Chapter 447

Medoid Partitioning

Introduction

The objective of cluster analysis is to partition a set of objects into two or more clusters such that objects within a cluster are similar and objects in different clusters are dissimilar. The medoid partitioning algorithms presented here attempt to accomplish this by finding a set of representative objects called *medoids*. The *medoid* of a cluster is defined as that object for which the average dissimilarity to all other objects in the cluster is minimal. If *k* clusters are desired, *k* medoids are found. Once the medoids are found, the data are classified into the cluster of the nearest medoid.

Two algorithms are available in this procedure to perform the clustering. The first, from Spath (1985), uses random starting cluster configurations. The second, from Kaufman and Rousseeuw (1990), makes special use of silhouette statistics to help determine the appropriate number of clusters. Both of these algorithms will be explained in more detail later.

Dissimilarities

The fundamental value used in cluster analysis is the dissimilarity between two objects. This section discusses how the dissimilarity is computed for the various types of data.

For multivariate data, a critical issue is how the distance between individual variables is combined to form the overall dissimilarity. This depends on the variable type, scaling type, and distance type that is selected.

We begin with a brief discussion of the possible types of variables.

Types of Cluster Variables

Interval Variables

Interval variables are continuous measurements that follow a linear scale. Examples include height, weight, age, price, temperature, and time. These values may be positive or negative.

Ordinal Variables

Ordinal variables are measurements that may be ordered according to magnitude. For example, a survey question may require you to pick one of five possible choices: strongly disagree (5), disagree (4), neutral (3), agree (2), or strongly agree (1).

Ratio Variables

Ratio variables are positive measurements in which the distinction between two numbers is constant if their ratio is constant. For example, the distinction between 3 and 30 would have the same meaning as the distinction between 30 and 300. Examples are chemical concentration or radiation intensity.

Nominal Variables

Nominal variables are those in which the number represents the state of the variable, but does not represent magnitude. The number is used for identification purposes only. Examples include gender, race, hair color, city of birth, or zipcode.

Symmetric-Binary Variables

Symmetric-binary variables have two possible outcomes, each of which carry the same information and weight. Examples include gender, marital status, or membership in a particular group. Usually, they are coded as 1 for yes and 0 for no, although this is not necessary.

Assymetric-Binary Variables

Asymmetric-binary variables are concerned with the presence or absence of a relatively rare event, the absence of which is rather unimportant and uninformative. For example, if a person has a scar on his face, he might be more easily identified. But if you know the person does not have a scar, that will not help you identify him.

Distance Calculation

The dissimilarity (distance) between two objects is fundamental to cluster analysis since the techniques goal is to place similar objects in the same cluster and dissimilar objects in different clusters. Unfortunately, the measurement of dissimilarity depends on the type of variable. For interval variables, the distance between to objects is simply the difference in their values. However, how do you quantify the difference between males and females? Is it simply 1 - 0 = 1? How do you combine the difference between males and females with the difference in age to form an overall dissimilar? These are the questions that will be answered in this section. This discussion follows Kaufman and Rousseeuw (1990) very closely.

Assume that you have N rows (observations) which are separated to be clustered into K groups. Each row consists of P variables. Two types of distance measures are available in the program: Euclidean and Manhattan.

The *Euclidean distance* d_{jk} between rows j and k is computed using

$$d_{jk} = \sqrt{\frac{\sum_{i=1}^{P} \delta_{ijk}^2}{P}}$$

and *Manhattan distance* d_{jk} between rows j and k is computed using

$$d_{jk} = \frac{\sum_{i=1}^{P} \left| \delta_{ijk} \right|}{P}$$

where for interval, ordinal, and ratio variables

$$\delta_{ijk} = z_{ij} - z_{ik}$$

and for asymmetric-binary, symmetric-binary, and nominal variables

$$\delta_{ijk} = \begin{cases} 1 & if \ x_{ij} \neq x_{ik} \\ 0 & if \ x_{ij} = x_{ik} \end{cases}$$

with the exception that for asymmetric-binary, the variable is completely ignored (P is decreased by one for this row) if both x_{ij} and x_{ik} are equal to zero (the non-rare event).

The value of z_{ij} for interval, ordinal, and ratio variables is defined next.

Interval Variables

You most likely have variables with several different scales. For example, you might have percentages, ages, rates, income levels, and so on. In order to remove distortions due to these differences in scales, the data are transformed to a common scale.

Four types of scaling are available: absolute value, standard deviation, range, and none. Each of these have the general form:

$$z_{ij} = \frac{x_{ij} - A_i}{B_i}$$

where x_{ij} represents the original data value for variable i and row j and z_{ij} represents the corresponding scale value. The scaling choice determines the values used for A_i and B_i.

The following table shows the scaling mechanism used for each type of scaling.

Type of Scaling	Value of A_i	Value of B_i
Absolute Value	$\frac{\sum_{j=1}^{N} x_{ij}}{N}$	$\frac{\sum_{j=1}^{N} \lvert x_{ij} - A_i \rvert}{N}$
Standard Deviation	$\frac{\sum_{j=1}^{N} x_{ij}}{N}$	$\sqrt{\frac{\sum_{j=1}^{N} \left(x_{ij} - A_i\right)^2}{N-1}}$
Range	$\underset{over\ j}{Min}\left(x_{ij}\right)$	$\underset{over\ j}{Max}\left(x_{ij}\right) - \underset{over\ j}{Min}\left(x_{ij}\right)$
None	0	1

Ordinal and Ratio Variables

The distance calculations for the ordinal and ratio variables are the same as for interval variables except that the values are transformed to an interval scale before distance calculations begin. The ranks of the ordinal variables and the natural logarithms of the ratio variables are substituted for the actual values. Once these transformations are made, the interval distance formulas are used.

Algorithm Details

Medoid Algorithm of Spath

The first medoid algorithm is presented in Spath (1985). The method minimizes an objective function by swapping objects from one cluster to another. Beginning at a random starting configuration, the algorithm proceeds to a local minimum by intelligently moving objects from one cluster to another. When no object moving would result in a reduction of the objective function, the procedure terminates. Unfortunately, this local minimum is not necessarily the global minimum. To overcome this limitation, the program lets you rerun the algorithm using several random starting configurations and the best solution is kept.

The objective function D is the total distance between the objects within a cluster. Mathematically, it is represented as follows:

$$D = \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j \in C_k} d_{ij}$$

where K is the number of clusters, d_{ij} is the distance between objects i and j, and C_k is the set of all objects in cluster k.

Medoid Algorithm of Kaufman and Rousseeuw

Kaufman and Rousseeuw (1990) present a medoid algorithm which they call PAM (Partition Around Medoids). This algorithm also attempts to minimize the total distance D (formula given above) between objects within each cluster. The algorithm proceeds through two phases.

In the first phase, a representative set of k objects is found. The first object selected has the shortest distance to all other objects. That is, it is in the center. An addition $k-1$ objects are selected one at a time in such a manner that at each step, they decrease D as much as possible.

In the second phase, possible alternatives to the k objects selected in phase one are considered in an iterative manner. At each step, the algorithm searches the unselected objects for the one that if exchanged with one of the k selected objects will lower the objective function the most. The exchange is made and the step is repeated. These iterations continue until no exchanges can be found that will lower the objective function.

Note that all potential swaps are considered and that the algorithm does not depend on the order of the objects on the database.

Silhouettes

Two of the most difficult tasks in cluster analysis are deciding on the appropriate number of clusters and deciding how to tell a bad cluster from a good one. Kaufman and Rousseeuw (1990) define a set of values called *silhouettes* that provide key information about both of these tasks. First, we will explain how these are calculated and then we will show how they are used.

Calculating Silhouettes

A silhouette value s is constructed for each object as follows.

1. Consider a particular object i which is in cluster A. Compute the value

   a = average dissimilarity of i to all other objects in A

   If A contains only one object, set a to zero.
2. For every other cluster not equal to A, find the cluster B that has the smallest average dissimilarity between its objects and i. Set

   b = average dissimilarity between i and the object in B.

   The cluster B is the nearest neighbor of object i.
3. Compute the silhouette s of object i as follows:

   If A contains only one object, set s = 0.

   If $a < b$, $s = 1 - a/b$.

   If $a > b$, $s = b/a - 1$.

   If $a = b$, $s = 0$.

Interpreting Silhouettes

A silhouette value is constructed for each object. The value can range from minus one to one. It measures how well an object has been classified by comparing its dissimilarity within its cluster to its dissimilarity with its nearest neighbor.

When *s* is close to one, the object is well classified. Its dissimilarity with other objects in its cluster is much less than its dissimilarity with objects in the nearest cluster.

When *s* is near zero, the object was just between clusters *A* and *B*. It was arbitrarily assigned to *A*.

When *s* is close to negative one, the object is poorly classified. Its dissimilarity with other objects in its cluster is much greater than its dissimilarity with objects in the nearest cluster. Why isn't it in the neighboring cluster?

Hence, the silhouette value summarizes how appropriate each object's cluster is.

Determining the Number of Clusters

One useful summary statistic is the average value of *s* across all objects. This summarizes how well the current configuration fits the data. An easy way to select the appropriate number of clusters is to choose that number of clusters which maximizes the average silhouette. We denote the maximum average silhouette across all values of *k* as *SC*.

Kaufman and Rousseeuw (1990) present the following table to aid in the interpretation of *SC*.

SC	Proposed Interpretation
0.71 to 1.00	A strong structure has been found.
0.51 to 0.70	A reasonable structure has been found.
0.26 to 0.50	The structure is weak and could be artificial. Try other methods on this database.
-1 to 0.25	No substantial structure has been found.

Finding Good Clusters

A bar chart of the silhouette values, sorted by cluster number and silhouette value, will show how well each cluster is doing. These charts will be discussed more in the output section.

Further Analysis

Once a cluster analysis has been run and an appropriate solution found, the cluster numbers should be saved to an empty variable so that the cluster solution can be further analyzed. What are some additional procedures that should be run? The most common is a discriminant analysis since it will let you study the impact of each of the variables on the solution. Discriminant analysis will also quantify how well the rows have been clustered. This will show up in the Wilks' lambda statistic.

In addition to discriminant analysis, you will want to produce various scatter plots in which the cluster number is used as a grouping variable. This will greatly increase your understanding of what the clusters that have been found look like.

Data Structure

The data are entered in the standard columnar format in which each column represents a single variable. A discussion of the types of variables will be presented shortly.

The data given in the following table contain information on twelve superstars in basketball. The stats are on a per game basis for games played through the 1989 season.

BBall dataset (subset)

Player	Height	FgPct	Points	Rebounds
Jabbar K.A.	86.0	55.9	24.6	11.2
Barry R	79.0	44.9	23.2	6.7
Baylor E	77.0	43.1	27.4	13.5
Bird L	81.0	50.3	25	10.2
Chamberlain W	85.0	54.0	30.1	22.9
Cousy B	72.5	37.5	18.4	5.2
Erving J	78.5	50.6	24.2	8.5

Procedure Options

This section describes the options available in this procedure.

Variables Tab

This panel specifies the variables used in the analysis.

Variables

Interval Variables

Designates interval-type variables (if any) or the columns of the matrix if distance or correlation matrix input was selected. Interval variables are continuous measurements that may be either positive or negative and follow a linear scale. Examples include height, weight, age, price, temperature, and time.

In general, an interval should keep the same importance throughout the scale. For example, the length of time between 1905 and 1925 is the same as the length of time between 1995 and 2015.

Note that a nonlinear transformation of an interval variable is probably not an interval variable. For example, the logarithm of height is not an interval variable since the value of an interval along the scale changes depending upon where you are on the scale.

Ratio Variables

Specifies the ratio variables (if any). Ratio-type variables are positive measurements in which the distinction between two numbers is constant if their ratio is constant. For example, the distinction between 3 and 30 would have the same meaning as the distinction between 30 and 300. Examples are chemical concentration or radiation intensity.

The logarithms of ratio variables are analyzed as if they were interval variables.

Ordinal Variables

Specifies the ordinal-type variables (if any). Ordinal variables are measurements that may be ordered according to magnitude. For example, a survey question may require you to pick one of five possible choices: strongly disagree (5), disagree (4), neutral (3), agree (2), or strongly agree (1). Interval variables are ordinal, but ordinal variables

are not necessarily interval. The original values of ordinal variables are replaced by their ranks. These ranks are then analyzed as if they were interval variables.

Nominal Variables

Specifies the nominal-type variables (if any). Nominal variables are those in which the number represents the state of the variable. Examples include gender, race, hair color, country of birth, or zipcode. If a nominal variable has only two categories, it is often called a binary variable.

Nominal variables are analyzed using the number of matches between two individuals.

Symmetric-Binary Variables

Specifies the symmetric binary-type variables (if any). Symmetric binary variables have two possible outcomes, each of which carries the same information and weight. Examples include gender, marital status, or membership in a particular group. Usually, they are coded as 1 for yes or 0 for no, although this is not necessary. These variables are analyzed using the number of matches between two individuals.

Asymmetric-Binary Variables

Specifies the asymmetric binary-type variables (if any). Asymmetric binary-scaled variables are concerned with the presence or absences of a relatively rare event, the absence of which is unimportant.

These variables are analyzed using the number of matches in which both individuals have the trait of interest. Those cases in which both individuals do not have the trait are not of interest and are ignored.

Clustering Options

Cluster Method

This option specifies which of the two medoid algorithms to be used.

- **Spath**

  Perform the analysis using Spath's medoid partitioning algorithm. This algorithm was discussed in the introduction. When this option is selected, you must also set the Best Starting Configuration, Weighting Method, and Number Random Starts options.

- **Kaufman - Rousseeuw**

  Perform the analysis using Kaufman and Rousseeuw's medoid partitioning algorithm. This algorithm was discussed in the introduction.

Distance Method

This option specifies with Euclidean or Manhattan distance is used. Euclidean distance may be thought of as straight-line (or as the crow flies) distance. Manhattan distance is often referred to as city-block distance since it is analogous to walking along an imaginary sidewalk to get from point A to B. Most users will use Euclidean distance.

Scaling Method

Specify the type of scaling to be used from Interval, Ordinal, and Ratio variables. Possible choices are Standard Deviation, Average Absolute Deviation, Range, and None. These were discussed in the introduction to this chapter.

Max Iterations

This option sets a maximum number of iterations that are attempted before the algorithm terminates. This avoids the possible of the algorithm going into an infinite loop.

Clustering Options – Number of Clusters

Minimum Clusters

The minimum value of *K* to search. A separate cluster analysis is attempted for each value between the Minimum Clusters and the Maximum Clusters. The actual number of clusters used is set above by the Reported Clusters option.

Maximum Clusters

The maximum value of *K* to search. A separate cluster analysis is attempted for each value between the Minimum Clusters and the Maximum Clusters. The actual number of clusters used is set above by the Reported Clusters option.

Reported Clusters

This is the number of clusters to be reported on. Although the program can find results for a range of cluster sizes, this option set the size that is actually used. It is used in the Row Detail section and by the Storage Tab section.

Clustering Options – Spath Cluster Method Options

Number Random Starts

This is the number of random starting configurations that are attempted during Spath's algorithm. Usually, ten starting configurations should be enough.

Best Starting Configuration

This option applies to Method = Spath only. In Spath's algorithm, a number of random starting configurations are tried and the best in for each cluster size is retained. This option determines which statistic is used to indicate the best.

- **Mean Distance**
  The configuration with the smallest average dissimilarity is selected.

- **Silhouette**
  The configuration with the largest average silhouette is selected.

Weighting Method

This option designates which objective function is minimized during Spath's algorithm. Two types are possible.

- **Regular**
  Minimize the sum of the distances between all individuals within each cluster.

- **Weighted**
  Minimize the weighted sum of the distances between all individuals within each cluster. The weights are one over the number of objects in the cluster.

Format Options

Label Variable

This is an optional variable containing identification for each row (object). These labels are used to enhance the interpretability of the reports.

Input Format

Specify the type of data format that you have. The choices are

- **Raw Data**

  The variables are in the standard format in which each row represents an object and each column represents a variable.

- **Distances**

  The variables containing a distance matrix are specified in the Interval Variables option. Note that this matrix contains the distances between each pair of objects. Each object is represented by a row and the corresponding column. Also, the matrix must be complete. You cannot use only the lower triangular portion, for example.

- **Correlations 1**

  The variables containing a correlation matrix are specified in the Interval Variables option. Correlations are converted to distances using the formula:

$$d_{ij} = \frac{1 - r_{ij}}{2}$$

- **Correlations 2**

  The variables containing a correlation matrix are specified in the Interval Variables option. Correlations are converted to distances using the formula:

$$d_{ij} = 1 - \left| r_{ij} \right|$$

- **Correlations 3**

  The variables containing a correlation matrix are specified in the Interval Variables option. Correlations are converted to distances using the formula:

$$d_{ij} = 1 - r_{ij}^2$$

  Note that all three types of correlation matrices must be completely specified. You cannot specify only the lower or upper triangular portions. Also, the rows correspond to variables. That is, the values along the first row represent the correlations of the first variable with each of the other variables. Hence, you cannot rearrange the order of the matrix.

Reports Tab

The following options control the formatting of the reports.

Select Reports

Iteration Report - Row Detail Report

Specify whether to display the indicated reports.

Report Options

Precision

Specify the precision of numbers in the report. Single precision will display seven-place accuracy, while double precision will display thirteen-place accuracy.

Variable Names

This option lets you select whether to display variable names, variable labels, or both.

Storage Tab

Storage Variable

These options let you specify where to store various row-wise statistics.

Store Cluster Id in Variable

You can automatically store the cluster identification number of each row into the variable specified here. The configuration stored is for the number of clusters specified in the Reported Clusters option.

Warning: Any data already in this column are replaced by the cluster number. Be careful not to specify columns that contain important data.

Example 1 – Medoid Partitioning

This section presents an example of how to run a medoid partitioning analysis. The data used were shown above and are found in the BBall dataset.

You may follow along here by making the appropriate entries or load the completed template **Example 1** by clicking on Open Example Template from the File menu of the Medoid Partitioning window.

1 Open the BBall dataset.

- From the File menu of the NCSS Data window, select **Open Example Data**.
- Click on the file **BBall.NCSS**.
- Click **Open**.

2 Open the Medoid Partitioning window.

- Using the Analysis menu or the Procedure Navigator, find and select the **Medoid Partitioning** procedure.
- On the menus, select **File**, then **New Template**. This will fill the procedure with the default template.

3 Specify the variables.

- On the Medoid Partitioning window, select the **Variables tab**.
- Double-click in the **Interval Variables** box. This will bring up the variable selection window.
- Select **Height, Weight, FgPct, FtPct, Points, Rebounds** (hold down control button) from the list of variables and then click **Ok**. “Height-Points,Rebounds” will appear in the Interval Variables box.
- Enter **2** for **Reported Clusters**.
- Enter **4** for **Number Random Starts**.
- Double-click in the **Label Variables** box. This will bring up the variable selection window.
- Select **Player** from the list of variables and then click **Ok**. “Player” will appear in the Label Variable box.

4 Run the procedure.

- From the Run menu, select **Run Procedure**. Alternatively, just click the green Run button.

Iteration Detail Section

Iteration Detail Section

Number Clusters	(Minimize This) Average Distance	Adjusted Average Distance	(Maximize This) Average Silhouette
2	35.977405	5.996234	0.135735
2	34.352873	5.725479	0.185579
2	34.862052	5.810342	0.170356
2	36.031237	6.005206	0.101405
3	19.525066	4.881267	0.094407
3	21.106317	5.276579	0.033435
3	19.005957	4.751489	0.045621
3	22.202362	5.550590	-0.026350
4	12.547872	4.182624	-0.013869
4	12.318440	4.106147	0.044989
4	12.210147	4.070049	0.018876
4	14.209356	4.736452	-0.097672
5	9.344940	3.893725	-0.099737
5	10.556815	4.398673	-0.189487
5	8.274123	3.447551	-0.045335
5	8.049819	3.354091	-0.004580

The results of this report may vary from run to run. This report shows the values of the objective functions for each iteration and number of clusters. This report is only generated when the Method option is set to Spath.

The report is especially useful in determining if you have set the number of random starts correctly. If you can see that two or three configurations at the desired number of clusters are identical then you have set the Number Random Starts large enough. Otherwise, you should increase this value and rerun the analysis.

In this example, we will conclude that *k* is two (determined from a later report). However, we notice that we have not achieved the maximum silhouette value (0.185579) more than once. We should change the Number Replications options to ten and rerun the analysis.

Average Distance

This is the value of the average dissimilarity. It is computed using

$$D = \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j \in C_k} d_{ij}$$

Note that this value has been rescaled as a percentage from the maximum distance in the dissimilarity matrix to improve readability.

Adjusted Average Distance

This is the value of the adjusted average dissimilarity. It is computed using

$$D_{adjusted} = \frac{K}{N} \sum_{k=1}^{K} \sum_{i \in C_k} \sum_{j \in C_k} d_{ij}$$

Note that this value has been rescaled as a percentage from the maximum distance in the dissimilarity matrix to improve readability.

Average Silhouette

This is the average of the silhouette values of all rows.

Iteration Summary Section

Iteration Summary Section

Number Clusters	(Minimize This) Average Distance	Adjusted Average Distance	(Maximize This) Average Silhouette
2	34.352873	5.725479	0.185579
3	19.525066	4.881267	0.094407
4	12.318440	4.106147	0.044989
5	8.049819	3.354091	-0.004580

This report shows the values of the objective functions for each number of clusters.

This report is used to determine the appropriate number of clusters. The number selected corresponds to the maximum value of the last (Average Silhouette) column. Usually, the row selected will have a respectable value of the Adjusted Average Distance (this value should be near its minimum).

The definitions of the columns were given above and will not be repeated here.

Cluster Medoids Section

Cluster Medoids Section

Variable	Cluster1	Cluster2
Height	86	77
Weight	230	210
FgPct	55.9	48.5
FtPct	72.1	83.8
Points	24.6	25.7
Rebounds	11.2	7.5
Row	1 Jabbar K.A	10 Robertson

This report gives the medoid (most centrally located) of each cluster. It is provided to help you interpret and recognize each cluster. The last row of the report gives the row number (and label if designated) of the each cluster's medoid.

Notice that the players in cluster one are typically nine inches taller and pull down about four more rebounds than the players in cluster two. Apparently, cluster one represents centers (or tall forwards) and cluster two represents other players.

Row Detail Section

Row Detail Section

Row	Cluster	Nearest Neighbor	Average Distance Within	Average Distance Neighbor	Silhouette Value	Silhouette Bar
5 Chamberlai	1	2	58.85	74.48	0.2098	||||||||||
11 Russell B	1	2	58.40	57.83	-0.0098	|
1 Jabbar K.A	1	2	47.59	44.28	-0.0696	|
3 Baylor E	1	2	47.74	31.69	-0.3363	|
9 Jordan M	1	2	56.34	32.62	-0.4210	|
Cluster Average	1	(5)	53.79	48.18	-0.1254	
2 Barry R	2	1	24.20	47.65	0.4921	||||||||||||||||||||||||
10 Robertson	2	1	21.94	42.43	0.4830	||||||||||||||||||||||||
12 West J	2	1	29.13	51.74	0.4370	||||||||||||||||||||
7 Erving J	2	1	23.03	40.90	0.4369	||||||||||||||||||||
6 Cousy B	2	1	45.02	68.58	0.3435	||||||||||||||||
8 Johnson M	2	1	30.31	45.50	0.3339	||||||||||||||||
4 Bird L	2	1	27.19	40.44	0.3275	||||||||||||||||
Cluster Average	2	(7)	28.69	48.18	0.4077	
Overall Average		(12)	39.15	48.18	0.1856	= SC

This report displays information about each row that was clustered. The report is sorted by Silhouette Value within cluster.

Row

The row number and, if designated, label of this individual. Each row of the database is represented on this report.

Cluster

This is the number of the cluster into which this row was classified.

Nearest Neighbor

This is the identification number of the nearest cluster to this row (other than the one that it is in). This information is used in computing the silhouette value.

Average Distance Within

This is the average distance between this object and all other objects in the cluster. This is the value of *a* in the computation of the silhouette.

Average Distance Neighbor

This is the average distance between this object and the objects in the nearest neighbor. This is the value of *b* in the computation of the silhouette.

Silhouette Value

This is the value of the silhouette. Its interpretation was presented in the introduction and will not be repeated here. We note that the value should be positive and most rows should be greater than 0.50. The fact that several of the rows in this analysis have negative silhouette values would cause us to toss out this cluster configuration and look for a better one.