The Fuzzy Hierarchical Cross-Clustering Algorithm. Improvements and Comparative Study[§]

Horia F. Pop*,[†] and Costel Sârbu[‡]

Faculty of Mathematics and Computer Science and Faculty of Chemistry, "Babes-Bolyai" University, RO-3400 Cluj-Napoca, Romania

Received October 24, 1996[⊗]

The aim of this paper is to introduce a way to improve the behavior of the Fuzzy Hierarchical Cross-Clustering Algorithm. The Simultaneous Fuzzy *n*-Means Algorithm produces fuzzy *n*-partitions of objects and characteristics, but it does not offer a modality to associate each fuzzy set of objects to a fuzzy set of characteristics. Based on our previous experience we are able to propose such an association. In the second part of the paper we will study the behavior of the Fuzzy Hierarchical Cross-Clustering Algorithm using both the original and the Improved Simultaneous Fuzzy *n*-Means Algorithm.

1. INTRODUCTION

The fuzzy sets theory, developed by Zadeh[13] allows an object to belong to many clusters with different membership degrees. The membership degree of an object to a certain class is supposed to be gradual rather than abrupt (either 0 or 1), revealing a basis for considering uncertainty and imprecision.

There are two opposite approaches to hierarchical clustering, namely *agglomerative* and *divisive* procedures. An agglomerative hierarchical classification places each object in its own cluster and gradually merges the clusters into larger and larger clusters until all objects are in a single cluster. The divisive hierarchical clustering reverses the process by starting with all the objects in a single cluster and subdividing it into smaller ones until, finally, each object is in a cluster of its own. The number of clusters to be generated may be either specified in advance or optimized by the algorithm itself according to certain criteria.

Among other interesting applications, the fuzzy clustering theory developed in refs 2−4 and 7 has been used for the selection and the optimal combination of solvents,[5,11] for the classification of Roman pottery,[8] for the cross-classification of Greek muds,[4] for the development of a fuzzy system of chemical elements,[10,12] and for producing a performant fuzzy regression algorithm.[9]

In the present paper we will approach the fuzzy hierarchical cross-classification algorithm (see also refs 3 and 4). We will show a problem in the algorithm, and we will propose a solution for it. In order to demonstrate the quality of our improvement, we will do a comparative study based on the behavior both of the original fuzzy hierarchical algorithm and of our improved version of it.

2. HIERARCHICAL CROSS-CLASSIFICATION

In certain situations the number of characteristics is very large. The design of a hierarchical classifier may be simplified if at every node is used only a small subset of the characteristics, enough for the classification decision at that node. So, at every step of the hierarchical classification process we determine a fuzzy partition of a certain class and the relevant characteristics for each of the subclasses obtained. Thus, it is necessary to classify both the objects and the characteristics. This classification procedure will be called cross-classification (or simultaneous classification).

In what follows we will recall[4,3] a method which allows us to obtain an objects hierarchy and a characteristics hierarchy, so that the two hierarchical classifications should correspond to each other. This method is iterative, and the classification is done alternatively on the data set and on the characteristics set, until we will obtain two "compatible" classifications.

2.1. Fuzzy Substructure of a Fuzzy Set. In this section we will recall the so-called Generalized Fuzzy *n*-Means Algorithm.[2−4] This algorithm is a generalization of the well-known Fuzzy *n*-Means Algorithm.[3,1]

Let us consider a set of objects $X = \{x^1, ..., x^p\} \subset R^s$ and let C be a fuzzy set on X. We search for the cluster substructure of the fuzzy set C.

Let us suppose that the fuzzy partition corresponding to this substructure is $\{A_1, ..., A_n\}$. We admit that each class A_i may be represented by a prototype L^i from the representation space, R^s. If L^i is from X we may suppose that L^i has the greatest membership degree to A_i, that is

$$A_i(L^i) = \max_{x \in X} A_i(x) \tag{1}$$

Let us denote by d a distance in the space R^s. For example, we may consider the distance induced by the norm of the space.

The dissimilarity $D_i(x^j, L^i)$ between a point x^j and the prototype L^i is defined as the square distance to the class A_i and is interpreted as a measure of the inadequacy of the representation of the point x^j by the prototype L^i.

If L^i is not a point from the data set X, then we have from refs 3 and 2 that

$$D_i(x^j, L^i) = (A_i(x^j))^2 d^2(x^j, L^i) \tag{2}$$

[†] Faculty of Mathematics and Computer Science. E-mail address: hfpop@cs.ubbcluj.ro.

[‡] Faculty of Chemistry. E-mail address: csarbu@chem.ubbcluj.ro.

[§] Keywords: fuzzy hierarchical cross-classification, pattern recognition, solvents.

[⊗] Abstract published in *Advance ACS Abstracts,* April 1, 1997.

S0095-2338(96)00148-5 CCC: $14.00

The inadequacy between the fuzzy partition P and its representation, $L = \{L^1, ..., L^n\}$ is given by the following function:

$$J(P, L) = \sum_{i=1}^{n}\sum_{j=1}^{p}(A_i(x^j))^2 d^2(x^j, L^i) \quad (3)$$

$J(P, L)$ may also be interpreted as the representation error of P by L.

Let us observe that J is a criteria function of the type of square errors sum. The classification problem becomes the determination of the fuzzy partition P and its representation L for which the inadequacy $J(P, L)$ is minimal. Let us note that, intuitively, to minimize J means to give small membership degrees to A_i for those points in X for which the dissimilarity to the prototype L^i is large and vice versa.

If we admit that d is a distance induced by the norm, we may write

$$J(P, L) = \sum_{i=1}^{n}\sum_{j=1}^{p}(A_i(x^j))^2 ||x^j - L^i||^2 \quad (4)$$

If the norm is induced by the inner product, we have

$$||x^j - L^i||^2 = (x^j - L^i)^T M(x^j - L^i)$$

where M is a symmetrical and positively defined matrix. The transposing operation was denoted by ^{T}.

The criteria function becomes

$$J(P, L) = \sum_{i=1}^{n}\sum_{j=1}^{p}(A_i(x^j))^2 (x^j - L^i)^T M(x^j - L^i) \quad (5)$$

Because an algorithm to obtain an exact solution of the problem 5 is not known, we will use an approximate method in order to determine a local solution. The minimum problem will be solved using an iterative (relaxation) method, where J is successively minimized with respect to P and L.

Supposing that L is given, the minimum of the function $J(\cdot, L)$ is obtained[3,2] for

$$A_i(x^j) = \frac{C(x^j)}{\sum_{k=1}^{n}\frac{d^2(x^j, L^i)}{d^2(x^j, L^k)}} \quad (6)$$

if for all i, $d(x^j, L^i) \neq 0$, and, respectively, if for a certain x^j there is an L^i such that $d(x^j, L^i) = 0$, the memberships of x^j verify the condition

$$A_i(x^j) = 0, \forall i \text{ such that } d(x^j, L^i) \neq 0$$

For a given P, the minimum of the function $J(P, \cdot)$ is obtained for

$$L^i = \frac{\sum_{j=1}^{p}(A_i(x^j))^2 x^j}{\sum_{j=1}^{p}(A_i(x^j))^2}, \quad i = 1, 2, ..., n \quad (7)$$

We observe[3,2] that L^i is the weighting center of the class A_i.

The iterative procedure for obtaining the cluster substructure of the fuzzy class C is called Generalized Fuzzy n-Means (GFNM).[2] Essentially, the GFNM algorithm works with Picard iterations using the relations 6 and 7. The iterative process begins with an arbitrary initialization of the partition P. The process ends when two successive partitions are close enough. To measure the distance between two partitions, we will associate to each partition P a matrix Q with the dimensions $n \times p$. Q is named the representation matrix of the fuzzy partition P and is defined as

$$Q_{ij} = A_i(x^j), \quad i = 1, 2, ..., n; \quad j = 1, 2, ..., p \quad (8)$$

Considering that Q_1 and Q_2 are the representation matrices of the partitions P_1 and P_2, we may define

$$d(P_1,P_2) = ||Q_1 - Q_2|| \quad (9)$$

where $||Q|| = \max_{i,j}|A_i(x^j)|$.

The process ends at the rth iteration if

$$d(P_r,P_{r+1}) < \epsilon \quad (10)$$

where ϵ is an admissible error (usually, 10^{-5}).

For $C = X$ this procedure is the well-known algorithm Fuzzy n-Means (FNM).[1]

2.2. Simultaneous Fuzzy *n*-Means Algorithm. Let $X = \{x^1, ..., x^p\} \subset R^d$ be the set of objects to be classified. A characteristic may be specified by its values corresponded to the p objects. So, we may say that $Y = \{y^1, ..., y^d\} \subset R^p$ is the set of characteristics. y_j^k is the value of the characteristic k with respect to the object j, so we may write $y_j^k = x_k^j$.

Let P be a fuzzy partition of the fuzzy set C of objects and Q be a fuzzy partition of the fuzzy set D of characteristics. The problem of the cross-classification (or simultaneous classification) is to determine the pair (P, Q) which optimizes a certain criterion function. By starting with an initial partition P^0 of C and an initial partition Q^0 of D, we will obtain a new partition P^1. The pair (P^1, Q^0) allows us to determine a new partition Q^1 of the characteristics. The algorithm consists in producing a sequence (P^k, Q^k) of pairs of partitions, starting from the initial pair (P^0, Q^0), in the following steps:

$$\text{(i) } (P^k, Q^k) \rightarrow (P^{k+1}, Q^k)$$

$$\text{(ii) } (P^{k+1}, Q^k) \rightarrow (P^{k+1}, Q^{k+1})$$

The rationale of the hierarchical cross-classification method[3] essentially supposes the splitting the sets X and Y in two subclasses. The obtained classes are alternatively divided in two subclasses and so on. The two hierarchies may be represented by the same tree, having in each node a pair (C, D), where C is a fuzzy set of objects and D is a fuzzy set of characteristics.

As a first step we propose ourselves to simultaneously determine the fuzzy partitions (as a particular case, the binary fuzzy partitions) of the classes C and D, so that the two partitions should be highly correlated.

With the Generalized Fuzzy *n*-Means algorithm, we will determine a fuzzy partition $P = \{A_1, ..., A_n\}$ of the class C, using the original characteristics.

In order to classify the characteristics, we will compute their values for the classes A_i, $i = 1, ..., n$. The value $\bar{y}_i^k$ of the characteristic k with respect to the class A_i is defined as

$$\bar{y}_i^k = \sum_{j=1}^{p} A_i(x^j)x_k^j, \quad i = 1, ..., n; \quad k = 1, ..., d \qquad (11)$$

Thus, from the original d p-dimensional characteristics we computed d new n-dimensional characteristics which are conditioned by the classes A_i, $i = 1, ..., n$. We may admit that these new characteristics do not describe objects, but they characterize the classes A_i.

Let us consider now the set $\bar{Y} = \{\bar{y}^1, ..., \bar{y}^d\}$ of the modified characteristics. We define the fuzzy set $\bar{D}$ on $\bar{Y}$, given by

$$\bar{D}(\bar{y}^k) = D(y^k), \quad k = 1, ..., d$$

The way the set $\bar{Y}$ has been obtained lets us to conclude that if we will obtain an optimal partition of the fuzzy set D, this partition will be highly correlated to the optimal partition of the class C.

With the Generalized Fuzzy *n*-Means algorithm we will determine a fuzzy partition $Q = \{B_1, ..., B_n\}$ of the class D, by using the characteristics given by the relation 11.

We may now characterize the objects in X with respect to the classes of properties B_i, $i = 1, ..., n$. The value $\bar{x}_i^j$ of the object j with respect to the class B_i is defined as

$$\bar{x}_i^j = \sum_{k=1}^{d} B_i(\bar{y}^k)x_k^j, \quad i = 1, ..., n; \quad k = 1, ..., d \qquad (12)$$

Thus, from the original p d-dimensional objects we have computed p new n-dimensional objects, which correspond to the classes of characteristics B_i, $i = 1, ..., n$.

Let us now consider the set $\bar{X} = \{\bar{x}^1, ..., \bar{x}^p\}$ of the modified objects. We define the fuzzy set $\bar{C}$ on $\bar{X}$, given by

$$\bar{C}(\bar{x}^j) = C(x^j), \quad j = 1, ..., p$$

With the Generalized Fuzzy *n*-Means algorithm we will determine a fuzzy partition $P' = \{A_1', ..., A_n'\}$, of the class C by using the objects given by the relation 12.

The process continues until two successive partitions of objects (or of characteristics) are close enough to each other. Thus, we have obtained **The Simultaneous Fuzzy *n*-Means Algorithm** (see refs 3 and 4).

S1 Set $l = 0$. With the Generalized Fuzzy *n*-Means Algorithm we determine a fuzzy *n*-partition $P^{(l)}$ of the class C by using the initial objects.

S2 With the Generalized Fuzzy *n*-Means Algorithm we determine a fuzzy *n*-partition $Q^{(l)}$ of the class D by using the characteristics defined in (11).

S3 With the Generalized Fuzzy *n*-Means Algorithm we determine a fuzzy *n*-partition $P^{(l+1)}$ of the class C by using the objects defined in (12).

S4 If the partitions $P^{(l)}$ and $P^{(l+1)}$ are close enough, that is if

$$||P^{(l+1)} - P^{(l)}|| < \epsilon$$

where ϵ is a preset value, then **Stop**, else increase l by 1 and go to **S2**.

2.3. Associative Simultaneous Fuzzy *n*-Means (ASF) Algorithm. Let us remark that the Simultaneous Fuzzy *n*-Means Algorithm makes no explicit association of a fuzzy set A_i on X with a fuzzy set B_j on Y, i.e., what is the fuzzy set B_j that best describes the essential characteristics corresponding to the fuzzy set A_i. It only supposes that A_i is to be associated with B_i, and this is not always true, and thus it may sometimes produce wrong results. In this section we will correct this problem by proposing a better way of associating the fuzzy sets of objects and characteristics.

Let us have the set $X = \{x^1, ..., x^n\} \subset \mathrm{R}^d$ of classification objects, and let Y be the correspondent set of characteristics. Let us also consider the fuzzy sets C on X and D on Y.

Let us suppose that $P = \{A_1, ..., A_n\}$ is the fuzzy *n*-partition of X and $Q = \{B_1, ..., B_n\}$ is the fuzzy *n*-partition of Y produced by the Simultaneous Fuzzy *n*-Means Algorithm.

Let us denote by S_n the set of all permutations on $\{1, ..., n\}$. We wish to build that permutation $\sigma \in S_n$ which best associates the fuzzy set A_i with the fuzzy set $B_{\sigma(i)}$, for every $i = 1, ..., n$.

Our aim is to build some function $J : S_n \rightarrow \mathrm{R}$ so that the optimal permutation σ is that which maximizes this function.

Let us consider the matrix $Z \in \mathrm{R}^{n,n}$ given by

$$z_{kl} = \sum_{j=1}^{p}\sum_{i=1}^{d} A_k(x^j)\, B_l(y^i)x_j^i, \; k, l = 1, ..., n \qquad (13)$$

Let us remark the similarity between the way we compute the matrix Z in (13) and the way was computed the new objects and characteristics in relations 11 and 12.

The experience enables us to consider the function J as given by

$$J(\sigma) = \prod_{i=1}^{n} z_{i,\sigma(i)} \qquad (14)$$

Thus, supposing that the permutation σ maximizes the function J defined above, we will be able to associate the fuzzy set A_i with the fuzzy set $B_{\sigma(i)}$, $i = 1, ..., n$. As we will see in the comparative study below, this association is more natural than the association of A_i with B_i, $i = 1, ..., n$.

Based on these considerations we are able to introduce the following algorithm, that we have called **The Associative Simultaneous Fuzzy *n*-Means Algorithm (ASF)**.

S1 Set $l = 0$. With the Generalized Fuzzy *n*-Means Algorithm we determine a fuzzy *n*-partition $P^{(l)}$ of the class C by using the initial objects.

S2 With the Generalized Fuzzy *n*-Means Algorithm we determine a fuzzy *n*-partition $Q^{(l)}$ of the class D by using the characteristics defined in (11).

S3 With the Generalized Fuzzy *n*-Means Algorithm we determine a fuzzy *n*-partition $P^{(l+1)}$ of the class C by using the objects defined in (12).

Table 1. Composition of the Tested Data Set

object	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	9	7	8	6	0	0	0	0	0	0	0	0	0	0	0	0
2	5	4	6	7	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	9	4	5	7	0	0	0	0	0	0	0	0	0
4	0	0	0	8	2	3	6	0	0	0	0	0	0	0	0	0
5	0	0	0	2	7	5	3	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	8	6	7	9	3	0	0	0	0
7	0	0	0	0	0	0	0	9	5	4	8	7	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	4	9	8	0	0
9	0	0	0	0	0	0	0	0	0	0	0	7	6	5	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	2	3	4
11	0	0	0	0	0	0	0	0	0	0	0	0	0	9	8	6

S4 If the partitions $P^{(l)}$ and $P^{(l+1)}$ are close enough, that as if

$$||P^{(l+1)} - P^{(l)}|| < \epsilon$$

where ϵ is a preset value, then go to **S5**, else increase l by 1 and go to **S2**.

S5 Compute the permutation σ that maximizes the function J given in relation 14.

S6 Relabel the fuzzy sets B_i, so that $B_{\sigma(i)}$ becomes B_i, $i = 1, ..., n$.

Let us remark now that, after the steps **S5** and **S6**, we are able to associate the fuzzy set A_i with the fuzzy set B_i, $i = 1, ..., n$.

The single problem that needs to be detailed is the computation implied at the step **S5**. As we will see in the following section, this algorithm is intended to be used in a hierarchic procedure for obtaining a binary classification tree. Thus, the ASF algorithm will only be used for $n = 2$, and the problem of the computation at the step **S5** is trivial. Of course, for the general case where $n > 2$, the problem of this computation needs to be addressed separately, but, as we said, we are interested here in the hierarchic algorithm.

2.4. Fuzzy Hierarchical Cross-Classification Algorithm. In this section we will present three variants of the procedure of hierarchical cross-classification. As we will see, these variants are useful when approaching different kinds of problems related to objects−characteristics relationships.

Variant A. The method described below is the straightforward way of developing a hierarchical algorithm that should use at each node of the classification tree our ASF algorithm for $n = 2$.

We will first show the way of building the classification binary tree.

The nodes of the tree are labeled with a pair (C, D), where C is a fuzzy set from a fuzzy partition of objects and D is a fuzzy set from a fuzzy partition of characteristics. The root node corresponds to the pair (X, Y). In the first step the two subnodes (A_1, B_1) and respectively (A_2, B_2) will be computed by using the ASF algorithm. Of course, these two nodes will be effectively built only if the fuzzy partitions $\{A_1, A_2\}$ and $\{B_1, B_2\}$ describe real clusters.

For each of the terminal nodes of the tree we try to determine partitions having the form $\{A_1, A_2\}$ and $\{B_1, B_2\}$, by using the ASF algorithm, modified as we have mentioned before. In this way the binary classification tree is extended with two new nodes, (A_1, B_1) and (A_2, B_2). The process continues until for any terminal node we are not able to determine a structure of real clusters, either for the set of objects or for the set of characteristics. The final fuzzy partitions will contain the fuzzy sets corresponding to the terminal nodes of the binary classification tree.

This variant of the algorithm, called **FHCCA algorithm**, seems to be suitable for application where the important idea is to get most of the relationships between different classes of objects and different classes of characteristics.

Variant B. The classification binary tree is built in the same way as for the variant A of the algorithm. However, we slightly change the way the ASF algorithm is applied.

Let us denote the final partitions obtained at a certain node of the classification tree by $P = \{A_1, ..., A_n\}$ and $Q = \{B_1, ..., B_n\}$.

When trying to build the fuzzy partitions corresponding to the fuzzy sets A_i and B_i, $i = 1, ..., n$, with the help of the ASF algorithm, at the step **S1** of the algorithm, instead of using the original objects we may use the objects defined in the relation 12 and determined at the step **S3** before the end of the algorithm that produced the fuzzy partitions P and Q.

This variant of the algorithm, called **FHCCB algorithm**, is quite similar to the FHCCA algorithm. However, let us remark the way the information obtained at a certain node of the classification tree is used when working on the subsequent nodes. Thus, suitable for application where the important idea is to get most of the relationships between different classes of objects and different classes of characteristics.

Variant C. The classification binary tree is built in the same way as for the variants A and B of the algorithm.

This time, the change in how we apply the ASF algorithm is more substantial.

When trying to build the fuzzy partition at a certain node of the classification tree, at the step **S3** of the ASF algorithm, instead of using the objects defined in the relation 12, we will use the original objects, as in step **S1**.

This variant of the algorithm, called **FHCCC algorithm**, is very different from the FHCCA and FHCCB algorithms. Due to the way it works, it seems to be suitable for applications where we are interested not only in a fuzzy partition corresponding to the cluster substructure of the data set but also in knowing which fuzzy class of characteristics is responsible for the separation of each fuzzy class of objects. Let us remark the way this behavior is different from the behaviors of the FHCCA and FHCCB algorithms.

3. CLASSIFICATION RESULTS AND DISCUSSIONS

In this section we will present the results obtained by using the algorithms introduced in this paper (FHCCA, FHCCB, and FHCCC) in order to simultaneously classify two different

Table 2. Membership Degrees to the Classes of the Simultaneous Final Fuzzy Partition Obtained Using the FHCCA Algorithm

object	11	12	21	221	222
1	0.92	0.03	0.01	0.02	0.02
2	0.96	0.04	0.00	0.00	0.00
3	0.02	0.97	0.00	0.00	0.00
4	0.01	0.98	0.00	0.00	0.01
5	0.00	0.97	0.01	0.01	0.01
6	0.03	0.04	0.94	0.00	0.00
7	0.03	0.04	0.94	0.00	0.00
8	0.00	0.01	0.00	0.97	0.01
9	0.02	0.03	0.01	0.93	0.01
10	0.14	0.20	0.06	0.13	0.48
11	0.00	0.00	0.01	0.01	0.97
	characteristics				
1	0.99	0.00	0.00	0.00	0.00
2	0.92	0.01	0.03	0.01	0.02
3	0.99	0.00	0.00	0.00	0.00
4	0.25	0.55	0.10	0.03	0.07
5	0.01	0.97	0.01	0.00	0.01
6	0.01	0.97	0.01	0.00	0.01
7	0.01	0.99	0.00	0.00	0.00
8	0.00	0.00	0.97	0.01	0.02
9	0.03	0.02	0.92	0.01	0.02
10	0.03	0.02	0.92	0.01	0.02
11	0.00	0.00	0.97	0.01	0.02
12	0.03	0.02	0.38	0.52	0.04
13	0.00	0.00	0.01	0.97	0.02
14	0.06	0.04	0.11	0.56	0.24
15	0.03	0.03	0.05	0.00	0.89
16	0.06	0.05	0.09	0.00	0.79

Table 3. Membership Degrees to the Classes of the Simultaneous Final Fuzzy Partition Obtained Using the FHCCB Algorithm

object	11	12	21	221	222
1	0.92	0.03	0.01	0.01	0.03
2	0.96	0.04	0.00	0.00	0.00
3	0.02	0.97	0.00	0.00	0.00
4	0.01	0.98	0.00	0.00	0.01
5	0.00	0.97	0.01	0.01	0.01
6	0.03	0.04	0.94	0.00	0.00
7	0.03	0.04	0.94	0.00	0.00
8	0.00	0.01	0.00	0.96	0.03
9	0.02	0.03	0.01	0.16	0.79
10	0.14	0.20	0.06	0.09	0.52
11	0.00	0.00	0.01	0.96	0.02
	characteristic				
1	0.99	0.00	0.00	0.00	0.00
2	0.92	0.01	0.03	0.01	0.03
3	0.99	0.00	0.00	0.00	0.00
4	0.25	0.55	0.10	0.02	0.08
5	0.01	0.97	0.01	0.00	0.01
6	0.01	0.97	0.01	0.00	0.01
7	0.01	0.99	0.00	0.00	0.00
8	0.00	0.00	0.97	0.00	0.02
9	0.03	0.02	0.92	0.00	0.02
10	0.03	0.02	0.92	0.00	0.02
11	0.00	0.00	0.97	0.00	0.02
12	0.03	0.02	0.38	0.04	0.52
13	0.00	0.00	0.01	0.10	0.89
14	0.06	0.04	0.11	0.80	0.00
15	0.03	0.03	0.05	0.02	0.87
16	0.06	0.05	0.09	0.02	0.78

sets of data: the former one is a theoretically produced data set, and the latter is an example from the thin layer chromatography (TLC).

3.1. Theoretical Experiments. The data set discussed here is presented in Table 1. As we may see, the matrix is of diagonal matrix type. In order to produce a simultaneous fuzzy partition that should correspond to the cluster substructure and to the interrelations from the data set, we used the FHCCA, FHCCB, and FHCCC algorithms. The results obtained are displayed in Tables 2−4, respectively.

Table 4. Membership Degrees to the Classes of the Simultaneous Final Fuzzy Partition Obtained Using the FHCCC Algorithm

object	11	12	21	22
1	0.58	0.07	0.15	0.21
2	0.71	0.02	0.10	0.17
3	0.06	0.74	0.08	0.12
4	0.04	0.80	0.06	0.10
5	0.13	0.58	0.10	0.19
6	0.18	0.19	0.60	0.02
7	0.17	0.17	0.63	0.02
8	0.14	0.15	0.07	0.64
9	0.11	0.12	0.07	0.70
10	0.17	0.20	0.12	0.50
11	0.18	0.19	0.14	0.49
	characteristic			
1	0.96	0.00	0.01	0.02
2	0.80	0.01	0.08	0.11
3	0.97	0.00	0.01	0.02
4	0.20	0.45	0.08	0.27
5	0.02	0.96	0.01	0.01
6	0.01	0.96	0.01	0.01
7	0.02	0.93	0.02	0.03
8	0.01	0.00	0.97	0.02
9	0.12	0.07	0.76	0.04
10	0.12	0.08	0.76	0.04
11	0.01	0.00	0.97	0.02
12	0.08	0.05	0.26	0.61
13	0.02	0.01	0.02	0.95
14	0.12	0.09	0.09	0.70
15	0.13	0.08	0.10	0.68
16	0.17	0.11	0.12	0.61

As we may notice from Table 2, the results produced by the FHCCA algorithm correspond to the results obtained by visual inspection. In order not to allow the final fuzzy classes to divide, we set the threshold of the separation index to a value of 0.71. Thus, the simultaneous final fuzzy partition corresponds to the results of the visual inspection. Our need to impose such a high threshold confirms the quality of the algorithms results in this case of well separated clusters.

Let us denote the membership degrees of the characteristics numbered 4, 12, and 14. These objects are, naturally, split between two different classes.

If we analyze now the results produced by the FHCCB algorithms, from Table 3, we remark the different way the classes 221 and 222 look. The system erroneously classified both the objects and the characteristics associated to the classes 221 and 222. However, the threshold of the separation index was set here to 0.71, either.

If we analyze the results displayed in Table 4, we note that the FHCCC algorithm produce incorrect results with respect to the same class, 22. The class denoted 22 could not be more divided, as the visual inspection would indicate. The partition obtained by classifying the class 22 does not correspond to real clusters. Moreover, in order to produce this simultaneous fuzzy partition, for the reason mentioned above, we used a threshold of the separation index of 0.40.

3.2. Practical Experiments. In thin layer chromatography (TLC) the choice of the minimum number of eluent systems containing different information is of crucial importance for the identification of unknowns and has been the topic of several statistical studies. The individual

Table 5. Values of hR_f for 20 Compounds in 18 Solvent Systems

eluent	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
atropine	20	16	29	23	62	33	4	2	13	47	25	42	18	12	0	0	5	8
biperide	91	90	87	68	92	87	73	72	64	85	81	86	68	94	11	40	40	65
caffeine	55	42	52	68	77	54	8	5	13	60	30	51	41	20	13	12	54	57
cocaine	81	82	81	71	87	82	46	41	38	81	72	80	52	72	6	24	30	57
codeine	38	31	44	39	71	43	12	9	16	49	29	36	22	14	0	0	15	21
cyclizin	71	72	80	64	85	80	49	47	40	75	71	73	39	59	2	9	34	54
diazepam	76	79	80	78	85	80	28	21	29	80	61	75	72	54	54	50	85	87
ketamine	77	79	80	76	86	79	41	33	32	81	66	76	67	66	27	37	66	79
lignocaine	77	79	80	73	86	80	35	30	28	84	73	77	66	64	25	54	68	84
lorazerpam	47	34	53	77	73	46	2	0	12	52	7	22	47	8	28	15	85	79
mebeverine	85	90	90	65	90	85	43	33	38	88	70	87	62	76	5	29	29	53
methadon	85	84	88	48	89	83	63	64	48	85	73	86	38	86	1	10	13	29
morphine	18	9	15	39	56	20	2	0	5	20	2	8	13	3	0	1	16	18
naloxone	48	40	62	75	79	48	15	11	22	52	26	40	60	21	18	21	67	77
papaverine	68	66	76	79	88	71	12	7	18	78	56	62	54	30	28	41	76	80
pentazoc	72	66	81	65	87	76	22	18	26	69	41	54	39	44	2	11	32	59
phenacet	64	58	62	79	87	66	4	1	13	68	18	41	58	24	41	40	86	84
phenazon	66	53	70	83	86	65	30	22	24	77	60	66	45	54	15	21	68	74
prazepam	81	83	86	83	88	81	41	31	35	83	66	82	75	74	65	67	86	88
procaine	64	60	70	65	82	73	8	5	16	66	24	54	37	50	1	11	29	53

Table 6. Membership Degrees to the Classes of the Simultaneous Final Fuzzy Partition Obtained Using the FHCCA Algorithm

drugs	1111	1112	112	121	122	211	212	221	222
atropine	0.85	0.05	0.00	0.00	0.02	0.02	0.05	0.01	0.01
diperide	0.02	0.01	0.02	0.02	0.03	0.60	0.06	0.04	0.21
caffeine	0.00	0.00	0.88	0.01	0.05	0.05	0.00	0.00	0.00
cocaine	0.00	0.00	0.00	0.00	0.00	0.94	0.01	0.01	0.02
codeine	0.42	0.05	0.26	0.02	0.21	0.00	0.02	0.00	0.00
cyclizin	0.01	0.00	0.01	0.01	0.01	0.79	0.17	0.00	0.00
diazepam	0.01	0.00	0.01	0.02	0.01	0.04	0.01	0.88	0.02
ketamine	0.00	0.00	0.00	0.00	0.00	0.06	0.01	0.01	0.92
lignocaine	0.00	0.00	0.00	0.00	0.00	0.04	0.01	0.00	0.94
lorazerpam	0.17	0.05	0.01	0.00	0.73	0.01	0.00	0.00	0.04
mebeverine	0.00	0.00	0.01	0.01	0.01	0.96	0.00	0.00	0.01
metadon	0.02	0.01	0.02	0.02	0.02	0.00	0.82	0.03	0.07
morphine	0.00	0.79	0.01	0.00	0.06	0.02	0.09	0.01	0.02
naloxone	0.01	0.00	0.01	0.79	0.06	0.04	0.00	0.00	0.09
papaverine	0.02	0.01	0.02	0.13	0.03	0.02	0.00	0.02	0.74
pentazoc	0.03	0.01	0.05	0.15	0.05	0.62	0.08	0.00	0.00
phenacet	0.03	0.02	0.43	0.07	0.08	0.00	0.00	0.19	0.18
phenazon	0.02	0.01	0.04	0.07	0.04	0.25	0.03	0.03	0.52
prazepam	0.01	0.01	0.01	0.02	0.02	0.07	0.02	0.82	0.02
procaine	0.03	0.01	0.54	0.07	0.05	0.22	0.08	0.00	0.00

	eluents								
1	0.82	0.17	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2	0.78	0.03	0.03	0.03	0.13	0.00	0.01	0.00	0.00
3	0.08	0.91	0.00	0.00	0.01	0.00	0.00	0.00	0.00
4	0.03	0.01	0.00	0.00	0.01	0.00	0.90	0.00	0.04
5	0.21	0.70	0.02	0.02	0.05	0.00	0.01	0.00	0.00
6	0.27	0.69	0.01	0.01	0.02	0.00	0.00	0.00	0.00
7	0.01	0.01	0.07	0.87	0.04	0.00	0.00	0.00	0.00
8	0.03	0.02	0.04	0.84	0.03	0.01	0.01	0.00	0.01
9	0.02	0.02	0.71	0.00	0.22	0.01	0.01	0.00	0.01
10	0.25	0.65	0.02	0.02	0.06	0.00	0.00	0.00	0.00
11	0.03	0.03	0.04	0.54	0.34	0.01	0.01	0.00	0.01
12	0.05	0.06	0.04	0.04	0.80	0.00	0.00	0.00	0.00
13	0.43	0.09	0.03	0.02	0.06	0.03	0.32	0.00	0.03
14	0.02	0.01	0.09	0.13	0.73	0.00	0.00	0.00	0.00
15	0.01	0.01	0.00	0.00	0.01	0.01	0.00	0.97	0.00
16	0.20	0.13	0.06	0.04	0.13	0.27	0.04	0.02	0.11
17	0.02	0.02	0.01	0.01	0.01	0.01	0.00	0.00	0.92
18	0.01	0.00	0.00	0.00	0.00	0.86	0.00	0.00	0.11

Table 7. Membership Degrees to the Classes of the Simultaneous Final Fuzzy Partition Obtained Using the FHCCB Algorithm

drugs	111	112	121	122	2
atropine	0.90	0.05	0.00	0.00	0.05
biperide	0.01	0.01	0.03	0.02	0.93
caffeine	0.00	0.00	0.65	0.04	0.31
cocaine	0.00	0.00	0.00	0.00	1.00
codeine	0.00	0.75	0.07	0.18	0.00
cyclizin	0.00	0.00	0.01	0.00	0.98
diazepam	0.00	0.00	0.01	0.01	0.98
ketamine	0.00	0.00	0.01	0.00	0.99
lignocaine	0.00	0.00	0.01	0.01	0.98
lorazerpam	0.00	0.01	0.00	0.78	0.20
mebeverine	0.00	0.00	0.01	0.00	0.99
methadon	0.00	0.00	0.00	0.00	0.99
morphine	0.82	0.03	0.02	0.03	0.10
naloxone	0.00	0.00	0.51	0.01	0.48
papaverine	0.00	0.00	0.01	0.01	0.98
pentazoc	0.00	0.01	0.15	0.03	0.81
phenacet	0.00	0.01	0.10	0.03	0.85
phenazon	0.00	0.00	0.02	0.01	0.97
prazepam	0.01	0.01	0.03	0.02	0.93
procaine	0.00	0.01	0.40	0.03	0.56

	eluents				
1	0.01	0.62	0.03	0.34	0.00
2	0.00	0.03	0.00	0.96	0.01
3	0.01	0.93	0.01	0.04	0.02
4	0.03	0.90	0.01	0.04	0.01
5	0.67	0.00	0.06	0.17	0.09
6	0.00	0.87	0.01	0.11	0.01
7	0.00	0.00	0.00	0.00	0.99
8	0.00	0.00	0.00	0.00	1.00
9	0.00	0.00	0.00	0.00	1.00
10	0.12	0.80	0.01	0.05	0.02
11	0.01	0.11	0.64	0.00	0.25
12	0.00	0.10	0.06	0.83	0.00
13	0.00	0.02	0.03	0.67	0.28
14	0.02	0.11	0.60	0.00	0.27
15	0.00	0.01	0.02	0.01	0.96
16	0.00	0.00	0.00	0.00	1.00
17	0.04	0.30	0.06	0.30	0.30
18	0.06	0.66	0.06	0.23	0.01

information provided by each eluent system and the correlation between such systems have been investigated using the information theory and numerical taxonomy technique. In this context principal components analysis (PCA) has been proven to have a great potential for the evaluation and the selection of eluent systems in TLC.

In this order in their book Meloun et al.[6] discussed the chromatographic behavior of a set of compounds (drugs) with several eluents. In Table 5 are presented the hR_f values obtained using 18 solvent systems. Applying PCA the main conclusions of this study were that drugs 4 (cocaine) and 11 (mebeverine) and drugs 8 and 9 (ketamine and lignocaine)

Table 8. Membership Degrees to the Classes of the Simultaneous Final Fuzzy Partition Obtained Using the FHCCC Algorithm

drugs	1111	1112	112	121	122	211	212	221	222
atropine	0.64	0.00	0.16	0.02	0.01	0.00	0.05	0.04	0.04
biperide	0.01	0.02	0.02	0.05	0.04	0.42	0.21	0.09	0.14
caffeine	0.01	0.03	0.02	0.73	0.14	0.01	0.02	0.02	0.02
cocaine	0.01	0.01	0.01	0.03	0.02	0.03	0.88	0.01	0.02
codeine	0.00	0.79	0.03	0.05	0.03	0.02	0.03	0.02	0.03
cyclizin	0.01	0.02	0.02	0.06	0.03	0.23	0.53	0.03	0.07
diazepam	0.01	0.01	0.01	0.06	0.06	0.02	0.04	0.74	0.03
ketamine	0.00	0.00	0.00	0.01	0.01	0.03	0.07	0.07	0.80
lignocaine	0.00	0.01	0.01	0.03	0.02	0.03	0.05	0.17	0.68
lorazerpam	0.03	0.04	0.06	0.11	0.59	0.03	0.04	0.06	0.05
mebeverine	0.01	0.01	0.01	0.04	0.02	0.22	0.60	0.03	0.07
methadon	0.02	0.03	0.03	0.08	0.05	0.52	0.15	0.04	0.07
morphine	0.00	0.00	0.73	0.05	0.04	0.04	0.05	0.04	0.05
naloxone	0.01	0.01	0.01	0.12	0.65	0.03	0.04	0.06	0.07
papaverine	0.01	0.02	0.02	0.15	0.14	0.04	0.08	0.28	0.26
pentazoc	0.02	0.04	0.03	0.26	0.08	0.11	0.22	0.09	0.15
phenacet	0.01	0.02	0.02	0.15	0.39	0.04	0.07	0.16	0.13
phenazon	0.01	0.01	0.01	0.12	0.07	0.08	0.15	0.18	0.37
prazepam	0.01	0.01	0.02	0.06	0.06	0.07	0.09	0.49	0.19
procaine	0.04	0.07	0.05	0.43	0.10	0.06	0.11	0.06	0.09

	eluents								
1	0.59	0.14	0.01	0.02	0.23	0.00	0.00	0.00	0.00
2	0.02	0.05	0.00	0.00	0.91	0.00	0.00	0.00	0.00
3	0.95	0.00	0.02	0.00	0.01	0.00	0.01	0.00	0.00
4	0.82	0.08	0.04	0.01	0.03	0.00	0.01	0.00	0.00
5	0.00	0.00	0.69	0.06	0.15	0.03	0.03	0.02	0.03
6	0.88	0.00	0.02	0.01	0.08	0.00	0.00	0.00	0.00
7	0.00	0.00	0.00	0.01	0.00	0.03	0.95	0.01	0.01
8	0.00	0.00	0.00	0.00	0.00	0.91	0.03	0.03	0.01
9	0.00	0.00	0.00	0.00	0.00	0.95	0.02	0.02	0.01
10	0.68	0.07	0.19	0.00	0.03	0.01	0.01	0.00	0.01
11	0.01	0.09	0.01	0.60	0.00	0.07	0.11	0.03	0.07
12	0.05	0.01	0.00	0.06	0.87	0.00	0.00	0.00	0.00
13	0.00	0.01	0.00	0.09	0.62	0.05	0.12	0.03	0.08
14	0.01	0.09	0.02	0.54	0.00	0.09	0.14	0.03	0.09
15	0.00	0.01	0.00	0.03	0.01	0.06	0.04	0.84	0.00
16	0.00	0.00	0.00	0.00	0.00	0.05	0.03	0.01	0.91
17	0.02	0.23	0.05	0.07	0.33	0.11	0.04	0.06	0.11
18	0.00	0.73	0.08	0.03	0.14	0.00	0.00	0.00	0.00

are generally not well separated. Also, it was possible to show by plotting scores for the eluents, that some eluents are very similar (e.g., 1, 3, 6, 10, 7, 8, 11, and 12) and some have a large impact on the first three eigenvectors: eluents 2 and 4 on the first; 16, 17 on the second; and 3 and 6 on the third. This information can be used to select a subset of eluents that will give good overall separation at a minimal cost.

By computation of the three cross-classification techniques presented above (Tables 6−8), it is easy to show that the FHCCA algorithm is the best. The results shown in Table 6 illustrate a good agreement with the results obtained by Meloun et al. using PCA. The best solvent systems appear to be 1, 2, 3, 4, 5, 6, 7, 8, 10, 11, 12, 13, 14, and 16, which are very similar, and the poorest solvent systems are 9, 15, 17, and 18. Moreover, to each class of solvent systems is associated the class of drugs unseparated by these solvents.

The method presented herein permits a rationale classification and selection of separating systems in TLC, providing an additional information in the form of membership values.

4. CONCLUDING REMARKS

The output of a fuzzy algorithm includes not only a partition but also additional information in the form of membership values.

Moreover, the fuzzy cross-classification algorithm, with the three variants presented here, produces both a fuzzy objects partition and a fuzzy characteristics partition, "compatible" with the former. We introduced a way to associate each fuzzy set of objects to a fuzzy set of characteristics. We have comparatively analyzed the behavior of the three variants of the hierarchical cross-classification algorithm: FHCCA, FHCCB, and FHCCC algorithms. As we have seen in the sections 3.1 and 3.2, the ASF algorithm produced in this way works better than the original Simultaneous Fuzzy *n*-Means introduced in ref 4 and presented here.

Thus, the advantages of this algorithm (with its three variants) include the ability to observe, on the one hand, the fuzzy classes obtained and the relations between them, and, on the other hand, the characteristics corresponding to each final class of objects (and which have contributed to the separation of the respective class).

The examples analyzed here confirm that the three versions of the Fuzzy Cross Clustering algorithm introduced in this paper, namely FHCCA, FHCCB, and FHCCC approach different aspects in the data, and for this reason all three of them should be used as necessary. The algorithm that is the most performant here is FHCCA, but other research, where other aspects of the data structure are exploited, may indicate that other variants are optimal for those data.

REFERENCES AND NOTES

(1) Bezdek, J. C. *Pattern Recognition with Fuzzy Objective Function Algorithms*; Plenum Press: New York, 1981.
(2) Dumitrescu, D. Hierarchical pattern classification. *Fuzzy Sets and Systems* **1988**, *28*, 145−162.
(3) Dumitrescu, D. *Classification Theory*; University of Cluj-Napoca Press: 1991.
(4) Dumitrescu, D.; Pop, H. F.; Sârbu, C. Fuzzy hierarchical cross-classification of Greek muds. *J. Chem. Inf. Comput. Sci.* **1995**, *35*, 851−857.
(5) Dumitrescu, D.; Sârbu, C.; Pop, H. F. A fuzzy divisive hierarchical clustering algorithm for the optimal choice of set of solvent systems. *Anal. Lett.* **1994**, *27*, 5, 1031−1054.
(6) Meloun, M.; Militky, J.; Forina, M. *Chemometrics for Analytical Chemistry, Vol. I: PC-Aided Statistical Data Analysis*; Ellis Horwood: Chichester, 1992.
(7) Pop, H. F. *Intelligent Systems in Classification Problems*. Ph.D. Thesis, "Babes-Bolyai" University, Faculty of Mathematics and Computer Science, Cluj-Napoca, 1995.
(8) Pop, H. F.; Dumitrescu, D.; Sârbu, C. A study of Roman pottery (terra sigillata) using hierarchical fuzzy clustering. *Anal. Chim. Acta* **1995**, *310*, 269−279.
(9) Pop, H. F.; Sârbu, C. A new fuzzy regression algorithm. *J. Anal. Chem.* **1996**, *68*, 771−778.
(10) Pop, H. F.; Sârbu, C.; Horowitz, O.; Dumitrescu, D. A fuzzy classification of the chemical elements. *J. Chem. Inf. Comput. Sci.* **1996**, *36*, 465−482.
(11) Sârbu, C.; Dumitrescu, D.; Pop, H. F. Selecting and optimally combining the systems of solvents in the thin film cromatography using the fuzzy sets theory. *Rev. Chim.* **1993**, *44*, 5, 450−459.
(12) Sârbu, C.; Horowitz, O.; Pop, H. F. A fuzzy cross-classification of the chemical elements, based both on their physical, chemical and structural features. *J. Chem. Inf. Comput. Sci.* **1996**, *36*, 1098−1108.
(13) Zadeh, L. A. Fuzzy sets. *Inf. Control* **1965**, *8*, 338−353.

CI960148V