CLUSTERING BY MEANS OF MEDOIDS

Leonard KAUFMAN

Vrije Universiteit Brussel
Dienst FABI
Laarbeeklaan 103
B-1090 Brussels
Belgium

Peter J. ROUSSEEUW

Delft Univ. of Technology
Dept. of Math. and Inf.
Julianalaan 132
2628 BL Delft
The Netherlands

When partitioning a set of objects into k clusters the main objective is to find clusters, the objects of which show a high degree of similarity. There are several possible criteria for quantifying this objective, the best known of which are based on sums of squares. An alternative approach, used in the k-medoid method, is of the L_1 type. It searches for k "representative" objects, called medoids, which minimize the average dissimilarity of all objects of the data set to the nearest medoid. A cluster is then defined as the set of objects which have been assigned to the same medoid. An algorithm for this method has been implemented in program PAM which is described in the paper. The k-medoid method offers the advantage of not requiring the actual measurement data as it can also be applied to a collection of dissimilarities. Such dissimilarities are for example subjective assessments of relationships between objects, in which case no measurements exist. An example of the latter type is analyzed. Furthermore, the k-medoid method is less susceptible to outlying values.

1. INTRODUCTION

When partitioning a set of objects into k clusters the main objective is to find clusters, the objects of which show a high degree of similarity, while objects belonging to different clusters are as dissimilar as possible. The model described in this paper is based upon the search for k representative objects, which should represent the various aspects of the structure of the data. In this model the representative object of a cluster is the object for which the average dissimilarity (or equivalently the total dissimilarity) to all the objects of the cluster is minimal. This object is called the medoid of the cluster and the model is called the k-medoid model. (In the literature one also encounters the name k-median which to us seems less appropriate, as confusion can arise with the classical notion of median).

When constructing partitions with a fixed number k of clusters, it is often assumed that there exists a function which measures the quality of a clustering. This is also the case for the k-medoid model, which is based upon a location model with the following general formulation : "Given a finite number of users, whose demands for some service are known and must be satisfied, and given a finite set of possible locations among which k must be chosen for the location of service centers, select the locations in such a way as to minimize the total distance travelled by the users". In the formulation used in clustering, the sets of users and of possible locations coincide, and both correspond to the set of objects to be clustered. The location of a center is interpreted as the selection of an object as a representative object (or centrotype, median or medoid) of a cluster. The distance travelled by a user corresponds to the dissimilarity between an object and the medoid of the cluster to which it belongs. The idea to use this model for cluster analysis was introduced by Vinod (1969) and later also discussed by Rao (1971), Church (1978) and Mulvey and Cowder (1979).

In the mathematical formulation of the k-medoid model the set of objects is denoted by X :

$$X = \{x_1, x_2, \ldots , x_n\}$$

and the dissimilarity between objects x_i and x_j (also called objects i and j) is denoted by d(i,j).

A solution of the model is determined by two types of decisions :

a. the selection of objects as representative objects in clusters :
   y_i is defined as a zero-one variable, equal to one if and only if object i (i=1,2,...,n) is selected as a representative object

b. the assignment of each object j to one of the selected representative objects : z_{ij} is a zero-one variable, equal to one if and only if object j is assigned to the cluster of which i is the representative object (and also the medoid).

The corresponding optimization model, which was first proposed by Vinod (1969), can then be written as :

$$\text{minimize} \quad \sum_{i=1}^{n} \sum_{j=1}^{n} d(i,j)\, z_{ij} \tag{1}$$

subject to

$$\sum_{i=1}^{n} z_{ij} = 1 \qquad j = 1,2,\ldots,n \qquad (2)$$

$$z_{ij} \leq y_i \qquad i,j = 1,2,\ldots,n \qquad (3)$$

$$\sum_{i=1}^{n} y_i = k \qquad k = \text{number of clusters} \qquad (4)$$

$$y_i, z_{ij} \in \{0,1\} \qquad i,j = 1,2,\ldots,n. \qquad (5)$$

This is called a zero-one linear program. Constraints (2) express that each object j must be assigned to a single representative object. They imply (together with constraints (5)) that for a given j one of the z_{ij} is equal to one and all others are zero. Constraints (3) ensure that an object j can only be assigned to an object i if this last object has been selected as a representative object. Indeed, if this is not the case then y_i is zero and the constraint (together with constraints (5)) implies that all z_{ij} are zero. If i is a representative object, then all the z_{ij} (for this i) can be either zero or one. Equation (4) expresses that exactly k objects are to be chosen as representative objects. As the clusters are formed by assigning each object to the most similar representative object, there will be exactly k non-empty clusters. (In case of ties, the object is assigned to the representative object which was entered first.) Equation (2) implies that the dissimilarity between an object j and its representative object is given by

$$\sum_{i=1}^{n} d(i,j)\, z_{ij} \,.$$

As all objects must be assigned, the total dissimilarity is given by

$$\sum_{j=1}^{n} \sum_{i=1}^{n} d(i,j)\, z_{ij} \qquad (6)$$

which is the function to be minimized in the model. In Section 3 an algorithm for the k-medoid model is discussed.

Usually one does not know the number of clusters present in a data set. As most partitioning methods provide a fixed number k of clusters one must apply them for several values of k in order to find the most meaningful clustering. A possible way of selecting a value of k is by means of the silhouette coefficient (see Rousseeuw 1985). Another approach is to search for sets of objects which persistently stay together in the clusterings obtained for several values of k. This approach has been investigated by Massart et al. (1983) and Plastria (1986). Another method, based upon the bootstrap technique, was proposed by Moreau (1986).

In the k-medoid method each cluster is characterized by a centrally located _object_. In case the objects are defined by measurement values there exists an alternative way of characterizing a cluster, namely by its _centroid_. This approach has been used extensively in the literature as a basis for partitioning algorithms. The centroid of a cluster is defined as a point in p-dimensional space found by averaging the measurement values along each dimension :

$$\bar{x}_j(u) = \frac{1}{n_u} \sum_{i \in C_u} x_{ij}$$

where C_u represents the set of indices of cluster u which contains n_u objects, and $\bar{x}(u) = (\bar{x}_1(u), \bar{x}_2(u), \ldots , \bar{x}_p(u))$. We note two important facts : the centroid does not have to be one of the objects in the original data set, and it cannot be defined when the data is a set of dissimilarities not based on interval scaled measurement values.

A much used measure for the tightness of a cluster, based upon its centroid, is the error sum of squares, defined as the sum of squares of (Euclidean) distances between the objects of a cluster and its centroid :

$$ESS(C_u) = \sum_{i \in C_u} \sum_{j=1}^{p} (x_{ij} - \bar{x}_j(u))^2 . \qquad (7)$$

The error sum of squares of the whole clustering is

$$ESS = \sum_{u=1}^{k} ESS(C_u) \ . \qquad (8)$$

Therefore a possible approach to the clustering problem is to look for a partition into k non-empty subsets which minimizes the error sum of squares. Methods which try to achieve this aim are called <u>variance minimization techniques</u>. Observe that a similar objective could be attained in a variant of the k-medoid model, by using the <u>squares</u> of the dissimilarities in the function to be minimized (eqn. 1).

The next section discusses the types of data that can be used in the k-medoid model. Section 3 describes an algorithm and a computer program and in Section 4 the results of an example are given. Finally, in Section 5 some advantages of the k-medoid method are considered.

2. TYPES OF DATA

There are basically two data structures used in cluster analysis. The most common is a matrix of measurement values. The rows of this matrix represent the objects, and the columns correspond to variables.

Alternatively the program can be used by entering a matrix of dissimilarities between objects. Such dissimilarities can be obtained in several ways. Often they are computed from variables, which are not necessarily on an interval scale but which may also be binary, ordinal or nominal. A measure used for computing dissimilarities for mixed data sets was proposed by Gower (1971). A program for this purpose, called DAISY, is discussed by Kaufman and Rousseeuw (1987). It also happens that dissimilarities are given directly, without resorting to any measurement values. Note that such dissimilarities do not have to satisfy the triangle inequality $d(i,h) \leqslant d(i,j) + d(j,h)$.

Let us now consider an example in which the basic data consists of a dissimilarity matrix. A questionnaire was distributed in an economics postgraduate class, asking the students to provide subjective dissimilarity coefficients between eleven scientific disciplines. These coefficients had to be given as integer numbers on a scale from 0 (identical) to 10 (very different). The final dissimilarity coefficients, given in Table 1, were obtained by taking the averages of the coefficients given by the students.

TABLE 1
Subjective dissimilarities between 11 sciences

Astronomy	0.00										
Biology	7.86	0.00									
Chemistry	6.50	2.93	0.00								
Computer Science	5.00	6.86	6.50	0.00							
Economics	8.00	8.14	8.21	4.79	0.00						
Geography	4.29	7.00	7.64	7.71	5.93	0.00					
History	8.07	8.14	8.71	8.57	5.86	3.86	0.00				
Mathematics	3.64	7.14	4.43	1.43	3.57	7.07	9.07	0.00			
Medicine	8.21	2.50	2.93	6.36	8.43	7.86	8.43	6.29	0.00		
Physics	2.71	5.21	4.57	4.21	8.36	7.29	8.64	2.21	5.07	0.00	
Psychology	9.36	5.57	7.29	7.21	6.86	8.29	7.64	8.71	3.79	8.64	0.00

3. THE PROGRAM PAM

The k-medoid method can be applied by using program PAM (from "Partitioning Around Medoids") . This program is written in FORTRAN, operates entirely interactively, and runs on an IBM PC,XT,AT or compatible computer.

In program PAM the user starts by selecting one of the two types of data that can be used for the clustering algorithm. If he chooses measurements, the program will itself compute the dissimilarities before performing the clustering. This results in several additional options concerning the treatment of the measurements. The most important of these are the possibility of standardizing the measurements and the choice between Euclidean and Manhattan distance.

In the other situation, the clustering data consists of a set of dissimilarities. These are usually given in a matrix like Table 1. Because of symmetry, only the lower triangular part of the n by n matrix of dissimilarities is needed. We chose to read the lower triangular matrix instead of the upper one because this makes it very easy to add objects to the data set, by simply adding lines at the end of the data file.

In some applications one starts with similarities (which become larger when the objects are closer to each other) instead of dissimilarities. These cannot be used as input to the program but must first be converted to dissimilarities.

The algorithm we are using in PAM consists of two phases. In a first phase, called BUILD, an initial clustering is obtained by the successive selection of representative objects until k objects have been found. The first object is the one for which the sum of the dissimilarities to all other objects is as small as possible. This object is the most centrally located in the set of objects. Subsequently, at each iteration, another object is selected which decreases the sum (over all objects) of the dissimilarities to the most similar selected object, as much as possible. The process is continued until k objects have been found.

In the second phase of the algorithm (called SWAP), it is attempted to improve the set of representative objects and therefore also to improve the clustering yielded by this set. This is done by considering all pairs of objects (i,h) for which object i has been selected and object h has not. It is determined what effect is obtained when a swap is carried out; i.e. when object i is no longer selected as a representative object but object h is. The most profitable pair is then indeed swapped. This procedure is iterated until no further reduction is possible. As all potential swaps are considered, the results of the algorithm do not depend upon the order of the objects in the input file (except in case some of the distances between objects are tied). For more details see Kaufman and Rousseeuw (1987).

The algorithm used in program PAM yields a good but not necessarily optimal solution to model (1)-(5). A branch and bound method which always yields an optimal solution was discussed in Massart et al.(1983). Unfortunately this approach is only computationally feasible for relatively small problems with up to approximately 50 or 60 objects. Another exact method was used by Klastorin (1985) in a comparison of several clustering algorithms. It is based on an algorithm proposed by Erlenkotter (1978) for the simple plant location problem.

For very large data sets (with more than 400 or 500 objects), even the heuristic algorithm of program PAM becomes impossible to use. The same is true for most other algorithms found in the literature. One of the reasons for this is the necessity to store and frequently retrieve the n(n-1)/2 dissimilarities between the n objects of the data set. Another obstacle to solving large problems is the considerable computation time necessary for solving such problems. Based upon the algorithm used in program PAM, and using the same objective, a method was proposed which proceeds by analyzing repeated samples of objects (see Kaufman and Rousseeuw 1986).

4. AN EXAMPLE OF L_1 ANALYSIS

Figure 1 contains the first part of the output and also the clustering results for a single cluster (selection of one representative object). An interesting feature is that mathematics is found as medoid for the entire data set.

```
            *********************************
            *                               *
            *  PARTITIONING AROUND MEDOIDS  *
            *                               *
            *********************************

  TITLE   : science data

  DATA SPECIFICATIONS AND CHOSEN OPTIONS
  --------------------------------------
   THERE ARE   11 OBJECTS
   LABELS OF OBJECTS ARE READ
   INPUT OF DISSIMILARITIES
   SMALL OUTPUT IS WANTED
   NO GRAPHICAL OUTPUT IS WANTED
   CLUSTERINGS ARE CARRIED OUT IN    1 TO    3 CLUSTERS
   THE DISSIMILARITIES WILL BE READ IN FREE FORMAT
   THE DATA WILL BE SAVED ON FILE : b:science.dat

  **********************************************
  *  NUMBER OF REPRESENTATIVE OBJECTS     1    *
  **********************************************

  FINAL RESULTS
    AVERAGE DISSIMILARITY =        4.869

    CLUSTERS
    NUMBER  MEDOID   SIZE      OBJECTS

      1       mat      11     ast bio che com eco geo his mat
                              med phy psy

   DIAMETER OF EACH CLUSTER
        9.36

   AVERAGE DISSIMILARITY TO EACH MEDOID
        4.87

   MAXIMUM DISSIMILARITY TO EACH MEDOID
        9.07
```

FIGURE 1
First part of the output of the example

Further results given are the diameter of the cluster, and the average and maximum dissimilarity to the medoid. The respective values of these cluster characteristics are 9.36 , 4.87 and 9.07. Two of these values (diameter and maximum dissimilarity) can be found in Table 1 in which they have been encircled.

In Figure 2 the output is given for k equals 3. The clustering obtained corresponds rather well to three groups of scientific disciplines: exact sciences (including economics), biomedical sciences (with chemistry) and liberal arts (although this last characterization is arguable). Remarkable is that cluster 3, consisting of geography and history, is the only L* cluster (meaning that the diameter of this cluster is less than its separation, which is the smallest dissimilarity of an object of the cluster to an object of any other cluster). Geography and history are perceived as being quite similar (within the limited set of disciplines we considered).

5. ADVANTAGES OF THE APPROACH

The objective chosen in the k-medoid model is appealing as it is more robust than the error sum of squares employed in most methods. Furthermore it allows a good characterization of all clusters which are not too elongated and still makes it possible to isolate outliers in most situations.

The choice of the k-medoid model shows our preference for methods based upon average dissimilarities instead of sums of squares of dissimilarities. Indeed, the k-medoid model minimizes the average dissimilarity of the objects to the representative objects they are assigned to. In many branches of univariate and multivariate statistics it has been known for a long time that methods based on the minimization of sums (or averages) of dissimilarities or absolute residuals (the so-called L_1 methods) are much more robust than methods based on sums of squares (which are called L_2 methods). The computational simplicity of many of the latter methods does not make up for the fact that they are extremely sensitive to the effect of one or more outliers. (The effect of error perturbation on clustering algorithms was already examined by Milligan (1980), but his study did not yet contain the k-medoid method.)

Also note that the clustering found by PAM does not depend on the order in which the objects are presented (except when equivalent solutions exist, which very rarely occurs in

```
**********************************************
*                                            *
*  NUMBER OF REPRESENTATIVE OBJECTS     3    *
*                                            *
**********************************************


RESULT OF BUILD
  AVERAGE DISSIMILARITY =         2.175

FINAL RESULTS
  AVERAGE DISSIMILARITY =         2.175

  CLUSTERS
  NUMBER  MEDOID   SIZE      OBJECTS

    1       mat       5     ast com eco mat phy

    2       med       4     bio che med psy

    3       geo       2     geo his


CLUSTERING VECTOR
*****************

             1  2  2  1  1  3  3  1  2  1  2


CLUSTERING CHARACTERISTICS
**************************

 CLUSTER    3 IS ISOLATED,
        WITH DIAMETER  =    3.86 AND SEPARATION =    4.29
        THEREFORE IT IS AN L*-CLUSTER.

 THE NUMBER OF ISOLATED CLUSTERS =    1


 DIAMETER OF EACH CLUSTER
      8.36      7.29      3.86

 SEPARATION OF EACH CLUSTER
      4.29      4.43      4.29

 AVERAGE DISSIMILARITY TO EACH MEDOID
      2.17      2.31      1.93

 MAXIMUM DISSIMILARITY TO EACH MEDOID
      3.64      3.79      3.86


The output is on file : b:science.res
```

FIGURE 2
Output of the example for k=3

practice). This is not the case for many other algorithms. Also, if we ask for k clusters we do obtain exactly k clusters, and not less.

The L_1 method described here is invariant with respect to translations and orthogonal transformations of the data points, but not with respect to affine transformations which change the inter-object distances. However, we wanted to construct a method able to deal with general dissimilarity data (like the subjective assessments of Table 1 above) to which the notion of affine invariance does not apply. On the other hand, the affine invariant clustering methods in the literature make use of geometric notions (like centroids or tolerance ellipsoids) which do not exist for dissimilarity data. The medoids considered in this paper always exist, even when the data is a collection of dissimilarities, and may also be used for further investigation when one wishes to perform data reduction.

REFERENCES

CHURCH, R. (1978), "Contrasts Between Facility Location Approaches and Non-hierarchical Cluster Analysis," paper presented at the ORSA/TIMS Joint National Meeting, Los Angeles, November 1978.

ERLENKOTTER, D. (1978), "A Dual-Based Procedure for Uncapacitated Facility Location," Operations Research, 26, 992-1009.

GOWER, J.C. (1971), "A General Coefficient of Similarity and Some of its Properties," Biometrics, 27, 857-871.

KAUFMAN, L. and ROUSSEEUW, P. (1986), "Clustering Large Data Sets," in Pattern Recognition in Practice II, edited by E.S. Gelsema and L.N. Kanal, North-Holland, 425-437.

KAUFMAN, L. and ROUSSEEUW, P. (1987), Finding Groups in Data, Wiley, New York (in preparation).

KLASTORIN, T.D. (1985), "The p-Median Problem for Cluster Analysis : A Comparative Test Using the Mixture Model Approach," Management Science, 31(1), 84-95.

MASSART, D.L., PLASTRIA, F., and KAUFMAN, L. (1983), "Non-Hierarchical Clustering with MASLOC," Pattern Recognition, 16(5), 507-516.

MILLIGAN, G.W. (1980), "An Examination of the Effect of Six Types of Error Perturbation on Fifteen Clustering Algorithms," Psychometrika, 45, 325-342.

MOREAU, J.-V. (1986), "The bootstrap approach to clustering," paper presented at the NATO advanced study institute : Pattern Recognition Theory and Applications, Spa, Belgium, June 1986.

MULVEY, J. and COWDER, H. (1979), "Cluster Analysis : An Application of Lagrangian Relaxation," Management Science, 25, 329-340.

PLASTRIA, F. (1986), "Two hierarchies associated with each clustering scheme," Pattern Recognition, 19(2), 193-196.

RAO, M.R. (1971), "Cluster Analysis and Mathematical Programming," _Journal of the American Statistical Association_, _66_, 622-626.

ROUSSEEUW, P. (1985), "Representing Data Partitions," _Proceedings of the Statistical Computing Section of the American Statistical Association_, 275-280.

VINOD, H.D. (1969), "Integer Programming and the Theory of Grouping," _Journal of the American Statistical Association_, _64_, 506-519.