A New Approach for Semi-Supervised Clustering Based on Fuzzy C-Means

Valmir Macario and Francisco de A. T. de Carvalho

Abstract— **In traditional machine learning applications, only labeled data is used to train the classifier. Labeled data are difficult, expensive, time-consuming and require human experts to be obtained in several real applications. Semi-supervised learning address this issue. Semi-supervised learning uses large amount of unlabeled data, combined with the labeled data, to build better classifiers. The semi-supervised algorithm could be an extension of an unsupervised algorithm. Such algorithm would be based on unsupervised clustering algorithms, adding a term in its objective function that makes use of labeled information to guide the learning process. This study presents a new algorithm for semi-supervised clustering based on Fuzzy C-Means algorithm. The classifier was evaluated and compared against two semi-supervised clustering algorithms in the context of learning from partially labeled data. The behavior of the proposed algorithm is discussed and the results are validated using cross-validation and the confidence interval. Thus, it was possible to certify the better accuracy performance of the new algorithm when a few labeled data are available.**

I. Introduction

As technology develops and more people have access to it, interaction, search and release of information changes continuously. Thus, the necessity to construct tools and create techniques capable to help extract knowledge from this enormous contingent of information in a intelligent and automatic way is evident.

One of the techniques widely used in the task of extracting knowledge is machine learning [15]. This technique constructs computational models that learn how to extract knowledge from the data analysis. Machine learning can be characterized by supervised and unsupervised learning.

Clustering is commonly viewed as an example of unsupervised learning, learning without a teacher. Clustering is the agglomeration of objects (or samples) into clusters so that objects within the same cluster are more similar, according to some similarity measures, while objects from different clusters have a lower similarity. The clustering goal is maximize the homogeneity of the objects in same cluster while maximizing the heterogeneity of objects in different clusters [20]. Fuzzy clustering provides an additional conceptual enhancement by allowing the sample to be allocated to several clusters (classes) to various degrees (membership values). By this, patterns can be treated more realistically and the analysis is capable of identifying eventual outliers (samples that are more difficult to be assigned into a single category).

Valmir Macario and Francisco de A. T. de Carvalho are with Center of Informatics, Federal University of Pernambuco; Recife; PE; Brazil; 50732-970 (e-mails:{vmf2,fatc}@cin.ufpe.br).

The unsupervised learning presents some significant limitations. Unlike what happens in the supervised process, the results of an unsupervised learning are just the clusters or partitions, without descriptive information about the generated partitions. However, users are not often interested only in these partitions, but also in some explanation of what those partitions represent. To overcome this limitation, it is necessary to interpret the partitions found, which is not an easy task, since it must perform complex inferences to obtain such information.

Therefore, using a supervised algorithm can be considered the alternative to manual demarcate a data set. However, this task can be extremely complex, expensive, time-consuming and dependent on human experts in several real applications. As an example, the extraction process in an official journal requires each line of the document to be labeled. The publication of only one day of that journal may contain over one hundred thousand lines [13]. Another example is *Spam*, where the classifier is forced to learn the e-mails not allowed from a small percentage of labeled e-mails [9], we can also mention applications in text mining [16]. Thus, the use of supervised algorithms is not feasible because there are few labeled data available for training the algorithm. These factors suggest a technique that uses unlabeled examples to improve the accuracy of a classifier. This way, the use of supervised algorithms is unviable, because there are few labeled data available for training the algorithm. These factors suggest a technique that uses unlabeled examples to improve the accuracy of a classifier.

The semi-supervised learning [7][22] is an intermediate approach between supervised and unsupervised learning. In semi-supervised learning, labeled and unlabeled examples are used to guide the learning algorithm in order to build better classifiers.

In especial, the interest of this work is in clustering semi-supervised technique, due the importance of classic algorithms of clustering unsupervised algorithms. The clustering task has been applied in several problems, such as text mining, gene expression, image processing, among others.

The paper is organized as follows. Section II surveys the state of the art in the framework of semi-supervised clustering. Section III introduces the details of our semi-supervised clustering algorithm. Section IV discusses the analysis of the algorithm explaining the adopted methodology. Then, in Section V, the algorithm is compared against two semi-supervised learning algorithms on partially labeled data. Finally, Section VI highlights some future work and concludes the paper.

978-1-4244-8126-2/10/$26.00 ©2010 IEEE

II. Related Work

Several semi-supervised algorithms have been proposed. They can be categorized based on the computational model used. Based on the Zhu survey [22], the important models are: self-training, probabilistic model, the objective function optimization model and support vector machines. Some illustrative examples based on these models are presented below.

The self training technique trains the algorithm with a few labeled data set. The trained classifier labels the amount of unlabeled data set. samples with high confidence are added to the training data set. Thereby, the algorithm is trained again with this new training data set. This procedure is repeated until most data are classified with higher confidence. This technique have been used mainly in natural language processing [19][12].

Nigam et al. [16] investigated a probabilistic approach for text classification. The approach combines the Expectation-Maximization (EM) algorithm and a naive Bayes classifier. The algorithm trains the classifier using the labeled data only. Then, the labels of the unlabeled samples are iteratively estimated and the classifier is re-trained using all labeled data until convergence. Blum and Mitchell [3] used the co-training technique which assumes that the feature can be divided into two independent subsets. Each of these is used to train a particular algorithm. Then, each classifier is trained with the higher examples classified by other classifier. Zhou and Li [21] proposed tri-training model which uses three learners. If two of them agree on the classification of an unlabeled point, the classification is used to teach the third classifier. This approach thus avoids the need of explicitly measuring label confidence of any learner. It can be applied to datasets without different views, or different types of classifiers.

The transductive support vector machine (TSVM) [11] is an extension of the support vector machine (SVM) algorithm, that uses unlabeled data in the training phase. The Chapelle e Zien [6] work proposed a new algorithm called ∇SVM which approximates the hat loss with a Gaussian function, and performs gradient search in the primal space. Then, Chapelle et al.[8] added the global optimal solution for small datasets instead of finding a global optimal approximation solution.

More relevant to our work are the approaches investigated by Pedrycz and Waletzky [17] and Bouchachia and Pedrycz [4] which are typical examples of the objective function optimization model for clustering with partial supervision. The first algorithm modified a version of the classic Fuzzy C-Means (FCM) algorithm to deal with the problem of partial supervision. Currently, this same algorithm is being used to classify and categorize digital images [18]. The objective function was extended to include a second term that utilizes available labels to improve the membership degree assigned by the algorithm to the clusters that represent samples class. In this objective function, labeled and unlabeled data are identified by means of a boolean vector: $b = [b_i], i = 1, 2, ..., N$, where N is the size of the data set. $b_i = 1$ if sample x_i is labeled, otherwise 0. Likewise, the membership values of the labeled samples are arranged in a matrix $F = [f_{ik}]$ such that $k = 1, 2, ..., C$ where C is the number of clusters and $i = 1, 2, ..., N$. The objective function is:

$$J(U, V) = \sum_{i=1}^{C}\sum_{k=1}^{N} u_{ik}^2 d_{ik}^2 + \alpha \sum_{i=1}^{c}\sum_{k=1}^{N} (u_{ik} - f_{ik} b_i)^2 d_{ik}^2 \quad (1)$$

In the objectives functions, U is the partition matrix, such that each u_{ik} indicates the membership degree of the data point x_i to cluster v_k, and V designates the set of prototypes v_k associated with clusters k. The parameter d_{ik} indicate the distance between a cluster prototype k and a data point x_i. The parameter α is a scaling factor to maintain the balance between the supervised and unsupervised components of the objective function. If $\alpha = 0$, the objective function reduces to that of FCM. The membership degree follow the constraints:

$$\sum_{k=1}^{C} u_{ik} = 1 \; \forall \; i, \;\; 0 < \sum_{i=1}^{N} u_{ik} \; < \; N \; \forall \; k \quad (2)$$

The algorithm proposed by Bouchachia and Pedrycz [4] basically extends the objective function of the FCM algorithm. The purpose of this change is to capture data structures hidden and visible through two terms of the objective function. Hidden structures are acquired by the first term of the objective function, which is equal to the FCM objective function. The second term takes into account the structures reflected by the assessment of labels available. The strength of this algorithm is the possibility of one cluster representing more than one class. Thus, the objective function becomes:

$$J(U, V) = \sum_{i=1}^{C}\sum_{k=1}^{N} u_{ik}^2 d_{ik}^2 + \alpha \sum_{i=1}^{C}\sum_{k=1}^{N} (u_{ik} - \tilde{u}_{ik})^2 d_{ik}^2 \quad (3)$$

The term $\tilde{u}_{ik}$ of the matrix $\tilde{U}$ have a second optimization and is iteratively computed as follows:

$$\tilde{u}_{ik}^{(s)} = \tilde{u}_{ik}^{(s-1)} - \beta \frac{\delta Q(F, \tilde{U})}{\delta \tilde{u}_i k} \quad (4)$$

where s is the iteration counter and:

$$Q(F, \tilde{U}) = \sum_{h=1}^{H}\sum_{i=1}^{N} \delta_i (f_{ih} - \sum_{k \in \pi_h} \tilde{u}_{ik}) \; \tilde{u}_{ik} \; \in \; [0, 1] \quad (5)$$

Likewise, in Equation 5, $F = [f_{ih}]$ is a binary matrix $H \; x \; N$ such that $f_{ih} = 1$ if sample x_i belongs to class h, otherwise $f_{ih} = 0$. This matrix is to represent the labels available. π_h is the clusters set that belongs to a class h. $\delta_i = 1$ if sample x_i is labeled, otherwise 0.

The new approach suggested by this paper provides four main characteristics that differ from the other existing approaches. The new algorithm avoids the assumption that the number of clusters determined by the clustering algorithm should be the same as the number of classes reflected by data labels as in Bouchachia and Pedrycz [4]. The proposed algorithm distance is determined between two samples instead

of the distance from a sample to a prototype. As it follows, the proposed algorithm doesn't have the scaling factor α avoiding other user free parameter.

III. Semi-supervised Clustering Model

The main idea of semi-supervised clustering is to take advantage of some descriptive information from a few data. Usually this information is the label. Enjoying this information, the algorithm optimizes the objective function to label examples in a precise way. The developed algorithm adds a second term supervised to the original function of Fuzzy C-Means algorithm. In the second term, the new algorithm assumes that information from neighborhood labeled examples are important to discover the data structure. Thus, at optimization function, the neighbors examples are compared and, if they belong to the same class, the function is optimized to this direction. Optimization is performed by the difference between the membership values of the same class examples. We assume a short distance between same class examples. So, the difference of membership degree value for this class is penalized if this value is large.

The proposed semi-supervised objective function is:

$$J(U,V) = \sum_{k=1}^{C}\sum_{i=1}^{N} u_{ik}^2 d_{ik}^2 + \sum_{k=1}^{C}\sum_{l=1}^{P}\sum_{i=1}^{N}\sum_{j=1}^{N} t_{il}t_{jl}(u_{ik}-u_{jk})^2 d_{ij}^2 \quad (6)$$

such that U follow the constraints:

$$\sum_{k=1}^{C} u_{ik} = 1 \; \forall \; i, \;\; 0 < \sum_{i=1}^{N} u_{ik} \; < \; N \; \forall \; k \quad (7)$$

N is the size of the data set, C is the number of clusters formed by the algorithm and P is the number of classes previously known. Note that the proposed algorithm, as well as the Bouchachia and Pedrycz [4] algorithm, a class can be represented by one or more clusters. U is the partition matrix, such that each u_{ik} indicates the membership degree of the data point x_i to cluster v_k, and V designates the set of prototypes v_k associated with clusters k. The parameter d_{ij} indicate the distance between a data point x_i to a data point x_j. For all, $i = (1, 2, ..., N)$ and $k = (1, 2, ..., C)$. Note that this algorithm does not have the balancing factor α present in the other algorithms. The information is used by the supervised binary term $t_{ik},\ i = 1, 2, 3, ..., N$. This value is 1 if the sample x_i belongs to cluster v_k and 0 otherwise:

$$t_{ik} = \begin{cases} 1 & if \; \mathbf{x_i} \in \mathbf{v_k} \\ 0 & otherwise \end{cases}$$

To optimize the objective function, the Lagrange multiplier with respect to the matrices U and V. Then, we obtain the equations for compute iteratively the membership degree u_{ik} and prototypes v_k. Optimizing in relation to U we have the equation:

$$u_{ik}^{(s)} = \frac{1 + \sum_{h=1}^{C} \frac{\sum_{l=1}^{P}\sum_{j=1}^{N} t_{il}t_{jl}|[u_{jk}^{(s-1)} - u_{jh}^{(s-1)}]|d_{ij}^2}{d_{ih}^2 + \sum_{l=1}^{P}\sum_{j=1}^{N} t_{il}t_{jl}dij^2}}{\sum_{h=1}^{C} \frac{d_{ik}^2 + \sum_{l=1}^{P}\sum_{j=1}^{N} t_{il}t_{jl}d_{ij}^2}{d_{ih}^2 + \sum_{l=1}^{P}\sum_{j=1}^{N} t_{il}t_{jl}d_{ij}}} \quad (8)$$

where s is the iterations counter. One characteristic of this calculation is the requirement of a membership degree of a previous iteration. So, we need to populate the matrix U before using this equation for the first time.

Now, optimizing for the matrix V we have the equation for calculating the prototype. This equation is the as the one used in FCM algorithm:

$$\mathbf{v_k} = \frac{\sum_{i=1}^{N} u_{ik}^2 \mathbf{x_i}}{\sum_{i=1}^{N} u_{ik}^2} \quad (9)$$

After having formulated all required expressions, the clustering process can then be formulated. It consists of the steps shown in Algorithm 1.

1 *Initiate the algorithm parameters: set C such that $1 < C \leq N$; Fixing $Maxiter$ (number of iterations); Start the iterations counter $s = 0$; Assign a value to ϵ, such that $0 < \epsilon << 1$; Initiate the Prototype Matrix; Initiate the membership Matrix, including all known memberships*;
2 *Obtain the prototypes values v_k using* (9);
3 *Compute the objective function J using* (6);
4 **if** *($|J^{(s)} - J^{(s-1)}| \leq \epsilon$ or $s \geq MaxIter$) and $(s > 1)$* **then**
5 *Stop algorithm*
6 **end**
7 **else**
8 *Update the matrix partition $U^{(s)}$ using* (8);
9 *Go to step 2*;
10 **end**

Algorithm 1: Semi-supervised proposed algorithm

IV. Evaluation Of The Algorithm

This section we present the methods to evaluate the semi-supervised algorithms.

A. Cross-Validation

In order to evaluate the mean of the accuracy rate, is used the k-fold cross-validation procedure [15]. The data set is divided into k disjoint equal size sets. Then, training is performed in k steps, each time using a different fold as the test set, and the union of the remaining folds as the training set. Applying the distinct algorithms to the same folds with a k at least equal to thirty, the statistical significance of the differences between the methods can be measured, based on the mean of the accuracy rate from the test sets.

In unsupervised learning, when there is a *a priori* classification of the data set available, the comparison between two methods can also be done by detecting the statistical

significance of the difference between the mean value of a certain external index. It is important to point out that the *a priori* classification is not used in the training, but only to evaluate the results. In unsupervised cross-validation method, the training set is presented to the clustering method, the result is a partition (training partition). Then, the nearest centroid technique [14] is used to build a classifier from the training partition. This technique calculates the proximity of each example of the test set to each prototype, or each cluster center, in the training partition. Thus, each test example is assigned to the cluster whose proximity is the lowest calculated. Then the set test is compared with the previous partition using an external index.

As The semi-supervised learning is in the middle of supervised and unsupervised learning, a cross-validation methodology used for unsupervised methods was adapted [10]. One of the adaptations is the training partition generation. As the goal of a semi-supervised method is to improve classification with few labeled data available, the training partition should have different percentages of labeled an unlabeled data. In training fold, the labeled data percentages are $(0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100)$ of the total data available. The labeled examples are chosen randomly. Thus, for each different configurations of labeled data, the data is partitioned into k groups of approximately equal sizes.

After this step, the technique of nearest centroid is adjusted. In the original proposed by Costa [10], the example belongs to the cluster whose similarity is greater. The similarity is calculated using the original form of the algorithm. However, because of the fuzzy characteristic, in a fuzzy partition, a sample belongs to all clusters quantified by the membership degree of the sample to each cluster. Here, we have assigned an example to a cluster that the membership is the greater one. The membership degree is calculated by the original equation of the semi-supervised clustering algorithm.

Formally, let D be the data set; n the number of clusters; L_i the i^{th} the labeled data percentage; F_i the i^{th} test fold (or set); R_i the resulting partition of the training set $D - F_i$; C_i the set of centroids of partition R_i; T_i the resulting partition of test fold F_i; and P_i the priori partition with the objects from F_i, for $i = 1, ..., k$; then, the semi-supervised k-fold cross-validation procedure works as follows:

input : Database D

foreach L_i **do**
  Label D according to percentage L_i;
  D is randomly divided into k equal and disjoint folds F_i;
  for $i \leftarrow 1$ **to** k **do**
    Apply the clustering method to the set $D - Fi$ obtaining partition R_i with n clusters as result;
    Calculate the n centroids of the clusters in R_i, forming C_i;
    Calculate the membership degree between the centroids in C_i and the objects in F_i;
    Assign the objects of F_i, according to the greater membership degree in C_i obtaining partition T_i as result;
    Measure the agreement of partitions T_i and P_i with a accuracy rate;
  end
end

Algorithm 2: Cross-Validation

TABLE I
CONFUSION MATRIX

i/j	1	2	3
1'	50	20	10
2'	10	20	30
3'	0	60	0

B. Accuracy Rate

In fact, the results obtained with clustering methods consist of structures such as partitions or hierarchies, which cannot be directly used to classify other objects. Thus, we have to know which class the cluster represents to confirm if the algorithm assigned the example to the correct class. As it follows, there is explanation of this method.

A confusion matrix is generated. The confusion matrix is characterized by the lines l representing the cluster assigned by the algorithm and column c, which represents the class to which the example belongs originally. Thus, the matrix cell represents the number of examples that were allocated to cluster l and originally belong to class c. After this step, different row and column combinations are tested to discover the maximum number of examples correctly allocated. Consider the Table I.

The combinations are presented below, representing the line versus column value. The combination sum is presented in the last column. As it shows, the second line sum is the greatest. Thus, this combination is chosen. Cluster 1' represents class 1, cluster 3' represents class 2 and cluster 2' represents class 3.

1-1' 2-2' 3-3' 70
1-1' 2-3' 3-2' 140
1-2' 2-1' 3-3' 30
1-2' 2-3' 3-1' 50
1-3' 2-1' 3-2' 80
1-3' 2-2' 3-1' 30

Armed with the partition descriptive information, it is necessary to measure the accuracy of the clustering semi-supervised results to evaluate the algorithms. The basic idea is based on the comparison of two supervised learning method. This comparison is often accomplished by analyzing the statistical significance of the difference between the mean of the classification accuracy rate, on independent test sets, of the methods evaluated. The accuracy rate is here calculated by the Equation 10 below:

$$AR = \frac{Number\ of\ correctly\ assigned\ data\ points * 100}{Size\ of\ the\ testing\ data\ set} \quad (10)$$

C. Confidence Interval

To evaluate the accuracy rate found by the experiments, statistical tests such as confidence interval must be used. A confidence interval is a way to estimate a value of an unknown parameter. The main idea is to construct a confidence interval for the unknown parameter with a probability $(1-\alpha)$ that the interval contains the true parameter [5].

Thus, a higher confidence interval represents that is more likely to this interval to contain the actual value of θ. Moreover, the higher interval will have less information about the true value of θ. Thus, ideally, we have a relatively short interval with a high confidence.

We may use confidence intervals to hypothesis testing, since the confidence interval construction comes from hypothesis testing theory. The hypothesis test is a decision rule to accept or reject a statistical hypothesis to be tested based on sample elements. The hypothesis test namely, generally, H_0 (null hypothesis) the statistical hypothesis to be tested and H_1 the alternative hypothesis. The H_0 rejection will lead to the H_1 acceptance. The alternative hypothesis usually represents the assumption that the researcher wants to prove, H_0 is formulated with the express objective of being rejected. When we compare results, we construct a confidence interval for each sample. Therefore, we can say the null hypothesis H_0 could be rejected if the confidence intervals have no intersection.

D. Semi-Supervised Classification

The methodology presented by this work objectives the validation of labels found by the a newly proposed algorithm with different percentages of labeled data available for training the algorithm. This methodology is based on the previous work of Amini and Gallinari [1] and Pedrycz and Waletzky [17]. The experiment steps are explained below:

1) **Original data partitioning:** The database is divided into 2 data sets, the labeled data set and the unlabeled data set. The labeled data set contains a certain percentage of labeled examples chosen randomly from original database. The original data is partitioned in a stratified mode. Thereby, the original data class distribution is maintained. in other words, if originally the database have 30% of a class A and 70 % of a class B, these values are kept in the labeled and unlabeled data sets. The labeled percentage range from $(0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100)$ from the total database. The remnant data will be part of the second set, the unlabeled data set. Thus, for respective labeled set values, these sets use the percentages $(100, 90, 80, 70, 60, 50, 40, 30, 20, 10, 0)$ from total database. The experiments are performed by presenting the two data sets to the clustering semi-supervised algorithm with proper proportions of each of the sets. Thus, we can observe the influence of the amount labeled and unlabeled data in the algorithm accuracy performance.
2) **Algorithm execution:** The algorithms inputs are the two subsets constructed in step 1. The output is a partition with a certain number of clusters containing all data examples. These experiments are evaluated by 3 databases presented in next section: Iris, Wine and a Synthetic. To measure the statistical significance of those classification results, cross-validation using 30 runs is performed. In each run, the data is randomly shuffled. For each experiment iteration, the algorithm initialization is repeated 20 times, then the best iteration of these repetitions is chosen. The selected iteration is the one in which the objective function value obtained the lowest one against the others iterations. The cross-validation is performed using all semi-supervised clustering algorithms evaluated in this work: the proposed algorithm, the algorithm proposed by Pedrycz [17], and the algorithm proposed by Bouchachia [4].
3) **Partition Evaluation:** In this step, the partition obtained by the algorithms are compared with the original labeled partition. To compare the partitions we have calculated the accuracy rate mean to measure the quality of the partition obtained by each algorithm. Then, intervals with 95% of confidence are built to compare the clustering algorithms performance in a classification task.

Figure 1 illustrates the labeling process achieved in this experiment. First, data is partitioned into sets labeled and not labeled. The set containing gray triangles represents the unlabeled data, the set containing the stars and positive signals represent the data labeled in two classes, respectively. Then, the algorithm uses two sets as input. The results of the algorithm are fuzzy partitions representing by their prototypes, the dark star represents the star class and the clear gray star represents the positive signal class. The test data are assigned to the clusters according to the greater membership degree. So, this example generates 3 partitions with two classes represented by dark star prototypes and another class represented by the clear gray. The partition now has the test data labeled.

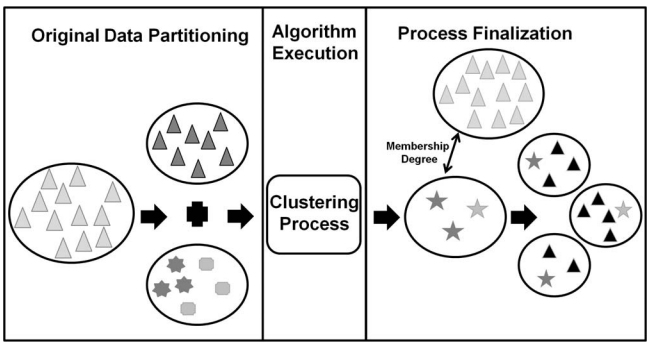


Fig. 1. Semi-supervised clustering

V. Comparative Study Of Some Semi-supervised Learning Algorithms

In this section, we will compare the classification task performance of the proposed algorithm against two methods: A clustering semi-supervised algorithm proposed by Pedrycz [17], and another semi-supervised clustering algorithm proposed by Bouchachia [4].

A. Iris Data

The Iris database contain 150 samples describing 3 types of the Iris plants. There are 4 attributes describing the length and width measures of petals and sepals of the plant. There are 50 samples to representing each plant. In this experiment (and also in all experiments in this paper) α is set to 1 in the Pedrycz and Bouchachia algorithms, which means that the provided labels of the labeled data are completely trustful and accurate. Because Iris has 3 classes, we set the algorithm to produce 3 clusters in the final partition. Figure 2 portray the results for the experiment using Iris

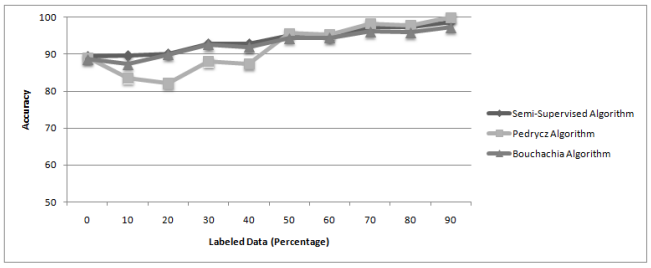
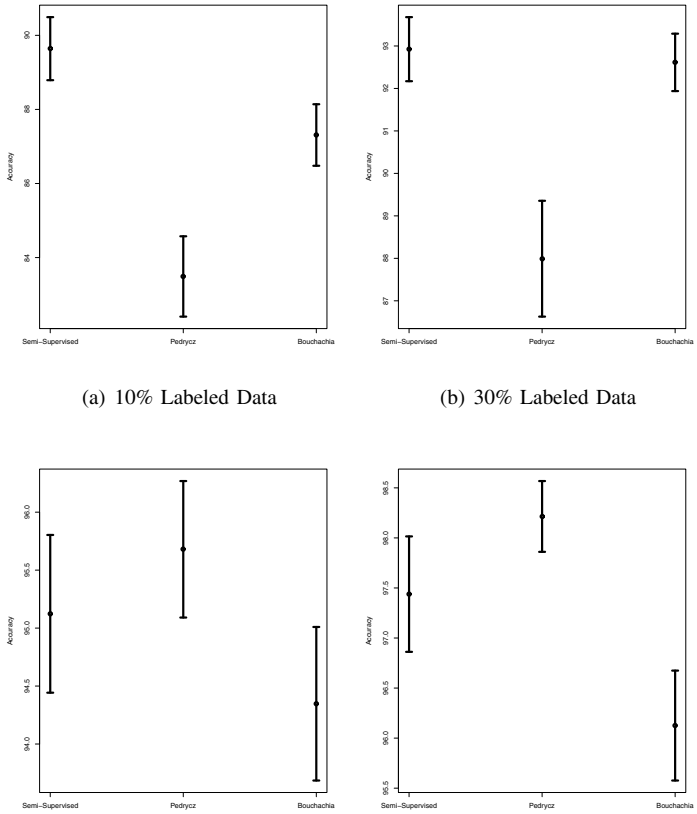


Fig. 2. Accuracy rate - iris data

data base. We can observe that the proposed algorithm accuracy performance follows the best results obtained by the two other algorithms. When one is better than the other, the proposed algorithm obtains results similar to the better outcome result. With 0% to 40% labeled data, the proposed algorithm had little advantage accuracy performance over the Bouchachia algorithm which obtained the second best performance. With 50% to 90%, the proposed algorithm have similar results to the ones obtained by Pedrycz algorithm. With this configuration the advantage is slightly towards the Pedrycz algorithm. To check the statistical significance of these results, we have chosen four configurations (10%, 30%, 50% and 70%) labeled data to construct a 95% confidence interval presented in Figure 3. The proposed algorithm obtained the best accuracy performance with 10% labeled data, the Bouchachia obtained the second best accuracy, followed by Pedrycz algorithm. With 30% labeled data, the proposed and Bouchachia algorithms have had similar accuracy performance, the Pedrycz has obtained a low performance. With 50%, the best performances were obtained by the proposed and Pedrycz algorithms. However, as showed, the Pedrycz algorithm performance was better than the Bouchachia performance, but the proposed algorithm did not have a better performance than the Bouchachia algorithm. For 70% labeled data, the best performance was again obtained by the Pedrycz and the proposed algorithm,



(a) 10% Labeled Data (b) 30% Labeled Data

(c) 50% Labeled Data (d) 70% Labeled Data

Fig. 3. Confidence interval for iris data

but in this configuration we could say the two algorithms have a better accuracy performance overall.

B. Wine Data

The Wine database has 768 samples that describe 3 types of wines. There are 13 attributes that describe chemical factors. The classes are unbalanced, with 59 samples representing wine 1, 71 samples representing Wine 2 and 48 samples representing wine 3. In this experiment, the algorithms produce 3 clusters in the final partition.

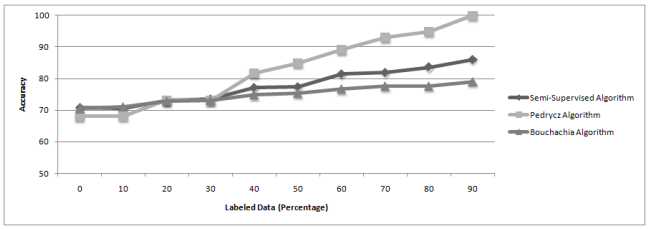
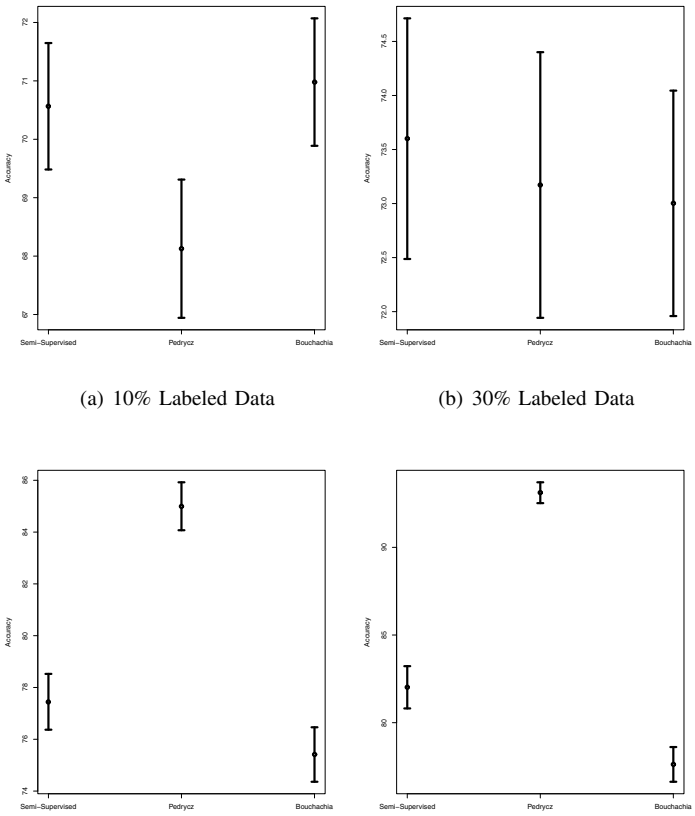


Fig. 4. Accuracy rate - wine data

The evaluation of these three approaches using Wine data has shown that by increasing the amount of labeled data, the accuracy of these approaches increases as presented at Figure 4. With a few labeled data, 0% and 10%, the proposed

algorithm has obtained similar results to the Bouchachia algorithm. With 20% and 30% labeled data the three approaches had presented similar accuracy performance. With 40% to 90% the best accuracy performance was obtained by the Pedrycz algorithm, the second best accuracy performance was obtained by the proposed algorithm and the worst performance was obtained by the Bouchachia algorithm. In this experiment, the Pedrycz algorithm gained prominence. The proposed algorithm has obtained good results when using few data labeled (0% to 30%), similarly to the Bouchachia algorithm.



(a) 10% Labeled Data (b) 30% Labeled Data

(c) 50% Labeled Data (d) 70% Labeled Data

Fig. 5. Confidence interval for wine data

To check the statistical significance of these results, we have chosen four configurations (10%, 30%, 50% and 70%) of labeled data to construct a 95% confidence interval. The proposed algorithm has obtained an accuracy performance equivalent to the Bouchachia algorithm 10% labeled data. When 30% labeled data were available, the three approaches had similar accuracy performance. With 50% and 70% labeled data, the Pedrycz algorithm had the best accuracy performance, the proposed algorithm has got the second best performance followed by the Bouchachia algorithm.

C. Synthetic Data

The Synthetic database is generated according to some characteristics statistical namely mean and standard deviation (μ, σ). These parameters were taken of the Bouchachia [4]. Bouchachia have also used this database in order to evaluate data sets having spatial distributions, which are difficult to handle using simple clustering algorithms. This base consists of two classes shown in Table II. The first class h_1 consists of one cluster and the second class consists of two clusters. Each cluster has 100 samples described by two attributes (namely att in the table).

TABLE II
SYNTHETIC DATA

Characteristic		μ(att 1)	μ(att 2)	σ(att 1)	σ(att 2)
Class	H_1	3.0	1.0	7.0	0.5
	H_2	4.0	4.5	1.0	1.0
		2.0	-2.5	1.0	1.0

Figure 6 shows the accuracy performance of the three approaches. With 0% to 30% labeled data, the proposed algorithm accuracy performance was better than the other approaches with a little over the Bouchachia algorithm and a clear advantage over the Pedrycz algorithm. With 40% to 60%, the Pedrycz algorithm had the best performance accuracy, and the proposed algorithm had similar accuracy performance compared to the Bouchachia algorithm. With 70 to 90% labeled data, the Pedrycz algorithm kept the best accuracy performance, the proposed algorithm had the second best performance followed by the Bouchachia algorithm.

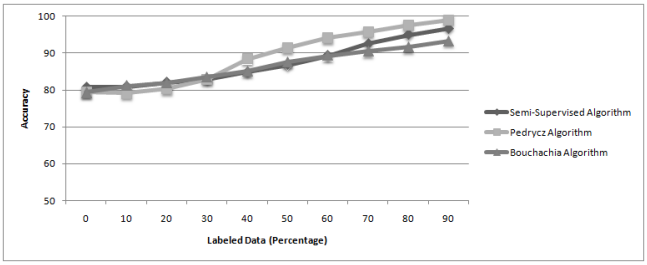
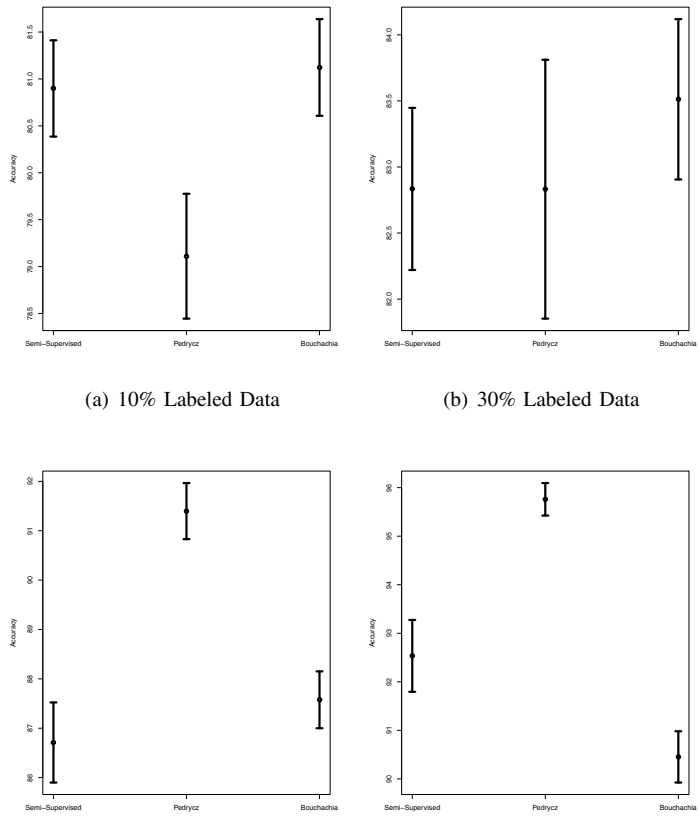


Fig. 6. Accuracy rate - synthetic data

The confidence interval is presented in Figure 7. The proposed algorithm has obtained best accuracy performance with 10% labeled data, the Bouchachia has obtained the second best accuracy and Pedrycz algorithm had the third accuracy for this configuration. With 30% labeled data, the proposed and Bouchachia algorithms had similar accuracy performance, the Pedrycz had a lower performance. With 50%, the best performances were obtained by the proposed and Pedrycz algorithms. However, as showed, the Pedrycz algorithm performance was better than the Bouchachia performance, but the proposed algorithm did not have a better performance than the Bouchachia algorithm. For 70% labeled data, the best performance was again obtained by the Pedrycz and the proposed algorithms, but in this configuration we could say that the two algorithms had a better accuracy performance than the Bouchachia algorithm.



(a) 10% Labeled Data (b) 30% Labeled Data

(c) 50% Labeled Data (d) 70% Labeled Data

Fig. 7. Confidence interval for synthetic data

VI. Conclusions

This paper discussed a new approach to performing fuzzy clustering with partial supervision. This approach exploits available knowledge about data to supervise the clustering process. The experimental evaluation has shown that the approach performs very well the ability to overcome a classification task. What draws more attention in results is the best accuracy performance of the proposed algorithm when there are only a few labeled data to train the algorithms. Another important point is that the performance increases as more labeled data is available to train the algorithms. Therefore, the new algorithm is able to obtain good results in real applications where primarily exist few labeled data.

The proposed algorithm performs a minimization of the objective function. The similarity measure used in this objective function is the basic Mahalonobis distance, transformed into Euclidean distance. As future goal, we intend to enhance this similarity function. The adaptive distances allow the construction of partitions in various formats, in addition to spherical shape generated by the Euclidean distance, thereby it may learn more complex data distribution structures. We intend to use adaptive distances in order to improve the overall algorithm performance.

Acknowledgment

We are grateful to the Brazilian agencies CNPq, CAPES and FACEPE for their financial support.

References

[1] M. R. Amini and P. Gallinari, "Semi-supervised learning with an imperfect supervisor heuristic programming," *Proc. Knowledge and Information Systems*, 2005, pp. 385–413.

[2] J.C. Bezdek, "Pattern Recognition With Fuzzy Objective Function Algorithms," *Plenum*, 1981.

[3] A. Blum and T. Mitchell. "Combining labeled and unlabeled data with co-training," *Proceedings of the Workshop on Computational Learning Theory*, 1998.

[4] A. Bouchachia and W. Pedrycz, "Data clustering with partial supervision," *Data Mining and Knowledge Discovery*, vol. 12, pp. 47–78, 2006.

[5] H. Bolfarine and C. Sandoval, "Introducao a Inferencia Estatistica," *Sociedade Brasileira de Matematica*, 2001.

[6] O. Chapelle and A. Zien, "Semi-supervised classification by low density separation," *Workshop on Artificial Intelligence and Statistics*, 2005.

[7] O. Chapelle, A. Zien, and B. Scholkopf, "Semi-supervised learning", *MIT Press*, 2006.

[8] O. Chapelle, V. Sindhwani, and Keerthi, "Branch and bound for semisupervised support vector machines," *Advances in Neural Information Processing Systems*, 2006.

[9] V. Cheng and C.H. Li, "Personalized spam filtering with semi-supervised classifier ensemble," *International Conference on Web Intelligence*, 2006.

[10] I. G. Costa, F. A.T. De Carvalho, and M. C.P. de Souto, " Comparative study on proximity indices for cluster analysis of gene expression time series," *Journal of Intelligent & Fuzzy Systems*, vol. 13, pp.133–142, 2003.

[11] T. Joachims, "Transductive inference for text classification using support vector," machines. *International Conference on Machine Learning*, Morgan Kaufmann, pp. 200–209, 1999.

[12] B. Maeireizo, D. Litman, and R. Hwa, "Co-training for predicting emotions with spoken dialogue data," *Proc. Annual Meeting of the Association for Computational Linguistics*, 2004.

[13] V. Macario, R. B. C. Prudncio, F. A. T. De Carvalho, L. Rodrigues L. R. Torres, and M. G. Lima, "Automatic information extraction in semi-structured official journals," *Brazilian Symposium on Neural Networks*, 2008.

[14] R. M. McIntyre and R. K Blashfield, "A nearest-centroid technique for evaluating the minimum-variance clustering procedure," *Multivariate Behavorial Research*, vol. 15, pp. 225–238, 1980.

[15] T. Mitchel, "Machine Learning," *McGraw Hill*, 1997.

[16] K. Nigam, A. K. McCallum, S. Thrun, and T. Mitchell, "Text classification from labeled and unlabeled documents using EM," *Machine Learning*, vol. 39, pp. 103–134, 2000.

[17] W. Pedrycz and J. Waletzky, "Fuzzy clustering with partial supervision," *IEEE transactions on system, man and cybernetics*, vol. 27, N. 5, 1997.

[18] W. Pedrycz A. Amato, V. D. Lecce, and V. Piuri, "Fuzzy clustering with partial supervision in organization and classification of digital images," *IEEE Transaction on Fuzzy Systems*, vol. 16, N. 4, pp. 10081026, 2008.

[19] E. Riloff, J. Wiebe, and T. Wilson, "Learning subjective nouns using extraction pattern bootstrapping," *Conference on Natural Language Learning*, 2003.

[20] R. E. Stepp and R. S. Michalski, "Machine Learning: An Artificial Intelligence Approach," , in chapter Conceptual Clustering: Inventing Goal-Oriented Classifictions of Structured Objects, vol. 2, pp. 471–478, Morgan Kaufmann, 1986.

[21] Z. H. Zhou and M. Li, "Tri-training: exploiting unlabeled data using three classifiers," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, pp. 1529–1541, 2005.

[22] X. Zhu, "Semi-Supervised Learning Literature Survey," *Carnegie Mellon University*, 2008.