Overview: CLUSTER Procedure (estratto dalle informazioni presenti in rete di SAS)

The CLUSTER procedure hierarchically clusters the observations in a SAS data set by using one of 11 methods. The data can be coordinates or distances. If the data are coordinates, PROC CLUSTER computes (possibly squared) Euclidean distances. If you want non-Euclidean distances, use the DISTANCE procedure (see Chapter 32) to compute an appropriate distance data set that can then be used as input to PROC CLUSTER.

The clustering methods are: average linkage, the centroid method, complete linkage, density linkage (including Wong's hybrid and kth-nearest-neighbor methods), maximum likelihood for mixtures of spherical multivariate normal distributions with equal variances but possibly unequal mixing proportions, the flexible-beta method, McQuitty's similarity analysis, the median method, single linkage, two-stage density linkage, and Ward's minimum-variance method. Each method is described in the section Clustering Methods.

All methods are based on the usual agglomerative hierarchical clustering procedure. Each observation begins in a cluster by itself. The two closest clusters are merged to form a new cluster that replaces the two old clusters. Merging of the two closest clusters is repeated until only one cluster is left. The various clustering methods differ in how the distance between two clusters is computed.

The CLUSTER procedure is not practical for very large data sets because the CPU time is roughly proportional to the square or cube of the number of observations. The FASTCLUS procedure (see Chapter 34) requires time proportional to the number of observations and thus can be used with much larger data sets than PROC CLUSTER. If you want to cluster a very large data set hierarchically, use PROC FASTCLUS for a preliminary cluster analysis to produce a large number of clusters. Then use PROC CLUSTER to cluster the preliminary clusters hierarchically. This method is illustrated in Example 29.3.

PROC CLUSTER displays a history of the clustering process, showing statistics useful for estimating the number of clusters in the population from which the data are sampled. PROC CLUSTER also creates an output data set that can be used by the TREE procedure to draw a tree diagram of the cluster hierarchy or to output the cluster membership at any desired level. For example, to obtain the six-cluster solution, you could first use PROC CLUSTER with the OUTTREE= option, and then use this output data set as the input data set to the TREE procedure. With PROC TREE, specify NCLUSTERS=6 and the OUT= options to obtain the six-cluster solution and draw a tree diagram. For an example, see Example 91.1 in Chapter 91, The TREE Procedure.

For coordinate data, Euclidean distances are computed from differences between coordinate values. The use of differences has several important consequences:

For differences to be valid, the variables must have an interval or stronger scale of measurement. Ordinal or ranked data are generally not appropriate for cluster analysis.

For Euclidean distances to be comparable, equal differences should have equal practical importance. You might need to transform the variables linearly or nonlinearly to satisfy this condition. For example, if one variable is measured in dollars and one in euros, you might need to convert to the same currency. Or, if ratios are more meaningful than differences, take logarithms.

Variables with large variances tend to have more effect on the resulting clusters than variables with small variances. If you consider all variables to be equally important, you can use the STD option in PROC CLUSTER to standardize the variables to mean 0 and standard deviation 1. However, standardization is not always appropriate. See Milligan and Cooper (1987) for a Monte Carlo study on various methods of variable standardization. You should remove outliers before using PROC CLUSTER with the STD option unless you specify the TRIM= option. The STDIZE procedure (see Chapter 81) provides additional methods for standardizing variables and imputing missing values.

The ACECLUS procedure (see Chapter 22) is useful for linear transformations of the variables if any of the following conditions hold:

You have no idea how the variables should be scaled.

You want to detect natural clusters regardless of whether some variables have more influence than others.

You want to use a clustering method designed for finding compact clusters, but you want to be able to detect elongated clusters.

Agglomerative hierarchical clustering is discussed in all standard references on cluster analysis, such as Anderberg (1973), Sneath and Sokal (1973), Hartigan (1975), Everitt (1980), and Spath (1980). An especially good introduction is given by Massart and Kaufman (1983). Anyone considering doing a hierarchical cluster analysis should study the Monte Carlo results of Milligan (1980), Milligan and Cooper (1985), and Cooper and Milligan (1988). Other essential, though more advanced, references on hierarchical clustering include Hartigan (1977, pp. 60–68; 1981), Wong (1982), Wong and Schaack (1982), and Wong and Lane (1983). See Blashfield and Aldenderfer (1978) for a discussion of the confusing terminology in hierarchical cluster analysis.

Getting Started: CLUSTER Procedure

The following example shows how you can use the CLUSTER procedure to compute hierarchical clusters of observations in a SAS data set.

Suppose you want to determine whether national figures for birth rates, death rates, and infant death rates can be used to categorize countries. Previous studies indicate that the clusters computed from this type of data can be elongated and elliptical. Thus, you need to perform a linear transformation on the raw data before the cluster analysis.

The following data[1] from Rouncefield (1995) are birth rates, death rates, and infant death rates for 97 countries. The DATA step creates the SAS data set *Poverty*:

```
   data Poverty;
      input Birth Death InfantDeath Country $20. @@;
      datalines;
   24.7  5.7  30.8 Albania              12.5 11.9  14.4 Bulgaria
   13.4 11.7  11.3 Czechoslovakia       12   12.4   7.6
Former_E._Germany
   11.6 13.4  14.8 Hungary              14.3 10.2    16 Poland
   13.6 10.7  26.9 Romania                14    9  20.2
Yugoslavia
```

```
   17.7   10    23 USSR                15.2  9.5  13.1
Byelorussia_SSR
   13.4 11.6    13 Ukrainian_SSR       20.7  8.4  25.7 Argentina
   46.6   18   111 Bolivia             28.6  7.9    63 Brazil
   23.4  5.8  17.1 Chile               27.4  6.1    40 Columbia
   32.9  7.4    63 Ecuador             28.3  7.3    56 Guyana
   34.8  6.6    42 Paraguay            32.9  8.3 109.9 Peru
     18  9.6  21.9 Uruguay             27.5  4.4  23.3 Venezuela
     29 23.2    43 Mexico                12 10.6   7.9 Belgium
   13.2 10.1   5.8 Finland             12.4 11.9   7.5 Denmark
   13.6  9.4   7.4 France              11.4 11.2   7.4 Germany
10.1  9.2    11 Greece              15.1  9.1   7.5 Ireland
    9.7  9.1   8.8 Italy               13.2  8.6   7.1
Netherlands
   14.3 10.7   7.8 Norway              11.9  9.5  13.1 Portugal
   10.7  8.2   8.1 Spain               14.5 11.1   5.6 Sweden
   12.5  9.5   7.1 Switzerland         13.6 11.5   8.4 U.K.
   14.9  7.4     8 Austria              9.9  6.7   4.5 Japan
   14.5  7.3   7.2 Canada              16.7  8.1   9.1 U.S.A.
   40.4 18.7 181.6 Afghanistan         28.4  3.8    16 Bahrain
   42.5 11.5 108.1 Iran                42.6  7.8    69 Iraq
   22.3  6.3   9.7 Israel              38.9  6.4    44 Jordan
   26.8  2.2  15.6 Kuwait              31.7  8.7    48 Lebanon
   45.6  7.8    40 Oman                42.1  7.6    71
Saudi_Arabia
   29.2  8.4    76 Turkey              22.8  3.8    26
United_Arab_Emirates
   42.2 15.5   119 Bangladesh          41.4 16.6   130 Cambodia
   21.2  6.7    32 China               11.7  4.9   6.1 Hong_Kong
   30.5 10.2    91 India               28.6  9.4    75 Indonesia
   23.5 18.1    25 Korea               31.6  5.6    24 Malaysia
   36.1  8.8    68 Mongolia            39.6 14.8   128 Nepal
   30.3  8.1 107.7 Pakistan            33.2  7.7    45
Philippines
   17.8  5.2   7.5 Singapore           21.3  6.2  19.4 Sri_Lanka
   22.3  7.7    28 Thailand            31.8  9.5    64 Vietnam
   35.5  8.3    74 Algeria             47.2 20.2   137 Angola
   48.5 11.6    67 Botswana            46.1 14.6    73 Congo
   38.8  9.5  49.4 Egypt               48.6 20.7   137 Ethiopia
   39.4 16.8   103 Gabon               47.4 21.4   143 Gambia
   44.4 13.1    90 Ghana                 47 11.3    72 Kenya
     44  9.4    82 Libya               48.3   25   130 Malawi
   35.5  9.8    82 Morocco               45 18.5   141
Mozambique
     44 12.1   135 Namibia             48.5 15.6   105 Nigeria
   48.2 23.4   154 Sierra_Leone        50.1 20.2   132 Somalia
   32.1  9.9    72 South_Africa        44.6 15.8   108 Sudan
   46.8 12.5   118 Swaziland           31.1  7.3    52 Tunisia
   52.2 15.6   103 Uganda              50.5   14   106 Tanzania
   45.6 14.2    83 Zaire               51.1 13.7    80 Zambia
   41.7 10.3    66 Zimbabwe
   ;
```

The data set *Poverty* contains the character variable *Country* and the numeric variables *Birth*, *Death*, and *InfantDeath*, which represent the birth rate per thousand, death rate per thousand, and infant death rate per thousand. The **`$20.`** in the INPUT statement specifies that the variable *Country* is a character variable with a length of 20. The double trailing at sign (@@) in the INPUT statement holds the input line for further iterations of the DATA step, specifying that observations are input from each line until all values are read.

Because the variables in the data set do not have equal variance, you must perform some form of scaling or transformation. One method is to standardize the variables to mean zero and variance one. However, when you suspect that the data contain elliptical clusters, you can use the ACECLUS procedure to transform the data such that the resulting within-cluster covariance matrix is spherical. The procedure obtains approximate estimates of the pooled within-cluster covariance matrix and then computes canonical variables to be used in subsequent analyses.

The following statements perform the ACECLUS transformation by using the SAS data set *Poverty*. The OUT= option creates an output SAS data set called *Ace* to contain the canonical variable scores:

```
   proc aceclus data=Poverty out=Ace p=.03 noprint;
      var Birth Death InfantDeath;
   run;
```

The P= option specifies that approximately 3% of the pairs are included in the estimation of the within-cluster covariance matrix. The NOPRINT option suppresses the display of the output. The VAR statement specifies that the variables *Birth*, *Death*, and *InfantDeath* are used in computing the canonical variables.

The following statements invoke the CLUSTER procedure, using the SAS data set ACE created in the previous PROC ACECLUS run:

```
   ods graphics on;
   proc cluster data=Ace method=ward ccc pseudo print=15
outtree=Tree;
      var can1 can2 can3 ;
      id country;
      format country $12.;
   run;
   ods graphics off;
```

The **`ods graphics on`** statement asks procedures to produce ODS graphics where possible. Ward's minimum-variance clustering method is specified by the METHOD= option. The CCC option displays the cubic clustering criterion, and the PSEUDO option displays pseudo $F$and t^2 statistics. The PRINT=15 option displays only the last 15 generations of the cluster history. The OUTTREE= option creates an output SAS data set called *Tree* that can be used by the TREE procedure to draw a tree diagram.

The VAR statement specifies that the canonical variables computed in the ACECLUS procedure are used in the cluster analysis. The ID statement specifies that the variable *Country* should be added to the *Tree* output data set.

The results of this analysis are displayed in the following figures.

PROC CLUSTER first displays the table of eigenvalues of the covariance matrix (Figure 29.1). These eigenvalues are used in the computation of the cubic clustering criterion. The first two columns list each eigenvalue and the difference between the eigenvalue and its successor. The last two columns display the individual and cumulative proportion of variation associated with each eigenvalue.

Figure 29.1 Table of Eigenvalues of the Covariance Matrix

The CLUSTER Procedure

Ward's Minimum Variance Cluster Analysis

Eigenvalues of the Covariance Matrix

	Eigenvalue	Difference	Proportion	Cumulative
1	64.5500051	54.7313223	0.8091	0.8091
2	9.8186828	4.4038309	0.1231	0.9321
3	5.4148519		0.0679	1.0000

Root-Mean-Square Total-Sample Standard Deviation	5.156987

Root-Mean-Square Distance Between Observations	12.63199

Figure 29.2 displays the last 15 generations of the cluster history. First listed are the number of clusters and the names of the clusters joined. The observations are identified either by the ID value or by CL*n*, where *n* is the number of the cluster. Next, PROC CLUSTER displays the number of observations in the new cluster and the semipartial R square. The latter value represents the decrease in the proportion of variance accounted for by joining the two clusters.

Figure 29.2 Cluster History

Cluster History

NCL	Clusters Joined	FREQ	SPRSQ	RSQ	ERSQ	CCC	PSF	PST2	Tie

Cluster History

NCL	Clusters Joined		FREQ	SPRSQ	RSQ	ERSQ	CCC	PSF	PST2	Tie
15	**Oman**	**CL37**	5	0.0039	.957	.933	6.03	132	12.1	
14	**CL31**	**CL22**	13	0.0040	.953	.928	5.81	131	9.7	
13	**CL41**	**CL17**	32	0.0041	.949	.922	5.70	131	13.1	
12	**CL19**	**CL21**	10	0.0045	.945	.916	5.65	132	6.4	
11	**CL39**	**CL15**	9	0.0052	.940	.909	5.60	134	6.3	
10	**CL76**	**CL27**	6	0.0075	.932	.900	5.25	133	18.1	
9	**CL23**	**CL11**	15	0.0130	.919	.890	4.20	125	12.4	
8	**CL10**	**Afghanistan**	7	0.0134	.906	.879	3.55	122	7.3	
7	**CL9**	**CL25**	17	0.0217	.884	.864	2.26	114	11.6	
6	**CL8**	**CL20**	14	0.0239	.860	.846	1.42	112	10.5	
5	**CL14**	**CL13**	45	0.0307	.829	.822	0.65	112	59.2	
4	**CL16**	**CL7**	28	0.0323	.797	.788	0.57	122	14.8	
3	**CL12**	**CL6**	24	0.0323	.765	.732	1.84	153	11.6	
2	**CL3**	**CL4**	52	0.1782	.587	.613	-.82	135	48.9	
1	**CL5**	**CL2**	97	0.5866	.000	.000	0.00	.	135	

Next listed is the squared multiple correlation, R square, which is the proportion of variance accounted for by the clusters. Figure 29.2 shows that, when the data are grouped into three clusters, the proportion of variance accounted for by the clusters (R square) is just under 77%. The approximate expected value of R square is given in the *ERSQ* column. This expectation is approximated under the null hypothesis that the data have a uniform distribution instead of forming distinct clusters.

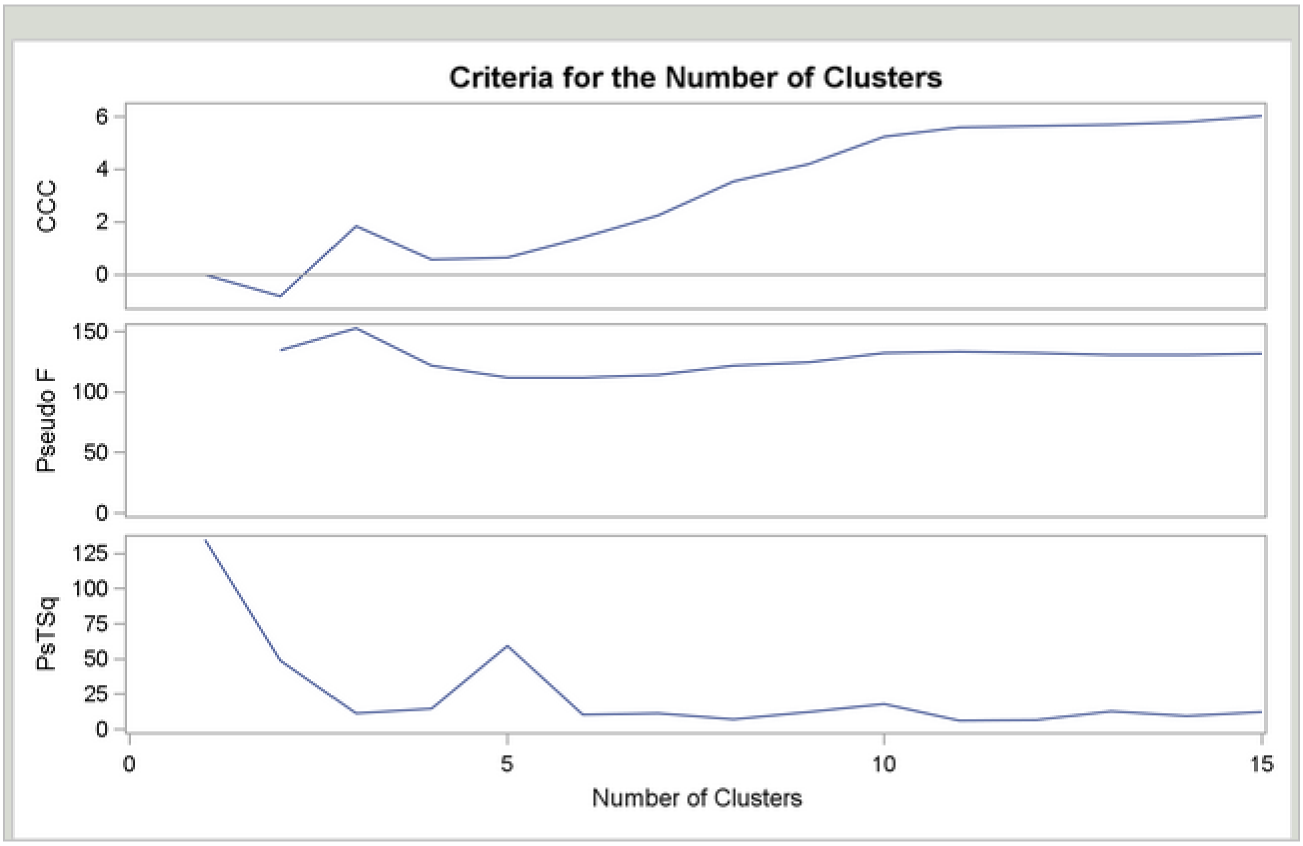
The next three columns display the values of the cubic clustering criterion (CCC), pseudo F(PSF), and t^2(PST2) statistics. These statistics are useful for estimating the number of clusters in the data.

The final column in Figure 29.2 lists ties for minimum distance; a blank value indicates the absence of a tie. A tie means that the clusters are indeterminate and that changing the order of the observations may change the clusters. See Example 29.4 for ways to investigate the effects of ties.

Figure 29.3 plots the three statistics for estimating the number of clusters. Peaks in the plot of the cubic clustering criterion with values greater than 2 or 3 indicate good clusters; peaks with values between 0 and 2 indicate possible clusters. Large negative values of the CCC can indicate outliers. In Figure 29.3, there is a local peak of the CCC when the number of clusters is 3. The CCC drops at 4 clusters and then steadily increases, leveling off at 11 clusters.

Another method of judging the number of clusters in a data set is to look at the pseudo Fstatistic (PSF). Relatively large values indicate good numbers of clusters. In Figure 29.3, the pseudo Fstatistic suggests 3 clusters or 11 clusters.

Figure 29.3 Plot of Statistics for Estimating the Number of Clusters



To interpret the values of the pseudo t^2statistic, look down the column or look at the plot from right to left until you find the first value markedly larger than the previous value, then move back up the column or to the right in the plot by one step in the cluster history. In Figure 29.3, you can see possibly good clustering levels at 11 clusters, 6 clusters, 3 clusters, and 2 clusters.

Considered together, these statistics suggest that the data can be clustered into 11 clusters or 3 clusters. The following statements examine the results of clustering the data into 3 clusters.

A graphical view of the clustering process can often be helpful in interpreting the clusters. The following statements use the TREE procedure to produce a tree diagram of the clusters:

```
goptions vsize=9in hsize=6.4in htext=.9pct htitle=3pct;
axis1 order=(0 to 1 by 0.2);
proc tree data=Tree out=New nclusters=3
          haxis=axis1 horizontal;
   height _rsq_;
   copy can1 can2 ;
   id country;
run;
```

The AXIS1 statement defines axis parameters that are used in the TREE procedure. The ORDER= option specifies the data values in the order in which they should appear on the axis.

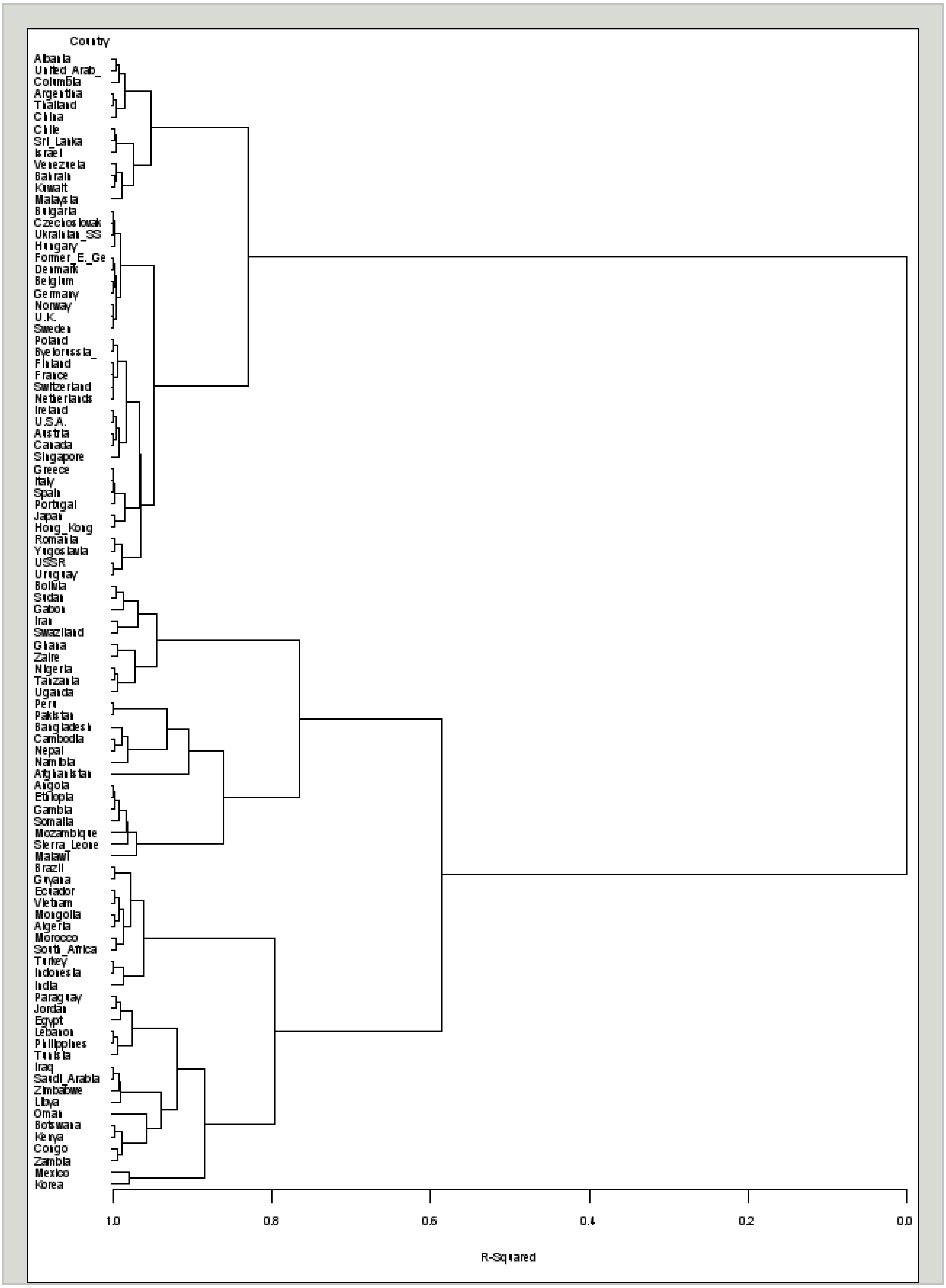
The preceding statements use the SAS data set *Tree* as input. The OUT= option creates an output SAS data set named *New* to contain information about cluster membership. The NCLUSTERS= option specifies the number of clusters desired in the data set *New*.

The TREE procedure produces high-resolution graphics by default. The HAXIS= option specifies AXIS1 to customize the appearance of the horizontal axis. The HORIZONTAL option orients the tree diagram horizontally. The HEIGHT statement specifies the variable *_RSQ_* (R square) as the height variable.

The COPY statement copies the canonical variables *can1* and *can2* (computed in the ACECLUS procedure) into the output SAS data set *New*. Thus, the SAS output data set *New* contains information for three clusters and the first two of the original canonical variables.

Figure 29.4 displays the tree diagram. The figure provides a graphical view of the information in Figure 29.2. As the number of branches grows to the left from the root, the R square approaches 1; the first three clusters (branches of the tree) account for over half of the variation (about 77%, from Figure 29.4). In other words, only three clusters are necessary to explain over three-fourths of the variation.

Figure 29.4 Tree Diagram of Clusters versus R-Square Values

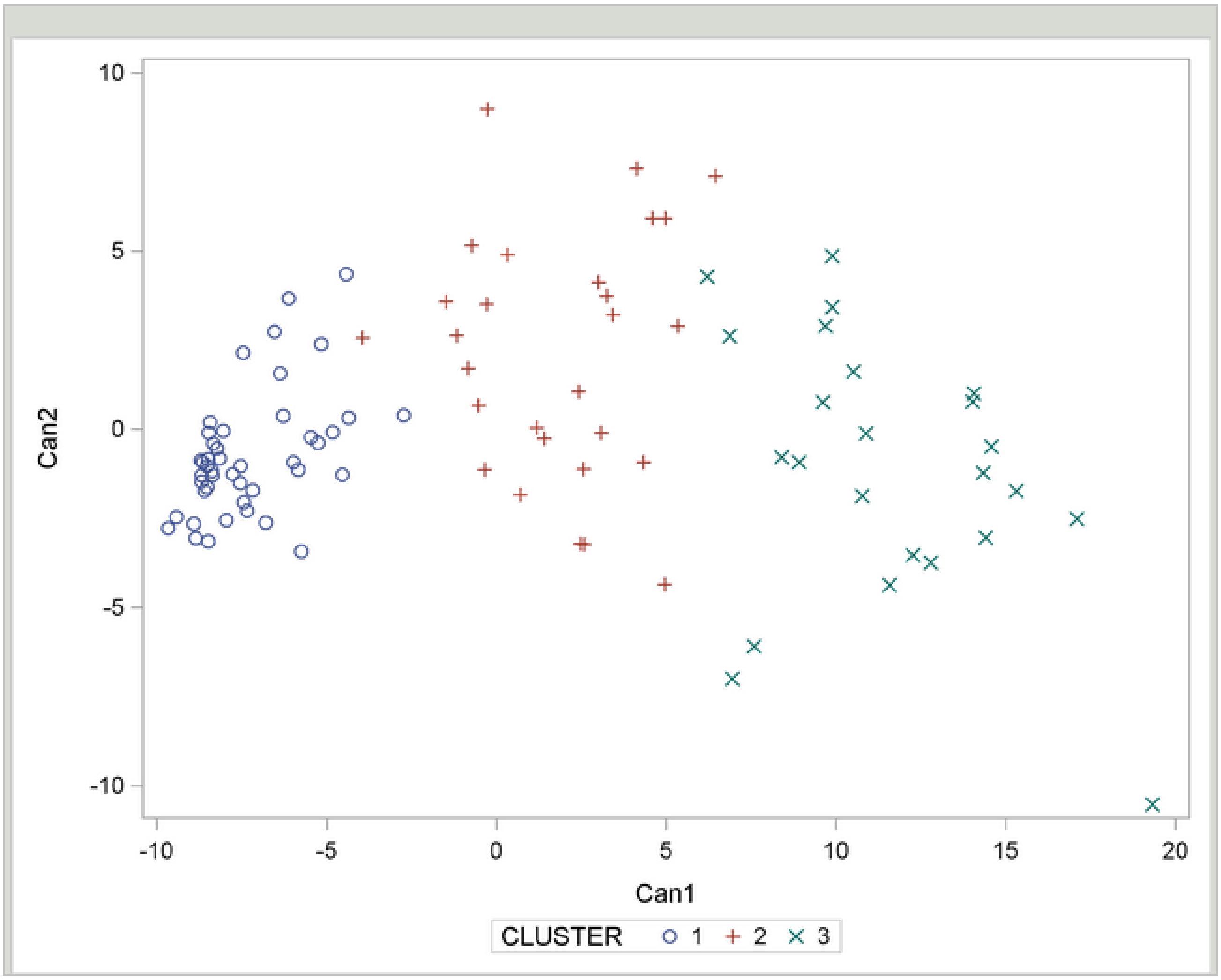


The following statements invoke the SGPLOT procedure on the SAS data set *New*:

```
proc sgplot data=New ;
   scatter y=can2 x=can1 / group=cluster ;
run;
```

The PLOT statement requests a plot of the two canonical variables, using the value of the variable *cluster* as the identification variable, as shown in Figure 29.5.

Figure 29.5 Plot of Canonical Variables and Cluster for Three Clusters



The statistics in Figure 29.2 and Figure 29.3, the tree diagram in Figure 29.4, and the plot of the canonical variables in Figure 29.5 assist in the estimation of clusters in the data. There seems to be reasonable separation in the clusters. However, you must use this information, along with experience and knowledge of the field, to help in deciding the correct number of clusters.

Footnotes

1. These data have been compiled from the ***United Nations Demographic Yearbook 1990*** (United Nations publications, Sales No. E/F.91.XII.1,

copyright 1991, United Nations, New York) and are reproduced with the permission of the United Nations.

Syntax: CLUSTER Procedure

The following statements are available in the CLUSTER procedure:

PROC CLUSTER METHOD = name <options> ;
    BY variables ;

    COPY variables ;

    FREQ variable ;

    ID variable ;

    RMSSTD variable ;

    VAR variables ;

Only the PROC CLUSTER statement is required, except that the FREQ statement is required when the RMSSTD statement is used; otherwise the FREQ statement is optional. Usually only the VAR statement and possibly the ID and COPY statements are needed in addition to the PROC CLUSTER statement. The rest of this section provides detailed syntax information for each of the preceding statements, beginning with the PROC CLUSTER statement. The remaining statements are covered in alphabetical order.

PROC CLUSTER Statement

PROC CLUSTER *METHOD=name <options>* ;

The PROC CLUSTER statement starts the CLUSTER procedure, specifies a clustering method, and optionally specifies details for clustering methods, data sets, data processing, and displayed output.

The METHOD= specification determines the clustering method used by the procedure. Any one of the following 11 methods can be specified for ***name***:

AVERAGE | AVE
    requests average linkage (group average, unweighted pair-group method using arithmetic averages, UPGMA). Distance data are squared unless you specify the NOSQUARE option.

CENTROID | CEN
    requests the centroid method (unweighted pair-group method using centroids, UPGMC, centroid sorting, weighted-group method). Distance data are squared unless you specify the NOSQUARE option.

COMPLETE | COM

requests complete linkage (furthest neighbor, maximum method, diameter method, rank order typal analysis). To reduce distortion of clusters by outliers, the TRIM= option is recommended.

DENSITY | DEN
: requests density linkage, which is a class of clustering methods using nonparametric probability density estimation. You must also specify either the K=, R=, or HYBRID option to indicate the type of density estimation to be used. See also the MODE= and DIM= options in this section.

EML
: requests maximum-likelihood hierarchical clustering for mixtures of spherical multivariate normal distributions with equal variances but possibly unequal mixing proportions. Use METHOD=EML only with coordinate data. See the PENALTY= option for details. The NONORM option does not affect the reported likelihood values but does affect other unrelated criteria. The EML method is much slower than the other methods in the CLUSTER procedure.

FLEXIBLE | FLE
: requests the Lance-Williams flexible-beta method. See the BETA= option in this section.

MCQUITTY | MCQ
: requests McQuitty’s similarity analysis (weighted average linkage, weighted pair-group method using arithmetic averages, WPGMA).

MEDIAN | MED
: requests Gower’s median method (weighted pair-group method using centroids, WPGMC). Distance data are squared unless you specify the NOSQUARE option.

SINGLE | SIN
: requests single linkage (nearest neighbor, minimum method, connectedness method, elementary linkage analysis, or dendritic method). To reduce chaining, you can use the TRIM= option with METHOD=SINGLE.

TWOSTAGE | TWO
: requests two-stage density linkage. You must also specify the K=, R=, or HYBRID option to indicate the type of density estimation to be used. See also the MODE= and DIM= options in this section.

WARD | WAR
: requests Ward’s minimum-variance method (error sum of squares, trace W). Distance data are squared unless you specify the NOSQUARE option. To reduce distortion by outliers, the TRIM= option is recommended. See the NONORM option.

Table 29.1 summarizes the options in the PROC CLUSTER statement.

Table 29.1 PROC CLUSTER Statement Options

Option	Description
Specify input and output data sets	
DATA=	specifies input data set
OUTTREE=	creates output data set
Specify clustering methods	

METHOD=	specifies clustering method
BETA=	specifies beta value for flexible beta method
MODE=	specifies the minimum number of members for modal clusters
PENALTY=	specifies the penalty coefficient for maximum likelihood
HYBRID	specifies Wong’s hybrid clustering method
Control data processing prior to clustering	
NOEIGEN	suppresses computation of eigenvalues
NONORM	suppresses normalizing of distances
NOSQUARE	suppresses squaring of distances
STANDARD	standardizes variables
TRIM=	omits points with low probability densities
Control density estimation	
K=	specifies number of neighbors for kth-nearest-neighbor density estimation
R=	specifies radius of sphere of support for uniform-kernel density estimation
Ties	
NOTIE	suppresses checking for ties
Control display of the cluster history	
CCC	displays cubic clustering criterion
NOID	suppresses display of ID values
PRINT=	specifies number of generations to display
PSEUDO	displays pseudo F and t^2 statistics
RMSSTD	displays root mean square standard deviation
RSQUARE	displays R square and semipartial R square
Control other aspects of output	
NOPRINT	suppresses display of all output
SIMPLE	displays simple summary statistics
PLOTS=	specifies ODS graphics details

The following list provides details on these options.

BETA=*n*

specifies the beta parameter for METHOD=FLEXIBLE. The value of $n$should be less than 1, usually between 0 and -1. By default, BETA=-0.25. Milligan (1987) suggests a somewhat smaller value, perhaps -0.5, for data with many outliers.

CCC

displays the cubic clustering criterion and approximate expected R square under the uniform null hypothesis (Sarle 1983). The statistics associated with the RSQUARE option, R square and semipartial R square, are also displayed. The CCC option applies only to coordinate data. The CCC option is not appropriate with METHOD=SINGLE because of the method’s tendency to chop off tails of distributions. Computation of the CCC requires the eigenvalues of the covariance matrix. If the number of variables is large, computing the eigenvalues requires much computer time and memory.

DATA=*SAS-data-set*

names the input data set containing observations to be clustered. By default, the procedure uses the most recently created SAS data set. If the data set is TYPE=DISTANCE, the data are interpreted as a distance matrix; the number of variables must equal the number of observations in the data set or in each BY group. The distances are assumed to be Euclidean, but the procedure accepts other types of distances or dissimilarities. If the data set is not TYPE=DISTANCE, the data are interpreted as coordinates in a Euclidean space, and Euclidean distances are computed. For more about TYPE=DISTANCE data sets, see Appendix A, Special SAS Data Sets.

You cannot use a TYPE=CORR data set as input to PROC CLUSTER, since the procedure uses dissimilarity measures. Instead, you can use a DATA step or the IML procedure to extract the correlation matrix from a TYPE=CORR data set and transform the values to dissimilarities such as $1-r$or $1-r^2$, where $r$is the correlation.

All methods produce the same results when used with coordinate data as when used with Euclidean distances computed from the coordinates. However, the DIM= option must be used with distance data if you specify METHOD=TWOSTAGE or METHOD=DENSITY or if you specify the TRIM= option.

Certain methods that are most naturally defined in terms of coordinates require ***squared*** Euclidean distances to be used in the combinatorial distance formulas (Lance and Williams 1967). For this reason, distance data are automatically squared when used with METHOD=AVERAGE, METHOD=CENTROID, METHOD=MEDIAN, or METHOD=WARD. If you want the combinatorial formulas to be applied to the (unsquared) distances with these methods, use the NOSQUARE option.

DIM=*n*

specifies the dimensionality used when computing density estimates with the TRIM= option, METHOD=DENSITY, or METHOD=TWOSTAGE. The values of n must be greater than or equal to 1. The default is the number of variables if the data are coordinates; the default is 1 if the data are distances.

HYBRID

requests Wong’s (1982) hybrid clustering method in which density estimates are computed from a preliminary cluster analysis using the k-means method. The DATA= data set must contain means, frequencies, and root mean square standard deviations of the preliminary clusters (see the FREQ and RMSSTD statements). To use HYBRID, you must use either a FREQ statement or a DATA= data set that contains a *_FREQ_* variable, and you must also use either an RMSSTD statement or a DATA= data set that contains an *_RMSSTD_* variable.

The MEAN= data set produced by the FASTCLUS procedure is suitable for input to the CLUSTER procedure for hybrid clustering. Since this data set contains *_FREQ_* and *_RMSSTD_* variables, you can use it as input and then omit the FREQ and RMSSTD statements.

You must specify either METHOD=DENSITY or METHOD=TWOSTAGE with the HYBRID option. You cannot use this option in combination with the TRIM=, K=, or R= option.

K=*n*

specifies the number of neighbors to use for kth-nearest-neighbor density estimation (Silverman 1986, pp. 19–21 and 96–99). The number of neighbors (n) must be at least two but less than the number of observations. See the MODE= option, which follows.

Density estimation is used with the TRIM=, METHOD=DENSITY, and METHOD=TWOSTAGE options.

MODE=*n*

specifies that, when two clusters are joined, each must have at least $n$members in order for either cluster to be designated a modal cluster. If you specify MODE=1, each cluster must also have a maximum density greater than the fusion density in order for either cluster to be designated a modal cluster.

Use the MODE= option only with METHOD=DENSITY or METHOD=TWOSTAGE. With METHOD=TWOSTAGE, the MODE= option affects the number of modal clusters formed. With METHOD=DENSITY, the MODE= option does not affect the clustering process but does determine the number of modal clusters reported on the output and identified by the *_MODE_* variable in the output data set.

If you specify the K= option, the default value of MODE= is the same as the value of K= because the use of kth-nearest-neighbor density estimation limits the resolution that can be obtained for clusters with fewer than $k$members. If you do not specify the K= option, the default is MODE=2.

If you specify MODE=0, the default value is used instead of 0.

If you specify a FREQ statement or if a *_FREQ_* variable appears in the input data set, the MODE= value is compared with the number of actual observations in the clusters being joined, not with the sum of the frequencies in the clusters.

NOEIGEN

suppresses computation of the eigenvalues of the covariance matrix and substitutes the variances of the variables for the eigenvalues when computing the cubic clustering criterion. The NOEIGEN option saves time if the number of variables is large, but it should be used only if the variables are nearly uncorrelated. If you specify the NOEIGEN option and the variables are highly correlated, the cubic clustering criterion might be very liberal. The NOEIGEN option applies only to coordinate data.

NOID

suppresses the display of ID values for the clusters joined at each generation of the cluster history.

NONORM

prevents the distances from being normalized to unit mean or unit root mean square with most methods. With METHOD=WARD, the NONORM option prevents the between-cluster sum of squares from being normalized by the total sum of squares to yield a squared semipartial correlation. The NONORM option does not affect the reported likelihood values with METHOD=EML, but it does affect other unrelated criteria, such as the *_DIST_* variable.

NOPRINT

suppresses the display of all output. Note that this option temporarily disables the Output Delivery System (ODS). For more information, see Chapter 20, Using the Output Delivery System.

NOSQUARE

prevents input distances from being squared with METHOD=AVERAGE, METHOD=CENTROID, METHOD=MEDIAN, or METHOD=WARD.

If you specify the NOSQUARE option with distance data, the data are assumed to be squared Euclidean distances for computing R-square and related statistics defined in a Euclidean coordinate system.

If you specify the NOSQUARE option with coordinate data with METHOD=CENTROID, METHOD=MEDIAN, or METHOD=WARD, then the combinatorial formula is applied to unsquared Euclidean distances. The resulting cluster distances do not have their usual Euclidean interpretation and are therefore labeled "False" in the output.

NOTIE

prevents PROC CLUSTER from checking for ties for minimum distance between clusters at each generation of the cluster history. If your data are measured with such precision that ties are unlikely, then you can specify the NOTIE option to reduce slightly the time and space required by the procedure. See the section Ties for more information.

OUTTREE=*SAS-data-set*

creates an output data set that can be used by the TREE procedure to draw a tree diagram. You must give the data set a two-level name to save it. See ***SAS***

Language Reference: Concepts for a discussion of permanent data sets. If you omit the OUTTREE= option, the data set is named by using the DATAn convention and is not permanently saved. If you do not want to create an output data set, use OUTTREE=_NULL_.

PENALTY=*p*

specifies the penalty coefficient used with METHOD=EML. See the section Clustering Methods for more information. Values for $p$must be greater than zero. By default, PENALTY=2.

PLOTS <(*global-plot-options*)> <= *plot-request* >
PLOTS <(*global-plot-options*)> <= (*plot-request* <... *plot-request* >)>

controls the plots produced through ODS Graphics.

PROC CLUSTER can produce line plots of the cubic clustering criterion, the pseudo Fstatistic, and the pseudo t^2statistic from the cluster history table. These statistics are useful for estimating the number of clusters. Each statistic is plotted against the number of clusters.

To obtain ODS Graphics plots from PROC CLUSTER, you must do two things. First, enable ODS Graphics before running PROC CLUSTER. For example:

```
ods graphics on;

proc cluster plots=all;
run;

ods graphics off;
```

Second, request that PROC CLUSTER compute the desired statistics by specifying the CCC or PSEUDO options, or by specifying the statistics in a ***plot-request*** in the PLOT option. PROC CLUSTER might be unable to compute the statistics in some cases; for details, see the CCC and PSEUDO options. If a statistic cannot be computed, it cannot be plotted. PROC CLUSTER plots all of these statistics that are computed unless you tell it specifically what to plot using PLOTS=.

The maximum number of clusters shown in all the plots is the minimum of the following quantities:

the number of observations

the value of the PRINT= option, if that option is specified

the maximum number of clusters for which CCC is computed, if CCC is plotted
The ***global-plot-options*** apply to all plots generated by the CLUSTER procedure. The global plot options are as follows:

UNPACKPANELS

breaks a plot that is otherwise paneled into plots separate plots for each statistic. This option can be abbreviated as UNPACK.

ONLY
has no effect, but is accepted for consistency with other procedures.

The following ***plot-requests*** can be specified:

ALL
implicitly specifies the CCC and PSEUDO options and, if possible, produces all three plots.

NONE
suppresses all plots.

CCC
implicitly specifies the CCC option and, if possible, plots the cubic clustering criterion against the number of clusters.

PSEUDO
implicitly specifies the PSEUDO option and, if possible, plots the pseudo F statistic and the pseudo t^2 statistic against the number of clusters.

PSF
implicitly specifies the PSEUDO option and, if possible, plots the pseudo F statistic against the number of clusters.

PST2
implicitly specifies the PSEUDO option and, if possible, plots the pseudo t^2 statistic against the number of clusters.

When you specify only one ***plot-request***, you can omit the parentheses around the ***plot-request***. You can specify one or more of the CCC, PSEUDO, PSF, or PST2 plot requests in the same PLOT option. For example, all of the following are valid:

```
   PROC CLUSTER PLOTS=(CCC PST2);
   PROC CLUSTER PLOTS=(PSF);
   PROC CLUSTER PLOTS=PSF;
```

The first statement plots both the cubic clustering criterion and the pseudo t^2 statistic, while the second and third statements plot the pseudo F statistic only.

The names of the graphs that PROC CLUSTER generates are listed in Table 29.5, along with the required statements and options.

PRINT=*n* | P=*n*

specifies the number of generations of the cluster history to display. The P= option displays the latest n generations; for example, P=5 displays the cluster history from 1 cluster through 5 clusters. The value of P= must be a nonnegative integer. The default is to display all generations. Specify PRINT=0 to suppress the cluster history.

PSEUDO

displays pseudo F and t^2 statistics. This option is effective only when the data are coordinates or when METHOD=AVERAGE, METHOD=CENTROID, or METHOD=WARD is specified. See the section Miscellaneous Formulas for more information. The PSEUDO option is not appropriate with METHOD=SINGLE because of the method’s tendency to chop off tails of distributions.

R=*n*

specifies the radius of the sphere of support for uniform-kernel density estimation (Silverman 1986, pp. 11–13 and 75–94). The value of R= must be greater than zero.

Density estimation is used with the TRIM=, METHOD=DENSITY, and METHOD=TWOSTAGE options.

RMSSTD

displays the root mean square standard deviation of each cluster. This option is effective only when the data are coordinates or when METHOD=AVERAGE, METHOD=CENTROID, or METHOD=WARD is specified.

See the section Miscellaneous Formulas for more information.

RSQUARE | RSQ

displays the R square and semipartial R square. This option is effective only when the data are coordinates or when METHOD=AVERAGE or METHOD=CENTROID is specified. The R square and semipartial R square statistics are always displayed with METHOD=WARD. See the section Miscellaneous Formulas for more information..

SIMPLE | S

displays means, standard deviations, skewness, kurtosis, and a coefficient of bimodality. The SIMPLE option applies only to coordinate data. See the section Miscellaneous Formulas for more information.

STANDARD | STD

standardizes the variables to mean 0 and standard deviation 1. The STANDARD option applies only to coordinate data.

TRIM=*p*

omits points with low estimated probability densities from the analysis. Valid values for the TRIM= option are $0 \leq p < 100$. If $p < 1$, then $p$is the proportion of observations omitted. If $p \geq 1$, then $p$is interpreted as a percentage. A specification of TRIM=10, which trims 10% of the points, is a reasonable value for many data sets. Densities are estimated by the kth-nearest-neighbor or uniform-kernel method. Trimmed points are indicated by a negative value of the *_FREQ_* variable in the OUTTREE= data set.

You must use either the K= or R= option when you use TRIM=. You cannot use the HYBRID option in combination with TRIM=, so you might want to use the DIM= option instead. If you specify the STANDARD option in combination with TRIM=, the variables are standardized both before and after trimming.

The TRIM= option is useful for removing outliers and reducing chaining. Trimming is highly recommended with METHOD=WARD or METHOD=COMPLETE because clusters from these methods can be severely distorted by outliers. Trimming is also valuable with METHOD=SINGLE since single linkage is the method most susceptible to chaining. Most other methods also benefit from trimming. However, trimming is unnecessary with METHOD=TWOSTAGE or METHOD=DENSITY when kth-nearest-neighbor density estimation is used.

Use of the TRIM= option can spuriously inflate the cubic clustering criterion and the pseudo $F$and t^2statistics. Trimming only outliers improves the accuracy of the statistics, but trimming saddle regions between clusters yields excessively large values.

BY Statement

BY *variables* ;

You can specify a BY statement with PROC CLUSTER to obtain separate analyses on observations in groups defined by the BY variables. When a BY statement appears, the procedure expects the input data set to be sorted in order of the BY variables.

If your input data set is not sorted in ascending order, use one of the following alternatives:

- Sort the data by using the SORT procedure with a similar BY statement.
- Specify the BY statement option NOTSORTED or DESCENDING in the BY statement for the CLUSTER procedure. The NOTSORTED option does not mean that the data are unsorted but rather that the data are arranged in groups (according to values of the BY variables) and that these groups are not necessarily in alphabetical or increasing numeric order.
- Create an index on the BY variables by using the DATASETS procedure.

For more information about the BY statement, see ***SAS Language Reference: Concepts***.

For more information about the DATASETS procedure, see the *Base SAS Procedures Guide*.

COPY Statement

COPY *variables* ;

The variables in the COPY statement are copied from the input data set to the OUTTREE= data set. Observations in the OUTTREE= data set that represent clusters of more than one observation from the input data set have missing values for the COPY variables.

FREQ Statement

FREQ *variable* ;

If one variable in the input data set represents the frequency of occurrence for other values in the observation, specify the variable’s name in a FREQ statement. PROC CLUSTER then treats the data set as if each observation appeared $n$times, where $n$is the value of the FREQ variable for the observation. Noninteger values of the FREQ variable are truncated to the largest integer less than the FREQ value.

If you omit the FREQ statement but the DATA= data set contains a variable called *_FREQ_*, then frequencies are obtained from the *_FREQ_* variable. If neither a FREQ statement nor an *_FREQ_* variable is present, each observation is assumed to have a frequency of one.

If each observation in the DATA= data set represents a cluster (for example, clusters formed by PROC FASTCLUS), the variable specified in the FREQ statement should give the number of original observations in each cluster.

If you specify the RMSSTD statement, a FREQ statement is required. A FREQ statement or *_FREQ_* variable is required when you specify the HYBRID option.

With most clustering methods, the same clusters are obtained from a data set with a FREQ variable as from a similar data set without a FREQ variable, if each observation is repeated as many times as the value of the FREQ variable in the first data set. The FLEXIBLE method can yield different results due to the nature of the combinatorial formula. The DENSITY and TWOSTAGE methods are also exceptions because two identical observations can be absorbed one at a time by a cluster with a higher density. If you are using a FREQ statement with either the DENSITY or TWOSTAGE method, see the MODE=option for details.

ID Statement

ID *variable* ;

The values of the ID variable identify observations in the displayed cluster history and in the OUTTREE= data set. If the ID statement is omitted, each observation is denoted by *OBn*, where *n* is the observation number.

RMSSTD Statement

RMSSTD *variable* ;

If the coordinates in the DATA= data set represent cluster means (for example, formed by the FASTCLUS procedure), you can obtain accurate statistics in the cluster histories for METHOD=AVERAGE, METHOD=CENTROID, or METHOD=WARD if the data set contains both of the following:

- a variable giving the number of original observations in each cluster (see the discussion of the FREQ statement earlier in this chapter)
- a variable giving the root mean squared standard deviation of each cluster

Specify the name of the variable containing root mean squared standard deviations in the RMSSTD statement. If you specify the RMSSTD statement, you must also specify a FREQ statement.

If you omit the RMSSTD statement but the DATA= data set contains a variable called *_RMSSTD_*, then the root mean squared standard deviations are obtained from the *_RMSSTD_* variable.

An RMSSTD statement or *_RMSSTD_* variable is required when you specify the HYBRID option.

A data set created by PROC FASTCLUS, using the MEAN= option, contains *_FREQ_* and *_RMSSTD_* variables, so you do not have to use FREQ and RMSSTD statements when using such a data set as input to the CLUSTER procedure.

VAR Statement

VAR *variables* ;

The VAR statement lists numeric variables to be used in the cluster analysis. If you omit the VAR statement, all numeric variables not listed in other statements are used.

Details: CLUSTER Procedure

Clustering Methods

The following notation is used, with lowercase symbols generally pertaining to observations and uppercase symbols pertaining to clusters:

n

number of observations

v

number of variables if data are coordinates

G

number of clusters at any given level of the hierarchy

x_i or $\mathbf{x}_i$

ith observation (row vector if coordinate data)

C_K

Kth cluster, subset of $\{1, 2, \ldots, n\}$

N_K

number of observations in C_K

$\bar{\mathbf{x}}$

sample mean vector

$\bar{\mathbf{x}}_K$

mean vector for cluster C_K

$\|\mathbf{x}\|$

Euclidean length of the vector $\mathbf{x}$—that is, the square root of the sum of the squares of the elements of $\mathbf{x}$

T

$\sum_{i=1}^{n} \|\mathbf{x}_i - \bar{\mathbf{x}}\|^2$

W_K

$\sum_{i \in C_k} \|\mathbf{x}_i - \bar{\mathbf{x}}_K\|^2$

P_G

$\sum W_J$, where summation is over the Gclusters at the Gth level of the hierarchy

B_{KL}

$W_M - W_K - W_L$if $C_M = C_K \cup C_L$

$d(\mathbf{x}, \mathbf{y})$

any distance or dissimilarity measure between observations or vectors $\mathbf{x}$and $\mathbf{y}$

D_{KL}

any distance or dissimilarity measure between clusters $C_K$and C_L

The distance between two clusters can be defined either directly or combinatorially (Lance and Williams 1967)—that is, by an equation for updating a distance matrix when two clusters are joined. In all of the following combinatorial formulas, it is assumed that clusters $C_K$and $C_L$are merged to form C_M, and the formula gives the distance between the new cluster $C_M$and any other cluster C_J.

For an introduction to most of the methods used in the CLUSTER procedure, see Massart and Kaufman (1983).

Average Linkage

The following method is obtained by specifying METHOD=AVERAGE. The distance between two clusters is defined by

$$D_{KL} = \frac{1}{N_K N_L} \sum_{i \in C_K} \sum_{j \in C_L} d(x_i, x_j)$$

If $d(\mathbf{x}, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|^2$, then

$$D_{KL} = \|\bar{\mathbf{x}}_K - \bar{\mathbf{x}}_L\|^2 + \frac{W_K}{N_K} + \frac{W_L}{N_L}$$

The combinatorial formula is

$$D_{JM} = \frac{N_K D_{JK} + N_L D_{JL}}{N_M}$$

In average linkage the distance between two clusters is the average distance between pairs of observations, one in each cluster. Average linkage tends to join clusters with small variances, and it is slightly biased toward producing clusters with the same variance.

Average linkage was originated by Sokal and Michener (1958).

Centroid Method

The following method is obtained by specifying METHOD=CENTROID. The distance between two clusters is defined by

$$D_{KL} = \|\bar{\mathbf{x}}_K - \bar{\mathbf{x}}_L\|^2$$

If $d(\mathbf{x},\mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|^2$, then the combinatorial formula is

$$D_{JM} = \frac{N_K D_{JK} + N_L D_{JL}}{N_M} - \frac{N_K N_L D_{KL}}{N_M^2}$$

In the centroid method, the distance between two clusters is defined as the (squared) Euclidean distance between their centroids or means. The centroid method is more robust to outliers than most other hierarchical methods but in other respects might not perform as well as Ward’s method or average linkage (Milligan 1980).

The centroid method was originated by Sokal and Michener (1958).

Complete Linkage

The following method is obtained by specifying METHOD=COMPLETE. The distance between two clusters is defined by

$$D_{KL} = \max_{i \in C_K} \max_{j \in C_L} d(x_i, x_j)$$

The combinatorial formula is

$$D_{JM} = \max(D_{JK}, D_{JL})$$

In complete linkage, the distance between two clusters is the maximum distance between an observation in one cluster and an observation in the other cluster. Complete linkage is strongly biased toward producing clusters with roughly equal diameters, and it can be severely distorted by moderate outliers (Milligan 1980).

Complete linkage was originated by Sorensen (1948).

Density Linkage

The phrase ***density linkage*** is used here to refer to a class of clustering methods that use nonparametric probability density estimates (for example, Hartigan 1975, pp. 205–212; Wong 1982; Wong and Lane 1983). Density linkage consists of two steps:

1. A new dissimilarity measure, d^*, based on density estimates and adjacencies is computed. If $x_i$and $x_j$are adjacent (the definition of ***adjacency*** depends on the method of density estimation), then $d^*(x_i,x_j)$is the reciprocal of an estimate of the density midway between $x_i$and x_j; otherwise, $d^*(x_i,x_j)$is infinite.

2. A single linkage cluster analysis is performed using d^*.

The CLUSTER procedure supports three types of density linkage: the kth-nearest-neighbor method, the uniform-kernel method, and Wong’s hybrid method. These are obtained by using METHOD=DENSITY and the K=, R=, and HYBRID options, respectively.

kth-Nearest-Neighbor Method

The kth-nearest-neighbor method (Wong and Lane 1983) uses kth-nearest-neighbor density estimates. Let $r_k(x)$be the distance from point $x$to the kth-nearest observation, where $k$is the value specified for the K= option. Consider a closed sphere centered at $x$with radius $r_k(x)$. The estimated density at x, $f(x)$, is the proportion of observations within the sphere divided by the volume of the sphere. The new dissimilarity measure is computed as

$$d^*(x_i,x_j) = \begin{cases} \frac{1}{2}\left(\frac{1}{f(x_i)} + \frac{1}{f(x_j)}\right) & \text{if } d(x_i,x_j) \le \max(r_k(x_i), r_k(x_j)) \\ \infty & \text{otherwise} \end{cases}$$

Wong and Lane (1983) show that kth-nearest-neighbor density linkage is strongly set consistent for high-density (density-contour) clusters if $k$is chosen such that $k/n \rightarrow 0$and $k/\ln(n) \rightarrow \infty$as $n \rightarrow \infty$. Wong and Schaack (1982) discuss methods for estimating the number of population clusters by using kth-nearest-neighbor clustering.

Uniform-Kernel Method

The uniform-kernel method uses uniform-kernel density estimates. Let $r$be the value specified for the R= option. Consider a closed sphere centered at point $x$with radius r. The estimated density at x, $f(x)$, is the proportion of observations within the sphere divided by the volume of the sphere. The new dissimilarity measure is computed as

$$d^*(x_i,x_j) = \begin{cases} \frac{1}{2}\left(\frac{1}{f(x_i)} + \frac{1}{f(x_j)}\right) & \text{if } d(x_i,x_j) \le r \\ \infty & \text{otherwise} \end{cases}$$

Wong’s Hybrid Method

Wong’s (1982) hybrid clustering method uses density estimates based on a preliminary cluster analysis by the k-means method. The preliminary clustering can be done by the FASTCLUS procedure, by using the MEAN= option to create a data set containing cluster means, frequencies, and root mean squared standard deviations. This data set is used as input to the

CLUSTER procedure, and the HYBRID option is specified with METHOD=DENSITY to request the hybrid analysis. The hybrid method is appropriate for very large data sets but should not be used with small data sets—say, than those with fewer than 100 observations in the original data. The term ***preliminary cluster*** refers to an observation in the DATA= data set.

For preliminary cluster C_K, $N_K$and $W_K$are obtained from the input data set, as are the cluster means or the distances between the cluster means. Preliminary clusters $C_K$and $C_L$are considered adjacent if the midpoint between $\bar{x}_K$and $\bar{\mathbf{x}}_L$is closer to either $\bar{\mathbf{x}}_K$or $\bar{\mathbf{x}}_L$than to any other preliminary cluster mean or, equivalently, if $d^2(\bar{\mathbf{x}}_K,\bar{\mathbf{x}}_L) < d^2(\bar{\mathbf{x}}_K,\bar{\mathbf{x}}_M) + d^2(\bar{\mathbf{x}}_L,\bar{\mathbf{x}}_M)$for all other preliminary clusters C_M, $M \neq K$or L. The new dissimilarity measure is computed as

$$d^*(\bar{\mathbf{x}}_K,\bar{\mathbf{x}}_L) = \begin{cases} \frac{\left(W_K+W_L+\frac{1}{4}(N_K+N_L)d^2(\bar{\mathbf{x}}_K,\bar{\mathbf{x}}_L)\right)^{\frac{v}{2}}}{(N_K+N_L)^{1+\frac{v}{2}}} & \text{if } C_K \text{ and } C_L \text{ are adjacent} \\ \infty & \text{otherwise} \end{cases}$$

Using the K= and R= Options

The values of the K= and R= options are called ***smoothing parameters***. Small values of K= or R= produce jagged density estimates and, as a consequence, many modes. Large values of K= or R= produce smoother density estimates and fewer modes. In the hybrid method, the smoothing parameter is the number of clusters in the preliminary cluster analysis. The number of modes in the final analysis tends to increase as the number of clusters in the preliminary analysis increases. Wong (1982) suggests using $n^{0.3}$preliminary clusters, where $n$is the number of observations in the original data set. There is no rule of thumb for selecting K= values. For all types of density linkage, you should repeat the analysis with several different values of the smoothing parameter (Wong and Schaack 1982).

There is no simple answer to the question of which smoothing parameter to use (Silverman 1986, pp. 43–61, 84–88, and 98–99). It is usually necessary to try several different smoothing parameters. A reasonable first guess for the R= option in many coordinate data sets is given by

$$\left[\frac{2^{v+2}(v+2)\Gamma(\frac{v}{2}+1)}{nv^2}\right]^{\frac{1}{v+4}} \sqrt{\sum_{l=1}^{v} s_l^2}$$

where $s_l^2$is the standard deviation of the lth variable. The estimate for R= can be computed in a DATA step by using the GAMMA function for Γ. This formula is derived under the assumption that the data are sampled from a multivariate normal distribution and tends, therefore, to be too large (oversmooth) if the true distribution is multimodal. Robust estimates of the standard deviations can be preferable if there are outliers. If the data are distances, the factor $\sum s_l^2$can be replaced by an average (mean, trimmed mean, median, root mean square, and so on) distance divided by $\sqrt{2}$. To prevent outliers from appearing as separate clusters, you can also specify K=2, or more generally K=m, $m \geq 2$, which in most cases forces clusters to have at least $m$members

If the variables all have unit variance (for example, if the STANDARD option is used), Table 29.2 can be used to obtain an initial guess for the R= option.

Since infinite d^*values occur in density linkage, the final number of clusters can exceed one when there are wide gaps between the clusters or when the smoothing parameter results in little smoothing.

Density linkage applies no constraints to the shapes of the clusters and, unlike most other hierarchical clustering methods, is capable of recovering clusters with elongated or irregular shapes. Since density linkage uses less prior knowledge about the shape of the clusters than do methods restricted to compact clusters, density linkage is less effective at recovering compact clusters from small samples than are methods that always recover compact clusters, regardless of the data.

Table 29.2 Reasonable First Guess for the R= Option for Standardized Data

Number of	**Number of Variables**									
Observations	1	2	3	4	5	6	7	8	9	10
20	1.01	1.36	1.77	2.23	2.73	3.25	3.81	4.38	4.98	5.60
35	0.91	1.24	1.64	2.08	2.56	3.08	3.62	4.18	4.77	5.38
50	0.84	1.17	1.56	1.99	2.46	2.97	3.50	4.06	4.64	5.24
75	0.78	1.09	1.47	1.89	2.35	2.85	3.38	3.93	4.50	5.09
100	0.73	1.04	1.41	1.82	2.28	2.77	3.29	3.83	4.40	4.99
150	0.68	0.97	1.33	1.73	2.18	2.66	3.17	3.71	4.27	4.85
200	0.64	0.93	1.28	1.67	2.11	2.58	3.09	3.62	4.17	4.75
350	0.57	0.85	1.18	1.56	1.98	2.44	2.93	3.45	4.00	4.56
500	0.53	0.80	1.12	1.49	1.91	2.36	2.84	3.35	3.89	4.45
750	0.49	0.74	1.06	1.42	1.82	2.26	2.74	3.24	3.77	4.32
1000	0.46	0.71	1.01	1.37	1.77	2.20	2.67	3.16	3.69	4.23
1500	0.43	0.66	0.96	1.30	1.69	2.11	2.57	3.06	3.57	4.11
2000	0.40	0.63	0.92	1.25	1.63	2.05	2.50	2.99	3.49	4.03

EML

The following method is obtained by specifying METHOD=EML. The distance between two clusters is given by

$$D_{KL} = nv \ln\left(1 + \frac{B_{KL}}{P_G}\right) - 2\left(N_M \ln(N_M) - N_K \ln(N_K) - N_L \ln(N_L)\right)$$

The EML method joins clusters to maximize the likelihood at each level of the hierarchy under the following assumptions:

multivariate normal mixture

equal spherical covariance matrices

unequal sampling probabilities

The EML method is similar to Ward’s minimum-variance method but removes the bias toward equal-sized clusters. Practical experience has indicated that EML is somewhat biased toward unequal-sized clusters. You can specify the PENALTY= option to adjust the degree of bias. If you specify PENALTY=p, the formula is modified to

$$D_{KL} = nv \ln\left(1 + \frac{B_{KL}}{P_G}\right) - p(N_M \ln(N_M) - N_K \ln(N_K) - N_L \ln(N_L))$$

The EML method was derived by W. S. Sarle of SAS Institute from the maximum likelihood formula obtained by Symons (1981, p. 37, Equation 8) for disjoint clustering. There are currently no other published references on the EML method.

Flexible-Beta Method

The following method is obtained by specifying METHOD=FLEXIBLE. The combinatorial formula is

$$D_{JM} = (D_{JK} + D_{JL})\frac{1-b}{2} + D_{KL}b$$

where $b$is the value of the BETA= option, or $-0.25$by default.

The flexible-beta method was developed by Lance and Williams (1967); see also Milligan (1987).

McQuitty’s Similarity Analysis

The following method is obtained by specifying METHOD=MCQUITTY. The combinatorial formula is

$$D_{JM} = \frac{D_{JK} + D_{JL}}{2}$$

The method was independently developed by Sokal and Michener (1958) and McQuitty (1966).

Median Method

The following method is obtained by specifying METHOD=MEDIAN. If $d(\mathbf{x},\mathbf{y}) = \|\mathbf{x}-\mathbf{y}\|^2$, then the combinatorial formula is

$$D_{JM} = \frac{D_{JK} + D_{JL}}{2} - \frac{D_{KL}}{4}$$

The median method was developed by Gower (1967).

Single Linkage

The following method is obtained by specifying METHOD=SINGLE. The distance between two clusters is defined by

$$D_{KL} = \min_{i \in C_K} \min_{j \in C_L} d(x_i, x_j)$$

The combinatorial formula is

$$D_{JM} = \min(D_{JK}, D_{JL})$$

In single linkage, the distance between two clusters is the minimum distance between an observation in one cluster and an observation in the other cluster. Single linkage has many desirable theoretical properties (Jardine and Sibson 1971; Fisher and Van Ness 1971; Hartigan 1981) but has fared poorly in Monte Carlo studies (for example, Milligan 1980). By imposing no constraints on the shape of clusters, single linkage sacrifices performance in the recovery of compact clusters in return for the ability to detect elongated and irregular clusters. You must also recognize that single linkage tends to chop off the tails of distributions before separating the main clusters (Hartigan 1981). The notorious chaining tendency of single linkage can be alleviated by specifying the TRIM= option (Wishart 1969, pp. 296–298).

Density linkage and two-stage density linkage retain most of the virtues of single linkage while performing better with compact clusters and possessing better asymptotic properties (Wong and Lane 1983).

Single linkage was originated by Florek et al. (1951a, 1951b) and later reinvented by McQuitty (1957) and Sneath (1957).

Two-Stage Density Linkage

If you specify METHOD=DENSITY, the modal clusters often merge before all the points in the tails have clustered. The option METHOD=TWOSTAGE is a modification of density linkage that ensures that all points are assigned to modal clusters before the modal clusters are permitted to join. The CLUSTER procedure supports the same three varieties of two-stage density linkage as of ordinary density linkage: kth-nearest neighbor, uniform kernel, and hybrid.

In the first stage, disjoint modal clusters are formed. The algorithm is the same as the single linkage algorithm ordinarily used with density linkage, with one exception: two clusters are joined only if at least one of the two clusters has fewer members than the number specified by the MODE= option. At the end of the first stage, each point belongs to one modal cluster.

In the second stage, the modal clusters are hierarchically joined by single linkage. The final number of clusters can exceed one when there are wide gaps between the clusters or when the smoothing parameter is small.

Each stage forms a tree that can be plotted by the TREE procedure. By default, the TREE procedure plots the tree from the first stage. To obtain the tree for the second stage, use the option HEIGHT=MODE in the PROC TREE statement. You can also produce a single tree diagram containing both stages, with the number of clusters as the height axis, by using the option HEIGHT=N in the PROC TREE statement. To produce an output data set from PROC TREE containing the modal clusters, use *_HEIGHT_* for the HEIGHT variable (the default) and specify LEVEL=0.

Two-stage density linkage was developed by W. S. Sarle of SAS Institute. There are currently no other published references on two-stage density linkage.

Ward’s Minimum-Variance Method

The following method is obtained by specifying METHOD=WARD. The distance between two clusters is defined by

$$D_{KL} = B_{KL} = \frac{\|\bar{\mathbf{x}}_K - \bar{\mathbf{x}}_L\|^2}{\frac{1}{N_K} + \frac{1}{N_L}}$$

If $d(\mathbf{x}, \mathbf{y}) = \frac{1}{2}\|\mathbf{x} - \mathbf{y}\|^2$, then the combinatorial formula is

$$D_{JM} = \frac{(N_J + N_K)D_{JK} + (N_J + N_L)D_{JL} - N_J D_{KL}}{N_J + N_M}$$

In Ward’s minimum-variance method, the distance between two clusters is the ANOVA sum of squares between the two clusters added up over all the variables. At each generation, the within-cluster sum of squares is minimized over all partitions obtainable by merging two clusters from the previous generation. The sums of squares are easier to interpret when they are divided by the total sum of squares to give proportions of variance (squared semipartial correlations).

Ward’s method joins clusters to maximize the likelihood at each level of the hierarchy under the following assumptions:

multivariate normal mixture

equal spherical covariance matrices

equal sampling probabilities

Ward’s method tends to join clusters with a small number of observations, and it is strongly biased toward producing clusters with roughly the same number of observations. It is also very sensitive to outliers (Milligan 1980).

Ward (1963) describes a class of hierarchical clustering methods including the minimum variance method.

Miscellaneous Formulas

The root mean squared standard deviation of a cluster $C_K$is

$$\text{RMSSTD} = \sqrt{\frac{W_K}{v(N_K - 1)}}$$

The R-square statistic for a given level of the hierarchy is

$$R^2 = 1 - \frac{P_G}{T}$$

The squared semipartial correlation for joining clusters $C_K$and $C_L$is

$$\text{semipartial } R^2 = \frac{B_{KL}}{T}$$

The bimodality coefficient is

$$b = \frac{m_3^2 + 1}{m_4 + \frac{3(n-1)^2}{(n-2)(n-3)}}$$

where $m_3$is skewness and $m_4$is kurtosis. Values of $b$greater than 0.555 (the value for a uniform population) can indicate bimodal or multimodal marginal distributions. The maximum of 1.0 (obtained for the Bernoulli distribution) is obtained for a population with only two distinct values. Very heavy-tailed distributions have small values of $b$regardless of the number of modes.

Formulas for the cubic-clustering criterion and approximate expected R square are given in Sarle (1983).

The pseudo Fstatistic for a given level is

$$\text{pseudo } F = \frac{\frac{T - P_G}{G - 1}}{\frac{P_G}{n - G}}$$

The pseudo t^2statistic for joining $C_K$and $C_L$is

$$\text{pseudo } t^2 = \frac{B_{KL}}{\frac{W_K + W_L}{N_K + N_L - 2}}$$

The pseudo $F$and t^2statistics can be useful indicators of the number of clusters, but they are ***not*** distributed as $F$and t^2random variables. If the data are independently sampled from a multivariate normal distribution with a scalar covariance matrix and if the clustering method allocates observations to clusters randomly (which no clustering method actually does), then the pseudo Fstatistic is distributed as an Frandom variable with $v(G - 1)$and $v(n - G)$degrees of freedom. Under the same assumptions, the pseudo t^2statistic is distributed as an Frandom variable with $v$and $v(N_K + N_L - 2)$degrees of freedom. The pseudo t^2statistic differs computationally from Hotelling's T^2in that the latter uses a general symmetric covariance matrix instead of a scalar covariance matrix. The pseudo Fstatistic was suggested by Calinski and Harabasz (1974). The pseudo t^2statistic is related to the $J_e(2)/J_e(1)$statistic of Duda and Hart (1973) by

$$\frac{J_e(2)}{J_e(1)} = \frac{W_K + W_L}{W_M} = \frac{1}{1 + \frac{t^2}{N_K + N_L - 2}}$$

See Milligan and Cooper (1985) and Cooper and Milligan (1988) regarding the performance of these statistics in estimating the number of population clusters. Conservative tests for the number of clusters using the pseudo $F$and t^2statistics can be obtained by the Bonferroni approach (Hawkins, Muller, and ten Krooden 1982, pp. 337–340).

Ultrametrics

A dissimilarity measure $d(x,y)$is called an ***ultrametric*** if it satisfies the following conditions:

$d(x,x) = 0$for all x

$d(x,y) \geq 0$for all x, y

$d(x,y) = d(y,x)$for all x, y

$d(x,y) \leq \max(d(x,z), d(y,z))$for all x, y, and z

Any hierarchical clustering method induces a dissimilarity measure on the observations—say, $h(x_i,x_j)$. Let $C_M$be the cluster with the fewest members that contains both $x_i$and x_j. Assume C_M was formed by joining $C_K$and C_L. Then define $h(x_i,x_j) = D_{KL}$.

If the fusion of $C_K$and $C_L$reduces the number of clusters from $g$to $g-1$, then define $D_{(g)} = D_{KL}$. Johnson (1967) shows that if

$$0 \leq D_{(n)} \leq D_{(n-1)} \leq \cdots \leq D_{(2)}$$

then $h(\cdot,\cdot)$is an ultrametric. A method that always satisfies this condition is said to be a ***monotonic*** or ***ultrametric clustering method***. All methods implemented in PROC CLUSTER except CENTROID, EML, and MEDIAN are ultrametric (Milligan 1979; Batagelj 1981).

Algorithms

Anderberg (1973) describes three algorithms for implementing agglomerative hierarchical clustering: stored data, stored distance, and sorted distance. The algorithms used by PROC CLUSTER for each method are indicated in Table 29.3. For METHOD=AVERAGE, METHOD=CENTROID, or METHOD=WARD, either the stored data or the stored distance algorithm can be used. For these methods, if the data are distances or if you specify the NOSQUARE option, the stored distance algorithm is used; otherwise, the stored data algorithm is used.

Table 29.3 Three Algorithms for Implementing Agglomerative Hierarchical Clustering

	Algorithm		
Stored	**Stored**	**Stored**	**Sorted**
Method	**Data**	**Distance**	**Distance**

AVERAGE	x	x	
CENTROID	x	x	
COMPLETE		x	
DENSITY			x
EML	x		
FLEXIBLE		x	
MCQUITTY		x	
MEDIAN		x	
SINGLE		x	
TWOSTAGE			x
WARD	x	x	

Computational Resources

The CLUSTER procedure stores the data (including the COPY and ID variables) in memory or, if necessary, on disk. If eigenvalues are computed, the covariance matrix is stored in memory. If the stored distance or sorted distance algorithm is used, the distances are stored in memory or, if necessary, on disk.

With coordinate data, the increase in CPU time is roughly proportional to the number of variables. The VAR statement should list the variables in order of decreasing variance for greatest efficiency.

For both coordinate and distance data, the dominant factor determining CPU time is the number of observations. For density methods with coordinate data, the asymptotic time requirements are somewhere between $n \ln(n)$and n^2, depending on how the smoothing parameter increases. For other methods except EML, time is roughly proportional to n^2. For the EML method, time is roughly proportional to n^3.

PROC CLUSTER runs much faster if the data can be stored in memory and, when the stored distance algorithm is used, if the distance matrix can be stored in memory as well. To estimate the bytes of memory needed for the data, use the following formula and round up to the nearest multiple of d.

$n(vd + 8d + i$

$+ i$ if density estimation or the

sorted distance algorithm is used

$+3d$	if stored data algorithm is used
$+3d$	if density estimation is used
$+$ max(8, length of ID variable)	if ID variable is used
$+$ length of ID variable	if ID variable is used
$+$ sum of lengths of COPY variables)	if COPY variables is used

where

n is the number of observations

v is the number of variables

d is the size of a C variable of type ***double***. For most computers, $d = 8$.

i is the size of a C variable of type ***int***. For most computers, $i = 4$.

The number of bytes needed for the distance matrix is $dn(n+1)/2$.

Missing Values

If the data are coordinates, observations with missing values are excluded from the analysis. If the data are distances, missing values are not permitted in the lower triangle of the distance matrix. The upper triangle is ignored. For more about TYPE=DISTANCE data sets, see Appendix A, Special SAS Data Sets.

TYPE=DISTANCE Data Sets

PROC DISTANCE creates a TYPE=DISTANCE or TYPE=SIMILAR data set, depending on the METHOD= option. TYPE=DISTANCE can be used as an input data set to PROC MODECLUS or PROC CLUSTER, but TYPE=SIMILAR cannot be used as an input to any procedures. The proximity measures are stored as a lower triangular matrix or a square matrix in the OUT= data set (depending on the SHAPE= option). See Chapter 32, The DISTANCE Procedure, for details. You can also create a TYPE=DISTANCE data set in a DATA step by reading or computing a lower triangular or symmetric matrix of dissimilarity values, such as a chart of mileage between cities. The number of observations must be equal to the number of variables used in the analysis. This type of data set is used as input by the CLUSTER and MODECLUS procedures. PROC CLUSTER ignores the upper triangular portion of a TYPE=DISTANCE data set and assumes that all main diagonal values are zero, even if they are missing. PROC MODECLUS uses the entire distance matrix and does not require the matrix to be symmetric. See Chapter 29, The CLUSTER Procedure, and Chapter 57, The MODECLUS Procedure, for examples and details.

Ties

At each level of the clustering algorithm, PROC CLUSTER must identify the pair of clusters with the minimum distance. Sometimes, usually when the data are discrete, there can be two or more pairs with the same minimum distance. In such cases the tie must be broken in some arbitrary way. If there are ties, then the results of the cluster analysis depend on the order of the observations in the data set. The presence of ties is reported in the SAS log and in the column of the cluster history labeled "Tie" unless the NOTIE option is specified.

PROC CLUSTER breaks ties as follows. Each cluster is identified by the smallest observation number among its members. For each pair of clusters, there is a smaller identification number and a larger identification number. If two or more pairs of clusters are tied for minimum distance between clusters, the pair that has the minimum larger identification number is merged. If there is a tie for minimum larger identification number, the pair that has the minimum smaller identification number is merged.

A tie means that the level in the cluster history at which the tie occurred and possibly some of the subsequent levels are not uniquely determined. Ties that occur early in the cluster history usually have little effect on the later stages. Ties that occur in the middle part of the cluster history are cause for further investigation. Ties that occur late in the cluster history indicate important indeterminacies.

The importance of ties can be assessed by repeating the cluster analysis for several different random permutations of the observations. The discrepancies at a given level can be examined by crosstabulating the clusters obtained at that level for all of the permutations. See Example 29.4 for details.

Size, Shape, and Correlation

In some biological applications, the organisms that are being clustered can be at different stages of growth. Unless it is the growth process itself that is being studied, differences in size among such organisms are not of interest. Therefore, distances among organisms should be computed in such a way as to control for differences in size while retaining information about differences in shape.

If coordinate data are measured on an interval scale, you can control for size by subtracting a measure of the overall size of each observation from each data item. For example, if no other direct measure of size is available, you could subtract the mean of each row of the data matrix, producing a row-centered coordinate matrix. An easy way to subtract the mean of each row is to use PROC STANDARD on the transposed coordinate matrix:

```
   proc transpose data= coordinate-datatype ;
   proc standard m=0;
   proc transpose out=row-centered-coordinate-data;
```

Another way to remove size effects from interval-scale coordinate data is to do a principal component analysis and discard the first component (Blackith and Reyment 1971).

If the data are measured on a ratio scale, you can control for size by dividing each observation by a measure of overall size; in this case, the geometric mean is a more natural measure of size than the arithmetic mean. However, it is often more meaningful to analyze the logarithms of ratio-scaled data, in which case you can subtract the arithmetic mean after taking logarithms. You must also consider the dimensions of measurement. For example, if you have measures of both length and weight, you might need to cube the measures of length or take the cube root of the weights. Various other complications can also arise in real applications, such as different growth rates for different parts of the body (Sneath and Sokal 1973).

Issues of size and shape are pertinent to many areas besides biology (for example, Hamer and Cunningham 1981). Suppose you have data consisting of subjective ratings made by several different raters. Some raters tend to give higher overall ratings than other raters. Some raters also tend to spread out their ratings over more of the scale than other raters. If it is impossible for you to adjust directly for rater differences, then distances should be computed in such a way as to control for differences both in size and variability. For example, if the data are considered to be measured on an interval scale, you can subtract the mean of each observation and divide by the standard deviation, producing a row-standardized coordinate matrix. With some clustering methods, analyzing squared Euclidean distances from a row-standardized coordinate matrix is equivalent to analyzing the matrix of correlations among rows, since squared Euclidean distance is an affine transformation of the correlation (Hartigan 1975, p. 64).

If you do an analysis of row-centered or row-standardized data, you need to consider whether the columns (variables) should be standardized before centering or standardizing the rows, after centering or standardizing the rows, or both before and after. If you standardize the columns after standardizing the rows, then strictly speaking you are not analyzing shape because the profiles are distorted by standardizing the columns; however, this type of double standardization might be necessary in practice to get reasonable results. It is not clear whether iterating the standardization of rows and columns can be of any benefit.

The choice of distance or correlation measure should depend on the meaning of the data and the purpose of the analysis. Simulation studies that compare distance and correlation measures are useless unless the data are generated to mimic data from your field of application. Conclusions drawn from artificial data cannot be generalized, because it is possible to generate data such that distances that include size effects work better or such that correlations work better.

You can standardize the rows of a data set by using a DATA step or by using the TRANSPOSE and STANDARD procedures. You can also use PROC TRANSPOSE and then have PROC CORR create a TYPE=CORR data set containing a correlation matrix. If you want to analyze a TYPE=CORR data set with PROC CLUSTER, you must use a DATA step to perform the following steps:

1. Set the data set TYPE= to DISTANCE.

2. Convert the correlations to dissimilarities by computing $1-r$, $\sqrt{1-r}$, $1-r^2$, or some other decreasing function.

3. Delete observations for which the variable *_TYPE_* does not have the value ’CORR’.

Output Data Set

The OUTTREE= data set contains one observation for each observation in the input data set, plus one observation for each cluster of two or more observations (that is, one observation for each node of the cluster tree). The total number of output observations is usually $2n-1$, where $n$is the number of input observations. The density methods can produce fewer output observations when the number of clusters cannot be reduced to one.

The label of the OUTTREE= data set identifies the type of cluster analysis performed and is automatically displayed when the TREE procedure is invoked.

The variables in the OUTTREE= data set are as follows:

the BY variables, if you use a BY statement

the ID variable, if you use an ID statement

the COPY variables, if you use a COPY statement

*_NAME_*, a character variable giving the name of the node. If the node is a cluster, the name is *CL*n, where $n$is the number of the cluster. If the node is an observation, the name is *OB*n, where $n$is the observation number. If the node is an observation and the ID statement is used, the name is the formatted value of the ID variable.

*_PARENT_*, a character variable giving the value of *_NAME_* of the parent of the node

*_NCL_*, the number of clusters

*_FREQ_*, the number of observations in the current cluster

*_HEIGHT_*, the distance or similarity between the last clusters joined, as defined in the section Clustering Methods. The variable *_HEIGHT_* is used by the TREE procedure as the default height axis. The label of the *_HEIGHT_* variable identifies the between-cluster distance measure. For METHOD=TWOSTAGE, the *_HEIGHT_* variable contains the densities at which clusters joined in the first stage; for clusters formed in the second stage, *_HEIGHT_* is a very small negative number.

If the input data set contains coordinates, the following variables appear in the output data set:

the variables containing the coordinates used in the cluster analysis. For output observations that correspond to input observations, the values of the coordinates are the same in both data sets except for some slight numeric error possibly introduced by standardizing and unstandardizing if the STANDARD option is used. For output observations that correspond to clusters of more than one input observation, the values of the coordinates are the cluster means.

*_ERSQ_*, the approximate expected value of R square under the uniform null hypothesis

*_RATIO_*, equal to $\frac{1-\_ERSQ\_}{1-\_RSQ\_}$

*_LOGR_*, natural logarithm of *_RATIO_*

*_CCC_*, the cubic clustering criterion

The variables *_ERSQ_*, *_RATIO_*, *_LOGR_*, and *_CCC_* have missing values when the number of clusters is greater than one-fifth the number of observations.

If the input data set contains coordinates and METHOD=AVERAGE, METHOD=CENTROID, or METHOD=WARD, then the following variables appear in the output data set:

*_DIST_*, the Euclidean distance between the means of the last clusters joined

*_AVLINK_*, the average distance between the last clusters joined

If the input data set contains coordinates or METHOD=AVERAGE, METHOD=CENTROID, or METHOD=WARD, then the following variables appear in the output data set:

*_RMSSTD_*, the root mean squared standard deviation of the current cluster

*_SPRSQ_*, the semipartial squared multiple correlation or the decrease in the proportion of variance accounted for due to joining two clusters to form the current cluster

*_RSQ_*, the squared multiple correlation

*_PSF_*, the pseudo Fstatistic

*_PST2_*, the pseudo t^2statistic

If METHOD=EML, then the following variable appears in the output data set

*_LNLR_*, the log-likelihood ratio

If METHOD=TWOSTAGE or METHOD=DENSITY, the following variable appears in the output data set:

*_MODE_*, pertaining to the modal clusters. With METHOD=DENSITY, the *_MODE_* variable indicates the number of modal clusters contained by the current cluster. With METHOD=TWOSTAGE, the *_MODE_* variable gives the maximum density in each modal cluster and the fusion density, d^*, for clusters containing two or more modal clusters; for clusters containing no modal clusters, *_MODE_* is missing.

If nonparametric density estimates are requested (when METHOD=DENSITY or METHOD=TWOSTAGE and the HYBRID option is not used; or when the TRIM= option is used), the output data set contains the following:

*_DENS_*, the maximum density in the current cluster

Displayed Output

If you specify the SIMPLE option and the data are coordinates, PROC CLUSTER produces simple descriptive statistics for each variable:

the Mean

the standard deviation, Std Dev

the Skewness

the Kurtosis

a coefficient of Bimodality

If the data are coordinates and you do not specify the NOEIGEN option, PROC CLUSTER displays the following:

the Eigenvalues of the Correlation or Covariance Matrix

the Difference between successive eigenvalues

the Proportion of variance explained by each eigenvalue

the Cumulative proportion of variance explained

If the data are coordinates, PROC CLUSTER displays the Root Mean Squared Total-Sample Standard Deviation of the variables

If the distances are normalized, PROC CLUSTER displays one of the following, depending on whether squared or unsquared distances are used:

the Root Mean Squared Distance Between Observations

the Mean Distance Between Observations

For the generations in the clustering process specified by the PRINT= option, PROC CLUSTER displays the following:

the Number of Clusters or NCL

the names of the Clusters Joined. The observations are identified by the formatted value of the ID variable, if any; otherwise, the observations are identified by OB*n*, where *n*is the observation number. The CLUSTER procedure displays the entire value of the ID variable in the cluster history instead of truncating at 16 characters. Long ID values might be split onto several lines. Clusters of two or more observations are identified as CL*n*, where *n*is the number of clusters existing after the cluster in question is formed.

the number of observations in the new cluster, Frequency of New Cluster or FREQ

If you specify the RMSSTD option and the data are coordinates, or if you specify METHOD=AVERAGE, METHOD=CENTROID, or METHOD=WARD, then PROC CLUSTER displays the root mean squared standard deviation of the new cluster, RMS Std of New Cluster or RMS Std.

PROC CLUSTER displays the following items if you specify METHOD=WARD. It also displays them if you specify the RSQUARE option and either the data are coordinates or you specify METHOD=AVERAGE or METHOD=CENTROID.

the decrease in the proportion of variance accounted for resulting from joining the two clusters, Semipartial R-Squared or SPRSQ. This equals the between-cluster sum of squares divided by the corrected total sum of squares.

the squared multiple correlation, R-Squared or RSQ. R square is the proportion of variance accounted for by the clusters.

If you specify the CCC option and the data are coordinates, PROC CLUSTER displays the following:

Approximate Expected R-Squared or ERSQ, the approximate expected value of R square under the uniform null hypothesis

the Cubic Clustering Criterion or CCC. The cubic clustering criterion and approximate expected R square are given missing values when the number of clusters is greater than one-fifth the number of observations.

If you specify the PSEUDO option and the data are coordinates, or if you specify METHOD=AVERAGE, METHOD=CENTROID, or METHOD=WARD, then PROC CLUSTER displays the following:

Pseudo For PSF, the pseudo Fstatistic measuring the separation among all the clusters at the current level

Pseudo t^2or PST2, the pseudo t^2statistic measuring the separation between the two clusters most recently joined

If you specify the NOSQUARE option and METHOD=AVERAGE, PROC CLUSTER displays the (Normalized) Average Distance or (Norm) Aver Dist, the average distance between pairs of objects in the two clusters joined with one object from each cluster.

If you do not specify the NOSQUARE option and METHOD=AVERAGE, PROC CLUSTER displays the (Normalized) RMS Distance or (Norm) RMS Dist, the root mean squared distance between pairs of objects in the two clusters joined with one object from each cluster.

If METHOD=CENTROID, PROC CLUSTER displays the (Normalized) Centroid Distance or (Norm) Cent Dist, the distance between the two cluster centroids.

If METHOD=COMPLETE, PROC CLUSTER displays the (Normalized) Maximum Distance or (Norm) Max Dist, the maximum distance between the two clusters.

If METHOD=DENSITY or METHOD=TWOSTAGE, PROC CLUSTER displays the following:

Normalized Fusion Density or Normalized Fusion Dens, the value of d^* as defined in the section Clustering Methods

the Normalized Maximum Density in Each Cluster joined, including the Lesser or Min, and the Greater or Max, of the two maximum density values

If METHOD=EML, PROC CLUSTER displays the following:

Log Likelihood Ratio or LNLR

Log Likelihood or LNLIKE

If METHOD=FLEXIBLE, PROC CLUSTER displays the (Normalized) Flexible Distance or (Norm) Flex Dist, the distance between the two clusters based on the Lance-Williams flexible formula.

If METHOD=MEDIAN, PROC CLUSTER displays the (Normalized) Median Distance or (Norm) Med Dist, the distance between the two clusters based on the median method.

If METHOD=MCQUITTY, PROC CLUSTER displays the (Normalized) McQuitty's Similarity or (Norm) MCQ, the distance between the two clusters based on McQuitty's similarity method.

If METHOD=SINGLE, PROC CLUSTER displays the (Normalized) Minimum Distance or (Norm) Min Dist, the minimum distance between the two clusters.

If you specify the NONORM option and METHOD=WARD, PROC CLUSTER displays the Between-Cluster Sum of Squares or BSS, the ANOVA sum of squares between the two clusters joined.

If you specify neither the NOTIE option nor METHOD=TWOSTAGE or METHOD=DENSITY, PROC CLUSTER displays *Tie*, where a T in the column indicates a tie for minimum distance and a blank indicates the absence of a tie.

After the cluster history, if METHOD=TWOSTAGE or METHOD=DENSITY, PROC CLUSTER displays the number of modal clusters.

ODS Table Names

PROC CLUSTER assigns a name to each table it creates. You can use these names to reference the table when using the Output Delivery System (ODS) to select tables and create output data sets. These names are listed in Table 29.4. For more information about ODS, see Chapter 20, Using the Output Delivery System.

Table 29.4 ODS Tables Produced by PROC CLUSTER

ODS Table	Description	Statement Option

Name			
ClusterHistory	Observation or clusters joined, frequencies and other cluster statistics	PROC	default
SimpleStatistics	Simple statistics, before or after trimming	PROC	SIMPLE
EigenvalueTable	Eigenvalues of the CORR or COV matrix	PROC	default
rmsstd	Root mean square total sample standard deviation	PROC	default
avdist	Root mean square distance between observations	PROC	default

ODS Graphics

To produce graphics from PROC CLUSTER, you must enable ODS Graphics by specifying the `ods graphics on` statement before running PROC CLUSTER. See Chapter 21, Statistical Graphics Using ODS, for more information.

PROC CLUSTER can produce line plots of the cubic clustering criterion, pseudo F, and pseudo t^2statistics. To plot a statistic, you must ask for it to be computed via one or more of the CCC, PSEUDO, or PLOT options.

You can reference every graph produced through ODS Graphics with a name. The names of the graphs that PROC CLUSTER generates are listed in Table 29.5, along with the required statements and options.

Table 29.5 ODS Graphics Produced by PROC CLUSTER

ODS Graph Name	Plot Description	Statement & Option
CubicClusCritPlot	Cubic clustering criterion for the number of clusters	PROC CLUSTER PLOTS=CCC
PseudoFPlot	Pseudo Fcriterion for the number of clusters	PROC CLUSTER PLOTS=PSF
PseudoTSqPlot	Pseudo t^2criterion for the number of clusters	PROC CLUSTER PLOTS=PST2
CccAndPsTSqPlot	Cubic clustering criterion and pseudo t^2	PROC CLUSTER PLOTS=(CCC PST2)
CccAndPsfPlot	Cubic clustering criterion and pseudo F	PROC CLUSTER PLOTS=(CCC PSF)
CccPsfAndPsTSqPlot	Cubic clustering criterion, pseudo F, and pseudo t^2	PROC CLUSTER PLOTS=ALL