Computación y Sistemas Vol. 5 No. 4 pp. 241 - 246
© 2002, CIC - IPN. ISSN 1405-5546 Impreso en México

C-Means Algorithm with Similarity Functions

Algoritmo C-means con Funciones de Similaridad

José Francisco Martínez Trinidad[1], Javier Raymundo García Serrano[2] and Irene Olaya Ayaquica Martínez[3]
[1]Instituto Nacional de Astrofísica, Óptica y Electrónica, Ciencias Computacionales
Luis Enrique Erro No. 1, Sta Ma. Tonanzintla, Puebla, México, C.P. 72840
[2]Centro Nacional de Investigación y Desarrollo Tecnológico, A.P. 5-164, CP. 62051, Cuernavaca, Morelos
[3]Centro de Investigación en Computación-IPN, Laboratorio de Procesamiento de Imágenes
Juan de Dios Bátiz s.n esq. Miguel Othón de Mendizabal, CP. 07738, México, D.F.
e-mail : fmartine@inaoep.mx, acjrgs@cenidet.edu.mx, ire_aya@hotmail.com

Article received on November 11, 2000; accepted on May 20, 2002

Abstract

In this paper, an extension to the C-means algorithm that considers object descriptions with quantitative and qualitative features (mixed data) is proposed. In addition, the algorithm considers missing data. These kinds of descriptions (mixed data) are very frequent in soft sciences as Medicine, Geology, Sociology, Marketing, etc. Therefore, the algorithm's application scope is very wide. In the algorithm, we propose the use of similarity functions that may be in function of partial similarity functions. These functions consequently allow compare objects through analysis of object sub descriptions. In addition, a comparison with the classical C-Means algorithm and results using standard public databases are shown.

Keywords: unsupervised pattern recognition, clustering, data analysis.

Resumen

En este trabajo, se propone una extensión del algoritmo C-means que considera descripciones de objetos con variables cualitativas y cuantitativas (datos mezclados). Además, el algoritmo considera ausencia de información. Este tipo de descripciones (datos mezclados) son muy frecuentes en ciencias poco formalizadas como Medicina, Geología, Sociología, Mercadotecnia, etc. Por lo tanto, el campo de aplicación del algoritmo es amplio. En el algoritmo, proponemos el uso de funciones de similaridad que pueden estar en función de funciones de similaridad parcial. Estas funciones por consiguiente permiten comparar objetos a través de subdescripciones de objetos. Además, se muestra una comparación con el algoritmo C-means clásico así como los resultados obtenidos usando bases de datos públicas estándar.

Palabras Clave: Reconocimiento de Patrones no supervisado, Agrupamiento, Análisis de datos.

1 Introduction

Restricted unsupervised classification (RUC) problems have been studied intensely in Statistical Pattern Recognition (Duda and Hart, 1973; Schalkoff, 1992). The C-means algorithm, which is based on a metric distance in a n-dimensional metric space, has shown its effectiveness in the solution for many unsupervised classification problems. This algorithm has been motive of many extensions since the first publication by Ball and Hall (1967). These extensions consider mainly the following aspects: the selection of initial seeds (see Bradley and Fayad, 1998); the determination of the optimal number of clusters (see Dubes, 1987) and the use of different functionals to generate the clusters (see Bobrowsky and Bezdek, 1991).

The C-means algorithm starts with an initial partition then it tries all possible moving or swapping of data from one group to others iteratively to optimize the objective measurement function. The objects must be described in terms of features such that a metric can be applied to evaluate the distance. Nevertheless, the conditions in soft sciences as Medicine, Geology, Sociology, Marketing, etc., are quit different. In these sciences, the objects are described in terms of quantitative and qualitative features. For example, if we look at geological data, features such as age, porosity, and permeability, are quantitative, while others such as rock types, crystalline structure and facies structure, are qualitative.

Likewise, missing data is common in this kind of problems. In these circumstances, only the degree of similarity between the objects can be determined. Nowadays, there are several algorithms to solve the RUC problem in a context as that mentioned previously. The conceptual C-means algorithm of Ralamboundrainy (1995) is the most representative of all.

This algorithm proposes a distance function to handle quantitative and qualitative features. The distance between two objects is computed evaluating the distance between quantitative features (with an Euclidean distance) plus the distance between qualitative features (using the chi-square distance in order to evaluate this distance, each value of a qualitative feature being coded as a binary feature). The above

mentioned distance is interpreted in the original n-dimensional space, and the centroids are computed. But, this way of interpreting and calculating is wrong because the partial distances are computed in different spaces. In this paper, we present an extension to the C-means algorithm that solves this situation.

Another motivation for the algorithm is the necessity of many specialists working in soft sciences to group data into a specific number of clusters (solve a RUC problem). Generally, these specialists are interested in clusters such that objects that are more similar tend to fall into the same group while objects that are relatively distinct tend to separate into different groups. This is precisely the main characteristic of C-means algorithm.

2 Algorithm Description

Let us consider a set of m objects $\{O_1, O_2, ..., O_m\}$ which must be grouped in c clusters. Each object is described by a set $R=\{x_1, x_2, ..., x_n\}$ of features. The features take values in a set of admissible values $x_i(O_j) \in M_i$, $i=1,...,n$. We assume that in M_i there exists a symbol "?" to denote missing data. Thus, the features can be of any nature (qualitative: Boolean, multi-valued, etc. or quantitative: integer, real) and incomplete descriptions of the objects can be considered. For each feature a comparison criterion $C_i: M_i \times M_i \rightarrow L_i$ $i=1,\ldots,n$ is defined, where L_i is a totally ordered set, besides, Let $\Gamma: ()^2 \rightarrow [0,1]$ be a similarity function. In some cases the similarity function Γ depends on or is a lineal combination of partial similarity functions $\Gamma': ()^2 \rightarrow L'_i$, with L'_i the same as L_i. This function allows us to compare descriptions of objects with $s<n$. A non empty subset of features to analyze the sub-descriptions of objects is named the *support set*. A set consisting of support sets is named a *support sets system*.

Let be the similarity between the objects O_j and O_k. The value satisfies the following three conditions:

1. $\Gamma(O_j, O_k) \in [0,1]$ for $1 \le j \le m$ and $1 \le k \le m$;
2. $\Gamma(O_j, O_j) = 1$ for $1 \le j \le m$;
3. $\Gamma(O_j, O_k) = \Gamma(O_k, O_j)$ for $1 \le j \le m$ and $1 \le k \le m$.

Let u_{ik} the degree of membership of the object O_k in the cluster C_i, and let R^{exm} be the set of all real $c \times m$ matrices. Any c-partition (see, Ruspini, 1969) of the data set is represented by a matrix $U=[u_{ik}] \in R^{exm}$, which satisfies:

1. $u_{ik} \in \{0,1\}$ for $1 \le j \le m$ and $1 \le k \le m$;
2. $\sum_{i=1}^{c} u_{ik} = 1$ for $1 \le k \le m$;
3. $\sum_{k=1}^{m} u_{ik} > 0$ for $1 \le i \le c$.

We want to get clusters such that objects that are more similar tend to fall into the same cluster and clusters tend to be less similar among them. Then we have that, the partition matrix U is determined from maximization of the objective function given by $J(U) = \sum_{i=1}^{c} \sum_{k=1}^{m} u_{ik} \Gamma(O_i^r, O_k)$ where $\Gamma(O_i^r, O_k)$ is the similarity between the *representative object* O_i^r ("the center") in the cluster C_i and the object O_k. Note that in our case "the center" is an object of the sample instead of a fictitious element (centroid) as in the classical C-means algorithm.

Now, how to select the representative object?. A good representative object O_i^r will be that object O_j which in average is the most similar with objects in the same cluster; this property is reflected by the expression (2). Moreover, we want that the similarity between the objects in the cluster and the representative object is near to the average similarity of the representative object with the objects in the cluster (see (3)), in other words, the variance of similarity respect to the average similarity must be little. If the variance is little then the cluster will be more compact. At the same time, the representative objects must be the least similar with the other representative objects (see (4)). Therefore, in order to determine the representative object for C_i, given U, and taking account previous properties, we introduce the following expression.

$$r_{\substack{C_i \\ O_j \in C_i \\ C_q \neq C_i}}(O_j) = \frac{\beta_{C_i}(O_j)}{\left(\alpha_{C_i}(O_j) + \left(1 - \beta_{C_i}(O_j)\right)\right)} + \eta_{C_q}(O_j) \quad (1)$$

Where, $\beta_{C_i}(O_j)$ evaluates the average of similarity (mean) between the object O_j and the other objects in the same cluster C_i. And it is computed as follows

$$\beta_{C_i}(O_j) = \frac{1}{C_i - 1} \sum_{\substack{O_j, O_q \in C_i \\ O_j \neq O_q}} \Gamma(O_j, O_q) \quad (2)$$

To increase the informational value of (2) we introduce the expression $\alpha_{C_i}(O_j)$.

$$\alpha_{C_i}(O_j) = \frac{1}{C_i - 1} \sum_{\substack{O_j, O_q \in C_i \\ O_j \neq O_q}} \beta_{C_i}(O_j) - \Gamma(O_j, O_q) \quad (3)$$

This expression evaluates the difference (variance) between the mean (2) and the similarity between the object O_j and the other objects in C_i. Then when (3) decreases, the values of (1) increases.

The expression $\left(1-\beta_{C_i}\left(O_j\right)\right)$ represents the average of dissimilarity of O_j with respect to the other objects in C_i.

$$\eta_{C_k}\left(O_j\right)=\sum_{\substack{q=1\\ i\neq q}}^{c}\left(1-\Gamma\left(O_q^r,O_j\right)\right) \tag{4}$$

Finally, the function (4) evaluates the dissimilarity between the object O_j and the other representative objects. This function is used to diminish cases where there exist two objects with the same value in (1).

When $|C_i|=1$, then the representative object for the cluster C_i is the object containing this cluster.

Consequently, it is quite reasonable that the representative object for the cluster C_i is defined as the object O_r which yield the maximum of $r_{C_i}\left(O_j\right)$.

$$r_{C_i}\left(O_r\right)=\max_{O_p\in C_i}\left\{r_{C_i}\left(O_p\right)\right\} \tag{5}$$

If the cluster centers are given, the functional $J(U)$ is maximized when u_{ik} is determined as:

$$u_{ik}=\begin{cases}1 & \text{if } \Gamma\left(O_i^r,O_k\right)=\max\limits_{1\le q\le c}\left\{\Gamma\left(O_q^r,O_k\right)\right.\\ 0 & \text{otherwise}\end{cases} \tag{6}$$

That is to say, an object O_k will be assigned to the cluster such that O_k is the most similar with their representative object.

C-means algorithm using similarity functions (SF C-means)

Step 1. Fix c, $2 \le c \le n$. Fix the number of iterations ni', and $ni=0$.

Step 2. Select c objects in the data as initial seeds.

Step 3. Calculate the partition matrix $U=U^{(ni)}$ using (6).

Step 4. Determine the representative objects of the clusters for the matrix $U^{(ni)}$, using (1) and (5).

Step 5. If the set of representative objects is the same that in the previous iteration stop. Otherwise increase $ni=ni+1$.

Step 6. If $ni > ni'$ stop. Otherwise, go to step 3.

Type of initial seeds	Average effectiveness by cluster			Algorithm	Total average effectiveness
	Cluster1	Cluster2	Cluster3		
Random	87.6%	86.8%	80.6%	Classical C-means	85 %
Random	100%	98%	80.4%	SF C-means	92.8 %
Representative	100%	95%	72%	Classical C-means	89 %
Representative	100%	98%	86.8%	SF C-means	94.9 %

Table 1. Results of classical C-means and SF C-means on Iris data in 10 tests

Database	Objects	Quantitative features	Qualitative features	Clusters	Tests	Missing values	Average effectiveness
Iris	150	4	0	3	10	0	92.8%
Wine	178	13	0	3	10	0	79.5 %
Mushroom	8124	0	22	2	10	2480	84.8 %
Credit	690	6	9	2	10	67	67.4 %
Diabetes	768	8	0	2	10	0	56.2 %

Table 2. Results of SF C-means algorithm in 10 tests

Classical C-means	SF C-Means
It is metric	It is not metric
It is based on the Euclidean distance	It uses comparison criteria and function of similarity
It works only with quantitative descriptions	It works with mixed descriptions
It does not consider missing data	It considers missing data
It does not consider comparing sub descriptions	It considers comparing sub descriptions in base to a support set.

Table 3. Differences between the classical C-means and the SF C-means algorithms

3 Experimental Results

Initially, we effect a comparison between our extension and the classical C-means algorithm considering the Iris data (see http://www.ics.uci.edu/pub/machine-learning-databases/).

$$C_s(x_s(O_i), x_s(O_j)) = |x_s(O_i) - x_s(O_j)| \quad (7)$$

$$\Gamma(O_i, O_j) = \frac{\sum_{x_s \in R} C_s(x_s(O_i), x_s(O_j))}{|R|} \quad (8)$$

We applied the classical C-means algorithm using the Euclidean distance. To apply our algorithm we employed the equation (7) as comparison criterion between features' values. In addition, the equation (8) was used as similarity function between object descriptions. It is important to highlight that we use (7) and (8) as a theoretical exercise but in the practice should be employed functions that model the way in that the expert do the comparison. Our algorithm has this flexibility.

In the table 1 the results of applying the classical C-means and the SF C-means on Iris data are shown. We did ten tests selecting initial seeds randomly and the same amount selecting representative seeds. For us the *representative seeds* are objects that priori we know must be in a certain class or cluster. In this database, objects are described in terms of four quantitative features as can be seen in the following examples.

flower 5.1 3.5 1.4 0.2
flower^1 4.9 3.0 1.4 0.2
flower$^2_3$ 4.7 3.2 1.3 0.2
...

The objects were classified into three classes. In the table 1 appears both the average effectiveness by cluster and the total average effectiveness in the ten tests. Particularly in this experimentation our algorithm obtained a better percent of classification than the classical C-means algorithm.

Additionally, we test the SF C-means algorithm with Iris, Wine, Mushroom, Credit and Diabetes databases taken from http://www.ics.uci.edu/pub/machine-learning-databases/.

In the Wine database, objects are described in terms of thirteen quantitative features. Some examples of descriptions are the following.

O 14.23 1.71 2.43 15.6 127 2.80 3.06 0.28 2.29 5.64 1.04 3.92 1065
O' 13.20 1.78 2.14 11.2 100 2.65 2.76 0.26 1.28 4.38 1.05 3.40 1050
O^2_3 13.16 2.36 2.67 18.6 101 2.80 3.24 0.30 2.81 5.68 1.03 3.17 1185
...

The objects were classified into three classes. In the table 2 you can see the average effectiveness in ten tests.

The objects in Mushroom database are described in terms of twenty-two qualitative features. The following three descriptions are examples of mushroom descriptions.

M x s y t a f c b k e c s s w w p w o p n n g
M^2 b s w t l f c b n e c s s w w p w o p n n m
M^3_5 x s g f n f w b k t e s s w w p w o e n a g
...

In this example the objects were classified in two classes. In the table 2 you can see the average effectiveness in ten tests.

Other database analyzed was Credit; the objects in this database are described in terms of six quantitative features and nine qualitative features. A part of the Credit database is the following.

C b 30.83 0.00 u g w v 1.25 t t 1 f g 202 0
C' a 58.67 4.46 u g q h 3.04 t t 6 f g 43 560
C^2_3 a 24.50 0.50 u g q h 1.50 t f 0 f g 280 824
...

In this experimentation the objects were classified in two classes. In the table 2 you can see the average effectiveness in ten tests.

Finally we applied the algorithm to the Diabetes database, the objects in this database are described in terms of eight quantitative features. The following are examples of descriptions in Credit database.

P1 6 148 72 35 0 33.6 0.627 50
P2 1 85 66 29 0 26.6 0.351 31
P3 8 183 64 0 0 23.3 0.672 32
...

Objects are classified into two classes and in the table 2 you can see the average effectiveness in ten tests.

In all before databases we know a priori how the objects are classified; therefore, we can evaluate the percentage of correct classification of our algorithm. In Iris, Wine and Diabetes databases was handled (7) as comparison criterion for features' values. In the case of Credit and Mushroom databases was utilized (9) as comparison criterion for all the features. In these last databases, there are missing values; the comparison criterion (9) let us comparing features' values including missing values ("?"). The similarity function (8) was used for all databases. Remind that the comparison criteria, similarity function and the treatment of missing values must be modeled together with the practical specialists. Here we use (9) to manage missing values but in the practice, it must reflect the criterion of analogy employed by the expert.

$$C(x_s(O_i), x_s(O_j)) = \begin{cases} 1 & \text{if } x_s(O_i) = x_s(O_j) \vee \\ & x_s(O_i) = ? \vee x_s(O_j) = ? \\ 0 & otherwise \end{cases} \quad (9)$$

In the table 2 are presented the results that were obtained through the SF C-means algorithm using random initial seeds. In the column, named "average effectiveness", appears the average effectiveness of the ten tests. A better modeling of (7) and (8) could improve this percent.

Finally in the Table 3 are shown the main differences between the classical C-means algorithm and SF C-means.

4 Conclusions

A new C-means algorithm using similarity functions is proposed in this work. The algorithm considers descriptions of the objects with mixed data, i.e. quantitative and qualitative features. Also, the algorithm accommodates missing data. These characteristics allow to the algorithm be potentially useful in many problems of Data Mining and knowledge Discovery in Soft Sciences.

In comparison with the classical C-means algorithm, our algorithm presents on average a better classification using the Iris data. Besides, it allows us analyze objects described with qualitative and quantitative features and missing data. Therefore, the algorithm can be applied in soft sciences (such as: Medicine, Marketing, Geology, Sociology, etc.) where the specialists face this kind of descriptions.

The use of comparison criteria for the features and their integration in a similarity function give us flexibility to model more precisely a problem. In this way, the expert's knowledge in soft sciences can be put in computer systems to solve data analysis and classification problems.

5 Future Work

The C-means algorithm is an iterative algorithm, which bases its operation on initial seeds, so as future work we will propose a method to select candidates as initial seeds.

Another interesting future work is developing an optimal algorithm that can be applied to solve problems with big bulk of data.

Finally, an extension of our algorithm in the fuzzy case will be proposed in the future.

References

Duda Richard O. and Hart Peter E., *Pattern Classification and Scene Analysis*, John Wiley & Sons, USA, Inc., 1973.

Schalkoff Robert J., *Pattern Recognition: Statistical, Structural and Neuronal Approaches*, John Wiley & Sons, USA, Inc., 1992.

Ball G. and Hall D., "A Clustering technique for summarizing multivariate data", *Behav. Sci.*, 12, 1967, pp. 153-155.

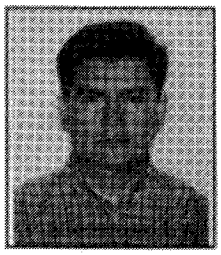
Bradley Paul and Fayyad Usama, "Refining Initial Points for K-Means Clustering", in *Proceedings of the Fifteenth International Conference on Machine Learning ICML98,* Morgan Kaufmann, San Francisco, 1998, pp. 91-99.

Dubes R. C., "How many clusters are best? -An experiment", *Pattern Recognition*, 20, 1987, pp. 645-663.

Bobrowsky Leon and Bezdek James C., "C-means clustering with the l_1 and l_∞ Norms", *IEEE Transactions on Systems man, and Cybernetics*, 21, 3, 1991, pp. 545-554.

Ralambondrainy H., "A conceptual version of the K-means algorithm". *Pattern Recognition Letters*, 16, 1995, pp. 1147-1157.

Ruspini, E. R, "A new approach to clustering". *Information and control*, 15, 1969, pp. 22-32.



José Francisco Martínez Trinidad *Received the B.S. Degree in Computer Science from Benemérita Universidad Autónoma de Puebla, Mexico in 1995, the M. Sc. Degree in Computer Science from Benemérita Universidad Autónoma de Puebla, Mexico in 1997 and the Ph. D. Degree in Computer Science from Centro de Investigación en Computación del IPN, Mexico in 2000. Currently is working at INAOE, Mexico, where he is a full time researcher. His research interests include Pattern Recognition, Conceptual Clustering, Symbolic Objects and Intelligent Data Analysis.*

Irene Olaya Ayaquica Martínez *Received the B.Sc. Degree in Computer Science from Benemérita Universidad Autónoma de Puebla, Mexico in 1998. She concluded her Master in Computer Science at Centro de Investigación en Computación of the IPN, Mexico. Currently is working in her M. Sc. thesis "Fuzzy C-means Algorithm using dissimilarity functions". Her research interests include Pattern Recognition, Data Mining and Unsupervised Clustering.*

Javier Raymundo García Serrano *Currently is working on his M. Sc. thesis "Classification of electric energy consuming through Logical Combinatorial Pattern Recognition" at CENIDET, Cuernavaca, México. His research interests include Pattern Recognition and Data Mining.*