SAS/STAT® 15.1 User's Guide
The DISTANCE Procedure

This document is an individual chapter from *SAS/STAT® 15.1 User's Guide*.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2018. *SAS/STAT® 15.1 User's Guide*. Cary, NC: SAS Institute Inc.

SAS/STAT® 15.1 User's Guide

Copyright © 2018, SAS Institute Inc., Cary, NC, USA

All Rights Reserved. Produced in the United States of America.

For a hard-copy book: No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

For a web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

The scanning, uploading, and distribution of this book via the Internet or any other means without the permission of the publisher is illegal and punishable by law. Please purchase only authorized electronic editions and do not participate in or encourage electronic piracy of copyrighted materials. Your support of others' rights is appreciated.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government. Use, duplication, or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a), and DFAR 227.7202-4, and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007). If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation. The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

November 2018

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

SAS software may be provided with certain third-party software, including but not limited to open-source software, which is licensed under its applicable third-party software license agreement. For license information about third-party software distributed with SAS software, refer to http://support.sas.com/thirdpartylicenses.

Chapter 40
The DISTANCE Procedure

Contents

Overview: DISTANCE Procedure **2671**
Levels of Measurement 2672
Symmetric versus Asymmetric Nominal Variables 2673
Standardization 2674
Getting Started: DISTANCE Procedure **2674**
Creating a Distance Matrix as Input for a Subsequent Cluster Analysis 2674
Syntax: DISTANCE Procedure **2679**
PROC DISTANCE Statement 2680
VAR Statement 2686
ID Statement 2690
COPY Statement 2691
BY Statement 2691
FREQ Statement 2691
WEIGHT Statement 2691
Details: DISTANCE Procedure **2693**
Proximity Measures 2693
Missing Values 2699
Formatted versus Unformatted Values 2700
Output Data Sets 2700
Examples: DISTANCE Procedure **2701**
Example 40.1: Divorce Grounds – the Jaccard Coefficient 2701
Example 40.2: Financial Data – Stock Dividends 2707
References **2713**

Overview: DISTANCE Procedure

The DISTANCE procedure computes various measures of distance, dissimilarity, or similarity between the observations (rows) of an input SAS data set, which can contain numeric or character variables, or both, depending on which proximity measure is used.

The proximity measures are stored as a lower triangular matrix or a square matrix (depending on the SHAPE= option) in an output data set that can then be used as input to the CLUSTER, MDS, and MODECLUS procedures.

The number of rows and columns in the output data set equals the number of observations in the input data set. If the input data set contains BY groups, an output matrix is computed for each BY group with the size determined by the maximum number of observations in any BY group.

The output data set is of type TYPE=DISTANCE or TYPE=SIMILAR, depending on the value of the METHOD= option. See the METHOD= option for more information about the association between the method and the output data set type.

Data set types do not persist when you copy or modify a data set. You must specify the TYPE= data set option for the new data set, as in the following example:

```
data dist2(type=distance);
   set dist;
run;
```

See the OUT= option for more information about data set type persistence.

PROC DISTANCE also provides various nonparametric and parametric methods for standardizing variables. Different variables can be standardized with different methods.

Distance matrices are used frequently in data mining, genomics, marketing, financial analysis, management science, education, chemistry, psychology, biology, and various other fields.

Levels of Measurement

Measurement of some attribute of a set of objects is the process of assigning numbers or other symbols to the objects in such a way that properties of the numbers or symbols reflect properties of the attribute being measured. There are different *levels* of measurement that involve different properties (relations and operations) of the numbers or symbols. Associated with each level of measurement is a set of transformations of the measurements that preserve the relevant properties; these transformations are called *permissible* transformations. A particular way of assigning numbers or symbols to measure something is called a *scale* of measurement.

The most commonly discussed levels of measurement are as follows:

Nominal — Two objects are assigned the same symbol if they have the same value of the attribute. Permissible transformations are any one-to-one or many-to-one transformation, although a many-to-one transformation loses information.

Ordinal — Objects are assigned numbers such that the order of the numbers reflects an order relation defined on the attribute. Two objects x and y with attribute values $a(x)$ and $a(y)$ are assigned numbers $m(x)$ and $m(y)$ such that if $m(x) > m(y)$, then $a(x) > a(y)$. Permissible transformations are any monotone increasing transformation, although a transformation that is not strictly increasing loses information.

Interval — Objects are assigned numbers such that differences between the numbers reflect differences of the attribute. If $m(x) - m(y) > m(u) - m(v)$, then $a(x) - a(y) > a(u) - a(v)$. Permissible transformations are any affine transformation $t(m) = c * m + d$, where c and d are constants; another way of saying this is that the origin and unit of measurement are arbitrary.

Log-interval	Objects are assigned numbers such that ratios between the numbers reflect ratios of the attribute. If $m(x)/m(y) > m(u)/m(v)$, then $a(x)/a(y) > a(u)/a(v)$. Permissible transformations are any power transformation $t(m) = c * m^d$, where c and d are constants.
Ratio	Objects are assigned numbers such that differences and ratios between the numbers reflect differences and ratios of the attribute. Permissible transformations are any linear (similarity) transformation $t(m) = c * m$, where c is a constant; another way of saying this is that the unit of measurement is arbitrary.
Absolute	Objects are assigned numbers such that all properties of the numbers reflect analogous properties of the attribute. The only permissible transformation is the identity transformation.

Proximity measures provided in the DISTANCE procedure accept four levels of measurement: nominal, ordinal, interval, and ratio. Ordinal variables are transformed to interval variables before processing. This is done by replacing the data with their rank scores, and by assuming that the classes of an ordinal variable are spaced equally along the interval scale. See the RANKSCORE= option in the section "PROC DISTANCE Statement" on page 2680 for choices on assigning scores to ordinal variables. There are also different approaches for how to transform an ordinal variable to an interval variable. See Anderberg (1973) for alternatives.

Symmetric versus Asymmetric Nominal Variables

A binary variable contains two possible outcomes: 1 (positive/present) or 0 (negative/absent). If there is no preference for which outcome should be coded as 0 and which as 1, the binary variable is called *symmetric*. For example, the binary variable "is evergreen?" for a plant has the possible states "loses leaves in winter" and "does not lose leaves in winter." Both are equally valuable and carry the same weight when a proximity measure is computed. Commonly used measures that accept symmetric binary variables include the Simple Matching, Hamann, Roger and Tanimoto, Sokal and Sneath 1, and Sokal and Sneath 3 coefficients.

If the outcomes of a binary variable are not equally important, the binary variable is called *asymmetric*. An example of such a variable is the presence or absence of a relatively rare attribute, such as "is color-blind" for a human being. While you say that two people who are color-blind have something in common, you cannot say that people who are not color-blind have something in common. The most important outcome is usually coded as 1 (present) and the other is coded as 0 (absent). The agreement of two 1's (a present-present match or a positive match) is more significant than the agreement of two 0's (an absent-absent match or a negative match). Usually, the negative match is treated as irrelevant. Commonly used measures that accept asymmetric binary variables include Jaccard, Dice, Russell and Rao, Binary Lance and Williams nonmetric, and Kulcynski coefficients.

When nominal variables are employed, the comparison of one data unit with another can only be in terms of whether the data units score the same or different on the variables. If a variable is defined as an asymmetric nominal variable and two data units score the same but fall into the absent category, the absent-absent match is excluded from the computation of the proximity measure.

Standardization

Since variables with large variances tend to have more effect on the proximity measure than those with small variances, it is recommended that you standardize the variables before the computation of the proximity measure. The DISTANCE procedure provides a convenient way to standardize each variable with its own method before the proximity measures are computed. You can also perform the standardization by using the STDIZE procedure, with the limitation that all variables must be standardized with the same method.

Mandatory Standardization

Variable standardization is not required if any of the following conditions is true:

- if there is only one level of measurement
- if only asymmetric nominal and nominal levels are specified
- if the NOSTD option is specified in the PROC DISTANCE statement
- if METHOD=JACCARD or METHOD=DJACCARD is specified

Otherwise, standardization is mandatory.

When standardization is mandatory and no standardization method is specified, a default method of standardization will be used. This default method is determined by the measurement level. In general, the default method is STD for interval variables except when METHOD=GOWER or METHOD=DGOWER is specified. See the STD= option in the section "VAR Statement" on page 2686 for the default methods for GOWER and DGOWER as well as methods available for standardizing variables.

When standardization is mandatory, PROC DISTANCE ignores the REPONLY option, if it is specified.

Getting Started: DISTANCE Procedure

Creating a Distance Matrix as Input for a Subsequent Cluster Analysis

The following example demonstrates how you can use the DISTANCE procedure to obtain a distance matrix that will be used as input to a subsequent clustering procedure.

The following data, originated by A. Weber and cited in Hand et al. (1994, p. 297), measure the amount of protein consumed for nine food groups in 25 European countries. The nine food groups are red meat (RedMeat), white meat (WhiteMeat), eggs (Eggs), milk (Milk), fish (Fish), cereal (Cereal), starch (Starch), nuts (Nuts), and fruits and vegetables (FruitVeg). Suppose you want to determine whether national figures in protein consumption can be used to determine certain types or categories of countries; specifically, you want to perform a cluster analysis to determine whether these 25 countries can be formed into groups suggested by the data.

The following DATA step creates the SAS data set Protein:

```
data Protein;
   length Country $ 14;
   input Country &$ RedMeat WhiteMeat Eggs Milk
                    Fish Cereal Starch Nuts FruitVeg;
   datalines;
Albania            10.1  1.4  0.5   8.9  0.2  42.3  0.6  5.5  1.7
Austria             8.9 14.0  4.3  19.9  2.1  28.0  3.6  1.3  4.3
Belgium            13.5  9.3  4.1  17.5  4.5  26.6  5.7  2.1  4.0
Bulgaria            7.8  6.0  1.6   8.3  1.2  56.7  1.1  3.7  4.2
Czechoslovakia      9.7 11.4  2.8  12.5  2.0  34.3  5.0  1.1  4.0
Denmark            10.6 10.8  3.7  25.0  9.9  21.9  4.8  0.7  2.4
E Germany           8.4 11.6  3.7  11.1  5.4  24.6  6.5  0.8  3.6
Finland             9.5  4.9  2.7  33.7  5.8  26.3  5.1  1.0  1.4
France             18.0  9.9  3.3  19.5  5.7  28.1  4.8  2.4  6.5
Greece             10.2  3.0  2.8  17.6  5.9  41.7  2.2  7.8  6.5
Hungary             5.3 12.4  2.9   9.7  0.3  40.1  4.0  5.4  4.2
Ireland            13.9 10.0  4.7  25.8  2.2  24.0  6.2  1.6  2.9
Italy               9.0  5.1  2.9  13.7  3.4  36.8  2.1  4.3  6.7
Netherlands         9.5 13.6  3.6  23.4  2.5  22.4  4.2  1.8  3.7
Norway              9.4  4.7  2.7  23.3  9.7  23.0  4.6  1.6  2.7
Poland              6.9 10.2  2.7  19.3  3.0  36.1  5.9  2.0  6.6
Portugal            6.2  3.7  1.1   4.9 14.2  27.0  5.9  4.7  7.9
Romania             6.2  6.3  1.5  11.1  1.0  49.6  3.1  5.3  2.8
Spain               7.1  3.4  3.1   8.6  7.0  29.2  5.7  5.9  7.2
Sweden              9.9  7.8  3.5   4.7  7.5  19.5  3.7  1.4  2.0
Switzerland        13.1 10.1  3.1  23.8  2.3  25.6  2.8  2.4  4.9
UK                 17.4  5.7  4.7  20.6  4.3  24.3  4.7  3.4  3.3
USSR                9.3  4.6  2.1  16.6  3.0  43.6  6.4  3.4  2.9
W Germany          11.4 12.5  4.1  18.8  3.4  18.6  5.2  1.5  3.8
Yugoslavia          4.4  5.0  1.2   9.5  0.6  55.9  3.0  5.7  3.2
;
```

The data set Protein contains the character variable Country and the nine numeric variables representing the food groups. The $14. in the INPUT statement specifies that the variable Country has a length of 14.

The following statements create the distance matrix and display part of it:

```
title 'Protein Consumption in Europe';
proc distance data=Protein out=Dist method=Euclid;
   var interval(RedMeat--FruitVeg / std=Std);
   id Country;
run;

proc print data=Dist(obs=10);
   title2 'First 10 Observations in Output Data Set from PROC DISTANCE';
run;
title2;
```

An output SAS data set called Dist, which contains the distance matrix, is created through the OUT= option. The METHOD=EUCLID option requests that Euclidean distances (which is the default) should be computed and produces an output data set of TYPE=DISTANCE.[1]

The VAR statement lists the variables (RedMeat—FruitVeg) along with their measurement level to be used in the analysis. An interval level of measurement is assigned to those variables. Since variables with large variances tend to have more effect on the proximity measure than those with small variances, each variable is standardized by the STD method to have a mean of 0 and a standard deviation of 1. This is done by adding "/ STD=STD" at the end of the variables list.

The ID statement specifies that the variable Country should be copied to the OUT= data set and used to generate names for the distance variables. The distance variables in the output data set are named by the values in the ID variable, and the maximum length for the names of these variables is 14.

There are 25 observations in the input data set; therefore, the output data set Dist contains a 25-by-25 lower triangular matrix.

The PROC PRINT statement displays the first 10 observations in the output data set Dist as shown in Figure 40.1.

[1] Data set types do not persist when you copy or modify a data set. You must specify the TYPE= data set option for the new data set. See the METHOD= and OUT= options for more information about data set types.

Figure 40.1 First 10 Observations in the Output Data Set from PROC DISTANCE

Protein Consumption in Europe
First 10 Observations in Output Data Set from PROC DISTANCE

Obs	Country	Albania	Austria	Belgium	Bulgaria	Czechoslovakia	Denmark	E_Germany	Finland
1	Albania	0.00000	.	.	.	.	.	.	.
2	Austria	6.12388	0.00000	.	.	.	.	.	.
3	Belgium	5.94109	2.44987	0.00000	.	.	.	.	.
4	Bulgaria	2.76446	4.88331	5.22711	0.00000	.	.	.	.
5	Czechoslovakia	5.13959	2.11498	2.21330	3.94761	0.00000	.	.	.
6	Denmark	6.61002	3.01392	2.52541	6.00803	3.34049	0.00000	.	.
7	E Germany	6.39178	2.56341	2.10211	5.40824	1.87962	2.72112	0.00000	.
8	Finland	5.81458	4.04271	3.45779	5.74882	3.91378	2.61570	3.99426	0.00000
9	France	6.29601	3.58891	2.19329	5.54675	3.36011	3.65772	3.78184	4.56796
10	Greece	4.24495	5.16330	4.69515	3.74849	4.86684	5.59084	5.61496	5.47453

Obs	France	Greece	Hungary	Ireland	Italy	Netherlands	Norway	Poland	Portugal	Romania	Spain
1	.	.	.	.	.	.	.	.	.	.	.
2	.	.	.	.	.	.	.	.	.	.	.
3	.	.	.	.	.	.	.	.	.	.	.
4	.	.	.	.	.	.	.	.	.	.	.
5	.	.	.	.	.	.	.	.	.	.	.
6	.	.	.	.	.	.	.	.	.	.	.
7	.	.	.	.	.	.	.	.	.	.	.
8	.	.	.	.	.	.	.	.	.	.	.
9	0.00000	.	.	.	.	.	.	.	.	.	.
10	4.54456	0	.	.	.	.	.	.	.	.	.

Obs	Sweden	Switzerland	UK	USSR	W_Germany	Yugoslavia
1	.	.	.	.	.	.
2	.	.	.	.	.	.
3	.	.	.	.	.	.
4	.	.	.	.	.	.
5	.	.	.	.	.	.
6	.	.	.	.	.	.
7	.	.	.	.	.	.
8	.	.	.	.	.	.
9	.	.	.	.	.	.
10	.	.	.	.	.	.

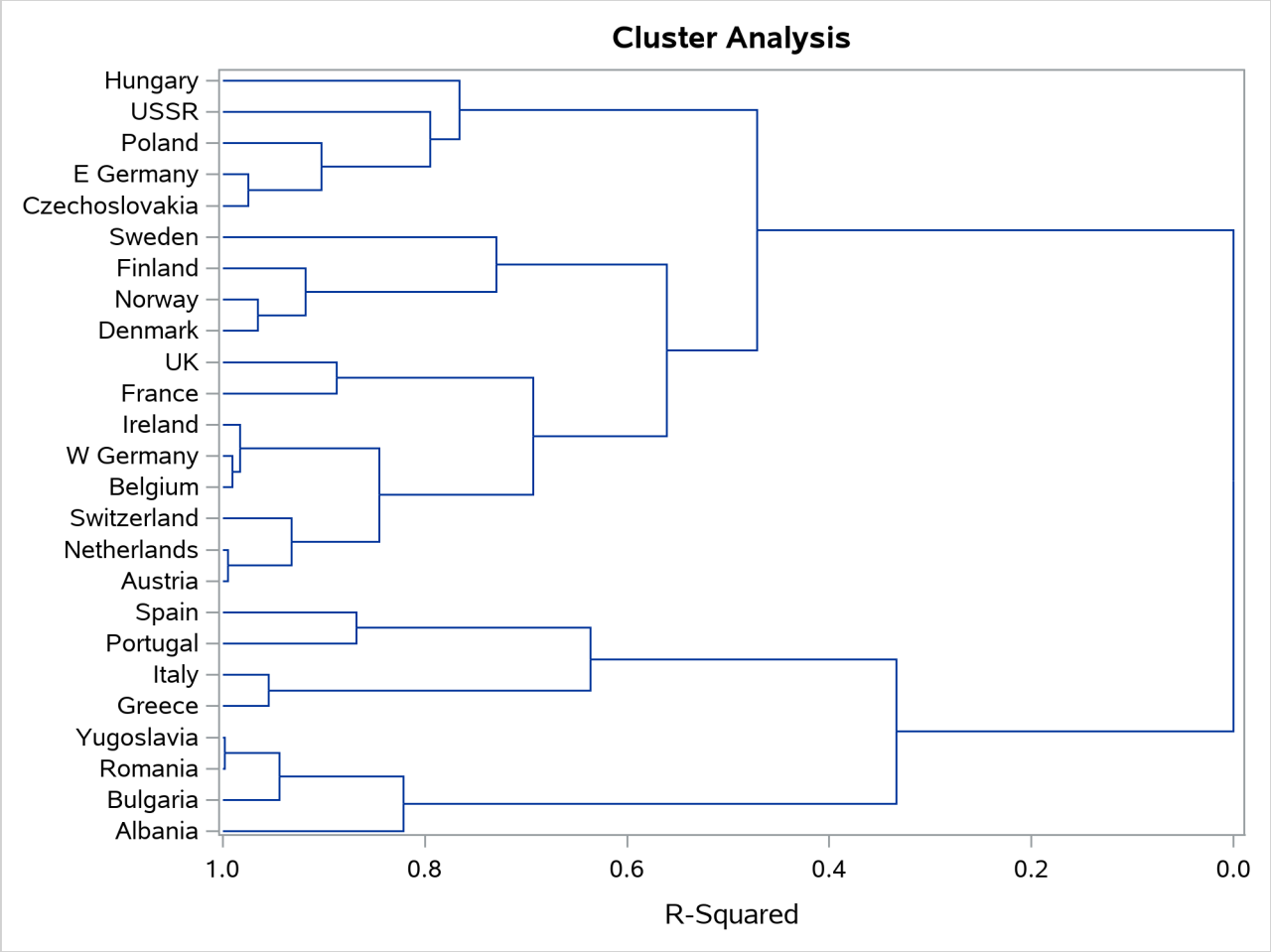
The following statements produce the dendrogram in Figure 40.2:

```
ods graphics on;

proc cluster data=Dist method=Ward plots=dendrogram(height=rsq);
   id Country;
run;
```

The CLUSTER procedure performs a Ward's minimum-variance cluster analysis based on the distance matrix created by PROC DISTANCE. PROC CLUSTER, along with ODS Graphics, produces the dendrogram shown in Figure 40.2. The option PLOTS=DENDROGRAM(HEIGHT=RSQ) specifies the squared multiple correlation as the height variable in the dendrogram.

Figure 40.2 Dendrogram of R Square



After inspecting the dendrogram in Figure 40.2, you will see that when the countries are grouped into six clusters, the proportion of variance accounted for by these clusters is slightly less than 70% (69.3%). The 25 countries are clustered as follows:

- Balkan countries: Albania, Bulgaria, Romania, and Yugoslavia
- Mediterranean countries: Greece and Italy
- Iberian countries: Portugal and Spain
- Western European countries: Austria, Netherlands, Switzerland, Belgium, former West Germany, Ireland, France, and U.K.
- Scandinavian countries: Denmark, Norway, Finland, and Sweden
- Eastern European countries: former Czechoslovakia, former East Germany, Poland, former U.S.S.R., and Hungary

Syntax: DISTANCE Procedure

The following statements are available in the DISTANCE procedure:

```
PROC DISTANCE < options > ;
    BY variables ;
    COPY variables ;
    FREQ variable ;
    ID variable ;
    VAR level (variables < / options >) ;
    WEIGHT variable ;
```

Both the PROC DISTANCE statement and the VAR statement are required.

PROC DISTANCE Statement

PROC DISTANCE < *options* > **;**

The PROC DISTANCE statement invokes the DISTANCE procedure. Table 40.1 summarizes the *options* available in the PROC DISTANCE statement. These *options* are discussed in the following section.

Table 40.1 Summary of PROC DISTANCE Statement Options

Option	Description
Standardize variables	
ADD=	Specifies the constant to add to each value after standardizing and multiplying by the value specified in the MULT= option
FUZZ=	Specifies the relative fuzz factor for writing the output
INITIAL=	Specifies the method for computing initial estimates for the A-estimates
MULT=	Specifies the constant to multiply each value by after standardizing
NORM	Normalizes the scale estimator to be consistent for the standard deviation of a normal distribution
NOSTD	Suppresses standardization
SNORM	Normalizes the scale estimator to have an expectation of approximately 1 for a standard normal distribution
STDONLY	Standardizes variables only (suppresses computation of the distance matrix)
VARDEF=	Specifies the variances divisor
Generate distance matrix	
ABSENT=	Specifies the value to be used as an absence value for all the asymmetric nominal variables
METHOD=	Specifies the method for computing proximity measures
PREFIX=	Specifies a prefix for naming the distance variables in the OUT= data set
RANKSCORE=	Specifies the method of assigning scores to ordinal variables
SHAPE=	Specifies the shape of the proximity matrix to be stored in the OUT= data set
UNDEF=	Specifies the numeric constant used to replace undefined distances
Replace missing values	
NOMISS	Omits observations with missing values from computation of the location and scale measures, if standardization applies; outputs missing values to the distance matrix for observations with missing values
REPLACE	Replaces missing data with zero in the standardized data
REPONLY	Replaces missing data with the location measure (does not standardize the data)
Specify data set details	
DATA=	Specifies the input data set
OUT=	Specifies the output data set
OUTSDZ=	Specifies the output data set for standardized scores

These *options* and their abbreviations are described (in alphabetical order) in the remainder of this section.

ABSENT=*number* | *qs*
specifies the value to be used as an absence value in an irrelevant absent-absent match for *all* of the asymmetric nominal variables. If you want to specify a different absence value for a particular variable, use the ABSENT= option in the VAR statement. See the ABSENT= option in the section "VAR Statement" on page 2686 for details.

An absence value for a variable can be either a numeric value or a quoted string consisting of combinations of characters. For instance, ., -999, and "NA" are legal values for the ABSENT= option.

The default absence value for a character variable is "NONE" (notice that a blank value is considered a missing value), and the default absence value for a numeric variable is 0.

ADD=*c*
specifies a constant, *c*, to add to each value after standardizing and multiplying by the value you specify in the MULT= option. The default value is 0.

DATA=*SAS-data-set*
specifies the input data set containing observations from which the proximity is computed. If you omit the DATA= option, the most recently created SAS data set is used.

FUZZ=*c*
specifies the relative fuzz factor for computing the standardized scores. The default value is 1E–14. For the OUTSDZ= data set, the score is computed as follows:

$$\text{if } |\text{standardized scores}| < m \times c, \text{ then standardized scores} = 0$$

where m is the numeric constant specified in the MULT= option, or 1 if MULT= option is not specified.

INITIAL=*method*
specifies the method of computing initial estimates for the A-estimates (ABW, AWAVE, and AHUBER). The following methods are not allowed for the INITIAL= option: ABW, AHUBER, AWAVE, and IN.

The default value is INITIAL=MAD.

METHOD=*method*
specifies the method of computing proximity measures.

For use in PROC CLUSTER, distance or dissimilarity measures such as METHOD=EUCLID or METHOD=DGOWER should be chosen.

The following six tables outline the proximity measures available for the METHOD= option. These tables are classified by levels of measurement accepted by each method. Each table contains four or five columns: the Method column shows the proximity measures, one or two Range columns show the upper and lower bounds, and the TYPE= column shows the type of proximity. The TYPE= column contains SIMILAR if a method generates similarity measures or DISTANCE if a method generates distance or dissimilarity measures. The output data set is of the type shown. For more information about the output data set, see the OUT= option.

For formulas and descriptions of these methods, see the section "Details: DISTANCE Procedure" on page 2693.

Table 40.2 shows the range and output matrix type of the GOWER and DGOWER methods. These two methods accept all measurement levels including ratio, interval, ordinal, nominal, and asymmetric nominal. METHOD=GOWER or METHOD=DGOWER always implies standardization. Assuming all the numeric (ordinal, interval, and ratio) variables are standardized by their corresponding default methods, the possible range values for both methods are from 0 and 1, inclusive. For more information about the default methods of standardization for METHOD=GOWER or METHOD=DGOWER, see the STD= option in the section "VAR Statement" on page 2686.

Table 40.2 Methods That Accept All Measurement Levels

Method	Description	Range	TYPE=
GOWER	Gower and Legendre (1986) similarity	0 to 1	**SIMILAR**
DGOWER	1 minus GOWER	0 to 1	**DISTANCE**

Table 40.3 shows *methods* that accept ratio, interval, and ordinal variables.

Table 40.3 Methods That Accept Ratio, Interval, and Ordinal Variables

Method	Description	Range	TYPE=
EUCLID	Euclidean distance	≥ 0	**DISTANCE**
SQEUCLID	Squared Euclidean distance	≥ 0	**DISTANCE**
SIZE	Size distance	≥ 0	**DISTANCE**
SHAPE	Shape distance	≥ 0	**DISTANCE**
COV	Covariance	≥ 0	**SIMILAR**
CORR	Correlation	-1 to 1	**SIMILAR**
DCORR	Correlation transformed to Euclidean distance	0 to $\sqrt{2}$	**DISTANCE**
SQCORR	Squared correlation	0 to 1	**SIMILAR**
DSQCORR	One minus squared correlation	0 to 1	**DISTANCE**
L(p)	Minkowski (L_p) distance, where p is a positive numeric value. Typical values of p include 1 and 2. Very small or large values of p might cause floating-point overflow.	> 0	**DISTANCE**
CITYBLOCK	L_1, city-block, or Manhattan distance	≥ 0	**DISTANCE**
CHEBYCHEV	L_∞	≥ 0	**DISTANCE**
POWER(p, r)	Generalized Euclidean distance, where p is a positive numeric value and r is a nonnegative numeric value. The distance between two observations is the rth root of sum of the absolute differences to the pth power between the values for the observations.	≥ 0	**DISTANCE**

Table 40.4 shows *methods* that accept ratio variables. Notice that all possible range values are nonnegative, because ratio variables are assumed to be positive.

Table 40.4 Methods That Accept Ratio Variables

Method	Description	Range	TYPE=
SIMRATIO	Similarity ratio (if variables are binary, this is the Jaccard coefficient)	0 to 1	**SIMILAR**
DISRATIO	One minus similarity ratio	0 to 1	**DISTANCE**
NONMETRIC	Lance and Williams nonmetric coefficient	0 to 1	**DISTANCE**
CANBERRA	Canberra metric distance coefficient	0 to 1	**DISTANCE**
COSINE	Cosine coefficient	0 to 1	**SIMILAR**
DOT	Dot (inner) product coefficient	≥ 0	**SIMILAR**
OVERLAP	Overlap similarity	≥ 0	**SIMILAR**
DOVERLAP	Overlap dissimilarity	≥ 0	**DISTANCE**
CHISQ	Chi-square coefficient	≥ 0	**DISTANCE**
CHI	Square root of chi-square coefficient	≥ 0	**DISTANCE**
PHISQ	Phi-square coefficient	≥ 0	**DISTANCE**
PHI	Square root of phi-square coefficient	≥ 0	**DISTANCE**

Table 40.5 shows *methods* that accept nominal variables.

Table 40.5 Methods That Accept Nominal Variables

Method	Description	Range	TYPE=
HAMMING	Hamming distance	0 to v	**DISTANCE**
MATCH	Simple matching coefficient	0 to 1	**SIMILAR**
DMATCH	Simple matching coefficient transformed to Euclidean distance	0 to 1	**DISTANCE**
DSQMATCH	Simple matching coefficient transformed to squared Euclidean distance	0 to 1	**DISTANCE**
HAMANN	Hamann coefficient	–1 to 1	**SIMILAR**
RT	Roger and Tanimoto	0 to 1	**SIMILAR**
SS1	Sokal and Sneath 1	0 to 1	**SIMILAR**
SS3	Sokal and Sneath 3	0 to 1	**SIMILAR**

Note that v denotes the number of variables (dimensionality).

Table 40.6 shows *methods* that accept asymmetric nominal variables. Use the ABSENT= option to create a value to be considered absent.

Table 40.6 Methods That Accept Asymmetric Nominal Variables

Method	Description	Range	TYPE=
DICE	Dice coefficient or Czekanowski/Sorensen similarity coefficient	0 to 1	**SIMILAR**
RR	Russell and Rao	0 to 1	**SIMILAR**
BLWNM	Binary Lance and Williams nonmetric, or Bray-Curtis coefficient	0 to 1	**DISTANCE**
K1	Kulcynski 1	≥ 0	**SIMILAR**

Table 40.7 shows *methods* that accept asymmetric nominal and ratio variables. Use the ABSENT= option to create a value to be considered absent. The table contains five columns. The third column contains possible range values if only one level of measurement (either ratio or asymmetric nominal but not both) is specified; the fourth column contains possible range values if both levels are specified.

The JACCARD method is equivalent to the SIMRATIO method if there is no asymmetric nominal variable; if both ratio and asymmetric nominal variables are present, the coefficient is computed as the sum of the coefficient from the ratio variables and the coefficient from the asymmetric nominal variables. See "Proximity Measures" in the section "Details: DISTANCE Procedure" on page 2693 for the formula and descriptions of the JACCARD method.

Table 40.7 Methods That Accept Asymmetric Nominal and Ratio Variables

Method	Description	Range (One Level)	Range (Two Levels)	TYPE=
JACCARD	Jaccard similarity coefficient	0 to 1	0 to 2	**SIMILAR**
DJACCARD	Jaccard dissimilarity coefficient	0 to 1	0 to 2	**DISTANCE**

MULT=*c*
specifies a numeric constant, *c*, by which to multiply each value after standardizing. The default value is 1.

NOMISS
omits observations with missing values from computation of the location and scale measures when standardizing; generates undefined (missing) distances for observations with missing values when computing distances. Use the UNDEF= option to specify the undefined values.

If a distance matrix is created to be used as an input to PROC CLUSTER, the NOMISS option should not be used because PROC CLUSTER does not accept distance matrices with missing values.

NORM
normalizes the scale estimator to be consistent for the standard deviation of a normal distribution when you specify the option STD=AGK, STD=IQR, STD=MAD, or STD=SPACING in the VAR statement.

NOSTD
suppresses standardization of the variables. The NOSTD option should not be specified with the STDONLY option or with the REPLACE option.

OUT=*SAS-data-set*
specifies the name of the SAS data set created by PROC DISTANCE. The output data set contains the BY variables, the ID variable, computed distance variables, the COPY variables, the FREQ variable, and the WEIGHT variables.

If you omit the OUT= option, PROC DISTANCE creates an output data set named according to the DATA*n* convention.

The output data set is of type TYPE=DISTANCE or TYPE=SIMILAR. See the METHOD= option for more information about the association between the method and the output data set type. Data set types do not persist when you copy or modify a data set. You must specify the TYPE= data set option for the new data set, as in the following example:

```
data dist2(type=distance);
   set dist;
run;
```

If you do not specify the TYPE=DISTANCE data set option, the new data set is the default TYPE=DATA. If you use the new data set in a procedure that accepts both TYPE=DATA or TYPE=DISTANCE data sets (such as PROC CLUSTER or PROC MODECLUS), the results will be incorrect.

OUTSDZ=*SAS-data-set*

specifies the name of the SAS data set containing the standardized scores. The output data set contains a copy of the DATA= data set, except that the analyzed variables have been standardized. Analyzed variables are those listed in the VAR statement.

PREFIX=*name*

specifies a prefix for naming the distance variables in the OUT= data set. By default, the names are Dist1, Dist2, ..., Dist*n*. If you specify PREFIX=ABC, the variables are named ABC1, ABC2, ..., ABCn. If the ID statement is also specified, the variables are named by appending the value of the ID variable to the prefix.

RANKSCORE=MIDRANK | INDEX

specifies the method of assigning scores to ordinal variables. The available methods are listed as follows:

MIDRANK assigns consecutive integers to each category with consideration of the frequency value. This is the default method.

INDEX assigns consecutive integers to each category regardless of frequencies.

The following example explains how each method assigns the rank scores. Suppose the data contain an ordinal variable ABC with values A, B, C. There are two ways to assign numbers. One is to use midranks, which depend on the frequencies of each category. Another is to assign consecutive integers to each category, regardless of frequencies.

Table 40.8 Example of Assigning Rank Scores

ABC	MIDRANK	INDEX
A	1.5	1
A	1.5	1
B	4	2
B	4	2
B	4	2
C	6	3

REPLACE

replaces missing data with zero in the standardized data (to correspond to the location measure before standardizing). To replace missing data with something else, use the MISSING= option in the VAR statement. The REPLACE option implies standardization.

You cannot specify the following *options* together:

- both the REPLACE and the REPONLY options
- both the REPLACE and the NOSTD options

REPONLY
replaces missing data by the location measure that is specified by the MISSING= option or the STD= option (if the MISSING= option is not specified), but does *not* standardize the data. If the MISSING= option is not specified and METHOD=GOWER is specified, missing values are replaced by the location measure from the RANGE method (the minimum value), and the STD= option is.

You cannot specify both the REPLACE and the REPONLY options.

SHAPE=TRIANGLE | TRI | SQUARE | SQU | SQR
specifies the shape of the proximity matrix to be stored in the OUT= data set. SHAPE=TRIANGLE requests the matrix to be stored as a lower triangular matrix; SHAPE=SQUARE requests that the matrix be stored as a square matrix. Use SHAPE=SQUARE if the output data set is to be used as input to the MODECLUS procedures. The default is TRIANGLE.

SNORM
normalizes the scale estimator to have an expectation of approximately 1 for a standard normal distribution when the STD=SPACING option is specified.

STDONLY
standardizes variables only and computes no distance matrix. You must use the OUTSDZ= option to save the standardized scores. You cannot specify both the STDONLY option and the NOSTD option.

UNDEF=*n*
specifies the numeric constant used to replace undefined distances, such as when an observation has all missing values, or if a divisor is zero.

VARDEF=DF | N | WDF | WEIGHT | WGT
specifies the divisor to be used in the calculation of distance, dissimilarity, or similarity measures, and for standardizing variables whenever a variance or covariance is computed. By default, VARDEF=DF. The values and associated divisors are as follows:

Value	Divisor	Formula
DF	degrees of freedom	$n - 1$
N	number of observations	n
WDF	sum of weights minus 1	$(\sum_i w_i) - 1$
WEIGHT | WGT	sum of weights	$\sum_i w_i$

VAR Statement

VAR *level* **(***variables* < */ options* >**)**< *level* **(***variables* < */ options* >**)**. . . > **;**

where the syntax for the *options* is as follows:

ABSENT=*value*
MISSING=*miss-method* | *value*
ORDER=*order-option*
STD=*std-method*
WEIGHTS=*weight-list*

The VAR statement lists variables from which distances are to be computed. The VAR statement is required. The variables can be numeric or character depending on their measurement levels. A variable cannot appear more than once in either the same list or a different list.

level is required. It declares the levels of measurement for those variables specified within the parentheses. Available values for *level* are as follows:

ANOMINAL variables are asymmetric nominal and can be either numeric or character.

NOMINAL variables are symmetric nominal and can be either numeric or character.

ORDINAL variables are ordinal and can be either numeric or character. Values of ordinal variables are replaced by their corresponding rank scores. If standardization is required, the standardized rank scores are output to the data set specified in the OUTSDZ= option. See the RANKSCORE= option in the PROC DISTANCE statement for methods available for assigning rank scores to ordinal variables. After being replaced by scores, ordinal variables are considered interval.

INTERVAL variables are interval and numeric.

RATIO variables are ratio and numeric. Ratio variables should always contain positive measurements.

Each variable list can be followed by an *option* list. Use "/ " after the list of variables to start the option list. An *option* list contains options that are applied to the variables. The following *options* are available in the option list:

ABSENT= specifies the value to be used as an absence value in an irrelevant absent-absent match for asymmetric nominal variables.

MISSING= specifies the method (or numeric value) with which to replace missing data.

ORDER= selects the order for assigning scores to ordinal variables.

STD= selects the standardization method.

WEIGHTS= assigns weights to the variables in the list.

If an *option* is missing from the current attribute list, PROC DISTANCE provides default values for all the variables in the current list.

For example, in the VAR statement

```
var ratio(x1-x4/std= mad weights= .5 .5 .1 .5 missing= -99)
    interval(x5/std= range)
    ordinal(x6/order= desc);
```

the first option list defines x1–x4 as ratio variables to be standardized by the MAD method. Also, any missing values in x1–x4 should be replaced by –99. x1 is given a weight of 0.5, x2 is given a weight of 0.5, x3 is given a weight of 0.1, and x4 is given a weight of 0.5.

The second option list defines x5 as an interval variable to be standardized by the RANGE method. If the REPLACE option is specified in the PROC DISTANCE statement, missing values in x5 are replaced by the location estimate from the RANGE method. By default, x5 is given a weight of 1.

The last option list defines x6 as an ordinal variable. The scores are assigned from highest to lowest by its unformatted values. Although the STD= option is not specified, x6 is standardized by the default method

(STD) because there is more than one level of measurements (ratio, interval, and ordinal) in the VAR statement. Again, if the REPLACE option is specified, missing values in x6 are replaced by the location estimate from the STD method. Finally, by default, x6 is given a weight of 1.

More details for the options are explained as follows.

STD=*std-method*

specifies the standardization method. Valid values for *std-method* are MEAN, MEDIAN, SUM, EUCLEN, USTD, STD, RANGE, MIDRANGE, MAXABS, IQR, MAD, ABW, AHUBER, AWAVE, AGK, SPACING, and L. Table 40.9 lists available methods of standardization as well as their corresponding location and scale measures.

Table 40.9 Available Standardization Methods

Method	Scale	Location
MEAN	1	mean
MEDIAN	1	median
SUM	sum	0
EUCLEN	Euclidean length	0
USTD	standard deviation about origin	0
STD	standard deviation	mean
RANGE	range	minimum
MIDRANGE	range/2	midrange
MAXABS	maximum absolute value	0
IQR	interval quartile range	median
MAD	median absolute deviation from median	median
ABW(c)	biweight A-estimate	biweight 1-step M-estimate
AHUBER(c)	Huber A-estimate	Huber 1-step M-estimate
AWAVE(c)	Wave 1-step M-estimate	Wave A-estimate
AGK(p)	AGK estimate (ACECLUS)	mean
SPACING(p)	minimum spacing	mid minimum-spacing
L(p)	L_p	L_p

These standardization methods are further documented in the section on the METHOD= option in the PROC STDIZE statement of the STDIZE procedure (see the section "Standardization Methods" on page 9347 in Chapter 112, "The STDIZE Procedure").

Standardization is not required if there is only one level of measurement, or if only asymmetric nominal and nominal levels are specified; otherwise, standardization is mandatory. When standardization is mandatory, a default method is provided when the STD= option is not specified. You can suppress the mandatory standardization by using the NOSTD option in the PROC DISTANCE statement. See the NOSTD option in the section "PROC DISTANCE Statement" on page 2680 and the section "Mandatory Standardization" on page 2674 for details.

The default method is STD for standardizing interval variables unless METHOD=GOWER or METHOD=DGOWER is specified. If METHOD=GOWER is specified, interval variables are standardized by the RANGE method, and whatever is specified in the STD= option is ignored; if METHOD=DGOWER is specified, the RANGE method is the default standardization method for

interval variables. The MAXABS method is the default standardization method for ratio variables for both the GOWER and DGOWER methods.

Notice that a ratio variable should always be positive.

Table 40.10 lists standardization methods and the levels of measurement that can be accepted by each method. For example, the SUM method can be used to standardize ratio variables but not interval or ordinal variables. Also, the AGK and SPACING methods should not be used to standardize ordinal variables. If you apply AGK and SPACING to ranks, the results are degenerate because all the spacings of a given order are equal.

Table 40.10 Legitimate Levels of Measurements for Each Method

Standardization Method	Legitimate Levels of Measurement
MEAN	ratio, interval, ordinal
MEDIAN	ratio, interval, ordinal
SUM	ratio
EUCLEN	ratio
USTD	ratio
STD	ratio, interval, ordinal
RANGE	ratio, interval, ordinal
MIDRANGE	ratio, interval, ordinal
MAXABS	ratio
IQR	ratio, interval, ordinal
MAD	ratio, interval, ordinal
ABW(c)	ratio, interval, ordinal
AHUBER(c)	ratio, interval, ordinal
AWAVE(c)	ratio, interval, ordinal
AGK(p)	ratio, interval
SPACING(p)	ratio, interval
L(p)	ratio, interval, ordinal

ABSENT=*number* | *qs*

specifies the value to be used as an absence value in an irrelevant absent-absent match for asymmetric nominal variables. The absence value specified here overwrites the absence value specified through the ABSENT= option in the PROC DISTANCE statement for those variables in the current variable list.

An absence value for a variable can be either a numeric value or a quoted string consisting of combinations of characters. For instance, ., –999, "NA" are legal values for the ABSENT= option.

The default for an absence value for a character variable is "NONE" (notice that a blank value is considered a missing value), and the default for an absence value for a numeric variable is 0.

MISSING=*miss-method* | *value*

specifies the method or a numeric value for replacing missing values. If you omit the MISSING= option, the REPLACE option replaces missing values with the location measure given by the STD= option. Specify the MISSING= option when you want to replace missing values with a different value. You can specify any method that is valid in the STD= option. The corresponding location measure is used to replace missing values.

If a numeric value is given, the value replaces missing values after standardizing the data. However, when standardization is not mandatory, you can specify the REPONLY option with the MISSING= option to suppress standardization for cases in which you want only to replace missing values.

If the NOSTD option is specified, there is no standardization, but missing values are replaced by the corresponding location measures or by the numeric value of the MISSING= option. See the section "Missing Values" on page 2699 for details about missing values replacement with and without standardization.

ORDER=ASCENDING | DESCENDING | ASCFORMATTED | DESFORMATTED | DSORDER
ORDER=ASC | DESC | ASCFMT | DESFMT | DATA

specifies the order for assigning score to ordinal variables. The value for the ORDER= option can be one of the following:

ASCENDING	scores are assigned in lowest-to-highest order of unformatted values.
DESCENDING	scores are assigned in highest-to-lowest order of unformatted values.
ASCFORMATTED	scores are assigned in ascending order by their formatted values. This option can be applied to character variables only, since unformatted values are always used for numeric variables.
DESFORMATTED	scores are assigned in descending order by their formatted values. This option can be applied to character variables only, since unformatted values are always used for numeric variables.
DSORDER	scores are assigned according to the order of their appearance in the input data set.

The default value is ASCENDING.

WEIGHTS=*weight-list*

specifies a list of values for weighting individual variables while computing the proximity. Values in this list can be separated by blanks or commas. You can include one or more items of the form *start* TO *stop* BY *increment*. This list should contain at least one weight. The maximum number of weights you can list is equal to the number of variables. If the number of weights is less than the number of variables, the last value in the *weight-list* is used for the rest of the variables; conversely, if the number of weights is greater than the number of variables, the trailing weights are discarded.

The default value is 1.

ID Statement

ID *variable* **;**

The ID statement specifies a single variable to be copied to the OUT= data set and used to generate names for the distance variables. The ID variable must be character.

Typically, each ID value occurs only once in the input data set or, if you use a BY statement, only once within a BY group.

If you specify both the ID and BY statements, the ID variable must have the same values in the same order in each BY group.

COPY Statement

COPY *variables* **;**

The COPY statement specifies a list of additional variables to be copied to the OUT= data set.

BY Statement

BY *variables* **;**

You can specify a BY statement in PROC DISTANCE to obtain separate analyses of observations in groups that are 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 you specify more than one BY statement, only the last one specified is used.

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 NOTSORTED or DESCENDING option in the BY statement in the DISTANCE 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 (in Base SAS software).

For more information about BY-group processing, see the discussion in *SAS Language Reference: Concepts*. For more information about the DATASETS procedure, see the discussion in the *Base SAS Procedures Guide*.

FREQ Statement

FREQ | **FREQUENCY** *variable* **;**

The frequency variable is used both for standardizing variables and for assigning rank scores to ordinal variables. It has no direct effect on computing the distances. It has no direct effect on computing the distances.

For standardizing variables and assigning rank scores, PROC DISTANCE treats the data set as if each observation appeared n times, where n is the value of the FREQ variable for the observation. Nonintegral values of the FREQ variable are truncated to the largest integer less than the FREQ value. If the FREQ variable has a value that is less than 1 or is missing, the observation is not used in the analysis, and distances between that observation and other observations are set to missing

WEIGHT Statement

WGT | **WEIGHT** *variable* **;**

The WEIGHT statement specifies a numeric variable in the input data set with values that are used to weight each observation. This weight variable is used for standardizing variables rather than computing the distances. Only one variable can be specified.

The WEIGHT variable values can be nonintegers. An observation is used in the analysis only if the value of the WEIGHT variable is greater than zero. The WEIGHT variable applies to variables that are standardized by the following *options*: STD=MEAN, STD=SUM, STD=EUCLEN, STD=USTD, STD=STD, STD=AGK, or STD=L.

PROC DISTANCE uses the value of the WEIGHT variable w_i to compute the sample mean, uncorrected sample variances, and sample variances as follows:

$$\overline{x}_w = \sum_i w_i x_i / \sum_i w_i$$

$$u_w^2 = \sum_i w_i x_i^2 / d$$

$$s_w^2 = \sum_i w_i (x_i - \overline{x}_w)^2 / d$$

w_i is the weight value of the ith observation, x_i is the value of the ith observation, and d is the divisor controlled by the VARDEF= option (see the VARDEF= option in the PROC DISTANCE statement for details).

PROC DISTANCE uses the value of the WEIGHT variable to calculate the following statistics for standardization:

MEAN — the weighted mean, $\overline{x}_w$

SUM — the weighted sum, $\sum_i w_i x_i$

USTD — the weighted uncorrected standard deviation, $\sqrt{u_w^2}$

STD — the weighted standard deviation, $\sqrt{s_w^2}$

EUCLEN — the weighted Euclidean length, computed as the square root of the weighted uncorrected sum of squares:

$$\sqrt{\sum_i w_i x_i^2}$$

AGK — the AGK estimate. This estimate is documented further in the ACECLUS procedure as the METHOD=COUNT option. See the discussion of the WEIGHT statement in Chapter 24, "The ACECLUS Procedure," for information about how the WEIGHT variable is applied to the AGK estimate.

L — the L_p estimate. This estimate is documented further in the FASTCLUS procedure as the LEAST= option. See the discussion of the WEIGHT statement in Chapter 42, "The FASTCLUS Procedure," for information about how the WEIGHT variable is used to compute weighted cluster means. Note that the number of clusters is always 1.

Details: DISTANCE Procedure

Proximity Measures

The following notation is used in this section:

v	the number of variables or the dimensionality
x_j	data for observation x and the jth variable, where $j = 1$ to v
y_j	data for observation y and the jth variable, where $j = 1$ to v
w_j	weight for the jth variable from the WEIGHTS= option in the VAR statement. $w_j = 0$ when either x_j or y_j is missing.
W	the sum of total weights. No matter if the observation is missing or not, its weight is added to this metric.
$\bar{x}$	mean for observation x $\bar{x} = \sum_{j=1}^{v} w_j x_j / \sum_{j=1}^{v} w_j$
$\bar{y}$	mean for observation y $\bar{y} = \sum_{j=1}^{v} w_j y_j / \sum_{j=1}^{v} w_j$
$d(x, y)$	the distance or dissimilarity between observations x and y
$s(x, y)$	the similarity between observations x and y

The factor $W / \sum_{j=1}^{v} w_j$ is used to adjust some of the proximity measures for missing values.

Methods That Accept All Measurement Levels

GOWER Gower's similarity

$s_1(x, y) = \sum_{j=1}^{v} w_j \delta_{x,y}^j d_{x,y}^j / \sum_{j=1}^{v} w_j \delta_{x,y}^j$

$\delta_{x,y}^j$ is computed as follows:

For nominal, ordinal, interval, or ratio variable,

$$\delta_{x,y}^j = 1$$

For asymmetric nominal variable,

$$\delta_{x,y}^j = 1, \text{ if either } x_j \text{ or } y_j \text{ is present}$$
$$\delta_{x,y}^j = 0, \text{ if both } x_j \text{ and } y_j \text{ are absent}$$

For nominal or asymmetric nominal variable,

$$d_{x,y}^j = 1, \text{ if } x_j = y_j$$
$$d_{x,y}^j = 0, \text{ if } x_j \neq y_j$$

For ordinal, interval, or ratio variable,

$$d_{x,y}^j = 1 - |x_j - y_j|$$

DGOWER 1 minus Gower
$d_2(x,y) = 1 - s_1(x,y)$

Methods That Accept Ratio, Interval, and Ordinal Variables

EUCLID Euclidean distance
$d_3(x,y) = \sqrt{(\sum_{j=1}^{v} w_j (x_j - y_j)^2) W / (\sum_{j=1}^{v} w_j)}$

SQEUCLID squared Euclidean distance
$d_4(x,y) = (\sum_{j=1}^{v} w_j (x_j - y_j)^2) W / (\sum_{j=1}^{v} w_j)$

SIZE size distance
$d_5(x,y) = |\sum_{j=1}^{v} w_j (x_j - y_j)| \sqrt{W} / (\sum_{j=1}^{v} w_j)$

SHAPE shape distance
$d_6(x,y) = \sqrt{(\sum_{j=1}^{v} w_j [(x_j - \bar{x}) - (y_j - \bar{y})]^2) W / (\sum_{j=1}^{v} w_j)}$

NOTE: squared shape distance plus squared size distance equals squared Euclidean distance.

COV covariance similarity coefficient
$s_7(x,y) = \sum_{j=1}^{v} w_j (x_j - \bar{x})(y_j - \bar{y}) / vardiv$, where

$$\begin{aligned} vardiv &= v \text{ if VARDEF} = \text{N} \\ &= v - 1 \text{ if VARDEF} = \text{DF} \\ &= \sum_{j=1}^{v} w_j \text{ if VARDEF} = \text{WEIGHT} \\ &= \sum_{j=1}^{v} w_j - 1 \text{ if VARDEF} = \text{WDF} \end{aligned}$$

CORR correlation similarity coefficient
$s_8(x,y) = \frac{\sum_{j=1}^{v} w_j (x_j - \bar{x})(y_j - \bar{y})}{\sqrt{\sum_{j=1}^{v} w_j (x_j - \bar{x})^2 \sum_{j=1}^{v} w_j (y_j - \bar{y})^2}}$

DCORR correlation transformed to Euclidean distance as sqrt(1–CORR)
$d_9(x,y) = \sqrt{1 - s_8(x,y)}$

SQCORR squared correlation
$s_{10}(x,y) = \frac{[\sum_{j=1}^{v} w_j (x_j - \bar{x})(y_j - \bar{y})]^2}{\sum_{j=1}^{v} w_j (x_j - \bar{x})^2 \sum_{j=1}^{v} w_j (y_j - \bar{y})^2}$

DSQCORR squared correlation transformed to squared Euclidean distance as (1–SQCORR)

$d_{11}(x,y) = 1 - s_{10}(x,y)$

L(p) Minkowski (L_p) distance, where p is a positive numeric value

$$d_{12}(x,y) = [(\sum_{j=1}^{v} w_j |x_j - y_j|^p) W / (\sum_{j=1}^{v} w_j)]^{1/p}$$

CITYBLOCK L_1

$$d_{13}(x,y) = (\sum_{j=1}^{v} w_j |x_j - y_j|) W / (\sum_{j=1}^{v} w_j)$$

CHEBYCHEV L_∞

$$d_{14}(x,y) = \max_{j=1}^{v} w_j |x_j - y_j|$$

POWER(*p,r*) generalized Euclidean distance, where *p* is a nonnegative numeric value and *r* is a positive numeric value. The distance between two observations is the *r*th root of sum of the absolute differences to the *p*th power between the values for the observations:

$$d_{15}(x,y) = [(\sum_{j=1}^{v} w_j |x_j - y_j|^p) W / (\sum_{j=1}^{v} w_j)]^{1/r}$$

Methods That Accept Ratio Variables

SIMRATIO similarity ratio

$$s_{16}(x,y) = \frac{\sum_j^v w_j (x_j y_j)}{\sum_{j=1}^{v} w_j (x_j y_j) + \sum_j^v w_j (x_j - y_j)^2}$$

DISRATIO one minus similarity ratio

$$d_{17}(x,y) = 1 - s_{16}(x,y)$$

NONMETRIC Lance-Williams nonmetric coefficient

$$d_{18}(x,y) = \frac{\sum_{j=1}^{v} w_j |x_j - y_j|}{\sum_{j=1}^{v} w_j (x_j + y_j)}$$

CANBERRA Canberra metric coefficient. See Sneath and Sokal (1973, pp. 125–126)

$$d_{19}(x,y) = \sum_{j=1}^{v} \frac{w_j |x_j - y_j|}{w_j (x_j + y_j)}$$

COSINE cosine coefficient

$$s_{20}(x,y) = \frac{\sum_{j=1}^{v} w_j (x_j y_j)}{\sqrt{\sum_{j=1}^{v} w_j {x_j}^2 \sum_{j=1}^{v} w_j {y_j}^2}}$$

DOT dot (inner) product coefficient

$$s_{21}(x,y) = [\sum_{j=1}^{v} w_j (x_j y_j)] / \sum_{j=1}^{v} w_j$$

OVERLAP sum of the minimum values

$$s_{22}(x,y) = \sum_{j=1}^{v} w_j [\min(x_j, y_j)]$$

DOVERLAP maximum of the sum of the *x* and the sum of *y* minus overlap

$$d_{23}(x,y) = \max(\sum_{j=1}^{v} w_j x_j, \sum_{j=1}^{v} w_j y_j) - s_{22}(x,y)$$

CHISQ chi-square

If the data represent the frequency counts, chi-square dissimilarity between two sets of

frequencies can be computed. A 2-by-v contingency table is illustrated to explain how the chi-square dissimilarity is computed as follows:

	Variable				Row
Observation	Var 1	Var 2	...	Var v	Sum
X	x_1	x_2	...	x_v	r_x
Y	y_1	y_2	...	y_v	r_y
Column Sum	c_1	c_2	...	c_v	T

where

$$\begin{aligned} r_x &= \sum_{j=1}^{v} w_j x_j \\ r_y &= \sum_{j=1}^{v} w_j y_j \\ c_j &= w_j(x_j + y_j) \\ T &= r_x + r_y = \sum_{j=1}^{v} c_j \end{aligned}$$

The chi-square measure is computed as follows:

$$d_{24}(x,y) = \left(\sum_{j=1}^{v} \frac{(w_j x_j - E(x_j))^2}{E(x_j)} + \sum_{j=1}^{v} \frac{(w_j y_j - E(y_j))^2}{E(y_j)}\right) W / \left(\sum_{j=1}^{v} w_j\right)$$

where for j= 1, 2, ..., v

$$\begin{aligned} E(x_j) &= r_x c_j / T \\ E(y_j) &= r_y c_j / T \end{aligned}$$

CHI — square root of chi-square
$d_{25}(x,y) = \sqrt{d_{23}(x,y)}$

PHISQ — phi-square
This is the CHISQ dissimilarity normalized by the sum of weights
$d_{26}(x,y) = d_{24}(x,y)/\left(\sum_{j=1}^{v} w_j\right)$

PHI — square root of phi-square
$d_{27}(x,y) = \sqrt{d_{25}(x,y)}$

Methods That Accept Symmetric Nominal Variables

The following notation is used for computing $d_{28}(x,y)$ to $s_{35}(x,y)$. Notice that only the nonmissing pairs are discussed below; all the pairs with at least one missing value will be excluded from any of the computations in the following section because $w_j = 0$, if either x_j or y_j is missing.

M — nonmissing matches
$M = \sum_{j=1}^{v} w_j \delta_{x,y}^{j}$, where

$$\begin{aligned} \delta_{x,y}^{j} &= 1, \text{ if } x_j = y_j \\ \delta_{x,y}^{j} &= 0, \text{ otherwise} \end{aligned}$$

X	nonmissing mismatches $X = \sum_{j=1}^{v} w_j \delta_{x,y}^j$, where $\delta_{x,y}^j = 1$, if $x_j \neq y_j$ $\delta_{x,y}^j = 0$, otherwise
N	total nonmissing pairs $N = \sum_{j=1}^{v} w_j$
HAMMING	Hamming distance $d_{28}(x, y) = X$
MATCH	simple matching coefficient $s_{29}(x, y) = M/N$
DMATCH	simple matching coefficient transformed to Euclidean distance $d_{30}(x, y) = \sqrt{1 - M/N} = \sqrt{(X/N)}$
DSQMATCH	simple matching coefficient transformed to squared Euclidean distance $d_{31}(x, y) = 1 - M/N = X/N$
HAMANN	Hamann coefficient $s_{32}(x, y) = (M - X)/N$
RT	Roger and Tanimoto $s_{33}(x, y) = M/(M + 2X)$
SS1	Sokal and Sneath 1 $s_{34}(x, y) = 2M/(2M + X)$
SS3	Sokal and Sneath 3. The coefficient between an observation and itself is always indeterminate (missing) since there is no mismatch. $s_{35}(x, y) = M/X$

The following notation is used for computing $s_{36}(x, y)$ to $d_{41}(x, y)$. Notice that only the nonmissing pairs are discussed in the following section; all the pairs with at least one missing value are excluded from any of the computations in the following section because $w_j = 0$, if either x_j or y_j is missing.

Also, the observed nonmissing data of an asymmetric binary variable can have only two possible outcomes: presence or absence. Therefore, the notation, *PX* (present mismatches), always has a value of zero for an asymmetric binary variable.

The following methods distinguish between the presence and absence of attributes.

X mismatches with at least one present

$X = \sum_{j=1}^{v} w_j \delta_{x,y}^{j}$, where

$$\begin{aligned} \delta_{x,y}^{j} &= 1, \text{ if } x_j \neq y_j \text{ and not both } x_j \text{ and } y_j \text{ are absent} \\ \delta_{x,y}^{j} &= 0, \text{ otherwise} \end{aligned}$$

PM present matches

$PM = \sum_{j=1}^{v} w_j \delta_{x,y}^{j}$, where

$$\begin{aligned} \delta_{x,y}^{j} &= 1, \text{ if } x_j = y_j \text{ and both } x_j \text{ and } y_j \text{ are present} \\ \delta_{x,y}^{j} &= 0, \text{ otherwise} \end{aligned}$$

PX present mismatches

$PX = \sum_{j=1}^{v} w_j \delta_{x,y}^{j}$, where

$$\begin{aligned} \delta_{x,y}^{j} &= 1, \text{ if } x_j \neq y_j \text{ and both } x_j \text{ and } y_j \text{ are present} \\ \delta_{x,y}^{j} &= 0, \text{ otherwise} \end{aligned}$$

PP both present = *PM* + *PX*

P at least one present = *PM* + *X*

PAX present-absent mismatches

$PAX = \sum_{j=1}^{v} w_j \delta_{x,y}^{j}$, where

$$\begin{aligned} \delta_{x,y}^{j} &= 1, \text{ if } x_j \neq y_j \text{ and either } x_j \text{ is present and } y_j \text{ is absent or} \\ &\quad\ x_j \text{ is absent and } y_j \text{ is present} \\ \delta_{x,y}^{j} &= 0 \text{ otherwise} \end{aligned}$$

N total nonmissing pairs

$N = \sum_{j=1}^{v} w_j$

Methods That Accept Asymmetric Nominal and Ratio Variables

JACCARD Jaccard similarity coefficient

The JACCARD method is equivalent to the SIMRATIO method if there are only ratio variables; if there are both ratio and asymmetric nominal variables, the coefficient is computed as sum of the coefficient from the ratio variables (SIMRATIO) and the coefficient from the asymmetric nominal variables.

$s_{36}(x, y) = s_{16}(x, y) + PM/P$

DJACCARD Jaccard dissimilarity coefficient

The DJACCARD method is equivalent to the DISRATIO method if there are only ratio variables; if there are both ratio and asymmetric nominal variables, the coefficient is computed as sum of the coefficient from the ratio variables (DISRATIO) and the coefficient from the asymmetric nominal variables.

$d_{37}(x, y) = d_{17}(x, y) + X/P$

Methods That Accept Asymmetric Nominal Variables

DICE Dice coefficient or Czekanowski/Sorensen similarity coefficient
$s_{38}(x, y) = 2PM/(P + PM)$

RR Russell and Rao. This is the binary equivalent of the dot product coefficient.
$s_{39}(x, y) = PM/N$

BLWNM | BRAYCURTIS Binary Lance and Williams, also known as Bray and Curtis coefficient

$d_{40}(x, y) = X/(PAX + 2PP)$

K1 Kulcynski 1. The coefficient between an observation and itself is always indeterminate (missing) since there is no mismatch.
$d_{41}(x, y) = PM/X$

Missing Values

Standardization versus No Standardization

You can replace the missing values with or without standardization. Missing values are replaced after standardization by specifying either the REPLACE option in the PROC DISTANCE statement or the MISSING= option in the VAR statement.

To replace missing values without standardization, use the following two options:

- the NOSTD option in the PROC DISTANCE statement. The NOSTD option suppresses standardization but still replaces the missing values with the location of the method or the numeric value specified in the MISSING= option in the VAR statement.
- the REPONLY option in the PROC DISTANCE statement. PROC DISTANCE replaces missing values with the location of the standardization method or with the numeric value specified in the MISSING= option in the VAR statement. This approach assumes that standardization is not mandatory (see the section "Standardization" on page 2674).

Eliminating Observations with Missing Values

If you specify the NOMISS option, PROC DISTANCE omits observations with any missing values in the analyzed variables from computation of the location and scale measures.

Distance Measures

If you specify the NOMISS option, PROC DISTANCE generates missing distance for observations with missing values. If the NOMISS option is not specified, the sum of total weights, no matter if an observation is missing or not, is incorporated into the computation of some of the proximity measures. See the section "Details: DISTANCE Procedure" on page 2693 for the formulas and descriptions.

Formatted versus Unformatted Values

PROC DISTANCE uses the formatted values from a character variable, if the variable has a format—for example, one assigned by a format statement. PROC DISTANCE uses the unformatted values from a numeric variable, even if it has a format.

Output Data Sets

OUT= Data Set

The DISTANCE procedure always produces an output data set, regardless of whether you specify the OUT= option in the PROC DISTANCE statement. PROC DISTANCE displays no output. Use PROC PRINT to display the output data set.

The output data set contains the following variables:

- the ID variable, if any
- the BY variables, if any
- the COPY variables, if any
- the FREQ variable, if any
- the WEIGHT variable, if any
- the new distance variables, named from PREFIX= options along with the ID values, or from the default values

The output data set is of type TYPE=DISTANCE or TYPE=SIMILAR. See the METHOD= option for more information about the output data set types. Data set types do not persist when you copy or modify a data set. You must specify the TYPE= data set option for the new data set, as in the following example:

```
data dist2(type=distance);
   set dist;
run;
```

See the OUT= option for more information about data set type persistence.

OUTSDZ= Data Set

The output data set is a copy of the DATA= data set except that the analyzed variables have been standardized. Analyzed variables are those listed in the VAR statement.

Examples: DISTANCE Procedure

Example 40.1: Divorce Grounds – the Jaccard Coefficient

A wide variety of distance and similarity measures are used in cluster analysis (Anderberg 1973; Sneath and Sokal 1973). If your data are in coordinate form and you want to use a non-Euclidean distance for clustering, you can compute a distance matrix by using the DISTANCE procedure.

Similarity measures must be converted to dissimilarities before being used in PROC CLUSTER. Such conversion can be done in a variety of ways, such as taking reciprocals or subtracting from a large value. The choice of conversion method depends on the application and the similarity measure. If applicable, PROC DISTANCE provides a corresponding dissimilarity measure for each similarity measure.

In the following example, the observations are states. Binary-valued variables correspond to various grounds for divorce and indicate whether the grounds for divorce apply in each of the U.S. states. A value of "1" indicates that the ground for divorce applies, and a value of "0" indicates the opposite. The 0-0 matches are treated as totally irrelevant; therefore, each variable has an asymmetric nominal level of measurement. The absence value is 0.

The DISTANCE procedure is used to compute the Jaccard coefficient (Anderberg 1973, pp. 89, 115, and 117) between each pair of states. The Jaccard coefficient is defined as the number of variables that are coded as 1 for both states divided by the number of variables that are coded as 1 for either or both states. Since dissimilarity measures are required by PROC CLUSTER, the DJACCARD coefficient is selected. Output 40.1.1 displays the distance matrix between the first 10 states.

The CENTROID method is used to perform the cluster analysis, and the resulting tree diagram from PROC CLUSTER is saved into the tree output data set. Output 40.1.2 displays the cluster history.

The TREE procedure generates nine clusters in the output data set out. After being sorted by the state, the out data set is then merged with the input data set divorce. After being sorted by the state, the merged data set is printed to display the cluster membership as shown in Output 40.1.3.

The following statements produce Output 40.1.1 through Output 40.1.3:

```
data divorce;
   length State $ 15;
   input State &$
         Incompatibility Cruelty Desertion Non_Support Alcohol
         Felony Impotence Insanity Separation @@;
   datalines;
Alabama          1 1 1 1 1 1 1 1 1    Alaska           1 1 1 0 1 1 1 1 0
Arizona          1 0 0 0 0 0 0 0 0    Arkansas         0 1 1 1 1 1 1 1 1
California       1 0 0 0 0 0 0 1 0    Colorado         1 0 0 0 0 0 0 0 0
Connecticut      1 1 1 1 1 1 0 1 1    Delaware         1 0 0 0 0 0 0 0 1
```

```
   ... more lines ...

Wisconsin        1 0 0 0 0 0 0 0 1    Wyoming          1 0 0 0 0 0 0 1 1
;

title 'Grounds for Divorce';
proc distance data=divorce method=djaccard absent=0 out=distjacc;
   var anominal(Incompatibility--Separation);
   id state;
run;

proc print data=distjacc(obs=10);
   id state; var alabama--georgia;
   title2 'First 10 States';
run;
title2;

proc cluster data=distjacc method=centroid
             pseudo outtree=tree;
   id state;
   var alabama--wyoming;
run;

proc tree data=tree noprint n=9 out=out;
   id state;
run;

proc sort;
   by state;
run;

data clus;
   merge divorce out;
   by state;
run;

proc sort;
   by cluster;
run;

proc print;
   id state;
   var Incompatibility--Separation;
   by cluster;
run;
```

Output 40.1.1 Distance Matrix Based on the Jaccard Coefficient

Grounds for Divorce
First 10 States

State	Alabama	Alaska	Arizona	Arkansas	California	Colorado	Connecticut	Delaware	Florida	Georgia
Alabama	0.00000	.	.	.	.	.	.	.	.	.
Alaska	0.22222	0.00000	.	.	.	.	.	.	.	.
Arizona	0.88889	0.85714	0.00000	.	.	.	.	.	.	.
Arkansas	0.11111	0.33333	1.00000	0.00000	.	.	.	.	.	.
California	0.77778	0.71429	0.50000	0.88889	0.00000	.	.	.	.	.
Colorado	0.88889	0.85714	0.00000	1.00000	0.50000	0.00000	.	.	.	.
Connecticut	0.11111	0.33333	0.87500	0.22222	0.75000	0.87500	0.00000	.	.	.
Delaware	0.77778	0.87500	0.50000	0.88889	0.66667	0.50000	0.75000	0.00000	.	.
Florida	0.77778	0.71429	0.50000	0.88889	0.00000	0.50000	0.75000	0.66667	0.00000	.
Georgia	0.22222	0.00000	0.85714	0.33333	0.71429	0.85714	0.33333	0.87500	0.71429	0

Output 40.1.2 Clustering History

Grounds for Divorce

The CLUSTER Procedure
Centroid Hierarchical Cluster Analysis

Root-Mean-Square Distance Between Observations	0.694873

Output 40.1.2 *continued*

Cluster History

Number of Clusters	Clusters Joined		Freq	Pseudo F Statistic	Pseudo t-Squared	Norm Centroid Distance	Tie
49	Arizona	Colorado	2	.	.	0	T
48	California	Florida	2	.	.	0	T
47	Alaska	Georgia	2	.	.	0	T
46	Delaware	Hawaii	2	.	.	0	T
45	Connecticut	Idaho	2	.	.	0	T
44	CL49	Iowa	3	.	.	0	T
43	CL47	Kansas	3	.	.	0	T
42	CL44	Kentucky	4	.	.	0	T
41	CL42	Michigan	5	.	.	0	T
40	CL41	Minnesota	6	.	.	0	T
39	CL43	Mississippi	4	.	.	0	T
38	CL40	Missouri	7	.	.	0	T
37	CL38	Montana	8	.	.	0	T
36	CL37	Nebraska	9	.	.	0	T
35	North Dakota	Oklahoma	2	.	.	0	T
34	CL36	Oregon	10	.	.	0	T
33	Massachusetts	Rhode Island	2	.	.	0	T
32	New Hampshire	Tennessee	2	.	.	0	T
31	CL46	Washington	3	.	.	0	T
30	CL31	Wisconsin	4	.	.	0	T
29	Nevada	Wyoming	2	.	.	0	
28	Alabama	Arkansas	2	1561	.	0.1599	T
27	CL33	CL32	4	479	.	0.1799	T
26	CL39	CL35	6	265	.	0.1799	T
25	CL45	West Virginia	3	231	.	0.1799	
24	Maryland	Pennsylvania	2	199	.	0.2399	
23	CL28	Utah	3	167	3.2	0.2468	
22	CL27	Ohio	5	136	5.4	0.2698	
21	CL26	Maine	7	111	8.9	0.2998	
20	CL23	CL21	10	75.2	8.7	0.3004	
19	CL25	New Jersey	4	71.8	6.5	0.3053	T
18	CL19	Texas	5	69.1	2.5	0.3077	
17	CL20	CL22	15	48.7	9.9	0.3219	
16	New York	Virginia	2	50.1	.	0.3598	
15	CL18	Vermont	6	49.4	2.9	0.3797	
14	CL17	Illinois	16	47.0	3.2	0.4425	
13	CL14	CL15	22	29.2	15.3	0.4722	
12	CL48	CL29	4	29.5	.	0.4797	T
11	CL13	CL24	24	27.6	4.5	0.5042	
10	CL11	South Dakota	25	28.4	2.4	0.5449	
9	Louisiana	CL16	3	30.3	3.5	0.5844	
8	CL34	CL30	14	23.3	.	0.7196	
7	CL8	CL12	18	19.3	15.0	0.7175	
6	CL10	South Carolina	26	21.4	4.2	0.7384	
5	CL6	New Mexico	27	24.0	4.7	0.8303	
4	CL5	Indiana	28	28.9	4.1	0.8343	

Output 40.1.2 *continued*

Cluster History

Number of Clusters	Clusters Joined		Freq	Pseudo F Statistic	Pseudo t-Squared	Norm Centroid Distance	Tie
3	CL4	CL9	31	31.7	10.9	0.8472	
2	CL3	North Carolina	32	55.1	4.1	1.0017	
1	CL2	CL7	50	.	55.1	1.0663	

Output 40.1.3 Cluster Membership

Grounds for Divorce

CLUSTER=1

State	Incompatibility	Cruelty	Desertion	Non_Support	Alcohol	Felony	Impotence	Insanity	Separation
Arizona	1	0	0	0	0	0	0	0	0
Colorado	1	0	0	0	0	0	0	0	0
Iowa	1	0	0	0	0	0	0	0	0
Kentucky	1	0	0	0	0	0	0	0	0
Michigan	1	0	0	0	0	0	0	0	0
Minnesota	1	0	0	0	0	0	0	0	0
Missouri	1	0	0	0	0	0	0	0	0
Montana	1	0	0	0	0	0	0	0	0
Nebraska	1	0	0	0	0	0	0	0	0
Oregon	1	0	0	0	0	0	0	0	0

CLUSTER=2

State	Incompatibility	Cruelty	Desertion	Non_Support	Alcohol	Felony	Impotence	Insanity	Separation
California	1	0	0	0	0	0	0	1	0
Florida	1	0	0	0	0	0	0	1	0
Nevada	1	0	0	0	0	0	0	1	1
Wyoming	1	0	0	0	0	0	0	1	1

Output 40.1.3 *continued*

CLUSTER=3

State	Incompatibility	Cruelty	Desertion	Non_Support	Alcohol	Felony	Impotence	Insanity	Separation
Alabama	1	1	1	1	1	1	1	1	1
Alaska	1	1	1	0	1	1	1	1	0
Arkansas	0	1	1	1	1	1	1	1	1
Connecticut	1	1	1	1	1	1	0	1	1
Georgia	1	1	1	0	1	1	1	1	0
Idaho	1	1	1	1	1	1	0	1	1
Illinois	0	1	1	0	1	1	1	0	0
Kansas	1	1	1	0	1	1	1	1	0
Maine	1	1	1	1	1	0	1	1	0
Maryland	0	1	1	0	0	1	1	1	1
Massachusetts	1	1	1	1	1	1	1	0	1
Mississippi	1	1	1	0	1	1	1	1	0
New Hampshire	1	1	1	1	1	1	1	0	0
New Jersey	0	1	1	0	1	1	0	1	1
North Dakota	1	1	1	1	1	1	1	1	0
Ohio	1	1	1	0	1	1	1	0	1
Oklahoma	1	1	1	1	1	1	1	1	0
Pennsylvania	0	1	1	0	0	1	1	1	0
Rhode Island	1	1	1	1	1	1	1	0	1
South Dakota	0	1	1	1	1	1	0	0	0
Tennessee	1	1	1	1	1	1	1	0	0
Texas	1	1	1	0	0	1	0	1	1
Utah	0	1	1	1	1	1	1	1	0
Vermont	0	1	1	1	0	1	0	1	1
West Virginia	1	1	1	0	1	1	0	1	1

CLUSTER=4

State	Incompatibility	Cruelty	Desertion	Non_Support	Alcohol	Felony	Impotence	Insanity	Separation
Delaware	1	0	0	0	0	0	0	0	1
Hawaii	1	0	0	0	0	0	0	0	1
Washington	1	0	0	0	0	0	0	0	1
Wisconsin	1	0	0	0	0	0	0	0	1

CLUSTER=5

State	Incompatibility	Cruelty	Desertion	Non_Support	Alcohol	Felony	Impotence	Insanity	Separation
Louisiana	0	0	0	0	0	1	0	0	1
New York	0	1	1	0	0	1	0	0	1
Virginia	0	1	0	0	0	1	0	0	1

CLUSTER=6

State	Incompatibility	Cruelty	Desertion	Non_Support	Alcohol	Felony	Impotence	Insanity	Separation
South Carolina	0	1	1	0	1	0	0	0	1

Output 40.1.3 *continued*

CLUSTER=7

State	Incompatibility	Cruelty	Desertion	Non_Support	Alcohol	Felony	Impotence	Insanity	Separation
New Mexico	1	1	1	0	0	0	0	0	0

CLUSTER=8

State	Incompatibility	Cruelty	Desertion	Non_Support	Alcohol	Felony	Impotence	Insanity	Separation
Indiana	1	0	0	0	0	1	1	1	0

CLUSTER=9

State	Incompatibility	Cruelty	Desertion	Non_Support	Alcohol	Felony	Impotence	Insanity	Separation
North Carolina	0	0	0	0	0	0	1	1	1

Example 40.2: Financial Data – Stock Dividends

The following data set contains the average dividend yields for 15 utility stocks in the United States. The observations are names of the companies, and the variables correspond to the annual dividend yields for the period 1986–1990. The objective is to group similar stocks into clusters.

Before the cluster analysis is performed, the correlation similarity is chosen for measuring the closeness between each observation. Since distance type of measures are required by PROC CLUSTER, METHOD=DCORR is used in the PROC DISTANCE statement to transform the correlation measures to the distance measures. Notice that in Output 40.2.1, all the values in the distance matrix are between 0 and $\sqrt{2}$.

PROC CLUSTER performs hierarchical clustering by using agglomerative methods based on the distance data created from the previous PROC DISTANCE statement. Since the cubic clustering criterion is not suitable for distance data, only the pseudo F statistic is requested to identify the number of clusters.

The two clustering methods are Ward's and the average linkage methods. Since the results of the pseudo t^2 statistic from both Ward's and the average linkage methods contain many missing values, only the plot of the pseudo F statistic versus the number of clusters is requested along with the dendrogram by specifying PLOTS(ONLY)=(PSF DENDROGRAM) in the PROC CLUSTER statement.

Both Output 40.2.2 and Output 40.2.3 suggest four clusters. Both methods produce the same clustering result, as shown in Output 40.2.4 and Output 40.2.5. The four clusters are as follows:

- Cincinnati G&E and Detroit Edison
- Texas Utilities and Pennsylvania Power & Light
- Union Electric, Iowa-Ill Gas & Electric, Oklahoma Gas & Electric, and Wisconsin Energy
- Orange & Rockland Utilities, Kentucky Utilities, Kansas Power & Light, Allegheny Power, Green Mountain Power, Dominion Resources, and Minnesota Power & Light

```
title 'Stock Dividends';

data stock;
   length Company $ 27;
   input Company &$  Div_1986 Div_1987 Div_1988 Div_1989 Div_1990;
   datalines;
Cincinnati G&E               8.4    8.2    8.4    8.1    8.0
Texas Utilities              7.9    8.9   10.4    8.9    8.3
Detroit Edison               9.7   10.7   11.4    7.8    6.5
Orange & Rockland Utilities  6.5    7.2    7.3    7.7    7.9
Kentucky Utilities           6.5    6.9    7.0    7.2    7.5
Kansas Power & Light         5.9    6.4    6.9    7.4    8.0
Union Electric               7.1    7.5    8.4    7.8    7.7
Dominion Resources           6.7    6.9    7.0    7.0    7.4
Allegheny Power              6.7    7.3    7.8    7.9    8.3
Minnesota Power & Light      5.6    6.1    7.2    7.0    7.5
Iowa-Ill Gas & Electric      7.1    7.5    8.5    7.8    8.0
Pennsylvania Power & Light   7.2    7.6    7.7    7.4    7.1
Oklahoma Gas & Electric      6.1    6.7    7.4    6.7    6.8
Wisconsin Energy             5.1    5.7    6.0    5.7    5.9
Green Mountain Power         7.1    7.4    7.8    7.8    8.3
;

proc distance data=stock method=dcorr out=distdcorr;
   var interval(div_1986 div_1987 div_1988 div_1989 div_1990);
   id company;
run;

proc print data=distdcorr;
   id company;
   title2 'Distance Matrix for 15 Utility Stocks';
run;
title2;

ods graphics on;

/* compute pseudo statistic versus number of clusters and create plot */
proc cluster data=distdcorr method=ward pseudo plots(only)=(psf dendrogram);
   id company;
run;

/* compute pseudo statistic versus number of clusters and create plot */
proc cluster data=distdcorr method=average pseudo plots(only)=(psf dendrogram);
   id company;
run;
```

Output 40.2.1 Distance Matrix Based on the DCORR Coefficient

Stock Dividends
Distance Matrix for 15 Utility Stocks

Company	Cincinnati_G_E	Texas_Utilities	Detroit_Edison	Orange___Rockland_Utilities	Kentucky_Utilities
Cincinnati G&E	0.00000	.	.	.	.
Texas Utilities	0.82056	0.00000	.	.	.
Detroit Edison	0.40511	0.65453	0.00000	.	.
Orange & Rockland Utilities	1.35380	0.88583	1.27306	0.00000	.
Kentucky Utilities	1.35581	0.92539	1.29382	0.12268	0.00000
Kansas Power & Light	1.34227	0.94371	1.31696	0.19905	0.12874
Union Electric	0.98516	0.29043	0.89048	0.68798	0.71824
Dominion Resources	1.32945	0.96853	1.29016	0.33290	0.21510
Allegheny Power	1.30492	0.81666	1.24565	0.17844	0.15759

Company	Kansas_Power___Light	Union_Electric	Dominion_Resources	Allegheny_Power
Cincinnati G&E	.	.	.	.
Texas Utilities	.	.	.	.
Detroit Edison	.	.	.	.
Orange & Rockland Utilities	.	.	.	.
Kentucky Utilities	.	.	.	.
Kansas Power & Light	0.00000	.	.	.
Union Electric	0.72082	0.00000	.	.
Dominion Resources	0.24189	0.76587	0.00000	.
Allegheny Power	0.17029	0.58452	0.27819	0.00000

Company	Minnesota_Power___Light	Iowa_Ill_Gas___Electric	Pennsylvania_Power___Light
Cincinnati G&E	.	.	.
Texas Utilities	.	.	.
Detroit Edison	.	.	.
Orange & Rockland Utilities	.	.	.
Kentucky Utilities	.	.	.
Kansas Power & Light	.	.	.
Union Electric	.	.	.
Dominion Resources	.	.	.
Allegheny Power	.	.	.

Company	Oklahoma_Gas___Electric	Wisconsin_Energy	Green_Mountain_Power
Cincinnati G&E	.	.	.
Texas Utilities	.	.	.
Detroit Edison	.	.	.
Orange & Rockland Utilities	.	.	.
Kentucky Utilities	.	.	.
Kansas Power & Light	.	.	.
Union Electric	.	.	.
Dominion Resources	.	.	.
Allegheny Power	.	.	.

Output 40.2.1 *continued*

Stock Dividends
Distance Matrix for 15 Utility Stocks

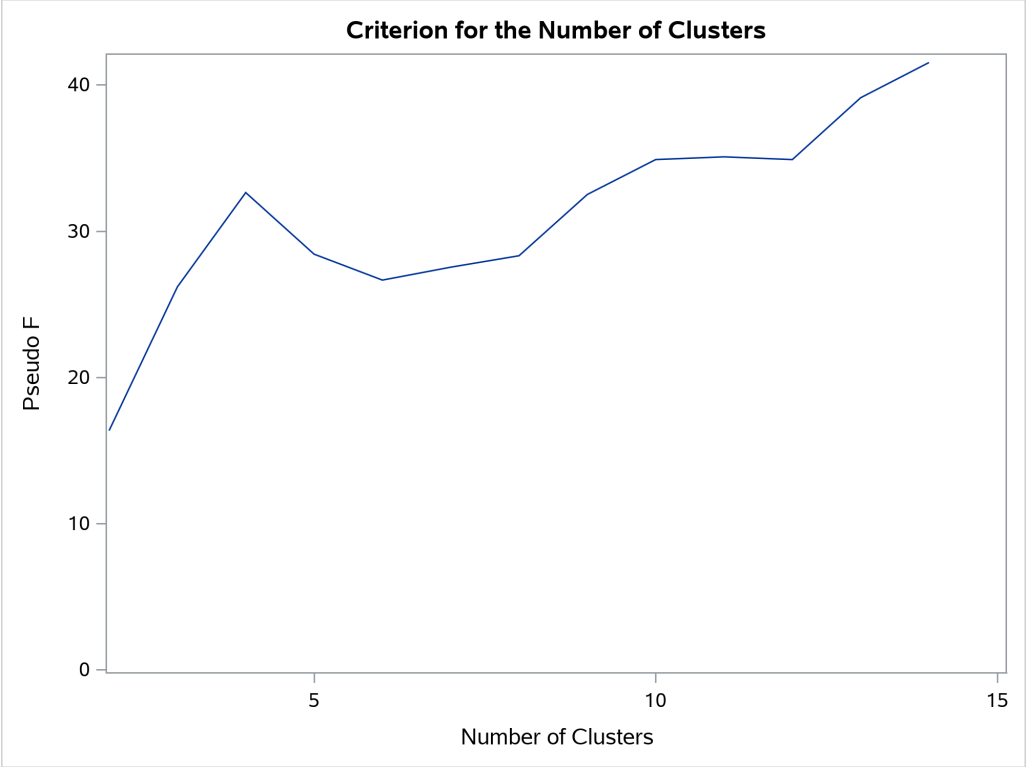
Company	Cincinnati_G_E	Texas_Utilities	Detroit_Edison	Orange___Rockland_Utilities	Kentucky_Utilities
Minnesota Power & Light	1.24069	0.74082	1.20432	0.32581	0.30462
Iowa-Ill Gas & Electric	1.04924	0.43100	0.97616	0.61166	0.61760
Pennsylvania Power & Light	0.74931	0.37821	0.44256	1.03566	1.08878
Oklahoma Gas & Electric	1.00604	0.30141	0.86200	0.68021	0.70259
Wisconsin Energy	1.17988	0.54830	1.03081	0.45013	0.47184
Green Mountain Power	1.30397	0.88063	1.27176	0.26948	0.17909

Company	Kansas_Power___Light	Union_Electric	Dominion_Resources	Allegheny_Power
Minnesota Power & Light	0.27231	0.48372	0.35733	0.15615
Iowa-Ill Gas & Electric	0.61736	0.16923	0.63545	0.47900
Pennsylvania Power & Light	1.12876	0.63285	1.14354	1.02358
Oklahoma Gas & Electric	0.73158	0.17122	0.72977	0.58391
Wisconsin Energy	0.53381	0.37405	0.51969	0.37522
Green Mountain Power	0.15377	0.64869	0.17360	0.13958

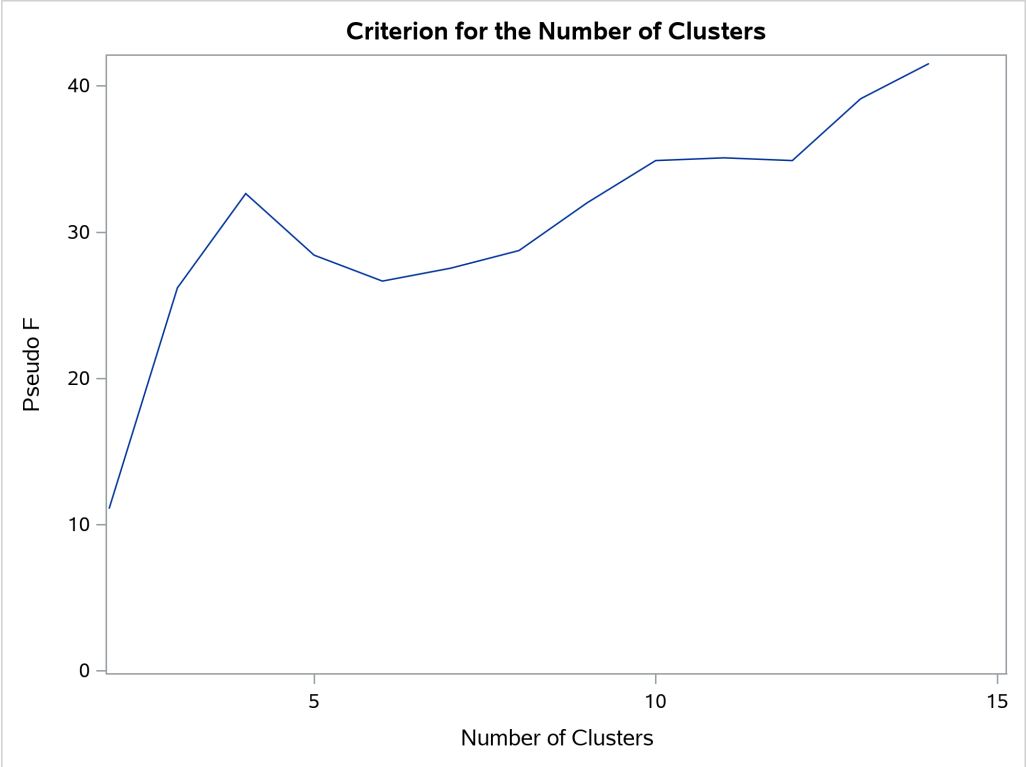
Company	Minnesota_Power___Light	Iowa_Ill_Gas___Electric	Pennsylvania_Power___Light
Minnesota Power & Light	0.00000	.	.
Iowa-Ill Gas & Electric	0.36368	0.00000	.
Pennsylvania Power & Light	0.99384	0.75596	0.00000
Oklahoma Gas & Electric	0.50744	0.19673	0.60216
Wisconsin Energy	0.36319	0.30259	0.76085
Green Mountain Power	0.19370	0.52083	1.09269

Company	Oklahoma_Gas___Electric	Wisconsin_Energy	Green_Mountain_Power
Minnesota Power & Light	.	.	.
Iowa-Ill Gas & Electric	.	.	.
Pennsylvania Power & Light	.	.	.
Oklahoma Gas & Electric	0.00000	.	.
Wisconsin Energy	0.28070	0.00000	.
Green Mountain Power	0.64175	0.44814	0

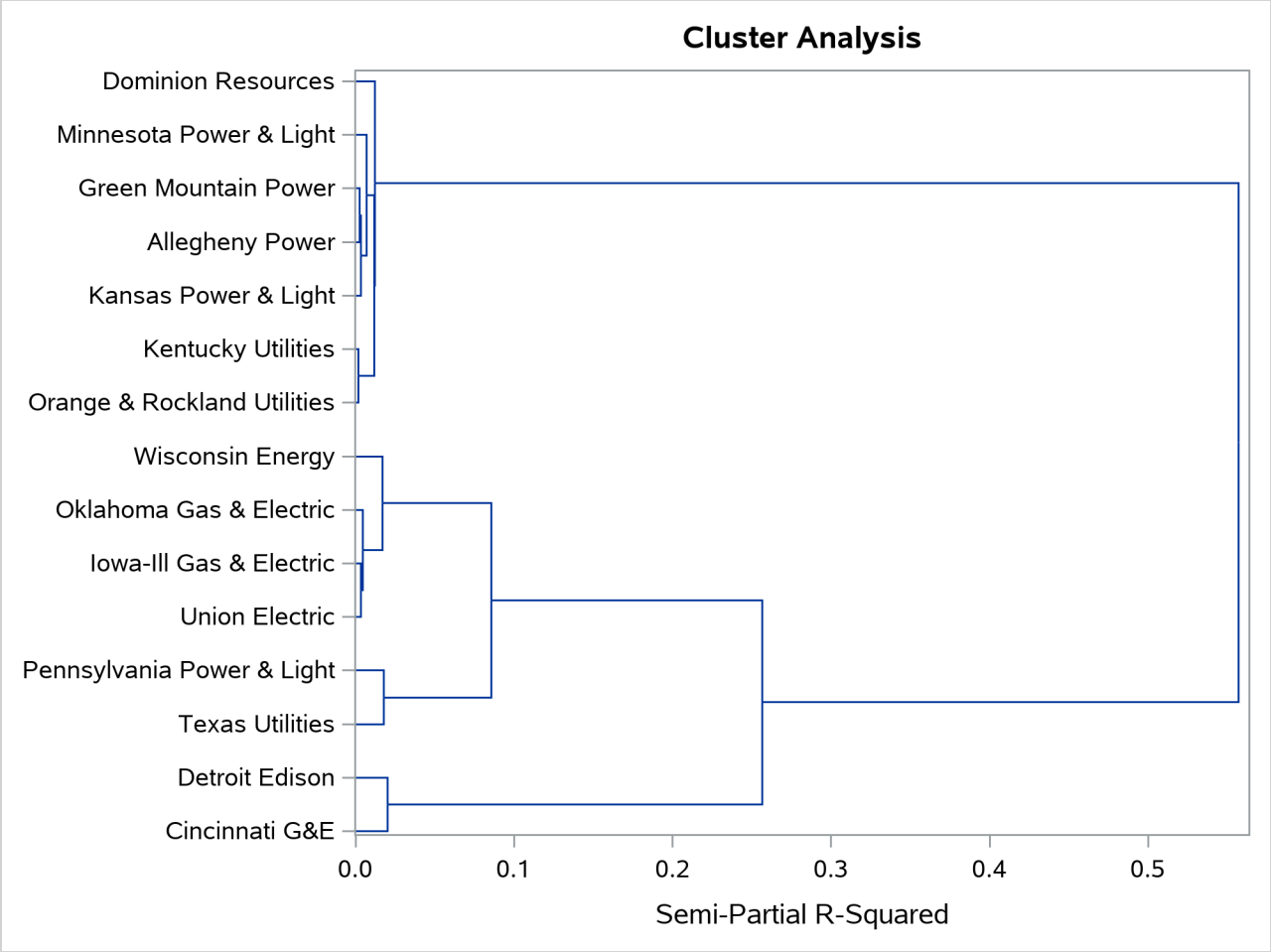
Output 40.2.2 Pseudo F versus Number of Clusters When METHOD=WARD



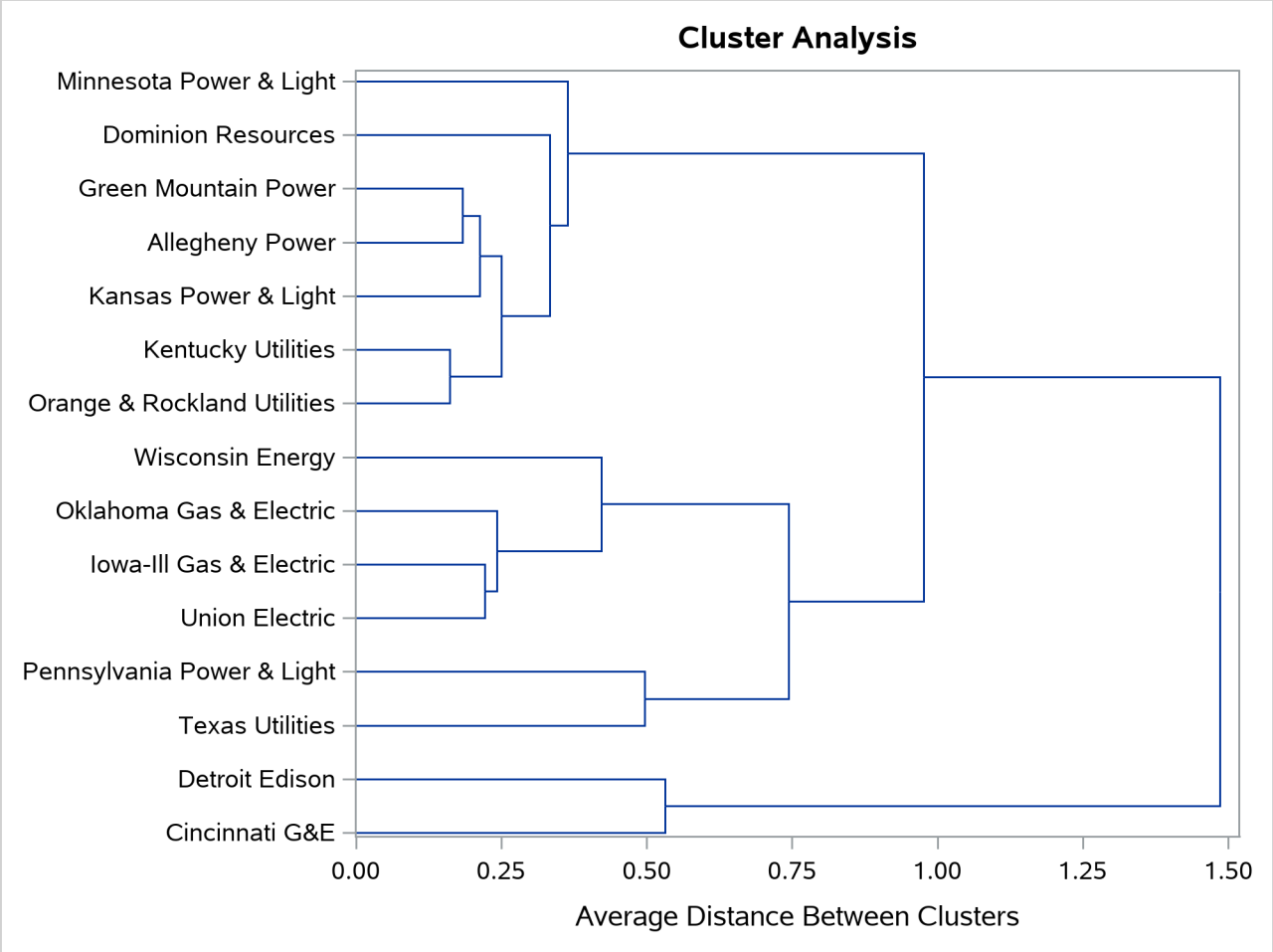
Output 40.2.3 Pseudo F versus Number of Clusters When METHOD=AVERAGE



Output 40.2.4 Dendrogram of Semipartial R-Square Values When METHOD=WARD



Output 40.2.5 Dendrogram of Average Distance between Clusters When METHOD=AVERAGE



References

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

Gower, J. C., and Legendre, P. (1986). "Metric and Euclidean Properties of Dissimilarity Coefficients." *Journal of Classification* 3:5–48.

Hand, D. J., Daly, F., Lunn, A. D., McConway, K. J., and Ostrowski, E. (1994). *A Handbook of Small Data Sets*. London: Chapman & Hall.

Sneath, P. H. A., and Sokal, R. R. (1973). *Numerical Taxonomy*. San Francisco: W. H. Freeman.

Subject Index

absolute level of measurement, definition
    DISTANCE procedure, 2673
asymmetric binary variable
    DISTANCE procedure, 2673

Binary Lance and Williams nonmetric coefficient
    DISTANCE procedure, 2699
Bray and Curtis coefficient
    DISTANCE procedure, 2699

Canberra metric coefficient
    DISTANCE procedure, 2695
Chebychev distance coefficient
    DISTANCE procedure, 2695
chi-square coefficient
    DISTANCE procedure, 2695, 2696
City-block distance coefficient
    DISTANCE procedure, 2695
clustering and computing distance matrix
    Correlation coefficients, example, 2707
    Jaccard coefficients, example, 2701
clustering and scaling
    DISTANCE procedure, example, 2676
correlation dissimilarity coefficient
    DISTANCE procedure, 2694
correlation similarity coefficient
    DISTANCE procedure, 2694
cosine coefficient
    DISTANCE procedure, 2695
covariance similarity coefficient
    DISTANCE procedure, 2694
Czekanowski/Sorensen similarity coefficient
    DISTANCE procedure, 2699

Dice coefficient
    DISTANCE procedure, 2699
distance measures available in DISTANCE procedure, *see* proximity measures
DISTANCE procedure
    absent-absent match, asymmetric binary variable, 2673
    absent-absent match, example, 2701
    absolute level of measurement, definition, 2673
    affine transformation, 2672
    asymmetric binary variable, 2673
    available levels of measurement, 2687
    available options for the option list, 2687
    Binary Lance and Williams nonmetric coefficient, 2699
    Bray and Curtis coefficient, 2699
    Canberra metric coefficient, 2695
    Chebychev distance coefficient, 2695
    chi-square coefficient, 2695, 2696
    City-block distance coefficient, 2695
    computing distances with weights, 2690
    correlation dissimilarity coefficient, 2694
    correlation similarity coefficient, 2694
    cosine coefficient, 2695
    covariance similarity coefficient, 2694
    Czekanowski/Sorensen similarity coefficient, 2699
    Dice coefficient, 2699
    dot Product coefficient, 2695
    Euclidean distance coefficient, 2694
    examples, 2674, 2701, 2707
    extension of binary variable, 2673
    formatted values, 2700
    formulas for proximity measures, 2693
    frequencies, 2691
    functional summary, 2680
    fuzz factor, 2681
    generalized Euclidean distance coefficient, 2695
    Gower's dissimilarity coefficient, 2694
    Gower's similarity coefficient, 2693
    Hamann coefficient, 2697
    Hamming distance coefficient, 2697
    identity transformation, 2673
    initial estimates for A-estimates, 2681
    interval level of measurement, 2672
    Jaccard dissimilarity coefficient, 2699
    Jaccard similarity coefficient, 2698
    Kulcynski 1 coefficient, 2699
    Lance-Williams nonmetric coefficient, 2695
    levels of measurement, 2672
    linear transformation, 2673
    log-interval level of measurement, 2673
    many-to-one transformation, 2672
    Minkowski L_p distance coefficient, 2694
    missing values, 2684–2686, 2689, 2699
    monotone increasing transformation, 2672
    nominal level of measurement, 2672
    nominal variable, 2673
    normalization, 2684, 2686
    one-to-one transformation, 2672
    ordinal level of measurement, 2672
    output data sets, 2684, 2685, 2700
    Overlap dissimilarity coefficient, 2695

Overlap similarity coefficient, 2695
phi-square coefficient, 2696
power distance coefficient, 2695
power transformation, 2673
ratio level of measurement, 2673
Roger and Tanimoto coefficient, 2697
Russell and Rao similarity coefficient, 2699
scaling variables, 2674
shape distance coefficient, 2694
similarity Ratio coefficient, 2695
similarity ratio coefficient, 2695
simple Matching coefficient, 2697
simple matching dissimilarity coefficient, 2697
size distance coefficient, 2694
Sokal and Sneath 1 coefficient, 2697
Sokal and Sneath 3 coefficient, 2697
squared correlation dissimilarity coefficient, 2694
squared correlation similarity coefficient, 2694
squared Euclidean distance coefficient, 2694
squared simple matching dissimilarity coefficient, 2697
standardization methods, 2688
standardization suppression, 2684
standardization with frequencies, 2691
standardization with weights, 2691
standardization, default methods, 2674, 2688
standardization, example, 2676
standardization, mandatory, 2674
strictly increasing transformation, 2672
summary of options, 2680
symmetric binary variable, 2673
transforming ordinal variables to interval, 2673, 2685
weights, 2690, 2691
dot Product coefficient
DISTANCE procedure, 2695

Euclidean distance coefficient
DISTANCE procedure, 2694

formatted values
DISTANCE procedure, 2700

generalized Euclidean distance coefficient
DISTANCE procedure, 2695
Gower's dissimilarity coefficient
DISTANCE procedure, 2694
Gower's similarity coefficient
DISTANCE procedure, 2693

Hamann coefficient
DISTANCE procedure, 2697
Hamming distance coefficient
DISTANCE procedure, 2697

interval level of measurement
DISTANCE procedure, 2672

Jaccard dissimilarity coefficient
DISTANCE procedure, 2699
Jaccard similarity coefficient
DISTANCE procedure, 2698

Kulcynski 1 coefficient
DISTANCE procedure, 2699

Lance-Williams nonmetric coefficient
DISTANCE procedure, 2695
levels of measurement
DISTANCE procedure, 2672
log-interval level of measurement
DISTANCE procedure, 2673

Minkowski L_p distance coefficient
DISTANCE procedure, 2694
missing values
DISTANCE procedure, 2684–2686, 2689, 2699

nominal level of measurement
DISTANCE procedure, 2672
nominal variable
DISTANCE procedure, 2673

ordinal level of measurement
DISTANCE procedure, 2672
ordinal variables
transformed to interval (RANKSCORE=), 2685
Overlap dissimilarity coefficient
DISTANCE procedure, 2695
Overlap similarity coefficient
DISTANCE procedure, 2695

phi-square coefficient
DISTANCE procedure, 2696
power distance coefficient
DISTANCE procedure, 2695
proximity measures
available methods for computing (DISTANCE), 2681
formulas(DISTANCE), 2693

ratio level of measurement
DISTANCE procedure, 2673
Roger and Tanimoto coefficient
DISTANCE procedure, 2697
Russell and Rao similarity coefficient
DISTANCE procedure, 2699

scaling variables
DISTANCE procedure, 2674
shape distance coefficient

DISTANCE procedure, 2694
similarity Ratio coefficient
DISTANCE procedure, 2695
similarity ratio coefficient
DISTANCE procedure, 2695
simple Matching coefficient
DISTANCE procedure, 2697
simple matching dissimilarity coefficient
DISTANCE procedure, 2697
size distance coefficient
DISTANCE procedure, 2694
Sokal and Sneath 1 coefficient
DISTANCE procedure, 2697
Sokal and Sneath 3 coefficient
DISTANCE procedure, 2697
squared correlation dissimilarity coefficient
DISTANCE procedure, 2694
squared correlation similarity coefficient
DISTANCE procedure, 2694
squared Euclidean distance coefficient
DISTANCE procedure, 2694
squared simple matching dissimilarity coefficient
DISTANCE procedure, 2697
standardization
comparisons between DISTANCE and STDIZE procedures, 2674
standardization suppression
DISTANCE procedure, 2684
STD= option (DISTANCE), 2688
symmetric binary variable
DISTANCE procedure, 2673

transformations
affine(DISTANCE), 2672
identity(DISTANCE), 2673
linear(DISTANCE), 2673
many-to-one(DISTANCE), 2672
monotone increasing(DISTANCE), 2672
one-to-one(DISTANCE), 2672
power(DISTANCE), 2673
strictly increasing(DISTANCE), 2672
transforming ordinal variables to interval
DISTANCE procedure, 2673

Syntax Index

ABSENT= option
    PROC DISTANCE statement, 2681
    VAR statement, 2689
ADD= option
    PROC DISTANCE statement, 2681

BY statement
    DISTANCE procedure, 2691

COPY statement
    DISTANCE procedure, 2691

DATA= option
    PROC DISTANCE statement, 2681
DISTANCE procedure, BY statement, 2691
DISTANCE procedure, COPY statement, 2691
DISTANCE procedure, FREQ statement, 2691
DISTANCE procedure, ID statement, 2690
DISTANCE procedure, PROC DISTANCE statement, 2680
    ABSENT= option, 2681
    ADD= option, 2681
    DATA= option, 2681
    FUZZ= option, 2681
    INITIAL= option, 2681
    METHOD= option, 2681
    MULT= option, 2684
    NOMISS, 2684
    NORM option, 2684
    NOSTD, 2684
    OUT= option, 2684
    OUTSDZ= option, 2685
    PREFIX= option, 2685
    RANKSCORE= option, 2685
    REPLACE, 2685
    REPONLY, 2686
    SHAPE= option, 2686
    SNORM option, 2686
    STDONLY option, 2686
    UNDEF= option, 2686
    VARDEF= option, 2686
DISTANCE procedure, VAR statement
    ABSENT= option, 2689
    MISSING= option, 2689
    ORDER= option, 2690
    WEIGHTS= option, 2690
DISTANCE procedure, WGT statement, 2691

FREQ statement
    DISTANCE procedure, 2691
FUZZ= option
    PROC DISTANCE statement, 2681

ID statement
    DISTANCE procedure, 2690
INITIAL= option
    PROC DISTANCE statement, 2681

METHOD= option
    PROC DISTANCE statement, 2681
MISSING= option
    VAR statement, 2689
MULT= option
    PROC DISTANCE statement, 2684

NOMISS option
    PROC DISTANCE statement, 2684
NORM option
    PROC DISTANCE statement, 2684
NOSTD option
    PROC DISTANCE statement, 2684

ORDER= option
    VAR statement, 2690
OUT= option
    PROC DISTANCE statement, 2684
OUTSDZ= option
    PROC DISTANCE statement, 2685

PREFIX= option
    PROC DISTANCE statement, 2685
PROC DISTANCE statement, *see* DISTANCE procedure

RANKSCORE= option
    PROC DISTANCE statement, 2685
REPLACE option
    PROC DISTANCE statement, 2685
REPONLY option
    PROC DISTANCE statement, 2686

SHAPE= option
    PROC DISTANCE statement, 2686
SNORM option
    PROC DISTANCE statement, 2686
STDONLY option
    PROC DISTANCE statement, 2686

UNDEF= option

PROC DISTANCE statement, 2686

VARDEF= option
PROC DISTANCE statement, 2686

WEIGHTS= option
VAR statement, 2690
WGT statement
DISTANCE procedure, 2691