An Evolutionary Algorithm for Automatic Recommendation of Clustering Methods and its Parameters

Jessica A. Carballido [1,2] Macarena A. Latini [1]
Ignacio Ponzoni [1] Rocío L. Cecchini [1]

Institute for Computer Science and Engineering (UNS–CONICET)
Department of Computer Science and Engineering
Universidad Nacional del Sur, Bahía Blanca, Argentina

Abstract

One of the main problems being faced at the time of performing data clustering consists in the deteremination of the best clustering method together with defining the ideal amount (k) of groups in which these data should be separated. In this paper, a preliminary approximation of a clustering recommender method is presented which, starting from a set of standardized data, suggests the best clustering strategy and also proposes an advisable k value. For this aim, the algorithm considers four indices for evaluating the final structure of clusters: Dunn, Silhouette, Widest Gap and Entropy. The prototype is implemented as a Genetic Algorithm in which individuals are possible configurations of the methods and their parameters. In this first prototype, the algorithm suggests between four partitioning methods namely K-means, PAM, CLARA and, Fanny. Also, the best set of parameters to execute the suggested method is obtained. The prototype was developed in an R environment, and its findings could be corroborated as consistent when compared with a combination of results provided by other methods with similar objectives. The idea of this prototype is to serve as the initial basis for a more complex framework that also incorporates the reduction of matrices with vast numbers of rows.

Keywords: evolutionary algorithms, partition clustering, clustering recommendation methods.

[1] Partially supported by CONICET (PIP 112-2012-0100471) and UNS (PGI 24/N042)
[2] Corresponding author email: jac@cs.uns.edu.ar

1 Introduction

This article presents a proposal, based on evolutionary algorithms, to solve one of the major known problems in the area of Clustering, which involves the identification of the most appropriate method to perform unsupervised clustering data and the estimation of its corresponding parameters [6]. Furthermore, the article intends to work as a basic guide of steps that should be followed when making Clustering of data, so initially, some elementary aspects are explained.

The clustering of data is the combinatorial optimization problem of finding classes of objects in such a way that those objects who belong to a group are more like each other than the similar they are to objects belonging to other groups. However, measuring the similarity among objects will depend on the types of clusters present in the data set on which they work, since data can be grouped into compact shapes, elongated or forming some kind of stroke within each cluster.

There are several issues that should be considered when choosing the strategy with which the grouping will be carried out. For example, whether it will be a strict grouping, in which case each element belongs to a single group. In turn, if a supervised method can be applied, for which it will be necessary to have labeled cases, or else if only an unsupervised method must be considered.

As a measure of general interpretation, in this article we will consider that each row of the data matrix to be processed is an *observation*, while each column is a *variable*. That is, if the data matrix is an $n \times m$ matrix, we will be working with n *observations* and m *variables*. It is important to bear in mind that the data must be standardized prior to the application of the clustering method, in order to make the variables comparable to each other. Another issue to contemplate is which metric to measure the distance will be used. The grouping of the observations will require some method that allows to evaluate the distance (similarity / dissimilarity) between them. Some clustering methods require a matrix of distances that can be built by using those distance metrics. Some of the most widely used distance metrics are Euclidean, Manhattan, Pearson, Kendall and Spearman [14,9,17]. Once the distance metric is defined, the distance matrix M_d, of $n \times n$, can be calculated when necessary, such that the component $M_d[i\,,\,j]$ indicates the distance between vector i and j of our original matrix. As expected, this matrix is symmetric and with zeros (or ones) on the diagonal.

The classic strategies to identify the characteristics of the underlying data groups within a particular data set, can be classified into hierarchical and non-hierarchical (or partitioning) methods [7]. The first ones allow to work with different types of variables, and are useful when the number of clusters is not previously known, as long as the data set is not very large. These algorithms can be, in turn, agglomerative or divisive. In the first case, the clusters are joined in order to obtain a smaller number

of groups at each stage. Meanwhile the divisive ones work in an inverse manner. In both cases some distance function is minimized (or maximized), for which the distance (or similarities) matrix is used. On the other hand, non-hierarchical methods, work trying to reach the best possible partition of the data for a given number of clusters (k). This number must be known prior to the execution of the algorithm. In general, these methods do not work on a distance matrix but on the original data. There is a great variety of non-hierarchical algorithms, among the most common we can mention K-means [11], K-medoids (PAM) [8], CLARA [10,17], DBSCAN [5].
It is expected that the found groups meet some basic characteristics, such as a greater density among the elements belonging to a group than among the ones belonging to different groups, and evaluating the goodness of a structure is not a trivial task. In addition, it must be considered that the quality of each group will be relative to the application or problem under study. There is a large number of indices that allow evaluating different aspects of the result of a clustering algorithm and, although they have been classified in several ways [1,17,2], the most widely used categorization classifies them into internal and external [1,17]. The main difference is whether the measure uses external information for validation or not, that is, information that is not a product of the clustering technique. In general, in one way or another, all internal measures seek to analyze two main characteristics of the structure: cohesion and separation. The first one looks for the member of each cluster to be as close as possible to the other members of the same cluster and the second aims to have widely separated clusters. The most used internal validation indices are *Dunn* [4,12] and *Silhouette* [12,15]. In this work, two additional indices, *Entropy* [13] and *Widest gap* [18,3], were chosen, which allow a more global analysis of the final result of the algorithm.

2 Proposed Algorithm

The evolutionary algorithm was implemented in R. After receiving an input data matrix, it proposes a clustering method and its corresponding parameters. Until now, the possible methods are K-means [11], PAM [8,10], CLARA[10,17] and Fanny [10], whose choice depends on the quality of the results obtained using them for the input matrix. The indices used to evaluate the quality of the result of the clustering method are: Dunn, Silhouette, Entropy and Widestgap.

Representation of the individuals. Each individual is represented with the triple $\langle MC,\ K,\ Algorithm \rangle$. MC is an integer that represents the clustering method (K-means, PAM, CLARA or Fanny). K is the number of clusters to be obtained (Fanny, CLARA) or a list of centroids (K-means) or a list of medoids (PAM)

depending on the value of *MC*. Finally, the *Algorithm* field saves, in the case of K-means, a reference to the name of the algorithm (Hartigan-Wong, Lloyd, Forgy or MacQueen), and in the other three cases (PAM, CLARA or Fanny) it keeps the method used to calculate the distance between two observations.
When creating the individual, it is essential to maintain consistency between the clustering method and its parameters. Therefore, the validation of the k value was carried out according to the restrictions of the corresponding method following these rules: the K-means, PAM, and CLARA methods require that $0 < k < n$. The Fanny method needs that $0 < k < (n/2) - 1$. These restrictions are considered throughout the algorithm to preserve the feasibility of the individuals. The initial population, conformed by 30 individuals, is created in a random manner respecting in each case the limits mentioned above, according to the parameters required by each method. The selection method used is the *binary tournament*.

Crossover. For the crossing of two individuals, parents are selected with a probability $PC = 0.7$, and they are crossed randomly using one of the following options:

- Option 1: The children inherit the value of k and receive the parents' clustering method.
- Option 2: The children inherit the father's clustering method and receive the value of k exchanged.

This process requires a correction of the value of k to preserve the property of feasible individuals mentioned above.

Mutation. To mutate an individual, we use a total replacement technique. The individual is selected with a probability $PM = 0.2$ and in his position a new individual is generated. This procedure exhibited better results than different strategies in which some parts of the individual where randomly changed. It is important to mention that mutation and crossover probabilities, and popultation size were tunned after several testing runs.

Fitness Function. The assessment of the fitness of a given individual is performed in two steps. In the first step, the method pointed by the individual is executed over the input data, using the parameters specified for it, for which the {stats} and {cluster} packages of R were used. Then, the function *cluster.stats* {fpc} is used to validate the result of the method. This function calculates several validity statistics for a clustering structure and a dissimilarity matrix. In this case, all the statistics returned by cluster.stats, the Dunn, Silhouette, Entropia and Widestgap index values were analyzed. The ultimate goal of the fitness function is to maximize the first two indices and minimize

the last two. Given that we are facing a problem with several objectives, the most direct way of joining them is through an aggregation function. It consists of combining all the objective functions $f_i(x)$ into a single function $F(f_1(x), \ldots, f_k(x))$. This first approach uses a linear aggregation of the objectives based on the following equation:

$$F = \sum_{i=1}^{k} w_i f_i(x) \tag{1}$$

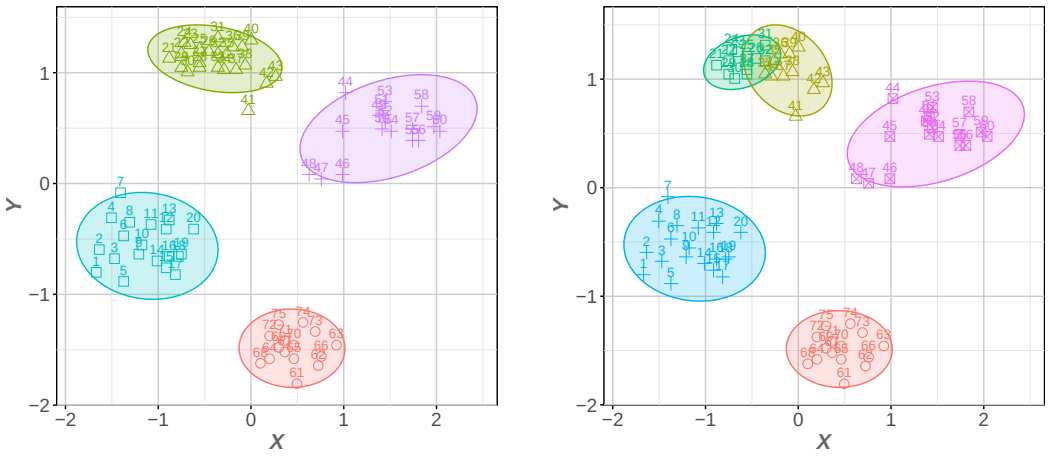
Where w_i are the weights of each objective function, being common to normalize them such that the sum of all the weights is equal to 1. In this work all the objectives have equal weights. The objectives to be maximized are added up and the rest are subtracted. More specifically, the objective functions were determined as: $f_1 = D$, $f_2 = S$, $f_3 = -H$ and $f_4 = -W_g$ based on the corresponding equations given by the methods Dunn (D), Silhouette (S), Entropy (H) and Widestgap (W_g).

3 Evaluation

To verify the performance of the algorithm, Ruspini was used [16]. This dataset constitutes a traditional example in the evaluation of clustering methods. It is composed of 75 observations on two variables, x and y. In Figure 1 you can see two possible clustering solutions found with different numbers of clusters. It can be seen that in the group of four clusters, the separation between clusters is visually recognizable, whereas as we move to the case of five groups, the interpretation and even the definition of the groups is less clear.

Experiments design. The experimentation was organized in 100 independent runs. For each run, the clustering configuration suggested by the best individual was recorded using the previously mentioned evaluation indices. Once the runs finished, the result was checked using the NbClust function of the homonymous package. This function uses 30 indices to determine the best number of clusters. However, unlike our method, it does not propose the algorithm that yields the best results. It only performs the analysis using a unique non-hierarchical algorithm, the k-means, and a hierarchical one, the HAC (Hierarchical Agglomerative Clustering), without giving the possibility of varying these methods.

Analysis of results. Table 1 shows a summary of the values obtained in 100 runs of the proposed algorithm. It should be noted that only one of the 100 times, the algorithm suggested a configuration with 2 clusters. The rest of the runs suggested 4 or 5 clusters. This constitutes a first relevant achievement



(a) Cluster plot for CLARA with $k = 4$. (b) Cluster plot for CLARA with $k = 5$.

Fig. 1. Possible groupings found for the *Ruspini* data set, for 4 and 5 groups respectivelly.

of our method. It is important to remake that our method decides when one configuration $\langle MC,\ K,\ Algorithm \rangle$ is better than another in terms of Dunn, Silhouette, Entropy and Widest gap indices. According to the results, the algorithm suggests that the best values for the pursued objectives are obtained using the CLARA method with $k = 5$. There were no important differences when looking at the frequences each metric (Euclidean or Manhattan) reaches the best cluster for this method. Besides, it can be seen that the PAM method with Euclidean distance also shows a good performance, for $k = 4$. At this point it becomes evident the importance of incorporating external evaluation measures that, in view of these results, help to complete the analysis. From the table, it can be seen that regarding the k value, the algorithm prefers predominatly structures with 4 clusters (68 times vs. 31 times). It should be noted that whenever $k = 4$ was suggested, the structure was identical for all cases, while for $k = 5$ there were different variants.

As aforementioned, the results of this case study were validated with the NbClust function. When invoking this function with the dataset of Ruspini and a variation of k from 2 to 8, the result obtained is that among all the indices: 1 proposes that the best number of clusters is 2, 3 propose that the best number of clusters is 3, 6 propose that the best number of clusters is 4, 1 proposes that the best number of clusters is 5, 2 propose that the best number of clusters is 8, and the conclusion is that "according to the majority rule, the best number of clusters is 4".

This result reveals two hints: the first is that the new method suggests the same number of clusters as the NbClust method, which is well known and

Table 1
Number of times that each possible combination of ⟨*MC, K, Algorithm*⟩ achieved the best performance. Where, H-W:Hartigan-Wong, L:Lloyd, F:Forgy, McQ:MacQueen, E:Euclídea, M:Manhattan.

	K-means				**PAM**		Fanny		**CLARA**		
	H-W	L	F	McQ	E	M	E	M	E	M	
k=4	5	1	3		14	8	9	8	**8**	**12**	68
k=5					5				**14**	**12**	31
	9				27		17		46		

widely used in the literature. The second is that we correctly have chosen four indices that summarize the desirable characteristics of a cluster structure. On the other hand, regarding the suggested clustering method, it cannot be fairly compared since there is not, as far as we know, an algorithm whose objective is also to propose the most appropriate clustering method for a given structure.

4 Conclusions

In this article, we present a new evolutionary algorithm that takes as input a data matrix and returns the best partition clustering method and its corresponding parameters. The individuals represent different configurations of clustering methods, parameters and values of k. The algorithm was validated with the Ruspini dataset, which is widely used in the clustering testing bibliography. The partition methods among which our algorithm suggests the best performer are K-means, CLARA, PAM and Fanny. To select the best configuration of method/parameters/k, we use Dunn, Silhouette, Entropy and Widest gap internal validation indices. The NbClust package of R was used to validate the results according to those indices. After 100 independent runs of our algorithm, we could verify that the new proposal suggests the best configuration for the Ruspini dataset. Therefore we consider that the method presented in this paper constitutes a good preliminary point for a future implementation in which, taking this prototype as a basis, we will add the possibility to evaluate and suggest different reductions of the input matrix. We also plan to include more clustering methods and, as an additional aim, the crossing and mutation operators of the genetic algorithm will be optimized.

References

[1] Aggarwal, C. C. and C. K. Reddy, "Data Clustering: Algorithms and Applications," Chapman & Hall/CRC, 2013, 1st edition.

[2] Brock, G., V. Pihur, S. Datta and S. Datta, *clvalid: An R package for cluster validation*, Journal of Statistical Software, Articles **25** (2008), pp. 1–22.

[3] Christian, H., *Cluster validation by measurement of clustering characteristics relevant to the user* (2017), arXiv:1703.09282v1.

[4] Dunn, J. C., *Well-separated clusters and optimal fuzzy partitions*, Journal of Cybernetics **4** (1974), pp. 95–104.

[5] Ester, M., H. peter Kriegel, J. Sander and X. Xu, *A density-based algorithm for discovering clusters in large spatial databases with noise*, in: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining* (1996), pp. 226–231.

[6] Fraley, C. and A. E. Raftery, *How many clusters? which clustering method? answers via model-based cluster analysis*, The Computer Journal **41** (1998), pp. 578–588.

[7] Jain, A. K., M. N. Murty and P. J. Flynn, *Data clustering: A review*, ACM Comput. Surv. **31** (1999), pp. 264–323.

[8] Kaufman, L. and P. J. Rousseeuw, *Clustering by means of medoids* (1987).

[9] Kendall, M. G., "Rank Correlation Methods," Griffin, London, England, 1970.

[10] Leonard Kaufman, P. J. R., "Finding groups in data: an introduction to cluster analysis," Wiley-Interscience, 1990, 9th edition.

[11] MacQueen, J., *Some methods for classification and analysis of multivariate observations*, in: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics* (1967), pp. 281–297.

[12] Maulik, U., S. Bandyopadhyay and A. Mukhopadhyay, "Multiobjective Genetic Algorithms for Clustering - Applications in Data Mining and Bioinformatics." Springer, 2011, I-XVI, 1-281 pp.

[13] Meilă, M., *Comparing clusteringsan information based distance*, Journal of Multivariate Analysis **98** (2007), pp. 873 – 895.

[14] Pearson, K., *Notes on the history of correlation*, Biometrika **13** (1920), pp. 25–45.

[15] Rousseeuw, P. J., *Silhouettes: A graphical aid to the interpretation and validation of cluster analysis*, Journal of Computational and Applied Mathematics **20** (1987), pp. 53 – 65.

[16] Ruspini, E. H., *Numerical methods for fuzzy clustering*, Inf. Sci. **2** (1970), pp. 319–350.

[17] Theodoridis, S. and K. Koutroumbas, "Pattern Recognition, Fourth Edition," Academic Press, 2009.

[18] Villanueva, B. S., K. Gibert and M. Sánchez-Marré, *Using CVI for understanding class topology in unsupervised scenarios.*, in: O. Luaces, J. A. Gmez, E. Barrenechea, A. Troncoso, M. Galar, H. Quintin and E. Corchado, editors, *CAEPIA*, Lecture Notes in Computer Science **9868** (2016), pp. 135–149.