Title

stata.com

cluster — Introduction to cluster-analysis commands

Description Syntax Remarks and examples References Also see

Description

Stata's cluster-analysis routines provide several hierarchical and partition clustering methods, postclustering summarization methods, and cluster-management tools. This entry presents an overview of cluster analysis, the `cluster` and `clustermat` commands (also see [MV] **clustermat**), as well as Stata's cluster-analysis management tools. The hierarchical clustering methods may be applied to the data by using the `cluster` command or to a user-supplied dissimilarity matrix by using the `clustermat` command.

The `cluster` command has the following *subcommand*s, which are detailed in their respective manual entries.

Partition-clustering methods for observations

`kmeans`	[MV] **cluster kmeans and kmedians**	Kmeans cluster analysis
`kmedians`	[MV] **cluster kmeans and kmedians**	Kmedians cluster analysis

Hierarchical clustering methods for observations

`singlelinkage`	[MV] **cluster linkage**	Single-linkage cluster analysis
`averagelinkage`	[MV] **cluster linkage**	Average-linkage cluster analysis
`completelinkage`	[MV] **cluster linkage**	Complete-linkage cluster analysis
`waveragelinkage`	[MV] **cluster linkage**	Weighted-average linkage cluster analysis
`medianlinkage`	[MV] **cluster linkage**	Median-linkage cluster analysis
`centroidlinkage`	[MV] **cluster linkage**	Centroid-linkage cluster analysis
`wardslinkage`	[MV] **cluster linkage**	Ward's linkage cluster analysis

Postclustering commands

`stop`	[MV] **cluster stop**	Cluster-analysis stopping rules
`dendrogram`	[MV] **cluster dendrogram**	Dendrograms for hierarchical cluster analysis
`generate`	[MV] **cluster generate**	Generate grouping variables from a cluster analysis

User utilities

`notes`	[MV] **cluster notes**	Cluster analysis notes
`dir`	[MV] **cluster utility**	Directory list of cluster analyses
`list`	[MV] **cluster utility**	List cluster analyses
`drop`	[MV] **cluster utility**	Drop cluster analyses
`use`	[MV] **cluster utility**	Mark cluster analysis as most recent one
`rename`	[MV] **cluster utility**	Rename cluster analyses
`renamevar`	[MV] **cluster utility**	Rename cluster-analysis variables

Programmer utilities

	[MV] **cluster programming subroutines**	Add cluster-analysis routines
`query`	[MV] **cluster programming utilities**	Obtain cluster-analysis attributes
`set`	[MV] **cluster programming utilities**	Set cluster-analysis attributes
`delete`	[MV] **cluster programming utilities**	Delete cluster-analysis attributes
`parsedistance`	[MV] **cluster programming utilities**	Parse (dis)similarity measure names
`measures`	[MV] **cluster programming utilities**	Compute (dis)similarity measures

The `clustermat` command has the following *subcommand*s, which are detailed along with the related `cluster` command manual entries. Also see [MV] **clustermat**.

Hierarchical clustering methods for matrices

`singlelinkage`	[MV] **cluster linkage**	Single-linkage cluster analysis
`averagelinkage`	[MV] **cluster linkage**	Average-linkage cluster analysis
`completelinkage`	[MV] **cluster linkage**	Complete-linkage cluster analysis
`waveragelinkage`	[MV] **cluster linkage**	Weighted-average linkage cluster analysis
`medianlinkage`	[MV] **cluster linkage**	Median-linkage cluster analysis
`centroidlinkage`	[MV] **cluster linkage**	Centroid-linkage cluster analysis
`wardslinkage`	[MV] **cluster linkage**	Ward's linkage cluster analysis

Also, the `clustermat stop` postclustering command has syntax similar to that of the `cluster stop` command; see [MV] **cluster stop**. For the remaining postclustering commands and user utilities, you may specify either `cluster` or `clustermat`—it does not matter which.

If you are new to Stata's cluster-analysis commands, we recommend that you first read this entry and then read the following:

[MV] ***measure_option***	Option for similarity and dissimilarity measures
[MV] **clustermat**	Cluster analysis of a dissimilarity matrix
[MV] **cluster kmeans and kmedians**	Kmeans and kmedians cluster analysis
[MV] **cluster linkage**	Hierarchical cluster analysis
[MV] **cluster dendrogram**	Dendrograms for hierarchical cluster analysis
[MV] **cluster stop**	Cluster-analysis stopping rules
[MV] **cluster generate**	Generate grouping variables from a cluster analysis

Syntax

Cluster analysis of data

```
cluster subcommand ...
```

Cluster analysis of a dissimilarity matrix

```
clustermat subcommand ...
```

Remarks and examples

stata.com

Remarks are presented under the following headings:

Introduction to cluster analysis

Cluster analysis attempts to determine the natural groupings (or clusters) of observations. Sometimes this process is called "classification", but this term is used by others to mean discriminant analysis, which is related but is not the same; see [MV] **discrim**. To avoid confusion, we will use "cluster analysis" or "clustering" when referring to finding groups in data. Defining cluster analysis is difficult (maybe impossible). Kaufman and Rousseeuw (1990) start their book by saying, "Cluster analysis is the art of finding groups in data." Everitt et al. (2011, 7) use the terms "cluster", "group", and "class" and say, concerning a formal definition for these terms, "In fact it turns out that such formal definition is not only difficult but may even be misplaced."

Everitt et al. (2011) and Gordon (1999) provide examples of the use of cluster analysis, such as in refining or redefining diagnostic categories in psychiatry, detecting similarities in artifacts by archaeologists to study the spatial distribution of artifact types, discovering hierarchical relationships in taxonomy, and identifying sets of similar cities so that one city from each class can be sampled in a market research task. Also, the activity now called "data mining" relies extensively on cluster-analysis methods.

We view cluster analysis as an exploratory data-analysis technique. According to Everitt, "Many cluster-analysis techniques have taken their place alongside other exploratory data-analysis techniques as tools of the applied statistician. The term exploratory is important here because it explains the largely absent 'p-value', ubiquitous in many other areas of statistics. ... Clustering methods are intended largely for generating rather than testing hypotheses" (1993, 10).

Although some have said that there are as many cluster-analysis methods as there are people performing cluster analysis. This is a gross understatement! There exist infinitely more ways to perform a cluster analysis than people who perform them.

There are several general types of cluster-analysis methods, each having many specific methods. Also, most cluster-analysis methods allow a variety of distance measures for determining the similarity or dissimilarity between observations. Some of the measures do not meet the requirements to be called a distance metric, so we use the more general term "dissimilarity measure" in place of distance. Similarity measures may be used in place of dissimilarity measures. There are an infinite number of similarity and dissimilarity measures. For instance, there are an infinite number of Minkowski distance metrics, with the familiar Euclidean, absolute-value, and maximum-value distances being special cases.

In addition to cluster method and dissimilarity measure choice, if you are performing a cluster analysis, you might decide to perform data transformations and/or variable selection before clustering. Then you might need to determine how many clusters there really are in the data, which you can do using stopping rules. There is a surprisingly large number of stopping rules mentioned in the literature. For example, Milligan and Cooper (1985) compare 30 different stopping rules.

Looking at all of these choices, you can see why there are more cluster-analysis methods than people performing cluster analysis.

Stata's cluster-analysis system

Stata's `cluster` and `clustermat` commands were designed to allow you to keep track of the various cluster analyses performed on your data. The main clustering subcommands—`singlelinkage`, `averagelinkage`, `completelinkage`, `waveragelinkage`, `medianlinkage`, `centroidlinkage`, `wardslinkage` (see [MV] **cluster linkage**), `kmeans`, and `kmedians` (see [MV] **cluster kmeans and kmedians**)—create named Stata cluster objects that keep track of the variables these methods create and hold other identifying information for the cluster analysis. These cluster objects become part of your dataset. They are saved with your data when your data are `saved` and are retrieved when you again `use` your dataset; see [D] **save** and [D] **use**.

Post–cluster-analysis subcommands are available with the `cluster` and `clustermat` commands so that you can examine the created clusters. Cluster-management tools are provided that allow you to add information to the cluster objects and to manipulate them as needed. The main clustering subcommands, postclustering subcommands, and cluster-management tools are discussed in the following sections.

Stata's clustering methods fall into two general types: partition and hierarchical. These two types are discussed below. There exist other types, such as fuzzy partition (where observations can belong to more than one group). Stata's `cluster` command is designed so that programmers can extend it by adding more methods; see [MV] **cluster programming subroutines** and [MV] **cluster programming utilities** for details.

❑ Technical note

If you are familiar with Stata's large array of estimation commands, be careful to distinguish between cluster analysis (the `cluster` command) and the `vce(cluster` *clustvar*`)` option (see [R] ***vce_option***) allowed with many estimation commands. Cluster analysis finds groups in data. The `vce(cluster` *clustvar*`)` option allowed with various estimation commands indicates that the observations are independent across the groups defined by the option but are not necessarily independent within those groups. A grouping variable produced by the `cluster` command will seldom satisfy the assumption behind the use of the `vce(cluster` *clustvar*`)` option.

❑

Data transformations and variable selection

Stata's `cluster` command has no built-in data transformations, but because Stata has full data management and statistical capabilities, you can use other Stata commands to transform your data before calling the `cluster` command. Standardizing the variables is sometimes important to keep a variable with high variability from dominating the cluster analysis. In other cases, standardizing variables hides the true groupings present in the data. The decision to standardize or perform other data transformations depends on the type of data and the nature of the groups.

Data transformations (such as standardization of variables) and the variables selected for use in clustering can also greatly affect the groupings that are discovered. These and other cluster-analysis data issues are covered in Milligan and Cooper (1988) and Schaffer and Green (1996) and in many of the cluster-analysis texts, including Anderberg (1973); Gordon (1999); Everitt et al. (2011); and Späth (1980).

Similarity and dissimilarity measures

Several similarity and dissimilarity measures have been implemented for Stata's clustering commands for both continuous and binary variables. For information, see [MV] ***measure_option***.

Partition cluster-analysis methods

Partition methods break the observations into a distinct number of nonoverlapping groups. Stata has implemented two partition methods, kmeans and kmedians.

One of the more commonly used partition clustering methods is called kmeans cluster analysis. In kmeans clustering, the user specifies the number of clusters, k, to create using an iterative process. Each observation is assigned to the group whose mean is closest, and then based on that categorization, new group means are determined. These steps continue until no observations change groups. The algorithm begins with k seed values, which act as the k group means. There are many ways to specify the beginning seed values.

A variation of kmeans clustering is kmedians clustering. The same process is followed in kmedians as in kmeans, except that medians, instead of means, are computed to represent the group centers at each step. See [MV] **cluster kmeans and kmedians** for the details of the `cluster kmeans` and `cluster kmedians` commands.

These partition-clustering methods will generally be quicker and will allow larger datasets than the hierarchical clustering methods outlined next. However, if you wish to examine clustering to various numbers of clusters, you will need to execute `cluster` many times with the partition methods. Clustering to various numbers of groups by using a partition method typically does not produce clusters that are hierarchically related. If this relationship is important for your application, consider using one of the hierarchical methods.

Hierarchical cluster-analysis methods

Hierarchical clustering creates hierarchically related sets of clusters. Hierarchical clustering methods are generally of two types: agglomerative or divisive.

Agglomerative hierarchical clustering methods begin with each observation's being considered as a separate group (N groups each of size 1). The closest two groups are combined ($N - 1$ groups, one of size 2 and the rest of size 1), and this process continues until all observations belong to the same group. This process creates a hierarchy of clusters.

In addition to choosing the similarity or dissimilarity measure to use in comparing 2 observations, you can choose what to compare between groups that contain more than 1 observation. The method used to compare groups is called a linkage method. Stata's `cluster` and `clustermat` commands provide several hierarchical agglomerative linkage methods, which are discussed in the next section.

Unlike hierarchical agglomerative clustering, divisive hierarchical clustering begins with all observations belonging to one group. This group is then split in some fashion to create two groups. One of these two groups is then split to create three groups; one of these three is then split to create four groups, and so on, until all observations are in their own separate group. Stata currently has no divisive hierarchical clustering commands. There are relatively few mentioned in the literature, and they tend to be particularly time consuming to compute.

To appreciate the underlying computational complexity of both agglomerative and divisive hierarchical clustering, consider the following information paraphrased from Kaufman and Rousseeuw (1990). The first step of an agglomerative algorithm considers $N(N-1)/2$ possible fusions of observations to find the closest pair. This number grows quadratically with N. For divisive hierarchical clustering, the first step would be to find the best split into two nonempty subsets, and if all possibilities were considered, it would amount to $2^{(N-1)}-1$ comparisons. This number grows exponentially with N.

Agglomerative methods

Stata's `cluster` and `clustermat` commands provide the following hierarchical agglomerative linkage methods: single, complete, average, Ward's method, centroid, median, and weighted average. There are others mentioned in the literature, but these are the best-known methods.

Single-linkage clustering computes the similarity or dissimilarity between two groups as the similarity or dissimilarity between the closest pair of observations between the two groups. Complete-linkage clustering, on the other hand, uses the farthest pair of observations between the two groups to determine the similarity or dissimilarity of the two groups. Average-linkage clustering uses the average similarity or dissimilarity of observations between the groups as the measure between the two groups. Ward's method joins the two groups that result in the minimum increase in the error sum of squares. The other linkage methods provide alternatives to these basic linkage methods.

The `cluster singlelinkage` and `clustermat singlelinkage` commands implement single-linkage hierarchical agglomerative clustering; see [MV] **cluster linkage** for details. Single-linkage clustering suffers (or benefits, depending on your point of view) from what is called chaining. Because the closest points between two groups determine the next merger, long, thin clusters can result. If this chaining feature is not what you desire, consider using one of the other methods, such as complete linkage or average linkage. Because of special properties that can be computationally exploited, single-linkage clustering is faster and uses less memory than the other linkage methods.

Complete-linkage hierarchical agglomerative clustering is implemented by the `cluster completelinkage` and `clustermat completelinkage` commands; see [MV] **cluster linkage** for details. Complete-linkage clustering is at the other extreme from single-linkage clustering. Complete linkage produces spatially compact clusters, so it is not the best method for recovering elongated cluster structures. Several sources, including Kaufman and Rousseeuw (1990), discuss the chaining of single linkage and the clumping of complete linkage.

Kaufman and Rousseeuw (1990) indicate that average linkage works well for many situations and is reasonably robust. The `cluster averagelinkage` and `clustermat averagelinkage` commands provide average-linkage clustering; see [MV] **cluster linkage**.

Ward (1963) presented a general hierarchical clustering approach where groups were joined to maximize an objective function. He used an error-sum-of-squares objective function to illustrate. Ward's method of clustering became synonymous with using the error-sum-of-squares criteria. Kaufman and Rousseeuw (1990) indicate that Ward's method does well with groups that are multivariate normal and spherical but does not do as well if the groups are of different sizes or have unequal numbers of observations. The `cluster wardslinkage` and `clustermat wardslinkage` commands provide Ward's linkage clustering; see [MV] **cluster linkage**.

At each step of the clustering, centroid linkage merges the groups whose means are closest. The centroid of a group is the componentwise mean and can be interpreted as the center of gravity for the group. Centroid linkage differs from average linkage in that centroid linkage is concerned with the distance between the means of the groups, whereas average linkage looks at the average distance between the points of the two groups. The `cluster centroidlinkage` and `clustermat centroidlinkage` commands provide centroid-linkage clustering; see [MV] **cluster linkage**.

Weighted-average linkage and median linkage are variations on average linkage and centroid linkage, respectively. In both cases, the difference is in how groups of unequal size are treated when merged. In average linkage and centroid linkage, the number of elements of each group is factored into the computation, giving correspondingly larger influence to the larger group. These two methods are called unweighted because each observation carries the same weight. In weighted-average linkage and median linkage, the two groups are given equal weighting in determining the combined group, regardless of the number of observations in each group. These two methods are said to be weighted because observations from groups with few observations carry more weight than observations from groups with many observations. The `cluster waveragelinkage` and `clustermat waveragelinkage` commands provide weighted-average linkage clustering. The `cluster medianlinkage` and `clustermat medianlinkage` commands provide median linkage clustering; see [MV] **cluster linkage**.

Lance and Williams's recurrence formula

Lance and Williams (1967) developed a recurrence formula that defines, as special cases, most of the well-known hierarchical clustering methods, including all the hierarchical clustering methods found in Stata. Anderberg (1973); Jain and Dubes (1988); Kaufman and Rousseeuw (1990); Gordon (1999); Everitt et al. (2011); and Rencher and Christensen (2012) discuss the Lance–Williams formula and how most popular hierarchical clustering methods are contained within it.

From the notation of Everitt et al. (2011, 78), the Lance–Williams recurrence formula is

$$d_{k(ij)} = \alpha_i d_{ki} + \alpha_j d_{kj} + \beta d_{ij} + \gamma |d_{ki} - d_{kj}|$$

where d_{ij} is the distance (or dissimilarity) between cluster i and cluster j; $d_{k(ij)}$ is the distance (or dissimilarity) between cluster k and the new cluster formed by joining clusters i and j; and α_i, α_j, β, and γ are parameters that are set based on the particular hierarchical cluster-analysis method.

The recurrence formula allows, at each new level of the hierarchical clustering, the dissimilarity between the newly formed group and the rest of the groups to be computed from the dissimilarities of the current grouping. This approach can result in a large computational savings compared with recomputing at each step in the hierarchy from the observation-level data. This feature of the recurrence formula allows `clustermat` to operate on a similarity or dissimilarity matrix instead of the data.

The following table shows the values of α_i, α_j, β, and γ for the hierarchical clustering methods implemented in Stata. n_i, n_j, and n_k are the number of observations in group i, j, and k, respectively.

Clustering linkage method	α_i	α_j	β	γ
Single	$\frac{1}{2}$	$\frac{1}{2}$	0	$-\frac{1}{2}$
Complete	$\frac{1}{2}$	$\frac{1}{2}$	0	$\frac{1}{2}$
Average	$\frac{n_i}{n_i+n_j}$	$\frac{n_j}{n_i+n_j}$	0	0
Weighted average	$\frac{1}{2}$	$\frac{1}{2}$	0	0
Centroid	$\frac{n_i}{n_i+n_j}$	$\frac{n_j}{n_i+n_j}$	$-\alpha_i\alpha_j$	0
Median	$\frac{1}{2}$	$\frac{1}{2}$	$-\frac{1}{4}$	0
Ward's	$\frac{n_i+n_k}{n_i+n_j+n_k}$	$\frac{n_j+n_k}{n_i+n_j+n_k}$	$\frac{-n_k}{n_i+n_j+n_k}$	0

For information on the use of various similarity and dissimilarity measures in hierarchical clustering, see the next two sections.

Dissimilarity transformations and the Lance and Williams formula

The Lance–Williams formula, which is used as the basis for computing hierarchical clustering in Stata, is designed for use with dissimilarity measures. Before performing hierarchical clustering, Stata transforms similarity measures, both continuous and binary, to dissimilarities. After cluster analysis, Stata transforms the fusion values (heights at which the various groups join in the hierarchy) back to similarities.

Stata's `cluster` command uses

$$\text{dissimilarity} = 1 - \text{similarity}$$

to transform from a similarity to a dissimilarity measure and back again; see Kaufman and Rousseeuw (1990, 21). Stata's similarity measures range from either 0 to 1 or -1 to 1. The resulting dissimilarities range from 1 down to 0 and from 2 down to 0, respectively.

For continuous data, Stata provides both the `L2` and `L2squared` dissimilarity measures, as well as both the `L(#)` and `Lpower(#)` dissimilarity measures. Why have both an `L2` and `L2squared` dissimilarity measure, and why have both an `L(#)` and `Lpower(#)` dissimilarity measure?

For single- and complete-linkage hierarchical clustering (and for kmeans and kmedians partition clustering), there is no need for the additional `L2squared` and `Lpower(#)` dissimilarities. The same cluster solution is obtained when using `L2` and `L2squared` (or `L(#)` and `Lpower(#)`), except that the resulting heights in the dendrogram are raised to a power.

However, for the other hierarchical clustering methods, there is a difference. For some of these other hierarchical clustering methods, the natural default for dissimilarity measure is `L2squared`. For instance, the traditional Ward's (1963) method is obtained by using the `L2squared` dissimilarity option.

Warning concerning similarity or dissimilarity choice

With hierarchical centroid, median, Ward's, and weighted-average linkage clustering, Lance and Williams (1967); Anderberg (1973); Jain and Dubes (1988); Kaufman and Rousseeuw (1990); Everitt et al. (2011); and Gordon (1999) give various levels of warnings about using many of the similarity and dissimilarity measures ranging from saying that you should never use anything other than the default squared Euclidean distance (or Euclidean distance) to saying that the results may lack a useful interpretation.

Example 2 of [MV] **cluster linkage** illustrates part of the basis for this warning. The simple matching coefficient is used on binary data. The range of the fusion values for the resulting hierarchy is not between 1 and 0, as you would expect for the matching coefficient. The conclusions from the cluster analysis, however, agree well with the results obtained in other ways.

Stata does not restrict your choice of similarity or dissimilarity. If you are not familiar with these hierarchical clustering methods, use the default dissimilarity measure.

Synonyms

Cluster-analysis methods have been developed by researchers in many different disciplines. Because researchers did not always know what was happening in other fields, many synonyms for the different hierarchical cluster-analysis methods exist.

Blashfield and Aldenderfer (1978) provide a table of equivalent terms. Jain and Dubes (1988) and Day and Edelsbrunner (1984) also mention some of the synonyms and use various acronyms. Here is a list of synonyms:

Single linkage
- Nearest-neighbor method
- Minimum method
- Hierarchical analysis
- Space-contracting method
- Elementary linkage analysis
- Connectedness method

Complete linkage
- Furthest-neighbor method
- Maximum method
- Compact method
- Space-distorting method
- Space-dilating method
- Rank-order typal analysis
- Diameter analysis

Average linkage
- Arithmetic-average clustering
- Unweighted pair-group method using arithmetic averages
- UPGMA
- Unweighted clustering
- Group-average method
- Unweighted group mean
- Unweighted pair-group method

Weighted-average linkage
- Weighted pair-group method using arithmetic averages
- WPGMA
- Weighted group-average method

Centroid linkage
- Unweighted centroid method
- Unweighted pair-group centroid method
- UPGMC
- Nearest-centroid sorting

Median linkage
- Gower's method
- Weighted centroid method
- Weighted pair-group centroid method
- WPGMC
- Weighted pair method
- Weighted group method

Ward's method
- Minimum-variance method
- Error-sum-of-squares method
- Hierarchical grouping to minimize tr(W)
- HGROUP

Reversals

Unlike the other hierarchical methods implemented in Stata, centroid linkage and median linkage (see [MV] **cluster linkage**) can (and often do) produce reversals or crossovers; see Anderberg (1973), Jain and Dubes (1988), Gordon (1999), and Rencher and Christensen (2012). Normally, the dissimilarity or clustering criterion increases monotonically as the agglomerative hierarchical clustering progresses from many to few clusters. (For similarity measures, it monotonically decreases.) The dissimilarity value at which $k+1$ clusters form will be larger than the value at which k clusters form. When the dissimilarity does not increase monotonically through the levels of the hierarchy, it is said to have reversals or crossovers.

The word *crossover*, in this context, comes from the appearance of the resulting dendrogram (see [MV] **cluster dendrogram**). In a hierarchical clustering without reversals, the dendrogram branches extend in one direction (increasing dissimilarity measure). With reversals, some of the branches reverse and go in the opposite direction, causing the resulting dendrogram to be drawn with crossing lines (crossovers).

When reversals happen, Stata still produces correct results. You can still generate grouping variables (see [MV] **cluster generate**) and compute stopping rules (see [MV] **cluster stop**). However, the `cluster dendrogram` command will not draw a dendrogram with reversals; see [MV] **cluster dendrogram**. In all but the simplest cases, dendrograms with reversals are almost impossible to interpret visually.

Hierarchical cluster analysis applied to a dissimilarity matrix

What if you want to perform a cluster analysis using a similarity or dissimilarity measure that Stata does not provide? What if you want to cluster variables instead of observations? The `clustermat` command gives you the flexibility to do either; see [MV] **clustermat**.

User-supplied dissimilarities

There are situations where the dissimilarity between objects is evaluated subjectively (perhaps on a scale from 1 to 10 by a rater). These dissimilarities may be entered in a matrix and passed to the `clustermat` command to perform hierarchical clustering. Likewise, if Stata does not offer the dissimilarity measure you desire, you may compute the dissimilarities yourself and place them in a matrix and then use `clustermat` to perform the cluster analysis. [MV] **clustermat** illustrates both of these situations.

Clustering variables instead of observations

Sometimes you want to cluster variables rather than observations, so you can use the `cluster` command. One approach to clustering variables in Stata is to use `xpose` (see [D] **xpose**) to transpose the variables and observations and then to use `cluster`. Another approach is to use the `matrix dissimilarity` command with the `variables` option (see [MV] **matrix dissimilarity**) to produce a dissimilarity matrix for the variables. This matrix is then passed to `clustermat` to obtain the hierarchical clustering. See [MV] **clustermat**.

Postclustering commands

Stata's `cluster stop` and `clustermat stop` commands are used to determine the number of clusters. Two stopping rules are provided, the Caliński and Harabasz (1974) pseudo-F index and the Duda, Hart, and Stork (2001, sec. 10.10) Je(2)/Je(1) index with associated pseudo-T^2. You can easily add stopping rules to the `cluster stop` command; see [MV] **cluster stop** for details.

The `cluster dendrogram` command presents the dendrogram (cluster tree) after a hierarchical cluster analysis; see [MV] **cluster dendrogram**. Options allow you to view the top portion of the tree or the portion of the tree associated with a group. These options are important with larger datasets because the full dendrogram cannot be presented.

The `cluster generate` command produces grouping variables after hierarchical clustering; see [MV] **cluster generate**. These variables can then be used in other Stata commands, such as those that tabulate, summarize, and provide graphs. For instance, you might use `cluster generate` to create a grouping variable. You then might use the `pca` command (see [MV] **pca**) to obtain the first two principal components of the data. You could follow that with a graph (see *Stata Graphics Reference Manual*) to plot the principal components, using the grouping variable from the `cluster generate` command to control the point labeling of the graph. This method would allow you to get one type of view into the clustering behavior of your data.

Cluster-management tools

You may add notes to your cluster analysis with the `cluster notes` command; see [MV] **cluster notes**. This command also allows you to view and delete notes attached to the cluster analysis.

The `cluster dir` and `cluster list` commands allow you to list the cluster objects and attributes currently defined for your dataset. `cluster drop` lets you remove a cluster object. See [MV] **cluster utility** for details.

Cluster objects are referred to by name. If no name is provided, many of the `cluster` commands will, by default, use the cluster object from the most recently performed cluster analysis. The `cluster use` command tells Stata which cluster object to use. You can change the name attached to a cluster object with the `cluster rename` command and the variables associated with a cluster analysis with the `cluster renamevar` command. See [MV] **cluster utility** for details.

You can exercise fine control over the attributes that are stored with a cluster object; see [MV] **cluster programming utilities**.

References

Anderberg, M. R. 1973. *Cluster Analysis for Applications*. New York: Academic Press.

Blashfield, R. K., and M. S. Aldenderfer. 1978. The literature on cluster analysis. *Multivariate Behavioral Research* 13: 271–295. https://doi.org/10.1207/s15327906mbr1303_2.

Caliński, T., and J. Harabasz. 1974. A dendrite method for cluster analysis. *Communications in Statistics—Theory and Methods* 3: 1–27. https://doi.org/10.1080/03610927408827101.

Day, W. H. E., and H. Edelsbrunner. 1984. Efficient algorithms for agglomerative hierarchical clustering methods. *Journal of Classification* 1: 7–24. https://doi.org/10.1007/BF01890115.

Duda, R. O., P. E. Hart, and D. G. Stork. 2001. *Pattern Classification*. 2nd ed. New York: Wiley.

Everitt, B. S. 1993. *Cluster Analysis*. 3rd ed. London: Arnold.

Everitt, B. S., S. Landau, M. Leese, and D. Stahl. 2011. *Cluster Analysis*. 5th ed. Chichester, UK: Wiley.

Gordon, A. D. 1999. *Classification*. 2nd ed. Boca Raton, FL: Chapman & Hall/CRC.

Jain, A. K., and R. C. Dubes. 1988. *Algorithms for Clustering Data*. Englewood Cliffs, NJ: Prentice Hall.

Kaufman, L., and P. J. Rousseeuw. 1990. *Finding Groups in Data: An Introduction to Cluster Analysis*. New York: Wiley.

Lance, G. N., and W. T. Williams. 1967. A general theory of classificatory sorting strategies: 1. Hierarchical systems. *Computer Journal* 9: 373–380. https://doi.org/10.1093/comjnl/9.4.373.

Lee, C. H., and D. G. Steigerwald. 2018. Inference for clustered data. *Stata Journal* 18: 447–460.

Meekes, J., and W. H. J. Hassink. 2018. flowbca: A flow-based cluster algorithm in Stata. *Stata Journal* 18: 564–584.

Milligan, G. W., and M. C. Cooper. 1985. An examination of procedures for determining the number of clusters in a dataset. *Psychometrika* 50: 159–179. https://doi.org/10.1007/BF02294245.

——. 1988. A study of standardization of variables in cluster analysis. *Journal of Classification* 5: 181–204. https://doi.org/10.1007/BF01897163.

Mooi, E., M. Sarstedt, and I. Mooi-Reci. 2018. *Market Research: The Process, Data, and Methods Using Stata*. Singapore: Springer.

Raciborski, R. 2009. Graphical representation of multivariate data using Chernoff faces. *Stata Journal* 9: 374–387.

Rencher, A. C., and W. F. Christensen. 2012. *Methods of Multivariate Analysis*. 3rd ed. Hoboken, NJ: Wiley.

Rohlf, F. J. 1982. Single-link clustering algorithms. In Vol. 2 of *Handbook of Statistics*, ed. P. R. Krishnaiah and L. N. Kanal, 267–284. Amsterdam: North-Holland. https://doi.org/10.1016/S0169-7161(82)02015-X.

Schaffer, C. M., and P. E. Green. 1996. An empirical comparison of variable standardization methods in cluster analysis. *Multivariate Behavioral Research* 31: 149–167. https://doi.org/10.1207/s15327906mbr3102_1.

Sibson, R. 1973. SLINK: An optimally efficient algorithm for the single-link cluster method. *Computer Journal* 16: 30–34. https://doi.org/10.1093/comjnl/16.1.30.

Späth, H. 1980. *Cluster Analysis Algorithms for Data Reduction and Classification of Objects*. Chichester, UK: Ellis Horwood.

Ward, J. H., Jr. 1963. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association* 58: 236–244. https://doi.org/10.1080/01621459.1963.10500845.

Also see

[MV] **clustermat** — Introduction to clustermat commands

[MV] **cluster programming subroutines** — Add cluster-analysis routines

[MV] **cluster programming utilities** — Cluster-analysis programming utilities

[MV] **discrim** — Discriminant analysis