{err} < \varepsilon$ **.or.** $t_1 = T_1)$;
return $(V, U) := \left(V^{(t_1)}, U^{(t_1)}\right)$.

The calculations of membership vectors $\mathbf{u}_k^{(t)} = \left[u_{ik}^{(t)}\right]$ are based on vectors $\mathbf{d}_k^{(t)} = \left[d_{ik}^{(t)}\right]$, where (27a)-(27b), shown at the bottom of the page, hold, and such that the projection of $\mathbf{d}_k^{(t)}$ in Q_β is to be taken as $\mathbf{u}_k^{(t)}$.

The FCPM_β algorithm is defined in A4.3.

For small β, this method may show the same pattern of nonconvergence as FCPM-0, removing some of the prototypes to infinity.

$$d_{ik}^{(t)} = \begin{cases} u_{ik}^{(t-1)} - 2\alpha_0 \left[\langle \mathbf{v}_i, \mathbf{v}_i \rangle u_{ik}^{(t-1)} - \langle \mathbf{y}_k, \mathbf{v}_i \rangle\right], & i \in I_k \quad (27a) \\ 0, & i \in C - I_k \quad (27b) \end{cases}$$

In our computations, the threshold β has been taken as the minimum β value for which FCPM_β converges, by starting from $\beta = 0$ and repeatedly incrementing it by 0.1.

E. Combining FCPM and FCM: FCPM-AE

Criteria (8) and (9) for fitting the FCPM model may be too restrictive in revealing cluster structures in data that have been generated differently from what the FCPM model suggests. In particular, FCPM-0 criterion (6) may underestimate the number of clusters present in data.

As we already have mentioned, FCM prototypes are convex combinations of data points whereas FCPM prototypes are not. Therefore, FCPM prototypes are not guaranteed to lie in the convex hull of the data set and may move out of the data set area.

Algorithm A4.3: FCPM$_\beta$ *Algorithm*

Given $Y = [\mathbf{y}_k]$
choose $c\,(2 \le c < n)$, $\beta\,(0 < \beta \le 1.0)$ $T_1, T_2, \varepsilon > 0$;
initialize $\left(V^{(0)}, U^{(0)}\right)$:= FCPM – 1 with $T_1 := 1$;
$t_1 := 0$
set I_k by **(10)**$(k = 1, \ldots, n)$;
Repeat
  $t_2 := 0$;
  $U^{(t_2)} := U^{(t_1)}$;
  Repeat
    $t_2 := t_2 + 1$;
    For $k = 1, \ldots n$ **do**
      calculate $\mathbf{d}_k^{(t_2)}$ with $\mathbf{V}^{(t_1)}$, $\mathbf{u}_k^{(t_2-1)}$ by **(27)**;
      $\mathbf{u}_k^{(t_2)} := Projection_{Q_\beta}(\mathbf{d}_k^{(t_2)})$
    endFor
  until $(\left|U^{(t_2)} - U^{(t_2-1)}\right|_{err} < \varepsilon$ **.or.** $t_2 = T_2)$
  update I_k by **(10)**$(k = 1, \ldots, n)$;
  $t_1 := t_1 + 1$;
  $U^{(t_1)} := U^{(t_2)}$
  calculate $V^{(t_1)}$ with $U^{(t_1)}$ by **(13)** with $m = 0$;
until $(\left|V^{(t_1)} - V^{(t_1-1)}\right|_{err} < \varepsilon$ **.or.** $t_1 = T_1)$
return $(V, U) := (V^{(t_1)}, U^{(t_2)})$.

In order to overcome this without losing the interpretability of the model-based FCPM proportional membership, it would be desirable to combine the advantages of FCM prototypes (4) with FCPM proportional memberships, this way relaxing the rigidity of FCPM.

Runkler and Bezdek [17] propose an approach to fuzzy clustering in which the unique objective function model is abandoned and substituted by a more general framework defined by the architecture of the alternating minimization algorithm and by user-specified equations for updating U and V. When the user selects updating equations not from a unique objective function model, clusters and cluster centers are referred to as *estimated* by alternatingly updating partitions and prototypes. This framework is called by the authors the *alternating cluster estimation* (ACE), and is considered a flexible version of the *alternating optimization* FCM approach [8].

To accomplish our goal, we follow the ACE framework to combine the FCPM and FCM approaches in the following FCPM-*AE* method: prototypes are updated as the gravity cluster centers (4), and partitions by the FCPM proportional membership function.

IV. Experimental Study

The main goal of this experimental study is threefold:

1) to analyze the ability of FCPM to recover the original prototypes from which data have been generated;
2) to study the behavior of FCPM-0 as an index of the number of clusters present in data;
3) to compare FCPM and FCM methods by using generated data sets.

A. Setting of Experiments

A number of distinct approaches have been proposed in the literature for generating artificial clustering data. In these approaches, data points are assumed to have been generated from some probability distribution, usually using multivariate normal clusters ranging from simple to complex covariance structures [24], [25].

The FCPM model should be tested on data exhibiting a typological structure, according to assumptions underlying the FCPM model, in particular: 1) there is a cluster structure underlying the model of data generation, and 2) in such a structure, each prototype is a "model" or "ideal" point such that any entity, $\mathbf{y}_k$, bears a proportion of it. In contrast to traditional data generation models (like ones in [24] and [25]) we do not pursue any specific geometric shape of the clusters, except for that in accordance with the generic proportional membership assumption: each observation can be associated with a proportion of corresponding cluster prototype. To accomplish this, a data generator has been constructed as follows.

Data Generator:

1) The dimension of the space (p), the number of clusters (c) and numbers $n_1, n_2, \ldots, n_c$ are randomly generated within prespecified intervals. The data set cardinality is defined as $n = \sum_{i=1}^{c} n_i$.
2) c cluster directions are defined as follows: vectors $\mathbf{o}_i \in \Re^p$ $(i = 1, \ldots, c)$ are randomly generated within a prespecified hyper-cube with side length between -100.0 and 100.0; then, their gravity center $\mathbf{o}$ is taken as the origin of the space.
3) For each i, define two p-dimensional sampling boxes, one within bounds $A_i = [.9\mathbf{o}_i, 1.1\mathbf{o}_i]$ and the other within $B_i = [\mathbf{o}, \mathbf{o}_i]$; then generate randomly $0.2n_i$ points in A_i and $0.8n_i$ points in B_i.
4) The data generated are normalized by centering to the origin and scaling by the range.

All randomly generated items are generated from a uniform distribution in the interval [0,1]. This way we could provide rather complex data structures with a small number of easily interpretable parameters, which would be much more difficult to achieve with more traditional multivariate normal distributions.

To visualize data, they are projected into a two-dimensional/three-dimensional (2-D/3-D) space of the best principal components (see Fig. 1).

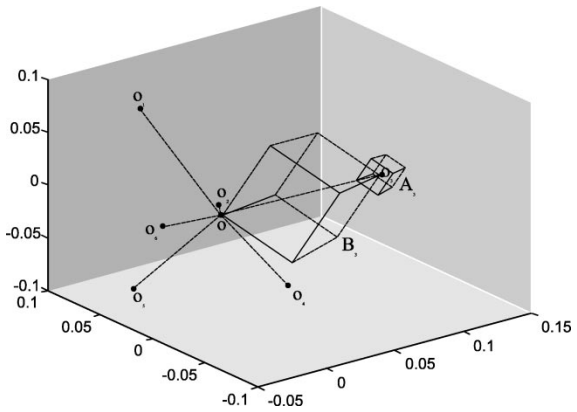


Fig. 1. Architecture of the data generator on a 3-D projection of the best three principal components, with p-dimensional sampling boxes A_i and B_i, for a data structure with six original prototypes.

All the algorithms and data generator have been written in MatLab 5.2 [26], and the experimental study was conducted on a PC with a PENTIUM II processor at 267 MHz.

In the discussion of the experimental results, the emphasis will be given to the clustering results rather than the performance of the algorithms. Our criteria (8) and (9) are more complex than that of FCM and thus require more calculations.

In our experiments, each of the six algorithms outlined above (FCM, FCPM-0, FCPM-1, FCPM-2, FCPM_β, and FCPM-*AE*) has been run on the same data set (with the same initial setting) for different values of c ($c = 2,3,4,\ldots$). The parameters of the algorithms are specified as: $T_1 = T_2 = 100$, $\varepsilon = 0.0001$ and $|\cdot|_{err}$ is L_1-norm in $\Re^{cp}$. The parameters of FCM have been specified as $m = 2$ and $\|\cdot\|$ equal to the Euclidean norm. Also, the FCM algorithm has been slightly modified to start with the prototypes $V^{(0)}$ rather than with the membership matrix $U^{(0)}$ in the original version (a similar modification has been adopted in [8] and [17]).

Cluster solutions found with FCPM algorithms have been characterized by the following four features: 1) the number of clusters found, c'; 2) the separability index, B_c; 3) the dissimilarity D_{FCM} from the FCM found prototypes; and 4) the dissimilarity D_O from the original prototypes. The separability index was also calculated for FCM solutions. The separability index B_c is

$$B_c = 1 - \frac{c}{c-1}\left(1 - \frac{1}{n}\sum_{k,i}(u_{ik})^2\right) \tag{28}$$

as defined in [1]. This index assesses the fuzziness of partition U; it takes values in the range [0,1]; $B_c = 1$ for hard partitions and $B_c = 0$ for the uniform memberships (cf. [1, p. 157]).

The dissimilarity between FCPM prototypes V' and "reference" prototypes V (in our experiments, either the original prototypes or FCM ones), is defined as

$$D_V = \frac{\sum_{i=1}^{c}\sum_{h=1}^{p}(v'_{ih} - v_{ih})^2}{\sum_{i=1}^{c}\sum_{h=1}^{p}v_{ih}^2 + \sum_{i=1}^{c}\sum_{h=1}^{p}v_{ih}'^2} \tag{29}$$

which measures the squared relative quadratic mean difference between corresponding prototypes V and V'. Matching between prototypes is determined according to minimal distances. In the case in which the number of prototypes c' found by FCPM-0 is smaller than c, only c' prototypes participate in (29). Coefficient D_V is not negative, and it equals 0 if and only if $v_{ih} = v'_{ih}$ for all $i = 1,\ldots,c$; $h = 1,\ldots,p$. In a typical situation, when $\mathbf{v}_i$ and $\mathbf{v}'_i$ are in the same orthants, D_V is not greater than 1. Notice that the dissimilarity measure D is more or less independent of the original $\mathbf{v}$'s, their cardinality (c_0) and dimension (p); thus, it can be used to compare cluster prototypes in different settings.

For a fixed pair, p and c_0, a group of 15 data sets were generated with different numbers of entities and different prototypes. The experiments comprised seven such groups with p ranging from 5 to 180 and c_0 from 3 to 6.

For each group of data sets of the same dimension p and the number of generated prototypes c_0, the FCPM algorithms have been compared based on the number of major iterations t_1, number of prototypes found c', separability coefficient B_c, the dissimilarity D_{FCM} from FCM prototypes and dissimilarity D_O from the original prototypes $\mathbf{o}_i$ ($i = 1,\ldots,c_0$).

B. Summary of the Results

Results of our experiments with FCM and FCPM algorithms lead us to distinguish between three types of data dimensionality: low, intermediate, and high, because the algorithms behave differently across these categories. With several hundred entities, $p/c_0 \leq 5$ is considered small and $p/c_0 \geq 25$ high.

In the following, we refer to three types of the numbers of prototypes: (1) the number of originally generated prototypes, c_0, (2) the number of prototypes prespecified in a run, c, and (3) the number of prototypes found by an algorithm, c'. The numbers c' and c, in the same computation, may differ because of either of two causes:

C1) some of the initial prototypes converge to the same stationary point;

C2) some of the initial prototypes have been removed by the algorithm from the data cloud (this concerns mostly FCPM-0).

In either case, $c' < c$.

In order to illustrate the kind of cluster structures we operate with in the experiments, a small data set was generated with $c_0 = 3$ original prototypes in $\Re^2$ ($p = 2$) with $n = 49$ points, as displayed in Fig. 2. The FCM and FCPM algorithms have been run starting from the same initial setting, seeking for $c = 3$ prototypes, which are also displayed in Fig. 2. The FCPM-0 algorithm has moved one of the prototypes (that corresponding to cluster 2) far away to the left from cluster 2, so that its points, in the end, share the prototype with cluster 3. Concerning the other FCPM and FCM algorithms, all of them found their partitions with $c' = 3$ prototypes. Method FCPM-2 produced the most extremal prototypes close to the original ones, and the others FCPM methods produced prototypes close to the prototypes found by FCM.

In the main series of experiments the number of prototypes looked for is taken coinciding with the number of original pro-

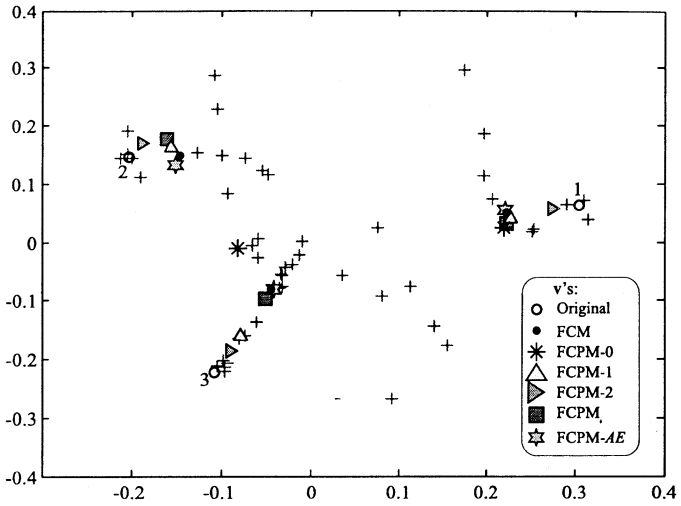


Fig. 2. Results of clustering for the illustrative data set ($c_0 = 3$, $p = 2$, $n = 49$). All FCPM and FCM algorithms find $c' = 3$ prototypes, except FCPM-0. The FCPM-0 algorithm has moved one of the prototypes (corresponding to cluster 2) far away to the left of cluster 2. Therefore, its points share the prototype with cluster 3.

totypes, $c = c_0$ (Table I). Another set of experiments have been carried out for $c = c_0 + 1$ (Table II).

Table I shows the average results of running FCM and FCPM algorithms with $c = c_0$ for each of the three groups of data sets: small dimension ($p = 5$, $c_0 = 3$), intermediate dimension ($p = 50$, $c_0 = 4$), and high dimension ($p = 180$, $c_0 = 6$). When $c' < c$, the cause, either C1 or C2, is shown in the upper index.

The results of the experiments can be summarized as follows.

1) With regard to correctness of the number of clusters identified in the data, the methods fall in three groups:
   a) methods retaining a prespecified number of clusters (FCPM-1, FCPM-2 and FCPM-*AE*);
   b) methods which can reduce the number of clusters (especially in the high-dimension data case) by either cause C1 (FCM) or C2 (FCPM-0);
   c) method FCPM_β whose behavior depends on the threshold β value.

   For low and intermediate dimension data sets, FCPM-0 almost always finds the correct number of clusters generated (column c' in Table I). In the high-dimensional spaces, FCPM-0 finds the correct number of clusters in 50% of the cases and it underestimates the number of clusters in other cases. This is carried out by moving some of the prototypes out of the data set area. In the high-dimensional spaces, FCM typically leads to even smaller number of clusters, making the initial prototypes converge to the same point. Further experiments show that this feature of FCM depends not only on the space dimension but also follows the generated data structure. Specifically, for the high dimension data, FCM seems to view the entire data set as just one cluster around the origin of the space, because there are not that many points generated "outside" of it. When the proportion of points generated around the original prototypes (within the boxes A_i in step 3 of the data generator) is increased from 0.2 to 0.8, FCM identifies the correct number of prototypes. (See also [27] for discussion of issues related to high-dimensional spaces).

   In Table I, the threshold parameter β in FCPM_β is chosen as the lowest value for which the algorithm converges. This depends on the space dimension: the higher the dimension, the larger minimum β for the algorithm to converge ($\beta = 0.1$, $\beta = 0.3$ and $\beta = 0.4$, for the small, intermediate and high-space dimensions, respectively).
2) The prototypes found by FCPM-*AE*, FCPM_β, FCPM-1 and FCPM-0 almost coincide with those found by FCM when the number of centroids is determined by FCM

TABLE I
AVERAGE RESULTS OF RUNNING FCM AND FCPM ALGORITHMS FOR THREE GROUPS OF DATA SETS: SMALL DIMENSION ($p = 5$), INTERMEDIATE ($p = 50$), AND HIGH DIMENSION ($p = 180$)

space dimension	small	intermediate	high
$c=c_0$	3	4	6
		c'	
FCM	3	4	1^{c1}
FCPM-0	3	4	6 or 5^{c2}
FCPM-1	3	4	6
FCPM-2	3	4	6
FCPM_β	3	4	6
FCPM-*AE*	3	4	6
		D_{FCM} (%)	
FCM	-	-	-
FCPM-0	0.49	1.23	143.50
FCPM-1	0.89	0.16	94.20
FCPM-2	7.10	11.44	97.18
FCPM_β	0.30	0.10	87.46
FCPM-*AE*	0.09	0.14	94.84
		D_O (%)	
FCM	14.70	17.90	96.83
FCPM-0	12.20	14.34	11.67
FCPM-1	10.20	15.31	15.82
FCPM-2	2.30	1.16	0.45
FCPM_β	12.20	15.90	16.34
FCPM-*AE*	13.38	15.47	15.14
		B_c	
FCM	0.61	0.47	0.01
FCPM-0	0.84	0.90	0.78
FCPM-1	0.80	0.98	1.00
FCPM-2	0.43	0.36	0.30
FCPM_β	0.89	1.00	1.00
FCPM-*AE*	0.86	0.95	0.89
		t_1	
FCM	12	15	27
FCPM-0	10	20	78/101
FCPM-1	11	9	11
FCPM-2	11	11	27
FCPM_β	4	3	3
FCPM-*AE*	8	7	7

TABLE II
NUMBERS OF PROTOTYPES FOUND BY THE FCM AND FCPM ALGORITHMS WITH $c = c_0 + 1$

space dimension	small	intermediate	high
c_0+1	4	5	7
FCM	4	4^{c1}	1^{c1}
FCPM-0	3	4^{c2}	$(6;5;4)^{c2}$
FCPM-1	4	4^{c1}	6^{c1}
FCPM-2	4	4^{c2}	6^{c2}
$\mathrm{FCPM}_{\beta=0.5}$	4	5	7
FCPM-*AE*	4	4^{c1}	6^{c1}

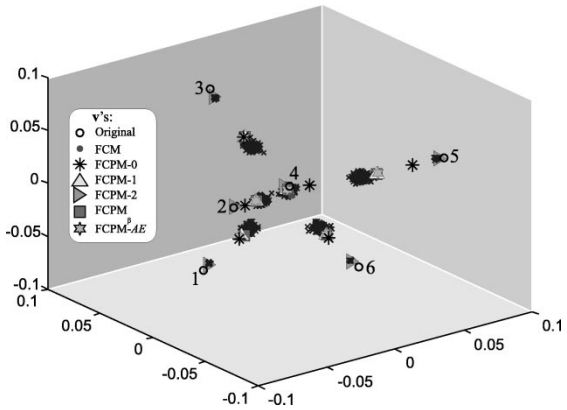


Fig. 3. 3-D plot of the prototypes found by FCM and FCPM with $c = c_0$ prototypes (a high-dimension case with $p = 180$, $c_0 = 6$).

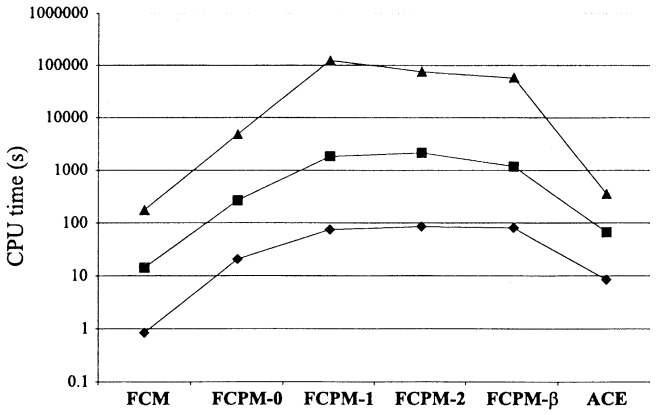


Fig. 4. Average CPU times (seconds on a log scale) taken by FCM and FCPM algorithms, for the three groups of data sets with low (diamonds), intermediate (squares), and high (triangles) space dimensions.

correctly. These prototypes differ from those originally generated. In contrast, FCPM-2 identifies the originally generated prototypes and, thus, yields results differing from FCM. This effect is especially visible when the ratio p/c increases. The prototypes found by FCPM-0 can be considered intermediate between those found by FCM and FCPM-2. Fig. 3 illustrates these aspects, displaying the relative locations of FCPM and FCM found prototypes, with regard to the original prototypes, for a high-dimensional data set.

3) According to the partition separability coefficient, B_c, FCPM-0, FCPM-1, FCPM_β and FCPM-*AE* partitions have more contrasting than FCM ones. In particular, in high-dimensional cases FCPM-1 and FCPM_β lead to hard clustering solutions. The FCPM-2 gives the fuzziest partitions, typically differing from those of FCM. On the other hand, the FCPM-*AE* partitions are more contrast than FCM ones. This probably can be explained by the fact that the proportional membership is more sensitive to the discriminate attribute values characterizing a cluster, when compared with the FCM distance-based membership.
4) On average, the number of major iterations (t_1) in FCPM-1, FCPM-2, FCPM_β and FCPM-*AE* is smaller than that in FCM, while in FCPM-0 this number does not differ significantly from that in FCM (for small dimensions). However, the running time is greater for FCPM algorithms, due to time spent in minor iterations with the gradient projection method. Fig. 4 displays the average CPU times (in seconds), on a logarithmic scale, taken by each algorithm, for the three groups of (15) data sets of low, intermediate and high space dimension, respectively. FCM is the fastest algorithm followed by FCPM-*AE*. The discrepancy in computational times of FCPM-0 from the other FCPM algorithms, is due to the fast convergence to a (global) optimum of the minor iteration cycle in FCPM-0, in contrast to the other FCPM algorithms.

Another series of experiments have been performed in order to analyze the sensibility of FCPM algorithms to prespecifying a larger number of clusters than those from which data are generated. Depending on the ratio p/c, the FCM and FCPM algorithms behave differently in this case. The results are as follows (Table II).

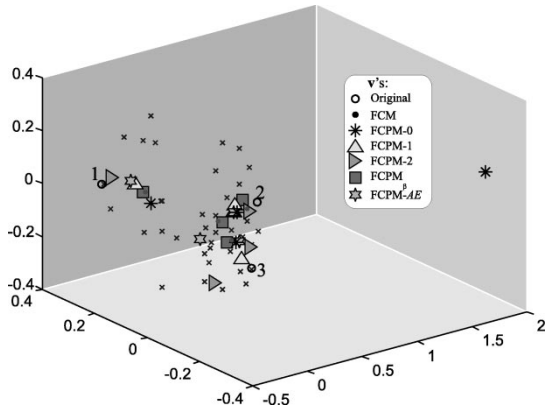


Fig. 5. 3-D plot of the prototypes found by FCM and FCPM with $c = c_0 + 1$ (small dimension case with $p = 5$, $c_0 = 3$). Only FCPM-0 finds the correct number of prototypes by moving the extra prototype out of the data space; all the other FCPM and FCM algorithms find $c' = c_0 + 1$ distinct prototypes.

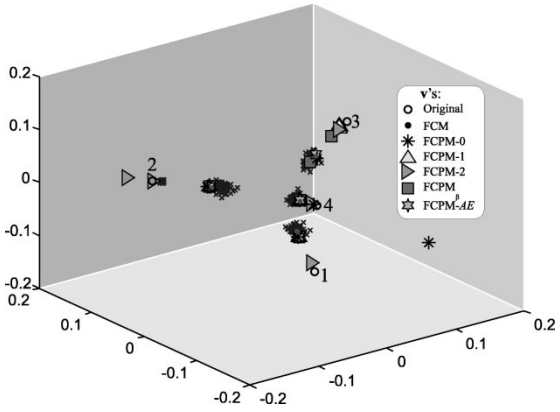


Fig. 6. 3-D plot of the prototypes found by FCM and FCPM with $c = c_0 + 1$ (intermediate dimension case with $p = 50$, $c_0 = 4$). FCPM-0 and FCPM-2 find the correct number of prototypes by moving the extra prototype out of the data area. In case of FCM, FCPM-1 and FCPM-AE, the corresponding extra prototype coincides with one of the remaining ones. Only FCPM_β finds $c' = c_0 + 1$ distinct prototypes.

For the small dimension, FCM, FCPM-1, FCPM-2, FCPM_β and FCPM-*AE* find $c' = c_0 + 1$ distinct prototypes. The FCPM-0 removes the extra prototype out of the data space (Fig. 5).

For the intermediate dimension case, FCM, FCPM-1, and FCPM-*AE* find just $c' = c_0$ distinct prototypes; the extra prototype almost always coincides with one of the others. Both FCPM-2 and FCPM-0 also find $c' = c_0$ prototypes but by removing an extra prototype out of the data set area (Fig. 6), rather than by merging two different prototypes. On the contrary, FCPM_β can identify the required (and wrong) number of clusters with β parameter increased to 0.5, thus leading to hard cluster structures.

For the high dimension cases both FCM and FCPM-0 lead to "degenerate" solutions by their respective means: FCM merges some prototypes and FCPM-0 removes some prototypes out of the data area, preventing the algorithm from convergence (see corresponding entry in Table II). The methods FCPM-1, FCPM-*AE* and FCPM-2, recover the number of prototypes that have been generated (Fig. 7). Only FCPM_β finds the required number of prototypes with β parameter increased to 0.5. However, FCPM-1 and FCPM_β lead to hard clusters in the high-dimension cases, which leaves FCPM-*AE* as the only genuine fuzzy clustering method in high dimension cases for this type of data.

Behavior of the other features (D_{FCM}, D_0, and B_c), does not differ from that shown in Table I. Overall, in the high dimension cases, the winners are FCPM-2 and FCPM-*AE*. These two differ in the structures found: FCPM-2 recovers the original "extremal" prototypes while FCPM-*AE* follows the averaged pattern of FCM.

We also have experimented with simple data sets taken from the literature (butterfly [18], MS [15], wine [28], and Iris [29]). The results are concordant to the observations above. In particular, in these experiments, FCPM-0 behaves as an indicator of the number of clusters present in data: for the butterfly, wine and MS data sets the maximal numbers of prototypes at which FCPM-0 converges correspond to those in the original data (2, 3, and 3, respectively). For the Iris data set, FCPM-0 converges only when $c' = 2$, even though the original data set contains three classes (i.e., $c = 3$). This is consistent with the claim made by some authors that the underlying structure in the Iris data set, actually, may consist of two clusters only [1].

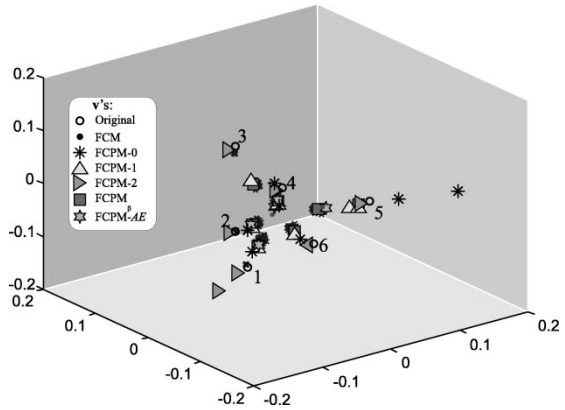


Fig. 7. 3-D plot of the prototypes found by FCM and FCPM with $c = c_0 + 1$ (high-dimension case with $p = 180$, $c_0 = 6$). FCM and FCPM-0 lead to different degenerate solutions: FCM makes all the prototypes coincide with each other, and FCPM-0 removes more than one prototype out of the data area. FCPM-1, FCPM-2 and FCPM-*AE* recover the original number of prototypes that have been generated; only FCPM_β finds $c' = c_0 + 1$ distinct prototypes.

TABLE III
CLUSTER PROTOTYPES ($\mathbf{v}_D$,$\mathbf{v}_M$,$\mathbf{v}_{Ss}$,$\mathbf{v}_{Sp}$), REVEALED BY FCM AND FCPM-2, PRESENTED IN THE ORIGINAL SPACE. VALUES CORRESPONDING TO THE MOST CONTRIBUTING FEATURES ARE MARKED

	FCM				FCPM-2			
h	v_D	v_M	v_{Ss}	v_{Sp}	v_D	v_M	v_{Ss}	v_{Sp}
h_1	4	1	3	3	6	0	2	4
h_2	4	2	2	4	5	1	0	5
h_3	5	[0]	5	3	6	[0]	6	4
h_4	2	3	3	4	1	3	3	6
h_5	[5]	1	1	1	[6]	0	0	0
h_6	2	5	2	4	1	6	0	5
h_7	1	1	2	3	0	0	3	4
h_8	[0]	[6]	[1]	5	[0]	[6]	[0]	6
h_9	[6]	1	2	2	[6]	0	1	1
h_{10}	2	4	2	5	0	4	1	6
h_{11}	2	2	2	[5]	1	2	1	[6]
h_{12}	1	1	1	4	0	1	0	6
h_{13}	[5]	[0]	2	[1]	[6]	[0]	3	[0]
h_{14}	3	4	2	5	2	4	1	6
h_{15}	3	3	3	[5]	2	2	2	[6]
h_{16}	2	[0]	[5]	2	3	[0]	[6]	1
h_{17}	[1]	[6]	[0]	4	[0]	[6]	[0]	5

V. CAPTURING IDEAL TYPES WITH FCPM

To show how a typological data structure can be caught with FCPM, we analyzed a specific data set from the field of psychiatry in which cluster prototypes, syndromes of mental conditions, are indeed extreme with regard to patients [30]. It is experimentally shown that prototypes found by FCPM remain unchanged when the set is augmented with patients having less severe syndromes. On the contrary, FCM as an averaging method tends to shift prototypes toward more moderate characteristics of the data. This highlights FCPM's suitability to model the concept of type in some domains.

A. Fuzzy Clustering of Mental Disorders Data

The mental disorders data set consists of 44 patients, described by seventeen psychosomatic features (h_1-h_{17}) (see [30] and [16]). The features are measured on a severity rating scale taking values of 0 to 6. The patients are partitioned into four classes of mental disorders: depressed (D), manic (M), simple schizophrenic (S_s), and paranoid schizophrenic (S_p). Each class contains eleven entities that are considered "archetypal psychiatric patients" of that class. Some properties of the data are as follows. First, there is always a pattern of features (a subset of h_1-h_{17}) that take extreme values (either 0 or 6) and clearly distinguish each class. Better still, some of these features take opposite values among distinct classes. However, some feature values are shared by classes leading to overlaps. Given these characteristics, each disease is characterized by "archetypal patients" that show a pattern of extreme psychosomatic feature values defining a syndrome.

The algorithms FCPM-2 and FCM (with its parameter $m = 2$) have been run starting from the same initial points, setting the number of clusters to four (i.e., $c = 4$). Table III shows the prototypes found by FCM and FCPM-2 in the

TABLE IV
PROTOTYPES FOUND BY FCM AND FCPM-2, IN THE ORIGINAL DATA SET (v'S) AND IN THE AUGMENTED DATA SET (v''S), CHARACTERIZING EACH MENTAL DISORDER: D, M, S_s, AND S_p

FCM

h	v_D	v'_D	v_M	v'_M	v_{Ss}	v'_{Ss}	v_{Sp}	v'_{Sp}
h_1	4	4	1	1	3	2	3	3
h_2	4	4	2	2	2	2	4	4
h_3	5	4	0	1	5	4	3	3
h_4	2	2	3	3	3	2	4	4
h_5	5	4	1	1	1	1	1	1
h_6	2	2	5	4	2	2	4	4
h_7	1	1	1	1	2	2	3	2
h_8	0	0	6	5	1	1	5	4
h_9	6	5	1	1	2	2	2	2
h_{10}	2	2	4	3	2	2	5	4
h_{11}	2	2	2	2	2	2	5	5
h_{12}	1	1	1	2	1	1	4	4
h_{13}	5	4	0	1	2	2	1	1
h_{14}	3	2	4	4	2	2	5	5
h_{15}	3	3	3	3	3	2	5	5
h_{16}	2	2	0	1	5	3	2	1
h_{17}	1	1	6	5	0	1	4	4

FCPM-2

h	v_D	v'_D	v_M	v'_M	v_{Ss}	v'_{Ss}	v_{Sp}	v'_{Sp}
h_1	6	5	0	0	2	2	4	4
h_2	5	5	1	1	0	0	5	5
h_3	6	5	0	0	6	5	4	4
h_4	1	1	3	3	3	2	6	5
h_5	6	6	0	0	0	0	0	0
h_6	1	1	6	6	0	0	5	5
h_7	0	0	0	0	3	2	4	4
h_8	0	0	6	6	0	0	6	6
h_9	6	6	0	0	1	1	1	1
h_{10}	0	0	4	4	1	1	6	6
h_{11}	1	1	2	2	1	0	6	6
h_{12}	0	1	1	1	0	0	6	6
h_{13}	6	6	0	0	3	2	0	0
h_{14}	2	2	4	4	1	1	6	6
h_{15}	2	2	2	3	2	1	6	6
h_{16}	3	3	0	0	6	6	1	1
h_{17}	0	0	6	6	0	0	5	4

original data scale. The marked values in the table belong to the set of most contributing features within a cluster (this is based on the concept of contribution weights of features [14]). The FCPM and FCM prototypes, at least in marked entries, are rather similar, though the feature values of FCPM-2 prototypes are somewhat more extremal than corresponding ones of FCM (which is in accordance with our simulation study).

Concerning the membership values found, both algorithms assign the highest belongingness of an entity to its original class, correctly clustering all entities to the corresponding class[1].

B. Clustering of Augmented Mental Disorders Data

In order to see the potential of FCPM-2 with regard to revealing extreme prototypes, the original data set should be modified by adding less expressed cases. To achieve that, each class was augmented with six mid-scale patient cases and three light-scale patient cases. Each new patient case, $\mathbf{x}_g = [x_{gh}]$, was generated from a randomly selected original patient, $\mathbf{x}_k = [x_{kh}]$, by applying the transformation

$$x_{gh} = \text{round}\,(s_F \cdot x_{kh}) + t, \qquad h = h_1, \ldots, h_{17}$$

with scale-factor $s_F = 0.6$ to obtain a mid-scale patient case and $s_F = 0.3$ to obtain a light-scale patient case. The shift parameter t takes values 0 or 1, randomly selected.

Table IV shows the prototypes ($\mathbf{v}$'s) found in the original data set followed by the corresponding prototypes ($\mathbf{v}'$'s) found in the augmented data set by FCPM-2 and FCM, where the values of most contributing features are boxed again.

We can see that FCM prototypes generally move toward intermediate feature values, showing FCM tendency to central prototypes. Contrastingly, in FCPM-2 prototypes, the most contributing features maintain their extreme values, reinforcing the extremal nature of FCPM-2 prototypes, despite the presence of mid- and light-scale patient cases. This is a situation for which the properties of the method make it an instrument capturing specifics of the extremal types that cannot be caught by averaging methods such as FCM (see also [14]).

In the augmented data set case all the original patients are still correctly assigned to the corresponding diseases, based on found memberships.

VI. CONCLUSION

The FCPM framework proposes a model of how data are generated from a cluster structure to be identified. This implies direct interpretability of the fuzzy membership values, which should be considered a motivation for introducing data-driven model-based methods. Another motivation comes from a restrictive character of such methods: each covers a specific type of cluster structure such as the FCPM reflected extremal type structure. On the other hand, the ability to reconstruct the data from the model is also a powerful characteristic of this approach.

The approach seems appealing in the sense that in many cases the experts of a knowledge domain have a conceptual understanding of how the domain is organized in terms of prototypes. This knowledge, put into the format of tentative prototypes, may well serve as the initial setting for data based structurization of the domain. In our approach, the belongingness of data entities to clusters is based on how much they share the features of corresponding prototypes. This seems useful in such application areas as mental disorders in psychiatry or consumer behavior in marketing. The extremal nature of prototypes in these domains can be well captured by the FCPM, as shown in the previous section.

Based on the experimental results of this research, we may conclude that FCPM-2 is able to recover the original prototypes from which data sets have been generated, while other criteria from the FCPM family tend to copy the FCM results. In other words, the FCPM-2 tends to find extremal prototypes while the other versions of FCPM favor central prototypes. The FCPM-0 (in the low-dimension case) and FCPM-2 (in the high-dimension case) can serve as indicators of the "natural" number of

[1]The only exception occurs for entity (21) from class M, which is assigned to class S_p; the same phenomenon is reported in [16] for other clustering algorithms such as complete linkage and K-means.

clusters present in the data according to the typological model. For the high dimension case, FCPM-1 and FCPM_β degenerate into hard clustering. FCM may decrease the number of prototypes in high dimension cases.

Results of the approach combining FCM and FCPM (algorithm FCPM-*AE*) point out some advantages of the FCPM proportional membership over the FCM distance-based membership, including increased discriminating power and robustness on high dimension spaces to identify a cluster structure (at least, with the generated data).

ACKNOWLEDGMENT

The first two authors would like to thank Dr. F. Roberts, Director of DIMACS, for providing them with opportunities to visit DIMACS for several times in 1998 and 1999. Dr. F. Roberts also coined the term "proportional membership for the proposed model." The authors would like to thank Dr. I. Muchnik, Prof. E. Baumann, Dr. B. Polyak, Prof. F. Klawonn for their valuable suggestions. The authors are grateful to the anonymous referees whose comments highly contributed to the revision of this paper.

REFERENCES

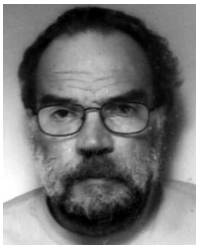
[1] J. Bezdek, *Pattern Recognition With Fuzzy Objective Function Algorithms*. New York: Plenum, 1981.
[2] J. Dunn, "A fuzzy relative of the isodata process and its use in detecting compact, well-separated clusters," *J. Cybernet.*, vol. 3, no. 3, pp. 32–57, 1974.
[3] J. Bezdek, J. Keller, R. Krishnapuram, and T. Pal, *Fuzzy Models and Algorithms for Pattern Recognition and Image Processing*. Norwell, MA: Kluwer, 1999.
[4] G. Gustafson and W. Kessel, "Fuzzy clustering with a fuzzy covariance matrix," in *Proc. IEEE Conf. Decision Control*, 1979, pp. 761–766.
[5] R. Davé, "Fuzzy shell-clustering and applications to circle detection of digital images," *Int. J. Gen. Syst.*, vol. 16, no. 4, pp. 343–355, 1990.
[6] L. Bobrowski and J. Bezdek, "c-Means with l_1 and l_∞ norms," *IEEE Trans. Syst., Man, Cybern.*, vol. 21, pp. 545–554, Mar. 1991.
[7] J. Bezdek, R. Hathaway, and N. Pal, "Norm induced shell prototype (NISP) clustering," *Neural, Parallel, Scient. Comput.*, vol. 3, pp. 431–450, 1995.
[8] F. Höppner, F. Klawonn, R. Kruse, and T. Runkler, *Fuzzy Cluster Analysis*. New York: Wiley, 1999.
[9] F. Klawonn and A. Keller, "Fuzzy clustering based on modified distance measures," in *Proc. 3rd Int. Symp. Advances Intelligent Data Analysis*, vol. 1642, J. Kok, D. Hand, and M. Berthold, Eds., 1999, pp. 291–301.
[10] R. Krishnapuram and J. Keller, "A possibilistic approach to clustering," *IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 98–110, Apr. 1993.
[11] R. Hathaway and J. Bezdek, "Switching regression models and fuzzy clustering," *IEEE Trans. Fuzzy Syst.*, vol. 1, pp. 195–204, June 1993.
[12] S. Nascimento, B. Mirkin, and F. Moura-Pires, "Multiple prototypes model for fuzzy clustering," in *Proc. 3rd Int. Symp. Advances Intelligent Data Analysis*, vol. 1642, J. Kok, D. Haud, and M. Berthold, Eds., 1999, pp. 269–279.
[13] ——, "A fuzzy clustering model of data and fuzzy c-means," in *Proc. 9th IEEE Int. Conf. Fuzzy Systems*, May 2000, pp. 302–307.
[14] ——, "Proportional membership in fuzzy clustering as a model of ideal types," in *Proc. 10th Portuguese Conf. Artificial Intelligence*, P. Brazdil and A. Jorge, Eds., 2001, pp. 52–62.
[15] B. Mirkin and G. Satarov, "Method of fuzzy additive types for analysis of multidimensional data," *Autom. Remote Control, I, II*, vol. 51, no. 5, 6, pp. 683–688, 1990.
[16] B. Mirkin, *Mathematical Classification and Clustering*. Norwell, MA: Kluwer, 1996.
[17] T. Runkler and J. Bezdek, "Alternating cluster estimation: A new tool for clustering and function approximation," *IEEE Trans. Fuzzy Syst.*, vol. 7, pp. 377–393, Apr. 1999.
[18] E. Ruspini, "A new approach to clustering," *Inform. Control*, vol. 15, pp. 22–32, 1969.
[19] J. Bezdek, R. Hathaway, M. Sabin, and W. Tucker, "Convergence theory for fuzzy c-means: Counterexamples and repairs," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-17, pp. 873–877, May 1987.
[20] S. Nascimento, "Soft clustering with the multiple prototype fuzzy clustering model," in *Applied Stochastic Models and Data Analysis: Quantitative Methods in Business and Industry Society*, F. C. Nicolau, H. B. Nicolau, and J. Janssen, Eds. Lisbon, Portugal: INE, 1999, pp. 249–255.
[21] S. Selim and M. Ismail, "Soft clustering of multidimensional data: A semi-fuzzy approach," *Pattern Recognit.*, vol. 17, no. 5, pp. 559–568, 1984.
[22] B. Polyak, *Introduction to Optimization*. New York: Optimization Software, Inc., 1987.
[23] D. Bertsekas, *Nonlinear Programming*. Belmont, MA: Athena Scientific, 1995.
[24] G. W. Milligan, "An algorithm for generating artificial test clusters," *Psychometrika*, vol. 50, no. 1, pp. 123–127, 1985.
[25] A. Jain and R. Dubes, *Algorithms for Clustering Data*. Upper Saddle River, NJ: Prentice-Hall, 1988.
[26] *MATLAB: The Language of Technical Computing (ver. 5.2)*, The MathWorks, Inc., Natick, MA, 1998.
[27] L. Jimenez and D. Landgrebe, "Supervised classification in high-dimensional space: Geometrical, statistical and asymptotical properties of multivariate data," *IEEE Trans. Syst., Man, Cybern. C*, vol. 28, pp. 39–54, Jan. 1998.
[28] C. Blake, E. Keogh, and C. Merz. (1998) UCI Repository of Machine Learning Databases. Dept. Inform. Comput. Sci., Univ. California, Irvine, CA. [Online]. Available: http://www.ics.uci.edu/learn/~ML-Repository.html
[29] J. Bezdek, J. M. Keller, L. Kuncheva, R. Krisnapuram, and N. Pal, "Will the real iris data please stand up?," *IEEE Trans. Fuzzy Syst.*, vol. 7, pp. 368–369, Mar. 1999.
[30] J. E. Mezzich and H. Solomon, *Taxonomy and Behavioral Science*. London, U.K.: Academic, 1980.
[31] B. Mirkin, "Concept learning and feature selection based on square-error clustering," *Mach. Learn.*, vol. 25, no. 1, pp. 25–40, 1999.

Susana Nascimento received the Diploma and Ph.D. degrees in computer science from the Universidade Nova de Lisboa, Lisbon, Portugal, in 1989 and 2002, respectively.

She is currently a Researcher at the Artificial Intelligence Research Center (CENTRIA) of the Faculty of Science and Technology, Universidade Nova de Lisboa, Lisbon, Portugal. Her research interests include fuzzy clustering, genetic algorithms, data mining, and machine learning.

Boris Mirkin received the M.S. and Ph.D. degrees in computer science from Saratov State University, Saratov, Russia, and the D.Sc. degree from the Institute of Systems Sciences, Russian Academy of Sciences, Moscow, Russia, in 1964, 1966, and 1990, respectively.

He is currently a Professor of Computer Science in the School of Computer Science and Information Systems, Birkbeck College, University of London, London, U.K. His research interests include developing methods for data mining and clustering with various types of data and applications in text analysis and bioinformatics.



Fernando Moura-Pires received the Diploma in electrical engineering from the University of Angola, Luanda, Angola, and the Ph.D. degree in computer science from the Universidade Nova de Lisboa, Lisbon, Portugal, in 1972 and 1993, respectively.

He is currently a Professor in the Computer Science Department of the Universidade de Évora, Évora, Portugal. His research interests include machine learning, data mining, and statistical learning.