Improving the family orientation process in Cuban Special Schools trough Nearest Prototype classification

Villuendas-Rey, Y.[1], Rey-Benguría[2], C., Caballero-Mota, Y.[3], García-Lorenzo, M. M.[4]
[1]*Computer Science Department, University of Ciego de Ávila, Ciego de Ávila, Cuba*
[2]*Educational Research Center, University “Manuel Ascunce”, Ciego de Ávila, Cuba*
[3]*Computer Science Department, University of Camagüey, Camagüey, Cuba*
[4]*Computer Science Department, University of Las Villas, Villa Clara, Cuba*

***Abstract* — Cuban Schools for children with Affective – Behavioral Maladies (SABM) have as goal to accomplish a major change in children behavior, to insert them effectively into society. One of the key elements in this objective is to give an adequate orientation to the children’s families; due to the family is one of the most important educational contexts in which the children will develop their personality. The family orientation process in SABM involves clustering and classification of mixed type data with non-symmetric similarity functions. To improve this process, this paper includes some novel characteristics in clustering and prototype selection. The proposed approach uses a hierarchical clustering based on compact sets, making it suitable for dealing with non-symmetric similarity functions, as well as with mixed and incomplete data. The proposal obtains very good results on the SABM data, and over repository databases.**

***Keywords* — special schools, nearest prototype classifiers, mixed data, non-symmetric similarities**

I. Introduction

In Cuba, the Ministry of Education has special educational schools for dealing with children with singular educational needs. Among them, there are Schools for children with Affective-Behavioral Maladies (SABM).SABM had been designed with the goal of offering a special educational context. In them, the needs of the children that had show maladies in their affective development and/or in their behavior are resolved. Therefore, the children that had have delinquent or anti-social behaviors are bewared in a personalized way in SABMs. The family is the basic cell of society, and in it is the closest educational context for children. When children get out of SABMs, they return to their homes and to their neighborhoods, where they often do not have the correct models to follow. The adequate orientation to the children’s family plays a key role to correct the deficiencies, and to insert effectively these children into society. That is why the personnel in charge of the family orientation process in the SABM of the province of Ciego de Ávila characterize the familiar dynamics of each family, and then proceed to design a personalized strategy for each group of families with similar dynamics.

To give an adequate orientation to the families, the headings of the SABM proceed on two stages: Clustering and Classification. On stage 1, they cluster the families according to their characteristics, and on stage 2, they assign a new arrived family to the group of its closest family, using Nearest Prototype Classification (see figure 1).

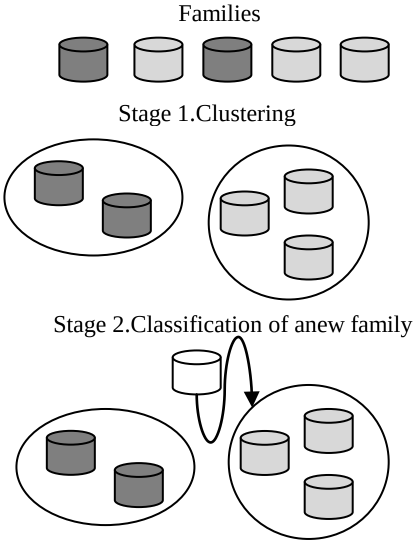

Fig.1. Stages of the Family orientation process at SABM.

Despite the challenges attached to clustering data, there is a need of structuralizing data in SABM School. In this domain, the description of each family has mixed and incomplete attributes. The sociologists associated to SABM selected these attributes to characterize the family dynamics of the SABM families. The data of the families of the SABM School of Ciego de Avila has fourteen attributes (Table I). These attributes measure the attitude of the family to the inclusion of a child in the SABM School, as well as the peculiarities of the family dynamic.

DOI: 10.9781/ijimai.2013.212
]

TABLE I
DESCRIPTION OF THE ATTRIBUTES OF SABM DATA

Att.	Name	Description
1.	impact	If exists impact or shock in the family
2.	attitude	The attitude adopted about the inclusion of a child in the SABM
3.	change	How the family reacts to the change, if they oppose (O), they resist (R), they have resignation (G) or they agree (A)
4.	guilty	If there are or there are not guilty feelings in the family
5.	clime	The kind of emotional clime, if it is positive or negative
6.	communication	The kind of communication that prevails in the family
7.	handling	The way the family handles the fact of including a child into the SABM
8.	relations	The way the interpersonal relations are developed into the family
9.	crisis	The kind of emotional crisis, by demoralization, disarranging, frustration, impotence or no crisis
10.	estimation	The way the self estimation of the family is
11.	consciousness	If there is or not consciousness of the reality
12.	linkage	If there is or not a favorable link with the SABM
13.	hopes	The hopes the family has to the future
14.	time	The time (in months) the child is at the SABM

To compare in effective way two families, and to decide whether the families have similar dynamics, it was needed to work together with the family orientation experts and the sociologists associated to SABM in Ciego de Ávila. After analyzing several similarity functions proposed in the literature for dealing with mixed and incomplete data, the experts decided that those similarities were not adequate for comparing SABM data.

It was decided then to design a personalized similarity function to deal with the peculiarities of SABM data.The sociologists and the family orientation experts of SABM decide that classical comparison criteria for nominal attributes were adequate to compare the nominal features of SABM data, but the 3rdattribute, “change”.

To compare the values of the 3rd attribute, it was needed to establish a non-symmetric comparison matrix as feature comparison criterion, due to the semantics of the different values of this attribute (table 2). For the numerical attribute, “time”, the selected comparison criterion was normalized difference.

From analysis with different expert and sociologist associated to SABM, a similarity function to compare the families is designed. It is a non-symmetric similarity, due to the non-symmetric comparison matrix for the 3rd attribute, change. Let be two families, f_i and f_j, and $f_i[k]$ the value of the k-th attribute (A_k) in the f_i family. The similarity for comparing SABM data is defined by:

$$S(f_i, f_j) = 1 - D(f_i, f_j) \tag{1}$$

where $D(f_i, f_j) = \sum_{k=1}^{14} D_k(f_i, f_j)$. For nominal attributes but 3rd attribute, the function $D_k(f_i, f_j)$ is as follows:

$$D_k(f_i, f_j) = \begin{cases} 0 & if \quad f_i[k] = f_j[k] \\ 1 & if \quad f_i[k] \neq f_j[k] \\ 0.5 & if \quad f_i[k] = "?" \vee f_j[k] = "?" \end{cases} \tag{2}$$

On the other hand, for the numerical attribute, the function $D_k(f_i, f_j)$ is as follows:

$$D_k(f_i, f_j) = \frac{|f_i[k] - f_j[k]|}{\max(A_k) - \min(A_k)} \tag{3}$$

In the case of the third attribute, “change”, the different attribute values have a peculiar meaning. Due to, their similarity depends of each value combination. This attribute defines the attitude the family adopts to face the fact that one of the family members, a child, will be allocate into the SABM.

Table II shows the comparison matrix of values for the attribute “change”. As shown, the dissimilarity between values “Resistance” (R) and “Resignation” (G) differ from “Resignation” to “Resistance”.

TABLE II
COMPARISON MATRIX OF THE VALUES FOR THE ATTRIBUTE “CHANGE”

Value	*O*	*R*	*G*	*A*
O	0	0.2	0.8	1
R	0.2	0	**0.4**	0.8
G	0.8	**0.8**	0	0.4
A	1	0.8	0.4	0

Each cell shows the dissimilarity values of the pair (row vs. column). In bold the non-symmetric values

The rest of the paper is as follows: section II introduces the proposed hierarchical clustering, based on Compact Sets structuralizations, and the proposed Nearest Prototype selection algorithm. Section III addresses the selection of the adequate cluster number for the families in SABM, to improve the family orientation process. Sections IV and V review some previous works on clustering mixed data and nearest prototype selection for mixed data, respectively. Section VI offers the numerical experiments comparing the proposals with respect other clustering and prototype selection algorithms, over SABM data and repository data. The paper ends with the conclusions and future works.

II. CLUSTERING AND NEAREST PROTOTYPE SELECTION BASED ON COMPACT SETS

A. Hierarchical clustering based on Compact Sets

Taking into consideration the nature of the problem of clustering and classifying SAMB data, described by mixed and incomplete features, and with a non-symmetric similarity function used to compare the families; it is necessary to develop a novel clustering algorithm able to deal with all these restrictions simultaneously. This section introduces a hierarchical clustering algorithm based on Compact Sets, to deal with SABM data.

Compact Sets structuralization was described in [1], and this structuralization is based on the concept of Maximum Similarity Graphs. Maximum Similarity Graphs (MSG), are directed graphs such that each instance $x \in X$ is connected to its most similar instance. A connected component of a MSG is a Compact Set (CS).

Formally, let be $G = (X, \theta)$ a MSG for a set of objects X, with arcs θ. In this graph, two objects $x_i, x_j \in X$ form an arc $(x_i, x_j) \in \theta$ if $\max_{x \in X}\{\mathrm{sim}(x_i, x)\} = \mathrm{sim}(x_i, x_j)$, where $\mathrm{sim}(x_i, x_j)$ is a similarity function. Usually $\mathrm{sim}(x_i, x_j) = 1 - \Delta(x_i, x_j)$ and $\Delta(x_i, x_j)$ is a dissimilarity function. In case of ties, the Maximum Similarity Graph establishes a connection between the object and each of its nearest neighbors. As mentioned before, Compact Sets are the connected components of such graph.

Formally, a subset $N \neq \emptyset$ of X is a Compact Set if and only if [1]:

$$a)\forall x_j \in X \left[x_i \in N \wedge \left(\begin{array}{l} \max\limits_{\substack{x_i \in X \\ x_i \neq o_j}} \{\mathrm{sim}(x_i, x_j)\} = \mathrm{sim}(x_i, x_j) \\ \vee \max\limits_{\substack{x_i \in X \\ x_i \neq x_j}} \{\mathrm{sim}(x_j, x_i)\} = \mathrm{sim}(x_j, x_i) \end{array} \right) \right] \Rightarrow x_j \in N$$

$$x_i = x_{i_1} \wedge j$$

(4)

$$b)\forall x_i, x_j \in N, \exists x_{i_1}, \cdots, x_{i_q} \in N \; \max$$

$c)$ Every isolated object is a Compact Set, degenerated.

All the instances connected between them belong to the same CS, such that the nearest neighbor of each instance is also in the same CS (figure 2). The proposed method follows a hierarchical agglomerative approach to clustering, but merging CSs instead of objects.

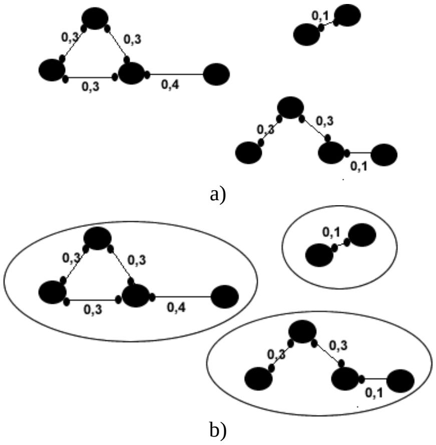

Fig. 2. a) Maximum Similarity Graph of instances and b) Compact Sets of instances.

As many other hierarchical agglomerative clustering algorithms [2], the proposed Compact Set Clustering (CSC) uses a multilayer clustering to produce the hierarchy. The algorithm (figure 3) starts by computing the Maximum Similarity Graph from dataset. Second, it is defined as initial groups each CS in the MSG. Then it merges the groups, until having the desired number of clusters. The merging is making with all possible groups that are more similar in a single step and it is avoided order dependence.

CSC algorithm uses the similarity between cluster representatives as inter group similarity function. Let be x and y the representatives of clusters Ci and Cj, respectively, and $S(x, y)$ is the similarity between those representatives. The similarity function between those clusters is:

$$sim(C_i, C_j) = S(x, y) \quad (5)$$

The instances that maximize the overall inter-group similarity correspond to the representatives of the clusters. Formally, the representative instance r of a group Cj will be:

$$r = arg\ \max_{x}\{S(x, y)\} \quad (6)$$

```
Compact Sets Clustering (CSC)
Inputs:    k: number of groups
           S: inter objects similarity function
           T: training set
Output:    C: resulted clustering
1. C = ϕ
2. Create a Maximum Similarity graph of the objects in T
     using the similarity function S
3. Add to C each connected component of the graph created at
     step 1
     3.1. Select the cluster representative instance as in (6)
4. While |C| < k
     4.1. Merge all more similar groups, using (5)
     4.2. Recalculate cluster representative instance
5. Return C
```

Fig. 3. Compact Set Clustering (CSC) algorithm.

Thus, the CSC algorithm selects real objects to represent the clusters, avoiding the construction of artificial centroids. This approach obtains compact and separated clusters, and it is able to detect the true partitions of data.

B. Nearest Prototype selection based on Compact Sets

In the classification phase of the SABM data, each new family must be compared to every family already in SABM, using the Nearest Neighbor (NN) classifier [3]. Despite the NN classifier is one of the most popular supervised classification methods in Pattern Recognition, it suffers from important drawbacks. NN has high storage and computational requirements, because it storages the entire training set, requiring large space. In addition, to determine the class of a new object, NN needs to compare it with every object in the training set. Another drawback of NN is its sensitivity to noisy and outlier objects.

To overcome these drawbacks, researchers have proposed the Nearest Prototype (NP) classification. NP classification use prototype selection methods to obtain a reduced set of representative objects (prototypes) as training data for classification.As NP classification has been extensively used for supervised classification with very good results [4] , [5], it was decided that the classification stage of SABM data was carried out using NP classification.

As stated before, the SAMB data is described by mixed and incomplete features, and it also uses a non-symmetric similarity function to compare the families; so, it is necessary to develop a novel prototype selection algorithm able to deal with all these restrictions simultaneously. This section introduces a prototype selection algorithm (figure 4) based on Compact Sets structuralization [6].

The proposed Prototype Selection (PS) algorithm allows deciding the desired amount of prototypes for the Nearest Prototype classification. It is also able to deal with arbitrarily similarity functions; due to the similarity to compare objects is a parameter of the algorithm.

```
Prototype Selection algorithm
Inputs:   k: desired number of prototypes
          S: inter objects similarity function
          T: training set
Output:   P: prototype set
1. P = ϕ
2. C = CSC(k, S, T)
3. For each cluster $C_i \in C$
      5.1.  Select the cluster representative as in (6)
      3.1.  Add to P the cluster representative
4. Return P
```

Fig. 4. Prototype Selection (PS) algorithm.

The PS algorithm starts with an empty prototype set. Then, it structuralizes the training set *T* using the Compact Sets Clustering (CSC) method, finding as many clusters as desired prototypes. Then, the PS algorithm will select the representing object of each cluster, and will add it to the prototype set.

The PS algorithm proposed includes several novel characteristics, differentiating it from previous prototype selection algorithms. It structuralizes data using a hierarchical clustering algorithm based on Compact Set structuralization. It also uses a data-dependant similarity function, which makes it applicable to several domains with non-metric similarities, such as social sciences and medicine. It also selects representing objects of clusters as prototypes instead of constructing artificial objects for the Nearest Prototype classification stage.

III. Finding the Adequate Clustering for SABM Data

As mentioned before,the data of the families of the SABM School of Ciego de Avila is described by mixed attributes that measure the attitude of the family to the inclusion of a child in the SABM, as well as the peculiarities of the family dynamic. It is also used a non-symmetric similarity (1) to compare family descriptions.

The first stage of the family orientation process is to cluster the families of the SABM. As no predefined number of clusters exists, it is needed to obtain several candidates clustering, and then select the one that best fits data. Internal cluster validity indexes allow comparing several candidate clustering, and deciding which of them best fits data. To determine the adequate cluster number of SABM data, it was clustered with cluster number varying from two to nine clusters, and then it were used internal cluster validity indexesto select the partition that best fits data. Among unsupervised cluster validity indexes, the Dunn's index measure how compact and well separated the clusters are. Let be d(Ci,Cj) the dissimilarity between clusters, and Δ(Ci) the cluster size, the Dunn's validation index is the ratio between the minimum dissimilarity between two clusters and the size of the largest cluster.

$$D = \frac{\min_{i=1..nc, j=1..nc, i\neq j}\{d(C_i, C_j)\}}{\{\Delta(C_i)\}} \qquad (7)$$

Where $d(C_i, C_j)$ is the dissimilarity between clusters, and $\Delta(C_i)$ is the cluster size.

Dunn's index was used with complete – linkage as dissimilarity measure and with single – linkage as cluster size measure. In figure 5, there are shown the results the Dunn's index with cluster number varying from two to nine clusters. The best partition has seven clusters.

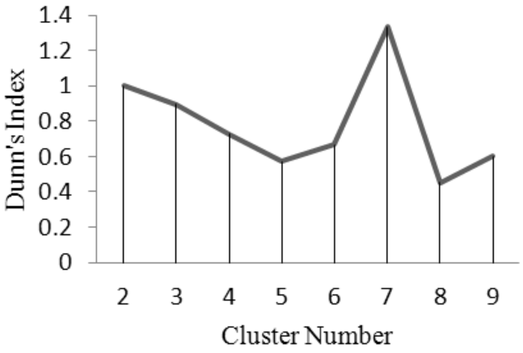

Fig.5. Values of the Dunn's index obtained by CSC using different cluster number.

In addition, it was also used the Silhouette index [7]. The Silhouette is the average, over all clusters, of the Silhouette width of their points.

If x is an object in the cluster c_i and n_i is the number of objects in c_i, then the Silhouette width of x is defined by the ratio:

$$S(x) = \frac{b(x) - a(x)}{\max\{a(x), b(x)\}} \tag{8}$$

where a(x) is the average dissimilarity between x and all other objects in c_i, and b(x) is the minimum of the average dissimilarities between x and the objects in the other clusters.

$$a(x) = \frac{1}{n_i - 1}\sum\nolimits_{y \in C_i, y \neq x} d(x, y) \tag{9}$$

$$b(x) = \min_{h=1..k, h \neq i}\left\{\frac{1}{n_h}\sum\nolimits_{y \in C_h} d(x, y)\right\} \tag{10}$$

Finally, the global Silhouette is as follows:

$$S(C) = \frac{1}{k}\sum\nolimits_{i=1}^{k}\frac{1}{n_i}\sum\nolimits_{x \in C_i} S(x) \tag{11}$$

For a given object x, its Silhouette width ranges from −1 to 1. If the value is close to –1, then it means that the object is more similar, on average, to another cluster than the one to which it belongs. If the value is close to 1, then it means that its average dissimilarity to its own cluster is significantly smaller than to any other cluster. The higher the Silhouette, the more compact and separated are the clusters.

In figure 6 it is shown the results of the Silhouette index, with cluster number varying from two to nine clusters. The best partition also had seven clusters.

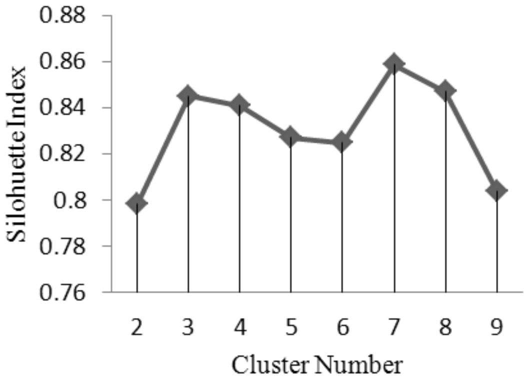

Fig.6. Values of the Silhouette index obtained by CSC using different cluster number.

According to both Dunn’s and Silhouette indexes, the structuralization that best fits the SABM data is the one with seven clusters. This structuralization will be used later in the classification stage of the family orientation process.

For the classification stage, each instance had as class label the number of the cluster it belongs. By this, the resulted clustered families of stage one, will constitute the training matrix for the supervised classifier.

IV. Previous Works on Clustering Mixed Data

It is impossible to address clustering techniques without referring to the k-means algorithm. The k-means algorithm is one of the oldest clustering techniques, and it has a proved efficiency to find compact and well separated clusters. At the first step, k-means randomly select a set of cluster centers from data. Then, it assigns each object to its closest center, using the Euclidean distance. After that, the algorithm iterates until no change is made on cluster centers. In the iterative process, it computes the new cluster centers, as the mean of all objects in the cluster, and reassigns every object to its closest center.

Several authors have proposed modifications to this simple, yet powerful technique, to handle mixed and incomplete data. All of them include a redefinition of the distance function, as well as the cluster centers.

In 1997, Huang proposed the k-prototypes (KP) algorithm [8]. The KP algorithm redefines cluster center as the mean of the numerical attributes, and the mode of the nominal attributes. Also, it uses as dissimilarity function, with weights $\omega=\{\omega_1, \ldots, \omega_d\}$ of each attribute. Although the KP algorithm deals with mixed type attributes, it does not handle missing data.

In 2007, Ahmad and Dey proposed another modification of the classical k-means algorithm [9]. They redefined the dissimilarity function. The proposed dissimilarity includes attribute weights. For categorical attributes, the dissimilarity takes into account the co-occurrences of each value pair, and then set as more similar the low frequency values pairs.

Ahmad and Dey [9] also redefine the cluster center. In their definition, the center consists on a cluster description. The description includes the mean of each numerical attribute, and a set of pairs (value, count) for each categorical attribute. Each pair has the attribute value and the count of objects in a cluster that have this value.

In 2011, the same authors [10] proposed a modification of the algorithm proposed previously in 2007. They do not give in the paper any name for the new method, so this paper refers to it as AD2011 (Ahmad and Dey proposal of 2011). The new method discretizes numeric attributes before the clustering process, using the Equal Width Discretization procedure. It also includes in the dissimilarity function the contribution λ of each attribute to the cluster.

The AD2011 algorithm includes two user-defined parameters. The first is the γ parameter, included in the attribute contribution computation, having a suggested value of 20, and the second is the S parameter, included in the discretization procedure of numeric attributes, having a suggested value of 5.

Among the main drawbacks of k-meansbased clustering are that the algorithms depend on the definition of cluster centers. They are also unable to form arbitrary shapes clusters.

Another family of clustering algorithms is the family of hierarchical algorithms. Hierarchical clustering algorithms create a decomposition of the objects by forming a binary tree called dendogram. All objects are at the root, at the intermediate nodes are groups of objects, and at leafs are single objects. The tree is usually created top down (divisive algorithms) or bottom up (agglomerative algorithms). In the last, each object is considered as a group, and at each step the two more similar groups are joined. The stopping condition is usual that all objects are in the same group, or the desired number of groups is reached. These methods are referred as Hierarchical Agglomerative Clustering (HAC). A HAC method for dealing with mixed and incomplete data is the HIMIC algorithm [11].

Other kind of clustering algorithms are model – based clustering. In these methods, a model or metaheuristic is used to evolve clusters. Each candidate clustering is a solution, having certain optimization value (cluster quality). The model or heuristic iterates, until it finds the desired clustering. Among model based clustering are the Genetic Algorithm cluster based AGKA, proposed by Roy and Sharma in 2010 [12] and the Flocking based method proposed in [13].

The AGKA algorithm is based on Genetic algorithms. Genetic Algorithms (GA) are one of the most used techniques in Artificial Intelligence, Pattern Recognition, and Data Mining. They offer a feasible solution for a huge number of optimization and classification problems.

The AKGA method uses a genetic procedure, and includes the dissimilarity proposed by Ahmad and Dey in 2007 [9] in the fitness function.

AGKA codifies each candidate clustering as an individual, using an integer array of length equal to the object count. Each position (gene) of the array indicates the cluster assigned to the object in that position. It has a mutation strategy that changes an object to its most probable cluster, offering a quickly convergence. It also has an elitist survival strategy.

Model based clustering can be applied in a huge number of situations, and they have numerous variants according to the parameters, evolution strategies, solution generation, and others. However, the algorithms belonging to this approach are often stochastic, and the quality of the resulted clustering depends on the parameter setting and internal evolution strategies used by a particular model.

V. PREVIOUS WORKS ON MIXED DATA PROTOTYPE SELECTION

As one of the main drawbacks of NN classifiers is its sensitivity to noisy and mislabeled objects (section II), there is a research interest in the Artificial Intelligence and Pattern Recognition community to overcome this difficulty [14], [15].

The algorithms to obtain a prototype set for the NN classifier are divided into prototype selection methods and prototype generation methods. This work is focused on prototype selection methods; due to these methods obtain a subset of the training matrix.

Prototype selection methods are divided into condensing algorithms, editing algorithm and hybrid algorithms [15]. Condensing algorithms aim at reducing the NN computational cost by obtaining a small subset of the training matrix, maintaining the accuracy as high as possible, while editing algorithms aim at improve classifier accuracy by deleting noisy and mislabeled objects. Hybrid methods usually combine both condensing and editing strategies in the selection procedure.

The first editing algorithm is the Edited Nearest Neighbor (ENN), proposed by Wilson in 1972 [15]. The ENN algorithm deletes the objects misclassified by a k-NN classifier, where k is a user-defined parameter, usually $k = 3$.

Another classical editing method is MULTIEDIT, proposed by Devijver and Kittler in 1980 [17]. MULTIEDIT works as follows: first, it divides the training matrix in ns partitions, in each partition it applies the ENN method, using a 1-NN classifier trained with the next partition. The last partition is trained with the first one. After each iteration, it joins the remaining objects in each partition and repeats the process until no change is achieved in successive iterations.

In 2000, Hattori and Takahashi [18] proposed a new editing method, referred in this paper as NENN. The method computes the k neighbors of each object, including all objects that have the same dissimilarity value of the last k neighbor. If at least one of the neighbors it is not of the same class of the object, it deletes the object of the training matrix.

In 2002, Toussaint used proximity graphs to obtain a reduced prototype set [19].

Caballero *et al*. introduced other editing algorithms in 2007, the EditRS1 and EditRS2 methods [20]. They used elements of the Rough Set Theory to obtain lower and upper approximations of the training matrix, and to compute the limit regions of each class. Both methods use a reduct as base of the editing process.

Condensing methods were proposed first by Hart in 1968 with the Condensed Nearest Neighbor (CNN) algorithm [21]. In this work, he introduced the concept of consistent subset, a subset of the training matrix such as training a NN classifier with this subset, every instance in the original training matrix is correctly classified.

The Reduced Nearest Neighbor (RNN) consists on a post processing of the CNN algorithm. After computing CNN, RNN tries to delete every object, if the deletion does not introduce any inconsistency. Gates [22] demonstrated that if a minimum consistent subset is a subset of the CNN result, the RNN methods always find it.

Another modification to classic CNN is the Generalized Condensed Nearest Neighbor (GCNN) method. It was proposed by Chou *et al*. in 2006 [23]. The GCNN treats CNN as a particular case, and includes a set of rules to “absorb” prototypes.

Other condensation method is the PSR, introduced by Olvera-López at al. in [24], which selects the prototype set based on prototype relevance. More recently, García-Borroto *et al*. proposed the CSESupport method [25]. It deletes the less important objects, guaranteeing the consistency of the subset by a mark strategy.

The mark strategy consists on the following: when deleting an object, it marks every object that supports it (in a Support Graph), and at least one of them must be included in the condensed subset. A support graph is a directed graph, such as it connects each object all objects of its same class closer than the NUN object [25].

The NUN (Nearest Unlike Neighbor) is the object of different class closest to *x* [26].In this strategy, when an object is the last with a mark, it is included in the result, same if an object does not have any outward edges in the graph.

The method initiates with all training matrix as a consistent subset, and at each iteration deletes the less important objects. It also updates the objects NUN, and builds the support graph with every object in the training matrix, to maintain the subset consistency [25].

CSESupport method handles missing and incomplete data, as well as asymmetric and non-symmetric dissimilarities. However, it does not allow defining the desired number of prototypes.

VI. Numerical Experiments

Numerical experiments were carried out using nine mixed and incomplete databases of the Machine Learning repository of the University of California at Irvine (UCI) [27].

TABLE III
Description of databases used in numerical experiments

Databases	Nominal Attributes	Numerical Attributes	Classes
autos	10	16	6
colic	15	7	2
dermatology	1	33	6
heart-c	7	6	5
hepatitis	13	6	2
labor	6	8	2
lymph	15	3	4
tae	2	3	3

The first experiment was to compare the performance of state of the art clustering algorithms with CSC, over the SABM data and over repository data, and the second experiment was to compare the performance of the proposed prototype selection procedure with respect to other prototype.

A. Numerical experiments on clustering mixed data

The family orientation process on SABM involves both clustering and Nearest Prototype classification. It was decided to consider both internal and external cluster validity indexes to compare the performance of the proposed CSC algorithm with respect to AD2011 [10] and AGKA [12] algorithms over the SABM data. Both AD2011 and AGKA had a predefined dissimilarity, and the CSC algorithm used the similarity function designed for SABM data (section I). It were selected both Dunn's index and the Silhouette index for internal clustering validation and Entropy and Cluster Error indexes for external clustering validation. The amount of clusters to obtain by each algorithm in SABM data was defined to be equal to seven. It was because seven clusters was the best partition of SABM data (section III). The results of the compared algorithms over the SABM data are shown in Table IV.

TABLE IV
Results of the clustering algorithms over SABM data

Algorithms	Internal indexes		External indexes	
	Dunn's index	*Silhouette index*	*Cluster Error*	*Entropy*
AD2011	0.0077	-0.1427	0.6364	2.5331
AGKA	0.1968	-0.2850	0.5686	1.7730
CSC	**1.3333**	**0.8585**	**0**	**0**

External evaluation measures for clustering can be applied when class labels for each data object in some evaluation set can be determined *a priori*. The clustering task is then used to assign these data points to any number of clusters. In each cluster must be all and only those data objects that are members of the same class [28].To compare the clustering results produced by the different algorithms, it is used the Cluster Error andthe Entropy measure.

Cluster Error [9] consists on counting the amount of objects not belonging to the majority class of each cluster. Let be *C* the resulted clustering, C_ia cluster in *C*, and n_i the number of object belonging to the majority class in the i-th cluster. The Cluster Error of C with respect to class labels is given by:

$$CE(C) = \sum_i \frac{|C_i| - n_i}{|C_i|} \tag{12}$$

Lower values of Cluster Error indicate a high performance of the algorithms.

The Entropy index, as described in [29], measures the dispersion of the classes in the clusters. Low Entropy indicates high similarity of clusters and classes. Let be C the resulted clustering, ci the i-th cluster in C, n_i^j the number of object of the j-th class in the i-th cluster and N the amount of objects. The Entropy of C with respect to class labels is given by:

$$E(C) = -\sum_i \frac{|c_i|}{N} * \sum_j \frac{n_i^j}{|c_i|} \log\left(\frac{n_i^j}{|c_i|}\right) \tag{13}$$

To compare the results of the selected clustering algorithms with respect to the proposed CSC over repository data, both Cluster Error and Entropy external indexes were selected. It was used as cluster count for each algorithm the amount of class each database has.

The CSC algorithm was applied to repository data using the HOEM dissimilarity function proposed by Wilson and Martinez [30]. The results of Cluster Error and Entropy over repository data are shown in table V and figure 7, and in table VI and figure 8, respectively. Then, to establish if the differences in performance were significant or not, the Wilcoxon test was applied.

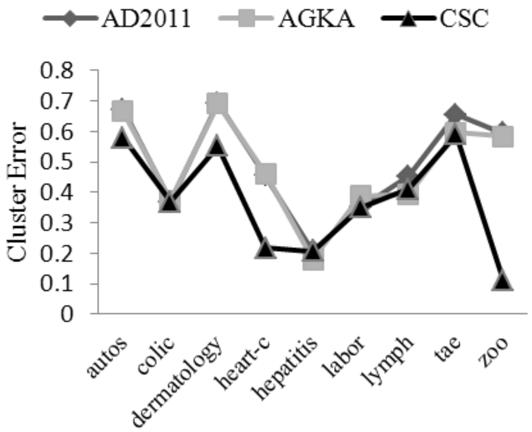

Fig.7. Results of the methods over UCI databases according to Cluster Error.

TABLE V
RESULTS OF THE METHODS ACCORDING TO CLUSTER ERROR

Databases	**AD2011**	**AGKA**	**CSC**
autos	0.6731	0.6650	0.**5804**
colic	**0.3695**	0.3724	**0.3695**
dermatology	0.6939	0.6910	**0.5519**
heart-c	0.4554	0.4615	**0.2178**
hepatitis	0.2064	**0.1803**	0.2064
labor	**0.3508**	0.3880	**0.3508**
lymph	0.4527	**0.3933**	0.4121
tae	0.6556	0.6158	**0.5894**
zoo	0.5940	0.5841	**0.1089**
Times Best	**2**	**2**	**7**

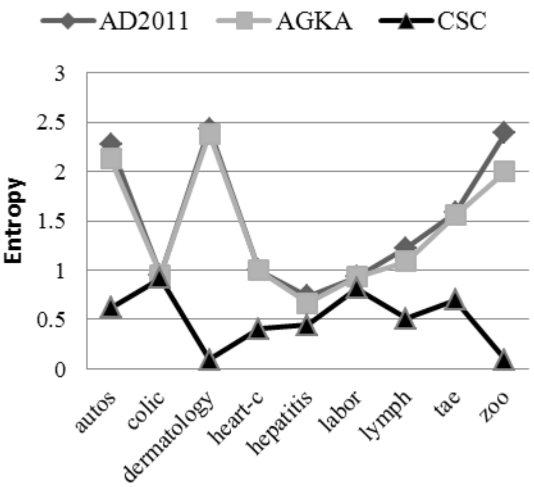

Fig.8. Results of the Entropy of the methods over the UCI databases.

TABLE VI
RESULTS OF THE METHODS ACCORDING TO ENTROPY

Databases	**AD2011**	**AGKA**	**CSC**
autos	2.2725	2.1314	**0.6198**
colic	0.9503	0.9525	**0.9205**
dermatology	2.4326	2.3793	**0.0947**
heart-c	0.9943	0.9956	**0.4063**
hepatitis	0.7346	0.6663	**0.4424**
labor	0.9348	0.9311	**0.8155**
lymph	1.2277	1.0914	**0.5120**
tae	1.5845	1.5593	**0.6985**
zoo	2.3906	1.9988	**0.0909**
Times Best	**0**	**0**	**9**

The Wilcoxon test (table VII) helps determining if the CSC significantly outperforms the other algorithms according to Cluster Error and Entropy. It is define the null hypothesis as no differences in performance, and the alternative hypothesis as the proposed method outperforms the other method. It is used an alpha value of 0.05, with a 95% confidence level.

TABLE VII
RESULTS OF THE WILCOXON TEST FOR PAIR WISE CLUSTERING COMPARISON

Our method	**Asymptotical Significance**	
	According to Cluster Error	*According to Entropy*
vs. AD2011	**0.028**	**0.008**
vs. AGKA	**0.036**	**0.008**

The proposed method has a significant better performance than the AD2011 and AGKA methods. This may be due to it uses a similarity function data dependant, which makes it applicable to several domains with non-metric similarities, such as social sciences and medicine. It also selects a cluster representative instead of constructing fictional cluster centers, guaranteeing a real object represents each cluster. Therefore, the proposed algorithm is able to detect the true partitions of data and to handle mixed and incomplete databases.

B. Numerical experiments on Nearest Prototype classification

This section offers the results of comparing the performance of the proposed Prototype Selection (PS) approach with some other prototype selection algorithms for mixed data [18], [23], [24], [25] and with the original classifier (ONN), using all objects.

The proposed PS method was applied to SABM data with cluster count equal to seven (selecting one prototype per class), and it was applied over repository data with cluster count equal to 50, so 50 prototypes were selected from each database, one for each cluster. PS used the HOEM proposed by Wilson and Martinez [30] as dissimilarity function for repository data.

The Classifier Error measure was used to compare the performance of the algorithms. Classifier Error (CE) is calculated as the ratio between the amount of misclassified objects and the amount of instances in the original training set. Let be $\alpha(x)$ the true class off the object x, and $NN(x)$ the class assigned to x by the Nearest Neighbor classifier. The Classifier Error is given by:

$$CE = \frac{|x \in T: \alpha(x) \neq NN(x)|}{|\text{Training set}|} \tag{14}$$

Another quality measure of prototype selection methods is Retention Rate. Retention Rate (RR) is calculated as the ratio between the amount of selected prototypes and the amount of instances in the original training set.

$$RR = \frac{|\text{Prototype set}|}{|\text{Training set}|} \tag{15}$$

The 10 fold cross validation procedure facilitates testing the performance of the Prototype Selection stage. On SABM data (table VIII), the classifier trained with the whole data obtained zero testing error, despite the use of a non-symmetric similarity. In addition, several prototype selection methods were able to classify correctly every instance in the testing sets, having zero error too. The PRS method deletes the entire dataset, whereas the GCNN method does not achieve any data reduction.

TABLE VIII
CLASSIFIER ERROR AND RETENTION RATES OF PROTOTYPE SELECTION METHODS OVER SABM DATA

Algorithm	Classifier Error	Retention Rates
CSESupport (CSES)	**0**	**0.1812**
GCNN	**0**	1
NENN	0.395	0.8421
PRS	-	0*
PS	**0**	**0.1812**
ONN	**0**	1

* The PRS method deletes the entire database.

The Classifier Error and Retention Rate results of the methods over repository data are shown in tables IX and X, respectively. Figures 9 and 10 also show these results.

TABLE IX
CLASSIFIER ERROR OF PROTOTYPE SELECTION METHODS OVER REPOSITORY DATA

Databases	CSES	GCNN	NENN	PRS	PS	ONN
autos	0.3026	0.3023	0.6054	0.3311	0.3450	**0.2926**
colic	0.2310	*0.1956*	**0.1819**	0.2176	0.3012	0.2064
dermatology	0.1172	0.0681	**0.0572**	0.0873	**0.0572**	0.0599
heart-c	0.2576	0.2282	**0.1621**	0.2312	*0.2213*	0.2282
hepatitis	0.2325	0.1875	0.2079	0.2325	0.1937	**0.1741**
labor	**0.1000**	0.1566	0.1700	0.2066	*0.1233*	0.1400
lymph	0.2361	0.2033	0.2433	0.2461	0.2033	**0.1823**
tae	0.3801	0.3841	0.7554	0.5691	0.5366	**0.3641**
zoo	*0.0300*	*0.0300*	0.1081	0.0490	0.0600	0.0400
Times better than ONN	**2**	**2**	**3**	**0**	**3**	

Error lower than original classifier in italics and sub-rayed, and best results in bold.

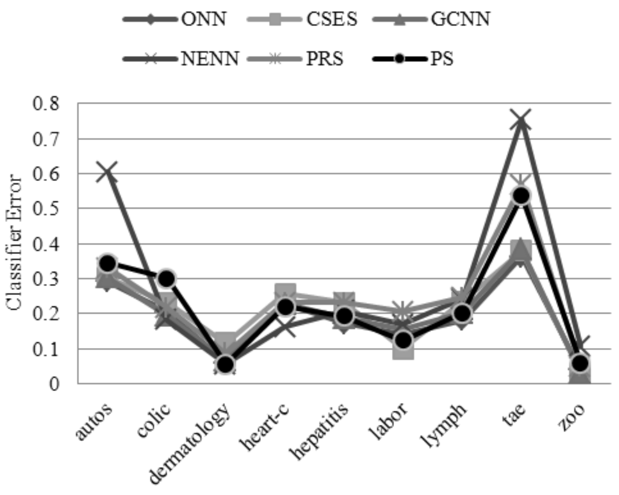

Fig.9. Results of Classifier Error of the methods over the UCI databases

The proposal was able to outperform classifier accuracy in three databases, as well as NENN, and does not have a significant increase of classifier error in the remaining databases.

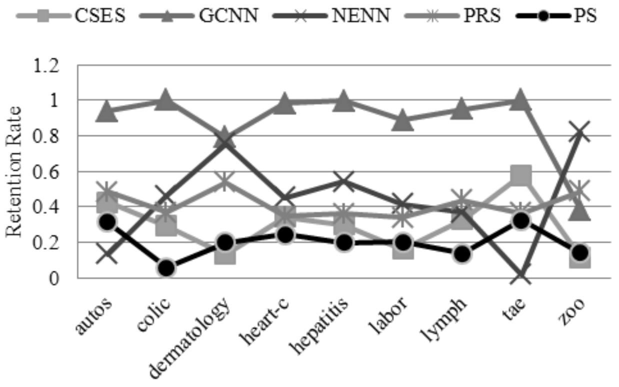

Fig.10. Results of Retention Rate of the methods over the UCI databases.

The proposal gets the lower object retention rates in four databases, and keeps it lower than 35% in the remaining. These results are due to the selected amount of prototypes, established to be 50.

TABLE X
RETENTION RATE OF PROTOTYPE SELECTION METHODS OVER REPOSITORY DATA

Databases	CSES	GCNN	NENN	PRS	PS
autos	0.4277	0.9393	**0.1328**	0.4856	0.3155
colic	0.2936	1.0000	0.4616	0.3702	**0.0590**
dermatology	**0.1345**	0.7905	0.7562	0.5395	0.1982
heart-c	0.3447	0.9817	0.4518	0.3487	**0.2472**
hepatitis	0.3011	0.9971	0.5426	0.3606	**0.2000**
labor	**0.1638**	0.8887	0.4174	0.3392	0.2046
lymph	0.3304	0.9504	0.3686	0.4369	**0.1381**
tae	0.5798	1.0000	**0.0184**	0.3642	0.3252
zoo	**0.1166**	0.3851	0.8196	0.4873	0.1430
Times Best	**3**	**0**	**2**	**0**	**4**

Best results are shown in bold.

Although the above results are very promising, again the Wilcoxon test (table XI) was used to establish the differences between the proposed approach and other algorithms, according to classifier error and object retention rates. Again, it is define the null hypothesis as no differences in performance, and the alternative hypothesis as the proposed method outperforms the other method. It is used an alpha value of 0.05, with a 95% confidence level.

TABLE XI
RESULTS OF WILCOXON TEST FOR PAIR WISE COMPARISON OF PROTOTYPE SELECTION METHODS OVER REPOSITORY DATA

Asymptotical	Our method vs.				
Significance	*CSES*	*GCNN*	*NENN*	*PRS*	*ONN*
Classifier Error	0.678	0.263	0.327	0.314	0.051
Retention Rate	0.051	**0.008**	0.051	**0.008**	**0.008**

According to classifier error, the proposed Prototype Selection (PS) ties with other prototype selection algorithms, and with the original classifier. In addition, this approach has a significant better performance than two other methods according to object retention rates, according to a 95% of confidence. These results reflect that the proposed method is able to maintain classifier accuracy, using only a reduced number of prototypes. In addition, the nature of the PS algorithm makes it suitable for dealing with quantitative and qualitative features, absences of information and non-symmetric dissimilarity functions.

VII. CONCLUSION

A conclusion might elaborate on the importance of the work or suggest applications and extensions. In Cuban special schools, the family orientation process has two stages: family clustering and family classification. This paper proposed a novel method for clustering and Nearest Prototype Classification. The proposed approach has its bases on hierarchical compact sets and handles mixed type data as well as non-symmetric similarity functions. It is compared the performance of the proposal with respect to existing clustering and prototype selection algorithms over repository and real Cuban special schools data. The proposal successfully clusters and classifies the families of children in Cuban special schools. This leads to a better orientation process, spending less time to correct the children deficiencies.

REFERENCES

[1] J. Ruiz-Shulcloper and M. A. Abidi, "Logical combinatorial pattern recognition: A Review," in *Recent Research Developments in Pattern Recognition*, S. G. Pandalai, Ed. USA: Transword Research Networks, 2002, pp. 133-176.

[2] A. K. Jain and R. C. Dubes, *Algorithms for clustering data*. New Jersey, USA: Prentice Hall, 1988.

[3] T. M. Cover and P. E. Hart, "Nearest Neighbor pattern classification," *IEEE T Inform Theory*, vol. 13, pp. 21-27, 1967.

[4] P. Perner, "Prototype based classification," *ApplIntell*, vol. 28, pp. 238-246, 2008.

[5] Y. Villuendas-Rey, Y. Caballero-Mota, and M. M. García-Lorenzo, "Using Rough Sets and Maximum Similarity Graphs for Nearest Prototype Classification**,"** *Lecture Notes in Comp Sci* 7441, pp. 300-307, 2012**.**

[6] Y. Villuendas-Rey, C. Rey-Benguría, Y. Caballero-Mota, and M. M. García-Lorenzo, "Nearest prototype classification of Special School families *based on hierarchical compact sets clustering," Lecture* Notes in Artif Intell 7637, pp. 662–671, 2012.

[7] M. Brun, C. Sima, J. Hua, J. Lowey, B. Carroll, E. Suh, and E. R. Dougherty, "Model-based evaluation of clustering validation measures," *Pattern Recogn*, vol. 40, pp. 807-824, 2007.

[8] Z. Huang, "Clustering large data sets with numeric and categorical values," in *Proc. 1^{st}Pacific - Asia Conference on Knowledge discovery and Data Mining*, 1997, pp. 21-34.

[9] A. Ahmad and L. Dey, "A k-means clustering algorithm for mixed numerical and categorical data," *Data KnowlEng*, vol. 63, pp. 503-527, 2007.

[10] A. Ahmad and L. Dey, "A k-means type clustering algorithm for subspace clustering of mixed numeric and categorical data," *Pattern RecognLett*, vol. 32, pp. 1062-1069, 2011.

[11] R. A. Ahmed, B. Borah, D. K. Bhattacharyya, and J. K. Kalita. (2005).*"HIMIC: A Hierarchical Mixed Type Data Clustering Algorithm,"*Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.61.6369&rep=rep1&type=pdf

[12] D. K. Roy and L. K. Sharma, "Genetic k-means clustering algorithm for mixed numeric and categorical datasets," *International Journal of Artificial Intelligence & Applications (IJAIA*), vol. 1, 2010.

[13] X. Cui, J. Gao, and T. E. Potok, "A Flocking Based Algorithm for Document Clustering Analysis," *J Syst Architect*, vol. 52, pp. 505-515, 2006.

[14] I. Triguero, J. Derrac, S. García, and F. Herrera, "Prototype generation for Nearest Neighbor classification: Taxonomy and Experimental Study," *IEEE TPatternAnal*, vol. 34, pp. 417-435, 2012.

[15] S. García, J. Derrac, J. R. Cano, and F. Herrera, "Prototype Selection for Nearest Neighbor Classification: Taxonomy and Empirical Study," *IEEE TPatternAnal*, vol. 34, pp. 417-435, 2012.

[16] D. L. Wilson, "Asymptotic properties of nearest neighbor rules using edited data," *IEEE T Syst Man Cy B*, vol. 2, pp. 408-421, 1972.

[17] P. A. Devijver and J. Kittler, "On the edited nearest neighbor rule," in *Proc. 5th International Conference on Pattern Recognition*, Los Alamitos, California, 1980, pp. 72-80.

[18] K. Hattori and M. Takahashi, "A new edited k-nearest neighbor rule in the pattern classification problem," *Pattern Recogn*, vol. 33, pp. 521-528, 2000.

[19] G. Toussaint, "Proximity Graphs for Nearest Neighbor Decision Rules: Recent Progress," in *Proc. 34 Symposium on Computing and Statistics INTERFACE-2002*, Montreal, Canada, 2002, pp. 1-20.

[20] Y. Caballero, R. Bello, Y. Salgado, and M. M. García, "A method to edit training set based on rough sets," *International Journal of Computational Intelligence Research*, vol. 3, pp. 219-229, 2007.

[21] P. E. Hart, "The condensed nearest neighbor rule," *IEEE TInform Theory*, vol. 14, pp. 515-516, 1968.

[22] G. W. Gates, "The reduced nearest neighbor rule," *IEEE T Inform Theory*, vol. IT-18, pp. 431-433, May 1972.

[23] C. H. Chou, B. A. Kuo, and F. Cheng, "The Generalized Condensed Nearest Neighbor rule as a data reduction technique," in *Proc. 18th International Conference on Pattern Recognition*, 2006, pp. 556-559.

[24] J. A. Olvera López, J. A. Carrasco-Ochoa, and J. F. Martinez Trinidad, "Prototype selection via prototype relevance," Lecture Notes in Comp Sci 5197, pp. 153-160, 2008.

[25] M. García-Borroto, Y. Villuendas-Rey, J.A. Carrasco-Ochoa, and J.F. Martínez Trinidad. "Finding Small Consistent Subset for the Nearest Neighbor Classifier Based on Support Graphs," Lecture Notes in Comp Sci 5856, pp. 465–472, 2009.

[26] B. D. Dasarathy, "Nearest unlike neighbor (NUN): an aid to decision confidence estimation," *OptEng*, vol. 34, pp. 2785-2792, 1995.

[27] C. J. Merz and P. M. Murphy, "UCI Repository of Machine Learning Databases," Department of Information and Computer Science, University of California at Irvine,1998.

[28] A. Rosenberg and J. Hirschberg, "V-Measure: A conditional entropy-based external cluster evaluation measure," in *Proc. 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, Prague, 2007, pp. 410-420.

[29] Y. Zhao and G. Karypis, "Criterion functions for document clustering: Experiments and analysis," Department of Computer Science, University of Minnesota, Technical Report TR 01-40, 2001.

[30] R. D. Wilson and T. R. Martinez, "Improved Heterogeneous Distance Functions," *J ArtifIntell Res*, vol. 6, pp. 1-34, 1997.

Y. Villuendas-Rey was born in Ciego de Ávila, Cuba, in 1983. She obtains her B.Sc on Computer Science Engineering in 2005, and her M.Sc. degree on Applied Informatics in 2007, both from the University of Ciego de Ávila, Cuba. She is currently working towards the PhD degree in the Computer Science Department of the University of Las Villas, Cuba.

She has worked as a lecturer in the Computer Science Department of the Engineering faculty of University of Ciego de Ávila, since 2006. She is currently the head of the Artificial Intelligence Research Group of the Computer Science Department of the University of Ciego de Ávila. Her research interests include supervised classification, data preprocessing, clustering and swarm and evolutionary computation .Prof. Villuendas-Rey is member of the Cuban Society of Mathematics, Physics and Computers, the Cuban Society of Pattern Recognition, and the Cuban Society of Pedagogies.