

Clustering Large Categorical Data Set via Genetic Algorithms

Algoritmi genetici per il raggruppamento di dati qualitativi

Raffaella Piccarreta
Istituto di Metodi Quantitativi
Università Bocconi, Milano
raffaella.piccarreta@uni-bocconi.it

Riassunto: Illustriamo due estensioni – proposte da Huang (1998) e da Kauffman e Rousseeuw (1990) – del noto algoritmo di raggruppamento a centri mobili al caso in cui si considerino variabili qualitative. Proponiamo una nuova metodologia di raggruppamento basata sulla logica degli algoritmi genetici. L'applicazione di tali algoritmi ad un data set reale e l'analisi dei risultati ottenuti evidenziano anche in questo caso, come già nel caso quantitativo, l'opportunità del loro utilizzo.

Parole chiave: clustering, categorical variables, genetic algorithms, *K*-means algorithm

1. Introduction

Cluster analysis is one of the most important operations for dealing with large data sets. The partition of objects into homogeneous clusters, which can, if desired, be analyzed separately, results, in fact, in a considerable simplification of analysis. The aim of this work is to present techniques to cluster large data sets in the case when measurements are available on Q categorical variables, $X_1, X_2, \dots, X_Q$, each describing a characteristic. Clustering can be performed either referring to hierarchical procedures or to non hierarchical procedures. Hierarchical procedures require as a preliminary step the definition of a dissimilarity matrix, $D = [d_{ij}]$, d_{ij} being a measure of the dissimilarity between the i -th and the j -th object. As regards to the categorical case, it is well known (see e.g. Kauffman and Rousseeuw, 1990) that there are different ways to properly measure the dissimilarity between two objects. In the case when the number of objects to be clustered is very high, hierarchical procedures are not efficient. Non hierarchical procedures are then preferred, such as, for example, the well known *K-means* algorithm (MacQueen, 1967) which is efficient in processing large data sets, and often leads to a local optimum solution (MacQueen, 1967). To improve results obtained by referring to the “classical” version of the algorithm, it is possible to refer to modern optimization techniques, such as, for example, *genetic algorithms* (Murthy and Chowdhury, 1996; Paterlini et al., 2001).

A first possible way to apply the *K-means* algorithm to cluster N objects on the basis of Q categorical variables is to convert multiple category attributes into binary attributes, each measuring the presence (1) or the absence (0) of a certain characteristic. This procedure can become inefficient in cases when the number of levels of the involved variables is very high.

In this work we will thus refer to two proposals, the first introduced by Kauffman and Rousseeuw (1990), the second introduced by Huang (1998), both attempting to extend

the K -means algorithm to the “categorical case”, still avoiding the transformation of the categorical problem into a quantitative one. More precisely, our aim is to improve results obtained by referring to these techniques by means of genetic algorithms.

The paper is organized as follows: Section 2 is devoted to the description of the two above mentioned techniques, and to the presentation of the genetic approach. In Section 3 an application to real data is illustrated.

2. K -means algorithms for categorical data

It is worthwhile recalling that the K -means algorithm consists of the following steps: (1) selection of the initial K seeds (2) calculation of the dissimilarity between each observation and the seeds (3) allocation of each observation to the seed it is nearest to (4) synthesis of the observations in each cluster: the k means of the obtained clusters constitute now the K seeds. Steps (2)-(4) are repeated until the algorithm converges.

The main features of Huang’s (1998) algorithm (from now onwards *HA*) concern steps (2) and (4). The dissimilarity measure referred to is the well known simple matching coefficient: in particular the distance between the i -th object, characterized by $\mathbf{x}_i = (x_{i1}, \dots, x_{iQ})$ and the j -th object, characterized by $\mathbf{x}_j = (x_{j1}, \dots, x_{jQ})$ is $d_{ij} = d(\mathbf{x}_i, \mathbf{x}_j) = \sum_q \delta(x_{iq}, x_{jq})$, with $\delta(x_{iq}, x_{jq}) = 0$ if $x_{iq} = x_{jq}$, 1 otherwise. The second modification

concerns the synthesis of the observations in each cluster. Instead of the means, the *modes* of the clusters are considered: each variable is thus synthesized by the most frequent category (notice that a set of data can be characterized by more modes). An important consideration has to be made with reference to step (1), concerning the choice of the initial seeds: in absence of prior information, K observations are randomly selected to constitute the seeds. The algorithm evidently depends upon the choice of the initial seeds. Notice that this dependence can be more crucial than in the quantitative case, the mode being in a sense less flexible than the mean in synthesizing a variable. Actually, Huang proposes a method to differently select the initial seeds. The method consists in assigning the most frequent categories of the different variables equally to the initial seeds, obtaining thus a set of (artificial) representative seeds. Subsequently, the object most similar to each seed is individualized and substituted to the seed itself. At least in our opinion, this way to proceed (which can not be performed automatically) is very hard to implement in the case when many variables are taken into account.

As concerns the *partitioning around medoids* algorithm (in the following *KR1*) introduced by Kauffman and Rousseeuw (1990), it is based upon the search of K representative objects, called *medoids*, among the objects of the data set. *KR1* proceeds as follows: in a first phase, K representative objects are found; subsequently, K clusters are formed by assigning each observation to the medoid it is nearest too (dissimilarity between two objects being measured again by the simple matching coefficient). As it is evident, the main problem to be addressed in this case concerns the selection of the medoids. Kauffman and Rousseeuw (1990) propose to this end an algorithm which can be implemented when the number of observations is limited. In cases when the number of objects to be clustered is high - like the one considered here - they propose a modification of the algorithm (in the following *KR2*): a sample of cases is firstly drawn and clustered into K groups using the K -medoids method. The objects not in the sample are then assigned to the nearest medoid (obtained on the basis of the sample). This procedure is then repeated for t times (the authors propose to set $t = 5$) and the best

clustering is then chosen. The goodness of the clustering is evaluated by referring to the total within dissimilarity. It is evident that in this case results strongly depend upon the drawn sub-samples.

Our aim in this work is to analyze the performance of genetic approaches (Mitchell, 1996) in improving results obtained by applying the algorithms described above. Genetic algorithms are used to solve optimization problems. At the first step of the algorithm an artificial population of candidate states is created. Each candidate state represents the possible solution to the considered optimization problem: its characteristics as a solver of the problem are embodied in its *genetic code*, and its performance with reference to the problem itself is evaluated by means of the so called *fitness function*. The artificial population is forced to evolve according to some genetic operators (such as selection, crossover, mutation). More precisely, some states of the current population are selected with probability proportional to their fitness function and randomly exposed to random mutations; new individuals are thus created, worse solutions are suppressed and the current population evolves to a new generation possibly constituted by better individuals (solutions). A crucial point in the definition of a genetic algorithm is the choice of the *genetic code* to attribute to a possible solution. In this work we adopt two coding schema. In the first algorithm, *GA1*, the artificial individual is characterized by a string of K cells. Each cell contains the identification number of an observation selected to be the *initial seed* in *HA* or the *medoid* in *KR1* and *KR2*. In the second algorithm, *GA2*, the artificial individual is characterized by a string of $Q \times K$ cells. Each group of Q cells represents an *initial seed* in *HA*: more precisely, the q -th cell contains one out of the categories of the q -th variable (randomly selected). This algorithm was defined by adapting to the categorical case the algorithm recently proposed by Paterlini et al. (2001) to improve the performance of the K -means algorithm in the quantitative case. Both the algorithms use the same genetic operators. As regards to the fitness function, we considered the *total within dissimilarity*, defined as $W = \sum_{k=1}^K \sum_{i=1}^{n_k} d(x_{ik}, s_k)$, s_k denoting the “representative individual” for the k -th cluster (a medoid or a vector of modes).

3. Empirical results

In this section we illustrate results obtained by applying the algorithms described in the previous section to a real data set containing different information about 1038 investors in the Italian market (portfolio choice, satisfaction about the performance of the chosen investment, socio-demographic information, and so on). The variables taken into consideration are $Q = 23$, and some variables (for example profession) are characterized by a high number of categories. In Table 1 we present results obtained in the case when $K = 10$. In the table, W represents the total within dissimilarity, while $T = \sum_{i=1}^N d(x_i, s)$, where N is the sample size and s denotes the vector of marginal modes in *HA* and the “central” observation (the observation which is most similar to all the others) in *KR*.

As is evident from the Table, the application of genetic algorithms permits us to improve the performances of the original algorithms. As regards to the algorithms introduced by Kauffman and Rousseeuw, notice that *KR1* is characterized (as expected) by the best performance; when algorithm *KR2* is used, a worse performance is observed.

The genetic algorithm improves results obtained by *KR2* and constitutes thus a valid alternative to the algorithm in the case when sample size is high, so that *KR1* can not be used.

Table 1. Fitness values for different algorithms

Method	W	1 - W/T
HA	5352	0.45988
GA1 for HA	5038	0.49157
GA2 for HA	5024	0.49299
KR1	5124	0.50363
KR2	5226	0.49375
GA1 for KR2	5134	0.50266

The illustrated results show the good performance of genetic algorithms even in cases when they are applied to the problem of clustering a large categorical data set by means of a properly defined extension of the *K*-means algorithm.

In the presented example, the number of clusters is considered as fixed, but, obviously, the genetic algorithm can be extended to simultaneously determine the best number of clusters and the best clustering. As a second, important point, the genetic algorithm can also be modified to automatically select either a sub-set of variables to be referred to in the clustering problem (a very important problem in data mining) or, with regard to categorical variables assuming a large number of levels, to simultaneously determine sub-sets of variables and of categories to be retained in analysis. This can be very important to evidence the source of information which is in effect relevant in determining clusters.

References

- Baragona R., Calzini C., Battaglia, F. (2001) Genetic algorithms and clustering: an application to Fisher's Iris Data, in S. Borra et al., eds., *Advances in Classification and Data Analysis*, Springer-Verlag, Berlin/Heidelberg.
- Huang Z. (1998) A Fast Clustering Algorithm to Cluster Very Large Categorical Data Sets in Data Mining
- Kauffman L., Rousseeuw P.J. (1990) *Finding Groups in Data*, Wiley & Sons, NY.
- Falkenauer E. (1998) *Genetic Algorithms and Grouping Problems*, Wiley & Sons, Chichester.
- MacQueen J.B. (1967), Some Methods for Classification and Analysis of Multivariate Observations, In *proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, pp. 281-297.
- Mitchell M. (1996) *An Introduction to Genetic Algorithms*, M.I.T. Press, Cambridge, Mass.
- Murthy C.A., Chowdhury N. (1996) In Search of Optimal Clusters Using Genetic Algorithms, *Pattern Recognition Letters*, 17, 825-832
- Paterlini S., Favaro S., Minerva T. (2001) Genetic approaches for Data Clustering Using Partial Information, *Book of short papers, Cladag 2001*, Palermo, 33-36.