SPSS Categories® 11.0

Jacqueline J. Meulman
Willem J. Heiser
SPSS Inc.

For more information about SPSS® software products, please visit our Web site at *http://www.spss.com* or contact

SPSS Inc.
233 South Wacker Drive, 11th Floor
Chicago, IL 60606-6412
Tel: (312) 651-3000
Fax: (312) 651-3668

SPSS Categories® 11.0

1 2 3 4 5 6 7 8 9 0 05 04 03 02 01

ISBN 1-56827-276-6

Preface

SPSS 11.0 is a powerful software package for microcomputer data management and analysis. The Categories option is an add-on enhancement that provides a comprehensive set of procedures for optimal scaling. The procedures in Categories must be used with the SPSS 11.0 Base and are completely integrated into that system.

The Categories option includes procedures for:

- Categorical regression
- Categorical principal components analysis
- Nonlinear canonical correlation analysis
- Correspondence analysis
- Homogeneity analysis
- Multidimensional scaling

Installation

To install Categories, follow the instructions for adding and removing features in the installation instructions supplied with the SPSS Base. (To start, double-click on the SPSS Setup icon.)

Compatibility

The SPSS system is designed to operate on many computer systems. See the materials that came with your system for specific information on minimum and recommended requirements.

Serial Numbers

Your serial number is your identification number with SPSS Inc. You will need this serial number when you call SPSS Inc. for information regarding support, payment, or an upgraded system. The serial number was provided with your Base system. Before using the system, please copy this number to the registration card.

Registration Card

Don't put it off: *fill out and send us your registration card*. Until we receive your registration card, you have an unregistered system. Even if you have previously sent a card to us, please fill out and return the card enclosed in your Categories package. Registering your system entitles you to:

- Technical support services
- New product announcements and upgrade announcements

Customer Service

If you have any questions concerning your shipment or account, contact your local office, listed on page vi. Please have your serial number ready for identification when calling.

Training Seminars

SPSS Inc. provides both public and onsite training seminars for SPSS. All seminars feature hands-on workshops. SPSS seminars will be offered in major U.S. and European cities on a regular basis. For more information on these seminars, call your local office, listed on page vi.

Technical Support

The services of SPSS Technical Support are available to registered customers. Customers may call Technical Support for assistance in using SPSS products or for installation help for one of the supported hardware environments. To reach Technical Support, see the SPSS Web site at *http://www.spss.com*, or call your local office, listed on page vi. Be prepared to identify yourself, your organization, and the serial number of your system.

Additional Publications

Except for academic course adoptions, additional copies of SPSS product manuals can be purchased directly from SPSS Inc. Visit our Web site at *http://www.spss.com*, or contact your local SPSS office, listed on page vi.

SPSS product manuals may also be purchased from Prentice Hall, the exclusive distributor of SPSS publications. To order, fill out and mail the Publications order form included with your system, or call 800-947-7700. If you represent a bookstore or have an account with Prentice Hall, call 800-382-3419. In Canada, call 800-567-3800. Outside of North America, contact your local Prentice Hall office.

Tell Us Your Thoughts

Your comments are important. Please let us know about your experiences with SPSS products. We especially like to hear about new and interesting applications using the SPSS system. Please send e-mail to *suggest@spss.com*, or write to SPSS Inc., Attn: Director of Product Planning, 233 South Wacker Drive, 11th Floor, Chicago, IL 60606-6412.

About This Manual

This manual is divided into two sections. The first section documents the graphical user interface and provides examples of the statistical techniques available. In addition, this section offers advice on interpreting the output. The second part of the manual is a Syntax Reference section that provides complete command syntax for all of the commands included in the Categories option. Most features of the system can be accessed through the dialog box interface, but some functionality can be accessed only through command syntax.

This manual contains two indexes: a subject index and a syntax index. The subject index covers both sections of the manual. The syntax index applies only to the Syntax Reference section.

Acknowledgments

The optimal scaling procedures and their SPSS implementation were developed by the Data Theory Scaling System Group (DTSS), consisting of members of the departments of Education and Psychology of the Faculty of Social and Behavioral Sciences at Leiden University.

Willem Heiser, Jacqueline Meulman, Gerda van den Berg, and Patrick Groenen were involved with the original 1990 procedures. Jacqueline Meulman and Peter Neufeglise participated in the development of procedures for categorical regression, correspondence analysis, categorical principal components analysis, and multidimensional scaling. In addition, Anita van der Kooij contributed especially to CATREG, CORRESPONDENCE, and CATPCA, and Frank Busing and Willem Heiser, to the PROXSCAL procedure. The development of PROXSCAL has profited from technical comments and suggestions from Jacques Commandeur and Patrick Groenen.

Contacting SPSS

If you would like to be on our mailing list, contact one of our offices, listed on page vi, or visit our Web site at *http://www.spss.com*. We will send you a copy of our newsletter and let you know about SPSS Inc. activities in your area.

SPSS Inc.
Chicago, Illinois, U.S.A.
Tel: 1.312.651.3000
or 1.800.543.2185
www.spss.com/corpinfo
Customer Service:
1.800.521.1337
Sales:
1.800.543.2185
sales@spss.com
Training:
1.800.543.6607
Technical Support:
1.312.651.3410
support@spss.com

SPSS Federal Systems
Tel: 1.703.740.2400
or 1.800.860.5762
www.spss.com

SPSS Argentina srl
Tel: +5411.4814.5030
www.spss.com

SPSS Asia Pacific Pte. Ltd.
Tel: +65.245.9110
www.spss.com

SPSS Australasia Pty. Ltd.
Tel: +61.2.9954.5660
www.spss.com

SPSS Belgium
Tel: +32.163.170.70
www.spss.com

SPSS Benelux BV
Tel: +31.183.651777
www.spss.com

SPSS Brasil Ltda
Tel: +55.11.5505.3644
www.spss.com

SPSS Czech Republic
Tel: +420.2.24813839
www.spss.cz

SPSS Denmark
Tel: +45.45.46.02.00
www.spss.com

SPSS East Africa
Tel: +254 2 577 262
spss.com

SPSS Finland Oy
Tel: +358.9.4355.920
www.spss.com

SPSS France SARL
Tel: +01.55.35.27.00
www.spss.com

SPSS Germany
Tel: +49.89.4890740
www.spss.com

SPSS BI Greece
Tel: +30.1.6971950
www.spss.com

SPSS Iberica
Tel: +34.902.123.606
SPSS.com

SPSS Hong Kong Ltd.
Tel: +852.2.811.9662
www.spss.com

SPSS Ireland
Tel: +353.1.415.0234
www.spss.com

SPSS BI Israel
Tel: +972.3.6166616
www.spss.com

SPSS Italia srl
Tel: +800.437300
www.spss.it

SPSS Japan Inc.
Tel: +81.3.5466.5511
www.spss.co.jp

SPSS Korea DataSolution Co.
Tel: +82.2.563.0014
www.spss.co.kr

SPSS Latin America
Tel: +1.312.651.3539
www.spss.com

SPSS Malaysia Sdn Bhd
Tel: +603.6203.2300
www.spss.com

SPSS Miami
Tel: 1.305.627.5700
SPSS.com

SPSS Mexico SA de CV
Tel: +52.5.682.87.68
www.spss.com

SPSS Norway AS
Tel: +47.22.99.25.50
www.spss.com

SPSS Polska
Tel: +48.12.6369680
www.spss.pl

SPSS Russia
Tel: +7.095.125.0069
www.spss.com

SPSS San Bruno
Tel: 1.650.794.2692
www.spss.com

SPSS Schweiz AG
Tel: +41.1.266.90.30
www.spss.com

SPSS BI (Singapore) Pte. Ltd.
Tel: +65.346.2061
www.spss.com

SPSS South Africa
Tel: +27.21.7120929
www.spss.com

SPSS South Asia
Tel: +91.80.2088069
www.spss.com

SPSS Sweden AB
Tel: +46.8.506.105.50
www.spss.com

SPSS Taiwan Corp.
Taipei, Republic of China
Tel: +886.2.25771100
www.sinter.com.tw/spss/main

SPSS (Thailand) Co., Ltd.
Tel: +66.2.260.7070
www.spss.com

SPSS UK Ltd.
Tel: +44.1483.719200
www.spss.com

Contents

1 **Introduction to SPSS Optimal Scaling Procedures for Categorical Data 1**

What Is Optimal Scaling? 1

Why Use Optimal Scaling? 1

Optimal Scaling Level and Measurement Level 2
- Selecting the Optimal Scaling Level 3
- Transformation Plots 4
- Category Codes 5

Which Procedure Is Best for Your Application? 7
- Categorical Regression 8
- Categorical Principal Components Analysis 9
- Nonlinear Canonical Correlation Analysis 9
- Correspondence Analysis 10
- Homogeneity Analysis 12
- Multidimensional Scaling 13

Displays with More than Two Dimensions 14
- Three-Dimensional Scatterplots 14
- Scatterplot Matrices 15

Aspect Ratio in Optimal Scaling Charts 16

2 **Categorical Regression (CATREG) 17**

To Obtain a Categorical Regression 18
- Define Scale in Categorical Regression 19
- To Define the Scale in CATREG 20
- Categorical Regression Discretization 20
- Categorical Regression Missing Values 22
- Categorical Regression Options 23
- Categorical Regression Output 24
- Categorical Regression Save 25
- Categorical Regression Plots 26
- CATREG Command Additional Features 26

3 **Categorical Principal Components Analysis (CATPCA) 27**

To Obtain a Categorical Principal Components Analysis 28
Define Scale and Weight in CATPCA 29
To Define the Scale and Weight in CATPCA 31
Categorical Principal Components Discretization 32
Categorical Principal Components Missing Values 33
Categorical Principal Components Category Plots 34
Categorical Principal Components Object and Variable Plots 35
Categorical Principal Components Loading Plots 36
Categorical Principal Components Output 37
Categorical Principal Components Save 38
Categorical Principal Components Options 39
CATPCA Command Additional Features 40

4 **Nonlinear Canonical Correlation Analysis (OVERALS) 41**

To Obtain a Nonlinear Canonical Correlation Analysis 42
Define Range and Scale in OVERALS 44
To Define an Optimal Scaling Range and Scale in OVERALS 45
Define Range in OVERALS 45
To Define an Optimal Scaling Range in OVERALS 46
Nonlinear Canonical Correlation Analysis Options 46
OVERALS Command Additional Features 48

5 **Correspondence Analysis 49**

To Obtain a Correspondence Analysis 50
Define Row Range in Correspondence Analysis 51
To Define a Row Range in Correspondence Analysis 51
Define Column Range in Correspondence Analysis 52
To Define a Column Range in Correspondence Analysis 53
Correspondence Analysis Model 54
Correspondence Analysis Statistics 56
Correspondence Analysis Plots 57
CORRESPONDENCE Command Additional Features 58

6 **Homogeneity Analysis (HOMALS) 59**

To Obtain a Homogeneity Analysis 60
Define Range in Homogeneity Analysis 62
To Define an Optimal Scaling Range in Homogeneity Analysis 62
Homogeneity Analysis Options 63
HOMALS Command Additional Features 64

7 **Multidimensional Scaling (PROXSCAL) 65**

To Obtain a Multidimensional Scaling 66
Proximities in Matrices across Columns 67
Proximities in Columns 68
Proximities in One Column 69
Create Proximities from Data 70
Measures Dialog Box 71
Define a Multidimensional Scaling Model 72
Multidimensional Scaling Restrictions 74
Multidimensional Scaling Options 75
Multidimensional Scaling Plots, Version 1 76
Multidimensional Scaling Plots, Version 2 78
Multidimensional Scaling Output 78
PROXSCAL Command Additional Features 80

8 **Categorical Regression Examples 81**

Example 1: Carpet Cleaner Data 81
A Standard Linear Regression Analysis 82
A Categorical Regression Analysis 84
Example 2: Ozone Data 93
Categorizing Variables 94
Selection of Transformation Type 95
Optimality of the Quantifications 101
Effects of Transformations 102

9 **Categorical Principal Components Analysis Examples 107**

Example 1: Interrelations of Social Systems 108
- Number of Dimensions 110
- Quantifications 111
- Object Scores 112
- Component Loadings 114
- Additional Dimensions 115

Example 2: Symptomatology of Eating Disorders 118
- Transformation Plots 121
- Model Summary 122
- Component Loadings 123
- Object Scores 124
- Examining the Structure of the Course of Illness 126

10 **Nonlinear Canonical Correlation Analysis Examples 131**

Example: An Analysis of Survey Results 131
- Examining the Data 132
- Accounting for Similarity between Sets 134
- Component Loadings 138
- Transformation Plots 138
- Single versus Multiple Category Coordinates 140
- Centroids and Projected Centroids 142

An Alternative Analysis 145

General Suggestions 149

11 **Correspondence Analysis Examples 151**

Example 1: Smoking Behavior by Job Category 152
- Profiles and Distances 156
- Inertia 157
- Row and Column Scores 158
- Dimensionality 159
- Supplementary Profiles 160
- Contributions 162
- Permutations of the Correspondence Table 165
- Confidence Statistics 166
- Normalization 168

Example 2: Perceptions of Coffee Brands 171
Principal Normalization 172
Dimensionality 172
Contributions 173
Plots 175
Symmetrical Normalization 177
Example 3: Flying Mileage between Cities 178
Row and Column Scores 181

12 Homogeneity Analysis Examples 183

Example: Characteristics of Hardware 184
Multiple Dimensions 185
Object Scores 186
Discrimination Measures 188
Category Quantifications 189
A More Detailed Look at Object Scores 191
Omission of Outliers 194

13 Multidimensional Scaling Examples 197

Example: An Examination of Kinship Terms 197
Choosing the Number of Dimensions 198
A Three-Dimensional Solution 199
A Three-Dimensional Solution with Nondefault Transformations 205
Discussion 208

Syntax Reference 209

Introduction 211
ANACOR 215
CATPCA 225
CATREG 243
CORRESPONDENCE 255
HOMALS 267
OVERALS 275

PRINCALS 285
PROXSCAL 295

Bibliography 313

Subject Index 317

Syntax Index 323

1 Introduction to SPSS Optimal Scaling Procedures for Categorical Data

SPSS Categories procedures use optimal scaling to analyze data that are difficult or impossible for "standard" statistical procedures to analyze.[1] This chapter describes what each procedure does, the situations in which each procedure is most appropriate, the relationships between the procedures, and the relationships of these procedures to their "standard" statistical counterparts.

What Is Optimal Scaling?

The idea behind optimal scaling is to assign numerical quantifications to the categories of each variable, thus allowing "standard" procedures to be used to obtain a solution on the quantified variables.

The optimal scale values are assigned to categories of each variable based on the optimizing criterion of the procedure in use. Unlike the original labels of the nominal or ordinal variables in the analysis, these scale values have metric properties.

In most Categories procedures, the optimal quantification for each scaled variable is obtained through an iterative method called **alternating least squares** in which, after the current quantifications are used to find a solution, the quantifications are updated using that solution. The updated quantifications are then used to find a new solution, which is used to update the quantifications, and so on until some criterion is reached that signals the process to stop.

Why Use Optimal Scaling?

Categorical data are often found in marketing research, survey research, and research in the social and behavioral sciences. In fact, many researchers deal almost exclusively with categorical data.

1. These procedures and their SPSS implementation were developed by the Data Theory Scaling System Group (DTSS), consisting of members of the departments of Education and Psychology, Faculty of Social and Behavioral Sciences, Leiden University.

While adaptations of most standard models exist specifically to analyze categorical data, they often do not perform well for data sets that feature:

- Too few observations
- Too many variables
- Too many values per variable

By quantifying categories, optimal scaling techniques avoid problems in these situations. Moreover, they are useful even when specialized techniques are appropriate.

Rather than interpreting parameter estimates, the interpretation of optimal scaling output is often based on graphical displays. Optimal scaling techniques offer excellent exploratory analyses, which complement other SPSS models well. By narrowing the focus of your investigation, visualizing your data through optimal scaling can form the basis of an analysis that centers on interpretation of model parameters.

Optimal Scaling Level and Measurement Level

This can be a very confusing concept when you first use Categories procedures. When specifying the level, you specify not the level at which variables are *measured,* but the level at which they are *scaled.* The idea is that the variables to be quantified may have nonlinear relations regardless of how they are measured.

For Categories purposes, there are three basic levels of measurement:

- The **nominal** level implies that a variable's values represent unordered categories. Examples of variables that might be nominal are region, zip code area, religious affiliation, and multiple choice categories.
- The **ordinal** level implies that a variable's values represent ordered categories. Examples include attitude scales representing degree of satisfaction or confidence and preference rating scores.
- The **numerical** level implies that a variable's values represent ordered categories with a meaningful metric, so that distance comparisons between categories are appropriate. Examples include age in years and income in thousands of dollars.

For example, suppose the variables *region*, *job*, and *age* are coded as shown in Table 1.1.

Table 1.1 Coding scheme for region, job, and age

Region		Job		Age	
1	North	1	intern	20	twenty years old
2	South	2	sales rep	22	twenty-two years old
3	East	3	manager	25	twenty-five years old
4	West			27	twenty-seven years old

The values shown represent the categories of each variable. *Region* would be a nominal variable. There are four categories of *region*, with no intrinsic ordering. Values 1 through 4 simply represent the four categories; the coding scheme is completely arbitrary. *Job*, on the other hand, could be assumed to be an ordinal variable. The original categories form a progession from intern to manager. Larger codes represent a job higher on the corporate ladder. However, only the order information is known—nothing can be said about the distance between adjacent categories. In contrast, *age* could be assumed to be a numerical variable. In the case of *age*, the distances between the values are intrinsically meaningful. The distance between 20 and 22 is the same as the distance between 25 and 27, while the distance between 22 and 25 is greater than either of these.

Selecting the Optimal Scaling Level

It is important to understand that there are no intrinsic properties of a variable that automatically predefine what optimal scaling level you should specify for it. You can explore your data in any way that makes sense and makes interpretation easier. By analyzing a numerical-level variable at the ordinal level, for example, the use of a nonlinear transformation may allow a solution in fewer dimensions.

The following two examples illustrate how the "obvious" level of measurement might not be the best optimal scaling level. Suppose that a variable sorts objects into age groups. Although age can be scaled as a numerical variable, it may be true that for people younger than 25 safety has a positive relation with age, whereas for people older than 60 safety has a negative relation with age. In this case, it might be better to treat age as a nominal variable.

As another example, a variable that sorts persons by political preference appears to be essentially nominal. However, if you order the parties from political left to political right, you might want the quantification of parties to respect this order by using an ordinal level of analysis.

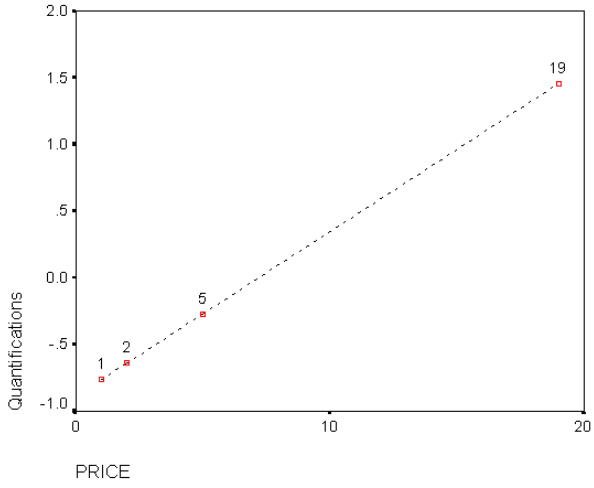
Even though there are no predefined properties of a variable that make it exclusively one level or another, there are some general guidelines to help the novice user. With single-nominal quantification, you don't usually know the order of the categories but you want the analysis to impose one. If the order of the categories is known, you should try ordinal quantification. If the categories are unorderable, you might try multiple-nominal quantification.

Transformation Plots

The different levels at which each variable can be scaled impose different restrictions on the quantifications. Transformation plots illustrate the relationship between the quantifications and the original categories resulting from the selected optimal scaling level. For example, a linear transformation plot results when a variable is treated as numerical. Variables treated as ordinal result in a nondecreasing transformation plot. Transformation plots for variables treated nominally that are U-shaped (or the reverse) display a quadratic relationship. Nominal variables could also yield transformation plots without apparent trends by changing the order of the categories completely. Figure 1.1 displays a sample transformation plot.

Transformation plots are particularly suited to determining how well the selected optimal scaling level performs. If several categories receive similar quantifications, collapsing these categories into one category may be warranted. Alternatively, if a variable treated as nominal receives quantifications that display an increasing trend, an ordinal transformation may result in a similar fit. If that trend is linear, numerical treatment may be appropriate. However, if collapsing categories or changing scaling levels is warranted, the analysis will not change significantly.

Figure 1.1 Transformation plot of price (numerical)



Although HOMALS does not currently offer the transformation plot in Figure 1.1 as an option, creating this plot is a straightforward procedure. For each variable, create a new variable containing the values (and labels) of the categories. Create a new variable containing the quantifications for that variable from the Categories analysis. Use the line facility with the data in the chart representing the values of individual cases. You can use the quantifications for the line in the chart and use the categories to label the chart.

Category Codes

Some care should be taken when coding categorical variables, because some coding schemes may yield unwanted output or incomplete analyses. Possible coding schemes for *job* are displayed in Table 1.2.

Table 1.2 Alternative coding schemes for job

	Scheme			
Category	**A**	**B**	**C**	**D**
intern	1	1	5	1
sales rep	2	2	6	5
manager	3	7	7	3

Some Categories procedures require that the range of every variable used be defined. Any value outside this range is treated as a missing value. The minimum category value is always 1. The maximum category value is supplied by the user. This value is not the *number* of categories for a variable; it is the *largest* category value. For example, in Table 1.2, scheme A has a maximum category of 3 and scheme B has a maximum category value of 7, yet both schemes code the same three categories.

The variable range determines which categories will be omitted from the analysis. Any categories with codes outside the defined range are omitted from the analysis. This is a simple method for omitting categories but can result in unwanted analyses. An incorrectly defined maximum category can omit *valid* categories from the analysis. For example, for scheme B, defining the maximum category value to be 3 indicates that *job* has categories coded from 1 to 3; the *manager* category is treated as missing. Because no category has actually been coded *3*, the third category in the analysis contains no cases. If you wanted to omit all manager categories, this analysis would be appropriate. However, if managers are to be included, the maximum category must be defined as 7, and missing values must be coded with values above 7 or below 1.

For variables treated as nominal or ordinal, the range of the categories does not affect the results. For nominal variables, only the label and not the value associated with that label is important. For ordinal variables, the order of the categories is preserved in the quantifications; the category values themselves are not important. All coding schemes resulting in the same category ordering will have identical results. For example, the first three schemes in Table 1.2 are functionally equivalent if *job* is analyzed at an ordinal level. The order of the categories is identical in these schemes. Scheme D, on the other hand, inverts the second and third categories and will yield different results than the other schemes.

Although many coding schemes for a variable are functionally equivalent, schemes with small differences between codes are preferred because the codes have an impact on the amount of output produced by a procedure. All categories coded with values between 1 and the user-defined maximum are valid. If any of these categories are empty, the corresponding quantifications will be either system missing or zero, depending on the procedure. Although neither of these assignments affect the analyses, output is produced for these categories. Thus, for scheme B, *job* has four categories that receive system-missing values. For scheme C, there are also four categories receiving system-missing indicators. In contrast, for scheme A there are no system-missing quantifications. Using consecutive integers as codes for variables treated as nominal or ordinal results in much less output without affecting the results.

Coding schemes for variables treated as numerical are more restricted than the ordinal case. For these variables, the differences between consecutive categories are important. Table 1.3 displays three coding schemes for *age*.

Table 1.3 Alternative coding schemes for age

	Scheme		
Category	**A**	**B**	**C**
20	20	1	1
22	22	3	2
25	25	6	3
27	27	8	4

Any recoding of numerical variables must preserve the differences between the categories. Using the original values is one method for ensuring preservation of differences. However, this can result in many categories having system-missing indicators. For example, scheme A in Table 1.3 employs the original observed values. For all Categories procedures except for correspondence analysis, the maximum category value is 27 and the minimum category value is set to 1. The first 19 categories are empty and receive system-missing indicators. The output can quickly become rather cumbersome if the maximum category is much greater than 1 and there are many empty categories between 1 and the maximum.

To reduce the amount of output, recoding can be done. However, in the numerical case, the Automatic Recode facility should not be used. Coding to consecutive integers results in differences of 1 between all consecutive categories and as a result, all quantifications will be equally spaced. The metric characteristics deemed important when treating a variable as numerical are destroyed by recoding to consecutive integers. For example, scheme C in Table 1.3 corresponds to automatically recoding *age*. The difference between categories *22* and *25* has changed from three to one, and the quantifications will reflect the latter difference.

An alternative recoding scheme that preserves the differences between categories is to subtract the smallest category value from every category and add one to each difference.

Scheme B results from this transformation. The smallest category value, 20, has been subtracted from each category, and 1 was added to each result. The transformed codes have a minimum of 1, and all differences are identical to the original data. The maximum category value is now eight, and the zero quantifications before the first nonzero quantification are all eliminated. Yet, the nonzero quantifications corresponding to each category resulting from scheme B are identical to the quantifications from scheme A.

Which Procedure Is Best for Your Application?

The techniques embodied in four of these procedures (Correspondence Analysis, Homogeneity Analysis, Categorical Principal Components Analysis, and Nonlinear Canonical Correlation Analysis) fall into the general area of multivariate data analysis known as **dimension reduction**. That is, relationships between variables are represented in a few dimensions—say two or three—as often as possible. This enables you to describe structures or patterns in the relationships that would be too difficult to fathom in their original richness and complexity. In market research applications, these techniques can be a form of **perceptual mapping**. A major advantage of these procedures is that they accommodate data with different levels of optimal scaling.

Categorical Regression describes the relationship between a categorical response variable and a combination of categorical predictor variables. The influence of each predictor variable on the response variable is described by the corresponding regression weight. As in the other procedures, data can be analyzed with different levels of optimal scaling.

Multidimensional Scaling describes relationships between objects in as few dimensions as possible, starting either with a matrix of proximities between the objects or with the original data from which the proximities are computed.

Following are brief guidelines for each of the procedures:

- Use Categorical Regression to predict the values of a categorical dependent variable from a combination of categorical independent variables.
- Use Categorical Principal Components Analysis to account for patterns of variation in a single set of variables of mixed optimal scaling levels.
- Use Nonlinear Canonical Correlation Analysis to assess the extent to which two or more sets of variables of mixed optimal scaling levels are correlated.
- Use Correspondence Analysis to analyze two-way contingency tables or data that can be expressed as a two-way table, such as brand preference or sociometric choice data.
- Use Homogeneity Analysis to analyze a categorical multivariate data matrix when you are willing to make no stronger assumption that all variables are analyzed at the nominal level.
- Use Multidimensional Scaling to analyze proximity data to find a least-squares representation of the objects in a low-dimensional space.

Categorical Regression

The use of Categorical Regression is most appropriate when the goal of your analysis is to predict a dependent (response) variable from a set of independent (predictor) variables. As with all optimal scaling procedures, scale values are assigned to each category of every variable such that these values are optimal with respect to the regression. The solution of a categorical regression maximizes the squared correlation between the transformed response and the weighted combination of transformed predictors.

Relation to other Categories procedures. Categorical regression with optimal scaling is comparable to optimal scaling canonical correlation analysis with two sets, one of which contains only the dependent variable. In the latter technique, similarity of sets is derived by comparing each set to an unknown variable that lies somewhere between all of the sets. In categorical regression, similarity of the transformed response and the linear combination of transformed predictors is assessed directly.

Relation to standard techniques. In standard linear regression, categorical variables can either be recoded as indicator variables or can be treated in the same fashion as interval level variables. In the first approach, the model contains a separate intercept and slope for each combination of the levels of the categorical variables. This results in a large number of parameters to interpret. In the second approach, only one parameter is estimated for each variable. However, the arbitrary nature of the category codings makes generalizations impossible.

If some of the variables are not continuous, alternative analyses are available. If the response is continuous and the predictors are categorical, analysis of variance is often employed. If the response is categorical and the predictors are continuous, logistic regression or discriminant analysis may be appropriate. If the response and the predictors are both categorical, loglinear models are often used.

Regression with optimal scaling offers three scaling levels for each variable. Combinations of these levels can account for a wide range of nonlinear relationships for which any single “standard” method is ill-suited. Consequently, optimal scaling offers greater flexibility than the standard approaches with minimal added complexity.

In addition, nonlinear transformations of the predictors usually reduce the dependencies among the predictors. If you compare the eigenvalues of the correlation matrix for the predictors with the eigenvalues of the correlation matrix for the optimally scaled predictors, the latter set will usually be less variable than the former. In other words, in categorical regression, optimal scaling makes the larger eigenvalues of the predictor correlation matrix smaller and the smaller eigenvalues larger.

Categorical Principal Components Analysis

The use of Categorical Principal Components Analysis is most appropriate when you want to account for patterns of variation in a single set of variables of mixed optimal scaling levels. This technique attempts to reduce the dimensionality of a set of variables while accounting for as much of the variation as possible. Scale values are assigned to each category of every variable such that these values are optimal with respect to the principal components solution. Objects in the analysis receive component scores based on the quantified data. Plots of the component scores reveal patterns among the objects in the analysis and can reveal unusual objects in the data. The solution of a categorical principal components analysis maximizes the correlations of the object scores with each of the quantified variables, for the number of components (dimensions) specified.

An important application of categorical principal components is to examine preference data, in which respondents rank or rate a number of items with respect to preference. In the usual SPSS data configuration, rows are individuals, columns are measurements for the items, and the scores across rows are preference scores (on a 0 to 10 scale, for example), making the data row-conditional. For preference data, you may want to treat the individuals as variables. Using the TRANSPOSE procedure, you can transpose the data. The raters become the variables, and all variables are declared ordinal. There is no objection to using more variables than objects in CATPCA.

Relation to other Categories procedures. If all variables are declared multiple nominal, categorical principal components analysis produces an analysis equivalent to a homogeneity analysis run on the same variables. Thus, categorical principal components analysis can be seen as a type of homogeneity analysis in which some of the variables are declared ordinal or numerical.

Relation to standard techniques. If all variables are scaled on the numerical level, categorical principal components analysis is equivalent to standard principal components analysis.

More generally, categorical principal components analysis is an alternative to computing the correlations between non-numerical scales and analyzing them using a standard principal components or factor-analysis approach. Naive use of the usual Pearson correlation coefficient as a measure of association for ordinal data can lead to nontrivial bias in estimation of the correlations.

Nonlinear Canonical Correlation Analysis

Nonlinear canonical correlation analysis is a very general procedure with many different applications.

The goal of nonlinear canonical correlation analysis is to analyze the relationships between two or more sets of variables instead of between the variables themselves, as in principal components analysis. For example, you may have two sets of variables, where

one set of variables might be demographic background items on a set of respondents, while a second set of variables might be responses to a set of attitude items. The scaling levels in the analysis can be any mix of nominal, ordinal, and numerical. Optimal scaling canonical correlation analysis determines the similarity among the sets by simultaneously comparing the canonical variables from each set to a compromise set of scores assigned to the objects.

Relation to other Categories procedures. If there are two or more sets of variables with only one variable per set, optimal scaling canonical correlation analysis is equivalent to optimal scaling principal components analysis. If all variables in a one-variable-per-set analysis are multiple nominal, optimal scaling canonical correlation analysis is equivalent to homogeneity analysis. If there are two sets of variables, one of which contains only one variable, optimal scaling canonical correlation analysis is equivalent to categorical regression with optimal scaling.

Relation to standard techniques. Standard canonical correlation analysis is a statistical technique that finds a linear combination of one set of variables and a linear combination of a second set of variables that are maximally correlated. Given this set of linear combinations, canonical correlation analysis can find subsequent independent sets of linear combinations, referred to as canonical variables, up to a maximum number equal to the number of variables in the smaller set.

If there are two sets of variables in the analysis and all variables are defined to be numerical, optimal scaling canonical correlation analysis is equivalent to a standard canonical correlation analysis. Although SPSS does not have a canonical correlation analysis procedure, many of the relevant statistics can be obtained from multivariate analysis of variance.

Optimal scaling canonical correlation analysis has various other applications. If you have two sets of variables and one of the sets contains a nominal variable declared as single nominal, optimal scaling canonical correlation analysis results can be interpreted in a similar fashion to regression analysis. If you consider the variable to be multiple nominal, the optimal scaling analysis is an alternative to discriminant analysis. Grouping the variables in more than two sets provides a variety of ways to analyze your data.

Correspondence Analysis

The goal of correspondence analysis is to make biplots for correspondence tables. In a correspondence table, the row and column variables are assumed to represent unordered categories; therefore, the nominal optimal scaling level is always used. Both variables are inspected for their nominal information only. That is, the only consideration is the fact that some objects are in the same category, while others are not. Nothing is assumed about the distance or order between categories of the same variable.

One specific use of correspondence analysis is the analysis of two-way contingency tables. If a table has *r* active rows and *c* active columns, the number of dimensions in the correspondence analysis solution is the minimum of *r* minus 1 or *c* minus 1, whichever is less. In other words, you could perfectly represent the row categories or the column categories of a contingency table in a space of $min(r, c) - 1$ dimensions. Practically speaking, however, you would like to represent the row and column categories of a two-way table in a low-dimensional space, say two dimensions, for the reason that two-dimensional plots are more easily comprehensible than multidimensional spatial representations.

When fewer than the maximum number of possible dimensions is used, the statistics produced in the analysis describe how well the row and column categories are represented in the low-dimensional representation. Provided that the quality of representation of the two-dimensional solution is good, you can examine plots of the row points and the column points to learn which categories of the row variable are similar, which categories of the column variable are similar, and which row and column categories are similar to each other.

Relation to other Categories procedures. Simple correspondence analysis is limited to two-way tables. If there are more than two variables of interest, you can combine variables to create **interaction variables**. For example, for the variables in Table 1.1, you can combine *region* and *job* to create a new variable *rejob* with the 12 categories in Table 1.4. This new variable forms a two-way table with age (12 rows, 4 columns), which can be analyzed in correspondence analysis.

Table 1.4 Combinations of region and job

Category Code	Category Definition	Category Code	Category Definition
1	North, intern	7	East, intern
2	North, sales rep	8	East, sales rep
3	North, manager	9	East, manager
4	South, intern	10	West, intern
5	South, sales rep	11	West, sales rep
6	South, manager	12	West, manager

One shortcoming of this approach is that any pair of variables can be combined. We can combine job and age, yielding another 12 category variable. Or we can combine region and age, which results in a new 16 category variable. Each of these interaction variables forms a two-way table with the remaining variable. Correspondence analyses of these three tables will not yield identical results, yet each is a valid approach. Furthermore, if there are four or more variables, two-way tables comparing an interaction variable with another interaction variable can be constructed. The number of possible tables to analyze can get quite large, even for a few variables. You can select one of these tables to

analyze, or you can analyze all of them. Alternatively, the Homogeneity Analysis procedure can be used to examine all of the variables simultaneously without the need to construct interaction variables.

Relation to standard techniques. The SPSS Crosstabs procedure can also be used to analyze contingency tables, with independence as a common focus in the analyses. However, even in small tables, detecting the cause of departures from independence may be difficult. The utility of correspondence analysis lies in displaying such patterns for two-way tables of any size. If there is an association between the row and column variables—that is, if the chi-square value is significant—correspondence analysis may help reveal the nature of the relationship.

Homogeneity Analysis

Homogeneity analysis tries to produce a solution in which objects within the same category are plotted close together and objects in different categories are plotted far apart. Each object is as close as possible to the category points of categories that apply to the object. In this way, the categories divide the objects into homogeneous subgroups. Variables are considered homogeneous when they classify objects in the same categories into the same subgroups.

For a one-dimensional solution, homogeneity analysis assigns optimal scale values (category quantifications) to each category of each variable in such a way that overall, on average, the categories have maximum spread. For a two-dimensional solution, homogeneity analysis finds a second set of quantifications of the categories of each variable unrelated to the first set, attempting again to maximize spread, and so on. Because categories of a variable receive as many scorings as there are dimensions, the variables in the analysis are assumed to be multiple nominal in optimal scaling level.

Homogeneity analysis also assigns scores to the objects in the analysis in such a way that the category quantifications are the averages, or centroids, of the object scores of objects in that category.

Relation to other Categories procedures. Homogeneity analysis is also known as multiple correspondence analysis or dual scaling. It gives comparable, but not identical, results to correspondence analysis when there are only two variables. Correspondence analysis produces unique output summarizing the fit and quality of representation of the solution, including stability information. Thus, correspondence analysis is usually preferable to homogeneity analysis in the two-variable case. Another difference between the two procedures is that the input to homogeneity analysis is a data matrix, where the rows are objects and the columns are variables, while the input to correspondence analysis can be the same data matrix, a general proximity matrix, or a joint contingency table, which is an aggregated matrix where both the rows and columns represent categories of variables.

Homogeneity analysis can also be thought of as principal components analysis of data scaled at the multiple nominal level.

Relation to standard techniques. Homogeneity analysis can be thought of as the analysis of a multiway contingency table. Multiway contingency tables can also be analyzed with the SPSS Crosstabs procedure, but Crosstabs gives separate summary statistics for each category of each control variable. With homogeneity analysis, it is often possible to summarize the relationship between all the variables with a single two-dimensional plot.

An advanced use of homogeneity analysis is to replace the original category values with the optimal scale values from the first dimension and perform a secondary multivariate analysis. Since homogeneity analysis replaces category labels with numerical scale values, many different procedures that require numerical data can be applied after the homogeneity analysis. For example, the Factor Analysis procedure produces a first principal component that is equivalent to the first dimension of homogeneity analysis. The component scores in the first dimension are equal to the object scores, and the squared component loadings are equal to the discrimination measures. The second homogeneity analysis dimension, however, is not equal to the second dimension of factor analysis.

Multidimensional Scaling

The use of multidimensional scaling is most appropriate when the goal of your analysis is to find the structure in a set of distance measures between objects or cases. This is accomplished by assigning observations to specific locations in a conceptual low-dimensional space such that the distances between points in the space match the given (dis)similarities as closely as possible. The result is a least-squares representation of the objects in that low-dimensional space which, in many cases, will help you further understand your data.

Relation to other Categories procedures. When you have multivariate data from which you create distances and then analyze with multidimensional scaling, the results are similar to analyzing the data using categorical principal components analysis with object principal normalization. This kind of PCA is also known as principal coordinates analysis.

Relation to standard techniques. The Categories multidimensional scaling procedure (PROXSCAL) offers several improvements upon the scaling procedure available in the Base system (ALSCAL). PROXSCAL offers an accelerated algorithm for certain models and allows you to put restrictions on the common space. Moreover, PROXSCAL attempts to minimize normalized raw stress, rather than S-stress (also referred to as **strain**). The normalized raw stress is generally preferred because it is a measure based on the distances, while the S-stress is based on the squared distances.

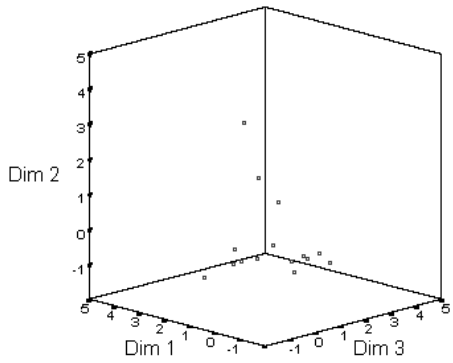
Displays with More than Two Dimensions

All of the optimal scaling procedures involving dimension reduction allow you to select the number of dimensions included in the analysis. For analyses with three or more dimensions, these procedures produce three-dimensional scatterplots or matrices of scatterplots.

Three-Dimensional Scatterplots

Figure 1.2 shows a three-dimensional plot of object scores produced by a homogeneity analysis with four dimensions.

Figure 1.2 Three-dimensional plot of object scores



Although only the first three dimensions are displayed on the scatterplot, information about all dimensions is included when the chart is created. You can choose to display different combinations of dimensions on the scatterplot by selecting *Displayed* from the Series menu in the Chart Editor. Figure 1.3 shows the 3-D Scatterplot Displayed Data dialog box, with dimension 4 selected to be displayed in place of dimension 3. The plot displaying these selections is shown in Figure 1.4.

Figure 1.3 3-D Scatterplot Displayed Data dialog box

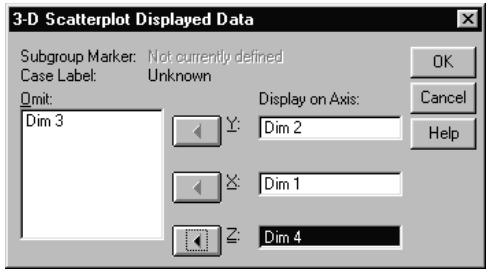
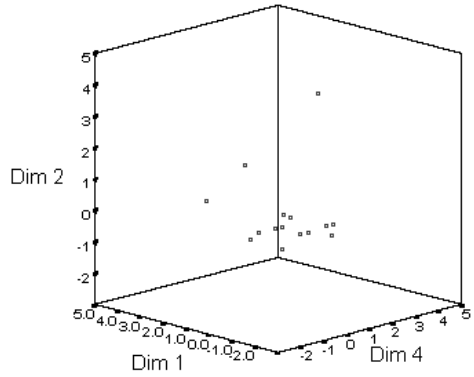


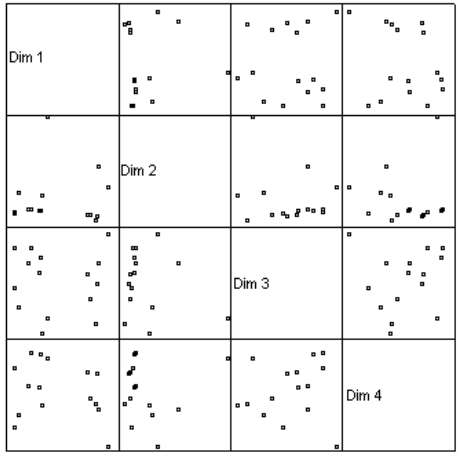
Figure 1.4 Three-dimensional plot of object scores, displaying dimensions 1, 2, and 4



Scatterplot Matrices

To view more than three dimensions on a single plot, it is useful to graphically display two-dimensional scatterplots for all dimensions in a matrix scatterplot. To convert a chart into a scatterplot matrix, use the scatterplot gallery in the Chart Editor. This option offers great flexibility in converting one chart type to another. A scatterplot matrix displaying four dimensions is shown in Figure 1.5.

Figure 1.5 Scatterplot matrix displaying object scores for four dimensions



In contrast to the other dimension reduction techniques, correspondence analysis produces a matrix of scatterplots similar to Figure 1.5 for all analyses. If you desire individual two- or three-dimensional scatterplots, use the Scatter option on the Gallery menu in the Chart Editor. Alternatively, to omit or add dimensions to an existing scatterplot matrix, use the *Displayed* command on the Series menu (see the *SPSS Base User's Guide* for information on editing charts and using the chart gallery).

Aspect Ratio in Optimal Scaling Charts

Aspect ratio in optimal scaling plots is **isotropic**. In a two-dimensional plot, the distance representing one unit in dimension 1 is equal to the distance representing one unit in dimension 2. If you change the range of a dimension in a two-dimensional plot, the system changes the size of the other dimension to keep the physical distances equal. Isotropic aspect ratio cannot be overridden for the optimal scaling procedures.

2 Categorical Regression (CATREG)

Categorical regression quantifies categorical data by assigning numerical values to the categories, resulting in an optimal linear regression equation for the transformed variables. Categorical regression is also known by the acronym CATREG, for *cat*egorical *reg*ression.

Standard linear regression analysis involves minimizing the sum of squared differences between a response (dependent) variable and a weighted combination of predictor (independent) variables. Variables are typically quantitative, with (nominal) categorical data recoded to binary or contrast variables. As a result, categorical variables serve to separate groups of cases, and the technique estimates separate sets of parameters for each group. The estimated coefficients reflect how changes in the predictors affect the response. Prediction of the response is possible for any combination of predictor values.

An alternative approach involves regressing the response on the categorical predictor values themselves. Consequently, one coefficient is estimated for each variable. However, for categorical variables, the category values are arbitrary. Coding the categories in different ways yield different coefficients, making comparisons across analyses of the same variables difficult.

CATREG extends the standard approach by simultaneously scaling nominal, ordinal, and numerical variables. The procedure quantifies categorical variables such that the quantifications reflect characteristics of the original categories. The procedure treats quantified categorical variables in the same way as numerical variables. Using nonlinear transformations allow variables to be analyzed at a variety of levels to find the best-fitting model.

Example. Categorical regression could be used to describe how job satisfaction depends on job category, geographic region, and amount of travel. You might find that high levels of satisfaction correspond to managers and low travel. The resulting regression equation could be used to predict job satisfaction for any combination of the three independent variables.

Statistics and plots. Frequencies, regression coefficients, ANOVA table, iteration history, category quantifications, correlations between untransformed predictors, correlations between transformed predictors, residual plots, and transformation plots.

Data. CATREG operates on category indicator variables. The category indicators should be positive integers. You can use the Discretization dialog box to convert fractional-value variables and string variables into positive integers.

Assumptions. Only one response variable is allowed, but the maximum number of predictor variables is 200. The data must contain at least three valid cases, and the number of valid cases must exceed the number of predictor variables plus one.

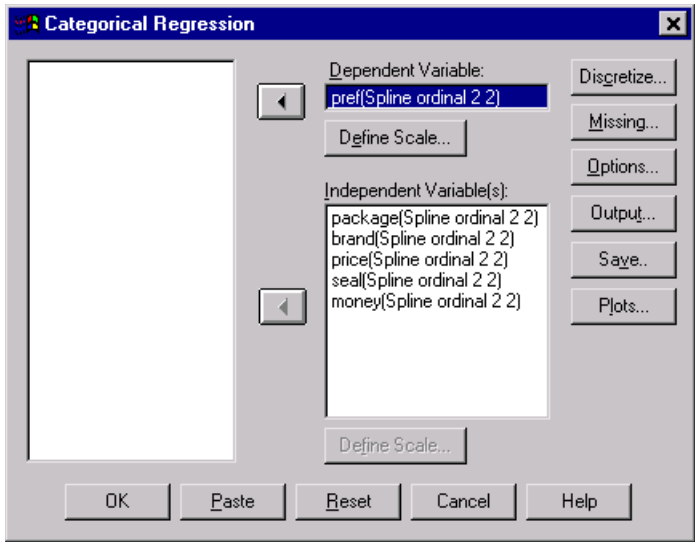
Related procedures. CATREG is equivalent to categorical canonical correlation analysis with optimal scaling (OVERALS) with two sets, one of which contains only one variable. Scaling all variables at the numerical level corresponds to standard multiple regression analysis.

To Obtain a Categorical Regression

▶ From the menus choose:

Analyze
  Regression
    Optimal Scaling…

Figure 2.1 Categorical Regression dialog box



▶ Select the dependent variable and independent variable(s).

▶ Click OK.

Optionally, change the scaling level for each variable.

Define Scale in Categorical Regression

You can set the optimal scaling level for the dependent and independent variables. By default, they are scaled as second-degree monotonic splines (ordinal) with two interior knots. Additionally, you can set the weight for analysis variables.

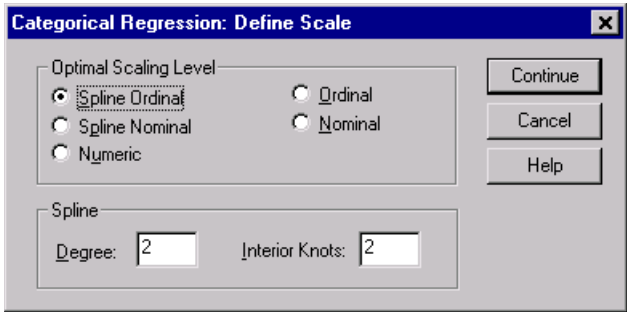
Optimal Scaling Level. You can also select the scaling level for quantifying each variable.

- **Spline Ordinal.** The order of the categories of the observed variable is preserved in the optimally scaled variable. Category points will be on a straight line (vector) through the origin. The resulting transformation is a smooth monotonic piecewise polynomial of the chosen degree. The pieces are specified by the user-specified number and procedure-determined placement of the interior knots.
- **Spline Nominal.** The only information in the observed variable that is preserved in the optimally scaled variable is the grouping of objects in categories. The order of the categories of the observed variable is not preserved. Category points will be on a straight line (vector) through the origin. The resulting transformation is a smooth, possibly nonmonotonic, piecewise polynomial of the chosen degree. The pieces are specified by the user-specified number and procedure-determined placement of the interior knots.
- **Ordinal.** The order of the categories of the observed variable is preserved in the optimally scaled variable. Category points will be on a straight line (vector) through the origin. The resulting transformation fits better than the spline ordinal transformation but is less smooth.
- **Nominal.** The only information in the observed variable that is preserved in the optimally scaled variable is the grouping of objects in categories. The order of the categories of the observed variable is not preserved. Category points will be on a straight line (vector) through the origin. The resulting transformation fits better than the spline nominal transformation but is less smooth.
- **Numeric.** Categories are treated as ordered and equally spaced (interval level). The order of the categories and the equal distances between category numbers of the observed variable are preserved in the optimally scaled variable. Category points will be on a straight line (vector) through the origin. When all variables are at the numeric level, the analysis is analogous to standard principal components analysis.

To Define the Scale in CATREG

- Select one or more variables on the variables list in the Categorical Regression dialog box.
- Click Define Scale.

Figure 2.2 Categorical Regression Define Scale dialog box

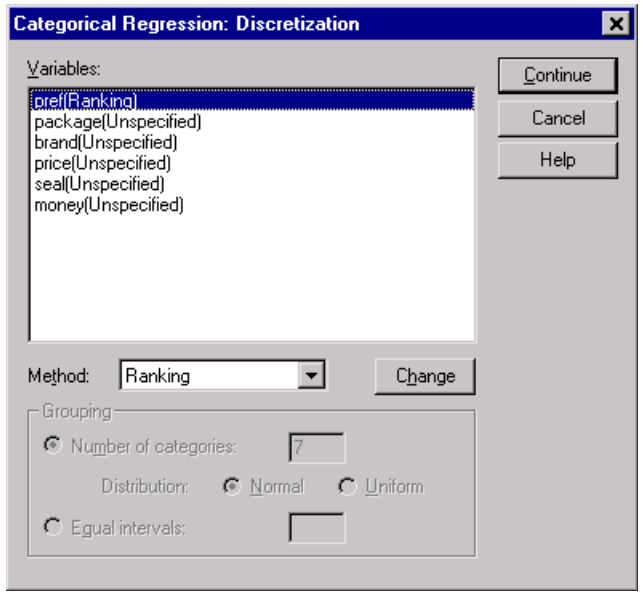


- Select the optimal scaling level to be used in the analysis.
- Click Continue.

Categorical Regression Discretization

The Discretization dialog box allows you to select a method of recoding your variables. Fractional-value variables are grouped into seven categories (or into the number of distinct values of the variable, if this number is less than seven) with an approximately normal distribution, unless specified otherwise. String variables are always converted into positive integers by assigning category indicators according to ascending alphanumeric order. Discretization for string variables applies to these integers. Other variables are left alone by default. The discretized variables are then used in the analysis.

Figure 2.3 Categorical Regression Discretization dialog box



Method. Choose between grouping, ranking, or multiplying.

- **Grouping.** Recode into a specified number of categories or recode by interval.
- **Ranking.** The variable is discretized by ranking the cases.
- **Multiplying.** The current values of the variable are standardized, multiplied by 10, rounded, and have a constant added such that the lowest discretized value is 1.

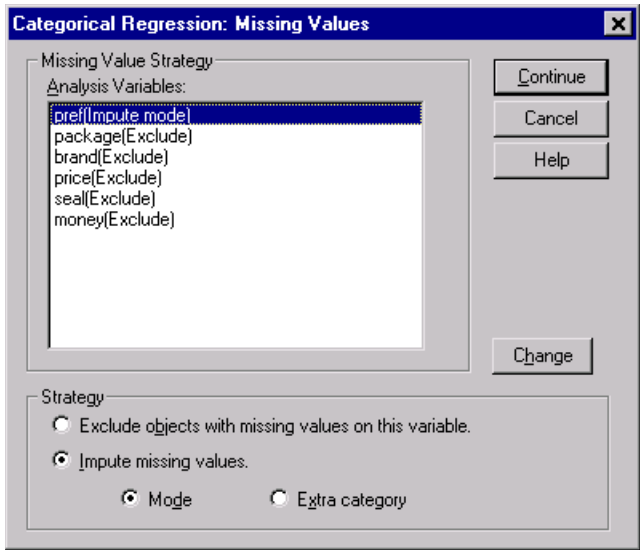
Grouping. The following options are available when discretizing variables by grouping:

- **Number of categories.** Specify a number of categories and whether the values of the variable should follow an approximately normal or uniform distribution across those categories.
- **Equal intervals.** Variables are recoded into categories defined by these equally sized intervals. You must specify the length of the intervals.

Categorical Regression Missing Values

The Missing Values dialog box allows you to choose the strategy for handling missing values in analysis variables and supplementary variables.

Figure 2.4 Categorical Regression Missing Values dialog box



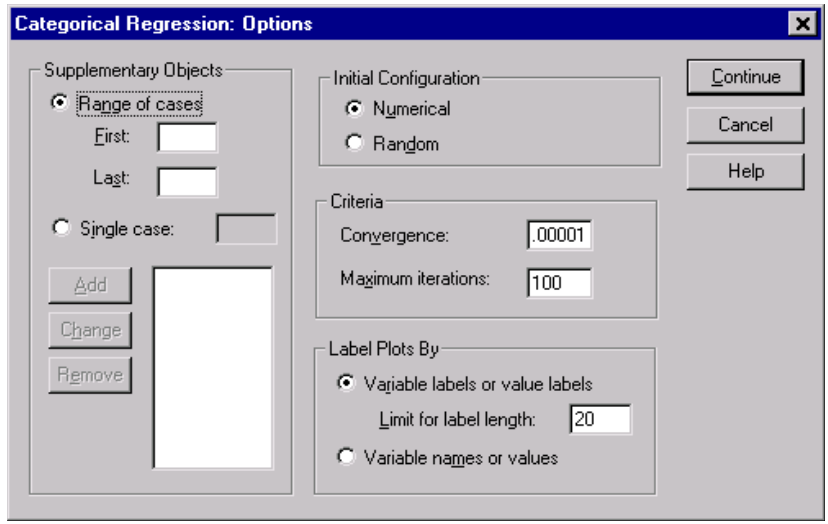
Strategy. Choose to impute missing values (active treatment) or exclude objects with missing values (listwise deletion).

- **Impute missing values.** Objects with missing values on the selected variable have those values imputed. You can choose the method of imputation. Select Mode to replace missing values with the most frequent category. When there are multiple modes, the one with the smallest category indicator is used. Select Extra category to replace missing values with the same quantification of an extra category. This implies that objects with a missing value on this variable are considered to belong to the same (extra) category.
- **Exclude objects with missing values on this variable.** Objects with missing values on the selected variable are excluded from the analysis. This strategy is not available for supplementary variables.

Categorical Regression Options

The Options dialog box allows you to select the initial configuration style, specify iteration and convergence criteria, select supplementary objects, and set the labeling of plots.

Figure 2.5 Categorical Regression Options dialog box



Supplementary Objects. This allows you to specify the objects that you want to treat as supplementary. Simply type the number of a supplementary object and click Add. You cannot weight supplementary objects (specified weights are ignored).

Initial Configuration. If no variables are treated as nominal, select the Numerical configuration. If at least one variable is treated as nominal, select the Random configuration.

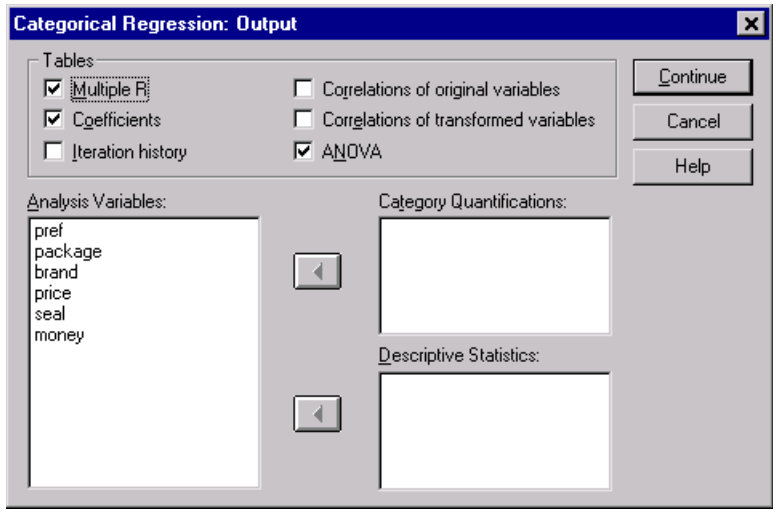
Criteria. You can specify the maximum number of iterations the regression may go through in its computations. You can also select a convergence criterion value. The regression stops iterating if the difference in total fit between the last two iterations is less than the convergence value or if the maximum number of iterations is reached.

Label Plots By. Allows you to specify whether variables and value labels or variable names and values will be used in the plots. You can also specify a maximum length for labels.

Categorical Regression Output

The Output dialog box allows you to select the statistics to display in the output.

Figure 2.6 Categorical Regression Output dialog box



Tables. Produces tables for:

- **Multiple R.** Includes R^2, adjusted R^2, and adjusted R^2 taking the optimal scaling into account.
- **Coefficients.** This option gives three tables: a Coefficients table that includes betas, standard error of the betas, *t* values, and significance; a Coefficients-Optimal Scaling table with the standard error of the betas taking the optimal scaling degrees of freedom into account; and a table with the zero-order, part, and partial correlation, Pratt's relative importance measure for the transformed predictors, and the tolerance before and after transformation.
- **Iteration history.** For each iteration, including the starting values for the algorithm, the multiple *R* and regression error are shown. The increase in multiple *R* is listed starting from the first iteration.
- **Correlations of the original variables.** A matrix showing the correlations between the untransformed variables is displayed.
- **Correlations of the transformed variables.** A matrix showing the correlations between the transformed variables is displayed.

- **ANOVA.** This option includes regression and residual sums of squares, mean squares, and *F*. Two ANOVA tables are displayed: one with degrees of freedom for the regression equal to the number of predictor variables and one with degrees of freedom for the regression taking the optimal scaling into account.

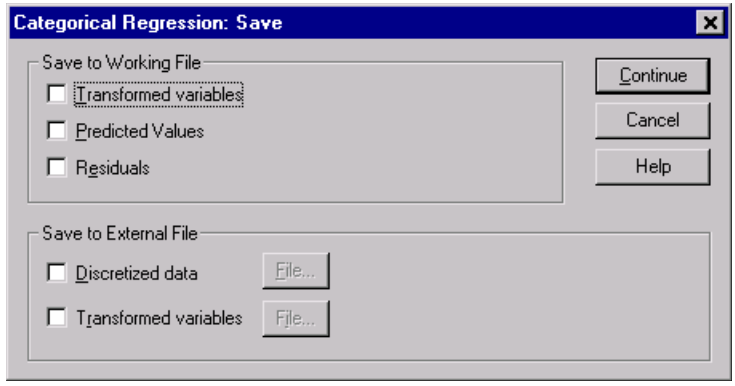
Category Quantifications. Tables showing the transformed values of the selected variables are displayed.

Descriptive Statistics. Tables showing the frequencies, missing values, and modes of the selected variables are displayed.

Categorical Regression Save

The Save dialog box allows you to save results to the working file or an external file.

Figure 2.7 Categorical Regression Save dialog box



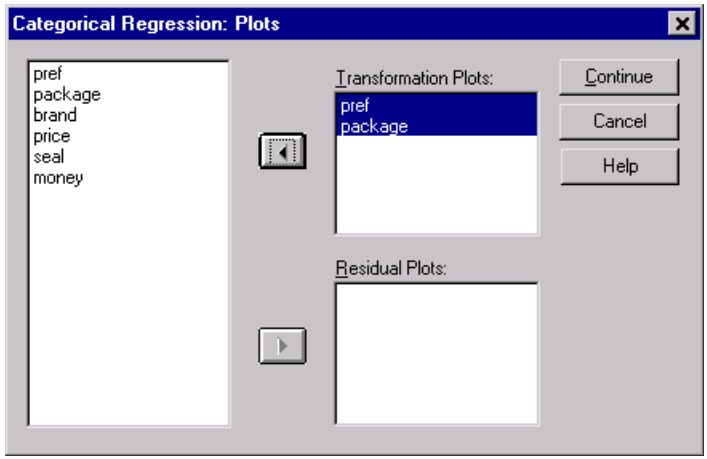
Save to Working File. You can save the transformed values of the variables, model-predicted values, and residuals to the working file.

Save to External File. You can save the discretized data and transformed variables to external files.

Categorical Regression Plots

The Plot dialog box allows you to specify the variables that will produce transformation and residual plots.

Figure 2.8 Categorical Regression Plot dialog box



Transformation Plots. For each of these variables, the category quantifications are plotted against the original category values. Empty categories appear on the horizontal axis but do not affect the computations. These categories are identified by breaks in the line connecting the quantifications.

Residual Plots. For each of these variables, residuals (computed for the dependent variable predicted from all predictor variables except the predictor variable in question) are plotted against category indicators and the optimal category quantifications multiplied with beta against category indicators.

CATREG Command Additional Features

You can customize your categorical regression if you paste your selections into a syntax window and edit the resulting CATREG command syntax. SPSS command language also allows you to:

- Specify rootnames for the transformed variables when saving them to the working data file (with the SAVE subcommand).

3 Categorical Principal Components Analysis (CATPCA)

This procedure simultaneously quantifies categorical variables while reducing the dimensionality of the data. Categorical principal components analysis is also known by the acronym CATPCA, for *cat*egorical *p*rincipal *c*omponents *a*nalysis.

The goal of principal components analysis is to reduce an original set of variables into a smaller set of uncorrelated components that represent most of the information found in the original variables. The technique is most useful when a large number of variables prohibits effective interpretation of the relationships between objects (subjects and units). By reducing the dimensionality, you interpret a few components rather than a large number of variables.

Standard principal components analysis assumes linear relationships between numeric variables. On the other hand, the optimal-scaling approach allows variables to be scaled at different levels. Categorical variables are optimally quantified in the specified dimensionality. As a result, nonlinear relationships between variables can be modeled.

Example. Categorical principal components analysis could be used to graphically display the relationship between job category, job division, region, amount of travel (high, medium, and low), and job satisfaction. You might find that two dimensions account for a large amount of variance. The first dimension might separate job category from region, whereas the second dimension might separate job division from amount of travel. You also might find that high job satisfaction is related to a medium amount of travel.

Statistics and plots. Frequencies, missing values, optimal scaling level, mode, variance accounted for by centroid coordinates, vector coordinates, total per variable and per dimension, component loadings for vector-quantified variables, category quantifications and coordinates, iteration history, correlations of the transformed variables and eigenvalues of the correlation matrix, correlations of the original variables and eigenvalues of the correlation matrix, object scores, category plots, joint category plots, transformation plots, residual plots, projected centroid plots, object plots, biplots, triplots, and component loadings plots.

Data. String variable values are always converted into positive integers by ascending alphanumeric order. User-defined missing values, system-missing values, and values less than 1 are considered missing; you can recode or add a constant to variables with values less than 1 to make them nonmissing.

Assumptions. The data must contain at least three valid cases. The analysis is based on positive integer data. The discretization option will automatically categorize a fractional-value variable by grouping its values into categories with a close to "normal" distribution and will automatically convert values of string variables into positive integers. You can specify other discretization schemes.

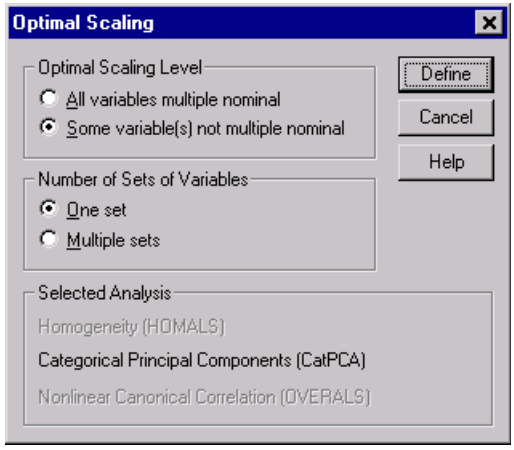
Related procedures. Scaling all variables at the numeric level corresponds to standard principal components analysis. Alternate plotting features are available by using the transformed variables in a standard linear principal components analysis. If all variables have multiple nominal scaling levels, categorical principal components analysis is identical to homogeneity analysis. If sets of variables are of interest, categorical (nonlinear) canonical correlation analysis should be used.

To Obtain a Categorical Principal Components Analysis

- From the menus choose:

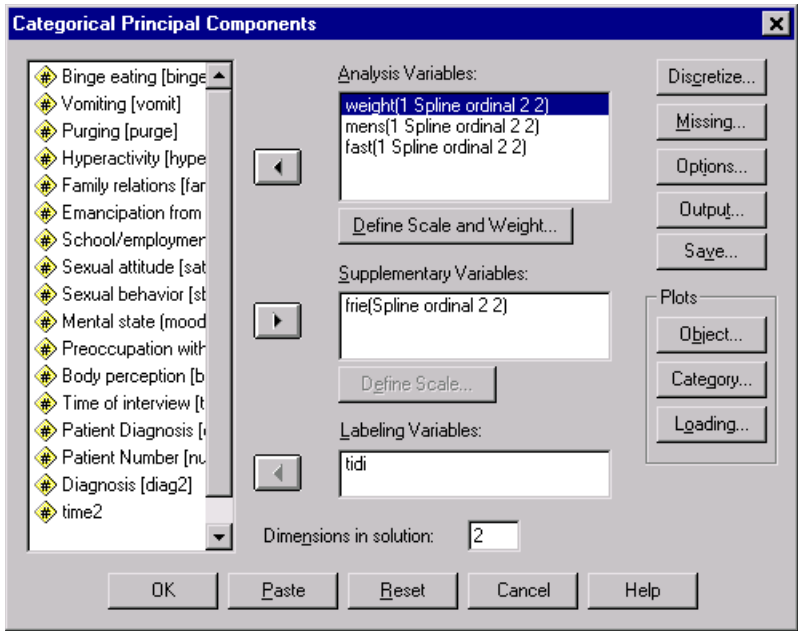
  Analyze
  Data Reduction
  Optimal Scaling...

Figure 3.1 Optimal Scaling dialog box



- Select Some variable(s) not multiple nominal.
- Select One set.
- Click Define.

Figure 3.2 Categorical Principal Components dialog box



- Select at least two analysis variables and specify the number of dimensions in the solution.
- Click OK.

You may optionally specify supplementary variables, which are fitted into the solution found, or labeling variables for the plots.

Define Scale and Weight in CATPCA

You can set the optimal scaling level for analysis variables and supplementary variables. By default, they are scaled as second-degree monotonic splines (ordinal) with two interior knots. Additionally, you can set the weight for analysis variables.

Variable weight. You can choose to define a weight for each variable. The value specified must be a positive integer. The default value is 1.

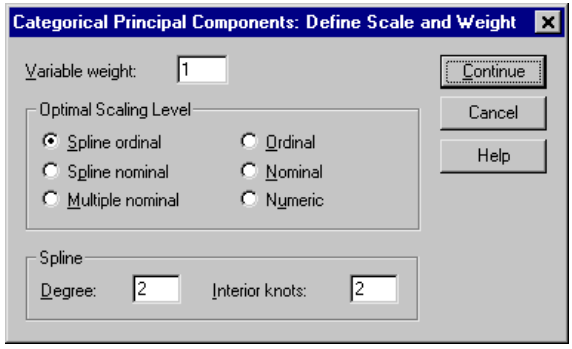
Optimal Scaling Level. You can also select the scaling level to be used to quantify each variable.

- **Spline ordinal.** The order of the categories of the observed variable is preserved in the optimally scaled variable. Category points will be on a straight line (vector) through the origin. The resulting transformation is a smooth monotonic piecewise polynomial of the chosen degree. The pieces are specified by the user-specified number and procedure-determined placement of the interior knots.
- **Spline nominal.** The only information in the observed variable that is preserved in the optimally scaled variable is the grouping of objects in categories. The order of the categories of the observed variable is not preserved. Category points will be on a straight line (vector) through the origin. The resulting transformation is a smooth, possibly nonmonotonic, piecewise polynomial of the chosen degree. The pieces are specified by the user-specified number and procedure-determined placement of the interior knots.
- **Multiple nominal.** The only information in the observed variable that is preserved in the optimally scaled variable is the grouping of objects in categories. The order of the categories of the observed variable is not preserved. Category points will be in the centroid of the objects in the particular categories. *Multiple* indicates that different sets of quantifications are obtained for each dimension.
- **Ordinal.** The order of the categories of the observed variable is preserved in the optimally scaled variable. Category points will be on a straight line (vector) through the origin. The resulting transformation fits better than the spline ordinal transformation but is less smooth.
- **Nominal.** The only information in the observed variable that is preserved in the optimally scaled variable is the grouping of objects in categories. The order of the categories of the observed variable is not preserved. Category points will be on a straight line (vector) through the origin. The resulting transformation fits better than the spline nominal transformation but is less smooth.
- **Numeric.** Categories are treated as ordered and equally spaced (interval level). The order of the categories and the equal distances between category numbers of the observed variable are preserved in the optimally scaled variable. Category points will be on a straight line (vector) through the origin. When all variables are at the numeric level, the analysis is analogous to standard principal components analysis.

To Define the Scale and Weight in CATPCA

- Select a variable in the Analysis Variables list in the Categorical Principal Components dialog box.
- Click Define Scale and Weight.

Figure 3.3 Categorical Principal Components Define Scale and Weight dialog box



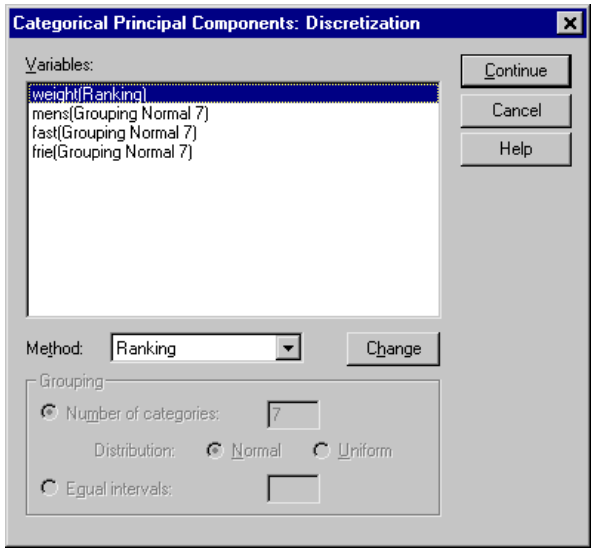
- Enter the weight value for the variable.
- Select the optimal scaling level to be used in the analysis. If you choose a spline transformation, you must also specify the degree of the polynomial and the number of interior knots.
- Click Continue.

You can alternatively define the scaling level for supplementary variables by selecting them from the list and clicking Define Scale.

Categorical Principal Components Discretization

The Discretization dialog box allows you to select a method of recoding your variables. Fractional-value variables are grouped into seven categories (or into the number of distinct values of the variable if this number is less than seven) with an approximately normal distribution, unless specified otherwise. String variables are always converted into positive integers by assigning category indicators according to ascending alphanumeric order. Discretization for string variables applies to these integers. Other variables are left alone by default. The discretized variables are then used in the analysis.

Figure 3.4 Categorical Principal Components Discretization dialog box



Method. Choose between grouping, ranking, and multiplying.

- **Grouping.** Recode into a specified number of categories or recode by interval.
- **Ranking.** The variable is discretized by ranking the cases.
- **Multiplying.** The current values of the variable are standardized, multiplied by 10, rounded, and have a constant added such that the lowest discretized value is 1.

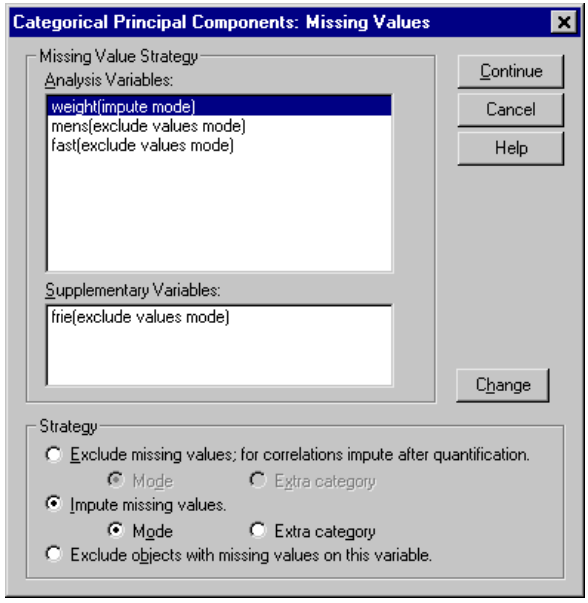
Grouping. The following options are available when discretizing variables by grouping:

- **Number of categories.** Specify a number of categories and whether the values of the variable should follow an approximately normal or uniform distribution across those categories.
- **Equal intervals.** Variables are recoded into categories defined by these equally sized intervals. You must specify the length of the intervals.

Categorical Principal Components Missing Values

The Missing Values dialog box allows you to choose the strategy for handling missing values in analysis variables and supplementary variables.

Figure 3.5 Categorical Principal Components Missing Values dialog box



Strategy. Choose to exclude missing values (passive treatment), impute missing values (active treatment), or exclude objects with missing values (listwise deletion).

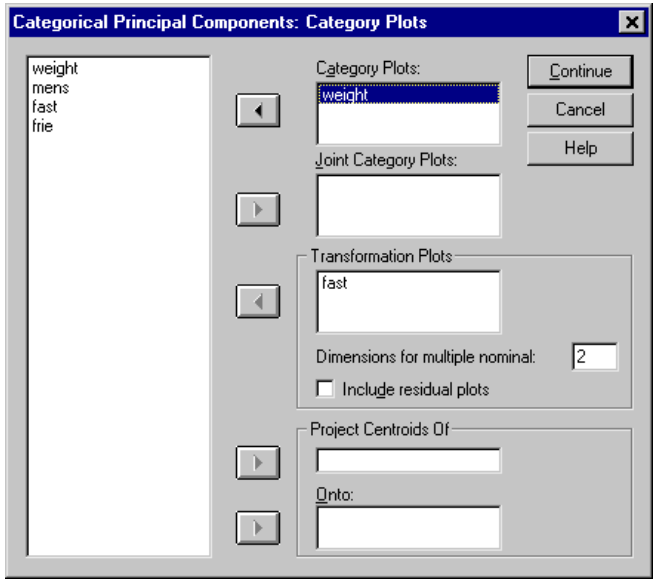
- **Exclude missing values; for correlations impute after quantification.** Objects with missing values on the selected variable do not contribute to the analysis for this variable. If all variables are given passive treatment, then objects with missing values on all variables are treated as supplementary. If correlations are specified in the Output dialog box, then (after analysis) missing values are imputed with the most frequent category, or mode, of the variable for the correlations of the original variables. For the correlations of the optimally scaled variables, you can choose the method of imputation. Select Mode to replace missing values with the mode of the optimally scaled variable. Select Extra category to replace missing values with the quantification of an extra category. This implies that objects with a missing value on this variable are considered to belong to the same (extra) category.

- **Impute missing values.** Objects with missing values on the selected variable have those values imputed. You can choose the method of imputation. Select Mode to replace missing values with the most frequent category. When there are multiple modes, the one with the smallest category indicator is used. Select Extra category to replace missing values with the same quantification of an extra category. This implies that objects with a missing value on this variable are considered to belong to the same (extra) category.
- **Exclude objects with missing values on this variable.** Objects with missing values on the selected variable are excluded from the analysis. This strategy is not available for supplementary variables.

Categorical Principal Components Category Plots

The Category Plots dialog box allows you to specify the types of plots desired and the variables for which plots will be produced.

Figure 3.6 Categorical Principal Components Category Plots dialog box



Category Plots. For each variable selected, a plot of the centroid and vector coordinates is plotted. For variables with multiple nominal scaling levels, categories are in the centroids of the objects in the particular categories. For all other scaling levels, categories are on a vector through the origin.

Joint Category Plots. This is a single plot of the centroid and vector coordinates of each selected variable.

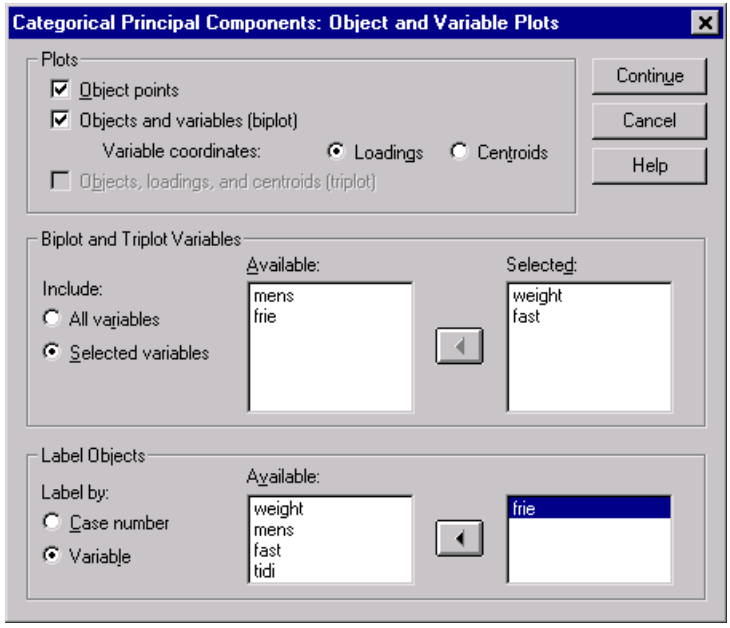
Transformation Plots. Displays a plot of the optimal category quantifications versus the category indicators. You can specify the number of dimensions desired for variables with multiple nominal scaling levels; one plot will be generated for each dimension. You can also choose to display residual plots for each variable selected.

Project Centroids Of. You may choose a variable and project its centroids onto selected variables. Variables with multiple nominal scaling levels cannot be selected to project on. When this plot is requested, a table with the coordinates of the projected centroids is also displayed.

Categorical Principal Components Object and Variable Plots

The Object and Variable Plots dialog box allows you to specify the types of plots desired and the variables for which plots will be produced.

Figure 3.7 Categorical Principal Components Object and Variable Plots dialog box



Object points. A plot of the object points is displayed.

Objects and variables (biplot). The object points are plotted with your choice of the variable coordinates—component loadings or variable centroids.

Objects, loadings, and centroids (triplot). The object points are plotted with the centroids of multiple nominal-scaling-level variables and the component loadings of other variables.

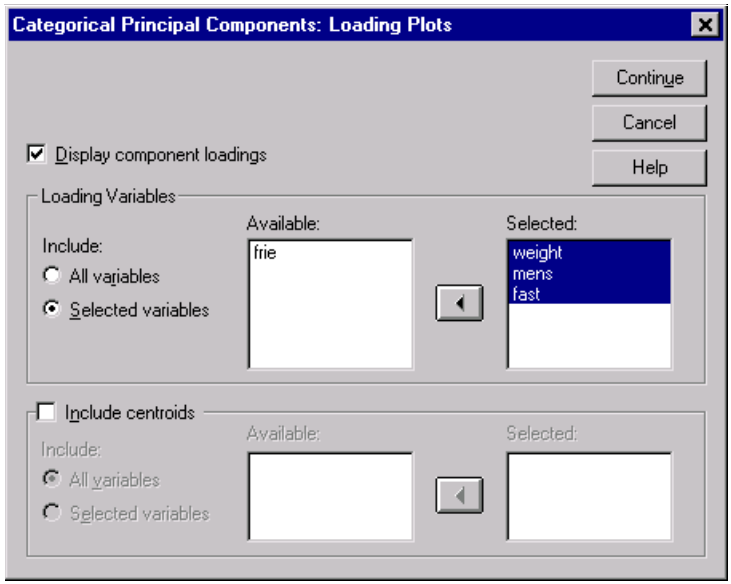
Biplot and Triplot Variables. You can choose to use all variables for the biplots and triplots, or select a subset.

Label Objects. You can choose to have objects labeled with the categories of selected variables (you may choose category indicator values or value labels in the Options dialog box) or with their case numbers. One plot is produced per variable, if Variable is selected.

Categorical Principal Components Loading Plots

The Loading Plots dialog box allows you to specify the variables which will be included in the plot, and whether or not to include centroids in the plot.

Figure 3.8 Categorical Principal Components Loading Plots dialog box



Display component loadings. If selected, a plot of the component loadings is displayed.

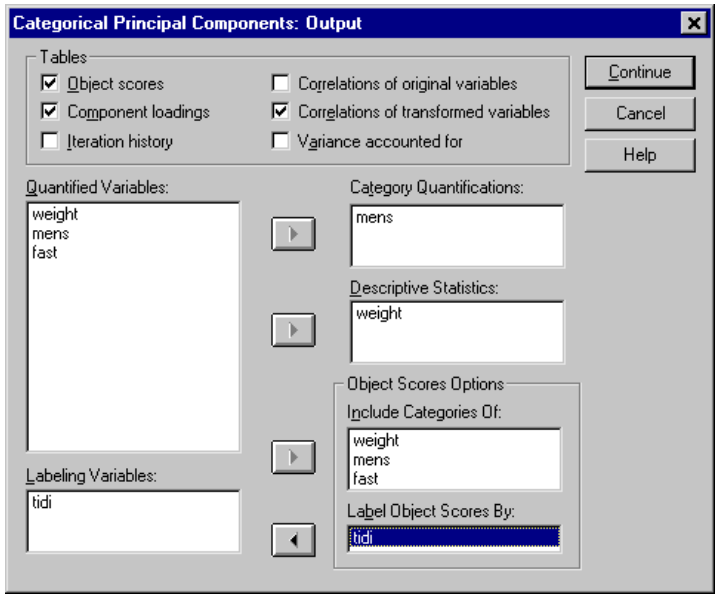
Loading Variables. You can choose to use all variables for the component loadings plot or select a subset.

Include centroids. Variables with multiple nominal scaling levels do not have component loadings, but you may choose to include the centroids of those variables in the plot. You can choose to use all multiple nominal variables or select a subset.

Categorical Principal Components Output

The Output dialog box allows you to produce tables for object scores, component loadings, iteration history, correlations of original and transformed variables, the variance accounted for per variable and per dimension, category quantifications for selected variables, and descriptive statistics for selected variables.

Figure 3.9 Categorical Principal Components Output dialog box



Object scores. Displays the object scores and has the following options:

- **Include Categories Of.** Displays the category indicators of the analysis variables selected.
- **Label Object Scores By.** From the list of variables specified as labeling variables, you can select one to label the objects.

Component loadings. Displays the component loadings for all variables that were not given multiple nominal scaling levels.

Iteration history. For each iteration, the variance accounted for, loss, and increase in variance accounted for are shown.

Correlations of original variables. Shows the correlation matrix of the original variables and the eigenvalues of that matrix.

Correlations of transformed variables. Shows the correlation matrix of the transformed (optimally scaled) variables and the eigenvalues of that matrix.

Variance accounted for. Displays the amount of variance accounted for by centroid coordinates, vector coordinates, and total (centroid and vector coordinates combined) per variable and per dimension.

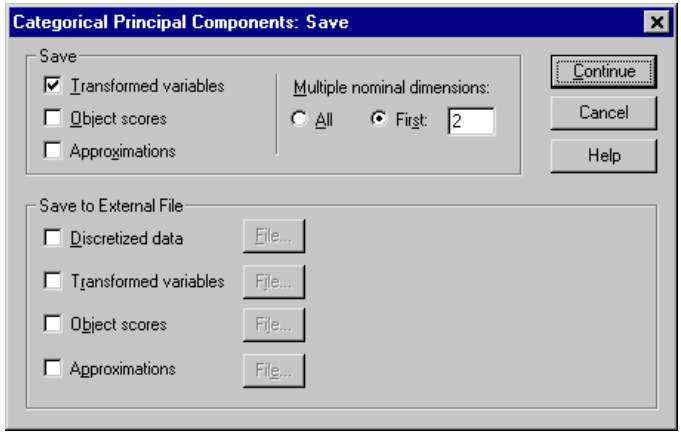
Category Quantifications. Gives the category quantifications and coordinates for each dimension of the variable(s) selected.

Descriptive Statistics. Displays frequencies, number of missing values, and mode of the variable(s) selected.

Categorical Principal Components Save

The Save dialog box allows you to add the transformed variables, object scores, and approximations to the working data file or as new variables in external files and save the discretized data as new variables in an external data file.

Figure 3.10 Categorical Principal Components Save dialog box



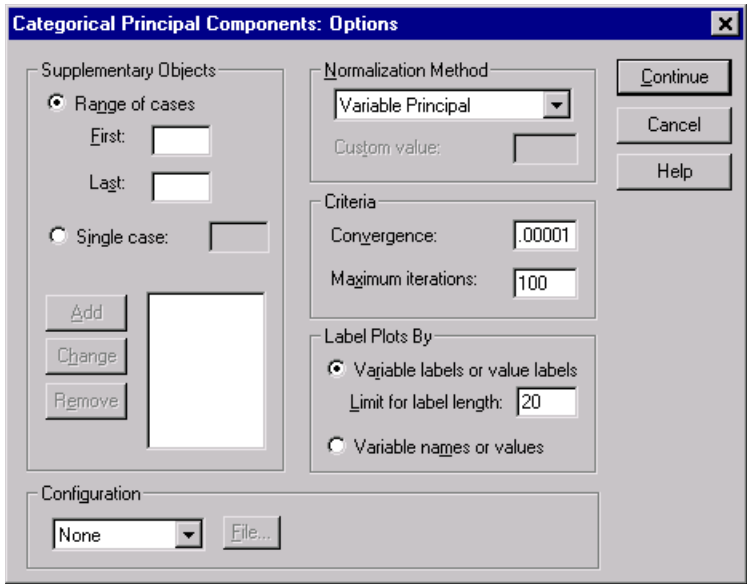
Save. Save selections to the working data file. If any variable has been given the multiple nominal scaling level, the number of dimensions to be saved must be specified.

Save to External File. Save selections to a new external file. Specify a filename for each selected option by clicking File. Each file specified must have a different name.

Categorical Principal Components Options

The Options dialog box allows you to select the initial configuration, specify iteration and convergence criteria, select a normalization method, choose the method for labeling plots, and specify supplementary objects.

Figure 3.11 Categorical Principal Components Options dialog box



Supplementary Objects. Specify the case number of the object, or the first and last case numbers of a range of objects, that you want to make supplementary and then click Add. Continue until you have specified all of your supplementary objects. If an object is specified as supplementary, then case weights are ignored for that object.

Normalization Method. You can specify one of five options for normalizing the object scores and the variables. Only one normalization method can be used in a given analysis.

- **Variable Principal.** This option optimizes the association between variables. The coordinates of the variables in the object space are the component loadings (correlations with principal components, such as dimensions and object scores). This is useful when you are primarily interested in the correlation between the variables.
- **Object Principal.** This option optimizes distances between objects. This is useful when you are primarily interested in differences or similarities between the objects.
- **Symmetrical.** Use this normalization option if you are primarily interested in the relation between objects and variables.

- **Independent.** Use this normalization option if you want to examine distances between objects and correlations between variables separately.
- **Custom.** You can specify any real value in the closed interval [–1, 1]. A value of 1 is equal to the Object Principal method, a value of 0 is equal to the Symmetrical method, and a value of –1 is equal to the Variable Principal method. By specifying a value greater than –1 and less than 1, you can spread the eigenvalue over both objects and variables. This method is useful for making a tailor-made biplot or triplot.

Criteria. You can specify the maximum number of iterations the procedure can go through in its computations. You can also select a convergence criterion value. The algorithm stops iterating if the difference in total fit between the last two iterations is less than the convergence value or if the maximum number of iterations is reached.

Configuration. You can read data from a file containing the coordinates of a configuration. The first variable in the file should contain the coordinates for the first dimension, the second variable should contain the coordinates for the second dimension, and so on.

- **Initial.** The configuration in the file specified will be used as the starting point of the analysis.
- **Fixed.** The configuration in the file specified will be used to fit in the variables. The variables that are fitted in must be selected as analysis variables, but because the configuration is fixed, they are treated as supplementary variables (so they do not need to be selected as supplementary variables).

Label Plots By. Allows you to specify whether variables and value labels or variable names and values will be used in the plots. You can also specify a maximum length for labels.

CATPCA Command Additional Features

You can customize your categorical principal components analysis if you paste your selections into a syntax window and edit the resulting CATPCA command syntax. SPSS command language also allows you to:

- Specify rootnames for the transformed variables, object scores, and approximations when saving them to the working data file (with the SAVE subcommand).
- Specify a maximum length for labels for each plot separately (with the PLOT subcommand).
- Specify a separate variable list for residual plots (with the PLOT subcommand).

4 Nonlinear Canonical Correlation Analysis (OVERALS)

Nonlinear canonical correlation analysis corresponds to categorical canonical correlation analysis with optimal scaling. The purpose of this procedure is to determine how similar sets of categorical variables are to one another. Nonlinear canonical correlation analysis is also known by the acronym OVERALS.

Standard canonical correlation analysis is an extension of multiple regression, where the second set does not contain a single response variable, but multiple ones. The goal is to explain as much as possible of the variance in the relationships among two sets of numerical variables in a low dimensional space. Initially, the variables in each set are linearly combined such that the linear combinations have a maximal correlation. Given these combinations, subsequent linear combinations are determined that are uncorrelated with the previous combinations and that have the largest correlation possible.

The optimal scaling approach expands the standard analysis in three crucial ways. First, OVERALS allows more than two sets of variables. Second, variables can be scaled as either nominal, ordinal, or numerical. As a result, nonlinear relationships between variables can be analyzed. Finally, instead of maximizing correlations between the variable sets, the sets are compared to an unknown compromise set defined by the object scores.

Example. Categorical canonical correlation analysis with optimal scaling could be used to graphically display the relationship between one set of variables containing job category and years of education and another set of variables containing minority classification and gender. You might find that years of education and minority classification discriminate better than the remaining variables. You might also find that years of education discriminates best on the first dimension.

Statistics and plots. Frequencies, centroids, iteration history, object scores, category quantifications, weights, component loadings, single and multiple fit, object scores plots, category coordinates plots, component loadings plots, category centroids plots, transformation plots.

Data. Use integers to code categorical variables (nominal or ordinal scaling level). To minimize output, use consecutive integers beginning with 1 to code each variable. Variables scaled at the numerical level should not be recoded to consecutive integers. To

minimize output, for each variable scaled at the numerical level, subtract the smallest observed value from every value and add 1. Fractional values are truncated after the decimal.

Assumptions. Variables can be classified into two or more sets. Variables in the analysis are scaled as multiple nominal, single nominal, ordinal, or numerical. The maximum number of dimensions used in the procedure depends on the optimal scaling level of the variables. If all variables are specified as ordinal, single nominal, or numerical, the maximum number of dimensions is the minimum of the number of observations minus 1 and the total number of variables. However, if only two sets of variables are defined, the maximum number of dimensions is the number of variables in the smaller set. If some variables are multiple nominal, the maximum number of dimensions is the total number of multiple nominal categories plus the number of nonmultiple nominal variables minus the number of multiple nominal variables. For example, if the analysis involves five variables, one of which is multiple nominal with four categories, the maximum number of dimensions is (4 + 4 – 1), or 7. If you specify a number greater than the maximum, the maximum value is used.

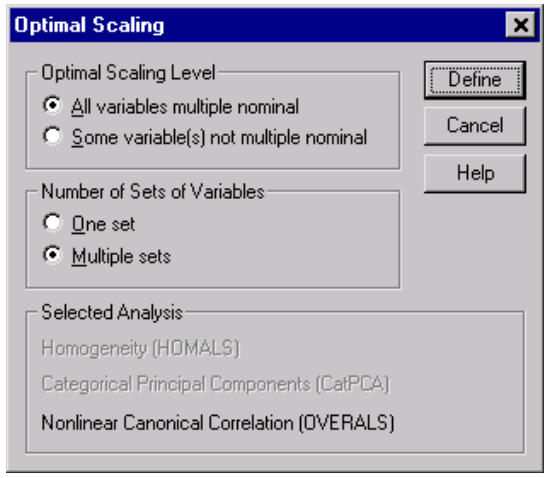
Related procedures. If each set contains one variable, nonlinear canonical correlation analysis is equivalent to principal components analysis with optimal scaling. If each of these variables is multiple nominal, the analysis corresponds to homogeneity analysis. If two sets of variables are involved and one of the sets contains only one variable, the analysis is identical to categorical regression with optimal scaling.

To Obtain a Nonlinear Canonical Correlation Analysis

▶ From the menus choose:

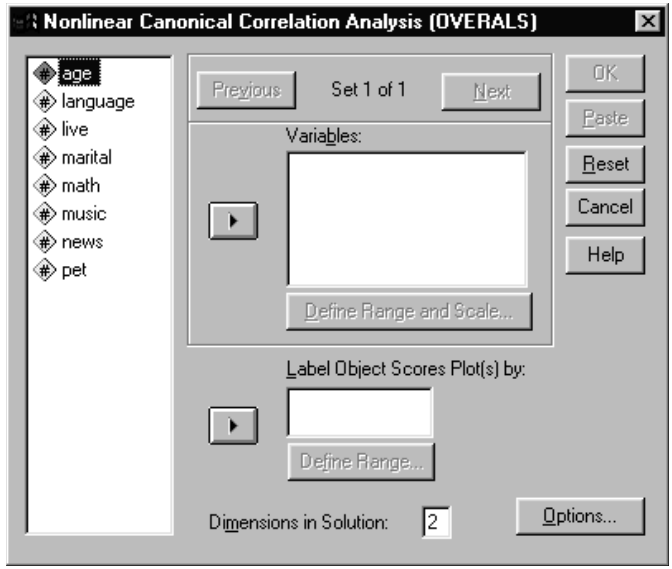
Analyze
  Data Reduction
    Optimal Scaling...

Figure 4.1 Optimal Scaling dialog box



- Select Multiple sets.
- Select either Some variable(s) not multiple nominal or All variables multiple nominal.
- Click Define.

Figure 4.2 Nonlinear Canonical Correlation Analysis (OVERALS) dialog box



- ▶ Define at least two sets of variables. Select the variable(s) that you want to include in the first set. To move to the next set, click Next, and select the variables that you want to include in the second set. You can add additional sets as desired. Click Previous to return to the previously defined variable set.

- ▶ Define the value range and measurement scale (optimal scaling level) for each selected variable.

- ▶ Click OK.

  Optionally, you can:

  - Select one or more variables to provide point labels for object scores plots. Each variable produces a separate plot, with the points labeled by the values of that variable. You must define a range for each of these plot label variables. Using the dialog box, a single variable cannot be used both in the analysis and as a labeling variable. If labeling the object scores plot with a variable used in the analysis is desired, use the Compute facility on the Transform menu to create a copy of that variable. Use the new variable to label the plot. Alternatively, command syntax can be used.
  - Specify the number of dimensions you want in the solution. In general, choose as few dimensions as needed to explain most of the variation. If the analysis involves more than two dimensions, SPSS produces three-dimensional plots of the first three dimensions. Other dimensions can be displayed by editing the chart.

Define Range and Scale in OVERALS

You must define a range for each variable. The maximum value specified must be an integer. Fractional data values are truncated in the analysis. A category value that is outside of the specified range is ignored in the analysis. To minimize output, use the Automatic Recode facility on the Transform menu to create consecutive categories beginning with 1 for variables treated as nominal or ordinal. Recoding to consecutive integers is not recommended for variables scaled at the numerical level. To minimize output for variables treated as numerical, for each variable, subtract the minimum value from every value and add 1.

You must also select the scaling to be used to quantify each variable.

Ordinal. The order of the categories of the observed variable is preserved in the quantified variable.

Single nominal. Objects in the same category receive the same score. When all variables are single nominal, the first dimension of this solution is the same as that of the first homogeneity analysis dimension.

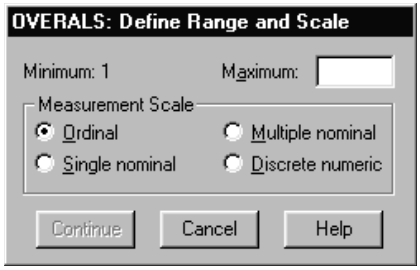
Multiple nominal. The quantifications can be different for each dimension. When all variables are multiple nominal and there is only one variable in each set, categorical canonical correlation analysis with optimal scaling produces the same results as homogeneity analysis.

Discrete numeric. Categories are treated as ordered and equally spaced. The differences between category numbers and the order of the categories of the observed variable are preserved in the quantified variable. When all variables are at the numerical level and there are two sets, the analysis is analogous to classical canonical correlation analysis.

To Define an Optimal Scaling Range and Scale in OVERALS

- In the OVERALS dialog box, select one or more variables in the Variables list.
- Click Define Range and Scale.

Figure 4.3 OVERALS Define Range and Scale dialog box



- Enter the maximum value for the variable. A minimum value of 1 is displayed. This minimum value cannot be changed.
- Select the measurement (optimal scaling) scale to be used in the analysis.
- Click Continue.

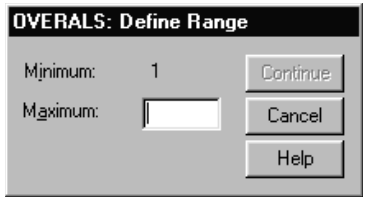
Define Range in OVERALS

You must define a range for each variable used to label the object scores plots. The maximum value specified must be an integer. Fractional data values are truncated in the analysis. Labels for category values outside of the specified range for a labeling variable do not appear in the plots. All cases with such category values are labeled with a single label corresponding to a data value outside of the defined range.

To Define an Optimal Scaling Range in OVERALS

- Select a variable for Label Object Scores Plot(s) By in the Nonlinear Canonical Correlation Analysis (OVERALS) dialog box.
- Click Define Range.

Figure 4.4 OVERALS Define Range dialog box

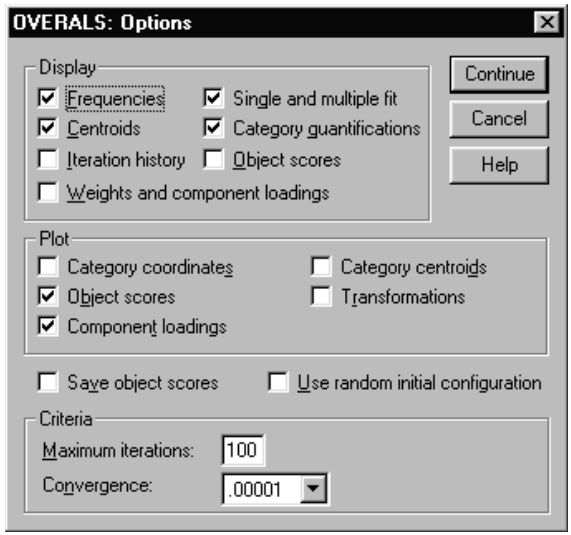


- Enter the maximum value for the variable. A minimum value of 1 is displayed. This minimum value cannot be changed.
- Click Continue.

Nonlinear Canonical Correlation Analysis Options

The Options dialog box allows you to select optional statistics and plots, save object scores as new variables in the working data file, specify iteration and convergence criteria, and specify an initial configuration for the analysis.

Figure 4.5 OVERALS Options dialog box



Display. Available statistics include marginal frequencies (counts), centroids, iteration history, weights and component loadings, category quantifications, object scores, and single and multiple fit statistics.

Plot. You can produce plots of category coordinates, object scores, component loadings, category centroids, and transformations.

Save object scores. You can save the object scores as new variables in the working data file. Object scores are saved for the number of dimensions specified in the main dialog box.

Use random initial configuration. A random initial configuration should be used if all or some of the variables are single nominal. If this option is not selected, a nested initial configuration is used.

Criteria. You can specify the maximum number of iterations the nonlinear canonical correlation analysis can go through in its computations. You can also select a convergence criterion value. The analysis stops iterating if the difference in total fit between the last two iterations is less than the convergence value or if the maximum number of iterations is reached.

OVERALS Command Additional Features

You can customize your nonlinear canonical correlation analysis if you paste your selections into a syntax window and edit the resulting OVERALS command syntax. SPSS command language also allows you to:

- Specify the dimension pairs to be plotted, rather than plotting all extracted dimensions (using the NDIM keyword on the PLOT subcommand).
- Specify the number of value label characters used to label points on the plots (with the PLOT subcommand).
- Designate more than five variables as labeling variables for object scores plots (with the PLOT subcommand).
- Select variables used in the analysis as labeling variables for the object scores plots (with the PLOT subcommand).
- Select variables to provide point labels for the quantification score plot (with the PLOT subcommand).
- Specify the number of cases to be included in the analysis, if you do not want to use all cases in the working data file (with the NOBSERVATIONS subcommand).
- Specify rootnames for variables created by saving object scores (with the SAVE subcommand).
- Specify the number of dimensions to be saved, rather than saving all extracted dimensions (with the SAVE subcommand).
- Write category quantifications to a matrix file (using the MATRIX subcommand).
- Produce low-resolution plots that may be easier to read than the usual high-resolution plots (using the SET command).
- Produce centroid and transformation plots for specified variables only (with the PLOT subcommand).

5 Correspondence Analysis

One of the goals of correspondence analysis is to describe the relationships between two nominal variables in a correspondence table in a low-dimensional space, while simultaneously describing the relationships between the categories for each variable. For each variable, the distances between category points in a plot reflect the relationships between the categories with similar categories plotted close to each other. Projecting points for one variable on the vector from the origin to a category point for the other variable describe the relationship between the variables.

An analysis of contingency tables often includes examining row and column profiles and testing for independence via the chi-square statistic. However, the number of profiles can be quite large, and the chi-square test does not reveal the dependence structure. The Crosstabs procedure offers several measures of association and tests of association but cannot graphically represent any relationships between the variables.

Factor analysis is a standard technique for describing relationships between variables in a low-dimensional space. However, factor analysis requires interval data, and the number of observations should be five times the number of variables. Correspondence analysis, on the other hand, assumes nominal variables and can describe the relationships between categories of each variable, as well as the relationship between the variables. In addition, correspondence analysis can be used to analyze any table of positive correspondence measures.

Example. Correspondence analysis could be used to graphically display the relationship between staff category and smoking habits. You might find that with regard to smoking, junior managers differ from secretaries, but secretaries do not differ from senior managers. You might also find that heavy smoking is associated with junior managers, whereas light smoking is associated with secretaries.

Statistics and plots. Correspondence measures, row and column profiles, singular values, row and column scores, inertia, mass, row and column score confidence statistics, singular value confidence statistics, transformation plots, row point plots, column point plots, and biplots.

Data. Categorical variables to be analyzed are scaled nominally. For aggregated data or for a correspondence measure other than frequencies, use a weighting variable with positive similarity values. Alternatively, for table data, use syntax to read the table.

Assumptions. The maximum number of dimensions used in the procedure depends on the number of active rows and column categories and the number of equality constraints. If no equality constraints are used and all categories are active, the maximum dimensionality is one fewer than the number of categories for the variable with the fewest categories. For example, if one variable has five categories and the other has four, the maximum number of dimensions is three. Supplementary categories are not active. For example, if one variable has five categories, two of which are supplementary, and the other variable has four categories, the maximum number of dimensions is two. Treat all sets of categories that are constrained to be equal as one category. For example, if a variable has five categories, three of which are constrained to be equal, that variable should be treated as having three categories when determining the maximum dimensionality. Two of the categories are unconstrained, and the third category corresponds to the three constrained categories. If you specify a number of dimensions greater than the maximum, the maximum value is used.

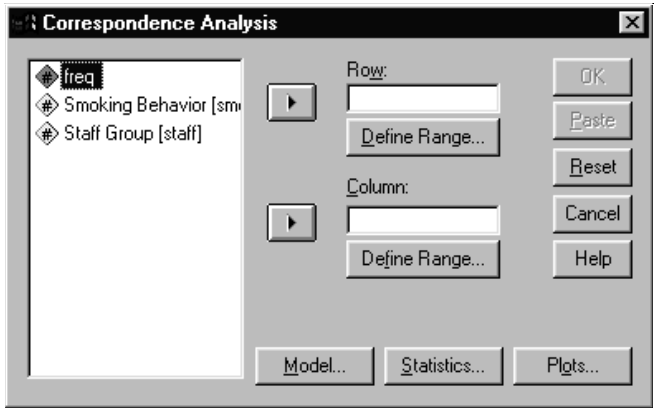
Related procedures. If more than two variables are involved, use homogeneity analysis. If the variables should be scaled ordinally, use principal components analysis with optimal scaling.

To Obtain a Correspondence Analysis

▶ From the menus choose:

Analyze
  Data Reduction
    Correspondence Analysis…

Figure 5.1 Correspondence Analysis dialog box



- ▶ Select a row variable.
- ▶ Select a column variable.
- ▶ Define the ranges for the variables.
- ▶ Click OK.

Define Row Range in Correspondence Analysis

You must define a range for the row variable. The minimum and maximum values specified must be integers. Fractional data values are truncated in the analysis. A category value that is outside of the specified range is ignored in the analysis.

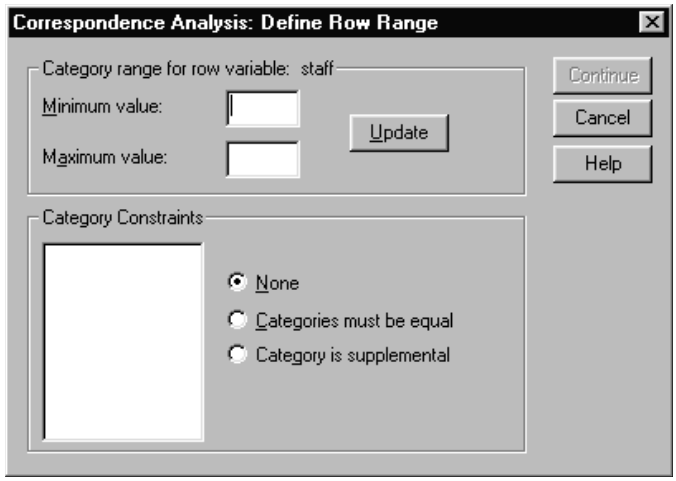
All categories are initially unconstrained and active. You can constrain row categories to equal other row categories, or you can define a row category as supplementary.

- **Categories must be equal.** Categories must have equal scores. Use equality constraints if the obtained order for the categories is undesirable or counterintuitive. The maximum number of row categories that can be constrained to be equal is the total number of active row categories minus 1. To impose different equality constraints on sets of categories, use syntax. For example, use syntax to constrain categories 1 and 2 to be equal and categories 3 and 4 to be equal.
- **Category is supplemental.** Supplementary categories do not influence the analysis but are represented in the space defined by the active categories. Supplementary categories play no role in defining the dimensions. The maximum number of supplementary row categories is the total number of row categories minus 2.

To Define a Row Range in Correspondence Analysis

- ▶ Select the row variable in the Correspondence Analysis dialog box.
- ▶ Click Define Range.

Figure 5.2 Correspondence Analysis Define Row Range dialog box



- Enter the minimum and maximum values for the row variable.
- Click Update.
- Click Continue.

Optionally, you can specify equality constraints on the row variable categories and define categories to be supplementary. For each category to be constrained or supplementary, select the category from the list of categories generated by Update and choose Category is supplemental or Categories must be equal. For equality constraints, at least two categories must be designated as equal.

Define Column Range in Correspondence Analysis

You must define a range for the column variable. The minimum and maximum values specified must be integers. Fractional data values are truncated in the analysis. A category value that is outside of the specified range is ignored in the analysis.

All categories are initially unconstrained and active. You can constrain column categories to equal other column categories or you can define a column category as supplementary.

- **Categories must be equal.** Categories must have equal scores. Use equality constraints if the obtained order for the categories is undesirable or counterintuitive. The maximum number of column categories that can be constrained to be equal is the total number of active column categories minus 1. To impose different equality con-

straints on sets of categories, use syntax. For example, use syntax to constrain categories 1 and 2 to be equal and categories 3 and 4 to be equal.

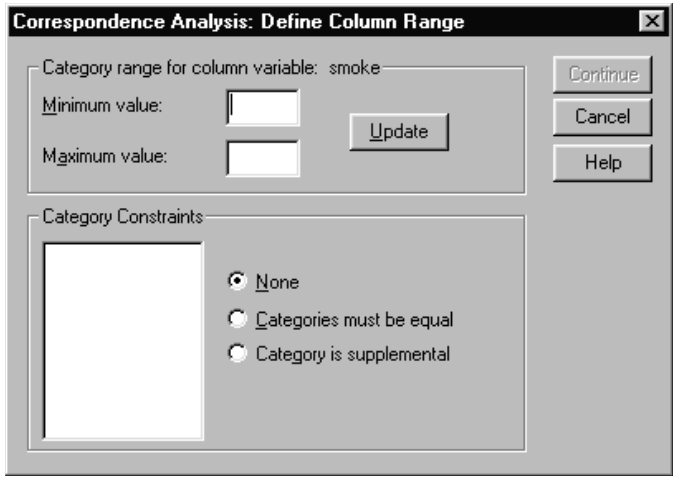
- **Category is supplemental.** Supplementary categories do not influence the analysis but are represented in the space defined by the active categories. Supplementary categories play no role in defining the dimensions. The maximum number of supplementary column categories is the total number of column categories minus 2.

To Define a Column Range in Correspondence Analysis

▶ Select the column variable in the Correspondence Analysis dialog box.

▶ Click Define Range.

Figure 5.3 Correspondence Analysis Define Column Range dialog box



▶ Enter the minimum and maximum values for the column variable.

▶ Click Update.

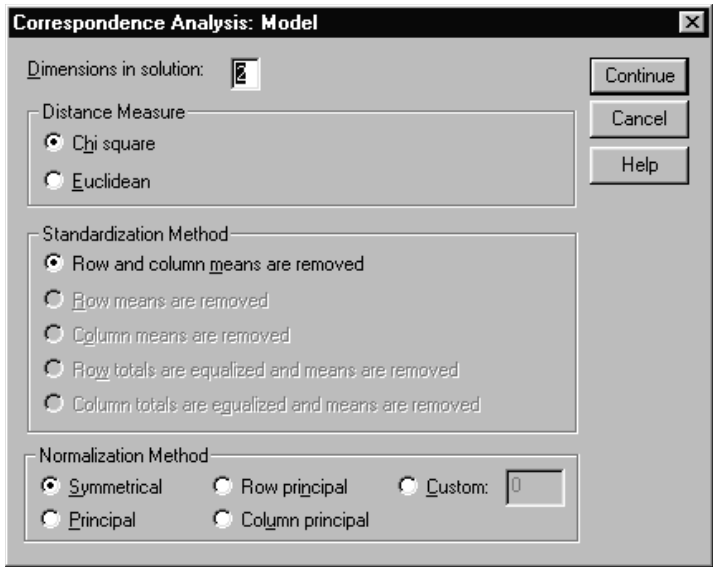
▶ Click Continue.

Optionally, you can specify equality constraints on the column variable categories and define categories to be supplementary. For each category to be constrained or supplementary, select the category from the list of categories generated by Update and choose Category is supplemental or Categories must be equal. For equality constraints, at least two categories must be designated as equal.

Correspondence Analysis Model

The Model dialog box allows you to specify the number of dimensions, the distance measure, the standardization method, and the normalization method.

Figure 5.4 Correspondence Analysis Model dialog box



Dimensions in solution. Specify the number of dimensions. In general, choose as few dimensions as needed to explain most of the variation. The maximum number of dimensions depends on the number of active categories used in the analysis and on the equality constraints. The maximum number of dimensions is the smaller of:

- The number of active row categories minus the number of row categories constrained to be equal, plus the number of constrained row category sets
- The number of active column categories minus the number of column categories constrained to be equal, plus the number of constrained column category sets

Distance Measure. You can select the measure of distance among the rows and columns of the correspondence table. Choose one of the following alternatives:

- **Chi square.** Use a weighted profile distance, where the weight is the mass of the rows or columns. This measure is required for standard correspondence analysis.
- **Euclidean.** Use the square root of the sum of squared differences between pairs of rows and pairs of columns.

Standardization Method. Choose one of the following alternatives:

- **Row and column means are removed.** Both the rows and columns are centered. This method is required for standard correspondence analysis.
- **Row means are removed.** Only the rows are centered.
- **Column means are removed.** Only the columns are centered.
- **Row totals are equalized and means are removed.** Before centering the rows, the row margins are equalized.
- **Column totals are equalized and means are removed.** Before centering the columns, the column margins are equalized.

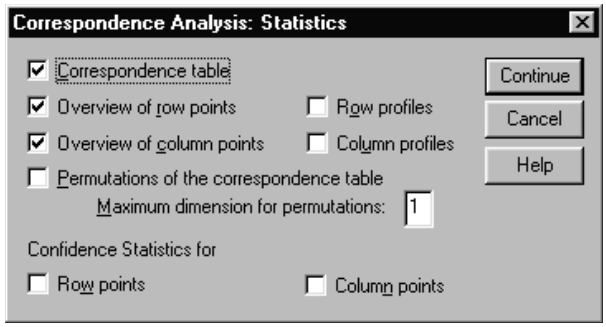
Normalization Method. Choose one of the following alternatives:

- **Symmetrical.** For each dimension, the row scores are the weighted average of the column scores divided by the matching singular value, and the column scores are the weighted average of row scores divided by the matching singular value. Use this method if you want to examine the differences or similarities between the categories of the two variables.
- **Principal.** The distances between row points and column points are approximations of the distances in the correspondence table according to the selected distance measure. Use this method if you want to examine differences between categories of either or both variables instead of differences between the two variables.
- **Row principal.** The distances between row points are approximations of the distances in the correspondence table according to the selected distance measure. The row scores are the weighted average of the column scores. Use this method if you want to examine differences or similarities between categories of the row variable.
- **Column principal.** The distances between column points are approximations of the distances in the correspondence table according to the selected distance measure. The column scores are the weighted average of the row scores. Use this method if you want to examine differences or similarities between categories of the column variable.
- **Custom.** You must specify a value between –1 and 1. A value of –1 corresponds to column principal. A value of 1 corresponds to row principal. A value of 0 corresponds to symmetrical. All other values spread the inertia over both the row and column scores to varying degrees. This method is useful for making tailor-made biplots.

Correspondence Analysis Statistics

The Statistics dialog box allows you to specify the numerical output produced.

Figure 5.5 Correspondence Analysis Statistics dialog box



Correspondence table. A crosstabulation of the input variables with row and column marginal totals.

Overview of row points. For each row category, the scores, mass, inertia, contribution to the inertia of the dimension, and the contribution of the dimension to the inertia of the point.

Overview of column points. For each column category, the scores, mass, inertia, contribution to the inertia of the dimension, and the contribution of the dimension to the inertia of the point.

Row profiles. For each row category, the distribution across the categories of the column variable.

Column profiles. For each column category, the distribution across the categories of the row variable.

Permutations of the correspondence table. The correspondence table reorganized such that the rows and columns are in increasing order according to the scores on the first dimension. Optionally, you can specify the maximum dimension number for which permuted tables will be produced. A permuted table for each dimension from 1 to the number specified is produced.

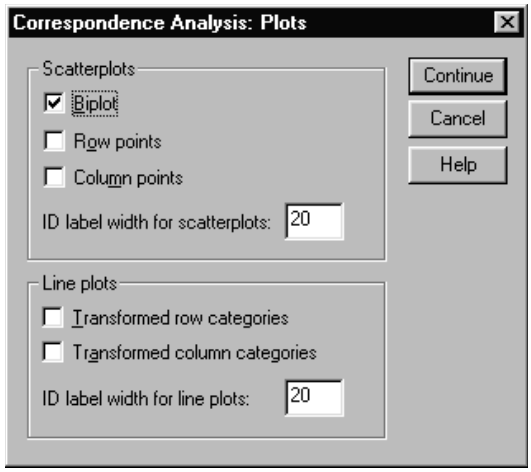
Confidence Statistics for Row points. Includes standard deviation and correlations for all nonsupplementary row points.

Confidence Statistics for Column points. Includes standard deviation and correlations for all nonsupplementary column points.

Correspondence Analysis Plots

The Plots dialog box allows you to specify which plots are produced.

Figure 5.6 Correspondence Analysis Plots dialog box



Scatterplots. Produces a matrix of all pairwise plots of the dimensions. Available scatterplots include:

- **Biplot.** Produces a matrix of joint plots of the row and column points. If principal normalization is selected, the biplot is not available.
- **Row points.** Produces a matrix of plots of the row points.
- **Column points.** Produces a matrix of plots of the column points.

Optionally, you can specify how many value label characters to use when labeling the points. This value must be a non-negative integer less than or equal to 20.

Line plots. Produces a plot for every dimension of the selected variable. Available line plots include:

- **Transformed row categories.** Produces a plot of the original row category values against their corresponding row scores.
- **Transformed column categories.** Produces a plot of the original column category values against their corresponding column scores.

Optionally, you can specify how many value label characters to use when labeling the category axis. This value must be a non-negative integer less than or equal to 20.

CORRESPONDENCE Command Additional Features

You can customize your correspondence analysis if you paste your selections into a syntax window and edit the resulting CORRESPONDENCE command syntax. SPSS command language also allows you to:

- Specify table data as input instead of using casewise data (using the TABLE = ALL subcommand).
- Specify the number of value-label characters used to label points for each type of scatterplot matrix or biplot matrix (with the PLOT subcommand).
- Specify the number of value-label characters used to label points for each type of line plot (with the PLOT subcommand).
- Write a matrix of row and column scores to an SPSS matrix data file (with the OUTFILE subcommand).
- Write a matrix of confidence statistics (variances and covariances) for the singular values and the scores to an SPSS matrix data file (with the OUTFILE subcommand).
- Specify multiple sets of categories to be equal (with the EQUAL subcommand).

6 Homogeneity Analysis (HOMALS)

Homogeneity analysis quantifies nominal (categorical) data by assigning numerical values to the cases (objects) and categories. Homogeneity analysis is also known by the acronym HOMALS, for *hom*ogeneity analysis by means of *a*lternating *l*east *s*quares.

The goal of HOMALS is to describe the relationships between two or more nominal variables in a low-dimensional space containing the variable categories as well as the objects in those categories. Objects within the same category are plotted close to each other, whereas objects in different categories are plotted far apart. Each object is as close as possible to the category points for categories that contain that object.

Homogeneity analysis is similar to correspondence analysis but is not limited to two variables. As a result, homogeneity analysis is also known in the literature as multiple correspondence analysis. Homogeneity analysis can also be viewed as a principal components analysis of nominal data.

Homogeneity analysis is preferred over standard principal components analysis when linear relationships between the variables may not hold or when variables are measured at a nominal level. Moreover, output interpretation is more straightforward in HOMALS than in other categorical techniques, such as crosstabulation tables and loglinear modeling. Because variable categories are quantified, techniques that require numerical data can be applied to the quantifications in subsequent analyses.

Example. Homogeneity analysis could be used to graphically display the relationship between job category, minority classification, and gender. You might find that minority classification and gender discriminate between people, but that job category does not. You might also find that the Latino and African-American categories are similar to each other.

Statistics and plots. Frequencies, eigenvalues, iteration history, object scores, category quantifications, discrimination measures, object scores plots, category quantifications plots, discrimination measures plots.

Data. All variables are categorical (nominal optimal scaling level). Use integers to code the categories. To minimize output, use consecutive integers beginning with 1 to code each variable.

Assumptions. All variables in the analysis have category quantifications that can be different for each dimension (multiple nominal). Only one set of variables will be used in the analysis. The maximum number of dimensions used in the procedure is either the

total number of categories minus the number of variables with no missing data, or the number of cases minus 1, whichever is smaller. For example, if one variable has five categories and the other has four (with no missing data), the maximum number of dimensions is seven ((5 + 4) – 2). If you specify a number greater than the maximum, the maximum value is used.

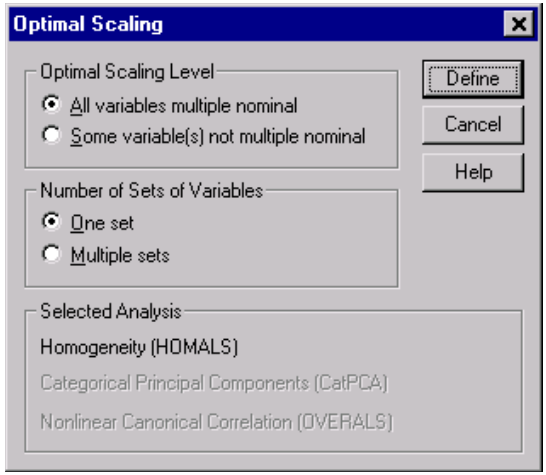
Related procedures. For two variables, homogeneity analysis is analogous to correspondence analysis. If you believe that variables possess ordinal or numerical properties, principal components with optimal scaling should be used. If you are interested in sets of variables, nonlinear canonical correlation analysis should be used.

To Obtain a Homogeneity Analysis

- From the menus choose:

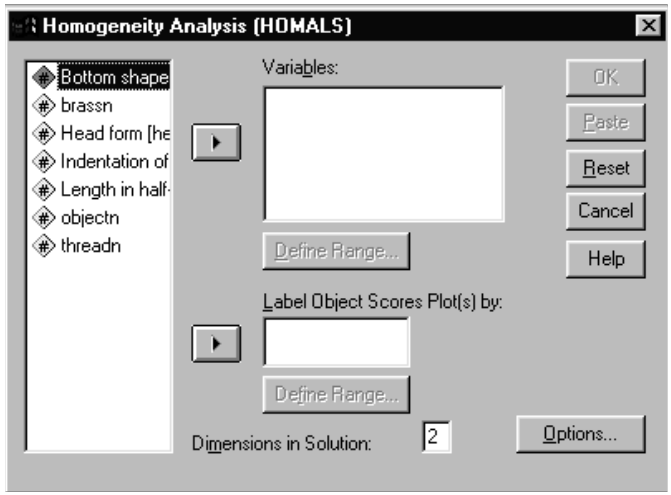
  Analyze
  Data Reduction
  Optimal Scaling...

Figure 6.1 Optimal Scaling dialog box



- In the Optimal Scaling dialog box, select All variables multiple nominal.
- Select One set.
- Click Define.

Figure 6.2 Homogeneity Analysis (HOMALS) dialog box



- ▶ Select two or more variables.
- ▶ Define the ranges for the variables.
- ▶ Click OK.

Optionally, you can:

- Select one or more variables to provide point labels for object scores plots. Each variable produces a separate plot, with the points labeled by the values of that variable. You must define a range for each of these plot label variables. Using the dialog box, a single variable cannot be used both in the analysis and as a labeling variable. If labeling the object scores plot with a variable used in the analysis is desired, use the Compute facility on the Transform menu to create a copy of that variable. Use the new variable to label the plot. Alternatively, command syntax can be used.
- Specify the number of dimensions you want in the solution. In general, choose as few dimensions as needed to explain most of the variation. If the analysis involves more than two dimensions, SPSS produces three-dimensional plots of the first three dimensions. Other dimensions can be displayed by editing the chart.

Define Range in Homogeneity Analysis

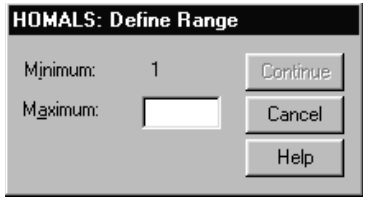
You must define a range for each variable. The maximum value specified must be an integer. Fractional data values are truncated in the analysis. A category value that is outside of the specified range is ignored in the analysis. To minimize output, use the Automatic Recode facility on the Transform menu to create consecutive categories beginning with 1.

You must also define a range for each variable used to label the object scores plots. However, labels for categories with data values outside of the defined range for the variable do appear on the plots.

To Define an Optimal Scaling Range in Homogeneity Analysis

- Select one or more variables in the Variables list in the Homogeneity Analysis (HOMALS) dialog box.
- Click Define Range.

Figure 6.3 HOMALS Define Range dialog box

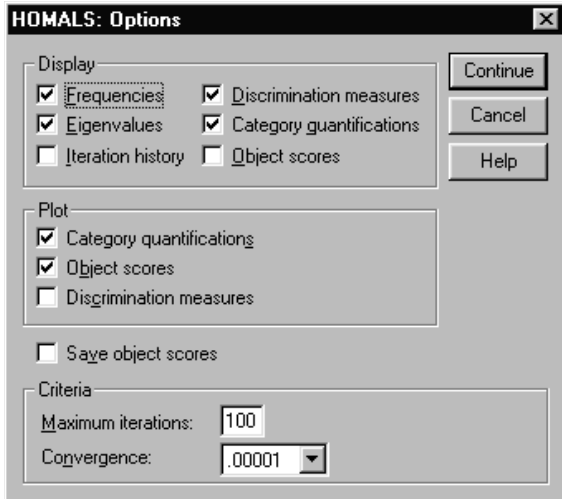


- Enter the maximum value for the variable. A minimum value of 1 is displayed. This minimum value cannot be changed.
- Click Continue.

Homogeneity Analysis Options

The Options dialog box allows you to select optional statistics and plots, save object scores as new variables in the working data file, and specify iteration and convergence criteria.

Figure 6.4 HOMALS Options dialog box



Display. These options control what statistics are included in the output. Available statistics include marginal frequencies, eigenvalues, iteration history, discrimination measures, category quantifications, and object scores.

Plot. These options produce plots of category quantifications, object scores, and discrimination measures.

Save object scores. You can save the object scores as new variables in the working data file. Object scores are saved for the number of dimensions specified in the main dialog box.

Criteria. You can specify the maximum number of iterations the homogeneity analysis can go through in its computations. You can also select a convergence criterion value. The homogeneity analysis stops iterating if the difference in total fit between the last two iterations is less than the convergence value, or if the maximum number of iterations is reached.

HOMALS Command Additional Features

You can customize your homogeneity analysis if you paste your selections into a syntax window and edit the resulting HOMALS command syntax. SPSS command language also allows you to:

- Specify the dimension pairs to be plotted, rather than plotting all extracted dimensions (using the NDIM keyword on the PLOT subcommand).
- Specify the number of value label characters used to label points on the plots (with the PLOT subcommand).
- Designate more than five variables as labeling variables for object scores plots (with the PLOT subcommand).
- Select variables used in the analysis as labeling variables for the object scores plots (with the PLOT subcommand).
- Select variables to provide point labels for the quantification score plot (with the PLOT subcommand).
- Specify the number of cases to be included in the analysis if you do not want to use all cases in the working data file (with the NOBSERVATIONS subcommand).
- Specify rootnames for variables created by saving object scores (with the SAVE subcommand).
- Specify the number of dimensions to be saved, rather than saving all extracted dimensions (with the SAVE subcommand).
- Write category quantifications to a matrix file (using the MATRIX subcommand).
- Produce low-resolution plots that may be easier to read than the usual high-resolution plots (using the SET command).

7 Multidimensional Scaling (PROXSCAL)

Multidimensional scaling attempts to find the structure in a set of proximity measures between objects. This is accomplished by assigning observations to specific locations in a conceptual low-dimensional space such that the distances between points in the space match the given (dis)similarities as closely as possible. The result is a least-squares representation of the objects in that low-dimensional space, which, in many cases, will help you to further understand your data.

Example. Multidimensional scaling can be very useful in determining perceptual relationships. For example, when considering your product image, you can conduct a survey in order to obtain a data set that describes the perceived similarity (or proximity) of your product to those of your competitors. Using these proximities and independent variables (such as price), you can try to determine which variables are important to how people view these products, and adjust your image accordingly.

Statistics and plots. Iteration history, Stress measures, Stress decomposition, coordinates of the common space, object distances within the final configuration, individual space weights, individual spaces, transformed proximities, transformed independent variables, Stress plots, common space scatterplots, individual space weight scatterplots, individual spaces scatterplots, transformation plots, Shepard residual plots, and independent variables transformation plots.

Data. Data can be supplied in the form of proximity matrices or variables that are converted into proximity matrices. The matrices may be formatted in columns or across columns. The proximities may be treated on the ratio, interval, ordinal, or spline scaling levels.

Assumptions. At least three variables must be specified. The number of dimensions may not exceed the number of objects minus one. Dimensionality reduction is omitted if combined with multiple random starts. If only one source is specified, all models are equivalent to the identity model; therefore, the analysis defaults to the identity model.

Related procedures. Scaling all variables at the numerical level corresponds to standard multidimensional scaling analysis.

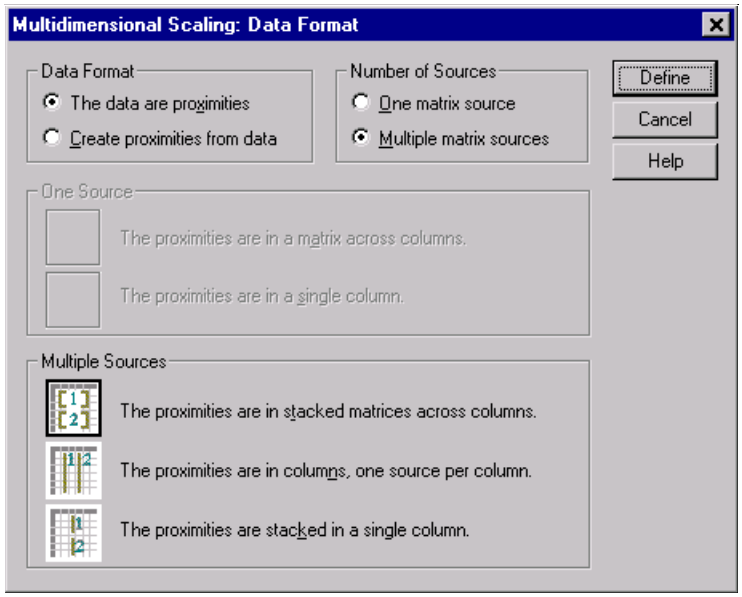
To Obtain a Multidimensional Scaling

▶ From the menus choose:

Analyze
 Scale
  Multidimensional Scaling (PROXSCAL)...

This opens the Data Format dialog box.

Figure 7.1 Multidimensional Scaling Data Format dialog box



You must specify the format of your data:

Data Format. Specify whether your data consist of proximity measures or you want to create proximities from the data.

Number of Sources. If your data are proximities, specify whether you have a single source or multiple sources of proximity measures.

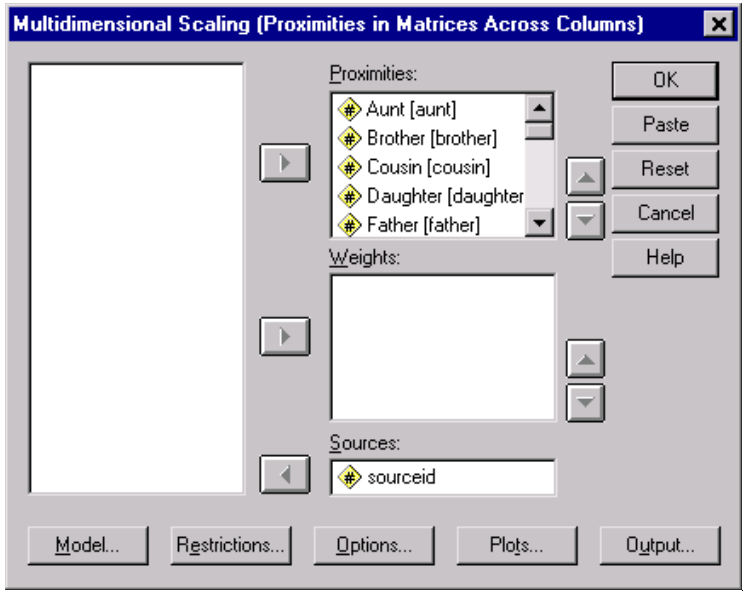
One Source. If there is one source of proximities, specify whether your data set is formatted with the proximities in a matrix across the columns or in a single column with two separate variables to identify the row and column of each proximity.

Multiple Sources. If there are multiple sources of proximities, specify whether the data set is formatted with the proximities in stacked matrices across columns, in multiple columns with one source per column, or in a single column.

Proximities in Matrices across Columns

If you select the proximities in matrices data model for either one source or multiple sources in the Data Format dialog box, the main dialog box will appear as follows:

Figure 7.2 Proximities in Matrices across Columns dialog box



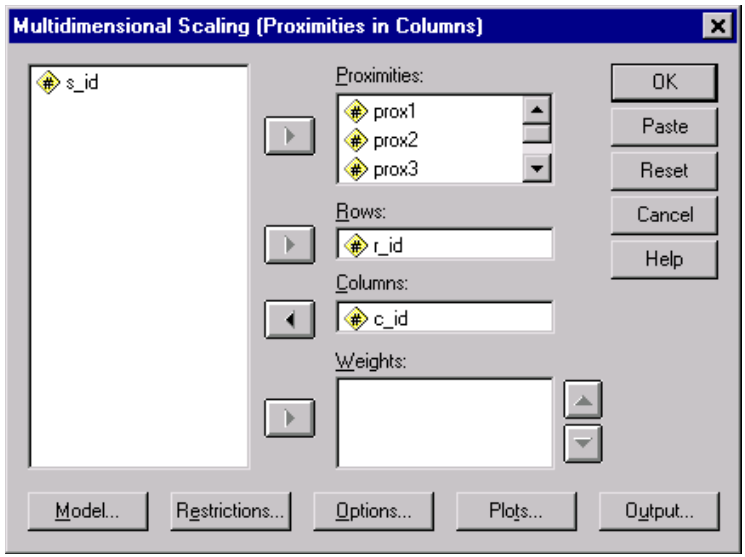
- Select three or more proximities variables. Please be sure that the order of the variables in the list matches the order of the columns of the proximities.
- Optionally, select a number of weights variables equal to the number of proximities variables. Again, be sure that the order of the weights matches the order of the proximities they weight.
- If there are multiple sources, optionally, select a sources variable. The number of cases in each proximities variable should equal the number of proximities variables times the number of sources.

Additionally, you can define a model for the multidimensional scaling, place restrictions on the common space, set convergence criteria, specify the initial configuration to be used, and choose plots and output.

Proximities in Columns

If you select the multiple columns model for multiple sources in the Data Format dialog box, the main dialog box will appear as follows:

Figure 7.3 Proximities in Columns dialog box



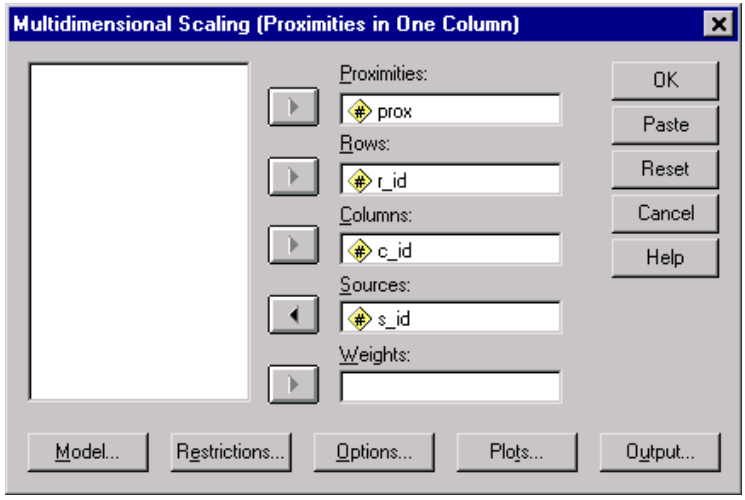
- Select two or more proximities variables. Each variable is assumed to be a matrix of proximities from a separate source.
- Select a rows variable. This defines the row locations for the proximities in each proximities variable.
- Select a columns variable. This defines the column locations for the proximities in each proximities variable. Cells of the proximity matrix that are not given a row/column designation are treated as missing.
- Optionally, select a number of weights variables equal to the number of proximities variables.

Additionally, you can define a model for the multidimensional scaling, place restrictions on the common space, set convergence criteria, specify the initial configuration to be used, and choose plots and output.

Proximities in One Column

If you select the one column model for either one source or multiple sources in the Data Format dialog box, the main dialog box will appear as follows:

Figure 7.4 Proximities in One Column dialog box



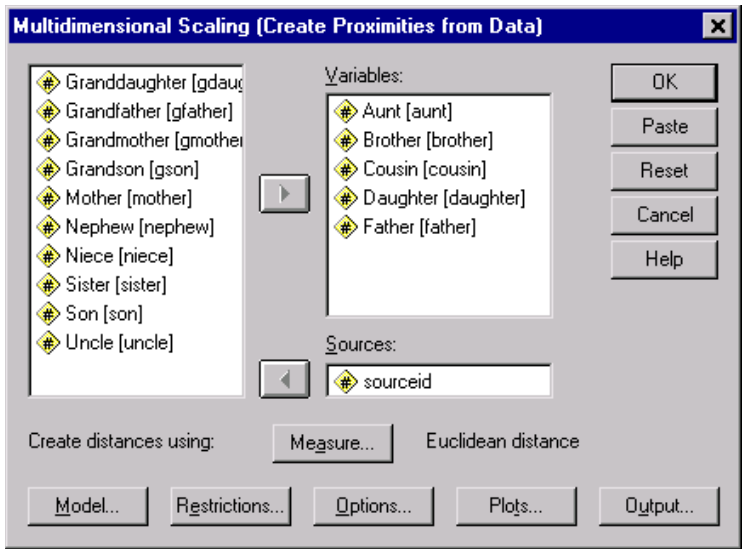
- Select a proximities variable. It is assumed to be one or more matrices of proximities.
- Select a rows variable. This defines the row locations for the proximities in the proximities variable.
- Select a columns variable. This defines the column locations for the proximities in the proximities variable.
- If there are multiple sources, select a sources variable. For each source, cells of the proximity matrix that are not given a row/column designation are treated as missing.
- Optionally, select a weights variable.

Additionally, you can define a model for the multidimensional scaling, place restrictions on the common space, set convergence criteria, specify the initial configuration to be used, and choose plots and output.

Create Proximities from Data

If you choose to create proximities from the data in the Data Format dialog box, the main dialog box will appear as follows:

Figure 7.5 Create Proximities from Data dialog box

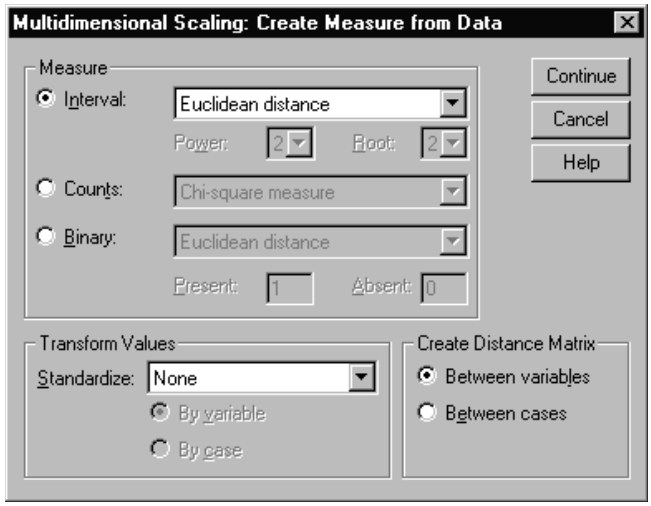


- If you create distances between variables (see the Measures dialog box), select at least three variables. These will be used to create the proximity matrix (or matrices, if there are multiple sources). If you create distances between cases, only one variable is needed.
- If there are multiple sources, select a sources variable.
- Optionally, choose a measure for creating proximities.

Additionally, you can define a model for the multidimensional scaling, place restrictions on the common space, set convergence criteria, specify the initial configuration to be used, and choose plots and output.

Measures Dialog Box

Figure 7.6 Multidimensional Scaling Create Measure from Data dialog box



Multidimensional scaling uses dissimilarity data to create a scaling solution. If your data are multivariate data (values of measured variables), you must create dissimilarity data in order to compute a multidimensional scaling solution. You can specify the details of creating dissimilarity measures from your data.

Measure. Allows you to specify the dissimilarity measure for your analysis. Select one alternative from the Measure group corresponding to your type of data, and then select one of the measures from the drop-down list corresponding to that type of measure. Available alternatives are:

- **Interval**. Euclidean distance, squared Euclidean distance, Chebychev, Block, Minkowski, or Customized.
- **Counts**. Chi-square measure or Phi-square measure.
- **Binary**. Euclidean distance, Squared Euclidean distance, Size difference, Pattern difference, Variance, or Lance and Williams.

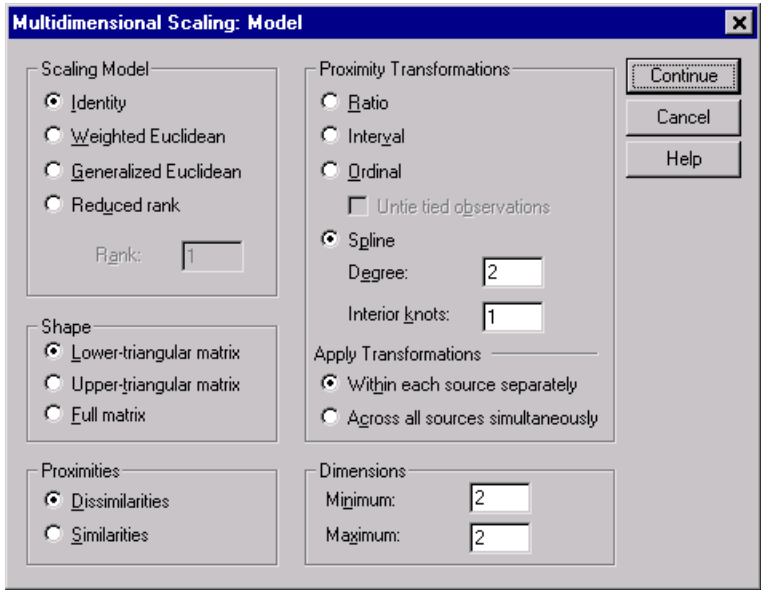
Create Distance Matrix. Allows you to choose the unit of analysis. Alternatives are Between variables or Between cases.

Transform Values. In certain cases, such as when variables are measured on very different scales, you may want to standardize values before computing proximities (not applicable to binary data). Select a standardization method from the Standardize drop-down list (if no standardization is required, select None).

Define a Multidimensional Scaling Model

The Model dialog box allows you to specify a scaling model, its minimum and maximum number of dimensions, the structure of the proximity matrix, the transformation to use on the proximities, and whether proximities are transformed within each source separately, or unconditionally on the source.

Figure 7.7 Multidimensional Scaling Model dialog box



Scaling Model. Choose from the following alternatives.

- **Identity.** All sources have the same configuration.
- **Weighted Euclidean.** This model is an individual differences model. Each source has an individual space in which every dimension of the common space is weighted differentially.
- **Generalized Euclidean.** This model is an individual differences model. Each source has an individual space that is equal to a rotation of the common space, followed by a differential weighting of the dimensions.
- **Reduced rank.** This is a Generalized Euclidean model for which you can specify the rank of the individual space. You must specify a rank that is greater than or equal to 1 and less than the maximum number of dimensions.

Shape. Specify whether the proximities should be taken from the lower-triangular part or the upper-triangular part of the proximity matrix. You may specify that the full matrix be used, in which case the weighted sum of the upper-triangular part and the lower-triangular part will be analyzed. In any case, the complete matrix should be specified, including the diagonal, though only the specified parts will be used.

Proximities. Specify whether your proximity matrix contains measures of similarity or dissimilarity.

Proximity Transformations. Choose from the following alternatives.

- **Ratio.** The transformed proximities are proportional to the original proximities. This is only allowed for positively valued proximities.
- **Interval.** The transformed proximities are proportional to the original proximities, plus an intercept term. The intercept assures all transformed proximities to be positive.
- **Ordinal.** The transformed proximities have the same order as the original proximities. You may specify whether tied proximities should be kept tied or allowed to become untied.
- **Spline.** The transformed proximities are a smooth nondecreasing piecewise polynomial transformation of the original proximities. You may specify the degree of the polynomial and the number of interior knots.

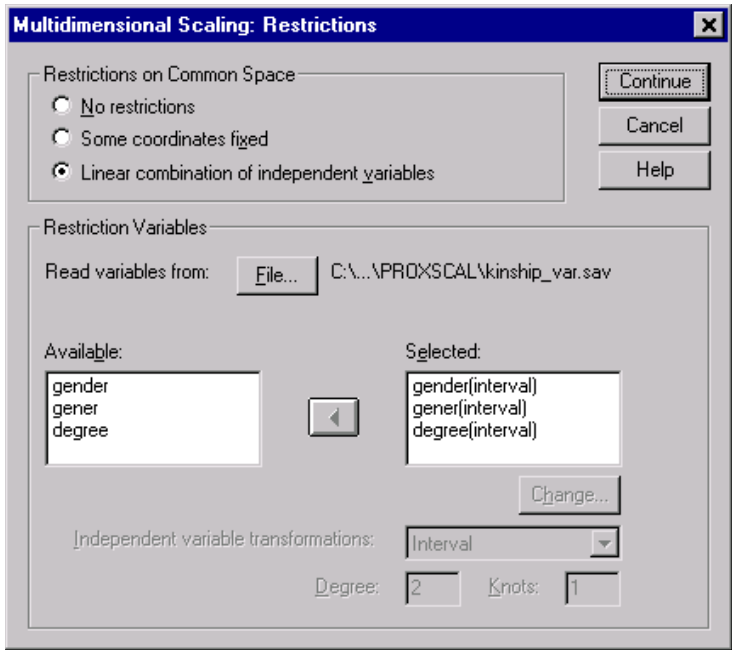
Apply Transformations. Specify whether only proximities within each source are compared with each other, or the comparisons are unconditional on the source.

Dimensions. By default, a solution is computed in two dimensions (Minimum=2, Maximum=2). You may choose an integer minimum and maximum from 1 to the number of objects minus 1, so long as the minimum is less than or equal to the maximum. The procedure computes a solution in the maximum dimensions and then reduces the dimensionality in steps, until the lowest is reached.

Multidimensional Scaling Restrictions

The Restrictions dialog box allows you to place restrictions on the common space.

Figure 7.8 Multidimensional Scaling Restrictions dialog box



Restrictions on Common Space. Specify the type of restriction desired.

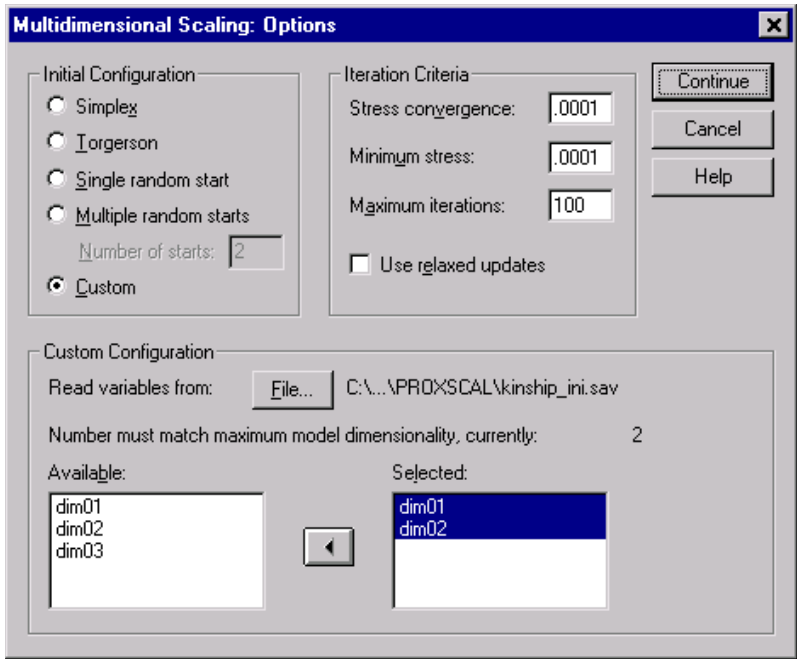
- **No restrictions.** No restrictions are placed on the common space.
- **Some coordinates fixed.** The first variable selected contains the coordinates of the objects on the first dimension. The second variable corresponds to coordinates on the second dimension, and so on. A missing value indicates that a coordinate on a dimension is free. The number of variables selected must equal the maximum number of dimensions requested.
- **Linear combination of independent variables.** The common space is restricted to be a linear combination of the variables selected.

Restriction Variables. Select the variables that define the restrictions on the common space. If you specified a linear combination, you may specify an interval, nominal, ordinal, or spline transformation for the restriction variables. In either case, the number of cases for each variable must equal the number of objects.

Multidimensional Scaling Options

The Options dialog box allows you to select the kind of initial configuration, specify iteration and convergence criteria, and select standard or relaxed updates.

Figure 7.9 Multidimensional Scaling Options dialog box



Initial Configuration. Choose one of the following alternatives.

- **Simplex.** Objects are placed at the same distance from each other in the maximum dimension. One iteration is taken to improve this high-dimensional configuration, followed by a dimensionality reduction operation to obtain an initial configuration that has the maximum number of dimensions that you specified in the Model dialog box.
- **Torgerson.** A classical scaling solution is used as the initial configuration.
- **Single random start.** A configuration is chosen at random.
- **Multiple random starts.** Several configurations are chosen at random, and the solution with the lowest normalized raw Stress is shown.
- **Custom.** You may select variables that contain the coordinates of your own initial configuration. The number of variables selected should equal the maximum number of dimensions specified, with the first variable corresponding to coordinates on dimension 1, the second variable corresponding to coordinates on dimension 2, and so on. The number of cases in each variable should equal the number of objects.

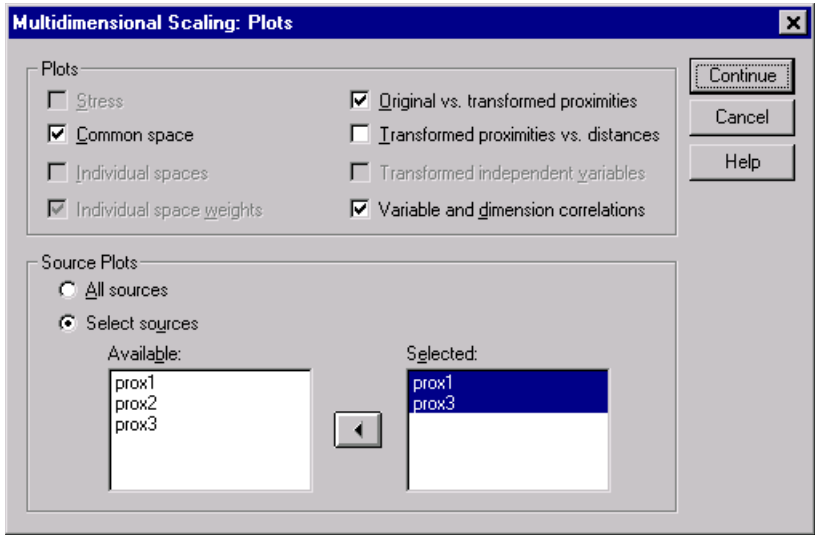
Iteration Criteria. Specify the iteration criteria values.

- **Stress convergence.** The algorithm will stop iterating when the difference in consecutive normalized raw Stress values is less than the number specified here, which must lie between 0.0 and 1.0.
- **Minimum stress.** The algorithm will stop when the normalized raw Stress falls below the number specified here, which must lie between 0.0 and 1.0.
- **Maximum iterations.** The algorithm will perform the number of iterations specified here, unless one of the above criteria is satisfied first.
- **Use relaxed updates.** Relaxed updates will speed up the algorithm; these cannot be used with models other than the identity model, or with restrictions.

Multidimensional Scaling Plots, Version 1

The Plots dialog box allows you to specify which plots will be produced. If you have the Proximities in Columns data format, the following Plots dialog box is displayed. For Individual space weights, Original vs. transformed proximities, and Transformed proximities vs. distances plots, you may specify the sources for which the plots should be produced. The list of available sources is the list of proximities variables in the main dialog box.

Figure 7.10 Multidimensional Scaling Plots dialog box, version 1



Stress. A plot is produced of normalized raw Stress versus dimensions. This plot is produced only if the maximum number of dimensions is larger than the minimum number of dimensions.

Common space. A scatterplot matrix of coordinates of the common space is displayed.

Individual spaces. For each source, the coordinates of the individual spaces are displayed in scatterplot matrices. This is only possible if one of the individual differences models is specified in the Model dialog box.

Individual space weights. A scatterplot is produced of the individual space weights. This is only possible if one of the individual differences models is specified in the Model dialog box. For the weighted Euclidean model, the weights are printed in plots with one dimension on each axis. For the generalized Euclidean model, one plot is produced per dimension, indicating both rotation and weighting of that dimension. The reduced rank model produces the same plot as the generalized Euclidean model, but reduces the number of dimensions for the individual spaces.

Original vs. transformed proximities. Plots are produced of the original proximities versus the transformed proximities.

Transformed proximities vs. distances. The transformed proximities versus the distances are plotted.

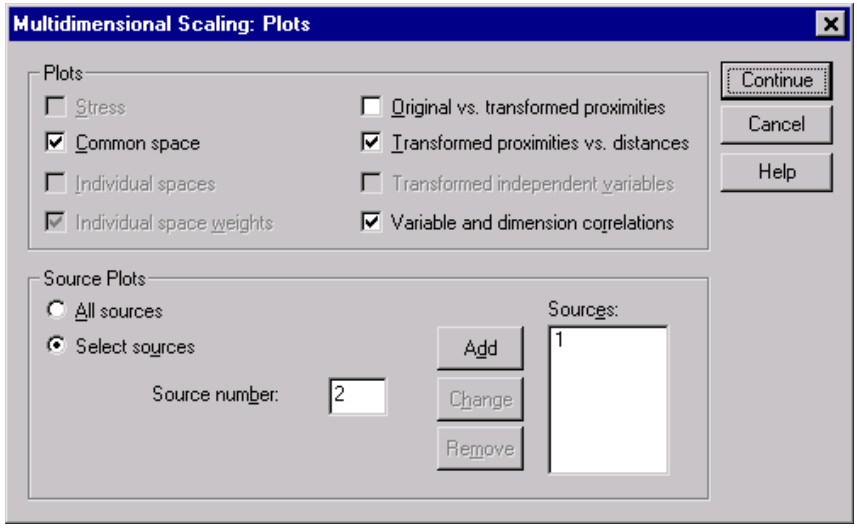
Transformed independent variables. Transformation plots are produced for the independent variables.

Variable and dimension correlations. A plot of correlations between the independent variables and the dimensions of the common space is displayed.

Multidimensional Scaling Plots, Version 2

The Plots dialog box allows you to specify which plots will be produced. If your data format is anything other than Proximities in Columns, the following Plots dialog box is displayed. For Individual spaces weights, Original vs. transformed proximities, and Transformed proximities vs. distances plots, you may specify the sources for which the plots should be produced. The source numbers entered must be values of the sources variable specified in the main dialog box, and range from 1 to the number of sources.

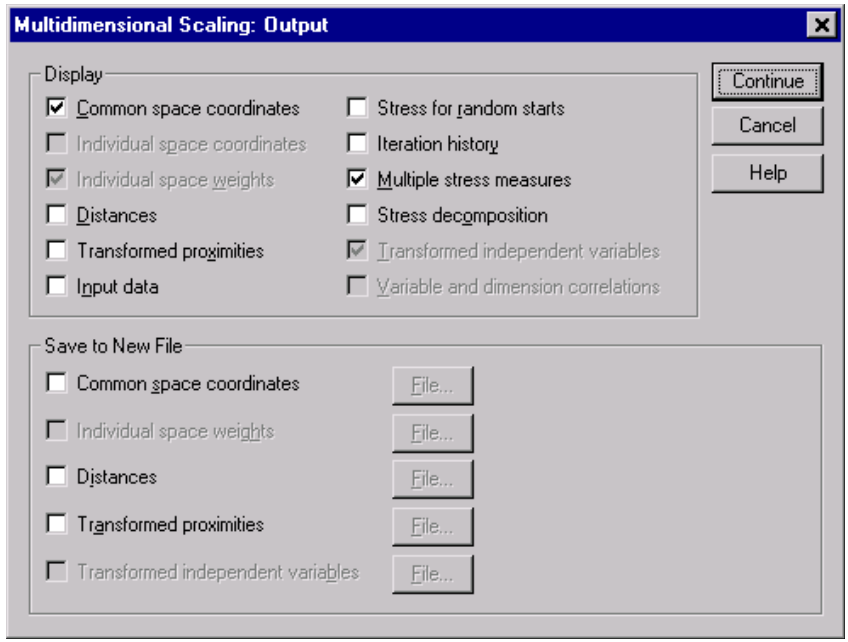
Figure 7.11 Multidimensional Scaling Plots dialog box, version 2



Multidimensional Scaling Output

The Output dialog box allows you to control the amount of displayed output and save some of it to separate files.

Figure 7.12 Multidimensional Scaling Output dialog box



Display. Select one or more of the following for display.

- **Common space coordinates.** Displays the coordinates of the common space.
- **Individual space coordinates.** The coordinates of the individual spaces are displayed, only if the model is not the identity model.
- **Individual space weights.** Displays the individual space weights, only if one of the individual differences models is specified. Depending on the model, the space weights are decomposed in rotation weights and dimension weights, which are also displayed.
- **Distances.** Displays the distances between the objects in the configuration.
- **Transformed proximities.** Displays the transformed proximities between the objects in the configuration.
- **Input data.** Includes the original proximities, and, if present, the data weights, the initial configuration, and the fixed coordinates or the independent variables.
- **Stress for random starts.** Displays the random number seed and normalized raw Stress value of each random start.
- **Iteration history.** Displays the history of iterations of the main algorithm.

- **Multiple stress measures.** Displays different Stress values. The table contains values for normalized raw Stress, Stress-I, Stress-II, S-Stress, Dispersion Accounted For (DAF), and Tucker's Coefficient of Congruence.
- **Stress decomposition.** Displays an objects and sources decomposition of the final normalized raw Stress, including the average per object and the average per source.
- **Transformed independent variables.** If a linear combination of independent variables restriction was selected, the transformed independent variables and the corresponding regression weights are displayed.
- **Variable and dimension correlations.** If a linear combination restriction was selected, the correlations between the independent variables and the dimensions of the common space are displayed.

Save to New File. You can save the common space coordinates, individual space weights, distances, transformed proximities, and transformed independent variables to separate SPSS data files.

PROXSCAL Command Additional Features

You can customize your multidimensional scaling of proximities analysis if you paste your selections into a syntax window and edit the resulting PROXSCAL command syntax. SPSS command language also allows you to:

- Specify separate variable lists for transformations and residuals plots (with the PLOT subcommand).
- Specify separate source lists for individual space weights, transformations, and residuals plots (with the PLOT subcommand).
- Specify a subset of the independent variables transformation plots to be displayed (with the PLOT subcommand).

8 Categorical Regression Examples

The goal of categorical regression with optimal scaling is to describe the relationship between a response and a set of predictors. By quantifying this relationship, values of the response can be predicted for any combination of predictors.

In this chapter, two examples serve to illustrate the analyses involved in optimal scaling regression. The first example uses a small data set to illustrate the basic concepts. The second example uses a much larger set of variables and observations in a practical example.

Example 1: Carpet Cleaner Data

In a popular example by Green and Wind (1973), a company interested in marketing a new carpet cleaner wants to examine the influence of five factors on consumer preference—package design, brand name, price, a *Good Housekeeping* seal, and a money-back guarantee. There are three factor levels for package design, each one differing in the location of the applicator brush; three brand names (*K2R*, *Glory*, and *Bissell*); three price levels; and two levels (either no or yes) for each of the last two factors. Table 8.1 displays the variables used in the carpet-cleaner study, with their variable labels and values.

Table 8.1 Explanatory variables in the carpet-cleaner study

	Variable label	Value labels
package	Package design	A*, B*, C*
brand	Brand name	K2R, Glory, Bissell
price	Price	$1.19, $1.39, $1.59
seal	*Good Housekeeping* seal	No, yes
money	Money-back guarantee	No, yes

Ten consumers rank 22 profiles defined by these factors. The variable *pref* contains the rank of the average rankings for each profile. Low rankings correspond to high preference. This variable reflects an overall measure of preference for each profile. Using categorical regression, you will explore how the five factors in Table 8.1 are related to preference. This data set can be found in *carpet.sav*.

A Standard Linear Regression Analysis

To produce standard linear regression output, from the menus choose:

Analyze
  Regression
    Linear...

▸ Dependent: *pref*
▸ Independent(s): *package*, *brand*, *price*, *seal*, *money*

Statistics...
  ☐ Descriptives (deselect)

Save...
  Residuals
  ☑ Standardized

The standard approach for describing the relationships in this problem is linear regression. The most common measure of how well a regression model fits the data is R^2. This statistic represents how much of the variance in the response is explained by the weighted combination of predictors. The closer R^2 is to 1, the better the model fits. Regressing *pref* on the five predictors results in an R^2 of 0.707, indicating that approximately 71% of the variance in the preference rankings is explained by the predictor variables in the linear regression.

Figure 8.1 Model summary for standard linear regression

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.841	.707	.615	3.9981

The standardized coefficients are shown in Figure 8.2. The sign of the coefficient indicates whether the predicted response increases or decreases when the predictor increases, all other predictors being constant. For categorical data, the category coding determines the meaning of an increase in a predictor. For instance, an increase in *money*, *package*, or *seal* will result in a decrease in predicted preference ranking. *money* is coded 1 for *no money-back guarantee* and 2 for *money-back guarantee*. An increase in *money* corresponds to the addition of a money-back guarantee. Thus, adding a money-back guarantee reduces the predicted preference ranking, which corresponds to an increased predicted preference.

Figure 8.2 Regression coefficients

Model		Standardized Coefficients	t	Sig.
		Beta		
1	(Constant)		4.352	.000
	Package design	-.560	-4.015	.001
	Brand name	.056	.407	.689
	Price	.366	2.681	.016
	Good Housekeeping seal	-.330	-2.423	.028
	Money-back guarantee	-.197	-1.447	.167

The value of the coefficient reflects the amount of change in the predicted preference ranking. Using standardized coefficients, interpretations are based on the standard deviations of the variables. Each coefficient indicates the number of standard deviations that the predicted response changes for a one standard deviation change in a predictor, all other predictors remaining constant. For example, a one standard deviation change in *brand* yields an increase in predicted preference of 0.056 standard deviations. The standard deviation of *pref* is 6.44, so *pref* increases by $0.056 \times 6.44 = 0.361$. Changes in *package* yield the greatest changes in predicted preference.

A regression analysis should always include an examination of the residuals. To produce residual plots, from the menus choose:

Graphs
 Scatter...

Select *Simple*. Click *Define*.

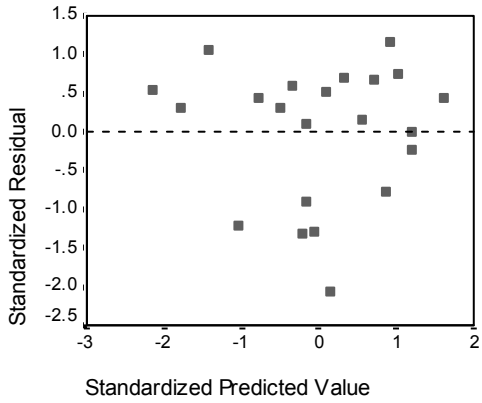
- Y Axis: *zre_1*
- X Axis: *zpr_1*

Then, recall the Simple Scatterplot dialog box and click *Reset* to clear the previous selections.

- Y Axis: *zre_1*
- X Axis: *package*

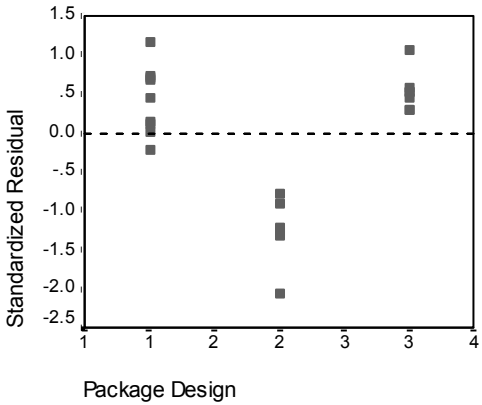
The standardized residuals are plotted against the standardized predicted values in Figure 8.3. No patterns should be present if the model fits well. Here you see a U-shape in which both low and high standardized predicted values have positive residuals. Standardized predicted values near 0 tend to have negative residuals.

Figure 8.3 Residuals versus predicted values



This shape is more pronounced in the plot of the standardized residuals against *package* in Figure 8.4. Every residual for Design B* is negative, whereas all but one of the residuals is positive for the other two designs. Because the regression model fits one parameter for each variable, the relationship cannot be captured by the standard approach.

Figure 8.4 Residuals versus package



A Categorical Regression Analysis

The categorical nature of the variables and the nonlinear relationship between *pref* and *package* suggest that regression on optimal scores may perform better than standard regression. The U-shape of Figure 8.4 indicates that a nominal treatment of *package* should be used. All other predictors will be treated at the numerical scaling level.

The response variable warrants special consideration. You want to predict the values of *pref*. Thus, recovering as many properties of its categories as possible in the quantifications is desirable. Using an ordinal or nominal scaling level ignores the differences between the response categories. However, linearly transforming the response categories preserves category differences. Consequently, scaling the response numerically is generally preferred and will be employed here.

To produce the following categorical regression output, from the menus choose:

Analyze
  Regression
    Optimal Scaling...

▸ Dependent: *pref*
▸ Independent(s): *package*, *brand*, *price*, *seal*, *money*

Select *pref*. Click *Define Scale*.

  Optimal Scaling Level
  ⊙ Numeric

Select *package*. Click *Define Scale*.

  Optimal Scaling Level
  ⊙ Nominal

Select *brand*, *price*, *seal*, and *money*. Click *Define Scale*.

  Optimal Scaling Level
  ⊙ Numeric

Output...

  Display
  ☑ Correlations of original predictors
  ☑ Correlations of transformed predictors
  ☐ Frequencies (deselect)
  ☐ ANOVA table (deselect)

Save...

  Save to Working File
  ☑ Transformed variables
  ☑ Residuals

Plots...

  ▸ Transformation Plots: *package*, *price*

Intercorrelations

The intercorrelations among the predictors are useful for identifying multicollinearity in the regression. Variables that are highly correlated will lead to unstable regression estimates. However, due to their high correlation, omitting one of them from the model only minimally affects prediction. The variance in the response that can be explained by the omitted variable is still explained by the remaining correlated variable. However, zero-order correlations are sensitive to outliers and also cannot identify multicollinearity due to a high correlation between a predictor and a combination of other predictors.

Figure 8.5 and Figure 8.6 show the intercorrelations of the predictors for both the untransformed and transformed predictors. All values are near 0, indicating that multicollinearity between individual variables is not a concern.

Notice that the only correlations that change involve package. Because all other predictors are treated numerically, the differences between the categories and the order of the categories are preserved for these variables. Consequently, the correlations cannot change.

Figure 8.5 Original predictor correlations

	Package design	Brand name	Price	Good Housekeeping seal	Money-back guarantee
Package design	1.000	-.189	-.126	.081	.066
Brand name	-.189	1.000	.065	-.042	-.034
Price	-.126	.065	1.000	.000	.000
Good Housekeeping seal	.081	-.042	.000	1.000	-.039
Money-back guarantee	.066	-.034	.000	-.039	1.000

Figure 8.6 Transformed predictor correlations

	Package design	Brand name	Price	Good Housekeeping seal	Money-back guarantee
Package design	1.000	-.156	-.089	.032	.102
Brand name	-.156	1.000	.065	-.042	-.034
Price	-.089	.065	1.000	.000	.000
Good Housekeeping seal	.032	-.042	.000	1.000	-.039
Money-back guarantee	.102	-.034	.000	-.039	1.000

Model Fit and Coefficients

The Categorical Regression procedure yields an R^2 of 0.948, indicating that almost 95% of the variance in the transformed preference rankings is explained by the regression on the optimally transformed predictors. Transforming the predictors improves the fit over the standard approach.

Figure 8.7 Model summary for categorical regression

Multiple R	R Square	Adjusted R Square
.974	.948	.932

Figure 8.8 shows the standardized regression coefficients. Categorical regression standardizes the variables, so only standardized coefficients are reported. These values are divided by their corresponding standard errors, yielding an F test for each variable. However, the test for each variable is contingent upon the other predictors being in the model. In other words, the test determines if omission of a predictor variable from the model with all other predictors present significantly worsens the predictive capabilities of the model. These values should not be used to omit several variables at one time for a subsequent model. Moreover, alternating least squares optimizes the quantifications, implying that these tests must be interpreted conservatively.

Figure 8.8 Standardized coefficients for transformed predictors

	Standardized Coefficients		
	Beta	Std. Error	F
Package design	-.748	.058	165.495
Brand name	4.530E-02	.058	.614
Price	.371	.057	41.986
Good Housekeeping seal	-.350	.057	37.702
Money-back guarantee	-.159	.057	7.669

The largest coefficient occurs for *package*. A one standard deviation increase in *package* yields a 0.748 standard deviation decrease in predicted preference ranking. However, *package* is treated nominally, so an increase in the quantifications need not correspond to an increase in the original category codes.

Standardized coefficients are often interpreted as reflecting the importance of each predictor. However, regression coefficients cannot fully describe the impact of a predictor or the relationships between the predictors. Alternative statistics must be used in conjunction with the standardized coefficients to fully explore predictor effects.

Correlational Analyses

To interpret the contributions of the predictors to the regression, it is not sufficient to only inspect the regression coefficients. In addition, the correlations, partial correlations, and part correlations should be inspected. Figure 8.9 contains these correlational measures for each variable.

The zero-order correlation is the correlation between the transformed predictor and the transformed response. For this data, the largest correlation occurs for *package*. However, if you can explain some of the variation in either the predictor or the response, you will get a better representation of how well the predictor is doing.

Figure 8.9 Zero-order, part, and partial correlations (transformed variables)

	Correlations		
	Zero-Order	Partial	Part
Package design	-.816	-.955	-.733
Brand name	.206	.192	.045
Price	.441	.851	.369
Good Housekeeping seal	-.370	-.838	-.350
Money-back guarantee	-.223	-.569	-.158

Other variables in the model can confound the performance of a given predictor in predicting the response. The partial correlation coefficient removes the linear effects of other predictors from both the predictor and the response. This measure equals the correlation between the residuals from regressing the predictor on the other predictors and the residuals from regressing the response on the other predictors. The squared partial correlation corresponds to the proportion of the variance explained relative to the residual variance of the response remaining after removing the effects of the other variables. For example, in Figure 8.9, *package* has a partial correlation of –0.955. Removing the effects of the other variables, package explains $(-0.955)^2 = 0.91 = 91\%$ of the variation in the preference rankings. Both *price* and *seal* also explain a large portion of variance if the effects of the other variables are removed.

Figure 8.10 displays the partial correlations for the untransformed variables. All of the partial correlations increase when optimal scores are used. In the standard approach, *package* explained 50% of the variation in *pref* when other variable effects were removed from both. In contrast, *package* explains 91% of the variation if optimal scaling is used. Similar results occur for *price* and *seal*.

Figure 8.10 Zero-order, part, and partial correlations (untransformed variables)

Model		Correlations		
		Zero-order	Partial	Part
1	(Constant)			
	Package design	-.657	-.708	-.544
	Brand name	.206	.101	.055
	Price	.440	.557	.363
	Good Housekeeping seal	-.370	-.518	-.328
	Money-back guarantee	-.223	-.340	-.196

As an alternative to removing the effects of variables from both the response and a predictor, you can remove the effects from just the predictor. The correlation between the response and the residuals from regressing a predictor on the other predictors is the part correlation. Squaring this value yields a measure of the proportion of variance explained relative to the total variance of response. From Figure 8.9, if you remove the effects of *brand*, *seal*, *money*, and *price* from *package*, the remaining part of *package* explains $(-0.733)^2 = 0.54 = 54\%$ of the variation in preference rankings.

Importance

In addition to the regression coefficients and the correlations, Pratt's measure of relative importance (Pratt, 1987) aids in interpreting predictor contributions to the regression. Large individual importances relative to the other importances correspond to predictors that are crucial to the regression. Also, the presence of suppressor variables is signaled by a low importance for a variable that has a coefficient of similar size to the important predictors.

Figure 8.11 displays the importances for the carpet cleaner predictors. In contrast to the regression coefficients, this measure defines the importance of the predictors additively—that is, the importance of a set of predictors is the sum of the individual importances of the predictors. Pratt's measure equals the product of the regression coefficient and the zero-order correlation for a predictor. These products add to R^2, so they are divided by R^2, yielding a sum of one. The set of predictors *package* and *brand*, for example, have an importance of 0.654. The largest importance corresponds to *package*, with *package*, *price*, and *seal* accounting for 95% of the importance for this combination of predictors.

Multicollinearity

Large correlations between predictors will dramatically reduce a regression model's stability. Correlated predictors result in unstable parameter estimates. Tolerance reflects how much the independent variables are linearly related to one another. This measure is the proportion of a variable's variance *not* accounted for by other independent variables in the equation. If the other predictors can explain a large amount of a predictor's variance, that predictor is not needed in the model. A tolerance value near 1 indicates that the variable cannot be predicted very well from the other predictors. In contrast, a variable with a very low tolerance contributes little information to a model, and can cause computational problems. Moreover, large negative values of Pratt's importance measure indicate multicollinearity.

Figure 8.11 shows the tolerance for each predictor. All of these measures are very high. None of the predictors are predicted very well by the other predictors and multicollinearity is not present.

Figure 8.11 Predictor tolerances and importances

	Importance	Tolerance	
		After Transformation	Before Transformation
Package design	.644	.959	.942
Brand name	.010	.971	.961
Price	.172	.989	.982
Good Housekeeping seal	.137	.996	.991
Money-back guarantee	.037	.987	.993

Transformation Plots

Plotting the original category values against their corresponding quantifications can reveal trends that might not be noticed in a list of the quantifications. Such plots are commonly referred to as transformation plots. Attention should be given to categories that receive similar quantifications. These categories affect the predicted response in the same manner. However, the transformation type dictates the basic appearance of the plot.

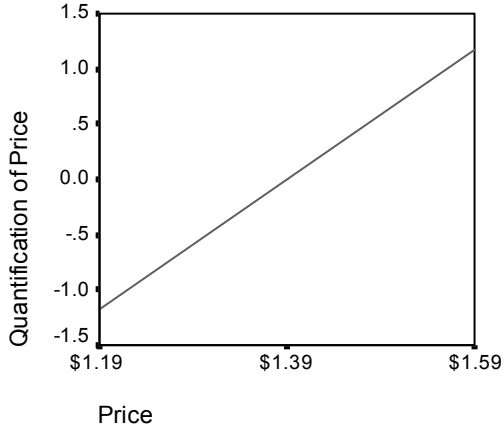
Variables treated as numerical result in a linear relationship between the quantifications and the original categories, corresponding to a straight line in the transformation plot. The order and the difference between the original categories is preserved in the quantifications.

The order of the quantifications for variables treated as ordinal correspond to the order of the original categories. However, the differences between the categories are not preserved. As a result, the transformation plot is nondecreasing but need not be a straight line. If consecutive categories correspond to similar quantifications, the category distinction may be unnecessary and the categories could be combined. Such categories result in a plateau on the transformation plot. However, this pattern can also result from imposing an ordinal structure on a variable that should be treated as nominal. If a subsequent nominal treatment of the variable reveals the same pattern, combining categories is warranted. Moreover, if the quantifications for a variable treated as ordinal fall along a straight line, a numerical transformation may be more appropriate.

For variables treated as nominal, the order of the categories along the horizontal axis corresponds to the order of the codes used to represent the categories. Interpretations of category order or of the distance between the categories is unfounded. The plot can assume any nonlinear or linear form. If an increasing trend is present, an ordinal treatment should be attempted. If the nominal transformation plot displays a linear trend, a numerical transformation may be more appropriate.

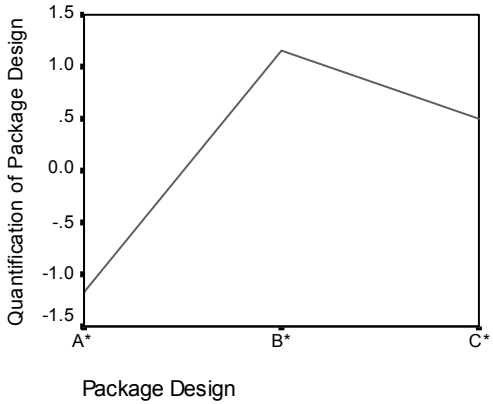
Figure 8.12 displays the transformation plot for *price*, which was treated as numerical. Notice that the order of the categories along the straight line correspond to the order of the original categories. Also, the difference between the quantifications for *$1.19* and *$1.39* (–1.173 and 0) is the same as the difference between the quantifications for *$1.39* and *$1.59* (0 and 1.173). The fact that categories 1 and 3 are the same distance from category 2 is preserved in the quantifications.

Figure 8.12 Transformation plot for price (numerical)



The nominal transformation of *package* yields the transformation plot in Figure 8.13. Notice the distinct nonlinear shape in which the second category has the largest quantification. In terms of the regression, the second category decreases predicted preference ranking, whereas the first and third categories have the opposite effect.

Figure 8.13 Transformation plot for package (nominal)



Residual Analysis

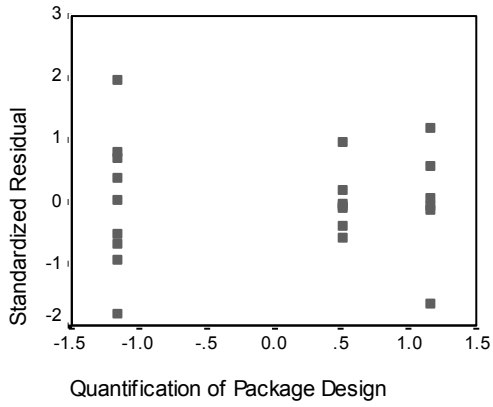
Using the transformed data and residuals that you saved to the working file allows you to create a scatterplot like the one in Figure 8.4.

To obtain such a scatterplot, recall the Simple Scatterplot dialog box and click *Reset* to clear your previous selections and restore the default options.

- Y Axis: *res_1*
- X Axis: *tra2_1*

Figure 8.14 shows the standardized residuals plotted against the optimal scores for *package*. All of the residuals are within two standard deviations of 0. A random scatter of points replaces the U-shape present in Figure 8.4. Predictive abilities are improved by optimally quantifying the categories.

Figure 8.14 Residuals for categorical regression



Example 2: Ozone Data

In this example, you will use a larger set of data to illustrate the selection and effects of optimal scaling transformations. The data include 330 observations on six meteorological variables analyzed by Breiman and Friedman (1985), and Hastie and Tibshirani (1990), among others. Table 8.2 describes the original variables. Your categorical regression attempts to predict the ozone concentration from the remaining variables. Previous researchers found nonlinearities among these variables, which hinder standard regression approaches.

Table 8.2 Original variables

Variable	Description
ozon	daily ozone level; categorized into one of 38 categories
ibh	inversion base height
dpg	pressure gradient (mm Hg)
vis	visibility (miles)
temp	temperature (degrees F)
doy	day of the year

This data set can be found in *ozone.sav*.

Categorizing Variables

In many analyses, variables need to be categorized or recoded before a categorical regression can be performed. For example, the Categorical Regression procedure truncates any decimals and treats negative values as missing. If either of these applications is undesirable, the data must be recoded before performing the regression. Moreover, if a variable has more categories than is practically interpretable, you should modify the categories before the analysis to reduce the category range to a more manageable number.

The variable *doy* has a minimum value of 3 and a maximum value of 365. Using this variable in a categorical regression corresponds to using a variable with 365 categories. Similarly, *vis* ranges from 0 to 350. To simplify analyses, divide each variable by 10, add 1, and round the result to the nearest integer. The resulting variables, denoted *ddoy* and *dvis*, have only 38 and 36 categories respectively, and are consequently much easier to interpret.

The variable *ibh* ranges from 111 to 5000. A variable with this many categories results in very complex relationships. However, dividing by 100 and rounding the result to the nearest integer yields categories ranging from 1 to 50 for the variable *dibh*. Using a 50-category variable rather than a 5000-category variable simplifies interpretations significantly.

Categorizing *dpg* differs slightly from categorizing the previous three variables. This variable ranges from –69 to 107. The procedure omits any categories coded with negative numbers from the analysis. To adjust for the negative values, add 70 to all observations to yield a range from 1 to 177. Dividing this range by 10 and adding 1 results in *ddpg*, a variable with categories ranging from 1 to 19.

The temperatures for *temp* range from 25 to 93 on the Fahrenheit scale. Converting to Celsius and rounding yields a range from –4 to 34. Adding 5 eliminates all negative numbers and results in *tempc*, a variable with 39 categories.

To compute the new variables as suggested, from the menus choose:

Transform
 Compute...

Target Variable: *ddoy*
Numeric Expression: RND(doy/10 +1)

Recall the Compute Variable dialog box. Click *Reset* to clear your previous selections.

Target Variable: *dvis*
Numeric Expression: RND(vis/10 +1)

Recall the Compute Variable dialog box. Click *Reset* to clear your previous selections.

Target Variable: *dibh*
Numeric Expression: RND(ibh/100)

Recall the Compute Variable dialog box. Click *Reset* to clear your previous selections.

Target Variable: *ddpg*
Numeric Expression: RND((dpg+70)/10 +1)

Recall the Compute Variable dialog box. Click *Reset* to clear your previous selections.

Target Variable: *tempc*
Numeric Expression: RND((temp-32)/1.8) +5

As described above, different modifications for variables may be required before conducting a categorical regression. The divisors used here are purely subjective. If you desire fewer categories, divide by a larger number. For example, *doy* could have been divided into months of the year or seasons.

Selection of Transformation Type

Each variable can be analyzed at one of three different levels. However, because prediction of the response is the goal, you should scale the response "as is" by employing the numerical optimal scaling level. Consequently, the order and the differences between categories will be preserved in the transformed variable.

To obtain a categorical regression in which the dependent variable is scaled at the numerical level and the independent variables are scaled at the nominal level, from the menus choose:

Analyze
  Regression
    Optimal Scaling...

Dependent: *ozon*
Independent(s): *ddpg*, *ddoy*, *dibh*, *dvis*, *tempc*

Select *ozon*. Click *Define Scale.*
  Optimal Scaling Level
  ⊙ Numerical

Select *ddpg*, *ddoy*, *dibh*, *dvis*, and *tempc*. Click *Define Scale.*
  Optimal Scaling Level
  ⊙ Nominal

Output...
  Display
  ☐ ANOVA table (deselect)

Plots...
  ▶ Transformation Plots: *ddpg*, *ddoy*, *dibh*, *dvis*, *tempc*

Treating all predictors as nominal yields an R^2 of 0.883. This large amount of variance accounted for is not surprising because nominal treatment imposes no restrictions on the quantifications. However, interpreting the results can be quite difficult.

Figure 8.15 Model summary

Multiple R	R Square	Adjusted R Square
.940	.883	.881

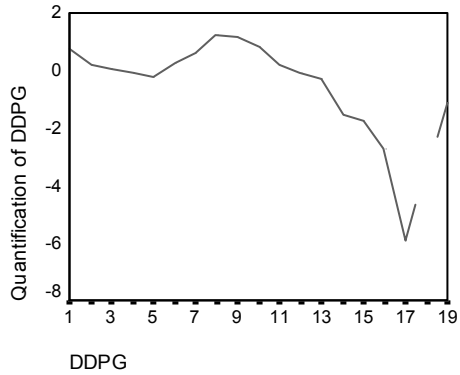
Figure 8.16 shows the standardized regression coefficients of the predictors. A common mistake made when interpreting these values involves focusing on the coefficients while neglecting the quantifications. You cannot assert that the large positive value of the *tempc* coefficient implies that as *tempc* increases, predicted *ozon* increases. Similarly, the negative coefficient for *dibh* does not suggest that as *dibh* increases, predicted *ozon* decreases. All interpretations must be relative to the *transformed* variables. As the quantifications for *tempc* increase, or as the quantifications for *dibh* decrease, predicted *ozon* increases. To examine the effects of the original variables, you must relate the categories to the quantifications.

Figure 8.16 Regression coefficients (all predictors nominal)

	Standardized Coefficients			
	Beta	Std. Error	F	Importance
DDOY	-.340	.020	279.077	.110
DVIS	-.199	.019	104.087	.073
DIBH	-.264	.020	175.887	.144
DDPG	.249	.020	152.275	.041
TEMPC	.681	.020	1124.375	.631

Figure 8.17 displays the transformation plot for *ddpg*. The initial categories (1 through 7) receive small quantifications and thus have minimal contributions to the predicted response. Categories 8 through 10 receive somewhat higher, positive values, resulting in a moderate increase in predicted *ozon*. The quantifications decrease up to category 17, where *ddpg* has its greatest decreasing effect on predicted *ozon*. Although the line increases after this category, using an ordinal scaling level for *ddpg* may not significantly reduce the fit, while simplifying the interpretations of the effects. However, the importance measure of 0.04 and the regression coefficient for *ddpg* indicates that this variable is not very useful in the regression.

Figure 8.17 Transformation plot for ddpg (nominal)



The transformation plots for *dvis* and *dibh* (Figure 8.18 and Figure 8.19) show no apparent pattern. As evidenced by the jagged nature of the plots, moving from low categories to high categories yields fluctuations in the quantifications in both directions. Thus, describing the effects of these variables requires focusing on the individual categories. Imposing ordinal or linear restrictions on the quantifications for either of these variables might significantly reduce the fit.

Figure 8.18 Transformation plot for dvis (nominal)

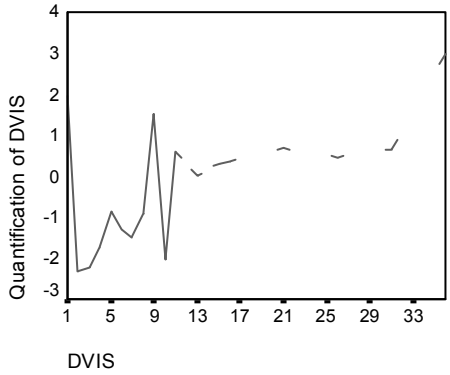


Figure 8.19 Transformation plot for dibh (nominal)

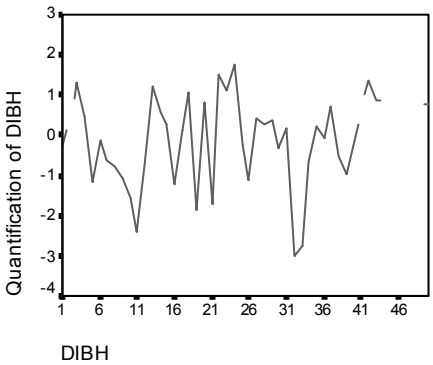
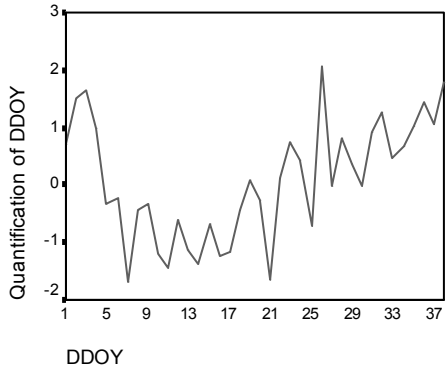


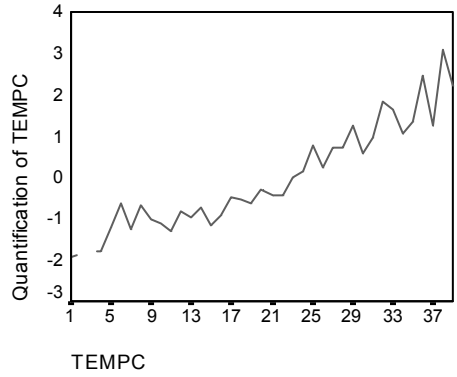
Figure 8.20 shows the transformation plot for *ddoy*. In contrast to Figure 8.18, this plot displays a pattern. The quantifications tend to decrease up to category 21, at which point they tend to increase, yielding a U-shape. Considering the sign of the regression coefficient for *ddoy*, the initial categories (1 through 5) receive quantifications that have a decreasing effect on predicted *ozon*. From category 6 onward, the effect of the quantifications on predicted *ozon* gets more increasing, reaching a maximum around category 21. Beyond category 21, the quantifications tend to decrease the predicted *ozon*. Although the line is quite jagged, the general shape is still identifiable.

Figure 8.20 Transformation plot for ddoy (nominal)



The transformation plot for *tempc* (Figure 8.21) displays an alternative pattern. As the categories increase, the quantifications tend to increase. As a result, as *tempc* increases, predicted *ozon* tends to increase. This pattern suggests scaling *tempc* at the ordinal level.

Figure 8.21 Transformation plot for tempc (nominal)



Thus, the transformation plots suggest scaling *tempc* at the ordinal level while keeping all other predictors nominally scaled. To recompute the regression, scaling *tempc* at the ordinal level, recall the Categorical Regression dialog box.

Select *tempc*. Click *Define Range and Scale.*

Optimal Scaling Level
⊙ Ordinal

Options...

☑ Save transformed data

Plot...

▶ Plot: *tempc*

This model results in an R^2 of 0.873, so the variance accounted for decreases negligibly when the quantifications for *tempc* are restricted to be ordered.

Figure 8.22 Model summary for regression with tempc ordinal

Multiple R	R Square	Adjusted R Square
.934	.873	.871

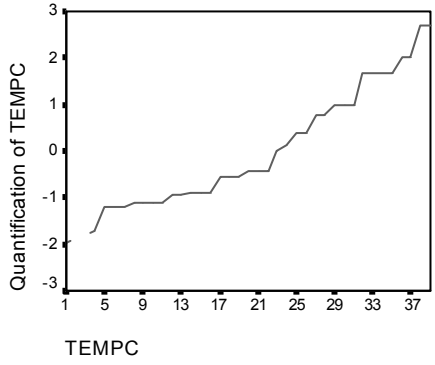
Figure 8.23 displays the coefficients, correlations, and importances. Comparing the coefficients to those in Figure 8.16, no large changes occur. The importance measures suggest that *tempc* is still much more important to the regression than the other variables. Now, however, as a result of the ordinal scaling level of *tempc* and the positive regression coefficient, you can assert that as *tempc* increases, predicted *ozon* increases.

Figure 8.23 Coefficients and importances

	Standardized Coefficients			
	Beta	Std. Error	F	Importance
DVIS	-.197	.020	95.122	.072
DIBH	-.269	.021	164.501	.151
DDPG	.240	.021	128.980	.034
TEMPC	.686	.021	1037.918	.642
DDOY	-.337	.021	253.566	.101

The transformation plot in Figure 8.24 illustrates the ordinal restriction on the quantifications for *tempc*. The jagged line in Figure 8.21 is here replaced by a smooth increasing line. Moreover, no long plateaus are present, indicating that collapsing categories is not needed.

Figure 8.24 Transformation plot for tempc (ordinal)



Optimality of the Quantifications

As stated previously, the transformed variables from a categorical regression can be used in a standard linear regression, yielding identical results. However, the quantifications are optimal only for the model that produced them. Using a subset of the predictors in linear regression does not correspond to an optimal scaling regression on the same subset.

For example, the categorical regression that you have computed has an R^2 of 0.873. You have saved the transformed variables, so in order to fit a linear regression using only *tempc*, *dvis*, and *dibh* as predictors, from the menus choose:

Analyze
  Regression
    Linear...

▶ Dependent: *trans1_1*
▶ Independent(s): *trans2_1*, *trans3_1*, *trans5_1*

Statistics...
  ☐ Descriptives (deselect)

  Regression Coefficients
  ☐ Estimates (deselect)

Figure 8.25 Model summary for regression with subset of optimally scaled predictors

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.870	.757	.755	.4962

Using the quantifications for the response, *tempc*, *dvis*, and *dibh* in a standard linear regression results in a fit of 0.757. To compare this to the fit of a categorical regression using just those three predictors, recall the Categorical Regression dialog box:

▸ Independent(s): *tempc*, *dvis*, *dibh*

Options...

Display

☐ Coefficients (deselect)

☐ Save transformed data (deselect)

Plot...

▸ Plot: (blank)

Figure 8.26 Model summary for categorical regression on three predictors

Multiple R	R Square	Adjusted R Square
.889	.791	.789

The categorical regression analysis has a fit of 0.791, which is better than the fit of 0.757. This demonstrates the property of the scalings that the quantifications obtained in the original regression are only optimal when all five variables are included in the model.

Effects of Transformations

Transforming the variables makes a nonlinear relationship between the original response and the original set of predictors linear for the transformed variables. However, when there are multiple predictors, pairwise relationships are confounded by the other variables in the model.

To focus your analysis on the relationship between *ozon* and *ddoy*, begin by looking at a scatterplot. From the menus choose:

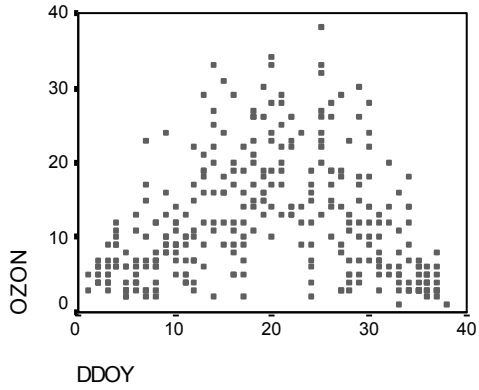
Graphs
 Scatter...

Select *Simple*. Click *Define*.

- Y Axis: *ozon*
- X Axis: *ddoy*

Figure 8.27 illustrates the relationship between *ozon* and *ddoy.* As *ddoy* increases to approximately 25, *ozon* increases. However, for *ddoy* values greater than 25, *ozon* decreases. This inverted U pattern suggests a quadratic relationship between the two variables. A linear regression cannot capture this relationship.

Figure 8.27 Scatterplot of ozon and ddoy



By excluding the other variables from the model, you can focus on the relationship between *ozon* and *ddoy.* However, all interpretations based on the reduced model apply only to the reduced model. Do not generalize the results to the regression involving all predictors.

To obtain a standard linear regression of *ozon* on *ddoy*, recall the Linear Regression dialog box:

- Dependent: *ozon*
- Independent(s): *ddoy*

Figure 8.28 Model summary for linear regression of ozon on ddoy

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.066	.004	.001	8.0057

The regression of *ozon* on *ddoy* yields an R^2 of 0.004. This fit suggests that *ddoy* has no predictive value for *ozon*. This is not surprising, given the pattern in Figure 8.27. By using optimal scaling, however, you can linearize the quadratic relationship and use the transformed *ddoy* to predict the response.

To obtain a categorical regression of *ozon* on *ddoy*, recall the Categorical Regression dialog box:

▸ Independent(s): *ddoy*

Select *ddoy*. Click *Define Scale.*

Optimal Scaling
⊙ Nominal

Save...

☑ Transformed variables

Plots...

▸ Transformation Plots: *ddoy*

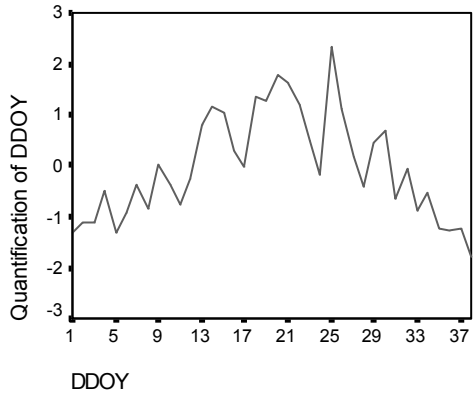
Figure 8.29 Model summary for categorical regression of ozon on ddoy

Multiple R	R Square	Adjusted R Square
.750	.562	.561

The optimal scaling regression treats *ozon* as numerical and *ddoy* as nominal. This results in an R^2 of 0.562. Although only 56% of the variation in *ozon* is accounted for by the categorical regression, this is a substantial improvement over the original regression. Transforming *ddoy* allows for the prediction of *ozon*.

Figure 8.30 displays the transformation plot for *ddoy*. The extremes of *ddoy* both receive negative quantifications, whereas the central values have positive quantifications. By applying this transformation, the low and high *ddoy* values have similar effects on predicted *ozon.*

Figure 8.30 Transformation plot for ddoy (nominal)

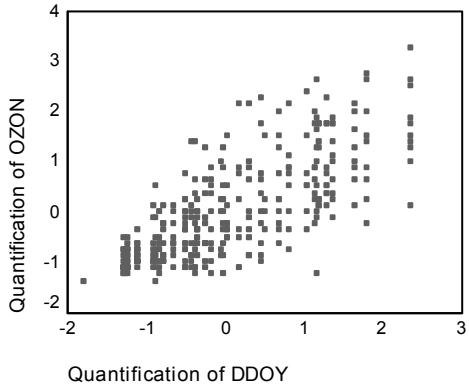


To see a scatterplot of the transformed variables, recall the Simple Scatterplot dialog box, and click *Reset* to clear your previous selections.

- Y Axis: *tra1_2*
- X Axis: *tra2_2*

Figure 8.31 depicts the relationship between the transformed variables. An increasing trend replaces the inverted U in Figure 8.27. The regression line has a slope of 0.750, indicating that as transformed *ddoy* increases, predicted *ozon* increases. Using optimal scaling linearizes the relationship and allows interpretations that would otherwise go unnoticed.

Figure 8.31 Scatterplot of the transformed variables



9 Categorical Principal Components Analysis Examples

Categorical principal components analysis can be thought of as a method of dimension reduction. A set of variables is analyzed to reveal major dimensions of variation. The original data set can then be replaced by a new, smaller data set with minimal loss of information. The method reveals relationships among variables, among cases, and among variables and cases.

The criterion used by categorical principal components analysis for quantifying the observed data is that the object scores (component scores) should have large correlations with each of the quantified variables. A solution is good to the extent that this criterion is satisfied.

Two examples of categorical principal components analysis will be presented. The first employs a rather small data set useful for illustrating the basic concepts and interpretations associated with the procedure. The second example examines a practical application.

Example 1: Interrelations of Social Systems

This example examines Guttman's (1968) adaptation of a table by Bell (1961). The data are also discussed by Lingoes (1968).

Bell presented a table to illustrate possible social groups. Guttman used a portion of this table, in which five variables describing such things as social interaction, feelings of belonging to a group, physical proximity of members, and formality of the relationship were crossed with seven theoretical social groups, including crowds (for example, people at a football game), audiences (for example, people at a theater or classroom lecture), public (for example, newspaper or television audiences), mobs (like a crowd but with much more intense interaction), primary groups (intimate), secondary groups (voluntary), and the modern community (loose confederation resulting from close physical proximity and a need for specialized services).

Table 9.1 shows the variables in the data set resulting from the classification into seven social groups used in the Guttman-Bell data, with their variable labels and the value labels (categories) associated with the levels of each variable. This data set can be found in *guttman.sav*. In addition to selecting variables to be included in the computation of the categorical principal components analysis, you can select variables that are used to label objects in plots. In this example, the first five variables in the data are included in the analysis, while cluster is used exclusively as a labeling variable. When you specify a categorical principal components analysis, you must specify the optimal scaling level for each analysis variable. In this example, an ordinal level is specified for all analysis variables.

Table 9.1 Variables in the Guttman-Bell data set

Variable name	Variable label	Value labels
intnsity	Intensity of interaction	Slight, low, moderate, high
frquency	Frequency of interaction	Slight, nonrecurring, infrequent, frequent
blonging	Feeling of belonging	None, slight, variable, high
proxmity	Physical proximity	Distant, close
formlity	Formality of relationship	No relationship, formal, informal
cluster		Crowds, audiences, public, mobs, primary groups, secondary groups, modern community

To produce categorical principal components output for this data set, from the menus choose:

Analyze
  Data Reduction
    Optimal Scaling...

Optimal Scaling Level
⊙ Some variable(s) not multiple nominal

Number of Sets of Variables
⊙ One set

▸ Analysis Variables: *intnsity*, *frquency*, *blonging*, *proxmity*, *formlity*
▸ Labeling Variables: *cluster*

Select *intnsity*, *frquency*, *blonging*, *proxmity*, *formlity*. Click *Define Scale and Weight*.

Optimal Scaling Level
⊙ Ordinal

Output...
  Tables
  ☑ Object scores
  ☐ Correlations of transformed variables (deselect)

  ▸ Category Quantifications: *intnsity*, *frquency*, *blonging*, *proxmity*, *formlity*

  Object Scores Options
  ▸ Label Object Scores By: *cluster*

Plots
Object...
  Plots
  Objects and variables (biplot)

  Label Objects
  Label by:
  ⊙ Variable
  ▸ Selected: *cluster*

Plots
Category...
  ▸ Joint Category Plots: *intnsity*, *frquency*, *blonging*, *proxmity*, *formlity*

Number of Dimensions

Figure 9.1 and Figure 9.2 show some of the initial output for the categorical principal components analysis. After the iteration history of the algorithm, the model summary, including the eigenvalues of each dimension, is displayed. These eigenvalues are equivalent to those of classical principal components analysis. They are measures of how much variance is accounted for by each dimension.

Figure 9.1 Iteration history

	Variance Accounted For		Loss		
Iteration Number	Total	Increase	Total	Centroid Coordinates	Restriction of Centroid to Vector Coordinates
31[1]	4.726009	.000008	5.273991	4.273795	1.000196

1. The iteration process stopped because the convergence test value was reached.

Figure 9.2 Model summary

		Variance Accounted For	
Dimension	Cronbach's Alpha	Total (Eigenvalue)	% of Variance
1	.881	3.389	67.774
2	.315	1.337	26.746
Total	.986[1]	4.726	94.520

1. Total Cronbach's Alpha is based on the total Eigenvalue.

The eigenvalues can be used as an indication of how many dimensions are needed. In this example, the default number of dimensions, 2, was used. Is this the right number? As a general rule, when all variables are either single nominal, ordinal, or numerical, the eigenvalue for a dimension should be larger than 1. Since the two-dimensional solution accounts for 94.5% of the variance, a third dimension probably would not add much more information.

For multiple nominal variables, there is no easy rule of thumb to determine the appropriate number of dimensions. If the number of variables is replaced by the total number of categories minus the number of variables, the above rule still holds. But this rule alone would probably allow more dimensions than are needed. When choosing the number of dimensions, the most useful guideline is to keep the number small enough so that meaningful interpretations are possible. The model summary table also shows Cronbach's alpha (a measure of reliability), which is maximized by the procedure.

Quantifications

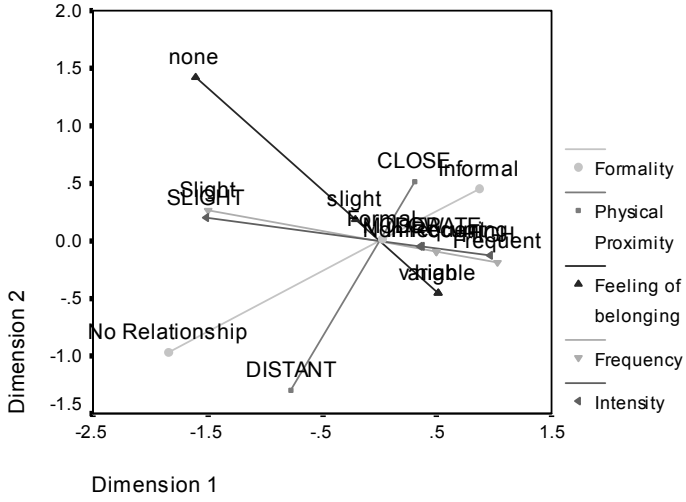
For each variable, the quantifications, the vector coordinates, and the centroid coordinates for each dimension are presented. The quantifications are the values assigned to each category. The centroid coordinates are the average of the object scores of objects in the same category. The vector coordinates are the coordinates of the categories when they are required to be on a line, representing the variable in the object space. This is required for variables with the ordinal and numerical scaling level.

Figure 9.3 Quantifications for intensity of interaction

			Centroid Coordinates		Vector Coordinates	
			Dimension		Dimension	
Category	Frequency	Quantification	1	2	1	2
SLIGHT	2	-1.530	-1.496	.308	-1.510	.208
LOW	2	.362	.392	.202	.358	-.049
MODERATE	1	.379	.188	-1.408	.374	-.051
HIGH	2	.978	1.010	.194	.965	-.133

Figure 9.4 shows the joint plot of the category points for the present example. Glancing at the quantifications, you can see that some of the categories of some variables were not clearly separated by the categorical principal components analysis as cleanly as would have been expected if the level had been truly ordinal. (Use the point identification feature to read obscured point labels. Choose *Interpolation* from the Format menu and click *Straight* to display the markers for the categories.) Variables *intnsity* and *frquency*, for example, have equal or almost equal quantifications for their two middle categories. This kind of result might suggest trying alternative categorical principal components analyses, perhaps with some categories collapsed, or perhaps with a different level of analysis, such as (multiple) nominal. Figure 9.4 resembles the plot for the component loadings (Figure 9.7), but it also shows where the endpoints are located that correspond to the lowest quantifications (for example, *slight* for *intnsity* and *none* for *blonging*).

Figure 9.4 Joint plot category points



To display markers, double-click on the graph and from the Chart Editor menus choose:

Format
Interpolation...

Interpolation Style
Straight

The two variables measuring interaction, *intnsity* and *frquency*, appear very close together and account for much of the variance in dimension 1. *Formlity* also appears close to *proxmity.*

By focusing on the category points, you can see the relationships even more clearly. Not only are *intnsity* and *frquency* close, but the directions of their scales are similar; that is, slight intensity is close to slight frequency, and frequent interaction is near high intensity of interaction. You also see that close physical proximity seems to go hand-in-hand with an informal type of relationship, and physical distance is related to no relationship.

Object Scores

You can also request a listing and plot of object scores. The plot of the object scores can be useful for detecting outliers, detecting typical groups of objects, or revealing some special patterns.

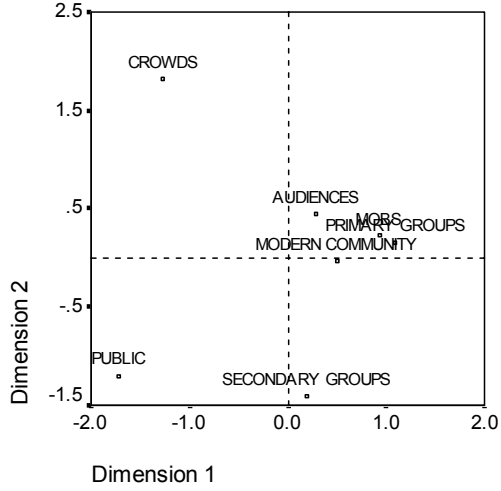
Figure 9.5 shows the listing of object scores labeled by social group for the Guttman-Bell data. By examining the values for the object points, you can identify specific objects in the plot.

Figure 9.5 Object scores

	Dimension	
	1	2
CROWDS	-1.266	1.816
AUDIENCES	.284	.444
PUBLIC	-1.726	-1.201
MOBS	.931	.229
PRIMARY GROUPS	1.089	.159
SECONDARY GROUPS	.188	-1.408
MODERN COMMUNITY	.500	-.039

The first dimension appears to separate *CROWDS* and *PUBLIC*, which have relatively large negative scores, from *MOBS* and *PRIMARY GROUPS*, which have relatively large positive scores. The second dimension has three clumps: *PUBLIC* and *SECONDARY GROUPS* with large negative values, *CROWDS* with large positive values, and the other social groups in between. This is easier to see by inspecting the plot of the object scores, shown in Figure 9.6.

Figure 9.6 Object scores plot



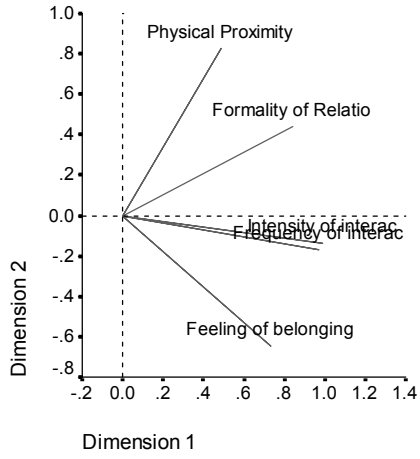
In the plot, you see *PUBLIC* and *SECONDARY GROUPS* at the bottom, *CROWDS* at the top, and the other social groups in the middle. Examining patterns among individual objects depends on the additional information available for the units of analysis. In this case, you know the classification of the objects. In other cases, you can use supplementary variables to label the objects. You can also see that the categorical principal components analysis does not separate *MOBS* from *PRIMARY GROUPS*. Although most

people usually don't think of their families as mobs, on the variables used, these two groups received the same score on four of the five variables! Obviously, you might want to explore possible shortcomings of the variables and categories used. For example, high intensity of interaction and informal relationships probably mean different things to these two groups. Alternatively, you might consider a higher dimensional solution.

Component Loadings

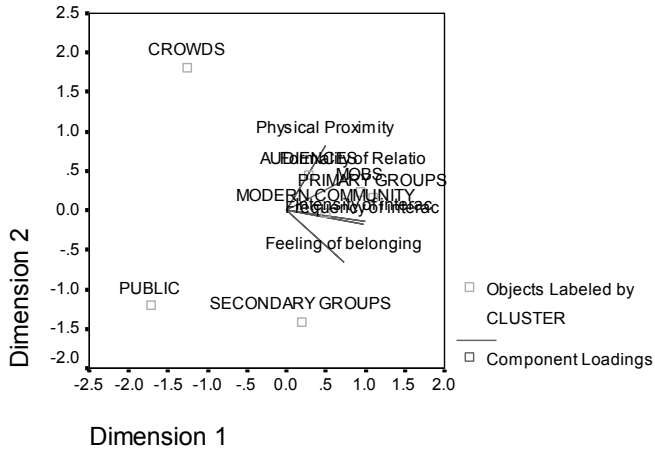
Figure 9.7 shows the plot of component loadings. The vectors (lines) are relatively long, indicating again that the first two dimensions account for most of the variance of all the quantified variables. On the first dimension, all variables have high (positive) component loadings. The second dimension is correlated mainly with quantified variables *blonging* and *proxmity*, in opposite directions. This means that objects with a large negative score in dimension 2 will have a high score in feeling of belonging and a low score in physical proximity. The second dimension, therefore, reveals a contrast between these two variables while having little relation with the quantified variables *frquency* and *intnsity.*

Figure 9.7 Component loadings



To examine the relation between the objects and the variables, look at the biplot of objects and component loadings in Figure 9.8. The vector of a variable points into the direction of the highest category of the variable. For example, for *proxmity* and *blonging* the highest categories are *close* and *high*, respectively. Therefore, *CROWDS* are characterized by close physical proximity and no feeling of belonging, and *SECONDARY GROUPS*, by distant physical proximity and a high feeling of belonging.

Figure 9.8 Biplot



Additional Dimensions

Increasing the number of dimensions will increase the amount of variation accounted for and may reveal differences concealed in lower dimensional solutions. As noted previously, in two dimensions *MOBS* and *PRIMARY GROUPS* cannot be separated. However, increasing the dimensionality may allow the two groups to be differentiated.

To obtain a three-dimensional solution, recall the Categorical Principal Components dialog box:

Dimensions in solution: 3

Output...
- ☐ Object scores (deselect)

▸ Category Quantifications: (empty)

Plots
Object...
- ☐ Component loadings (deselect)

Plots
Category...

▸ Joint Category Plots: (empty)

A three-dimensional solution has eigenvalues of 3.424, 0.844, and 0.732.

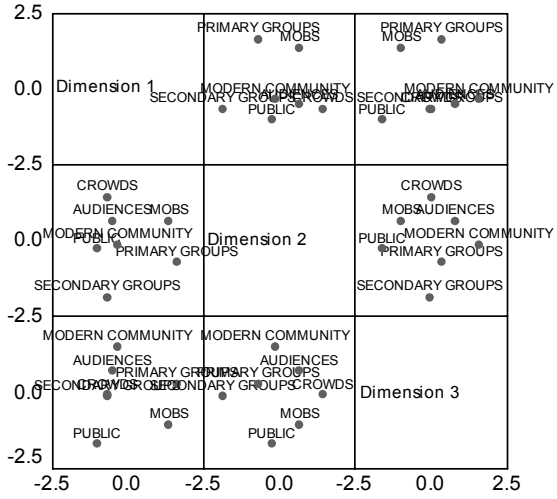
Figure 9.9 Model summary

Dimension	Cronbach's Alpha	Variance Accounted For	
		Total (Eigenvalue)	% of Variance
1	.885	3.424	68.480
2	-.232	.844	16.871
3	-.459	.732	14.649
Total	1.000[1]	5.000	99.999

1. Total Cronbach's Alpha is based on the total Eigenvalue.

The object scores for the three-dimensional solution were plotted in the scatterplot matrix in Figure 9.10. In a scatterplot matrix, every dimension is plotted against every other dimension in a series of two-dimensional scatterplots. Note that the first two eigenvalues in three dimensions are not equal to the eigenvalues in the two-dimensional solution; in other words, the solutions are not nested. Because the eigenvalues in dimensions 2 and 3 are now smaller than 1 (giving a Cronbach's alpha that is negative), you should prefer the two-dimensional solution. The three-dimensional solution is included for purposes of illustration.

Figure 9.10 Three-dimensional object scores scatterplot matrix



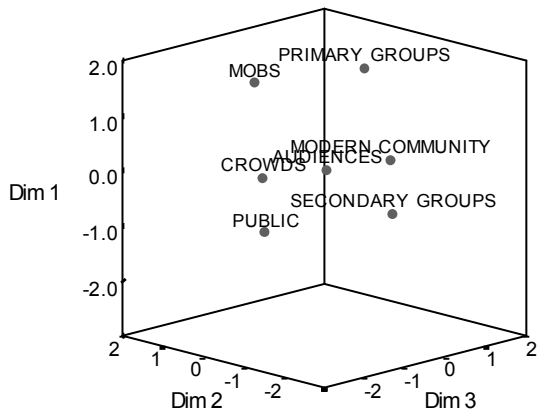
The top row of plots reveals that the first dimension separates *PRIMARY GROUPS* and *MOBS* from the other groups. Notice that the order of the objects along the vertical axis does not change in any of the plots in the top row; each of these plots employs dimension 1 as the *y* axis.

The middle row of plots allows for interpretation of dimension 2. The second dimension has changed slightly from the two-dimensional solution. Previously, the second dimension had three distinct clumps, but now the objects are more spread out along the axis.

The third dimension helps to separate *MOBS* from *PRIMARY GROUPS*, which did not occur in the two-dimensional solution.

Look more closely at the dimension 2 versus dimension 3 and dimension 1 versus dimension 2 plots. On the plane defined by dimensions 2 and 3, the objects form a rough rectangle, with *CROWDS*, *MODERN COMMUNITY*, *SECONDARY GROUPS*, and *PUBLIC* at the vertices. On this plane, *MOBS* and *PRIMARY GROUPS* appear to be convex combinations of *PUBLIC-CROWDS* and *SECONDARY GROUPS-MODERN COMMUNITY*, respectively. However, as previously mentioned, they are separated from the other groups along dimension 1. *AUDIENCES* is not separated from the other groups along dimension 1 and appears to be a combination of *CROWDS* and *MODERN COMMUNITY*. Figure 9.11 shows these relationships in a 3-D scatterplot.

Figure 9.11 Three-dimensional object scores space



Knowing how the objects are separated does not reveal which variables correspond to which dimensions. This is accomplished using the component loadings, which are presented in Figure 9.12. The first dimension corresponds primarily to *blonging*, *intnsity*, and *formlity*; the second dimension separates *frquency* and *proxmity*; and the third dimension separates these from the others.

Figure 9.12 Three-dimensional component loadings

	Dimension		
	1	2	3
INTENSITY	.980	-.005	-.201
FREQUENCY	.521	-.643	.561
BELONGING	.980	-.002	-.197
PROXIMITY	.519	.656	.549
FORMALITY	.981	.004	-.193

Example 2: Symptomatology of Eating Disorders

Eating disorders are debilitating illnesses associated with disturbances in eating behavior, severe body image distortion, and an obsession with weight that affects the mind and body simultaneously. Millions of people are affected each year, with adolescents particularly at risk. Treatments are available and most are helpful when the condition is identified early.

A health professional can attempt to diagnose an eating disorder through a psychological and medical evaluation. However, it can be difficult to assign a patient to one of several different classes of eating disorders because there is no standardized symptomatology of anorectic/bulimic behavior. Are there symptoms that clearly differentiate patients into the four groups? Which symptoms do they have in common?

In order to try to answer these questions, van der Ham, Meulman, van Strien, and van Engeland (1997) made a study of 55 adolescents with known eating disorders, as shown in Table 9.2.

Table 9.2 Patient diagnoses

Diagnosis	Number of Patients
Anorexia nervosa	25
Anorexia with bulimia nervosa	9
Bulimia nervosa after anorexia	14
Atypical eating disorder	7
Total	55

Each patient was seen four times over four years, for a total of 220 observations. At each observation, the patients were scored for each of the 16 symptoms outlined in Table 9.3. Symptom scores are missing for patient 71 at time 2, patient 76 at time 2, and patient 47 at time 3, leaving 217 valid observations. The data can be found in *anorectic.sav*.

Table 9.3 Modified Morgan-Russell subscales measuring well-being

Variable name	Variable label	Lower end (score1)	Upper end (score 3 or 4)
weight	Body weight	Outside normal range	Normal
mens	Menstruation	Amenorrhea	Regular periods
fast	Restriction of food intake (fasting)	Less than 1200 calories	Normal/regular meals
binge	Binge eating	Greater than once a week	No bingeing
vomit	Vomiting	Greater than once a week	No vomiting
purge	Purging	Greater than once a week	No purging
hyper	Hyperactivity	Not able to be at rest	No hyperactivity
fami	Family relations	Poor	Good
eman	Emancipation from family	Very dependent	Adequate
frie	Friends	No good friends	Two or more good friends
school	School/employment record	Stopped school/work	Moderate to good record
satt	Sexual attitude	Inadequate	Adequate
sbeh	Sexual behavior	Inadequate	Can enjoy sex
mood	Mental state (mood)	Very depressed	Normal
preo	Preoccupation with food and weight	Complete	No preoccupation
body	Body perception	Disturbed	Normal

Principal components analysis is ideal for this situation, since the purpose of the study is to ascertain the relationships between symptoms and the different classes of eating disorders. Moreover, categorical principal components analysis is likely to be more useful than classical principal components analysis because the symptoms are scored on an ordinal scale.

To produce categorical principal components output for this data set, from the menus choose:

Analyze
  Data Reduction
    Optimal Scaling...

Optimal Scaling Level
⊙ Some variable(s) not multiple nominal

Number of Sets of Variables
⊙ One set

▸ Analysis Variables: *weight*, *mens*, *fast*, *binge*, *vomit*, *purge*, *hyper*, *fami*, *eman*, *frie*, *school*, *satt*, *sbeh*, *mood*, *preo*, *body*
▸ Supplementary Variables: *tidi*
▸ Labeling Variables: *diag, time*

Select *weight*, *mens*, *fast*, *binge*, *vomit*, *purge*, *hyper*, *fami*, *eman*, *frie*, *school*, *satt*, *sbeh*, *mood*, *preo*, *body*. Click *Define Scale and Weight*.
  Optimal Scaling Level
  ⊙ Ordinal

Select *tidi*. Click *Define Scale*.
  Optimal Scaling Level
  ⊙ Multiple nominal

Options...
  Label Plots By
  ⊙ Variable names or values

Output...
  Tables
  ☑ Object scores
  ☐ Correlations of transformed variables (deselect)

  ▸ Category Quantifications: *tidi*

  Object Scores Options
  ▸ Include Categories Of: *diag*, *time*, *number*

Plots
Object...
  Label Objects
  Label by:
  ⊙ Variable
  ▸ Selected: *diag, time*

Plots
Category...
- Category Plots: *tidi*
- Transformation Plots: *weight*, *mens*, *fast*, *binge*, *vomit*, *purge*, *hyper*, *fami*, *eman*, *frie*, *school*, *satt*, *sbeh*, *mood*, *preo*, *body*
- Project Centroid Of: *tidi*
- Onto: *binge*, *satt*, *preo*

Save
Save to Working File
☑ Transformed variables

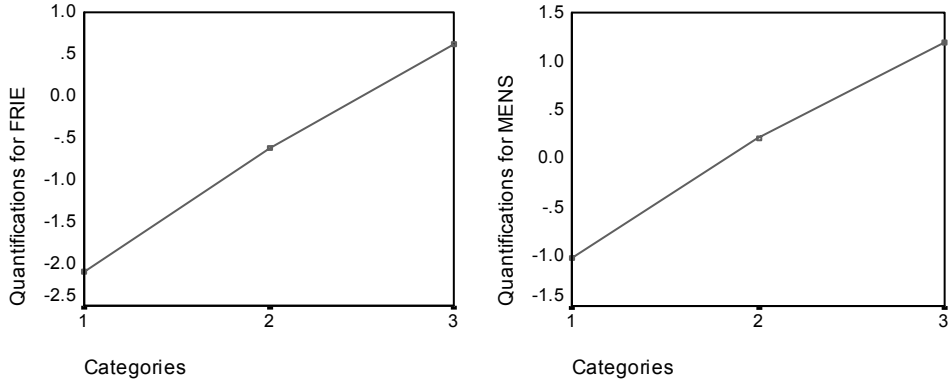
The procedure results in scores for the subjects (with mean 0 and unit variance) and quantifications of the categories that maximize the mean squared correlation of the subject scores and the transformed variables. In the present analysis, the category quantifications were constrained to reflect the ordinal information.

Transformation Plots

The transformation plots display the original category number on the horizontal axes; the vertical axes give the optimal quantifications.

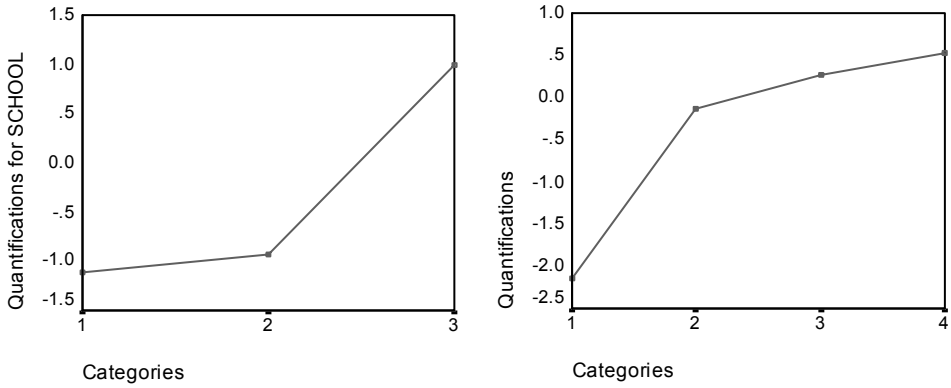
Some variables, like *frie* and *mens*, obtained nearly linear transformations, so in this analysis you may interpret them as numerical.

Figure 9.13 Transformation plots for friends and menstruation



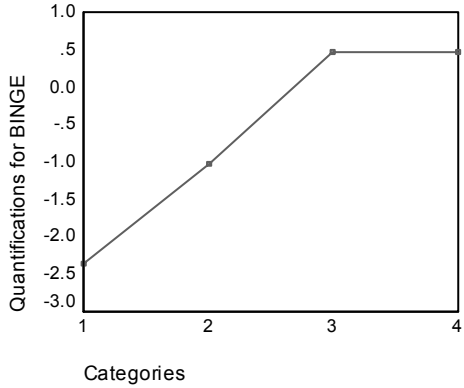
The quantifications for other variables like *school* and *purge* did not obtain linear transformations and should be interpreted at the ordinal scaling level. The difference between the second and third categories is much more important than that between the first and second categories.

Figure 9.14 Transformation plots for work/school record and purging



An interesting case arises in the quantifications for *binge*. The transformation obtained is linear for categories 1 through 3, but the quantified values for categories 3 and 4 are equal. This result shows that scores of 3 and 4 do not differentiate between patients and suggests that you could use the numerical scaling level in a two-component solution by recoding 4's as 3's.

Figure 9.15 Transformation plot for binge eating



Model Summary

To see how well your model fits the data, look at the model summary. About 47% of the total variance is explained by the two-component model, 35% by the first dimension and 12% by the second. So, almost half of the variability on the individual objects level is explained by the two-component model.

Figure 9.16 Model summary

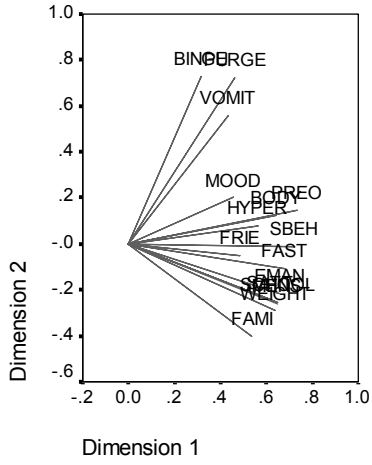
Dimension	Cronbach's Alpha	Variance Accounted For	
		Total (Eigenvalue)	% of Variance
1	.874	5.550	34.690
2	.522	1.957	12.234
Total	.925[1]	7.508	46.924

1. Total Cronbach's Alpha is based on the total Eigenvalue.

Component Loadings

To begin to interpret the two dimensions of your solution, look at the component loadings, shown in Figure 9.17. All variables have a positive component loading in the first dimension, which means there is a common factor that correlates positively with all the variables.

Figure 9.17 Component loadings plot



The second dimension separates the variables. The variables *binge*, *vomit*, and *purge* form a bundle having large positive loadings in the second dimension. These symptoms are typically considered to be representative of bulimic behavior.

The variables *eman*, *school*, *satt*, *weight*, and *mens* form another bundle, and you can include *fast* and *fami* in this bundle, because their vectors are close to the main cluster, and these variables are considered to be anorectic symptoms (*fast*, *weight*, *mens*) or are psychosocial in nature (*eman*, *school*, *satt*, *fami*). The vectors of this bundle are orthogonal (perpendicular) to the vectors of *binge*, *vomit*, and *purge*,

which means that this set of variables is uncorrelated with the set of bulimic variables.

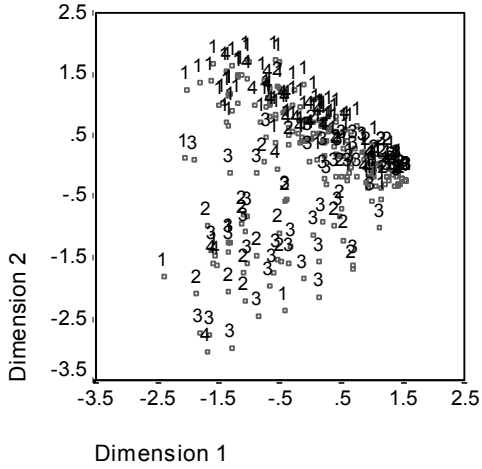
The variables *frie*, *mood*, and *hyper* do not appear to fit very well into the solution. You can see this in the plot by observing the lengths of each vector. The length of a given variable's vector corresponds to its fit, and *frie*, *mood*, and *hyper* have the shortest vectors. Based on a two-component solution, you would probably drop these variables from a proposed symptomatology for eating disorders. They may, however, fit better in a higher dimensional solution.

The variables *sbeh*, *preo*, and *body* form another theoretic group of symptoms, pertaining to how the patient experiences his or her body. While correlated with the two orthogonal bundles of variables, these variables have fairly long vectors and are strongly associated with the first dimension and therefore may provide some useful information about the "common" factor.

Object Scores

Figure 9.18 shows a plot of the object scores, in which the subjects are labeled with their diagnosis category.

Figure 9.18 Object scores plot labeled by diagnosis

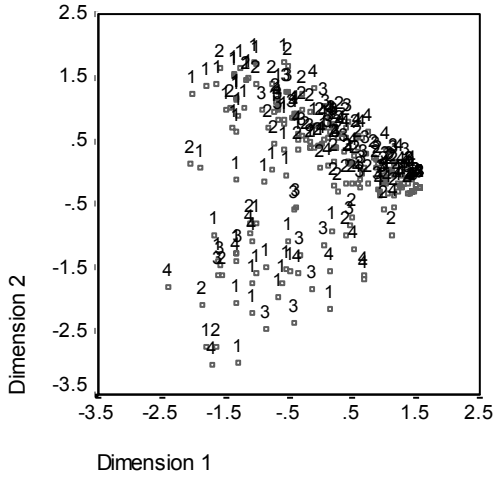


This plot does not help interpret the first dimension because patients are not separated by diagnosis along it. However, there is some information about the second dimension. Anorexia subjects (1) and patients with atypical eating disorder (4) form a group, located above subjects with some form of bulimia (2 and 3). Thus, the second dimension

separates bulimic patients from others, as you have also seen in the previous section (the variables in the bulimic bundle have large positive component loadings in the second dimension). This makes sense, given that the component loadings of the symptoms that are traditionally associated with bulimia have large values in the second dimension.

Figure 9.19 shows a plot of the object scores, in which the subjects are labeled with their time of diagnosis.

Figure 9.19 Object scores labeled by time



Labeling the object scores by time reveals that the first dimension has a relation to time because there seems to be a progression of times of diagnosis from the 1's mostly to the left and others to the right. Note that you can connect the time points in this plot by saving the object scores and creating a scatterplot using the dimension 1 scores on the *x* axis, the dimension 2 scores on the *y* axis, and setting the markers using the patient numbers (*number*).

Comparing the object scores plot labeled by time with the one labeled by diagnosis can give you some insight into unusual objects. For example, in the plot labeled by time, there is a patient whose diagnosis at time 4 lies to the left of all other points in the plot. This is unusual because the general trend of the points is for the later times to lie further to the right. Interestingly, this point that seems out of place in time also has an unusual diagnosis, in that the patient is an anorectic whose scores place the patient in the bulimic cluster. By looking in the table of object scores, you find that this is patient 43, diagnosed with anorexia nervosa, whose object scores are shown in Table 9.4.

Table 9.4 Object scores for patient 43

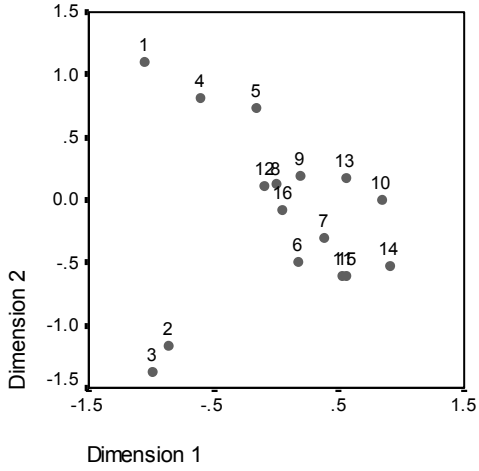
Time	Dimension 1	Dimension 2
1	–2.031	1.250
2	–2.067	0.131
3	–1.575	–1.467
4	–2.405	–1.807

The patient's scores at time 1 are prototypical for anorectics, with the large negative score in dimension 1 corresponding to poor body image, and the positive score in dimension 2 corresponding to no bulimic symptoms, and indication of anorectic symptoms or poor psychosocial behavior. However, unlike the majority of patients, there is little or no progress in dimension 1. In dimension 2, there is apparently some progress toward "normal" (around 0, between anorectic and bulimic behavior), but then the patient shifts to exhibit bulimic symptoms.

Examining the Structure of the Course of Illness

To find out more about how the two dimensions were related to the four diagnosis categories and the four time points, a supplementary variable *tidi* was created by a cross-classification of the four categories of *diag* and the four categories of *time*. Thus, *tidi* has 16 categories, where the first category indicates the anorexia nervosa patients at their first visit. The fifth category indicates the anorexia nervosa patients at time point 2, and so on, with the sixteenth category indicating the atypical eating disorder patients at time point 4. The use of the supplementary variable *tidi* allows for the study of the courses of illness for the different groups over time. The variable was given a multiple nominal scaling level, and the category points are displayed in Figure 9.20.

Figure 9.20 Category points for time/diagnosis interaction



Some of the structure is apparent from this plot: the diagnosis categories at time point 1 clearly separate anorexia nervosa and atypical eating disorder from anorexia nervosa with bulimia nervosa and bulimia nervosa after anorexia nervosa in the second dimension. After that, it's a little more difficult to see the patterns.

However, you can make the patterns more easily visible by creating a scatterplot based on the quantifications. To do this, from the menus choose:

Graphs
Scatter...

Select *Simple* and click *Define*.

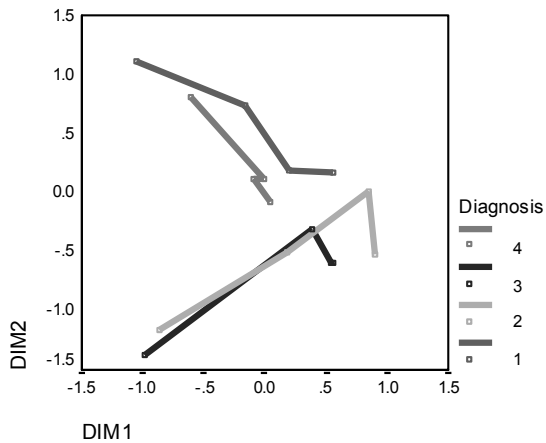
- Y Axis: *tr17_2_1*
- X Axis: *tr17_1_1*
- Set Markers By: *diag*

▶ Then, to connect the points, double-click on the graph, and from the Chart Editor menus choose:

Format
Interpolation...

Interpolation Style
Straight

Figure 9.21 Structures of the courses of illness



By connecting the category points for each diagnostic category across time, the patterns immediately suggest that the first dimension is related to time and the second, to diagnosis, as you previously determined from the object scores plots.

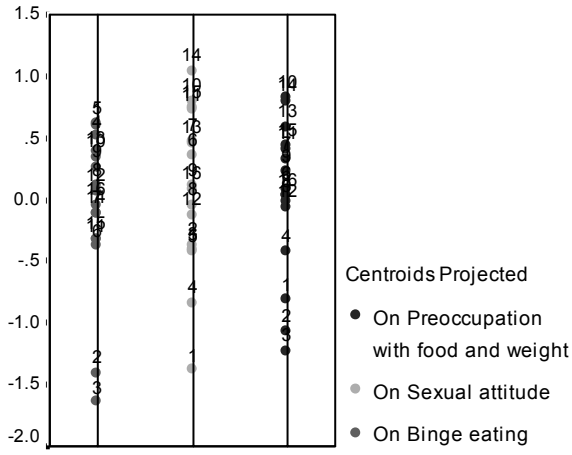
However, this plot further shows that, over time, the illnesses tend to become more alike. Moreover, for all groups, the progress is greatest between time points 1 and 2; the anorectic patients show some more progress from 2 to 3, but the other groups show little progress.

Differential Development for Selected Variables

One variable from each bundle of symptoms identified by the component loadings was selected as "representative" of the bundle. Binge eating was selected from the bulimic bundle, sexual attitude, from the anorectic/psychosocial bundle, and body preoccupation, from the third bundle.

In order to examine the possible differential courses of illness, the projections of *tidi* on *binge*, *satt*, and *preo* were computed and plotted in Figure 9.22.

Figure 9.22 Projected centroids of tidi on binge, satt, and preo



This plot shows that at the first time point, the symptom binge eating separates bulimic patients (2 and 3) from others (1 and 4); sexual attitude separates anorectic and atypical patients (1 and 4) from others (2 and 3); and body preoccupation does not really separate the patients. In many applications, this plot would be sufficient to describe the relationship between the symptoms and diagnosis, but because of the complication of multiple time points, the picture becomes muddled.

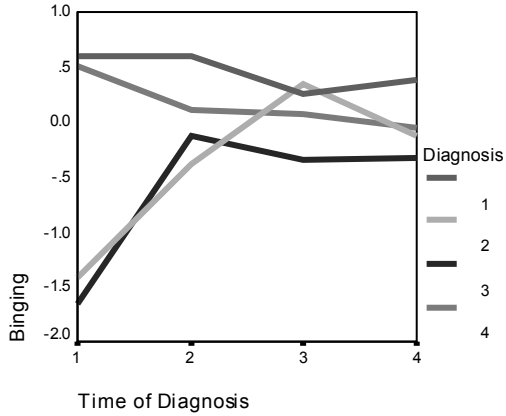
In order to view these projections over time, you need to:

- Copy the contents of the projected centroids table to three new variables, and call them *binge2*, *satt2*, and *preo2*.

- Recall the Simple Scatterplot dialog box and click *Reset* to clear your previous selections.

  ▸Y Axis: *binge2*
  ▸X Axis: *time2*
  ▸Set Markers By: *diag2*

Figure 9.23 Projected centroids of time of diagnosis on binging over time

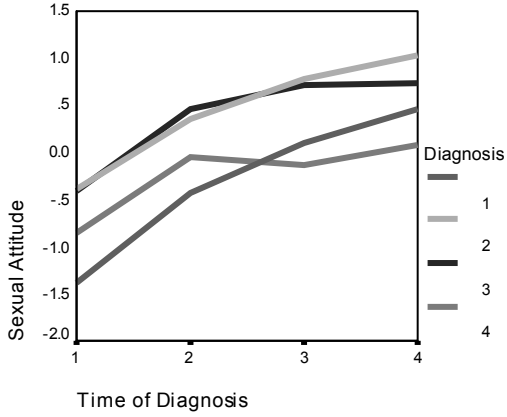


With respect to binge eating, it is clear that the anorectic groups have different starting values from the bulimic groups. This difference shrinks over time, as the anorectic groups hardly change, while the bulimic groups show progress.

▶ Recall the Simple Scatterplot dialog box and click *Reset* to clear your previous selections.

- Y Axis: *satt2*
- X Axis: *time2*
- Set Markers By: *diag2*

Figure 9.24 Projected centroids of time of diagnosis on sexual attitude over time

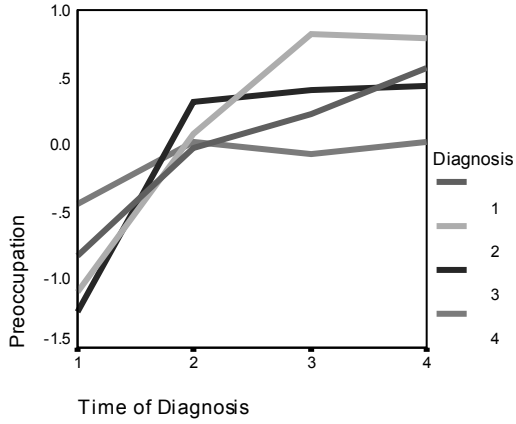


With respect to sexual attitude, the four trajectories are more or less parallel over time, and all groups show progress. The bulimic groups, however, have higher (better) scores than the anorectic group.

▶ Recall the Simple Scatterplot dialog box and click *Reset* to clear your previous selections.

▶Y Axis: *preo2*
▶X Axis: *time2*
▶Set Markers By: *diag2*

Figure 9.25 Projected centroids of time of diagnosis on body preoccupation over time



Body preoccupation is a variable that represents the core symptoms, which are shared by the four different groups. Apart from the atypical eating disorder patients, the anorectic group and the two bulimic groups have very similar levels both at the beginning and at the end.

10 Nonlinear Canonical Correlation Analysis Examples

The purpose of nonlinear canonical correlation analysis is to determine how similar two or more sets of variables are to one another. As in linear canonical correlation analysis, the aim is to account for as much of the variance in the relationships among the sets as possible in a low-dimensional space. Unlike linear canonical analysis, however, nonlinear canonical correlation analysis does not assume an interval level of measurement or that the relationships are linear. Another important difference is that nonlinear canonical correlation analysis establishes the similarity between the sets by simultaneously comparing linear combinations of the variables in each set to an unknown set—the object scores.

Example: An Analysis of Survey Results

The example in this chapter is from a survey by Verdegaal (1985). The responses of 15 subjects to eight variables were recorded. The variables, variable labels, and value labels (categories) in the data set are shown in Table 10.1.

Table 10.1 Survey data

Variable name	Variable label	Value labels
age	Age in years	20–25, 26–30, 31–35, 36–40, 41–45, 46–50, 51–55, 56–60, 61–65, 66–70
marital	Marital status	Single, Married, Other
pet	Pets owned	No, Cat(s), Dog(s), Other than cat or dog, Various domestic animals
news	Newspaper read most often	None, Telegraaf, Volkskrant, NRC, Other
music	Music preferred	Classical, New wave, Popular, Variety, Don't like music
live	Neighborhood preference	Town, Village, Countryside
math	Math test score	0–5, 6–10, 11–15
language	Language test score	0–5, 6–10, 11–15, 16–20

This data set can be found in *verd1985.sav*. The variables of interest are the first six, and they are divided into three sets. Set 1 includes *age* and *marital*, set 2 includes *pet* and *news*, and set 3 includes *music* and *live*. *Pet* is scaled as multiple nominal and *age* is scaled as ordinal; all of the other variables are scaled as single nominal. This analysis requests a random initial configuration. By default, the initial configuration is numerical. However, when some of the variables are treated as single nominal with no possibility of ordering, it is best to choose a random initial configuration. This is the case with most of the variables in this study.

Examining the Data

To obtain a nonlinear canonical correlation analysis for this data set, from the menus choose:

Analyze
  Data Reduction
    Optimal Scaling...

Optimal Scaling Level
⊙ Some variable(s) not multiple nominal

Number of Sets of Variables
⊙ Multiple sets

Set 1
▶ Variables: *age*, *marital*

Select *age*. Click *Define Range and Scale.*

Maximum: 10

Select *marital*. Click *Define Range and Scale.*

  Maximum: 3
  Optimal Scaling Level
  ⊙ Single nominal

Set 2
▶ Variables: *pet*, *news*

Select *pet*. Click *Define Range and Scale.*

  Maximum: 5
  Optimal Scaling Level
  ⊙ Multiple nominal

Select *news*. Click *Define Range and Scale.*

Maximum: 5
Optimal Scaling Level
⊙ Single nominal

Set 3
▸ Variables: *music*, *live*

Select *music*. Click *Define Range and Scale.*

Maximum: 5
Optimal Scaling Level
⊙ Single nominal

Select *live*. Click *Define Range and Scale.*

Maximum: 3
Optimal Scaling Level
⊙ Single nominal

Options...

Display
☐ Centroids (deselect)
☑ Weights and component loadings
Plot
☑ Category centroids
☑ Transformations

☑ Use random initial configuration

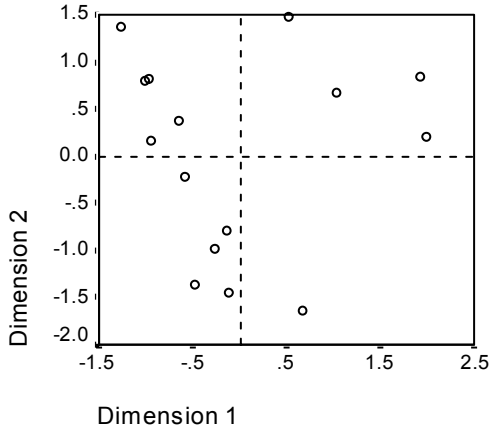
After a list of the variables with their levels of optimal scaling, categorical canonical correlation analysis with optimal scaling produces a table showing the frequencies of objects in categories. This table is especially important if there are missing data, since almost-empty categories are more likely to dominate the solution. In this example, there are no missing data.

A second preliminary check is to examine the plot of object scores. You want to see if there are any outliers that might tend to dominate the solution. Outliers have such different quantifications from the other objects that they will be at the boundaries of the plot, thus dominating one or more dimensions.

If you find outliers, you can handle them in one of two ways. First, you can simply eliminate them from the data and run the nonlinear canonical correlation analysis again. Second, you can try recoding the extreme responses of the outlying object(s) by collapsing (merging) some categories.

As shown in the plot of object scores in Figure 10.1, there were no outliers for the survey data.

Figure 10.1 Object scores



Accounting for Similarity between Sets

The fit and loss values tell you how well the nonlinear canonical correlation analysis solution fits the optimally quantified data with respect to the association between the sets. Figure 10.2 shows the fit value, loss values, and eigenvalues for the survey example.

Figure 10.2 Summary of analysis

		Dimension		
		1	2	Sum
Loss	Set 1	.238	.182	.420
	Set 2	.182	.414	.597
	Set 3	.177	.197	.375
	Mean	.199	.265	.464
Eigenvalue		.801	.735	
Fit				1.536

Loss is partitioned across dimensions and sets. For each dimension and set, loss represents the proportion of variation in the object scores that cannot be accounted for by the weighted combination of variables in the set. The average loss is labeled *Mean*. In this example, the average loss over sets is 0.464. Notice that more loss occurs for the second dimension than for the first.

The eigenvalue for each dimension equals 1 minus the average loss for the dimension and indicates how much of the relationship is shown by each dimension. The eigenvalues add up to the total fit. For Verdegaal's data, $0.801 / 1.536 = 52\%$ of the actual fit is accounted for by the first dimension.

The maximum fit value equals the number of dimensions and, if obtained, indicates that the relationship is perfect. The average loss value over sets and dimensions tells you the difference between the maximum fit and the actual fit. Fit plus the average loss equals the number of dimensions. Perfect similarity rarely happens and usually capitalizes on trivial aspects in the data.

Another measure of association is the multiple correlation between linear combinations from each set and the object scores. If no variables in a set are multiple nominal, you can compute this by multiplying the weight and component loading of each variable within the set, adding these products, and taking the square root of the sum.

Figure 10.3 gives the weights and Figure 10.4 gives the component loadings for the variables in this example. The multiple correlation (R) for the first weighted sum of optimally scaled variables (*age* and *marital*) with the first dimension of object scores is

$$\begin{aligned} R &= \sqrt{(0.701 \times 0.841 + (-0.273 \times -0.631))} \\ &= \sqrt{(0.5895 + 0.1723)} \\ &= 0.8728 \end{aligned}$$

For each dimension, $1 - \text{loss} = R^2$. For example, from Figure 10.2, $1 - 0.238 = 0.762$, which is 0.872 squared (plus some rounding error). Consequently, small loss values indicate large multiple correlations between weighted sums of optimally scaled variables and dimensions. Weights are not unique for multiple nominal variables. For multiple nominal variables, use $1 - \text{loss}$ per set.

Figure 10.3 Weights

Set		Dimension	
		1	2
1	Age in years	.701	.758
	Marital status	-.273	1.014
2	Newspaper read most often	-.853	-.350
3	Music preferred	.600	-.774
	Neighborhood preference	-.514	-.763

Figure 10.4 Component loadings

Set				Dimension 1	Dimension 2
1	Age in years			.841	.241
	Marital status			-.631	.627
2	Pets owned	Dimension	1	.385	-.429
			2	-.274	.680
	Newspaper read most often			-.677	-.379
3	Music preferred			.765	-.529
	Neighborhood preference			-.707	-.515

Another popular statistic with two sets of variables is the canonical correlation. Since the canonical correlation is related to the eigenvalue and thus provides no additional information, it is not included in the nonlinear canonical correlation analysis output. For two sets of variables, the canonical correlation per dimension is obtained by the formula:

$$\rho_d = 2 \times E_d - 1$$

where d is the dimension number and E is the eigenvalue. You can generalize the canonical correlation for more than two sets with the formula:

$$\rho_d = ((K \times E_d) - 1)/(K - 1)$$

where d is the dimension number, K is the number of sets, and E is the eigenvalue. For our example,

$$\rho_1 = ((3 \times 0.801) - 1)/2 = 0.701$$

and

$$\rho_2 = ((3 \times 0.735) - 1)/2 = 0.603$$

The loss of each set is partitioned by the nonlinear canonical correlation analysis in several ways. Figure 10.5 presents the multiple fit, single fit, and single loss tables produced by the nonlinear canonical correlation analysis for the survey example. Note that *multiple fit* minus *single fit* equals *single loss*.

Figure 10.5 Partitioning fit and loss

Set		Multiple Fit			Single Fit			Single Loss		
		Dimension			Dimension			Dimension		
		1	2	Sum	1	2	Sum	1	2	Sum
1	Age in years	.521	.628	1.149	.492	.575	1.067	.030	.053	.083
	Marital status	.076	1.028	1.103	.075	1.028	1.102	.001	.000	.001
2	Pets owned	.390	.443	.833						
	Newspaper read most often	.737	.182	.918	.727	.123	.850	.010	.059	.069
3	Music preferred	.387	.614	1.001	.361	.598	.959	.026	.016	.042
	Neighborhood preference	.265	.583	.848	.264	.583	.847	.000	.000	.000

Single loss indicates the loss resulting from restricting variables to one set of quantifications (that is, single nominal, ordinal, or nominal). If *single loss* is large, it is better to treat the variables as multiple nominal. In this example, however, *single fit* and *multiple fit* are almost equal, which means that the multiple coordinates are almost on a straight line in the direction given by the weights.

Multiple fit equals the variance of the multiple category coordinates for each variable. These measures are analogous to the discrimination measures found in homogeneity analysis. You can examine the multiple fit table to see which variables discriminate best. For example, look at the multiple fit table for *marital* and *news*. The fit values, summed across the two dimensions, are 1.103 for *marital* and 0.918 for *news*. This tells us that variable *news* discriminates less than *marital*.

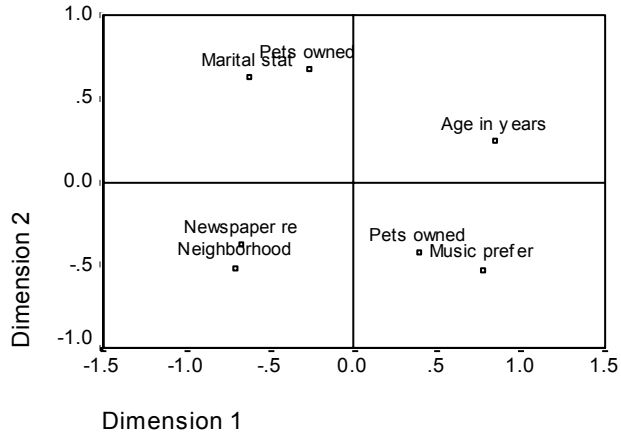
Single fit corresponds to the squared weight for each variable and equals the variance of the single category coordinates. As a result, the weights equal the standard deviations of the single category coordinates. Examining how the single fit is broken down across dimensions, we see that the variable *news* discriminates mainly on the first dimension and that the variable *marital* discriminates almost totally on the second. In other words, the categories of *news* are further apart in the first dimension than in the second, whereas the pattern is reversed for *marital*. In contrast, *age* discriminates in both the first and second dimensions; thus the spread of the categories is equal along both dimensions.

Component Loadings

Figure 10.6 shows the plot of component loadings for the survey data. When there are no missing data, the component loadings are equivalent to the Pearson correlations between the quantified variables and the object scores.

The distance from the origin to each variable point approximates the importance of that variable. The canonical variables are not plotted but can be represented by horizontal and vertical lines drawn through the origin.

Figure 10.6 Component loadings



The relationships between variables are apparent. There are two directions that do not coincide with the horizontal and vertical axes. One direction is determined by *age* (labeled *Age in years*), *news* (labeled *Newspaper re*), and *live* (labeled *Neighborhood*). The other direction is defined by the variables *marital* (labeled *Marital stat*), *music* (labeled *Music prefer*), and *pet* (labeled *Pets owned*). The *pet* variable is a multiple nominal variable, so there are two points plotted for it. Each quantification is interpreted as a single variable.

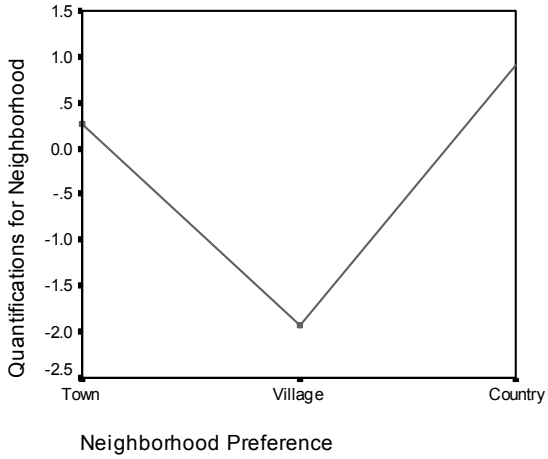
Transformation Plots

The different levels at which each variable can be scaled impose restrictions on the quantifications. Transformation plots illustrate the relationship between the quantifications and the original categories resulting from the selected optimal scaling level.

The transformation plot for *live* (Figure 10.7), which was treated as nominal, displays a U-shaped pattern, in which the middle category receives the lowest quantification and the extreme categories receive values similar to each other. This pattern indicates a

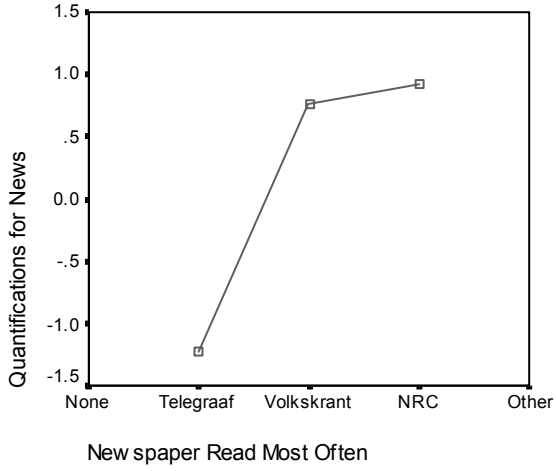
quadratic relationship between the original variable and the transformed variable. Using an alternative optimal scaling level is not suggested for *live*.

Figure 10.7 Transformation plot for variable live (nominal)



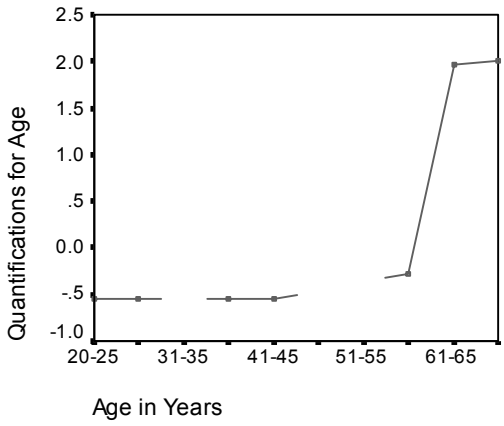
The quantifications for *news*, in contrast, correspond to an increasing trend across the three categories that have observed cases (Figure 10.8). The first category receives the lowest quantification, the second category receives a higher value, and the third category receives the highest value. Although the variable is scaled as nominal, the category order is retrieved in the quantifications.

Figure 10.8 Transformation plot for variable news (nominal)



In contrast, the transformation plot for *age* displays an S-shaped curve (Figure 10.9). The four youngest observed categories all receive the same negative quantification, whereas the two oldest categories receive similar positive values. Consequently, collapsing all of the younger ages into one common category (that is, below 50) and collapsing the two oldest categories into one may be attempted. However, the exact equality of the quantifications for the younger groups indicates that restricting the order of the quantifications to the order of the original categories may not be desirable. Because the quantifications for the 26–30, 36–40, and 41–45 groups cannot be lower than the quantification for the 20–25 group, these values are set equal to the boundary value. Allowing these values to be smaller than the quantification for the youngest group (that is, treating *age* as nominal) may improve the fit. So although *age* may be considered an ordinal variable, treating it as such does not appear appropriate in this case. Moreover, treating *age* as numerical, and thus maintaining the distances between the categories, would substantially reduce the fit.

Figure 10.9 Transformation plot for variable age (ordinal)



Single versus Multiple Category Coordinates

For every variable treated as single nominal, ordinal, or numerical, quantifications, single category coordinates, and multiple category coordinates are determined. These statistics for *age* are presented in Figure 10.10.

Figure 10.10 Coordinates for variable Age

	Marginal Frequency	Quantification	Single Category Coordinates		Multiple Category Coordinates	
			Dimension		Dimension	
			1	2	1	2
20-25	3	-.543	-.381	-.412	-.189	-.122
26-30	5	-.543	-.381	-.412	-.415	-.589
31-35	0	.000				
36-40	1	-.543	-.381	-.412	-.324	-.718
41-45	1	-.543	-.381	-.412	-.359	-.533
46-50	0	.000				
51-55	0	.000				
56-60	2	-.273	-.192	-.207	-.464	.045
61-65	1	1.966	1.379	1.491	1.742	1.155
66-70	2	2.005	1.406	1.520	1.256	1.659
Missing	0					

Every category for which no cases were recorded receives a quantification of 0. For *age*, this includes the 31–35, 46–50, and 51–55 categories. These categories are not restricted to be ordered with the other categories and do not affect any computations.

For multiple nominal variables, each category receives a different quantification on each dimension. For all other transformation types, a category has only one quantification, regardless of the dimensionality of the solution. The single category coordinates represent the locations of the categories on a line in the object space and equal the quantifications multiplied by the weights. For example, in Figure 10.10, the single category coordinates for category 8 (–0.192, –0.207) are the quantification multiplied by the weights (see Figure 10.3).

The multiple category coordinates for variables treated as single nominal, ordinal, or numerical represent the coordinates of the categories in the object space before ordinal or linear constraints are applied. These values are unconstrained minimizers of the loss. For multiple nominal variables, these coordinates represent the quantifications of the categories.

The effects of imposing constraints on the relationship between the categories and their quantifications are revealed by comparing the single with the multiple category coordinates. On the first dimension, the multiple category coordinates for *age* decrease to category 2 and remain relatively at the same level until category 9, at which point a dramatic increase occurs. A similar pattern is evidenced for the second dimension. These relationships are removed in the single category coordinates, in which the ordinal

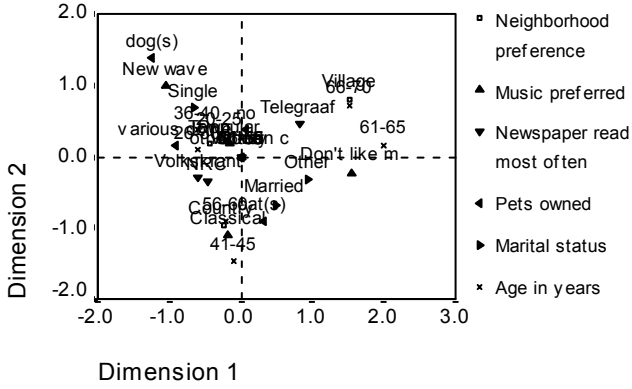
constraint is applied. On both dimensions, the coordinates are now nondecreasing. The differing structure of the two sets of coordinates suggests that a nominal treatment may be more appropriate.

Centroids and Projected Centroids

Figure 10.11 shows the plot of centroids labeled by variables. This plot should be interpreted in the same way as the category quantifications plot in homogeneity analysis or the multiple category coordinates in nonlinear principal components analysis. By itself, such a plot shows how well variables separate groups of objects (the centroids are in the center of gravity of the objects).

Notice that the categories for *age* are not separated very clearly. The younger *age* categories are grouped together at the left of the plot. As suggested previously, ordinal may be too strict a scaling level to impose on *age.*

Figure 10.11 Centroids labeled by variables



When you request centroid plots, individual centroid and projected centroid plots for each variable labeled by value labels are also produced. The projected centroids are on a line in the object space. Figure 10.12, Figure 10.13, and Figure 10.14 are the plots of centroids and projected centroids for *age*, *news*, and *live*, the variables of the first direction in the loadings plot.

Figure 10.12 Centroids and projected centroids for variable Age

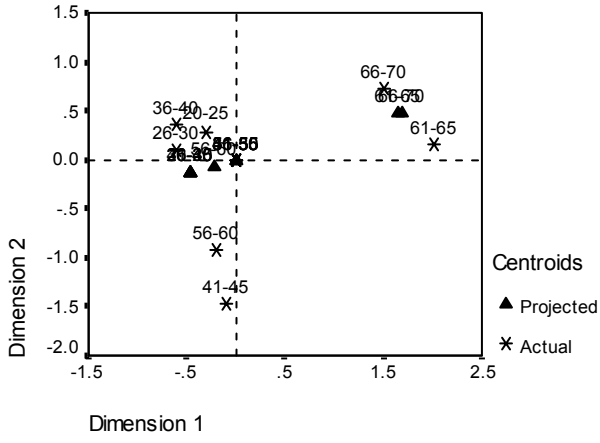


Figure 10.13 Centroids and projected centroids for variable News

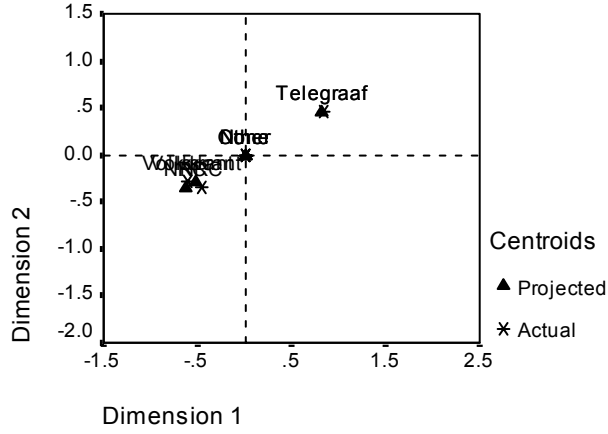
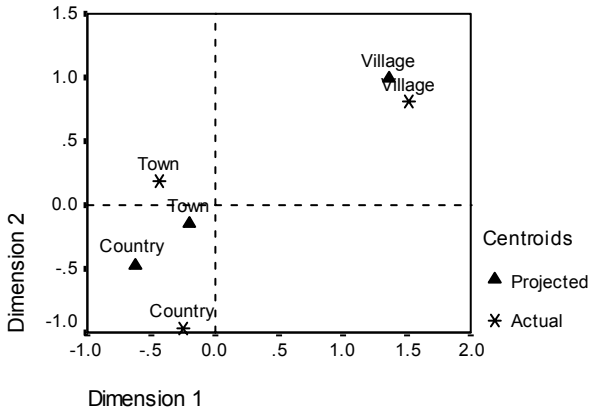


Figure 10.14 Centroids and projected centroids for variable Live



The actual centroids are projected onto the vectors defined by the component loadings. These vectors have been added to the centroid plots to aid in distinguishing the projected from the actual centroids. The projected centroids fall into one of four quadrants formed by extending two perpendicular reference lines through the origin. The interpretation of the direction of single nominal, ordinal, or numerical variables is obtained from the position of the projected centroids. For example, the variable *news* is specified as single nominal. The projected centroids show that *Volkskrant* and *NRC* are contrasted with *Telegraaf*.

The problem with *age* is evident from the projected centroids. Treating *age* as ordinal implies that the order of the age groups must be preserved. To satisfy this restriction, all age groups below age 45 are projected into the same point. Along the direction defined by *age*, *news*, and *live*, there is no separation of the younger age groups. Such a finding suggests treating the variable as nominal.

To understand the relationships among variables, find out what the specific categories (values) are for clusters of categories in the centroid plots. The relationships among *age*, *news*, and *live* can be described by looking at the upper right and lower left of the plots. In the upper right, the *age* groups are the older respondents (61–65 and 66–70) (Figure 10.12); they read the newspaper *Telegraaf* (Figure 10.13) and prefer living in a *Village* (Figure 10.14). Looking at the lower-left corner of each plot, you see that the younger to middle-aged respondents read the *Volkskrant* or *NRC* and want to live in the *Country* or in a *Town*. However, separating the younger groups is very difficult.

The same types of interpretations can be made about the other direction (*music*, *marital*, and *pet*) by focusing on the upper left and the lower right of the centroid plots. In the upper left corner, we find that single people tend to have dogs and like new wave music. The *married* and *other* categories for *marital* have cats; the former group prefers classical music and the latter group does not like music.

An Alternative Analysis

The results of the analysis suggest that treating *age* as ordinal does not appear appropriate. Although *age* is measured at an ordinal level, its relationships with other variables are not monotonic. To investigate the effects of changing the optimal scaling level to single nominal, you may rerun the analysis. Recall the Nonlinear Canonical Correlation Analysis dialog box:

Set 1

Select *age*. Click *Define Range and Scale.*

Optimal Scaling Level
⊙ Single nominal

Options

Display
☐ Frequencies (deselect)
☐ Category quantifications (deselect)
☐ Weights and component loadings (deselect)

Plot

☐ Object scores
☐ Component loadings

The eigenvalues for a two-dimensional solution are 0.807 and 0.757 respectively, for a total fit of 1.564.

Figure 10.15 Eigenvalues for the two-dimensional solution

		Dimension		
		1	2	Sum
Loss	Set 1	.246	.116	.362
	Set 2	.165	.431	.596
	Set 3	.168	.183	.352
	Mean	.193	.243	.436
Eigenvalue		.807	.757	
Fit				1.564

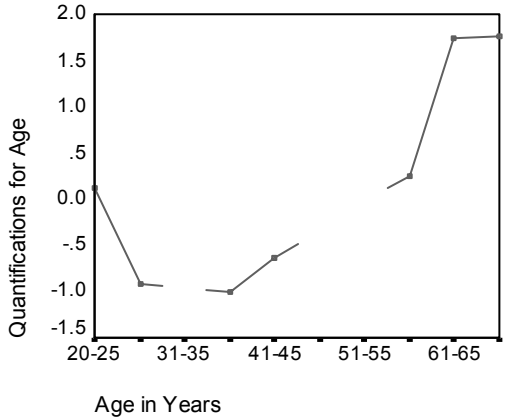
The multiple-fit and single-fit tables are presented in Figure 10.16. *Age* is still a highly discriminating variable, as evidenced by the sum of the multiple-fit values. In contrast to the earlier results, however, examination of the single-fit values reveals the discrimination to be almost entirely along the second dimension.

Figure 10.16 Partitioning fit and loss

		Multiple Fit			Single Fit			Single Loss		
		Dimension			Dimension			Dimension		
Set		1	2	Sum	1	2	Sum	1	2	Sum
1	Age in years	.336	1.037	1.373	.293	1.025	1.318	.043	.012	.055
	Marital status	.185	1.158	1.343	.184	1.158	1.342	.001	.000	.001
2	Pets owned	.493	.403	.896						
	Newspaper read most often	.685	.160	.845	.678	.119	.797	.007	.040	.047
3	Music preferred	.491	.560	1.051	.490	.558	1.048	.002	.002	.004
	Neighborhood preference	.137	.782	.919	.137	.782	.919	.000	.000	.000

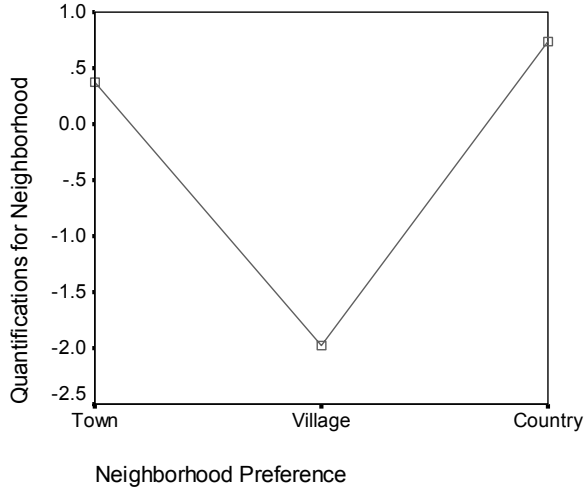
Figure 10.17 displays the transformation plot for *age*. The quantifications for a nominal variable are unrestricted, so the nondecreasing trend displayed when *age* was treated ordinally is no longer present (see Figure 10.9). We find a decreasing trend until the age of 40 and an increasing trend thereafter, corresponding to a U-shaped (quadratic) relationship. The two older categories still receive similar scores, and subsequent analyses may involve combining these categories.

Figure 10.17 Transformation plot for variable Age (nominal)



The transformation plot for *live* is given in Figure 10.18. Treating *age* as nominal does not affect the quantifications for *live* to any significant degree. The middle category receives the smallest quantification, with the extremes receiving large positive values.

Figure 10.18 Transformation plot for variable Live (age nominal)



A change is found in the transformation plot for *news* in Figure 10.19. Previously (Figure 10.8), an increasing trend was present in the quantifications, possibly suggesting an ordinal treatment for this variable. However, treating *age* as nominal removes this trend from the *news* quantifications.

Figure 10.19 Transformation plot for variable News (age nominal)

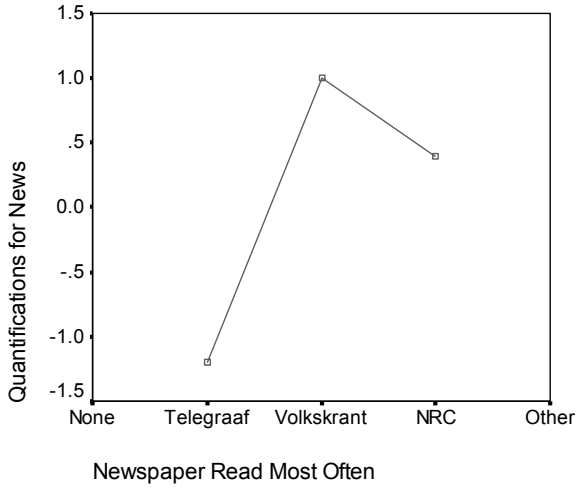
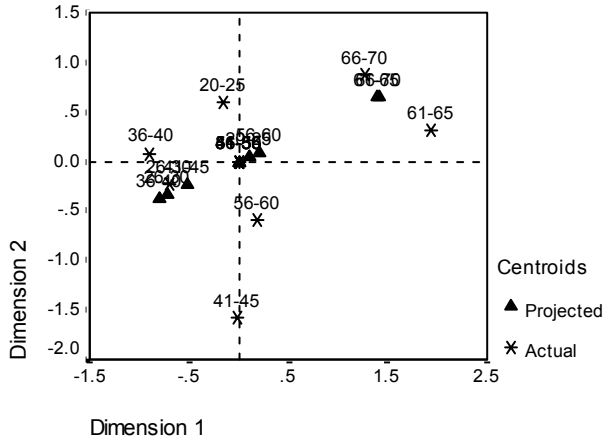


Figure 10.20 displays the centroid plot for *age*. Notice that the categories do not fall in chronological order along the line joining the projected centroids. The 20–25 group is situated in the middle rather than at the end. The spread of the categories is much improved over the ordinal counterpart presented previously.

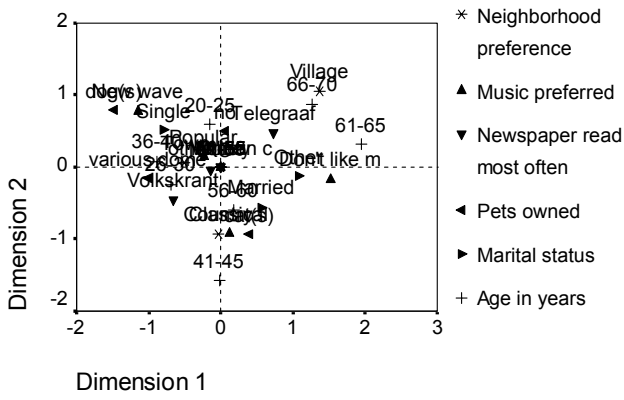
Figure 10.20 Centroids and projected centroids for variable Age (nominal)



Interpretation of the younger age groups is now possible from the centroid plot given in Figure 10.21. The *Volkskrant* and the *NRC* categories are also further apart than in the previous analysis, allowing for separate interpretations of each. The groups between the ages of 26 and 45 read the *Volkskrant* and prefer *Country* living. The 20–25 and 56–60 age groups read the *NRC*; the former group prefers to live in a *Town*, and the latter group prefers *Country* living. The oldest groups read the *Telegraaf* and prefer *Village* living.

Interpretation of the other direction (*music*, *marital*, and *pet*) is basically unchanged from the previous analysis. The only obvious difference is that people with a marital status of *other* have either cats or no pets.

Figure 10.21 Centroids labeled by variables (age nominal)



General Suggestions

Once you have examined the initial results, you will probably want to refine your analysis by changing some of the specifications on the nonlinear canonical correlation analysis. Here are some tips for structuring your analysis:

- Create as many sets as possible. Put an important variable that you want to predict in a separate set by itself.
- Put variables that you consider predictors together in a single set. If there are many predictors, try to partition them into several sets.
- Put each multiple nominal variable in a separate set by itself.
- If variables are highly correlated to each other and you don't want this relationship to dominate the solution, put those variables together in the same set.

11 Correspondence Analysis Examples

Correspondence analysis analyzes correspondence tables. A **correspondence table** is any two-way table whose cells contain some measurement of correspondence between the rows and the columns. The measure of correspondence can be any indication of the similarity, affinity, confusion, association, or interaction between the row and column variables. A very common type of correspondence table is a crosstabulation, where the cells contain frequency counts.

Such tables can be obtained easily with the crosstabs procedure. However, a crosstabulation does not always provide a clear picture of the nature of the relationship between the two variables. This is particularly true if the variables of interest are nominal (with no inherent order or rank) and contain numerous categories. Crosstabulation may tell you that the observed cell frequencies differ significantly from the expected values in a 10×9 crosstabulation of *occupation* and *breakfast cereal*, but it may be difficult to discern which occupational groups have similar tastes or what those tastes are.

Correspondence analysis allows you to examine the relationship between two nominal variables graphically in a multidimensional space. It computes row and column scores and produces plots based on the scores. Categories that are similar to each other appear close to each other in the plots. In this way, it is easy to see which categories of a variable are similar to each other or which categories of the two variables are related. The correspondence analysis procedure also allows you to fit supplementary points into the space defined by the active points.

If the ordering of the categories according to their scores is undesirable or counterintuitive, order restrictions can be imposed by constraining the scores for some categories to be equal. For example, suppose you expect the variable *smoking behavior* with categories *none*, *light*, *medium* and *heavy* to have scores which correspond to this ordering. However, if the analysis orders the categories *none*, *light*, *heavy* and *medium*, constraining the scores for *heavy* and *medium* to be equal preserves the ordering of the categories in their scores.

The interpretation of correspondence analysis in terms of distances depends on the normalization method used. The correspondence analysis procedure can be used to analyze either the differences between categories of a variable or differences between variables. With the default normalization, it analyzes the differences between the row and column variables.

The correspondence analysis algorithm is capable of many kinds of analyses. Centering the rows and columns and using chi-square distances corresponds to standard correspondence analysis. However, using alternative centering options combined with Euclidean distances allows for an alternative representation of a matrix in a low-dimensional space.

Three examples will be presented. The first employs a relatively small correspondence table and illustrates the concepts inherent in correspondence analysis. The second example demonstrates a practical marketing application. The final example uses a table of distances in a multidimensional scaling approach.

Example 1: Smoking Behavior by Job Category

The aim of correspondence analysis is to show the relationships between the rows and columns of a correspondence table. You will use a hypothetical table introduced by Greenacre (1984) to illustrate the basic concepts. This data set can be found in *smoking.sav*. Figure 11.1 shows the distribution of smoking behavior for five levels of job category. The rows of the correspondence table represent the job categories. The columns of the correspondence table represent the smoking behavior.

Figure 11.1 Correspondence table

	Smoking						
Staff Group	None	Light	Medium	Heavy	No Alcohol[1]	Alcohol[1]	Active Margin
Sr Managers	4	2	3	2	0	11	11
Jr Managers	4	3	7	4	1	17	18
Sr Employees	25	10	12	4	5	46	51
Jr Employees	18	24	33	13	10	78	88
Secretaries	10	6	7	2	7	18	25
National Average[2]	42	29	20	9			
Active Margin	61	45	62	25			193

1. Supplementary column
2. Supplementary row

In addition, the table contains one supplementary row and two supplementary columns. The supplementary row identifies the percentage of people in each of the smoking categories nationwide. The two supplementary columns contain the number of people in each staff category who do not drink alcohol and the number of people who do. Supplementary rows and columns do not influence the analysis and are not part of the marginal sums.

The marginal row totals show that the company has far more employees, both junior and senior, than managers and secretaries. However, the distribution of senior and junior positions for the managers is approximately the same as the distribution of senior and junior positions for the employees. Looking at the column totals, you see that there are similar numbers of nonsmokers and medium smokers. Furthermore, heavy smokers are outnumbered by each of the other three categories. But what, if anything, do any of these

job categories have in common regarding smoking behavior? And what is the relationship between job category and smoking?

Before you can answer these questions with correspondence analysis, the setup of the data requires that the cases be weighted by the variable *count*. To do this, from the menus choose:

Data
 Weight Cases...

⊙ Weight cases by
▸ Frequency Variable: *count*

Then, to obtain a correspondence analysis in three dimensions using row principal normalization, from the menus choose:

Analyze
 Data Reduction
  Correspondence Analysis...

▸ Row: *staff*
▸ Column: *smoke*

Select *staff*. Click *Define Range*.

 Category range for row variable: staff
 Minimum value: 1
 Maximum value: 5
 Click *Update*.

Select *smoke*. Click *Define Range*.

 Category range for row variable: smoke
 Minimum value: 1
 Maximum value: 4
 Click *Update*.

Model...

 Dimensions in solution: 3
 Normalization Method
 ⊙ Row principal

Statistics...

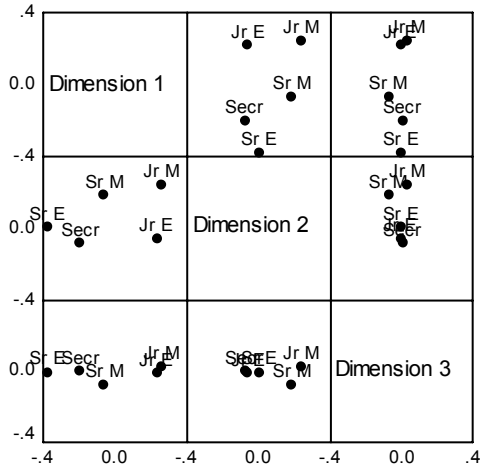
 ☑ Row profiles
 ☑ Column profiles

Plots...

 Scatterplots
 ☑ Row points

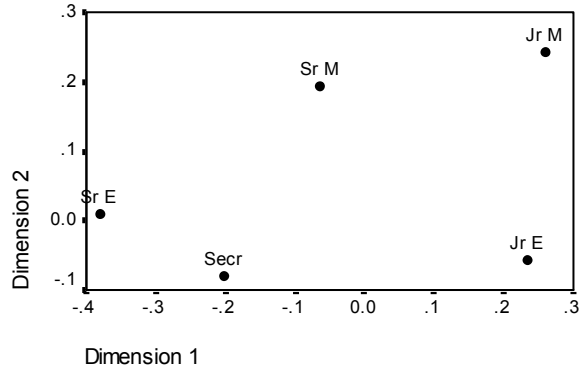
Correspondence analysis generates a variety of plots that graphically illustrate the underlying relationships between categories and between variables. Figure 11.2 shows the scatterplot matrix of row scores for a three-dimensional solution.

Figure 11.2 Scatterplot matrix of row scores (row principal normalization)



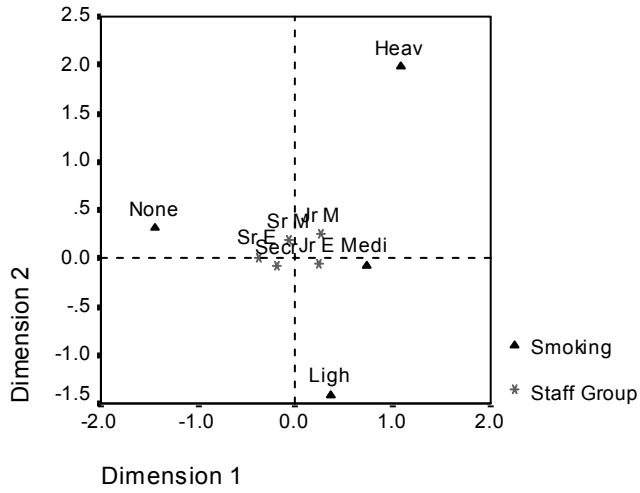
Scatterplot matrices can easily be converted to two- or three-dimensional scatterplots using the chart gallery available through the Chart Editor. Figure 11.3 displays the two-dimensional plot of the row scores for the first two dimensions. The remainder of this chapter uses two-dimensional plots derived from scatterplot matrices.

Figure 11.3 Two-dimensional plot of row column (row principal normalization)



An important choice in correspondence analysis is the normalization method. Although solutions under different choices of normalization are completely equivalent in terms of fit (the singular values), the plots, among other things, can be quite different. Figure 11.4 shows the correspondence analysis plot of row and column scores for the first two dimensions. You use row principal normalization to focus on the differences or similarities between job categories. Row principal normalization results in the Euclidean distance between a row point and the origin, approximating the chi-square distance between the row category and the average row category. Moreover, the Euclidean distance between any two points in the plot approximate the chi-square distance between the corresponding rows of the correspondence table. The chi-square distance is a weighted Euclidean distance, where the weights equal the masses.

Figure 11.4 Plot of row and column scores (row principal normalization)



The interpretation of the plot is fairly simple—row points that are close together are more alike than row points that are far apart. In Figure 11.4, you see that *secretaries* and *senior employees* are plotted near each other. This indicates that *secretaries* and *senior employees* are similar in their smoking behavior. *Junior managers* are relatively far from *senior employees* and are therefore very unlike them.

Although the distances between column points are artificially exaggerated by the row principal normalization, you can still get a general idea about the relationship between the row and column variables from this joint plot. If you draw a line from the origin to each column point (*smoke*) and then make an orthogonal projection (perpendicular line) from the row points (*staff categories*) to these lines, the distance from the intersection of the two lines to the column point gives you an indication of how categories of the two variables are related to each other. For example, vectors for *none* and *heavy* smoking help describe the relationship between each of these categories and *staff*. You can see

that *senior employees* are closest to *none*, followed by *secretaries* and *senior managers*. *Junior employees* are farthest from *none*. In contrast, *junior managers* are closest to *heavy*, followed by *senior managers* and *junior employees*. Similar interpretations are possible for the other two smoking categories. This order is identical for all normalization methods but principal normalization.

Profiles and Distances

To determine the distance between categories, correspondence analysis considers the marginal distributions as well as the individual cell frequencies. It computes **row** and **column profiles**, which give the row and column proportions for each cell, based on the marginal totals. Figure 11.5 shows the row profiles for this example.

Figure 11.5 Row profiles (row principal normalization)

	Smoking				
Staff Group	None	Light	Medium	Heavy	Active Margin
Senior Managers	.364	.182	.273	.182	1.000
Junior Managers	.222	.167	.389	.222	1.000
Senior Employees	.490	.196	.235	.078	1.000
Junior Employees	.205	.273	.375	.148	1.000
Secretaries	.400	.240	.280	.080	1.000
Mass	.316	.233	.321	.130	

The row profiles indicate the proportion of the row category in each column category. For example, among the senior employees, most are nonsmokers and very few are heavy smokers. In contrast, among the junior managers, most are medium smokers and very few are light smokers.

Figure 11.6 contains the column profiles. These values indicate the proportion of the column in each row category. For example, most of the light smokers are junior employees. Similarly, most of the medium and heavy smokers are junior employees. Recall that the sample contains predominantly junior employees. It is not surprising that this staff category dominates the smoking categories.

Figure 11.6 Column profiles

	Smoking				
Staff Group	None	Light	Medium	Heavy	Mass
Sr Managers	.066	.044	.048	.080	.057
Jr Managers	.066	.067	.113	.160	.093
Sr Employees	.410	.222	.194	.160	.264
Jr Employees	.295	.533	.532	.520	.456
Secretaries	.164	.133	.113	.080	.130
Active Margin	1.000	1.000	1.000	1.000	

If you think of difference in terms of distance, then the greater the difference between row profiles, the greater the distance between points in a plot. The goal of correspondence analysis with row principal normalization is to find a configuration in which Euclidean distances between row points in the full dimensional space equal the chi-square distances between rows of the correspondence table. In a reduced space, the Euclidean distances approximate the chi-square distances.

Chi-square distances are weighted profile distances. These weighted distances are based on the concept of mass. **Mass** is a measure that indicates the influence of an object based on its marginal frequency. Mass affects the **centroid**, which is the weighted mean row or column profile. The **row centroid** is the mean row profile. Points with a large mass, like *junior employees*, pull the centroid strongly to their location. A point with a small mass, like *senior managers*, pulls the row centroid only slightly to its location.

Inertia

If the entries in the correspondence table are frequencies and row principal normalization is used, then the weighted sum over all squared distances between the row profiles and the mean row profile equals the chi-square statistic. Euclidean distances between row points in the plot approximate chi-square distances between rows of the table.

The **total inertia** is defined as the weighted sum of all squared distances to the origin divided by the total over all cells, where the weights are the masses. Rows with a small mass influence the inertia only when they are far from the centroid. Rows with a large mass influence the total inertia, even when they are located close to the centroid. The same applies to columns.

Row and Column Scores

The row and column scores are the coordinates of the row and column points in Figure 11.4. Figure 11.7 and Figure 11.8 show the row and column scores, respectively.

Figure 11.7 Row scores (row principal normalization)

Staff Group	Mass	Score in Dimension 1	2	3	Inertia
Sr Managers	.057	-.066	.194	-.071	.003
Jr Managers	.093	.259	.243	.034	.012
Sr Employees	.264	-.381	.011	.005	.038
Jr Employees	.456	.233	-.058	-.003	.026
Secretaries	.130	-.201	-.079	.008	.006
Active Total	1.000				.085

Figure 11.8 Column scores (row principal normalization)

Smoking	Mass	Score in Dimension 1	2	3	Inertia
None	.316	-1.438	.305	.044	.049
Light	.233	.364	-1.409	-1.082	.007
Medium	.321	.718	-.074	1.262	.013
Heavy	.130	1.074	1.976	-1.289	.016
Active Total	1.000				.085

For row principal normalization, geometrically, the column scores are proportional to the weighted centroid of the active row points. The row points are in the weighted centroid of the active column points, where the weights correspond to the entries in the row profiles table. For example, the score of –0.066 for *senior managers* on the first dimension equals (see Figure 11.5 for the row profile):

$$(-1.438 \times 0.364) + (0.364 \times 0.182) + (0.718 \times 0.273) + (1.074 \times 0.182)$$

When the row points are the weighted average of the column points and the maximum dimensionality is used, the Euclidean distance between a row point and the origin equals the chi-square distance between the row and the average row. For example, the chi-square distance between the row profile for *secretaries* and the row centroid is:

$$\sqrt{\frac{(0.400-0.316)^2}{0.316}+\frac{(0.240-0.233)^2}{0.233}+\frac{(0.280-0.321)^2}{0.321}+\frac{(0.080-0.130)^2}{0.130}} = 0.217$$

The Euclidean distance from the *secretaries* point to the origin is:

$$\sqrt{(-0.201)^2 + (-0.079)^2 + 0.008^2} = 0.216$$

Inertia of a row equals the weighted chi-squared distance to the average row. With row principal normalization, inertia of a row point equals the weighted squared Euclidean distance to the origin in the full dimensional space, where the weight is the mass. Figure 11.7 and Figure 11.8 display the inertias for all points. These inertias sum to the total inertia across rows and columns. Because the chi-square statistic is equivalent to the total inertia times the sum of all cells of the correspondence table, you can think of the orientation of the row points as a pictorial representation of the chi-square statistic. For other normalization methods, interpretations differ and are discussed later.

Dimensionality

Ideally, you want a correspondence analysis solution that represents the relationship between the row and column variables in as few dimensions as possible. But it is frequently useful to look at the maximum number of dimensions to see the relative contribution of each dimension. The maximum number of dimensions for a correspondence analysis solution equals the number of active rows minus 1 or the number of active columns minus 1, whichever is less. An active row or column is one for which a distinct set of scores is found. Supplementary rows or columns are not active. If two row or column categories are constrained to be equal, one set of scores is determined for both. Consequently, each equality constraint is equivalent to one active row or column. In the present example, the maximum number of dimensions is $min(5, 4) - 1$, or 3.

The first dimension displays as much of the inertia as possible, the second is orthogonal to the first and displays as much of the remaining inertia as possible, and so on. It is possible to split the total inertia into components attributable to each dimension. You can then evaluate the inertia shown by a particular dimension by comparing it to the total inertia. For example, Figure 11.9 shows that the first dimension displays 87.8% (0.075/0.085) of the total inertia, whereas the second dimension displays only 11.8% (0.010/0.085).

Figure 11.9 Inertia per dimension

Dimension	Singular Value	Inertia	Chi Square
1	.273	.075	
2	.100	.010	
3	.020	.000	
Total		.085	16.442

If you decide that the first p dimensions of a q dimensional solution show enough of the total inertia, then you do not have to look at higher dimensions. In this example, you might decide to omit the last dimension, knowing that it represents less than 1.0% of the total inertia.

The singular values shown in Figure 11.9 can be interpreted as the correlation between the row and column scores. They are analogous to the Pearson correlation coefficient *(r)* in correlation analysis. For each dimension, the singular value squared (eigenvalue) equals the inertia and thus is another measure of the importance of that dimension.

Supplementary Profiles

In correspondence analysis, additional categories can be represented in the space describing the relationships between the active categories. A supplementary profile defines a profile across categories of either the row or column variable and does not influence the analysis in any way. Figure 11.1 contains one supplementary row and two supplementary columns.

The national average of people in each smoking category defines a supplementary row profile. The two supplementary columns define two column profiles across the categories of staff. The supplementary profiles define a point in either the row space or the column space. Because you will focus on both the rows and the columns separately, you will use principal normalization.

To add the supplementary categories and obtain a principal normalization solution, recall the Correspondence Analysis dialog box:

Select *staff*. Click *Define Range.*

Category range for row variable: staff
Maximum value: 6
Click *Update*.

Category Constraints
Select 6.
⊙ Category is supplemental

Select *smoke*. Click *Define Range.*

Category range for column variable: smoke
Maximum value: 6
Click *Update*.

Category Constraints
Select 5.
⊙ Category is supplemental
Select 6.
⊙ Category is supplemental

Model...
Dimensions in solution: 2

Normalization Method
⊙ Principal

Statistics...
☐ Correspondence table (deselect)
☐ Overview of row points (deselect)
☐ Overview of column points (deselect)
☐ Row profiles (deselect)
☐ Column profiles (deselect)

Plots...
Scatterplots
☑ Biplot (grayed out)
☑ Column points

Figure 11.10 shows the first two dimensions for the row points with the supplementary point for *national average*. *National average* lies far from the origin, indicating that the sample is not representative of the nation in terms of smoking behavior. *Secretaries* and *senior employees* are close to the national average, whereas *junior managers* are not. Thus, secretaries and senior employees have smoking behaviors similar to the national average, but junior managers do not.

Figure 11.10 Row points (principal normalization)

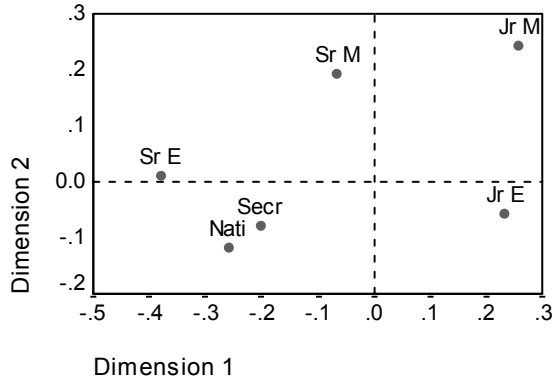
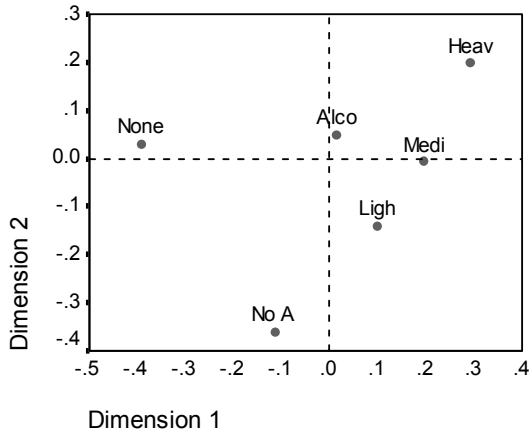


Figure 11.11 displays the column space with the two supplementary points for alcohol consumption. *Alcohol* lies near the origin, indicating a close correspondence between the alcohol profile and the average column profile (see Figure 11.6). However, *no alcohol* differs from the average column profile, illustrated by the large distance from the origin. The closest point to *no alcohol* is *light smokers*. The light profile is most similar to

the nondrinkers. Among the smokers, *medium* is next closest and *heavy* is farthest. Thus, there is a progression in similarity to nondrinking from light to heavy smoking. However, the relatively high proportion of secretaries in the no alcohol group prevents any close correspondence to any of the smoking categories.

Figure 11.11 Column points (principal normalization)



Contributions

As discussed in the section on "Dimensionality" above, it may be better to compute a solution in two dimensions for this data. To compute a two-dimensional solution with row principal normalization and supplementary categories, recall the Correspondence Analysis dialog box:

Model...
- Normalization Method
- ⊙ Row Principal

Statistics...
- ☑ Correspondence table
- ☑ Overview of row points
- ☑ Overview of column points
- ☑ Permutation of the correspondence table
- Maximum dimension for permutations: 1

- Confidence statistics for
- ☑ Row points
- ☑ Column points

Plots...
Scatterplots
☐ Biplot (deselect)
☐ Row points (deselect)
☐ Column points (deselect)

It is possible to compute the inertia displayed by a particular dimension. The scores on each dimension correspond to an orthogonal projection of the point onto that dimension. Thus, the inertia for a dimension equals the weighted sum of the squared distances from the scores on the dimension to the origin. However, whether this applies to row or column scores (or both) depends on the normalization method used. Each row and column point contributes to the inertia. Row and column points that contribute substantially to the inertia of a dimension are important to that dimension. The contribution of a point to the inertia of a dimension is the weighted squared distance from the projected point to the origin divided by the inertia for the dimension. Figure 11.12 and Figure 11.13 show these contributions for the row and column points respectively for a two-dimensional representation.

Figure 11.12 Contributions of row points (row principal normalization)

	Contribution	
	Of Point to Inertia of Dimension	
Staff Group	1	2
Sr Managers	.003	.214
Jr Managers	.084	.551
Sr Employees	.512	.003
Jr Employees	.331	.152
Secretaries	.070	.081
National Average[1]	.000	.000
Active Total	1.000	1.000

1. Supplementary point

Figure 11.13 Contributions of column points (row principal normalization)

	Contribution	
	Of Point to Inertia of Dimension	
Smoking	1	2
None	.654	.029
Light	.031	.463
Medium	.166	.002
Heavy	.150	.506
No Alcohol[1]	.000	.000
Alcohol[1]	.000	.000
Active Total	1.000	1.000

1. Supplementary point

The diagnostics that measure the contributions of points are an important aid in the interpretation of a correspondence analysis solution. Dominant points in the solution can easily be detected. For example, *senior employees* and *junior employees* are dominant points in the first dimension, contributing 84% of the inertia. Among the column points, *none* contributes 65% of the inertia for the first dimension alone.

The contribution of a point to the inertia of the dimensions depends on both the mass and the distance from the origin. Points that are far from the origin and have a large mass contribute most to the inertia of the dimension. Because supplementary points do not play any part in defining the solution, they do not contribute to the inertia of the dimensions.

In addition to examining the contribution of the points to the inertia per dimension, you can examine the contribution of the dimensions to the inertia per point. You can examine how the inertia of a point is spread over the dimensions by computing the percentage of the point inertia contributed by each dimension. Figure 11.14 and Figure 11.15 display these contributions.

Figure 11.14 Contributions of dimensions to the row point inertias

	Contribution		
	Of Dimension to Inertia of Point		
Staff Group	1	2	Total
Sr Managers	.092	.800	.893
Jr Managers	.526	.465	.991
Sr Employees	.999	.001	1.000
Jr Employees	.942	.058	1.000
Secretaries	.865	.133	.999
National Average[1]	.631	.131	.761
Active Total			

1. Supplementary point

Figure 11.15 Contributions of dimensions to the column point inertias

	Contribution		
	Of Dimension to Inertia of Point		
Smoking	1	2	Total
None	.994	.006	1.000
Light	.327	.657	.984
Medium	.982	.001	.983
Heavy	.684	.310	.995
No Alcohol[1]	.040	.398	.439
Alcohol[1]	.040	.398	.439
Active Total			

1. Supplementary point

Notice that the contributions of the dimensions to the point inertias do not all sum to one. In a reduced space, the inertia that is contributed by the higher dimensions is not represented. Using the maximum dimensionality would reveal the unaccounted inertia amounts.

In Figure 11.14, the first two dimensions contribute all of the inertia for *senior employees* and *junior employees*, and virtually all of the inertia for *junior managers* and *secretaries*. For senior managers, 11% of the inertia is not contributed by the first two dimensions. Two dimensions contribute a very large proportion of the inertias of the row points.

Similar results occur for the column points in Figure 11.15. For every active column point, two dimensions contribute at least 98% of the inertia. The third dimension contributes very little to these points. The low totals for the supplementary column points indicate that these points are not very well represented in the space defined by the active points. Including these points in the analysis as active might result in quite a different solution.

Permutations of the Correspondence Table

Sometimes it is useful to order the categories of the rows and the columns. For example, you might have reason to believe that the categories of a variable correspond to a certain order, but you don't know the precise order. This ordination problem is found in various disciplines—the seriation problem in archaeology, the ordination problem in phytosociology, and Guttman's scalogram problem in the social sciences. Ordering can be achieved by taking the row and column scores as ordering variables. If you have row and column scores in p dimensions, p permuted tables can be made. When the first singular value is large, the first table will show a particular structure, with larger-than-expected relative frequencies close to the "diagonal."

Figure 11.16 shows the permutation of the correspondence table along the first dimension for the example. Looking at the row scores for dimension 1 in Figure 11.7, you can see that the ranking from lowest to highest is *senior employees* (–0.381), *national average* (–0.258), *secretaries* (–0.201), *senior managers* (–0.066), *junior employees* (0.233), and *junior managers* (0.259). Looking at the column scores for dimension 1 in Figure 11.8, you see that the ranking is *none*, *no alcohol*, *alcohol*, *light*, *medium*, and then *heavy*. These rankings are reflected in the ordering of the rows and columns of the table.

Figure 11.16 Permutation of the correspondence table

	Smoking						
Staff Group	None	No Alcohol[1]	Alcohol[1]	Light	Medium	Heavy	Active Margin
Sr Employees	25	5	46	10	12	4	51
National Average[2]	42			29	20	9	
Secretaries	10	7	18	6	7	2	25
Sr Managers	4	0	11	2	3	2	11
Jr Employees	18	10	78	24	33	13	88
Jr Managers	4	1	17	3	7	4	18
Active Margin	61			45	62	25	193

1. Supplementary column
2. Supplementary row

Confidence Statistics

Assuming that the table to be analyzed is a frequency table and that the data are a random sample from an unknown population, the cell frequencies follow a multinomial distribution. From this, it is possible to compute the standard deviations and correlations of the singular values, row scores, and column scores.

In a one-dimensional correspondence analysis solution, you can compute a confidence interval for each score in the population. If the standard deviation is large, correspondence analysis is very uncertain of the location of the point in the population. On the other hand, if the standard deviation is small, then the correspondence analysis is fairly certain that this point is located very close to the point given by the solution.

In a multidimensional solution, if the correlation between dimensions is large, it may not be possible to locate a point in the correct dimension with much certainty. In such cases, multivariate confidence intervals must be calculated using the variance/covariance matrix that can be written to a file.

The standard deviations for the singular values are 0.07 for the first dimension and 0.076 for the second dimension. These small values indicate that the correspondence analysis would produce the same solution for a slightly different sample from the same population. The fact that the first two singular values are very different is reflected in the small correlation of 0.02 between the two dimensions.

Figure 11.17 and Figure 11.18 show the confidence statistics for the row and column scores. The standard deviations for the rows are quite small, so you can conclude that the correspondence analysis has obtained an overall stable solution. The standard deviations for the column scores are much larger due to the row principal normalization. If you look at the correlations between the dimensions for the scores, you see that the correlations are small for the row scores and the column scores with one exception. The column scores for *none* have a correlation of 0.617. However, the correlations for the column scores can be inflated by using column principal normalization.

Figure 11.17 Confidence statistics for row scores

	Standard Deviation in Dimension		Correlation
Staff Group	1	2	1-2
Sr Managers	.321	.316	.101
Jr Managers	.248	.225	.067
Sr Employees	.102	.050	.046
Jr Employees	.081	.056	.350
Secretaries	.094	.070	-.184

Figure 11.18 Confidence statistics for column scores

	Standard Deviation in Dimension		Correlation
Smoking	1	2	1-2
None	.138	.442	.617
Light	.534	.861	.054
Medium	.328	1.044	.016
Heavy	.682	1.061	-.250

Normalization

Normalization is used to distribute the inertia over the row scores and column scores. Some aspects of the correspondence analysis solution, such as the singular values, the inertia per dimension, and the contributions, do not change under the various normalizations. The row and column scores and their variances are affected.

Correspondence analysis has several ways to spread the inertia. The three most common include spreading the inertia over the row scores only, spreading the inertia over the column scores only, or spreading the inertia symmetrically over both the row scores and the column scores. The normalization used in this example is called **row principal normalization**. In row principal normalization, the Euclidean distances between the row points approximate chi-square distances between the rows of the correspondence table. The row scores are the weighted average of the column scores. The column scores are standardized to have a weighted sum of squared distances to the centroid of 1. Since this method maximizes the distances between row categories, you should use row principal normalization if you are primarily interested in seeing how categories of the row variable differ from each other.

On the other hand, you might want to approximate the chi-square distances between the columns of the correspondence table. In that case, the column scores should be the weighted average of the row scores. The row scores are standardized to have a weighted sum of squared distances to the centroid of 1. This is called **column principal normalization**. This method maximizes the distances between column categories and should be used if you are primarily concerned with how categories of the column variable differ from each other.

You can also treat the rows and columns symmetrically. This normalization spreads inertia symmetrically over the rows and over the columns. The inertia is divided equally over the row scores and the column scores. Note that neither the distances between the row points nor the distances between the column points are approximations of chi-square distances in this case. This is called **symmetrical normalization**. Use this method if you are primarily interested in the differences or similarities between the two variables. Usually, this is the preferred method to make biplots.

A fourth option is called **principal normalization**, in which the inertia is spread twice in the solution, once over the row scores and once over the column scores. You should use this method if you are interested in the distances between the row points and the distances between the column points separately, but not in how the row and column points are related to each other. Biplots, such as the one in Figure 11.4, are not appropriate for this normalization option and are therefore not available if you have specified the principal normalization method.

The normalization options allow you to rescale the desired graphical representation in a flexible way. For example, to obtain a biplot for a two-dimensional solution with symmetrical normalization, recall the Correspondence Analysis dialog box:

Model...
- Normalization Method
- ⊙ Symmetrical

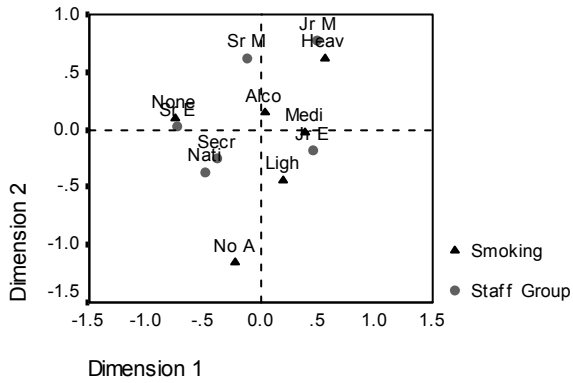
Statistics...
- ☐ Correspondence table (deselect)
- ☐ Overview of row points (deselect)
- ☐ Overview of column points (deselect)
- ☐ Permutation of the correspondence table (deselect)

- Confidence Statistics for
- ☐ Row points (deselect)
- ☐ Column points (deselect)

Plots...
- Scatterplots
- ☑ Biplot

Figure 11.19 shows the biplot of row and column points using symmetrical normalization. In this plot, *junior managers* are closest to *heavy* and *senior employees* are closest to *none*. However, this plot does not indicate the relationship between categories of the row variable or of the column variable.

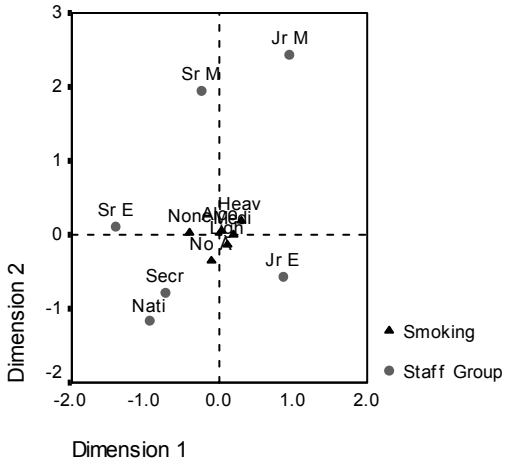
Figure 11.19 Biplot with symmetrical normalization



Row principal normalization results in the biplot in Figure 11.4. Notice that the column points have moved toward the edges and the row points have clustered about the origin. Row points are in the weighted centroid of the corresponding column points.

Figure 11.20 displays the biplot for column principal normalization. The column points have moved toward the origin and the row points have moved away from it. Here, column points are in the weighted centroid of the corresponding row points.

Figure 11.20 Biplot with column principal normalization



The most flexible way to spread the inertia involves dividing the inertia unevenly over the row and column scores. This option results in an expansion of one set of points and a contraction of the other set of points. The points in the resulting biplot are oriented somewhere between the corresponding points for row and column principal normalization. This option is particularly useful for constructing a tailor-made biplot.

Example 2: Perceptions of Coffee Brands

The previous example involved a small table of hypothetical data. Actual applications often involve much larger tables. In this example, you will use data introduced by Kennedy, Riquier, and Sharp (1996) pertaining to perceived images of six iced coffee brands. This data set can be found in *coffee.sav*.

For each of twenty-three iced coffee image attributes, people selected all brands that were described by the attribute. Table 11.1 contains the attributes and their corresponding labels. The six brands are denoted *AA*, *BB*, *CC*, *DD*, *EE*, and *FF* to preserve confidentiality.

Table 11.1 Iced coffee attributes

Image Attribute	Label	Image Attribute	Label
good hangover cure	cure	fattening brand	fattening
low fat/calorie brand	low fat	appeals to men	men
brand for children	children	South Australian brand	South Australian
working class brand	working	traditional/old fashioned brand	traditional
rich/sweet brand	sweet	premium quality brand	premium
unpopular brand	unpopular	healthy brand	healthy
brand for fat/ugly people	ugly	high caffeine brand	caffeine
very fresh	fresh	new brand	new
brand for yuppies	yuppies	brand for attractive people	attractive
nutritious brand	nutritious	tough brand	tough
brand for women	women	popular brand	popular
minor brand	minor		

The setup of the data requires that the cases be weighted by the variable *freq*. To do this, from the menus choose:

Data
  Weight Cases...

⊙ Weight cases by
▸ Frequency variable: *freq*

Principal Normalization

Initially, you will focus on how the attributes are related to each other and how the brands are related to each other. Using principal normalization spreads the total inertia once over the rows and once over the columns. Although this prevents biplot interpretation, the distances between the categories for each variable can be examined.

Dimensionality

In order to decide how many dimensions to use, you can find an initial solution in five dimensions and choose a number of dimensions that accounts for the bulk of the inertia. To obtain an initial solution in five dimensions with principal normalization, from the menus choose:

Analyze
  Data Reduction
    Correspondence Analysis...

▸ Row: *image*
▸ Column: *brand*

Select *image*. Click *Define Range*.
  Category range for row variable: image
  Minimum value: 1
  Maximum value: 23
  Click *Update*.

Select *brand*. Click *Define Range*.
  Category range for row variable: image
  Minimum value: 1
  Maximum value: 6
  Click *Update*.

Model...
  Dimensions in solution: 5

  Normalization Method
  ⊙ Principal

Figure 11.21 shows the decomposition of the total inertia along each dimension. Two dimensions account for 83% of the total inertia. Adding a third dimension adds only 8.6% to the accounted for inertia. Thus, you elect to use a two-dimensional representation.

Figure 11.21 Inertia per dimension

Dimension	Singular Value	Inertia	Chi Square	Proportion of Inertia	
				Accounted for	Cumulative
1	.711	.506		.629	.629
2	.399	.159		.198	.827
3	.263	.069		.086	.913
4	.234	.055		.068	.982
5	.121	.015		.018	1.000
Total		.804	3746.968	1.000	1.000

To compute a two-dimensional solution, recall the Correspondence Analysis dialog box:

Model...
Dimensions in solution: 2

Plots...
Scatterplots
☑ Row points
☑ Column points

Contributions

Figure 11.22 shows the contributions of the row points to the inertia of the dimensions and the contributions of the dimensions to the inertia of the row points. If all points contributed equally to the inertia, the contributions would be 0.043. *Healthy* and *low fat* both contribute a substantial portion to the inertia of the first dimension. *Men* and *tough* contribute the largest amounts to the inertia of the second dimension. Both *ugly* and *fresh* contribute very little to either dimension.

Figure 11.22 Attribute contributions

	Contribution				
	Of Point to Inertia of Dimension		Of Dimension to Inertia of Point		
IMAGE	1	2	1	2	Total
fattening	.042	.035	.652	.173	.825
men	.073	.219	.512	.480	.992
South Australian	.010	.044	.114	.152	.266
traditional	.039	.071	.454	.260	.715
premium	.016	.090	.296	.509	.805
healthy	.152	.010	.953	.020	.973
caffeine	.019	.005	.702	.053	.755
new	.086	.006	.893	.021	.914
attractive	.035	.001	.911	.007	.918
tough	.056	.246	.404	.560	.964
popular	.058	.001	.771	.003	.774
cure	.008	.011	.446	.209	.655
low fat	.175	.013	.941	.021	.962
children	.006	.041	.179	.380	.559
working	.055	.064	.693	.255	.948
sweet	.020	.112	.212	.368	.580
unpopular	.011	.005	.585	.085	.670
ugly	.000	.002	.000	.131	.131
fresh	.001	.002	.196	.214	.410
yuppies	.010	.019	.392	.246	.637
nutritious	.041	.001	.946	.006	.951
women	.062	.001	.965	.007	.972
minor	.027	.001	.593	.007	.600
Active Total	1.000	1.000			

Two dimensions contribute a large amount to the inertia for most row points. The large contributions of the first dimension to *healthy*, *new*, *attractive*, *low fat*, *nutritious*, and *women* indicate that these points are very well represented in one dimension. Consequently, the higher dimensions contribute little to the inertia of these points, which will lie very near the horizontal axis. The second dimension contributes most to *men*, *premium*, and *tough*. Both dimensions contribute very little to the inertia for *South Australian* and *ugly*, so these points are poorly represented.

Figure 11.23 displays the contributions involving the column points. Brands *CC* and *DD* contribute the most to the first dimension, whereas *EE* and *FF* explain a large amount of the inertia for the second dimension. *AA* and *BB* contribute very little to either dimension.

Figure 11.23 Brand contributions

	Contribution				
	Of Point to Inertia of Dimension		Of Dimension to Inertia of Point		
BRAND	1	2	1	2	Total
AA	.187	.003	.744	.004	.748
BB	.021	.134	.135	.272	.407
CC	.362	.007	.951	.006	.957
DD	.267	.010	.928	.011	.939
EE	.127	.477	.420	.494	.914
FF	.036	.369	.169	.550	.718
Active Total	1.000	1.000			

In two dimensions, all brands but *BB* are well represented. *CC* and *DD* are represented well in one dimension. The second dimension contributes the largest amounts for *EE* and *FF*. Notice that *AA* is represented well in the first dimension, but does not have a very high contribution to that dimension.

Plots

Figure 11.24 displays the plot of the row points. *Fresh* and *ugly* are both very close to the origin, indicating that they differ little from the average row profile. Three general classifications emerge. Located in the upper left of the plot, *tough*, *men*, and *working* are all similar to each other. The lower left contains *sweet*, *fattening*, *children*, and *premium*. In contrast, *healthy*, *low fat*, *nutritious*, and *new* cluster on the right side of the plot.

Figure 11.24 Plot of image attributes (principal normalization)

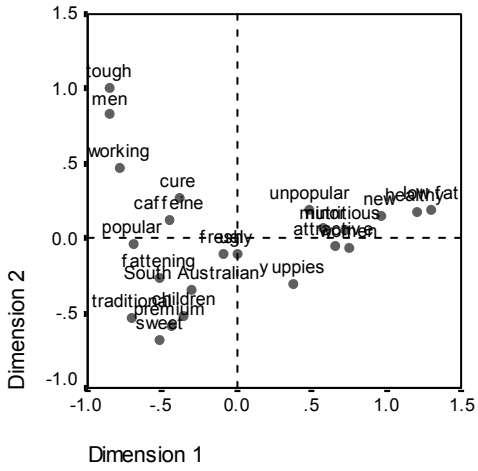
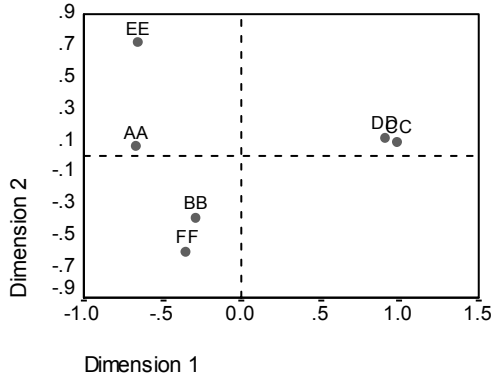


Figure 11.25 shows the plot of the brands. Notice that all brands are far from the origin, so no brand is similar to the overall centroid. Brands *CC* and *DD* group together at the right, whereas brands *BB* and *FF* cluster in the lower half of the plot. Brands *AA* and *EE* are not similar to any other brand.

Figure 11.25 Plot of brands (principal normalization)



Symmetrical Normalization

How are the brands related to the image attributes? Principal normalization cannot address these relationships. To focus on how the variables are related to each other, use symmetrical normalization. Rather than spread the inertia twice (as in principal normalization), symmetrical normalization divides the inertia equally over both the rows and columns. Distances between categories for a single variable cannot be interpreted, but distances between the categories for different variables are meaningful.

To produce the following solution with symmetrical normalization, recall the Correspondence Analysis dialog box:

Model...
  Normalization Method
  ⊙ Symmetrical

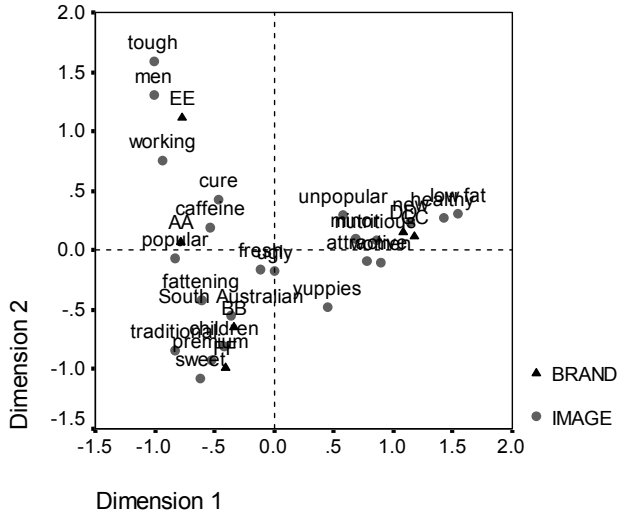
Statistics...
  ☐ Correspondence table (deselect)
  ☐ Overview of row points (deselect)
  ☐ Overview of column points (deselect)

Plots...
  Scatterplots
  ☐ Row points (deselect)
  ☐ Column points (deselect)

Figure 11.26 displays the biplot of the row and column scores. In the upper left, brand *EE* is the only tough, working brand and appeals to men. Brand *AA* is the most popular and also viewed as the most highly caffeinated. The *sweet*, *fattening* brands include *BB* and *FF*. Brands *CC* and *DD*, while perceived as *new* and *healthy*, are also the most *unpopular*.

For further interpretation, you can draw a line through the origin and the two image attributes *men* and *yuppies*, and project the brands onto this line. The two attributes are opposed to each other, indicating that the association pattern of brands for *men* is reversed compared to the pattern for *yuppies*. That is, *men* are most frequently associated with brand *EE* and least frequently with brand *CC*, whereas *yuppies* are most frequently associated with brand *CC* and least frequently with brand *EE*.

Figure 11.26 Biplot of the brands and the attributes (symmetrical normalization)



Example 3: Flying Mileage between Cities

Correspondence analysis is not restricted to frequency tables. The entries can be any positive measure of correspondence. In this example, you use the flying mileages between ten American cities. The cities are shown in Table 11.2.

Table 11.2 City labels

City	Label	City	Label
Atlanta	Atl	Miami	Mia
Chicago	Chi	New York	NY
Denver	Den	San Francisco	SF
Houston	Hou	Seattle	Sea
Los Angeles	LA	Washington, DC	DC

To view the flying mileages, first weight the cases by the variable *dist*. From the menus choose:

Data
Weight cases...

⊙ Weight cases by
▸ Frequency Variable: *dist*

Now, to view the mileages as a crosstabulation, from the menus choose:

Analyze
  Descriptive Statistics
    Crosstabs...

- Row(s): *row*
- Column(s): *col*

Figure 11.27 contains the flying mileages between the cities. Notice that there is only one variable for both rows and columns and that the table is symmetric; the distance from Los Angeles to Miami is the same as the distance from Miami to Los Angeles. Moreover, the distance between any city and itself is 0. The active margin reflects the total flying mileage from each city to all other cities.

Figure 11.27 Flying mileages between 10 American cities

Count

		COL										
		Atl	Chi	Den	Hou	LA	Mia	NY	SF	Sea	DC	Total
ROW	Atl		587	1212	701	1936	604	748	2139	2182	543	10652
	Chi	587		920	940	1745	1188	713	1858	1737	597	10285
	Den	1212	920		879	831	1726	1631	949	1021	1494	10663
	Hou	701	940	879		1374	968	1420	1645	1891	1220	11038
	LA	1936	1745	831	1374		2339	2451	347	959	2300	14282
	Mia	604	1188	1726	968	2339		1092	2594	2734	923	14168
	NY	748	713	1631	1420	2451	1092		2571	2408	205	13239
	SF	2139	1858	949	1645	347	2594	2571		678	2442	15223
	Sea	2182	1737	1021	1891	959	2734	2408	678		2329	15939
	DC	543	597	1494	1220	2300	923	205	2442	2329		12053
Total		10652	10285	10663	11038	14282	14168	13239	15223	15939	12053	127542

In general, distances are dissimilarities; large values indicate a large difference between the categories. However, correspondence analysis requires an association measure; thus, you need to convert dissimilarities into similarities. In other words, a large table entry must correspond to a small difference between the categories. Subtracting every table entry from the largest table entry converts the dissimilarities into similarities.

To create the similarities and store them in a new variable *sim*, from the menus choose:

Transform
  Compute...

▸ Target Variable: *sim*
▸ Numeric Expression: 2734 - dist

Now re-weight the cases by the similarity measure by recalling the Weight Cases dialog box:

▸ Frequency Variable: *sim*

Finally, to obtain a correspondence analysis for the similarities, from the menus choose:

Analyze
  Data Reduction
    Correspondence Analysis...

▸ Row: *row*
▸ Column: *col*

Select *row*. Click *Define Range*.
  Category range for row variable: row
  Minimum value: 1
  Maximum value: 10
  Click *Update*.

Select *row*. Click *Define Range*.
  Category range for row variable: row
  Minimum value: 1
  Maximum value: 10
  Click *Update*.

Model...
  Normalization Method
  ⊙ Principal

Plots...
  Scatterplots
  ☑ Row points

The new distance of 0 between Seattle and Miami indicates that they are most distant (least similar), whereas the distance of 2529 between New York and Washington, D.C., indicates that they are the least distant (most similar) pair of cities.

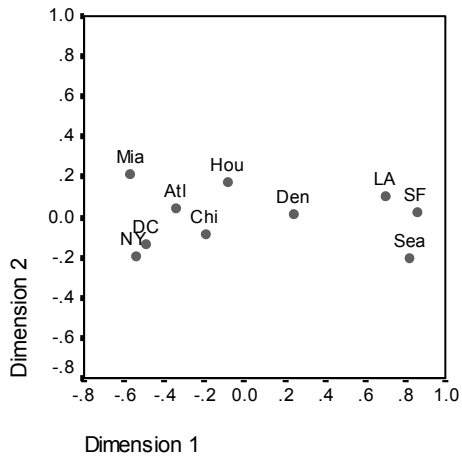
Figure 11.28 Correspondence table for similarities

	COL										
ROW	Atl	Chi	Den	Hou	LA	Mia	NY	SF	Sea	DC	Active Margin
Atl	2734	2147	1522	2033	798	2130	1986	595	552	2191	16688
Chi	2147	2734	1814	1794	989	1546	2021	876	997	2137	17055
Den	1522	1814	2734	1855	1903	1008	1103	1785	1713	1240	16677
Hou	2033	1794	1855	2734	1360	1766	1314	1089	843	1514	16302
LA	798	989	1903	1360	2734	395	283	2387	1775	434	13058
Mia	2130	1546	1008	1766	395	2734	1642	140	0	1811	13172
NY	1986	2021	1103	1314	283	1642	2734	163	326	2529	14101
SF	595	876	1785	1089	2387	140	163	2734	2056	292	12117
Sea	552	997	1713	843	1775	0	326	2056	2734	405	11401
DC	2191	2137	1240	1514	434	1811	2529	292	405	2734	15287
Active Margin	16688	17055	16677	16302	13058	13172	14101	12117	11401	15287	145858

Row and Column Scores

By using flying mileages instead of driving mileages, the terrain of the United States does not impact the distances. Consequently, all similarities should be representable in two dimensions. You center both the rows and columns and use principal normalization. Because of the symmetry of the correspondence table and the principal normalization, the row and column scores are equal and the total inertia is in both, so it does not matter whether you inspect the row or column scores. Figure 11.29 shows the orientation of the scores in two dimensions.

Figure 11.29 Points for 10 cities



The locations of the cities are very similar to their actual geographical locations, rotated about the origin. Cities which are further south have larger values along the second dimension, whereas cities which are further west have larger values along the first dimension.

12 Homogeneity Analysis Examples

The purpose of homogeneity analysis is to find quantifications that are optimal in the sense that the categories are separated from each other as much as possible. This implies that objects in the same category are plotted close to each other and objects in different categories are plotted as far apart as possible. The term **homogeneity** also refers to the fact that the analysis will be most successful when the variables are homogeneous; that is, when they partition the objects into clusters with the same or similar categories.

Example: Characteristics of Hardware

To explore how homogeneity analysis works, you will use data from Hartigan (1975), which can be found in *screws.sav*. This data set contains information on the characteristics of screws, bolts, nuts, and tacks. Table 12.1 shows the variables, along with their variable labels, and the value labels assigned to the categories of each variable in the Hartigan hardware data set.

Table 12.1 Hartigan hardware data set

Variable name	Variable label	Value labels
thread	Thread	Yes_Thread, No_Thread
head	Head form	Flat, Cup, Cone, Round, Cylinder
indhead	Indentation of head	None, Star, Slit
bottom	Bottom shape	sharp, flat
length	Length in half inches	1/2_in, 1_in, 1_1/2_ in, 2_in, 2_1/2_in
brass	Brass	Yes_Br, Not_Br
object	Object	tack, nail1, nail2, nail3, nail4, nail5, nail6, nail7, nail8, screw1, screw2, screw3, screw4, screw5, bolt1, bolt2, bolt3, bolt4, bolt5, bolt6, tack1, tack2, nailb, screwb

This example includes all of the variables in the homogeneity analysis with the exception of *object*, which is used only to label a plot of the object scores.

In order to label the object scores plots with variables used in obtaining the solution, you must create copies of the analysis variables. To do this:

- In the Data Editor, Ctrl-click on the column headings to select the contents of the variables *thread*, *head*, *indhead*, *brass*, and *length*.
- To copy the contents of these columns, from the menus choose:

  Edit
  Copy
- To paste the contents into new variables, select five empty columns and from the menus choose:

  Edit
  Paste
- Rename the new variables *thrd_lab*, *head_lab*, *ind_lab*, *brss_lab*, and *len_lab*.

Now, to obtain a homogeneity analysis, from the menus choose:

Analyze
  Data Reduction
    Optimal Scaling...

Optimal Scaling Level
⊙ All variables multiple nominal (default)

Number of Sets of Variables
⊙ One set (default)

Variables: *thread*, *head*, *indhead*, *bottom*, *brass*, *length*
Label Object Scores Plot(s) by: *object*, *thrd_lab*, *head_lab*, *brss_lab*, *len_lab*

Select *thread*, *bottom*, *brass*. Click *Define Range*.
  Maximum: 2

Select *head*, *length*. Click *Define Range*.
  Maximum: 5

Select *indhead*. Click *Define Range*.
  Maximum: 3

Select *object*. Click *Define Range*.
  Maximum: 24

Select *thrd_lab*, *brss_lab*. Click *Define Range*.
  Maximum: 2

Select *head_lab*, *len_lab*. Click *Define Range*.
  Maximum: 5

Options...
  Plot
  ☑ Discrimination measures

Multiple Dimensions

Homogeneity analysis can compute a solution for several dimensions. The maximum number of dimensions equals either the number of categories minus the number of variables with no missing data, or the number of observations minus one, whichever is smaller. However, you should rarely use the maximum number of dimensions. A smaller number of dimensions is easier to interpret, and, after a certain number of dimensions, the amount of additional association accounted for becomes negligible. A one-, two-, or three-dimensional solution in homogeneity analysis is very common.

The eigenvalues measure how much of the categorical information is accounted for by each dimension and are similar to the total variance accounted for. However, because

the quantifications differ for each dimension, the total variance accounted for is defined on a different set of quantified variables for each dimension. For this example, a two-dimensional solution produces eigenvalues of 0.62 and 0.37 for dimensions 1 and 2, respectively. The largest possible eigenvalue for each dimension is 1.

Figure 12.1 Eigenvalues

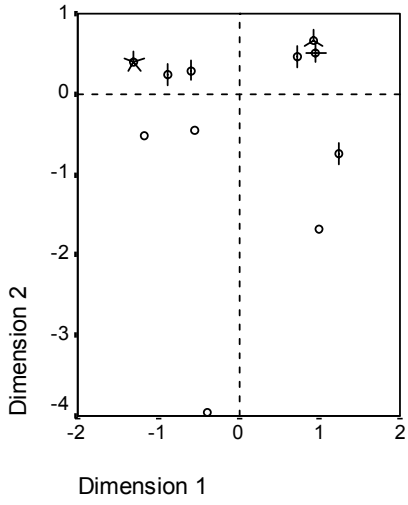
Dimension	Eigenvalue
1	.621
2	.368

The two dimensions together provide an interpretation in terms of distances. If a variable discriminates well, the objects will be close to the categories to which they belong. Ideally, objects in the same category will be close to each other (that is, they should have similar scores), and categories of different variables will be close if they belong to the same objects (that is, two objects that have similar scores for one variable should also score close to each other for the other variables in the solution).

Object Scores

After examining the frequency table and eigenvalues, you should look at the object scores. The default object scores plot, shown in Figure 12.2, is useful for spotting outliers. In Figure 12.2, there is one object at the bottom of the plot that might be considered an outlier. Later, we'll consider what happens if you drop this object.

Figure 12.2 Object scores plot



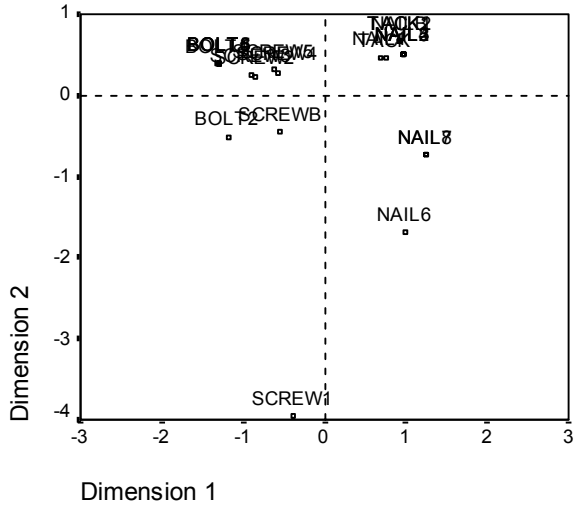
The plot shown in Figure 12.2 groups the object scores and displays them as **sunflowers**. Each **petal** on the sunflower represents a number of cases. This provides an easy way to see at a glance if many cases fall close together. Sunflowers with many petals indicate that a large number of cases fall in that area, while sunflowers with fewer petals indicate that a smaller number of cases fall in that area.

The distance from an object to the origin reflects variation from the "average" response pattern. This average response pattern corresponds to the most frequent category for each variable. Objects with many characteristics corresponding to the most frequent categories lie near the origin. In contrast, objects with unique characteristics are located far from the origin.

With a large data set, a sunflower plot is probably sufficient for most purposes. With smaller data sets such as the one in this example, however, it would be nice to see exactly where each case (object) falls on the plot. It is difficult to see specific relationships among individual objects unless you can tell which object is number 1, which object is number 2, and so on. You can specify one or more variables to label the object scores plot. Each labeling variable produces a separate plot labeled with the values of that variable.

We'll take a look at the plot of object scores labeled by the variable *object*. This is just a case-identification variable and was not used in any computations. Figure 12.3 shows the plot of object scores labeled with *object*.

Figure 12.3 Object scores labeled by variable object



Examining the plot, you see that the first dimension (the horizontal axis) discriminates the screws and bolts (which have threads) from the nails and tacks (which don't have threads). This is easily seen on the plot since screws and bolts are on one end of the horizontal axis and tacks and nails are on the other. To a lesser extent, the first dimension also separates the bolts (which have flat bottoms) from all the others (which have sharp bottoms).

The second dimension (the vertical axis) seems to separate *SCREW1* and *NAIL6* from all other objects. What *SCREW1* and *NAIL6* have in common are their values on variable *length*—they are the longest objects in the data. Moreover, *SCREW1* lies much farther from the origin than the other objects, suggesting that, taken as a whole, many of the characteristics of this object are not shared by the other objects.

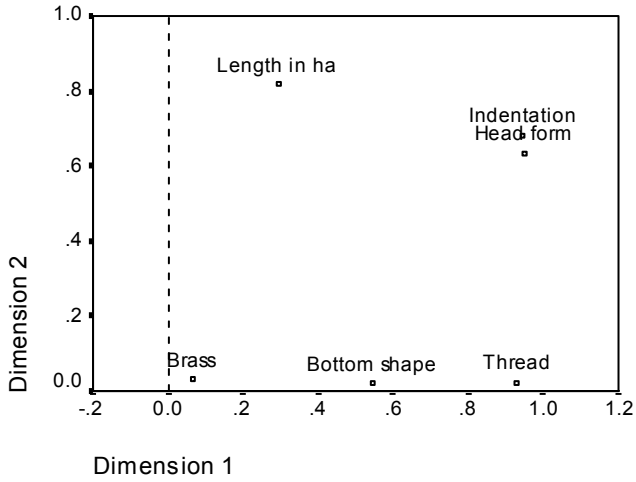
Discrimination Measures

Before examining the rest of the object scores plots, let's see if the discrimination measures agree with what we've said so far. For each variable, a discrimination measure, which can be regarded as a squared component loading, is computed for each dimension. This measure is also the variance of the quantified variable in that dimension. It has a maximum value of 1, which is achieved if the object scores fall into mutually exclusive groups and all object scores within a category are identical. (*Note*: This measure may have a value greater than 1 if there are missing data.) Large discrimination measures correspond to a large spread among the categories of the variable and, consequently, indicate a high degree of discrimination between the categories of a variable along that dimension.

The average of the discrimination measures for any dimension equals the eigenvalue (the total variance accounted for) for that dimension. Consequently, the dimensions are ordered according to average discrimination. The first dimension has the largest average discrimination, the second dimension has the second largest average discrimination, and so on for all dimensions in the solution.

As noted on the object scores plot, Figure 12.4 shows that the first dimension is related to variables *thread* and *bottom* (labeled *Bottom shape*). These variables have large discrimination measures on the first dimension and small discrimination measures on the second dimension. Thus, for both of these variables, the categories are spread far apart along the first dimension only. Variable *length* (labeled *Length in ha*) has a large value on the second dimension, but a small value on the first dimension. As a result, *length* is closest to the second dimension, agreeing with the observation from the object scores plot that the second dimension seems to separate the longest objects from the rest. *indhead* (labeled *Indentation*) and *head* (labeled *Head form*) have relatively large values on both dimensions, indicating discrimination in both the first and second dimensions.

Figure 12.4 Plot of discrimination measures



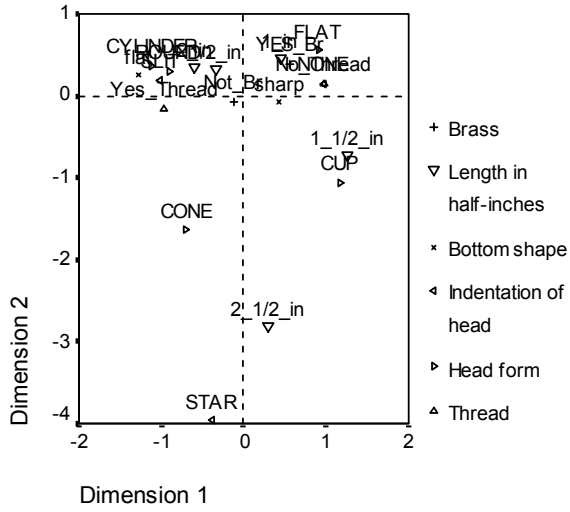
Variable *brass*, located very close to the origin, does not discriminate at all in the first two dimensions. This makes sense since all of the objects can be made of brass or not made of brass. Moreover, variable *length* only discriminates in the second dimension for the same reason.

Category Quantifications

Recall that a discrimination measure is the variance of the quantified variable along a particular dimension. The discrimination measures plot contains these variances, indicating which variables discriminate along which dimension. However, the same variance could correspond to all of the categories being spread moderately far apart or to most of the categories being close together, with a few categories differing from this group. The discrimination plot cannot differentiate between these two conditions.

Category quantification plots provide an alternative method of displaying discrimination of variables that can identify category relationships. In this plot, the coordinates of each category on each dimension are displayed. Thus, you can determine which categories are similar for each variable. The category quantifications are plotted in Figure 12.5.

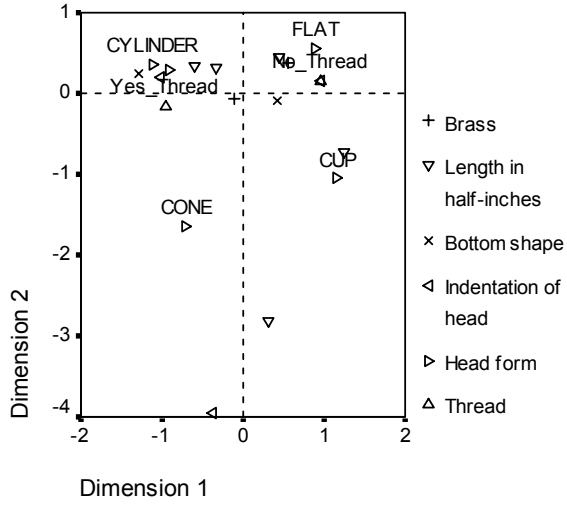
Figure 12.5 Category quantifications



Variable *length* has five categories, three of which group together near the top of the plot. The remaining two categories are in the lower half of the plot, with the *2_1/2_in* category very far from the group. The large discrimination for *length* along dimension 2 is a result of this one category being very different from the other categories of *length*. Similarly, for *head*, the category *STAR* is very far from the other categories and yields a large discrimination measure along the second dimension. These patterns cannot be illustrated in a plot of discrimination measures.

The spread of the category quantifications for a variable reflects the variance and thus indicates how well that variable is discriminated in each dimension. This is illustrated in Figure 12.6, in which some of the category labels have been omitted. Focusing on dimension 1, the categories for *thread* are far apart. However, along dimension 2, the categories for this variable are very close. Thus, *thread* discriminates better in dimension 1 than in dimension 2. In contrast, the categories for *head* are spread far apart along both dimensions, suggesting that this variable discriminates well in both dimensions.

Figure 12.6 Selected category quantifications



In addition to determining the dimensions along which a variable discriminates and how that variable discriminates, the category quantification plot also compares variable discrimination. A variable with categories that are far apart discriminates better than a variable with categories that are close together. In Figure 12.5, for example, along dimension 1, the two categories of *brass* are much closer to each other than the two categories of *thread*, indicating that *thread* discriminates better than *brass* along this dimension. However, along dimension 2, the distances are very similar, suggesting that these variables discriminate to the same degree along this dimension. The discrimination measures plot discussed previously identifies these same relationships by using variances to reflect the spread of the categories.

A More Detailed Look at Object Scores

A greater insight into the data can be gained by examining the object scores plots labeled by each variable. Ideally, similar objects should form exclusive groups, and these groups should be far from each other. Figure 12.7, Figure 12.8, Figure 12.9, and Figure 12.10 present object scores labeled by *thrd_lab*, *head_lab*, *len_lab*, and *brss_lab*. Figure 12.7 shows that the first dimension separates *Yes_Thread* and *No_Thread* perfectly. All of the objects with threads have negative object scores, whereas all of the nonthreaded objects have positive scores. Although the two categories do not form compact groups, the perfect differentiation between the categories is generally considered a good result.

Figure 12.7 Object scores labeled with variable thread

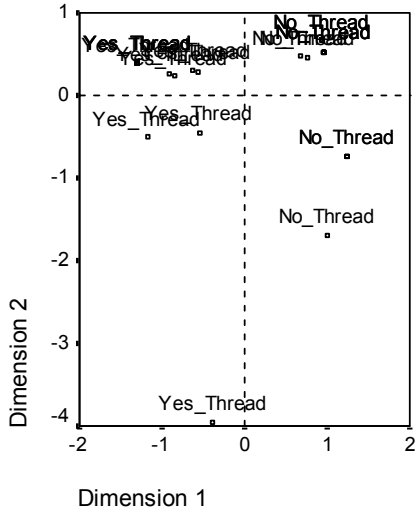


Figure 12.8 Object scores labeled with variable head

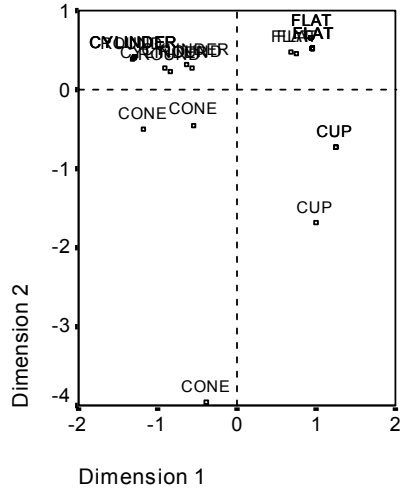


Figure 12.9 Object scores labeled with variable length

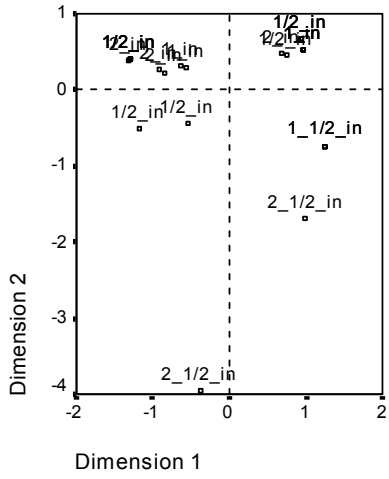


Figure 12.10 Object scores labeled with variable brass

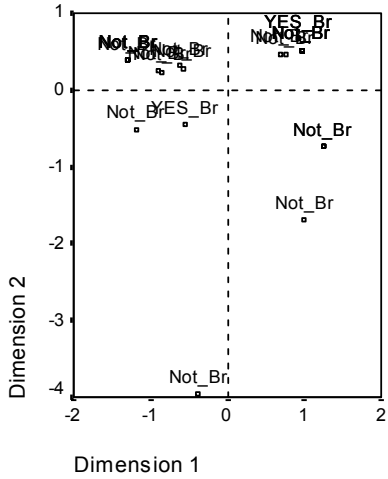


Figure 12.8 shows that *head* discriminates in both dimensions. The *FLAT* objects group together in the upper right corner of the plot, whereas the *CUP* objects group together in the lower right. *CONE* objects all lie in the lower left. However, these objects are more spread out than the other groups and, thus, are not as homogeneous. Finally, *CYLINDER* objects cannot be separated from *ROUND* objects, both of which lie in the upper left corner of the plot.

Figure 12.9 shows that *length* does not discriminate in the first dimension. The categories of *length* display no grouping when projected onto a horizontal line. However, *length* does discriminate in the second dimension. The shorter objects correspond to positive scores, and the longer objects correspond to large negative scores.

Figure 12.10 shows that *brass* has categories that can't be separated very well in the first or second dimensions. The object scores are widely spread throughout the space. The brass objects cannot be differentiated from the nonbrass objects.

Omission of Outliers

In homogeneity analysis, outliers are objects that have too many unique features. As noted in Figure 12.2, *SCREW1* might be considered an outlier.

▶ To delete this object and run the analysis again, from the menus choose:

Data
  Select Cases...

⊙ If condition is satisfied
Click *If*.
  object ~= 16

▶ Then, recall the Homogeneity Analysis dialog box.

▶ Label Object Scores Plot(s) by: *brss_lab*, *ind_lab*

Select *ind_lab*. Click *Define Range*.
  Maximum: 3

Options...
  Display
  ☐ Frequencies (deselect)
  ☐ Discrimination measures (deselect)
  ☐ Category quantifications (deselect)

  Plot
  ☐ Category quantifications (deselect)

The solution changes considerably. The eigenvalues for a two-dimensional solution are 0.64 and 0.35.

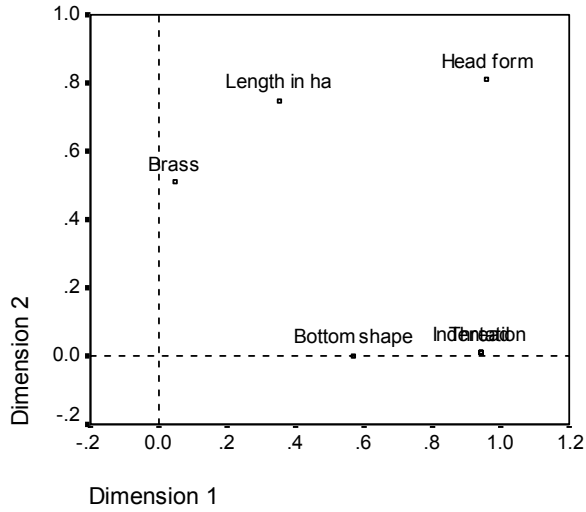
Figure 12.11 Eigenvalues

Eigenvalues

Dimension	Eigenvalue
1	.636
2	.347

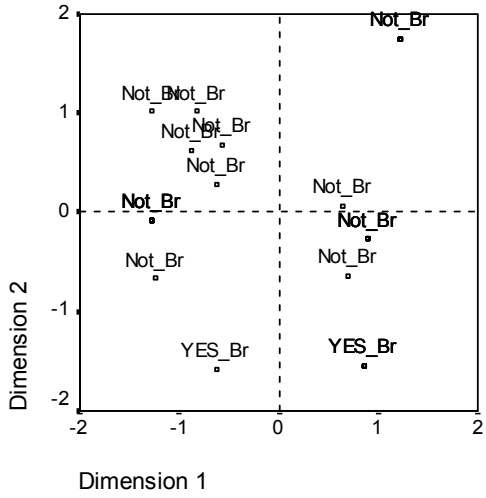
As shown in the discrimination plot in Figure 12.12, *indhead* no longer discriminates in the second dimension, whereas *brass* changes from no discrimination in either dimension to discrimination in the second dimension. Discrimination for the other variables is largely unchanged.

Figure 12.12 Discrimination measures



The object scores plot labeled by *brass* is shown in Figure 12.13. The four brass objects all appear near the bottom of the plot (three objects occupy identical locations), indicating high discrimination along the second dimension. As was the case for *thread* in the previous analysis, the objects do not form compact groups, but the differentiation of objects by categories is perfect.

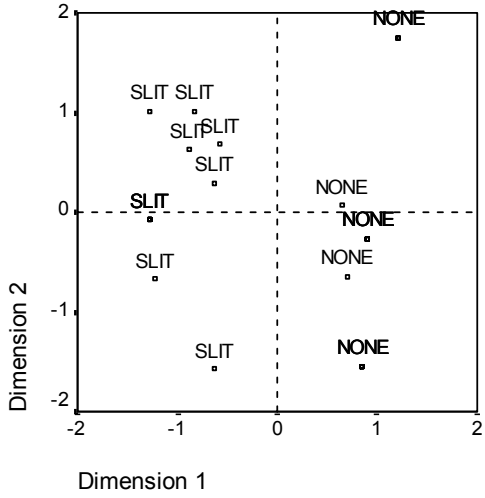
Figure 12.13 Object scores labeled with variable brass



The object scores plot labeled by *indhead* is shown in Figure 12.14. The first dimension discriminates perfectly between the non-indented objects and the indented objects, as in the previous analysis. In contrast to the previous analysis, however, the second dimension cannot now distinguish the two *indhead* categories.

Thus, omission of *SCREW1*, which is the only object with a star-shaped head, dramatically affects the interpretation of the second dimension. This dimension now differentiates objects based on *brass, head,* and *length*.

Figure 12.14 Object scores labeled with variable indhead



13 Multidimensional Scaling Examples

Given a set of objects, the goal of multidimensional scaling is to find a representation of the objects in a low-dimensional space. This solution is found using the **proximities** between the objects. The procedure minimizes the squared deviations between the original, possibly transformed, object proximities and their Euclidean distances in the low-dimensional space.

The purpose of the low-dimensional space is to uncover relationships between the objects. By restricting the solution to be a linear combination of independent variables, you may be able to interpret the dimensions of the solution in terms of these variables. In the following example, you will see how 15 different kinship terms can be represented in 3 dimensions, and how that space can be interpreted with respect to the gender, generation, and degree of separation of each of the terms.

Example: An Examination of Kinship Terms

Rosenberg and Kim (1975) set out to analyze 15 kinship terms (aunt, brother, cousin, daughter, father, granddaughter, grandfather, grandmother, grandson, mother, nephew, niece, sister, son, uncle). They asked four groups of college students (two female, two male) to sort these terms on the basis of similarities. Two groups (one female, one male) were asked to sort twice, with the second sorting based on a different criteria from the first sort. Thus, a total of six "sources" were obtained, as outlined in Table 13.1.

Table 13.1 Source structure of the kinship data

Source	Gender	Condition	Sample size
1	Female	Single sort	85
2	Male	Single sort	85
3	Female	First sort	80
4	Female	Second sort	80
5	Male	First sort	80
6	Male	Second sort	80

Each source corresponds to a 15×15 proximity matrix, whose cells are equal to the number of people in a source minus the number of times the objects were partitioned together in that source. This data set can be found in *kinship_dat.sav*.

Choosing the Number of Dimensions

It is up to you to decide how many dimensions the solution should have. A good tool to help you make this decision is the **scree plot**. To create a scree plot, from the menus choose:

Analyze
  Scale
    Multidimensional Scaling (PROXSCAL)...

Data Format
⊙ The data are proximities

Number of Sources
⊙ Multiple matrix sources

Multiple Sources
⊙ The proximities are in stacked matrices across columns

Proximities: *aunt*, *brother cousin*, *daughter*, *father*, *gdaugh*, *gfather*, *gmother*, *gson*, *mother*, *nephew*, *niece*, *sister*, *son*, *uncle*
Sources: *sourceid*

Model...
  Dimensions
  Maximum: 10

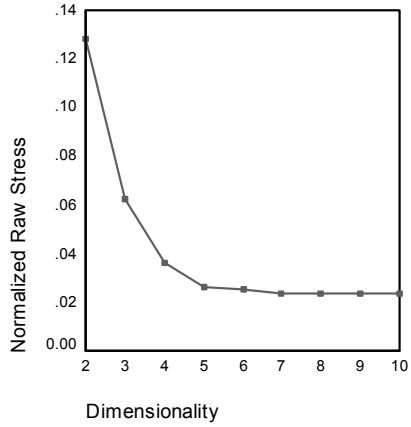
Restrictions...
  Restrictions on Common Space
  ⊙ Linear combination of independent variables

  Restriction Variables
  Read variables from: *kinship_var.sav*
  Selected: *gender*, *gener*, *degree*

Plots...
  ☑ Stress
  ☐ Common space (deselect)
  ☐ Individual space weights (deselect)
  ☐ Variable and dimension correlations (deselect)

Output...
  ☐ Common space coordinates (deselect)
  ☐ Individual space weights (deselect)
  ☐ Multiple stress measures (deselect)
  ☐ Transformed independent variables (deselect)

Figure 13.1 Scree Plot



The procedure begins with a 10-dimensional solution, and works down to a 2-dimensional solution. The scree plot shows the normalized raw Stress of the solution at each dimension. You can see from the plot that increasing the dimensionality from 2 to 3 and from 3 to 4 offers large improvements in the Stress. After 4, the improvements are rather small. You will choose to analyze the data using a 3-dimensional solution, since the results are easier to interpret.

A Three-Dimensional Solution

Independent variables *gender*, *gener*(ation), and *degree* (of separation) were constructed with the intention of using them to interpret the dimensions of the solution. The independent variables were constructed as follows:

gender 1=male, 2=female, 9=missing (for cousin)

gener The number of generations from you if the term refers to your kin, with lower numbers corresponding to older generations. Thus, grandparents are –2, grandchildren are 2, and siblings are 0.

degree The number of degrees of separation along your family tree. Thus, your parents are up one node, while your children are down one node. Your siblings are up one node to your parents, then down one node to them, for 2 degrees of separation. Your cousin is 4 degrees away; 2 up to your grandparents, then 2 down through your aunt/uncle to them.

The external variables can be found in *kinship_var.sav*.

Figure 13.2 Independent variables

	Variable		
	gender	generation	degree
Aunt	2.000	-1.000	3.000
Brother	1.000	.000	2.000
Cousin	.	.000	4.000
Daughter	2.000	1.000	1.000
Father	1.000	-1.000	1.000
Granddaughter	2.000	2.000	2.000
Grandfather	1.000	-2.000	2.000
Grandmother	2.000	-2.000	2.000
Grandson	1.000	2.000	2.000
Mother	2.000	-1.000	1.000
Nephew	1.000	1.000	3.000
Niece	2.000	1.000	3.000
Sister	2.000	.000	2.000
Son	1.000	1.000	1.000
Uncle	1.000	-1.000	3.000

Additionally, an initial configuration from an earlier analysis is supplied in *kinship_ini.sav*. To obtain a three-dimensional solution, recall the Proximities in Matrices Across Columns dialog box:

Model...

Dimensions
Minimum: 3
Maximum: 3

Options...

Initial Configuration
⊙ Custom

Custom Configuration
Read variables from: *kinship_ini.sav*
Selected: *dim01*, *dim02*, *dim03*

Plots...

- ☐ Stress (deselect)
- ☑ Common space
- ☑ Original vs. transformed proximities
- ☑ Transformed independent variables

Output...

- ☑ Input data
- ☑ Multiple stress measures
- ☑ Stress decomposition
- ☑ Variable and dimension correlations

Stress Measures

The Stress and fit measures give an indication of how well the distances in the solution approximate the original distances.

Figure 13.3 Stress and fit measures

Normalized Raw Stress	.06234
Stress-I	.24968
Stress-II	.87849
S-Stress	.14716
Dispersion Accounted For (D.A.F.)	.93766
Tucker's Coefficient of Congruence	.96833

Each of the four Stress statistics measures the misfit of the data, while the dispersion accounted for and Tucker's coefficient of congruence measure the fit. Lower Stress measures (to a minimum of 0) and higher fit measures (to a maximum of 1) indicate better solutions.

Figure 13.4 Decomposition of normalized raw Stress

		Source						
		SRC_1	SRC_2	SRC_3	SRC_4	SRC_5	SRC_6	Mean
Object	Aunt	.0991	.0754	.0629	.0468	.0391	.0489	.0620
	Brother	.1351	.0974	.0496	.0813	.0613	.0597	.0807
	Cousin	.0325	.0336	.0480	.0290	.0327	.0463	.0370
	Daughter	.0700	.0370	.0516	.0229	.0326	.0207	.0391
	Father	.0751	.0482	.0521	.0225	.0272	.0298	.0425
	Granddaughter	.1410	.0736	.0801	.0707	.0790	.0366	.0802
	Grandfather	.1549	.1057	.0858	.0821	.0851	.0576	.0952
	Grandmother	.1550	.0979	.0858	.0844	.0816	.0627	.0946
	Grandson	.1374	.0772	.0793	.0719	.0791	.0382	.0805
	Mother	.0813	.0482	.0526	.0229	.0260	.0227	.0423
	Nephew	.0843	.0619	.0580	.0375	.0317	.0273	.0501
	Niece	.0850	.0577	.0503	.0353	.0337	.0260	.0480
	Sister	.1361	.0946	.0496	.0816	.0629	.0588	.0806
	Son	.0689	.0373	.0456	.0242	.0337	.0253	.0392
	Uncle	.0977	.0761	.0678	.0489	.0383	.0498	.0631
Mean		.1035	.0681	.0613	.0508	.0496	.0407	.0623

The decomposition of Stress helps you to identify which sources and objects contribute the most to the overall Stress of the solution. In this case, most of the Stress among the sources is attributable to sources 1 and 2, while among the objects, most of the Stress is attributable to *Brother*, *Granddaughter*, *Grandfather*, *Grandmother*, *Grandson*, and *Sister*.

By referring to Table 13.1, you can see that the two sources accountable for most of the Stress are the two groups that sorted the terms only once. This suggests that the students considered multiple factors when sorting the terms, and those who were allowed to sort twice focused on a portion of those factors for the first sort, and then considered the remaining factors during the second sort.

The objects that account for most of the Stress are those with a *degree* of 2. These are relations who are not part of the "nuclear" family (mother, father, daughter, son), but are nonetheless closer than other relations. This middle position could easily cause some differential sorting of these terms.

Final Coordinates of the Common Space

The common space plot gives a visual representation of the relationships between the objects. The following two figures are taken from the scatterplot matrix, for closer viewing.

Figure 13.5 Common space coordinates (dimensions 1 and 3)

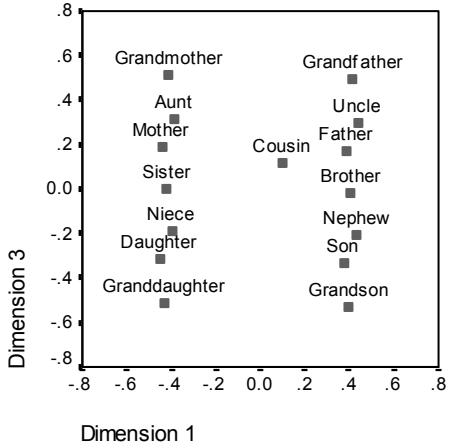


Figure 13.5 shows the final coordinates for the objects in dimensions 1 and 3; this is the plot in the lower-left corner of the scatterplot matrix. This plot shows that dimension 1 (on the *x* axis) is correlated with the variable *gender* and dimension 3 (on the *y* axis) is correlated with *gener*. From left to right, you see that dimension 1 separates the female and male terms, with the genderless term *Cousin* in the middle. From the bottom of the plot to the top, increasing values along the axis correspond to terms that are older.

Figure 13.6 Common space coordinates (dimensions 2 and 3)

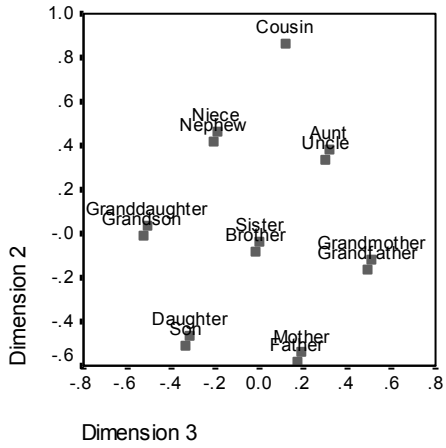


Figure 13.6 shows the final coordinates for the objects in dimensions 2 and 3; this is the plot in the middle-right side of the scatterplot matrix. From this plot, you can see that the second dimension (along the *y* axis) corresponds to the variable *degree*, with larger values along the axis corresponding to terms that are further from the "nuclear" family.

Correlations

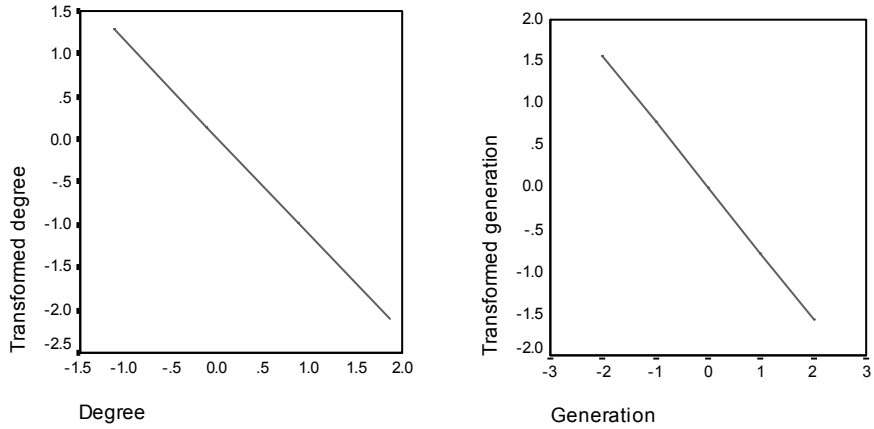
The correlations between the transformed independent variables and the dimensions of the common space summarize the findings of the common space plot.

Figure 13.7 Correlations

	Dimension		
Variable	1	2	3
gender	-.999	.062	.027
generation	.013	-.118	.985
degree	-.079	-.992	-.169

It may be disconcerting at first to see a negative correlation between dimension 2 and *degree*, since both larger values of *degree* and larger values of dimension 2 correspond to terms that are further from the "nuclear" family. A similar situation occurs with *gener* and dimension 3. Increasing values of *gener* correspond to younger terms, while increasing values of dimension 3 correspond to terms that are older, yet *gener* and dimension 3 are positively correlated. However, this is easily explained by the transformation plots for *degree* and *gener.*

Figure 13.8 Transformed degree and gener



Since *degree* was scaled at the default interval level, its transformation plot is linear, but the optimally scaled values of *degree* are negatively correlated with the original values. This is shown by the downward slope of the line in the transformation plot. While the original values of *degree* are positively correlated with dimension 1, the correlations in the table are computed for the transformed values, and since the original values of *degree* are negatively correlated with the transformed values, the correlation between dimension 2 and the transformed values is negative.

Likewise, since the optimally scaled values of *gener* are negatively correlated with the original values, the correlation between dimension 3 and the transformed values is positive.

A Three-Dimensional Solution with Nondefault Transformations

The previous solution was computed using the default ratio transformation for proximities and interval transformations for the independent variables. The results are pretty good, but you may be able to do better by using other transformations. For example, *gender* has no inherent ordering, so it may be better to scale at the nominal level. The proximities, *gener*, and *degree* all have natural orderings, but they may be better modeled by an ordinal transformation than a linear transformation. To rerun the analysis, scaling

gender at the nominal level and the proximities, *gener*, and *degree* at the ordinal level (keeping ties), recall the Proximities in Matrices Across Columns dialog box:

Model...
  Proximity Transformations
  ⊙ Ordinal

Restrictions...
  Restriction Variables
  Selected: *gender*, *gener*, *degree*

  Select *gender*.
  Independent variable transformation: Nominal

  Select *gener*, *degree*.
  Independent variable transformation: Ordinal (keep ties)

Plots...
  ☑ Common space
  ☑ Original vs. transformed proximities
  ☑ Transformed independent variables

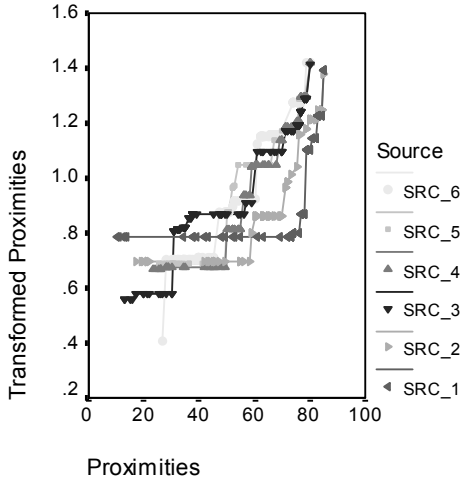
Output...
  ☐ Input data (deselect)
  ☑ Multiple stress measures
  ☐ Stress decomposition (deselect)
  ☐ Variable and dimension correlations (deselect)

Transformation Plots

The transformation plots are a good first check to see whether the original transformations were appropriate. If the plots are approximately linear, then the linear assumption is appropriate. If not, then you need to check the Stress measures to see if there is an improvement in fit, and the common space plot to see if the interpretation is more useful.

The independent variables each obtain approximately linear transformations, so it may be appropriate to interpret them as numerical. However, the proximities do not obtain a linear transformation, so it is possible that the ordinal transformation is more appropriate for the proximities.

Figure 13.9 Transformed proximities



Stress Measures

The Stress for the current solution supports the argument for scaling the proximities at the ordinal level.

Figure 13.10 Stress and fit measures

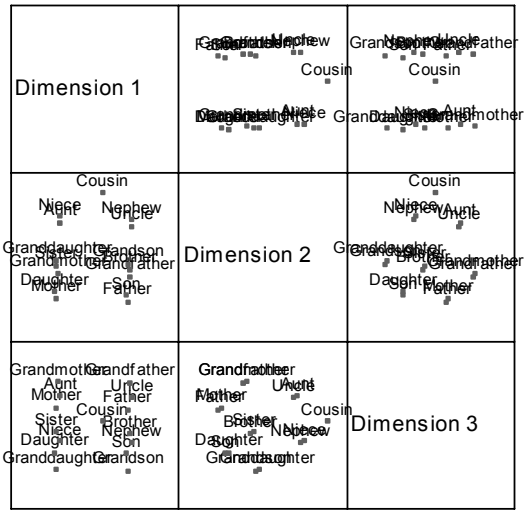
Normalized Raw Stress	.03157
Stress-I	.17769
Stress-II	.62320
S-Stress	.08231
Dispersion Accounted For (D.A.F.)	.96843
Tucker's Coefficient of Congruence	.98409

The normalized raw Stress for the previous solution, found in Figure 13.3, is 0.06234. Scaling the variables using nondefault transformations halves the Stress to 0.03157.

Final Coordinates of the Common Space

The common space plots offer essentially the same interpretation of the dimensions as the previous solution.

Figure 13.11 Common space coordinates



Discussion

It is best to treat the proximities as ordinal variables, since there is a great improvement in the Stress measures. As a next step, you may want to "untie" the ordinal variables—that is, allow equivalent values of the original variables to obtain different transformed values. For example, in the first source, the proximities between *Aunt* and *Son*, and *Aunt* and *Grandson*, are 85. The "tied" approach to ordinal variables forces the transformed values of these proximities to be equivalent, but there is no particular reason for you to assume that they should be. In this case, allowing the proximities to become untied frees you from an unnecessary restriction.

Syntax Reference

Introduction

This syntax reference guide describes the SPSS command language underlying SPSS Categories. Most of the features of these commands are implemented in the dialog boxes and can be used directly from the dialog boxes. Or you can paste the syntax into a syntax window and edit it or build a command file, which you can save and reuse. The features that are available only in command syntax are summarized following the discussion of the dialog box interface in the corresponding chapter on each statistical procedure.

A Few Useful Terms

All terms in the SPSS command language fall into one or more of the following categories:

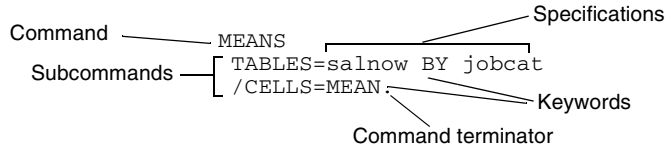
Keyword. A word already defined by SPSS to identify a command, subcommand, or specification. Most keywords are, or resemble, common English words.

Command. A specific instruction that controls the execution of SPSS.

Subcommand. Additional instructions for SPSS commands. A command can contain more than one subcommand, each with its own specifications.

Specifications. Instructions added to a command or subcommand. Specifications may include subcommands, keywords, numbers, arithmetic operators, variable names, special delimiters, and so forth.

Each command begins with a command keyword (which may contain more than one word). The command keyword is followed by at least one blank space and then any additional specifications. Each command ends with a command terminator, which is a period. For example:



Syntax Diagrams

Each SPSS command described in this manual includes a syntax diagram that shows all of the subcommands, keywords, and specifications allowed for that command. These syntax diagrams are also available in the online Help system for easy reference when entering commands in a syntax window. By remembering the following rules, you can use the syntax diagram as a quick reference for any command:

- Elements shown in all capital letters are keywords defined by SPSS to identify commands, subcommands, functions, operators, and other specifications.
- Elements in lower case describe specifications you supply.

- Elements in boldface type are defaults. A default indicated with two asterisks (**) is in effect when the keyword is not specified. (Boldface is not used in the online Help system syntax diagrams.)
- Parentheses, apostrophes, and quotation marks are required where indicated.
- Elements enclosed in square brackets ([ ]) are optional.
- Braces ({ }) indicate a choice among elements. You can specify any one of the elements enclosed within the aligned braces.
- Ellipses indicate that an element can be repeated.
- Most abbreviations are obvious; for example, varname stands for variable name and varlist stands for a list of variables.
- The command terminator is not shown in the syntax diagrams.

Syntax Rules

Keep in mind the following simple rules when writing and editing commands in a syntax window:

- Each command must begin on a new line and end with a period.
- Subcommands are separated by slashes. The slash before the first subcommand in a command is optional in most commands.
- SPSS keywords are not case-sensitive, and three-letter abbreviations can be used for most keywords.
- Variable names must be spelled out in full.
- You can use as many lines as you want to specify a single command. However, text included within apostrophes or quotation marks must be contained on a single line.
- You can add space or break lines at almost any point where a single blank is allowed, such as around slashes, parentheses, arithmetic operators, or between variable names.
- Each line of syntax cannot exceed 80 characters.
- The period must be used as the decimal indicator, regardless of your language settings.

For example,

```
FREQUENCIES
 VARIABLES=JOBCAT SEXRACE
 /PERCENTILES=25 50 75
 /BARCHART.
```

and

```
freq var=jobcat sexrace /percent=25 50 75 /bar.
```

are both acceptable alternatives that generate the same results. The second example uses three-letter abbreviations and lower case, and the command is on one line.

INCLUDE Files

If your SPSS commands are contained in a command file that is specified on the SPSS INCLUDE command, the syntax rules are slightly different:

- Each command must begin in the first column of a new line.
- Continuation lines within a command must be indented at least one space.
- The period at the end of the command is optional.

If you generate command syntax by pasting dialog box choices into a syntax window, the format of the commands is suitable for both INCLUDE files and commands run in a syntax window.

ANACOR

```
ANACOR  TABLE={row var (min, max) BY column var (min, max)}
               {ALL (# of rows, # of columns)              }

[/DIMENSION={2**  }]
            {value}

[/NORMALIZATION={CANONICAL**}]
                {PRINCIPAL  }
                {RPRINCIPAL }
                {CPRINCIPAL }
                {value      }

[/VARIANCES=[SINGULAR] [ROWS] [COLUMNS]]

[/PRINT=[TABLE**] [PROFILES] [SCORES**] [CONTRIBUTIONS**]
        [DEFAULT] [PERMUTATION] [NONE]]

[/PLOT=[NDIM=({1, 2**      })]
              {value, value}
              {ALL, MAX    }
       [ROWS**[(n)]][COLUMNS**[(n)]][DEFAULT[(n)]]
       [TRROWS] [TRCOLUMNS] [JOINT[(n)]] [NONE]]

[/MATRIX OUT=[SCORE({*   })] [VARIANCE({*   })]]
                    {file}             {file}
```

**Default if subcommand or keyword is omitted.

Overview

ANACOR performs correspondence analysis, which is an isotropic graphical representation of the relationships between the rows and columns of a two-way table.

Options

Number of dimensions. You can specify how many dimensions ANACOR should compute.

Method of normalization. You can specify one of five different methods for normalizing the row and column scores.

Computation of variances and correlations. You can request computation of variances and correlations for singular values, row scores, or column scores.

Data input. You can analyze the usual individual casewise data or aggregated data from table cells.

Display output. You can control which statistics are displayed and plotted. You can also control how many value-label characters are used on the plots.

Writing matrices. You can write matrix data files containing row and column scores and variances for use in further analyses.

Basic Specification

- The basic specification is ANACOR and the TABLE subcommand. By default, ANACOR computes a two-dimensional solution, displays the TABLE, SCORES, and CONTRIBUTIONS statistics, and plots the row scores and column scores of the first two dimensions.

Subcommand Order

- Subcommands can appear in any order.

Operations

- If a subcommand is specified more than once, only the last occurrence is executed.

Limitations

- The data within table cells cannot contain negative values. ANACOR will treat such values as 0.

Example

```
ANACOR TABLE=MENTAL(1,4) BY SES(1,6)
 /PRINT=SCORES CONTRIBUTIONS
 /PLOT=ROWS COLUMNS.
```

- Two variables, *MENTAL* and *SES*, are specified on the TABLE subcommand. *MENTAL* has values ranging from 1 to 4 and *SES* has values ranging from 1 to 6.
- The row and column scores and the contribution of each row and column to the inertia of each dimension are displayed.
- Two plots are produced. The first one plots the first two dimensions of row scores and the second one plots the first two dimensions of column scores.

TABLE Subcommand

TABLE specifies the row and column variables along with their value ranges for individual casewise data. For table data, TABLE specifies the keyword ALL and the number of rows and columns.

- The TABLE subcommand is required.

Casewise Data

- Each variable is followed by a value range in parentheses. The value range consists of the variable's minimum value, a comma, and its maximum value.
- Values outside of the specified range are not included in the analysis.
- Values do not have to be sequential. Empty categories receive scores of 0 and do not affect the rest of the computations.

Example

```
DATA LIST FREE/VAR1 VAR2.
BEGIN DATA
3 1
6 1
3 1
4 2
4 2
6 3
6 3
6 3
3 2
4 2
6 3
END DATA.
ANACOR TABLE=VAR1(3,6) BY VAR2(1,3).
```

- DATA LIST defines two variables, *VAR1* and *VAR2*.
- *VAR1* has three levels, coded 3, 4, and 6, while *VAR2* also has three levels, coded 1, 2, and 3.
- Since a range of (3,6) is specified for *VAR1*, ANACOR defines four categories, coded 3, 4, 5, and 6. The empty category, 5, for which there is no data, receives zeros for all statistics but does not affect the analysis.

Table Data

- The cells of a table can be read and analyzed directly by using the keyword ALL after TABLE.
- The columns of the input table must be specified as variables on the DATA LIST command. Only columns are defined, not rows.
- ALL is followed by the number of rows in the table, a comma, and the number of columns in the table, in parentheses.
- The number of rows and columns specified can be smaller than the actual number of rows and columns if you want to analyze only a subset of the table.
- The variables (columns of the table) are treated as the column categories, and the cases (rows of the table) are treated as the row categories.
- Rows cannot be labeled when you specify TABLE=ALL. If labels in your output are important, use the WEIGHT command method to enter your data (see "Analyzing Aggregated Data" on p. 222).

Example

```
DATA LIST /COL01 TO COL07 1-21.
BEGIN DATA
 50 19 26  8 18  6  2
 16 40 34 18 31  8  3
 12 35 65 66123 23 21
 11 20 58110223 64 32
 14 36114185714258189
  0  6 19 40179143 71
END DATA.
ANACOR TABLE=ALL(6,7).
```

- DATA LIST defines the seven columns of the table as the variables.
- The TABLE=ALL specification indicates that the data are the cells of a table. The (6,7) specification indicates that there are six rows and seven columns.

DIMENSION Subcommand

DIMENSION specifies the number of dimensions you want ANACOR to compute.

- If you do not specify the DIMENSION subcommand, ANACOR computes two dimensions.
- DIMENSION is followed by an integer indicating the number of dimensions.
- In general, you should choose as few dimensions as needed to explain most of the variation. The minimum number of dimensions that can be specified is 1. The maximum number of dimensions that can be specified is equal to the number of levels of the variable with the least number of levels, minus 1. For example, in a table where one variable has five levels and the other has four levels, the maximum number of dimensions that can be specified is (4 – 1), or 3. Empty categories (categories with no data, all zeros, or all missing data) are not counted toward the number of levels of a variable.
- If more than the maximum allowed number of dimensions is specified, ANACOR reduces the number of dimensions to the maximum.

NORMALIZATION Subcommand

The NORMALIZATION subcommand specifies one of five methods for normalizing the row and column scores. Only the scores and variances are affected; contributions and profiles are not changed.

The following keywords are available:

CANONICAL *For each dimension, rows are the weighted average of columns divided by the matching singular value, and columns are the weighted average of rows divided by the matching singular value*. This is the default if the NORMALIZATION subcommand is not specified. DEFAULT is an alias for CANONICAL. Use this normalization method if you are primarily interested in differences or similarities between variables.

PRINCIPAL *Distances between row points and column points are approximations of chi-square distances*. The distances represent the distance between the row or

column and its corresponding average row or column profile. Use this normalization method if you want to examine both differences between categories of the row variable and differences between categories of the column variable (but not differences between variables).

RPRINCIPAL *Distances between row points are approximations of chi-square distances*. This method maximizes distances between row points. This is useful when you are primarily interested in differences or similarities between categories of the row variable.

CPRINCIPAL *Distances between column points are approximations of chi-square distances*. This method maximizes distances between column points. This is useful when you are primarily interested in differences or similarities between categories of the column variable.

The fifth method has no keyword. Instead, any value in the range –2 to +2 is specified after NORMALIZATION. A value of 1 is equal to the RPRINCIPAL method, a value of 0 is equal to CANONICAL, and a value of –1 is equal to the CPRINCIPAL method. The inertia is spread over both row and column scores. This method is useful for interpreting joint plots.

VARIANCES Subcommand

Use VARIANCES to display variances and correlations for the singular values, the row scores, and/or the column scores. If VARIANCES is not specified, variances and correlations are not included in the output.

The following keywords are available:

SINGULAR *Variances and correlations of the singular values*.

ROWS *Variances and correlations of the row scores*.

COLUMNS *Variances and correlations of the column scores*.

PRINT Subcommand

Use PRINT to control which of several correspondence statistics are displayed. If PRINT is not specified, the numbers of rows and columns, all nontrivial singular values, proportions of inertia, and the cumulative proportion of inertia accounted for are displayed.

The following keywords are available:

TABLE *A crosstabulation of the input variables showing row and column marginals*.

PROFILES *The row and column profiles*. PRINT=PROFILES is analogous to the CELLS=ROW COLUMN subcommand in CROSSSTABS.

SCORES *The marginal proportions and scores of each row and column*.

CONTRIBUTIONS *The contribution of each row and column to the inertia of each dimension, and the proportion of distance to the origin accounted for in each dimension.*

PERMUTATION *The original table permuted according to the scores of the rows and columns for each dimension.*

NONE *No output other than the singular values.*

DEFAULT *TABLE, SCORES, and CONTRIBUTIONS.* These statistics are displayed if you omit the PRINT subcommand.

PLOT Subcommand

Use PLOT to produce plots of the row scores, column scores, row and column scores, transformations of the row scores, and transformations of the column scores. If PLOT is not specified, a plot of the row scores in the first two dimensions and a plot of the column scores in the first two dimensions are produced.

The following keywords are available:

TRROWS *Plot of transformations of the row category values into row scores.*

TRCOLUMNS *Plot of transformations of the column category values into column scores.*

ROWS *Plot of row scores.*

COLUMNS *Plot of column scores.*

JOINT *A combined plot of the row and column scores.* This plot is not available when NORMALIZATION=PRINCIPAL.

NONE *No plots.*

DEFAULT *ROWS and COLUMNS.*

- The keywords ROWS, COLUMNS, JOINT, and DEFAULT can be followed by an integer value in parentheses to indicate how many characters of the value label are to be used on the plot. The value can range from 1 to 20; the default is 3. Spaces between words count as characters.
- TRROWS and TRCOLUMNS plots use the full value labels up to 20 characters.
- If a label is missing for any value, the actual values are used for all values of that variable.
- Value labels should be unique.
- The first letter of a label on a plot marks the place of the actual coordinate. Be careful that multiple-word labels are not interpreted as multiple points on a plot.

In addition to the plot keywords, the following can be specified:

NDIM *Dimension pairs to be plotted.* NDIM is followed by a pair of values in parentheses. If NDIM is not specified, plots are produced for dimension 1 by dimension 2.

- The first value indicates the dimension that is plotted against all higher dimensions. This value can be any integer from 1 to the number of dimensions minus 1.
- The second value indicates the highest dimension to be used in plotting the dimension pairs. This value can be any integer from 2 to the number of dimensions.
- Keyword ALL can be used instead of the first value to indicate that all dimensions are paired with higher dimensions.
- Keyword MAX can be used instead of the second value to indicate that plots should be produced up to, and including, the highest dimension fit by the procedure.

Example

```
ANACOR TABLE=MENTAL(1,4) BY SES(1,6)
 /PLOT NDIM(1,3) JOINT(5).
```

- The NDIM (1,3) specification indicates that plots should be produced for two dimension pairs—dimension 1 versus dimension 2 and dimension 1 versus dimension 3.
- JOINT requests combined plots of row and column scores. The (5) specification indicates that the first five characters of the value labels are to be used on the plots.

Example

```
ANACOR TABLE=MENTAL(1,4) BY SES(1,6)
 /PLOT NDIM(ALL,3) JOINT(5).
```

- This plot is the same as above except for the ALL specification following NDIM. This indicates that all possible pairs up to the second value should be plotted, so JOINT plots will be produced for dimension 1 versus dimension 2, dimension 2 versus dimension 3, and dimension 1 versus dimension 3.

MATRIX Subcommand

Use MATRIX to write row and column scores and variances to matrix data files.

MATRIX is followed by keyword OUT, an equals sign, and one or both of the following keywords:

SCORE (file) *Write row and column scores to a matrix data file.*

VARIANCE (file) *Write variances to a matrix data file.*

- You can specify the file with either an asterisk (*) to replace the working data file with the matrix file or the name of an external file.
- If you specify both SCORE and VARIANCE on the same MATRIX subcommand, you must specify two different files.

The variables in the SCORE matrix data file and their values are:

ROWTYPE_ *String variable containing the value ROW for all of the rows and COLUMN for all of the columns.*

LEVEL *String variable containing the values (or value labels, if present) of each original variable.*

VARNAME_ *String variable containing the original variable names.*

DIM1...DIMn *Numeric variables containing the row and column scores for each dimension.* Each variable is labeled *DIMn*, where *n* represents the dimension number.

The variables in the VARIANCE matrix data file and their values are:

ROWTYPE_ *String variable containing the value COV for all of the cases in the file.*

SCORE *String variable containing the values SINGULAR, ROW, and COLUMN.*

LEVEL *String variable containing the system-missing value for SINGULAR and the sequential row or column number for ROW and COLUMN.*

VARNAME_ *String variable containing the dimension number.*

DIM1...DIMn *Numeric variable containing the covariances for each dimension.* Each variable is labeled *DIMn*, where *n* represents the dimension number.

See the *SPSS Syntax Reference Guide* for more information on matrix data files.

Analyzing Aggregated Data

To analyze aggregated data, such as data from a crosstabulation where cell counts are available but the original raw data are not, you can use the TABLE=ALL option or the WEIGHT command before ANACOR.

Example

To analyze a 3×3 table such as the one shown in Table 1, you could use these commands:

```
DATA LIST FREE/ BIRTHORD ANXIETY COUNT.
BEGIN DATA
1 1 48
1 2 27
1 3 22
2 1 33
2 2 20
2 3 39
3 1 29
3 2 42
3 3 47
END DATA.
WEIGHT BY COUNT.
ANACOR TABLE=BIRTHORD (1,3) BY ANXIETY (1,3).
```

- The WEIGHT command weights each case by the value of *COUNT*, as if there are 48 subjects with *BIRTHORD*=1 and *ANXIETY*=1, 27 subjects with *BIRTHORD*=1 and *ANXIETY*=2, and so on.
- ANACOR can then be used to analyze the data.
- If any of the table cell values equal 0, the WEIGHT command issues a warning, but the ANACOR analysis is done correctly.

- The table cell values (the WEIGHT values) cannot be negative. WEIGHT changes system-missing and negative values to 0.
- For large aggregated tables, you can use the TABLE=ALL option or the transformation language to enter the table "as is."

Table 1 3 x 3 table

		Anxiety		
		High	**Med**	**Low**
Birth order	**First**	48	27	22
	Second	33	20	39
	Other	29	42	47

CATPCA

```
CATPCA [VARIABLES =] varlist

      /ANALYSIS varlist
      [[(WEIGHT={1**}] [LEVEL={SPORD**}] [DEGREE={2}] [INKNOT={2}]]
                {n  }                            {n}          {n}
                              {SPNOM  }  [DEGREE={2}] [INKNOT={2}]
                                                 {n}          {n}
                              {ORDI   }
                              {NOMI   }
                              {MNOM   }
                              {NUME   }

     [/DISCRETIZATION = [varlist [([{GROUPING}] [{NCAT={7}}] [DISTR={NORMAL }])]]]
                                                      {n}          {UNIFORM}
                                    {RANKING }   {EQINTV={n}}
                                   {MULTIPLYING}

      [/MISSING = [varlist [([{PASSIVE**}] [{MODEIMPU}])]]]
                                           {EXTRACAT}
                              {ACTIVE   }  {MODEIMPU}
                                           {EXTRACAT}
                                           {LISTWISE}

      [/SUPPLEMENTARY = [OBJECT(varlist)] [VARIABLE(varlist)]]

      [/CONFIGURATION = [{INITIAL}] (file)]
                         {FIXED  }

      [/DIMENSION = {2**}]
                    {n  }

      [/NORMALIZATION = {VPRINCIPAL**}]
                        {OPRINCIPAL  }
                        {SYMMETRICAL }
                        {INDEPENDENT }
                        {n           }

      [/MAXITER = {100**}]
                  {n    }

      [/CRITITER = {.00001**}]
                   {value   }

     [/PRINT = [DESCRIP**[(varlist)]] [VAF] [LOADING**][QUANT[(varlist)]][HISTORY]
               [CORR**] [OCORR] [OBJECT[([(varname)]varlist)]] [NONE]]

      [/PLOT = [OBJECT**[(varlist)][(n)]]
               [LOADING**[(varlist [(CENTR[(varlist)])])][(n)]]
               [CATEGORY (varlist)[(n)]]
               [JOINTCAT[({varlist})][(n)]] [TRANS[(varlist[({1})])]
                                                    {n}
               [BIPLOT[({LOADING}[(varlist)])[(varlist)]] [(n)]]
                        {CENTR  }
               [TRIPLOT[(varlist[(varlist)])][(n)]]
               [RESID(varlist[({1})])[(l)]]
               [PROJCENTR(varname, varlist)[(n)]] [NONE]]
                                        {n}

      [/SAVE = [TRDATA[({TRA     }[{n}])]] [OBJECT[({OBSCO   }[{n}])]]
                        {rootname}                 {rootname}
               [APPROX[({APP     })]]]
                        {rootname}

      [/OUTFILE = [TRDATA*[(file)]] [DISCRDATA[(file)]]
                  [OBJECT[(file)]] [APPROX[(file)]]].
```

** Default if the subcommand is omitted.

Overview

CATPCA performs principal components analysis on a set of variables. The variables can be given mixed optimal scaling levels, and the relationships among observed variables are not assumed to be linear.

In CATPCA, dimensions correspond to components (that is, an analysis with two dimensions results in two components), and object scores correspond to component scores.

Options

Optimal scaling level. You can specify the optimal scaling level (spline ordinal, spline nominal, ordinal, nominal, multiple nominal, or numerical) at which you want to analyze each variable.

Discretization. You can use the DISCRETIZATION subcommand to discretize fractional-value variables or to recode categorical variables.

Missing data. You can specify the treatment of missing data on a per variable basis with the MISSING subcommand.

Supplementary objects and variables. You can specify objects and variables that you want to treat as supplementary to the analysis and then fit them into the solution.

Read configuration. CATPCA can read a principal components configuration from a file through the CONFIGURATION subcommand. This can be used as the starting point for your analysis or as a fixed solution in which to fit objects and variables.

Number of dimensions. You can specify how many dimensions (components) CATPCA should compute.

Normalization. You can specify one of five different options for normalizing the objects and variables.

Tuning the algorithm. You can control the values of algorithm-tuning parameters with the MAXITER and CRITITER subcommands.

Optional output. You can request optional output through the PRINT subcommand.

Optional plots. You can request a plot of object points, transformation plots per variable, and plots of category points per variable or a joint plot of category points for specified variables. Other plot options include residuals plots, a biplot, a triplot, component loadings plot, and a plot of projected centroids.

Writing discretized data, transformed data, object (component) scores, and approximations. You can write the discretized data, transformed data, object scores, and approximations to external files for use in further analyses.

Saving transformed data, object (component) scores, and approximations. You can save the transformed variables, object scores, and approximations to the working data file.

Basic Specification

The basic specification is the CATPCA command with the VARIABLES and ANALYSIS subcommands.

Syntax Rules

- The VARIABLES and ANALYSIS subcommands must always appear, and the VARIABLES subcommand must be the first subcommand specified. The other subcommands can be specified in any order.
- Variables specified in the ANALYSIS subcommand must be found in the VARIABLES subcommand.
- Variables specified in the SUPPLEMENTARY subcommand must be found in the ANALYSIS subcommand.

Operations

- If a subcommand is repeated, it causes a syntax error and the procedure terminates.

Limitations

- CATPCA operates on category indicator variables. The category indicators should be positive integers. You can use the DISCRETIZATION subcommand to convert fractional-value variables and string variables into positive integers.
- In addition to system-missing values and user-defined missing values, CATPCA treats category indicator values less than 1 as missing. If one of the values of a categorical variable has been coded 0 or a negative value and you want to treat it as a valid category, use the COMPUTE command to add a constant to the values of that variable such that the lowest value will be 1 (see the COMPUTE command or the *SPSS Base User's Guide* for more information on COMPUTE). You can also use the RANKING option of the DISCRETIZATION subcommand for this purpose, except for variables you want to treat as numerical, since the characteristic of equal intervals in the data will not be maintained.
- There must be at least three valid cases.
- Split-file has no implications for CATPCA.

Example

```
CATPCA VARIABLES = TEST1 TEST2 TEST3 TO TEST6 TEST7 TEST8
  /ANALYSIS = TEST1 TO TEST2(WEIGHT=2 LEVEL=ORDI)
              TEST3 TO TEST5(LEVEL=SPORD INKNOT=3)
              TEST6 TEST7(LEVEL=SPORD DEGREE=3)
              TEST8(LEVEL=NUME)
  /DISCRETIZATION = TEST1(GROUPING NCAT=5 DISTR=UNIFORM)
              TEST6(GROUPING) TEST8(MULTIPLYING)
  /MISSING = TEST5(ACTIVE) TEST6(ACTIVE EXTRACAT) TEST8(LISTWISE)
  /SUPPLEMENTARY = OBJECT(1 3) VARIABLE(TEST1)
  /CONFIGURATION = ('iniconf.sav')
  /DIMENSION = 2
  /NORMALIZATION = VPRINCIPAL
  /MAXITER = 150
  /CRITITER = .000001
  /PRINT =  DESCRIP LOADING CORR QUANT(TEST1 TO TEST3) OBJECT
  /PLOT = TRANS(TEST2 TO TEST5) OBJECT(TEST2 TEST3)
  /SAVE = TRDATA OBJECT
  /OUTFILE = TRDATA('c:\data\trans.sav') OBJECT('c:\data\obs.sav').
```

- VARIABLES defines variables. The keyword TO refers to the order of the variables in the working data file.
- The ANALYSIS subcommand defines variables used in the analysis. It is specified that *TEST1* and *TEST2* have a weight of 2. For the other variables, WEIGHT is not specified; thus, they have the default weight value of 1. The optimal scaling level for *TEST1* and *TEST2* is ordinal, for *TEST3* to *TEST7* spline ordinal, and for *TEST8* numerical. The keyword TO refers to the order of the variables in the VARIABLES subcommand. The splines for *TEST3* to *TEST5* have degree 2 (default because unspecified) and 3 interior knots. The splines for *TEST6* and *TEST7* have degree 3 and 2 interior knots (default because unspecified).
- DISCRETIZATION specifies that *TEST6* and *TEST8*, which are fractional-value variables, are discretized: *TEST6* by recoding into 7 categories with a normal distribution (default because unspecified) and *TEST8* by "multiplying." *TEST1*, which is a categorical variable, is recoded into 5 categories with a close-to-uniform distribution.
- MISSING specifies that objects with missing values on *TEST5* and *TEST6* are included in the analysis; missing values on *TEST5* are replaced with the mode (default if not specified) and missing values on *TEST6* are treated as an extra category. Objects with a missing value on *TEST8* are excluded from the analysis. For all other variables, the default is in effect; that is, missing values (*Note*: values, not objects) are excluded from the analysis.
- CONFIGURATION specifies *iniconf.sav* as the file containing the coordinates of a configuration that is to be used as the initial configuration (default because unspecified).
- DIMENSION specifies the number of dimensions to be 2; that is, 2 components are computed. This is the default, so this subcommand could be omitted here.
- The NORMALIZATION subcommand specifies optimization of the association between variables, and the normalization is given to the objects. This is the default, so this subcommand could be omitted here.
- MAXITER specifies the maximum number of iterations to be 150 instead of the default value of 100.
- CRITITER sets the convergence criterion to a value smaller than the default value.

- PRINT specifies descriptives, component loadings and correlations (all default), quantifications for *TEST1* to *TEST3*, and the object (component) scores.
- PLOT is used to request transformation plots for the variables *TEST2* to *TEST5*, an object points plot labeled with the categories of *TEST2*, and an object points plot labeled with the categories of *TEST3*.
- The SAVE subcommand adds the transformed variables and the component scores to the working data file.
- The OUTFILE subcommand writes the transformed data to a data file called *trans.sav* and the component scores to a data file called *obs.sav*, both in the directory *c:\data*.

VARIABLES Subcommand

VARIABLES specifies the variables that may be analyzed in the current CATPCA procedure.

- The VARIABLES subcommand is required and precedes all other subcommands. The actual keyword VARIABLES can be omitted.
- At least two variables must be specified, except if the CONFIGURATION subcommand is used with the FIXED keyword.
- The keyword TO on the VARIABLES subcommand refers to the order of variables in the working data file. This behavior of TO is different from that in the variable list in the ANALYSIS subcommand.

ANALYSIS Subcommand

ANALYSIS specifies the variables to be used in the computations, the optimal scaling level, and the variable weight for each variable or variable list. ANALYSIS also specifies supplementary variables and their optimal scaling level. No weight can be specified for supplementary variables.

- At least two variables must be specified, except if the CONFIGURATION subcommand is used with the FIXED keyword.
- All the variables on ANALYSIS must be specified on the VARIABLES subcommand.
- The ANALYSIS subcommand is required and follows the VARIABLES subcommand.
- The keyword TO in the variable list honors the order of variables in the VARIABLES subcommand.
- Optimal scaling levels and variable weights are indicated by the keywords LEVEL and WEIGHT in parentheses following the variable or variable list.

WEIGHT *Specifies the variable weight with a positive integer.* The default value is 1. If WEIGHT is specified for supplementary variables, it is ignored and a syntax warning is issued.

LEVEL *Specifies the optimal scaling level.*

Level Keyword

The following keywords are used to indicate the optimal scaling level:

SPORD *Spline ordinal (monotonic).* This is the default. The order of the categories of the observed variable is preserved in the optimally scaled variable. Category points will be on a straight line (vector) through the origin. The resulting transformation is a smooth monotonic piecewise polynomial of the chosen degree. The pieces are specified by the user-specified number and procedure-determined placement of the interior knots.

SPNOM *Spline nominal (nonmonotonic).* The only information in the observed variable that is preserved in the optimally scaled variable is the grouping of objects in categories. The order of the categories of the observed variable is not preserved. Category points will lie on a straight line (vector) through the origin. The resulting transformation is a smooth, possibly nonmonotonic, piecewise polynomial of the chosen degree. The pieces are specified by the user-specified number and procedure-determined placement of the interior knots.

MNOM *Multiple nominal.* The only information in the observed variable that is preserved in the optimally scaled variable is the grouping of objects in categories. The order of the categories of the observed variable is not preserved. Category points will be in the centroid of the objects in the particular categories. Multiple indicates that different sets of quantifications are obtained for each dimension.

ORDI *Ordinal.* The order of the categories on the observed variable is preserved in the optimally scaled variable. Category points will be on a straight line (vector) through the origin. The resulting transformation fits better than SPORD transformation but is less smooth.

NOMI *Nominal.* The only information in the observed variable that is preserved in the optimally scaled variable is the grouping of objects in categories. The order of the categories of the observed variable is not preserved. Category points will be on a straight line (vector) through the origin. The resulting transformation fits better than SPNOM transformation but is less smooth.

NUME *Numerical.* Categories are treated as equally spaced (interval level). The order of the categories and the equal distances between category numbers of the observed variables are preserved in the optimally scaled variable. Category points will be on a straight line (vector) through the origin. When all variables are scaled at the numerical level, the CATPCA analysis is analogous to standard principal components analysis.

SPORD and SPNOM Keywords

The following keywords are used with SPORD and SPNOM:

DEGREE *The degree of the polynomial.* It can be any positive integer. The default degree is 2.

INKNOT *The number of interior knots.* The minimum is 0, and the maximum is the number of categories of the variable minus 2. The procedure adjusts the number of interior knots to the maximum if the specified value is too large. The default number of interior knots is 2.

DISCRETIZATION Subcommand

DISCRETIZATION specifies fractional-value variables you want to discretize. Also, you can use DISCRETIZATION for ranking or for two ways of recoding categorical variables.

- A string variable's values are always converted into positive integers, according to the internal numeric representations. DISCRETIZATION for string variables applies to these integers.
- When the DISCRETIZATION subcommand is omitted or when the DISCRETIZATION subcommand is used without a variable list, fractional-value variables are converted into positive integers by grouping them into seven categories with a close to "normal" distribution.
- When no specification is given for variables in a variable list following DISCRETIZATION, these variables are grouped into seven categories with a close to "normal" distribution.
- In CATPCA, values less than 1 are considered to be missing (see MISSING subcommand). However, when discretizing a variable, values less than 1 are considered to be valid and are thus included in the discretization process.

GROUPING *Recode into the specified number of categories.*

RANKING *Rank cases.* Rank 1 is assigned to the case with the smallest value on the variable.

MULTIPLYING *Multiplying the standardized values of a fractional-value variable by 10, rounding, and adding a value such that the lowest value is 1.*

GROUPING Keyword

GROUPING has the following keywords:

NCAT *Number of categories.* When NCAT is not specified, the number of categories is set to 7. You may either specify a number of categories or use the keyword DISTR.

EQINTV *Recode intervals of equal size.* The size of the intervals must be specified (no default). The resulting number of categories depends on the interval size.

DISTR Keyword

DISTR has the following keywords:

NORMAL *Normal distribution.* This is the default when DISTR is not specified.

UNIFORM *Uniform distribution.*

MISSING Subcommand

In CATPCA, we consider a system-missing value, user-defined missing values, and values less than 1 as missing values. The MISSING subcommand allows you to indicate how to handle missing values for each variable.

PASSIVE *Exclude missing values on a variable from analysis.* This is the default when MISSING is not specified. Passive treatment of missing values means that in optimizing the quantification of a variable, only objects with nonmissing values on the variable are involved and that only the nonmissing values of variables contribute to the solution. Thus, when PASSIVE is specified, missing values do not affect the analysis. Further, if all variables are given passive treatment of missing values, then objects with missing values on every variable are treated as supplementary.

ACTIVE *Impute missing values.* You can choose to use mode imputation. You can also consider objects with missing values on a variable as belonging to the same category and impute missing values with an extra category indicator.

LISTWISE *Exclude cases with missing value on a variable.* The cases used in the analysis are cases without missing values on the variables specified. This is the default applied to all variables when the MISSING subcommand is omitted or is specified without variable names or keywords. Also, any variable that is not included in the subcommand receives this specification.

- The ALL keyword may be used to indicate all variables. If it is used, it must be the only variable specification.
- A mode or extracat imputation is done before listwise deletion.

PASSIVE Keyword

If correlations are requested on the PRINT subcommand and passive treatment of missing values is specified for a variable, the missing values must be imputed. For the correlations of the quantified variables, you can specify the imputation with one of the following keywords:

MODEIMPU *Impute missing values on a variable with the mode of the quantified variable.* MODEIMPU is the default.

EXTRACAT *Impute missing values on a variable with the quantification of an extra category.* This implies that objects with a missing value are considered to belong to the same (extra) category.

Note that with passive treatment of missing values, imputation applies only to correlations and is done afterward. Thus, the imputation has no effect on the quantification or the solution.

ACTIVE Keyword

The ACTIVE keyword has the following keywords:

MODEIMPU *Impute missing values on a variable with the most frequent category (mode).* When there are multiple modes, the smallest category indicator is used. MODEIMPU is the default.

EXTRACAT *Impute missing values on a variable with an extra category indicator.* This implies that objects with a missing value are considered to belong to the same (extra) category.

Note that with active treatment of missing values, imputation is done before the analysis starts and thus will affect the quantification and the solution.

SUPPLEMENTARY Subcommand

The SUPPLEMENTARY subcommand specifies the objects and/or variables that you want to treat as supplementary. Supplementary variables must be found in the ANALYSIS subcommand. You cannot weight supplementary objects and variables (specified weights are ignored). For supplementary variables, all options on the MISSING subcommand can be specified except LISTWISE.

- The SUPPLEMENTARY subcommand is ignored when CONFIGURATION=FIXED.

OBJECT *Objects you want to treat as supplementary are indicated with an object number list in parentheses following OBJECT.* The keyword TO is allowed.

VARIABLE *Variables you want to treat as supplementary are indicated with a variable list in parentheses following VARIABLE.* The keyword TO is allowed and honors the order of variables in the VARIABLES subcommand.

CONFIGURATION Subcommand

The CONFIGURATION subcommand allows you to read data from a file containing the coordinates of a configuration. The first variable in this file should contain the coordinates for the first dimension, the second variable should contain the coordinates for the second dimension, and so forth.

INITIAL(file) *Use configuration in the external file as the starting point of the analysis.*

FIXED(file) *Fit objects and variables in the fixed configuration found in the external file.* The variables to fit in should be specified on the ANALYSIS subcommand but will be treated as supplementary. The SUPPLEMENTARY subcommand and variable weights are ignored.

DIMENSION Subcommand

DIMENSION specifies the number of dimensions (components) you want CATPCA to compute.

- The default number of dimensions is 2.
- DIMENSION is followed by an integer indicating the number of dimensions.
- If there are no variables specified as MNOM (multiple nominal), the maximum number of dimensions you can specify is the smaller of the number of observations minus 1 and the total number of variables.
- If some or all of the variables are specified as MNOM (multiple nominal), the maximum number of dimensions is the smaller of a) the number of observations minus 1 or b) the total number of valid MNOM variable levels (categories) plus the number of SPORD, SPNOM, ORDI, NOMI, and NUME variables minus the number of MNOM variables without missing values.
- CATPCA adjusts the number of dimensions to the maximum if the specified value is too large.
- The minimum number of dimensions is 1.

NORMALIZATION Subcommand

The NORMALIZATION subcommand specifies one of five options for normalizing the object scores and the variables. Only one normalization method can be used in a given analysis.

VPRINCIPAL *This option optimizes the association between variables.* With VPRINCIPAL, the coordinates of the variables in the object space are the component loadings (correlations with principal components such as dimensions and object scores). This is the default if the NORMALIZATION subcommand is not specified. This is useful when you are primarily interested in the correlations between the variables.

OPRINCIPAL *This option optimizes distances between objects.* This is useful when you are primarily interested in differences or similarities between the objects.

SYMMETRICAL *Use this normalization option if you are primarily interested in the relation between objects and variables.*

INDEPENDENT *Use this normalization option if you want to examine distances between objects and correlations between variables separately.*

The fifth method allows the user to specify any real value in the closed interval [−1 , 1]. A value of 1 is equal to the OPRINCIPAL method, a value of 0 is equal to the SYMMETRICAL method, and a value of −1 is equal to the VPRINCIPAL method. By specifying a value greater than −1 and less than 1, the user can spread the eigenvalue over both objects and variables. This method is useful for making a tailor-made biplot or triplot. If the user specifies a value outside of this interval, the procedure issues a syntax error message and terminates.

MAXITER Subcommand

MAXITER specifies the maximum number of iterations the procedure can go through in its computations. If not all variables are specified as NUME and/or MNOM, the output starts from iteration 0, which is the last iteration of the initial phase, in which all variables except MNOM variables are treated as NUME.

- If MAXITER is not specified, the maximum number of iterations is 100.
- The specification on MAXITER is a positive integer indicating the maximum number of iterations. There is no uniquely predetermined (that is, hard-coded) maximum for the value that can be used.

CRITITER Subcommand

CRITITER specifies a convergence criterion value. CATPCA stops iterating if the difference in fit between the last two iterations is less than the CRITITER value.

- If CRITITER is not specified, the convergence value is 0.00001.
- The specification on CRITITER is any value less than or equal to 0.1.

PRINT Subcommand

The model summary and the HISTORY statistics for the last iteration are always displayed. That is, they cannot be controlled by the PRINT subcommand. The PRINT subcommand controls the display of additional optional output. The output of the procedure is based on the transformed variables. However, the correlations of the original variables can be requested as well by the keyword OCORR.

The default keywords are DESCRIP, LOADINGS, and CORR. However, when some keywords are specified, the default is nullified and only what was specified comes into effect. If a keyword is duplicated or if a contradicting keyword is encountered, then the last one silently becomes effective (in case of contradicting use of NONE, this means that only the keywords following NONE are effective). For example,

```
/PRINT <=> /PRINT = DESCRIP LOADING CORR
```

```
/PRINT = VAF VAF <=> /PRINT = VAF
```

```
/PRINT = VAF NONE CORR <=> /PRINT = CORR
```

If a keyword that can be followed by a variable list is duplicated, it will cause a syntax error, and the procedure will terminate.

The following keywords can be specified:

DESCRIP(varlist) *Descriptive statistics (frequencies, missing values, optimal scaling level, and mode).* The variables in the varlist must be specified on the VARIABLES subcommand but need not appear on the ANALYSIS subcommand. If DESCRIP is not followed by a varlist, descriptives tables are displayed for all the variables in the varlist on the ANALYSIS subcommand.

VAF *Variance accounted for (centroid coordinates, line coordinates, and total) per variable and per dimension.*

LOADING *Component loadings for variables with optimal scaling level that result in line quantification* (that is, SPORD, SPNOM, ORDI, NOMI, and NUME).

QUANT(varlist) *Category quantifications and category coordinates for each dimension.* Any variable in the ANALYSIS subcommand may be specified in parentheses after QUANT. (For MNOM variables, the coordinates are the quantifications.) If QUANT is not followed by a variable list, quantification tables are displayed for all variables in the varlist on the ANALYSIS subcommand.

HISTORY *History of iterations.* For each iteration (including 0), the variance accounted for, the variance not accounted for, and the increase in variance accounted for are shown.

CORR *Correlations of the transformed variables and the eigenvalues of this correlation matrix.* If the analysis includes variables with optimal scaling level MNOM, *ndim* (the number of dimensions in the analysis) correlation matrices are computed; in the *i*th matrix, the quantifications of dimension *i*, *i* = 1,...*ndim*, of MNOM variables are used to compute the correlations. For variables with missing values specified to be treated as PASSIVE on the MISSING subcommand, the missing values are imputed according to the specification on the PASSIVE keyword (if not specified, mode imputation is used).

OCORR *Correlations of the original variables and the eigenvalues of this correlation matrix.* For variables with missing values specified to be treated as PASSIVE or ACTIVE on the MISSING subcommand, the missing values are imputed with the variable mode.

OBJECT((varname)varlist)

Object scores (component scores). Following the keyword, a varlist can be given in parentheses to display variables (category indicators) along with object scores. If you want to use a variable to label the objects, this variable must occur in parentheses as the first variable in the varlist. If no labeling variable is specified, the objects are labeled with case numbers. The variables to display along with the object scores and the variable to label the objects must be specified on the VARIABLES subcommand but need not appear on the ANALYSIS subcommand. If no variable list is given, only the object scores are displayed.

NONE *No optional output is displayed.* The only output shown is the model summary and the HISTORY statistics for the last iteration.

The keyword TO in a variable list can only be used with variables that are in the ANALYSIS subcommand, and TO applies only to the order of the variables in the ANALYSIS subcommand. For variables that are in the VARIABLES subcommand but not in the ANALYSIS subcommand, the keyword TO cannot be used. For example, if `/VARIABLES = v1 TO v5` and `/ANALYSIS = v2 v1 v4`, then `/PLOT OBJECT(v1 TO v4)` will give two object plots, one labeled with *v1* and one labeled with *v4*.

PLOT Subcommand

The PLOT subcommand controls the display of plots. The default keywords are OBJECT and LOADING. That is, the two keywords are in effect when the PLOT subcommand is omitted or when the PLOT subcommand is given without any keyword. If a keyword is duplicated (for example, /PLOT = RESID RESID), then only the last one is effective. If the keyword NONE is used together with other keywords (for example, /PLOT = RESID NONE LOADING), then only the keywords following NONE are effective. That is, when keywords contradict, the later one overwrites the earlier ones.

- All the variables to be plotted must be specified on the ANALYSIS subcommand.
- If the variable list following the keywords CATEGORIES, TRANS, RESID, and PROJCENTR is empty, it will cause a syntax error, and the procedure will terminate.
- The variables in the variable list for labeling the object point following OBJECT, BIPLOT, and TRIPLOT must be specified on the VARIABLES subcommand but need not appear on the ANALYSIS subcommand. This means that variables not included in the analysis can still be used to label plots.
- The keyword TO in a variable list can only be used with variables that are in the ANALYSIS subcommand, and TO applies only to the order of the variables in the ANALYSIS subcommand. For variables that are in the VARIABLES subcommand but not in the ANALYSIS subcommand, the keyword TO cannot be used. For example, if `/VARIABLES = v1 TO v5` and `/ANALYSIS = v2 v1 v4`, then `/PLOT OBJECT(v1 TO v4)` will give two object plots, one labeled with *v1* and one labeled with *v4*.
- For a one-dimensional solution, only unidimensional plots (transformation plot, residuals plot, and plot of projected centroids) are produced.
- For multidimensional plots, all of the dimensions specified on the DIMENSION subcommand are produced in a matrix scatterplot if the specified number of dimensions is greater than two; if the specified number of dimensions is two, a scatterplot is produced.

The following keywords can be specified:

OBJECT (varlist)(n) *Plots of the object points.* Following the keyword, a list of variables in parentheses can be given to indicate that plots of object points labeled with the categories of the variables should be produced (one plot for each variable). If the variable list is omitted, a plot labeled with case numbers is produced.

CATEGORY(varlist)(n)

Plots of the category points. Both the centroid coordinates and the line coordinates are plotted. A list of variables must be given in parentheses following the keyword. For variables with optimal scaling level MNOM, categories are in the centroids of the objects. For all other optimal scaling levels, categories are on a vector through the origin.

LOADING(varlist(CENTR(varlist)))(l)

Plot of the component loadings optionally with centroids. By default, all variables with an optimal scaling level that results in vector quantification (that is, SPORD, SPNOM, ORDI, NOMI, and NUME) are included in this plot. LOADING can be followed by a varlist to select the loadings to include in the plot. When "LOADING(" or the varlist following "LOADING(" is followed by the keyword CENTR in parentheses, centroids are plotted for all variables with optimal scaling level MNOM. CENTR can be followed by a varlist in parentheses to select MNOM variables whose centroids are to be included in the plot. When there is no variable whose optimal scaling level is SPORD, SPNOM, ORDI, NOMI, or NUME in the analysis, this plot cannot be produced.

TRANS(varlist(n))

Transformation plots (optimal category quantifications against category indicators). A list of variables must be given in parentheses following the keyword. MNOM variables in the varlist can be followed by a number of dimensions in parentheses to indicate that you want to display *p* transformation plots, one for each of the first *p* dimensions.

RESID(varlist(n))(n)

Plot of residuals per variable. Following the keyword, a list of variables in parentheses must be given. A variable with optimal scaling level MNOM can be followed by a number in parentheses to indicate the number of dimensions you want a residuals plot for. If the number of dimensions is not specified, a plot for the first dimension is produced.

BIPLOT(keyword(varlist))(varlist)(n)

Plot of objects and variables. The coordinates for the variables can be chosen to be component loading or centroids, using keywords LOADING and/or CENTR in parentheses following BIPLOT. When no keyword is given, component loadings are plotted. When NORMALIZATION = INDEPENDENT, this plot is incorrect and therefore not available.

Following LOADING and CENTR, a list of variables in parentheses can be given to indicate the variables to be included in the plot. If the variable list is omitted, a plot including all variables is produced. Following BIPLOT, a list of variables in parentheses can be given to indicate that plots with objects labeled with the categories of the variables should be produced (one plot for each variable). If the variable list is omitted, a plot with objects labeled with case numbers is produced.

TRIPLOT(varlist(varlist))(n)

A plot of object points, component loadings for variables with an optimal scaling level that results in line quantification (that is, SPORD*,* SPNOM*,* ORDI*,* NOMI*, and* NUME*), and centroids for variables with optimal scaling level* MNOM*.* Following the keyword, a list of variables in parentheses can be given to indicate the variables to include in the plot. If the variable list is omitted, all variables are included. The varlist can contain a second varlist in parentheses to indicate that triplots with objects labeled with the categories of the variables in this variable list should be produced (one plot for each variable). If this second variable list is omitted, a plot with objects labeled with case numbers is produced. When NORMALIZATION = INDEPENDENT, this plot is incorrect and therefore not available.

JOINTCAT(varlist)(n) *Joint plot of the category points for the variables in the varlist.* If no varlist is given, the category points for all variables are displayed.

PROJCENTR(varname, varlist)(n)

Plot of the centroids of a variable projected on each of the variables in the varlist. You cannot project centroids of a variable on variables with MNOM optimal scaling; thus, a variable that has MNOM optimal scaling can be specified as the variable to be projected but not in the list of variables to be projected on. When this plot is requested, a table with the coordinates of the projected centroids is also displayed.

NONE *No plots.*

BIPLOT Keyword

BIPLOT takes the following keywords:

LOADING(varlist) *Object points and component loadings.*

CENTR(varlist) *Object points and centroids.*

For all of the keywords except TRANS and NONE, the user can specify an optional parameter in order to control the global upper boundary of variable/category label lengths in the plot. Note that this boundary is applied uniformly to all variables in the list.

The variable/category label-length parameter can take any non-negative integer less than or equal to 20. The default length is 20. If the length is set to 0, names/values instead of variable/value labels are displayed to indicate variables/categories. If the specified length is greater than 20, the procedure simply resets it to 20.

When variables/values do not have labels, then the names/values themselves are used as the labels.

SAVE Subcommand

The SAVE subcommand is used to add the transformed variables (category indicators replaced with optimal quantifications), the object scores, and the approximation to the working data file. Excluded cases are represented by a dot (the system-missing symbol) on every saved variable.

TRDATA *Transformed variables.* Missing values specified to be treated as passive are represented by a dot.

OBJECT *Object (component) scores.*

APPROX *Approximation for variables that do not have optimal scaling level MNOM.*

- Following TRDATA, a rootname and the number of dimensions to be saved for variables specified as MNOM can be specified in parentheses.
- For variables that are not specified as MNOM, CATPCA adds two numbers separated by the symbol _. For variables that are specified as MNOM, CATPCA adds three numbers. The first number uniquely identifies the source variable names and the last number uniquely identifies the CATPCA procedures with the successfully executed SAVE subcommands. For variables that are specified as MNOM, the middle number corresponds to the dimension number (see the next bullet for more details). Only one rootname can be specified, and it can contain up to five characters for variables that are not specified as MNOM and three characters for variables that are specified as MNOM (if more than one rootname is specified, the first rootname is used; if a rootname contains more than five/three characters, the first five/three characters are used at most).
- If a rootname is not specified for TRDATA, rootname *TRA* is used to automatically generate unique variable names. The formulas are *ROOTNAMEk_n* and *ROOTNAMEk_m_n*, where *k* increments from 1 to identify the source variable names by using the source variables' position numbers in the ANALYSIS subcommand, *m* increments from 1 to identify the dimension number, and *n* increments from 1 to identify the CATPCA procedures with the successfully executed SAVE subcommands for a given data file in a continuous SPSS session. For example, with three variables specified on ANALYSIS, LEVEL = MNOM for the second variable, and two dimensions to save, the first set of default names, if they do not exist in the data file, would be *TRA1_1*, *TRA2_1_1*, *TRA2_2_1*, and *TRA3_1*. The next set of default names, if they do not exist in the data file, would be *TRA1_2*, *TRA2_1_2*, *TRA2_2_2*, and *TRA3_2*. However, if, for example, *TRA1_2* already exists in the data file, then the default names should be attempted as *TRA1_3*, *TRA2_1_3*, *TRA2_2_3*, and *TRA3_3*. That is, the last number increments to the next available integer.
- As *k* and/or *m* and/or *n* increase for TRDATA, the rootname is truncated to keep variable names within eight characters. For example, if *TRANS* is specified as rootname, *TRANS1_9* would be followed by *TRAN1_10*. Note that the truncation is done variable-wise, not analysis-wise.
- Following OBJECT, a rootname and the number of dimensions can be specified in parentheses to which CATPCA adds two numbers separated by the symbol _. The first number corresponds to the dimension number. The second number uniquely identifies the CATPCA procedures with the successfully executed SAVE subcommands (see the next bullet for more details). Only one rootname can be specified, and it can contain up to five

characters (if more than one rootname is specified, the first rootname is used; if a rootname contains more than five characters, the first five characters are used at most).

- If a rootname is not specified for OBJECT, rootname *OBSCO* is used to automatically generate unique variable names. The formula is *ROOTNAMEm_n*, where *m* increments from 1 to identify the dimension number and *n* increments from 1 to identify the CATPCA procedures with the successfully executed SAVE subcommands for a given data file in a continuous SPSS session. For example, if two dimensions are specified following OBJECT, the first set of default names, if they do not exist in the data file, would be *OBSCO1_1* and *OBSCO2_1*. The next set of default names, if they do not exist in the data file, would be *OBSCO1_2* and *OBSCO2_2*. However, if, for example, *OBSCO2_2* already exists in the data file, then the default names should be attempted as *OBSCO1_3* and *OBSCO2_3*. That is, the second number increments to the next available integer.
- As *m* and/or *n* increase for OBJECT, the rootname is truncated to keep variable names within eight characters. For example, *OBSCO9_1* would be followed by *OBSC10_1*. The initial character (*O* for the default rootnames) is required. Note that the truncation is done variable-wise, not analysis-wise.
- Following APPROX, a rootname can be specified in parentheses, to which CATPCA adds two numbers separated by the symbol _. The first number uniquely identifies the source variable names, and the last number uniquely identifies the CATPCA procedures with the successfully executed SAVE subcommands (see the next bullet for more details). Only one rootname can be specified, and it can contain up to five characters (if more than one rootname is specified, the first rootname is used; if a rootname contains more than five characters, the first five characters are used at most).
- If a rootname is not specified for APPROX, rootname *APP* is used to automatically generate unique variable names. The formula is *ROOTNAMEk_n*, where *k* increments from 1 to identify the source variable names by using the source variables' position numbers in the ANALYSIS subcommand, and *n* increments from 1 to identify the CATPCA procedures with the successfully executed SAVE subcommands for a given data file in a continuous SPSS session. For example, with three variables specified on ANALYSIS, and LEVEL = MNOM for the second variable, the first set of default names, if they do not exist in the data file, would be *APP1_1*, *APP2_1*, and *APP3_1*. The next set of default names, if they do not exist in the data file, would be *APP1_2*, *APP2_2*, and *APP3_2*. However, if, for example, *APP1_2* already exists in the data file, then the default names should be attempted as *APP1_3*, *APP2_3*, and *APP3_3*. That is, the last number increments to the next available integer.
- As *k* and/or *n* increase for APPROX, the rootname is truncated to keep variable names within eight characters. For example, if *APPRO* is specified as a rootname, *APPRO1_9* would be followed by *APPR1_10*. Note that the truncation is done variable-wise, not analysis-wise.
- Variable labels are created automatically. (They are shown in the procedure information table, or the notes table, and can also be displayed in the Data Editor window.)
- If the number of dimensions is not specified, the SAVE subcommand saves all dimensions.

OUTFILE Subcommand

The OUTFILE subcommand is used to write the discretized data, transformed data (category indicators replaced with optimal quantifications), the object scores, and the approximation to an external data file. Excluded cases are represented by a dot (the system-missing symbol) on every saved variable.

DISCRDATA(file) *Discretized data.*

TRDATA(file) *Transformed variables.* Missing values specified to be treated as passive are represented by a dot.

OBJECT(file) *Object (component) scores.*

APPROX(file) A*pproximation for variables that do not have optimal scaling level MNOM.*

- Following the keyword, a filename enclosed by single quotation marks should be specified. The filenames should be different for each of the keywords.

In principle, a working data file should not be replaced by this subcommand, and the asterisk (*) file specification is not supported. This strategy also prevents the OUTFILE interference with the SAVE subcommand.

CATREG

```
CATREG [VARIABLES =] varlist

/ANALYSIS
         depvar [([LEVEL={SPORD**}] [DEGREE={2}] [INKNOT={2}])]
                                             {n}          {n}
                           {SPNOM  }  [DEGREE={2}] [INKNOT={2}]
                                             {n}          {n}
                           {ORDI   }
                           {NOMI   }
                           {NUME   }
    WITH indvarlist [([LEVEL={SPORD**}] [DEGREE={2}] [INKNOT={2}])]
                                           {n}          {n}
                               {SPNOM  }  [DEGREE={2}] [INKNOT={2}]
                                           {n}          {n}
                               {ORDI   }
                               {NOMI   }
                               {NUME   }

[/DISCRETIZATION = [varlist [([{GROUPING   }] [{NCAT*={7}}] [DISTR={NORMAL }])]]]
                                                {n}          {UNIFORM}
                                 {EQINTV=d   }
                                 {RANKING    }
                                 {MULTIPLYING}

[/MISSING = [{varlist}({LISTWISE**})]]
             {ALL**  } {MODEIMPU  }
                       {EXTRACAT  }

[/SUPPLEMENTARY = OBJECT(objlist)]

[/INITIAL = [{NUMERICAL**}]]
             {RANDOM     }

[/MAXITER = [{100**}]]
             {n    }

[/CRITITER = [{.00001**}]]
              {n       }

[/PRINT = [R**] [COEFF**] [DESCRIP**[(varlist)]] [HISTORY] [ANOVA**]
          [CORR] [OCORR] [QUANT[(varlist)]] [NONE]]

[/PLOT = {TRANS(varlist)[(h)]} {RESID(varlist)[(h)]}]

[/SAVE = {TRDATA[({TRA     })]} {PRED[({PRE     })]} {RES[({RES     })]}]
                 {rootname}           {rootname}          {rootname}

[/OUTFILE = {TRDATA('filename')} {DISCRDATA('filename')}] .
```

** Default if subcommand or keyword is omitted.

Overview

CATREG (Categorical regression with optimal scaling using alternating least squares) quantifies categorical variables using optimal scaling, resulting in an optimal linear regression equation for the transformed variables. The variables can be given mixed optimal scaling levels and no distributional assumptions about the variables are made.

Options

Transformation type. You can specify the transformation type (spline ordinal, spline nominal, ordinal, nominal, or numerical) at which you want to analyze each variable.

Discretization. You can use the DISCRETIZATION subcommand to discretize fractional-value variables or to recode categorical variables.

Initial configuration. You can specify the kind of initial configuration through the INITIAL subcommand.

Tuning the algorithm. You can control the values of algorithm-tuning parameters with the MAXITER and CRITITER subcommands.

Missing data. You can specify the treatment of missing data with the MISSING subcommand.

Optional output. You can request optional output through the PRINT subcommand.

Transformation plot per variable. You can request a plot per variable of its quantification against the category numbers.

Residual plot per variable. You can request an overlay plot per variable of the residuals and the weighted quantification, against the category numbers.

Writing external data. You can write the transformed data (category numbers replaced with optimal quantifications) to an outfile for use in further analyses. You can also write the discretized data to an outfile.

Saving variables. You can save the transformed variables, the predicted values, and/or the residuals in the working data file.

Basic Specification

The basic specification is the command CATREG with the VARIABLES and ANALYSIS subcommands.

Syntax Rules

- The VARIABLES and ANALYSIS subcommands must always appear, and the VARIABLES subcommand must be the first subcommand specified. The other subcommands, if specified, can be in any order.
- Variables specified in the ANALYSIS subcommand must be found in the VARIABLES subcommand.
- In the ANALYSIS subcommand, exactly one variable must be specified as a dependent variable and at least one variable must be specified as an independent variable after the keyword WITH.
- The word WITH is reserved as a keyword in the CATREG procedure. Thus, it may not be a variable name in CATREG. Also, the word TO is a reserved word in SPSS.

Operations

- If a subcommand is specified more than once, the last one is executed but with a syntax warning. Note this is true also for the VARIABLES and ANALYSIS subcommands.

Limitations

- If more than one dependent variable is specified in the ANALYSIS subcommand, CATREG is not executed.
- CATREG operates on category indicator variables. The category indicators should be positive integers. You can use the DISCRETIZATION subcommand to convert fractional-value variables and string variables into positive integers. If DISCRETIZATION is not specified, fractional-value variables are automatically converted into positive integers by grouping them into seven categories with a close to normal distribution and string variables are automatically converted into positive integers by ranking.
- In addition to system missing values and user defined missing values, CATREG treats category indicator values less than 1 as missing. If one of the values of a categorical variable has been coded 0 or some negative value and you want to treat it as a valid category, use the COMPUTE command to add a constant to the values of that variable such that the lowest value will be 1. (See the *SPSS Syntax Reference Guide* or the *SPSS Base User's Guide* for more information on COMPUTE). You can also use the RANKING option of the DISCRETIZATION subcommand for this purpose, except for variables you want to treat as numerical, since the characteristic of equal intervals in the data will not be maintained.
- There must be at least three valid cases.
- The number of valid cases must be greater than the number of independent variables plus 1.
- The maximum number of independent variables is 200.
- Split-File has no implications for CATREG.

Example

```
CATREG VARIABLES = TEST1 TEST3 TEST2 TEST4 TEST5 TEST6
                   TEST7 TO TEST9 STATUS01 STATUS02
  /ANALYSIS TEST4 (LEVEL=NUME)
   WITH TEST1 TO TEST2 (LEVEL=SPORD DEGREE=1 INKNOT=3) TEST5 TEST7
   (LEVEL=SPNOM) TEST8 (LEVEL=ORDI) STATUS01 STATUS02 (LEVEL=NOMI)
  /DISCRETIZATION = TEST1(GROUPING NCAT=5 DISTR=UNIFORM)
                    TEST5(GROUPING) TEST7(MULTIPLYING)
  /INITIAL = RANDOM
  /MAXITER = 100
  /CRITITER = .000001
  /MISSING = MODEIMPU
  /PRINT = R COEFF DESCRIP ANOVA QUANT(TEST1 TO TEST2 STATUS01
           STATUS02)
  /PLOT = TRANS (TEST2 TO TEST7 TEST4)
  /SAVE
  /OUTFILE = 'c:\data\qdata.sav'.
```

- VARIABLES defines variables. The keyword TO refers to the order of the variables in the working data file.
- The ANALYSIS subcommand defines variables used in the analysis. It is specified that *TEST4* is the dependent variable, with optimal scaling level numerical and that the variables *TEST1*, *TEST2*, *TEST3*, *TEST5*, *TEST7*, *TEST8*, *STATUS01*, and *STATUS02* are the independent variables to be used in the analysis. (The keyword TO refers to the order of the variables in the VARIABLES subcommand.) The optimal scaling level for *TEST1*, *TEST2*, and *TEST3* is spline ordinal, for *TEST5* and *TEST7* spline nominal, for *TEST8* ordinal, and for *STATUS01* and *STATUS02* nominal. The splines for *TEST1* and *TEST2* have degree 1 and three interior knots, the splines for *TEST5* and *TEST7* have degree 2 and two interior knots (default because unspecified).
- DISCRETIZATION specifies that *TEST5* and *TEST7*, which are fractional-value variables, are discretized: *TEST5* by recoding into seven categories with a normal distribution (default because unspecified) and *TEST7* by "multiplying." *TEST1*, which is a categorical variable, is recoded into five categories with a close-to-uniform distribution.
- Because there are nominal variables, a random initial solution is requested by the INITIAL subcommand.
- MAXITER specifies the maximum number of iterations to be 100. This is the default, so this subcommand could be omitted here.
- CRITITER sets the convergence criterion to a value smaller than the default value.
- To include cases with missing values, the MISSING subcommand specifies that for each variable, missing values are replaced with the most frequent category (the mode).
- PRINT specifies the correlations, the coefficients, the descriptive statistics for all variables, the ANOVA table, the category quantifications for variables *TEST1*, *TEST2*, *TEST3*, *STATUS01*, and *STATUS02*, and the transformed data list of all cases.
- PLOT is used to request quantification plots for the variables *TEST2*, *TEST5*, *TEST7*, and *TEST4*.
- The SAVE subcommand adds the transformed variables to the working data file. The names of these new variables are *TRANS1_1*, ..., *TRANS9_1*.
- The OUTFILE subcommand writes the transformed data to a data file called *qdata.sav* in the directory *c:\data*.

VARIABLES Subcommand

VARIABLES specifies the variables that may be analyzed in the current CATREG procedure.

- The VARIABLES subcommand is required and precedes all other subcommands. The actual keyword VARIABLES can be omitted. (Note that the equals sign is always optional in SPSS syntax.)
- The keyword TO on the VARIABLES subcommand refers to the order of variables in the working data file. (Note that this behavior of TO is different from that in the indvarlist on the ANALYSIS subcommand.)

ANALYSIS Subcommand

ANALYSIS specifies the dependent variable and the independent variables following the keyword WITH.

- All the variables on ANALYSIS must be specified on the VARIABLES subcommand.
- The ANALYSIS subcommand is required and follows the VARIABLES subcommand.
- The first variable list contains exactly one variable as the dependent variable, while the second variable list following WITH contains at least one variable as an independent variable. Each variable may have at most one keyword in parentheses indicating the transformation type of the variable.
- The keyword TO in the independent variable list honors the order of variables on the VARIABLES subcommand.
- Optimal scaling levels are indicated by the keyword LEVEL in parentheses following the variable or variable list.

LEVEL *Specifies the optimal scaling level.*

LEVEL Keyword

The following keywords are used to indicate the optimal scaling level:

SPORD *Spline ordinal (monotonic).* This is the default for a variable listed without any optimal scaling level, for example, one without LEVEL in the parentheses after it or with LEVEL without a specification. Categories are treated as ordered. The order of the categories of the observed variable is preserved in the optimally scaled variable. Categories will be on a straight line through the origin. The resulting transformation is a smooth nondecreasing piecewise polynomial of the chosen degree. The pieces are specified by the number and the placement of the interior knots.

SPNOM *Spline nominal (non-monotonic).* Categories are treated as unordered. Objects in the same category obtain the same quantification. Categories will be on a straight line through the origin. The resulting transformation is a smooth piecewise polynomial of the chosen degree. The pieces are specified by the number and the placement of the interior knots.

ORDI *Ordinal.* Categories are treated as ordered. The order of the categories of the observed variable is preserved in the optimally scaled variable. Categories will be on a straight line through the origin. The resulting transformation fits better than SPORD transformation, but is less smooth.

NOMI *Nominal.* Categories are treated as unordered. Objects in the same category obtain the same quantification. Categories will be on a straight line through the origin. The resulting transformation fits better than SPNOM transformation, but is less smooth.

NUME *Numerical.* Categories are treated as equally spaced (interval level). The order of the categories and the differences between category numbers of the observed variables are preserved in the optimally scaled variable. Categories will be on a straight line through the origin. When all variables are scaled at the numerical level, the CATREG analysis is analogous to standard multiple regression analysis.

SPORD and SPNOM Keywords

The following keywords are used with SPORD and SPNOM :

DEGREE *The degree of the polynomial.* If DEGREE is not specified the degree is assumed to be 2.

INKNOT *The number of the interior knots.* If INKNOT is not specified the number of interior knots is assumed to be 2.

DISCRETIZATION Subcommand

DISCRETIZATION specifies fractional-value variables that you want to discretize. Also, you can use DISCRETIZATION for ranking or for two ways of recoding categorical variables.

- A string variable's values are always converted into positive integers by assigning category indicators according to the ascending alphanumeric order. DISCRETIZATION for string variables applies to these integers.
- When the DISCRETIZATION subcommand is omitted, or when the DISCRETIZATION subcommand is used without a varlist, fractional-value variables are converted into positive integers by grouping them into seven categories (or into the number of distinct values of the variable if this number is less than 7) with a close to normal distribution.
- When no specification is given for variables in a varlist following DISCRETIZATION, these variables are grouped into seven categories with a close-to-normal distribution.
- In CATREG, a system-missing value, user-defined missing values, and values less than 1 are considered to be missing values (see next section). However, in discretizing a variable, values less than 1 are considered to be valid values, and are thus included in the discretization process. System-missing values and user-defined missing values are excluded.

GROUPING *Recode into the specified number of categories.*

RANKING *Rank cases.* Rank 1 is assigned to the case with the smallest value on the variable.

MULTIPLYING *Multiplying the standardized values (z-scores) of a fractional-value variable by 10, rounding, and adding a value such that the lowest value is 1.*

GROUPING Keyword

NCAT *Recode into ncat categories.* When NCAT is not specified, the number of categories is set to 7 (or the number of distinct values of the variable if this number is less than 7). The valid range is from 2 to 36. You may either specify a number of categories or use the keyword DISTR.

EQINTV *Recode intervals of equal size into categories.* The interval size must be specified (there is no default value). The resulting number of categories depends on the interval size.

DISTR Keyword

DISTR has the following keywords:

NORMAL *Normal distribution.* This is the default when DISTR is not specified.

UNIFORM *Uniform distribution.*

MISSING Subcommand

In CATREG, we consider a system missing value, user defined missing values, and values less than 1 as missing values. However, in discretizing a variable (see previous section), values less than 1 are considered as valid values. The MISSING subcommand allows you to indicate how to handle missing values for each variable.

LISTWISE *Exclude cases with missing values on the specified variable(s).* The cases used in the analysis are cases without missing values on the variable(s) specified. This is the default applied to all variables, when the MISSING subcommand is omitted or is specified without variable names or keywords. Also, any variable which is not included in the subcommand gets this specification.

MODEIMPU *Impute missing value with mode.* All cases are included and the imputations are treated as valid observations for a given variable. When there are multiple modes, the smallest mode is used.

EXTRACAT *Impute missing values on a variable with an extra category indicator.* This implies that objects with a missing value are considered to belong to the same (extra) category. This category is treated as nominal, regardless of the optimal scaling level of the variable.

- The ALL keyword may be used to indicate all variables. If it is used, it must be the only variable specification.
- A mode or extra-category imputation is done before listwise deletion.

SUPPLEMENTARY Subcommand

The SUPPLEMENTARY subcommand specifies the objects that you want to treat as supplementary. You cannot weight supplementary objects (specified weights are ignored).

OBJECT *Supplementary objects.* Objects that you want to treat as supplementary are indicated with an object number list in parentheses following OBJECT. The keyword TO is allowed, for example, OBJECT(1 TO 1 3 5 TO 9).

INITIAL Subcommand

INITIAL specifies the method used to compute the initial value/configuration.

- The specification on INITIAL is keyword NUMERICAL or RANDOM. If INITIAL is not specified, NUMERICAL is the default.

NUMERICAL *Treat all variables as numerical.* This is usually best to use when there are only numerical and/or ordinal variables.

RANDOM *Provide a random initial value.* This should be used only when there is at least one nominal variable.

MAXITER Subcommand

MAXITER specifies the maximum number of iterations CATREG can go through in its computations. Note that the output starts from the iteration number 0, which is the initial value before any iteration, when INITIAL = NUMERICAL is in effect.

- If MAXITER is not specified, CATREG will iterate up to 100 times.
- The specification on MAXITER is a positive integer indicating the maximum number of iterations. There is no uniquely predetermined (hard coded) maximum for the value that can be used.

CRITITER Subcommand

CRITITER specifies a convergence criterion value. CATREG stops iterating if the difference in fit between the last two iterations is less than the CRITITER value.

- If CRITITER is not specified, the convergence value is 0.00001.
- The specification on CRITITER is any value less than or equal to 0.1 and greater than or equal to .000001. (Values less than the lower bound might seriously affect performance. Therefore, they are not supported.)

PRINT Subcommand

The PRINT subcommand controls the display of output. The output of the CATREG procedure is always based on the transformed variables. However, the correlations of the original predictor variables can be requested as well by the keyword OCORR. The default keywords are R, COEFF, DESCRIP, and ANOVA. That is, the four keywords are in effect when the PRINT subcommand is omitted or when the PRINT subcommand is given without any keyword. If a keyword is duplicated or it encounters a contradicting keyword, such as /PRINT = R R NONE, then the last one silently becomes effective.

R *Multiple R.* Includes R^2, adjusted R^2, and adjusted R^2 taking the optimal scaling into account.

COEFF *Standardized regression coefficients (beta).* This option gives three tables: a Coefficients table that includes betas, standard error of the betas, *t* values, and significance; a Coefficients-Optimal Scaling table, with the standard error of the betas taking the optimal scaling degrees of freedom into account; and a table with the zero-order, part, and partial correlation, Pratt's relative importance measure for the transformed predictors, and the tolerance before and after transformation. If the tolerance for a transformed predictor is lower than the default tolerance value in the SPSS Regression procedure (0.0001), but higher than 10E–12, this is reported in an annotation. If the tolerance is lower than 10E–12, then the COEFF computation for this variable is not done and this is reported in an annotation. Note that the regression model includes the intercept coefficient but that its estimate does not exist because the coefficients are standardized.

DESCRIP(varlist) *Descriptive statistics (frequencies, missing values, and mode).* The variables in the varlist must be specified on the VARIABLES subcommand, but need not appear on the ANALYSIS subcommand. If DESCRIP is not followed by a varlist, Descriptives tables are displayed for all of the variables in the variable list on the ANALYSIS subcommand.

HISTORY *History of iterations.* For each iteration, including the starting values for the algorithm, the multiple *R* and the regression error (square root of (1–multiple R^2)) are shown. The increase in multiple *R* is listed from the first iteration.

ANOVA *Analysis-of-variance tables.* This option includes regression and residual sums of squares, mean squares and *F*. This options gives two ANOVA tables: one with degrees of freedom for the regression equal to the number of predictor variables and one with degrees of freedom for the regression taking the optimal scaling into account.

CORR *Correlations of the transformed predictors.*

OCORR *Correlations of the original predictors.*

QUANT(varlist) *Category quantifications.* Any variable in the ANALYSIS subcommand may be specified in parentheses after QUANT. If QUANT is not followed by a varlist, Quantification tables are displayed for all variables in the variable list on the ANALYSIS subcommand.

NONE *No* PRINT *output is shown.* This is to suppress the default PRINT output.

- The keyword TO in a variable list can only be used with variables that are in the ANALYSIS subcommand, and TO applies only to the order of the variables in the ANALYSIS subcommand. For variables that are in the VARIABLES subcommand but not in the ANALYSIS subcommand, the keyword TO cannot be used. For example, if /VARIABLES = v1 TO v5 and /ANALYSIS is v2 v1 v4, then /PRINT QUANT(v1 TO v4) will give two quantification plots, one for *v1* and one for *v4*. (/PRINT QUANT(v1 TO v4 v2 v3 v5) will give quantification tables for *v1*, *v2*, *v3*, *v4*, and *v5*.)

PLOT Subcommand

The PLOT subcommand controls the display of plots.

- In this subcommand, if no plot keyword is given, then no plot is created. Further, if the variable list following the plot keyword is empty, then no plot is created, either.
- All the variables to be plotted must be specified in the ANALYSIS subcommand. Further, for the residual plots, the variables must be independent variables.

TRANS(varlist)(l) *Transformation plots (optimal category quantifications against category indicators).* A list of variables must come from the ANALYSIS variable list and must be given in parentheses following the keyword. Further, the user can specify an optional parameter l in parentheses after the variable list in order to control the global upper boundary of category label lengths in the plot. Note that this boundary is applied uniformly to all transformation plots.

RESID(varlist)(l) *Residual plots (residuals when the dependent variable is predicted from all predictor variables in the analysis except the predictor variable in varlist, against category indicators, and the optimal category quantifications multiplied with Beta against category indicators).* A list of variables must come from the ANALYSIS variable list's independent variables and must be given in parentheses following the keyword. Further, the user can specify an optional parameter l in parentheses after the variable list in order to control the global upper boundary of category label lengths in the plot. Note that this boundary is applied uniformly to all residual plots.

- The category label length parameter (l) can take any non-negative integer less than or equal to 20. If l = 0, values instead of value labels are displayed to indicate the categories on the *x* axis in the plot. If l is not specified, CATREG assumes that each value label at its full length is displayed as a plot's category label, but currently LINE CHART in GRAPH limit them to 20. Thus, it is equivalent to (l = 20). (Note that the VALUE LABELS command allows up to 60 characters.) If l is an integer larger than 20, then we reset it to 20 and issue a warning saying l must be a non-negative integer less than or equal to 20.
- If a positive value of l is given, but if some or all of the values do not have value labels, then for those values, the values themselves are used as the category labels, and they obey the label length constraint.

- The keyword TO in a variable list can only be used with variables that are in the ANALYSIS subcommand, and TO applies only to the order of the variables in the ANALYSIS subcommand. For variables that are in the VARIABLES subcommand but not in the ANALYSIS subcommand, the keyword TO cannot be used. For example, if /VARIABLES = v1 TO v5 and /ANALYSIS is v2 v1 v4, then /PLOT TRANS(v1 TO v4) will give two transformation plots, one for *v1* and for *v4*. (/PLOT TRANS(v1 TO v4 v2 v3 v5) will give transformation plots for *v1*, *v2*, *v3*, *v4*, and *v5*.)

SAVE Subcommand

The SAVE subcommand is used to add the transformed variables (category indicators replaced with optimal quantifications), the predicted values, and the residuals to the working data file.

Excluded cases are represented by a dot (the sysmis symbol) on every saved variable.

TRDATA *Transformed variables.*

PRED *Predicted values.*

RES *Residuals.*

- A variable rootname can be specified with each of the keywords. Only one rootname can be specified with each keyword, and it can contain up to five characters (if more than one rootname is specified with a keyword, the first rootname is used; if a rootname contains more than five characters, the first five characters are used at most). If a rootname is not specified, the default rootnames (TRA, PRE, and RES) are used.
- CATREG adds two numbers separated by an underscore (_) to the rootname. The formula is *ROOTNAMEk_n* where *k* increments from 1 to identify the source variable names by using the source variables' position numbers in the ANALYSIS subcommand (that is, the dependent variable has the position number 1, and the independent variables have the position numbers 2, 3, ... as they are listed), and *n* increments from 1 to identify the CATREG procedures with the successfully executed SAVE subcommands for a given data file in a continuous SPSS session. For example, with two predictor variables specified on ANALYSIS, the first set of default names for the transformed data, if they do not exist in the data file, would be *TRA1_1*, for the dependent variable, and *TRA2_1*, *TRA3_1* for the predictor variables. The next set of default names, if they do not exist in the data file, would be *TRA1_2*, *TRA2_2*, *TRA3_2*. However, if, for example, *TRA1_2* already exists in the data file, then the default names should be attempted as *TRA1_3*, *TRA2_3*, *TRA3_3*—that is, the last number increments to the next available integer.
- As *k* and/or *n* increase, the rootname is truncated to keep variable names within eight characters. For example, if *TRANS* is specified as rootname, *TRANS1_9* would be followed by *TRAN1_10*. The initial character (*T* in this example) is required. Note that the truncation is done variable-wise, not analysis-wise.
- Variable labels are created automatically. (They are shown in the Procedure Information Table (the Notes table) and can also be displayed in the Data Editor window.)

OUTFILE Subcommand

The OUTFILE subcommand is used to write the discretized data and/or the transformed data (category indicators replaced with optimal quantifications) to an external data file. Excluded cases are represented by a dot (the sysmis symbol) on every saved variable.

DISCRDATA('filename') *Discretized data.*

TRDATA('filename') *Transformed variables.*

- Following the keyword, a filename enclosed by single quotation marks should be specified. The filenames should be different for the each of the keywords.
- A working data file, in principle, should not be replaced by this subcommand, and the asterisk (*) file specification is not supported. This strategy also prevents the OUTFILE interference with the SAVE subcommand.

CORRESPONDENCE

```
CORRESPONDENCE

/TABLE = {rowvar (min, max) BY colvar (min, max)}
         {ALL (# of rows, # of columns )        }

 [/SUPPLEMENTARY = [rowvar (valuelist)] [colvar (valuelist)]]

[/EQUAL = [rowvar (valuelist)... (valuelist)]
          [colvar (valuelist)... (valuelist)]]

[/MEASURE = {CHISQ**}]
            {EUCLID }

[/STANDARDIZE = {RMEAN   }]
                {CMEAN   }
                {RCMEAN**}
                {RSUM    }
                {CSUM    }

[/DIMENSION = {2**  }]
              {value}

 [/NORMALIZATION = {SYMMETRICAL**}]
                   {PRINCIPAL    }
                   {RPRINCIPAL   }
                   {CPRINCIPAL   }
                   {value        }

[/PRINT = [TABLE**] [RPROF] [CPROF] [RPOINTS**] [CPOINTS**]
          [RCONF] [CCONF] [PERMUTATION[(n)]] [DEFAULT] [NONE]]

[/PLOT = [NDIM({1**  ,2**  })]
               {value,value}
               {ALL  ,MAX  }
         [RPOINTS[(n)]] [CPOINTS[(n)] [TRROWS[(n)]]
         [TRCOLUMNS[(n)]] [BIPLOT**[(n)]] [NONE]]

[/OUTFILE = {SCORE(filename)                    }]
            {                  VARIANCE(filename)}
            {SCORE(filename)   VARIANCE(filename)}
```

**Default if subcommand or keyword is omitted.

Overview

CORRESPONDENCE displays the relationships between rows and columns of a two-way table graphically by a scatterplot matrix. It computes the row and column scores and statistics and produces plots based on the scores. Also, confidence statistics are computed.

Options

Number of dimensions. You can specify how many dimensions CORRESPONDENCE should compute.

Supplementary points. You can specify supplementary rows and columns.

Equality restrictions. You can restrict rows and columns to have equal scores.

Measure. You can specify the distance measure to be the chi-square of Euclidean.

Standardization. You can specify one of five different standardization methods.

Method of normalization. You can specify one of five different methods for normalizing the row and column scores.

Confidence statistics. You can request computation of confidence statistics (standard deviations and correlations) for row and column scores. For singular values, confidence statistics are always computed.

Data input. You can analyze individual casewise data, aggregated data, or table data.

Display output. You can control which statistics are displayed and plotted.

Writing matrices. You can write a matrix data file containing the row and column scores, and a matrix data file containing confidence statistics (variances and covariances) for the singular values, row scores, and column scores.

Basic Specification

- The basic specification is CORRESPONDENCE and the TABLE subcommand. By default, CORRESPONDENCE computes a two-dimensional solution and displays the correspondence table, the summary table, an overview of the row and column points, and a scatterplot matrix of biplots of the row and column scores for the first two dimensions.

Subcommand Order

- The TABLE subcommand must appear first.
- All other subcommands can appear in any order.

Syntax Rules

- Only one keyword can be specified on the MEASURE subcommand.
- Only one keyword can be specified on the STANDARDIZE subcommand.
- Only one keyword can be specified on the NORMALIZATION subcommand.
- Only one parameter can be specified on the DIMENSION subcommand.

Operations

- If a subcommand is specified more than once, only the last occurrence is executed.

Limitations

- The table input data and the aggregated input data cannot contain negative values. CORRESPONDENCE will treat such values as 0.
- Rows and columns that are specified as supplementary cannot be equalized.
- The maximum number of supplementary points for a variable is 200.
- The maximum number of equalities for a variable is 200.

Example

```
CORRESPONDENCE TABLE=MENTAL(1,4) BY SES(1,6)
/PRINT=RPOINTS CPOINTS
/PLOT=RPOINTS CPOINTS.
```

- Two variables, *MENTAL* and *SES*, are specified on the TABLE subcommand. *MENTAL* has values ranging from 1 to 4 and *SES* has values ranging from 1 to 6.
- The summary table and overview tables of the row and column points are displayed.
- Two scatterplot matrices are produced. The first one plots the first two dimensions of row scores and the second one plots the first two dimensions of column scores.

TABLE Subcommand

TABLE specifies the row and column variables along with their integer value ranges. The two variables are separated by the keyword BY.

- The TABLE subcommand is required.

Casewise Data

- Each variable is followed by an integer value range in parentheses. The value range consists of the variable's minimum value and its maximum value.
- Values outside of the specified range are not included in the analysis.
- Values do not have to be sequential. Empty categories yield a zero in the input table and do not affect the statistics for other categories.

Example

```
DATA LIST FREE/VAR1 VAR2.
BEGIN DATA
3 1
6 1
3 1
4 2
4 2
6 3
6 3
6 3
3 2
4 2
6 3
END DATA.
CORRESPONDENCE TABLE=VAR1(3,6) BY VAR2(1,3).
```

- DATA LIST defines two variables, *VAR1* and *VAR2*.
- *VAR1* has three levels, coded 3, 4, and 6. *VAR2* also has three levels, coded 1, 2, and 3.
- Since a range of (3,6) is specified for *VAR1*, CORRESPONDENCE defines four categories, coded 3, 4, 5, and 6. The empty category, 5, for which there is no data, receives system-missing values for all statistics and does not affect the analysis.

Table Data

- The cells of a table can be read and analyzed directly by using the keyword ALL after TABLE.
- The columns of the input table must be specified as variables on the DATA LIST command. Only columns are defined, not rows.
- ALL is followed by the number of rows in the table, a comma, and the number of columns in the table, all in parentheses.
- The row variable is named *ROW*, and the column variable is named *COLUMN*.
- The number of rows and columns specified can be smaller than the actual number of rows and columns if you want to analyze only a subset of the table.
- The variables (columns of the table) are treated as the column categories, and the cases (rows of the table) are treated as the row categories.
- Row categories can be assigned values (category codes) when you specify TABLE=ALL by the optional variable *ROWCAT_*. This variable must be defined as a numeric variable with unique values corresponding to the row categories. If *ROWCAT_* is not present, the row index numbers are used as row category values.

Example

```
DATA LIST /ROWCAT_ 1 COL1 3-4 COL2 6-7 COL3 9-10.
BEGIN DATA
1 50 19 26
2 16 40 34
3 12 35 65
4 11 20 58
END DATA.
VALUE LABELS ROWCAT_ 1 'ROW1' 2 'ROW2' 3 'ROW3' 4 'ROW4'.
CORRESPONDENCE TABLE=ALL(4,3).
```

- DATA LIST defines the row category naming variable *ROWCAT_* and the three columns of the table as the variables.

- The TABLE=ALL specification indicates that the data are the cells of a table. The (4,3) specification indicates that there are four rows and three columns.
- The column variable is named *COLUMN* with categories labeled *COL1*, *COL2*, and *COL3*.
- The row variable is named *ROW* with categories labeled *ROW1*, *ROW2*, *ROW3*, and *ROW4*.

DIMENSION Subcommand

DIMENSION specifies the number of dimensions you want CORRESPONDENCE to compute.

- If you do not specify the DIMENSION subcommand, CORRESPONDENCE computes two dimensions.
- DIMENSION is followed by a positive integer indicating the number of dimensions. If this parameter is omitted, a value of 2 is assumed.
- In general, you should choose as few dimensions as needed to explain most of the variation. The minimum number of dimensions that can be specified is 1. The maximum number of dimensions that can be specified equals the minimum of the number of active rows and the number of active columns, minus 1. An active row or column is a nonsupplementary row or column that is used in the analysis. For example, in a table where the number of rows is 5 (2 of which are supplementary) and the number of columns is 4, the number of active rows (3) is smaller than the number of active columns (4). Thus, the maximum number of dimensions that can be specified is $(5 - 2) - 1$, or 2. Rows and columns that are restricted to have equal scores count as 1 toward the number of active rows or columns. For example, in a table with five rows and four columns, where two columns are restricted to have equal scores, the number of active rows is 5 and the number of active columns is $(4 - 1)$, or 3. The maximum number of dimensions that can be specified is $(3 - 1)$, or 2. Empty rows and columns (rows or columns with no data, all zeros, or all missing data) are not counted toward the number of rows and columns.
- If more than the maximum allowed number of dimensions is specified, CORRESPONDENCE reduces the number of dimensions to the maximum.

SUPPLEMENTARY Subcommand

The SUPPLEMENTARY subcommand specifies the rows and columns that you want to treat as supplementary (also called passive or illustrative).

- For casewise data, the specification on SUPPLEMENTARY is a variable name, followed by a value list in parentheses. The values must be in the value range specified on the TABLE subcommand for the row or column variable.
- For table data, the specification on SUPPLEMENTARY is *ROW* and/or *COLUMN*, followed by a value list in parentheses. The values represent the row or column indices of the table input data.
- The maximum number of supplementary rows or columns is the number of active rows or columns minus 2.
- Supplementary rows and columns cannot be equalized.

Example

```
CORRESPONDENCE TABLE=MENTAL(1,8) BY SES(1,6)
/SUPPLEMENTARY MENTAL(3) SES(2,6).
```

- SUPPLEMENTARY specifies the third level of *MENTAL* and the second and sixth levels of *SES* to be supplementary.

Example

```
CORRESPONDENCE TABLE=ALL(8,6)
/SUPPLEMENTARY ROW(3) COLUMN(2,6).
```

- SUPPLEMENTARY specifies the third level of the row variable and the second and sixth levels of the column variable to be supplementary.

EQUAL Subcommand

The EQUAL subcommand specifies the rows or columns that you want to restrict to have equal scores.

- For casewise data, the specification on EQUAL is a variable name, followed by a list of at least two values in parentheses. The values must be in the value range specified on the TABLE subcommand for the row or column variable.
- For table data, the specification on EQUAL is *ROW* and/or *COLUMN*, followed by a value list in parentheses. The values represent the row or column indices of the table input data.
- Rows or columns that are restricted to have equal scores cannot be supplementary.
- The maximum number of equal rows or columns is the number of active rows or columns minus 1.

Example

```
CORRESPONDENCE TABLE=MENTAL(1,8) BY SES(1,6)
/EQUAL MENTAL(1,2) (6,7) SES(1,2,3).
```

- EQUAL specifies the first and second level of *MENTAL*, the sixth and seventh level of *MENTAL*, and the first, second, and third levels of *SES* to have equal scores.

MEASURE Subcommand

The MEASURE subcommand specifies the measure of distance between the row and column profiles.

- Only one keyword can be used in a given analysis.

The following keywords are available:

CHISQ *Chi-square distance*. This is the weighted distance, where the weight is the mass of the rows or columns. This is the default specification for MEASURE and is the necessary specification for standard correspondence analysis.

EUCLID *Euclidean distance.* The distance is the square root of the sum of squared differences between the values for two rows or columns.

STANDARDIZE Subcommand

When MEASURE=EUCLID, the STANDARDIZE subcommand specifies the method of standardization.

- Only one keyword can be used.
- If MEASURE is CHISQ, the standardization is automatically set to RCMEAN and corresponds to standard correspondence analysis.

The following keywords are available:

RMEAN *The row means are removed.*

CMEAN *The column means are removed.*

RCMEAN *Both the row and column means are removed. This is the default specification.*

RSUM *First the row totals are equalized and then the row means are removed.*

CSUM *First the column totals are equalized and then the column means are removed.*

NORMALIZATION Subcommand

The NORMALIZATION subcommand specifies one of five methods for normalizing the row and column scores. Only the scores and confidence statistics are affected; contributions and profiles are not changed.

The following keywords are available:

SYMMETRICAL *For each dimension, rows are the weighted average of columns divided by the matching singular value, and columns are the weighted average of rows divided by the matching singular value*. This is the default if the NORMALIZATION subcommand is not specified. Use this normalization method if you are primarily interested in differences or similarities between rows and columns.

PRINCIPAL *Distances between row points and column points are approximations of chi-square distances or of Euclidean distances (depending on MEASURE)*. The distances represent the distance between the row or column and its corresponding average row or column profile. Use this normalization method if you want to examine both differences between categories of the row variable and differences between categories of the column variable (but not differences between variables).

RPRINCIPAL *Distances between row points are approximations of chi-square distances or of Euclidean distances (depending on MEASURE)*. This method maximizes distances between row points. The row points are weighted averages of the column points. This is useful when you are primarily interested in differences or similarities between categories of the row variable.

CPRINCIPAL *Distances between column points are approximations of chi-square distances or of Euclidean distances (depending on MEASURE).* This method maximizes distances between column points. The column points are weighted averages of the row points. This is useful when you are primarily interested in differences or similarities between categories of the column variable.

The fifth method allows the user to specify any value in the range –1 to +1, inclusive. A value of 1 is equal to the RPRINCIPAL method, a value of 0 is equal to the SYMMETRICAL method, and a value of –1 is equal to the CPRINCIPAL method. By specifying a value between –1 and 1, the user can spread the inertia over both row and column scores to varying degrees. This method is useful for making tailor-made biplots.

PRINT Subcommand

Use PRINT to control which of several correspondence statistics are displayed. The summary table (singular values, inertia, proportion of inertia accounted for, cumulative proportion of inertia accounted for, and confidence statistics for the maximum number of dimensions) is always produced. If PRINT is not specified, the input table, the summary table, the overview of row points table, and the overview of column points table are displayed.

The following keywords are available:

TABLE *A crosstabulation of the input variables showing row and column marginals.*

RPROFILES *The row profiles.* PRINT=RPROFILES is analogous to the CELLS=ROW subcommand in CROSSTABS.

CPROFILES *The column profiles.* PRINT=CPROFILES is analogous to the CELLS=COLUMN subcommand in CROSSTABS.

RPOINTS *Overview of row points (mass, scores, inertia, contribution of the points to the inertia of the dimension, and the contribution of the dimensions to the inertia of the points).*

CPOINTS *Overview of column points (mass, scores, inertia, contribution of the points to the inertia of the dimension, and the contribution of the dimensions to the inertia of the points).*

RCONF *Confidence statistics (standard deviations and correlations) for the active row points.*

CCONF *Confidence statistics (standard deviations and correlations) for the active column points.*

PERMUTATION(n) *The original table permuted according to the scores of the rows and columns.* PERMUTATION can be followed by a number in parentheses indicating the maximum number of dimensions for which you want permuted tables. The default number of dimensions is 1.

NONE *No output other than the SUMMARY table.*

DEFAULT *TABLE, RPOINTS, CPOINTS, and the SUMMARY tables.* These statistics are displayed if you omit the PRINT subcommand.

PLOT Subcommand

Use PLOT to produce plots of the row scores, column scores, row and column scores, transformations of the row scores, and transformations of the column scores. If PLOT is not specified or is specified without keywords, a biplot is produced.

The following keywords are available:

TRROWS(n) *Line chart of transformations of the row category values into row scores.*

TRCOLUMNS(n) *Line chart of transformations of the column category values into column scores.*

RPOINTS(n) *Scatterplot matrix of row scores.*

CPOINTS(n) *Scatterplot matrix of column scores.*

BIPLOT(n) *Biplot matrix of the row and column scores.* This is the default plot. This plot is not available when NORMALIZATION=PRINCIPAL. From the Chart Editor, you can create a two-dimensional biplot of any pair of dimensions in the biplot matrix. You can also create a three-dimensional biplot of any three dimensions in the biplot matrix.

NONE *No plots.*

- All keywords can be followed by an integer value in parentheses to indicate how many characters of the value label are to be used in the plot. The value can range from 0 to 20. Spaces between words count as characters. A value of 0 corresponds to using the values instead of the value labels.
- If a label is missing for a value, the actual value is used. However, the length of the value is truncated in accordance with the length parameter. For example, a category coded as 100 with no value label appears as 10 if the length parameter is 2.
- TRROWS and TRCOLUMNS produce line charts. RPOINTS and CPOINTS produce scatterplot matrices. BIPLOT produces a biplot matrix. For line charts, the value labels are used to label the category axis. For scatterplot matrices and biplot matrices, the value labels are used to label the points in the plot.

In addition to the plot keywords, the following can be specified:

NDIM *Dimensions to be plotted.* NDIM is followed by a pair of values in parentheses. If NDIM is not specified, NDIM(1,2) is assumed.

- The first value must be any integer from 1 to the number of dimensions minus 1.

- The second value can be any integer from 2 to the number of dimensions. The second value must exceed the first. Alternatively, the keyword MAX can be used instead of a value to indicate the highest dimension of the solution.
- For TRROWS and TRCOLUMNS, the first and second values indicate the range of dimensions for which the plots are created.
- For RPOINTS, CPOINTS, and BIPLOT, the first and second values indicate the range of dimensions included in the scatterplot matrix or biplot matrix.

Example

```
CORRESPONDENCE TABLE=MENTAL(1,4) BY SES(1,6)
/PLOT NDIM(1,3) BIPLOT(5).
```

- BIPLOT and NDIM(1,3) request a biplot matrix of the first three dimensions.
- The 5 following BIPLOT indicates that only the first five characters of each label are to be shown in the biplot matrix.

Example

```
CORRESPONDENCE TABLE=MENTAL(1,4) BY SES(1,6)
/DIMENSION = 3
/PLOT NDIM(1,MAX) TRROWS.
```

- Three transformation plots of row categories into row points are produced, one for each dimension from 1 to the highest dimension of the analysis (in this case, 3).

OUTFILE Subcommand

Use OUTFILE to write row and column scores and/or confidence statistics (variances and covariances) for the singular values and row and column scores to matrix data files.

OUTFILE must be followed by one or both of the following keywords:

SCORE (filename) *Write row and column scores to a matrix data file.*

VARIANCE (filename) *Write variances and covariances to a matrix data file.*

- You must specify the name of an external file.
- If you specify both SCORE and VARIANCE on the same OUTFILE subcommand, you must specify two different filenames.
- For VARIANCE, supplementary and equality constrained rows and columns are not produced in the matrix file.

The variables in the SCORE matrix data file and their values are:

ROWTYPE_ *String variable containing the value ROW for all of the rows and COLUMN for all of the columns.*

LEVEL_ *String variable containing the values (or value labels, if present) of each original variable.*

VARNAME_ *String variable containing the original variable names.*

DIM1...DIMn *Numerical variables containing the row and column scores for each dimension*. Each variable is labeled *DIMn*, where *n* represents the dimension number.

The variables in the VARIANCE matrix data file and their values are:

ROWTYPE_ *String variable containing the value COV for all of the cases in the file*.

SCORE_ *String variable containing the value SINGULAR, the row variable's name (or label), and the column variable's name (or label).*

LEVEL_ *String variable containing the row variable's values (or labels), the column variable's values (or labels), and a blank value for score_ = SINGULAR*.

VARNAME_ *String variable containing the dimension number*.

DIM1...DIMn *Numerical variables containing the variances and covariances for each dimension*. Each variable is named *DIMn*, where *n* represents the dimension number.

See the *SPSS Syntax Reference Guide* for more information on matrix data files.

Analyzing Aggregated Data

To analyze aggregated data, such as data from a crosstabulation where cell counts are available but the original raw data are not, you can use the WEIGHT command before CORRESPONDENCE.

Example

To analyze a 3×3 table such as the one shown in Table 1, you could use these commands:

```
DATA LIST FREE/ BIRTHORD ANXIETY COUNT.
BEGIN DATA
1 1 48
1 2 27
1 3 22
2 1 33
2 2 20
2 3 39
3 1 29
3 2 42
3 3 47
END DATA.
WEIGHT BY COUNT.
CORRESPONDENCE TABLE=BIRTHORD (1,3) BY ANXIETY (1,3).
```

- The WEIGHT command weights each case by the value of *COUNT*, as if there are 48 subjects with *BIRTHORD*=1 and *ANXIETY*=1, 27 subjects with *BIRTHORD*=1 and *ANXIETY*=2, and so on.
- CORRESPONDENCE can then be used to analyze the data.
- If any of the table cell values equals 0, the WEIGHT command issues a warning, but the CORRESPONDENCE analysis is done correctly.

- The table cell values (the WEIGHT values) cannot be negative.

Table 1 3 x 3 table

		Anxiety		
		High	**Med**	**Low**
Birth order	**First**	48	27	22
	Second	33	20	39
	Other	29	42	47

HOMALS

```
HOMALS  VARIABLES=varlist(max)

[/ANALYSIS=varlist]

[/NOBSERVATIONS=value]

[/DIMENSION={2**  }]
            {value}

[/MAXITER={100**}]
          {value}

[/CONVERGENCE={.00001**}]
              {value   }

[/PRINT=[DEFAULT**] [FREQ**] [EIGEN**] [DISCRIM**]
        [QUANT**] [OBJECT] [HISTORY] [ALL] [NONE]]

[/PLOT=[NDIM=({1, 2         }**)]
              {value, value}
              {ALL, MAX     }
       [QUANT**[(varlist)][(n)]] [OBJECT**[(varlist)][(n)]]
       [DEFAULT**[(n)]] [DISCRIM[(n)]] [ALL[(n)]] [NONE]]

[/SAVE=[rootname] [(value)]]

[/MATRIX=OUT({*   })]
            {file}
```

**Default if subcommand or keyword is omitted.

Overview

HOMALS (*hom*ogeneity analysis by means of *a*lternating *l*east *s*quares) estimates category quantifications, object scores, and other associated statistics that separate categories (levels) of nominal variables as much as possible and divide cases into homogeneous subgroups.

Options

Data and variable selection. You can use a subset of the variables in the analysis and restrict the analysis to the first *n* observations.

Number of dimensions. You can specify how many dimensions HOMALS should compute.

Iterations and convergence. You can specify the maximum number of iterations and the value of a convergence criterion.

Display output. The output can include all available statistics, just the default frequencies, eigenvalues, discrimination measures and category quantifications, or just the specific statistics you request. You can also control which statistics are plotted and specify the number of characters used in plot labels.

Saving scores. You can save object scores in the working data file.

Writing matrices. You can write a matrix data file containing category quantifications for use in further analyses.

Basic Specification

- The basic specification is HOMALS and the VARIABLES subcommand. By default, HOMALS analyzes all of the variables listed for all cases and computes two solutions. Frequencies, eigenvalues, discrimination measures, and category quantifications are displayed and category quantifications and object scores are plotted.

Subcommand Order

- Subcommands can appear in any order.

Syntax Rules

- If ANALYSIS is specified more than once, HOMALS is not executed. For all other subcommands, if a subcommand is specified more than once, only the last occurrence is executed.

Operations

- HOMALS treats every value in the range 1 to the maximum value specified on VARIABLES as a valid category. If the data are not sequential, the empty categories (categories with no valid data) are assigned zeros for all statistics. You may want to use RECODE or AUTORECODE before HOMALS to get rid of these empty categories and avoid the unnecessary output (see the *SPSS Syntax Reference Guide* for more information on AUTORECODE and RECODE).

Limitations

- String variables are not allowed; use AUTORECODE to recode string variables into numeric variables.
- The data (category values) must be positive integers. Zeros and negative values are treated as system-missing, which means that they are excluded from the analysis. Fractional values are truncated after the decimal and are included in the analysis. If one of the levels of a variable has been coded 0 or a negative value and you want to treat it as a valid category, use the AUTORECODE or RECODE command to recode the values of that variable.
- HOMALS ignores user-missing value specifications. Positive user-missing values less than the maximum value specified on the VARIABLES subcommand are treated as valid category values and are included in the analysis. If you do not want the category included, use COMPUTE or RECODE to change the value to something outside of the valid range. Values outside of the range (less than 1 or greater than the maximum value) are treated as system-missing and are excluded from the analysis.

Example

```
HOMALS  VARIABLES=ACOLA(2) BCOLA(2) CCOLA(2) DCOLA(2)
  /PRINT=FREQ EIGEN QUANT OBJECT.
```

- The four variables are analyzed using all available observations. Each variable has two categories, 1 and 2.
- The PRINT subcommand lists the frequencies, eigenvalues, category quantifications, and object scores.
- By default, plots of the category quantifications and the object scores are produced.

VARIABLES Subcommand

VARIABLES specifies the variables that will be used in the analysis.

- The VARIABLES subcommand is required. The actual word VARIABLES can be omitted.
- After each variable or variable list, specify in parentheses the maximum number of categories (levels) of the variables.
- The number specified in parentheses indicates the number of categories *and* the maximum category value. For example, *VAR1(3)* indicates that *VAR1* has three categories coded 1, 2, and 3. However, if a variable is not coded with consecutive integers, the number of categories used in the analysis will differ from the number of observed categories. For example, if a three-category variable is coded {2, 4, 6}, the maximum category value is 6. The analysis treats the variable as having six categories, three of which (categories 1, 3, and 5) are not observed and receive quantifications of 0.
- To avoid unnecessary output, use the AUTORECODE or RECODE command before HOMALS to recode a variable that does not have sequential values (see the *SPSS Syntax Reference Guide* for more information on AUTORECODE and RECODE).

Example

```
DATA LIST FREE/V1 V2 V3.
BEGIN DATA
3 1 1
6 1 1
3 1 3
3 2 2
3 2 2
6 2 2
6 1 3
6 2 2
3 2 2
6 2 1
END DATA.
AUTORECODE V1 /INTO NEWVAR1.
HOMALS VARIABLES=NEWVAR1 V2(2) V3(3).
```

- DATA LIST defines three variables, *V1*, *V2*, and *V3*.
- *V1* has two levels, coded 3 and 6, *V2* has two levels, coded 1 and 2, and *V3* has three levels, coded 1, 2, and 3.
- The AUTORECODE command creates *NEWVAR1* containing recoded values of *V1*. Values of 3 are recoded to 1; values of 6 are recoded to 2.

- The maximum category value for both *NEWVAR1* and *V2* is 2. A maximum value of 3 is specified for *V3*.

ANALYSIS Subcommand

ANALYSIS limits the analysis to a specific subset of the variables named on the VARIABLES subcommand.

- If ANALYSIS is not specified, all variables listed on the VARIABLES subcommand are used.
- ANALYSIS is followed by a variable list. The variables on the list must be specified on the VARIABLES subcommand.
- Variables listed on the VARIABLES subcommand but not on the ANALYSIS subcommand can still be used to label object scores on the PLOT subcommand.

Example

```
HOMALS VARIABLES=ACOLA(2) BCOLA(2) CCOLA(2) DCOLA(2)
  /ANALYSIS=ACOLA BCOLA
  /PRINT=OBJECT QUANT
  /PLOT=OBJECT(CCOLA).
```

- The VARIABLES subcommand specifies four variables.
- The ANALYSIS subcommand limits analysis to the first two variables. The PRINT subcommand lists the object scores and category quantifications from this analysis.
- The plot of the object scores is labeled with variable *CCOLA*, even though this variable is not included in the computations.

NOBSERVATIONS Subcommand

NOBSERVATIONS specifies how many cases are used in the analysis.

- If NOBSERVATIONS is not specified, all available observations in the working data file are used.
- NOBSERVATIONS is followed by an integer indicating that the first *n* cases are to be used.

DIMENSION Subcommand

DIMENSION specifies the number of dimensions you want HOMALS to compute.

- If you do not specify the DIMENSION subcommand, HOMALS computes two dimensions.
- The specification on DIMENSION is a positive integer indicating the number of dimensions.
- The minimum number of dimensions is 1.
- The maximum number of dimensions is equal to the smaller of the two values below:

  The total number of valid variable categories (levels) minus the number of variables without missing values.

  The number of observations minus 1.

MAXITER Subcommand

MAXITER specifies the maximum number of iterations HOMALS can go through in its computations.

- If MAXITER is not specified, HOMALS will iterate up to 100 times.
- The specification on MAXITER is a positive integer indicating the maximum number of iterations.

CONVERGENCE Subcommand

CONVERGENCE specifies a convergence criterion value. HOMALS stops iterating if the difference in total fit between the last two iterations is less than the CONVERGENCE value.

- If CONVERGENCE is not specified, the default value is 0.00001.
- The specification on CONVERGENCE is a positive value.

PRINT Subcommand

PRINT controls which statistics are included in your display output. The default display includes the frequencies, eigenvalues, discrimination measures, and category quantifications.

The following keywords are available:

FREQ *Marginal frequencies for the variables in the analysis.*

HISTORY *History of the iterations.*

EIGEN *Eigenvalues.*

DISCRIM *Discrimination measures for the variables in the analysis.*

OBJECT *Object scores.*

QUANT *Category quantifications for the variables in the analysis.*

DEFAULT *FREQ, EIGEN, DISCRIM, and QUANT.* These statistics are also displayed when you omit the PRINT subcommand.

ALL *All available statistics.*

NONE *No statistics.*

PLOT Subcommand

PLOT can be used to produce plots of category quantifications, object scores, and discrimination measures.

- If PLOT is not specified, plots of the object scores and of the quantifications are produced.
- No plots are produced for a one-dimensional solution.

The following keywords can be specified on PLOT:

DISCRIM *Plots of the discrimination measures.*

OBJECT *Plots of the object scores.*

QUANT *Plots of the category quantifications.*

DEFAULT *QUANT and OBJECT.*

ALL *All available plots.*

NONE *No plots.*

- Keywords OBJECT and QUANT can each be followed by a variable list in parentheses to indicate that plots should be labeled with those variables. For QUANT, the labeling variables must be specified on both the VARIABLES and ANALYSIS subcommands. For OBJECT, the variables must be specified on the VARIABLES subcommand but need not appear on the ANALYSIS subcommand. This means that variables not used in the computations can be used to label OBJECT plots. If the variable list is omitted, the default object and quantification plots are produced.
- Object score plots labeled with variables that appear on the ANALYSIS subcommand use category labels corresponding to all categories within the defined range. Objects in a category that is outside the defined range are labeled with the label corresponding to the category immediately following the defined maximum category value.
- Object score plots labeled with variables not included on the ANALYSIS subcommand use all category labels, regardless of whether or not the category value is inside the defined range.
- All keywords except NONE can be followed by an integer value in parentheses to indicate how many characters of the variable or value label are to be used on the plot. (If you specify a variable list after OBJECT or QUANT, specify the value in parentheses after the list.) The value can range from 1 to 20; the default is to use 12 characters. Spaces between words count as characters.
- DISCRIM plots use variable labels; all other plots use value labels.
- If a variable label is not supplied, the variable name is used for that variable. If a value label is not supplied, the actual value is used.
- Variable and value labels should be unique.
- When points overlap, the points involved are described in a summary following the plot.

Example

```
HOMALS VARIABLES COLA1 (4) COLA2 (4) COLA3 (4) COLA4 (2)
 /ANALYSIS COLA1 COLA2 COLA3 COLA4
 /PLOT OBJECT(COLA4).
```

- Four variables are included in the analysis.
- OBJECT requests a plot of the object scores labeled with the values of *COLA4*. Any object whose *COLA4* value is not 1 or 2, is labeled 3 (or the value label for category 3, if supplied).

Example

```
HOMALS VARIABLES COLA1 (4) COLA2 (4) COLA3 (4) COLA4 (2)
 /ANALYSIS COLA1 COLA2 COLA3
 /PLOT OBJECT(COLA4).
```

- Three variables are included in the analysis.
- OBJECT requests a plot of the object scores labeled with the values of *COLA4,* a variable not included in the analysis. Objects are labeled using all values of *COLA4.*

In addition to the plot keywords, the following can be specified:

NDIM *Dimension pairs to be plotted.* NDIM is followed by a pair of values in parentheses. If NDIM is not specified, plots are produced for dimension 1 versus dimension 2.

- The first value indicates the dimension that is plotted against all higher dimensions. This value can be any integer from 1 to the number of dimensions minus 1.
- The second value indicates the highest dimension to be used in plotting the dimension pairs. This value can be any integer from 2 to the number of dimensions.
- Keyword ALL can be used instead of the first value to indicate that all dimensions are paired with higher dimensions.
- Keyword MAX can be used instead of the second value to indicate that plots should be produced up to and including the highest dimension fit by the procedure.

Example

```
HOMALS COLA1 COLA2 COLA3 COLA4 (4)
 /PLOT NDIM(1,3) QUANT(5).
```

- The NDIM(1,3) specification indicates that plots should be produced for two dimension pairs—dimension 1 versus dimension 2 and dimension 1 versus dimension 3.
- QUANT requests plots of the category quantifications. The (5) specification indicates that the first five characters of the value labels are to be used on the plots.

Example

```
HOMALS COLA1 COLA2 COLA3 COLA4 (4)
 /PLOT NDIM(ALL,3) QUANT(5).
```

- This plot is the same as above except for the ALL specification following NDIM. This indicates that all possible pairs up to the second value should be plotted, so QUANT plots will be produced for dimension 1 versus dimension 2, dimension 2 versus dimension 3, and dimension 1 versus dimension 3.

SAVE Subcommand

SAVE lets you add variables containing the object scores computed by HOMALS to the working data file.

- If SAVE is not specified, object scores are not added to the working data file.

- A variable rootname can be specified on the SAVE subcommand to which HOMALS adds the number of the dimension. Only one rootname can be specified and it can contain up to six characters.
- If a rootname is not specified, unique variable names are automatically generated. The variable names are *HOMn_m*, where *n* is a dimension number and *m* is a set number. If three dimensions are saved, the first set of names is *HOM1_1*, *HOM2_1*, and *HOM3_1*. If another HOMALS is then run, the variable names for the second set are *HOM1_2*, *HOM2_2*, *HOM3_2*, and so on.
- Following the rootname, the number of dimensions for which you want to save object scores can be specified in parentheses. The number cannot exceed the value on the DIMENSION subcommand.
- If the number of dimensions is not specified, the SAVE subcommand saves object scores for all dimensions.
- If you replace the working data file by specifying an asterisk (*) on a MATRIX subcommand, the SAVE subcommand is not executed.

Example

```
HOMALS CAR1 CAR2 CAR3 CAR4(5)
 /DIMENSION=3
 /SAVE=DIM(2).
```

- Four variables, each with five categories, are analyzed.
- The DIMENSION subcommand specifies that results for three dimensions will be computed.
- SAVE adds the object scores from the first two dimensions to the working data file. The names of these new variables will be *DIM00001* and *DIM00002*, respectively.

MATRIX Subcommand

The MATRIX subcommand is used to write category quantifications to a matrix data file.

- The specification on MATRIX is keyword OUT and a file enclosed in parentheses.
- You can specify the file with either an asterisk (*) to indicate that the working data file is to be replaced or with the name of an external file.
- The matrix data file has one case for each value of each original variable.

The variables of the matrix data file and their values are:

ROWTYPE_ *String variable containing value QUANT for all cases.*

LEVEL *String variable LEVEL containing the values (or value labels if present) of each original variable.*

VARNAME_ *String variable containing the original variable names.*

DIM1...DIMn *Numeric variable containing the category quantifications for each dimension.* Each variable is labeled *DIMn*, where *n* represents the dimension number.

See the *SPSS Syntax Reference Guide* for more information on matrix data files.

OVERALS

```
OVERALS VARIABLES=varlist (max)

/ANALYSIS=varlist[({ORDI**})]
                  {SNOM  }
                  {MNOM  }
                  {NUME  }

/SETS= n (# of vars in set 1, ..., # of vars in set n)

[/NOBSERVATIONS=value]

[/DIMENSION={2**  }]
            {value}

[/INITIAL={NUMERICAL**}]
          {RANDOM     }

[/MAXITER={100**}]
          {value}

[/CONVERGENCE={.00001**}]
              {value   }

[/PRINT=[DEFAULT] [FREQ**] [QUANT] [CENTROID**]
        [HISTORY] [WEIGHTS**]
        [OBJECT] [FIT] [NONE]]

[/PLOT=[NDIM=({1    ,2    }**)]
               {value,value}
               {ALL  ,MAX  }
       [DEFAULT[(n)]] [OBJECT**[(varlist)][(n)]]
       [QUANT[(varlist)][(n)]] [LOADINGS**[(n)]]
       [TRANS[(varlist)]Æ[CENTROID[(varlist)][(n)]]
       [NONE]]

[/SAVE=[rootname][(value)]]

[/MATRIX=OUT({*   })]
             {file}
```

**Default if subcommand or keyword is omitted.

Overview

OVERALS performs nonlinear canonical correlation analysis on two or more sets of variables. Variables can have different optimal scaling levels, and no assumptions are made about the distribution of the variables or the linearity of the relationships.

Options

Optimal scaling levels. You can specify the level of optimal scaling at which you want to analyze each variable.

Number of dimensions. You can specify how many dimensions OVERALS should compute.

Iterations and convergence. You can specify the maximum number of iterations and the value of a convergence criterion.

Display output. The output can include all available statistics, just the default statistics, or just the specific statistics you request. You can also control whether some of these statistics are plotted.

Saving scores. You can save object scores in the working data file.

Writing matrices. You can write a matrix data file containing quantification scores, centroids, weights, and loadings for use in further analyses.

Basic Specification

- The basic specification is command OVERALS, the VARIABLES subcommand, the ANALYSIS subcommand, and the SETS subcommand. By default, OVERALS estimates a two-dimensional solution and displays a table listing optimal scaling levels of each variable by set, eigenvalues and loss values by set, marginal frequencies, centroids and weights for all variables, and plots of the object scores and component loadings.

Subcommand Order

- The VARIABLES subcommand, ANALYSIS subcommand, and SETS subcommand must appear in that order before all other subcommands.
- Other subcommands can appear in any order.

Operations

- If the ANALYSIS subcommand is specified more than once, OVERALS is not executed. For all other subcommands, if a subcommand is specified more than once, only the last occurrence is executed.
- OVERALS treats every value in the range 1 to the maximum value specified on VARIABLES as a valid category. To avoid unnecessary output, use the AUTORECODE or RECODE command to recode a categorical variable with nonsequential values or with a large number of categories. For variables treated as numeric, recoding is not recommended because the characteristic of equal intervals in the data will not be maintained (see the *SPSS Syntax Reference Guide* for more information on AUTORECODE and RECODE).

Limitations

- String variables are not allowed; use AUTORECODE to recode nominal string variables.
- The data must be positive integers. Zeros and negative values are treated as system-missing, which means that they are excluded from the analysis. Fractional values are truncated after the decimal and are included in the analysis. If one of the levels of a categorical variable has been coded 0 or some negative value and you want to treat it

as a valid category, use the AUTORECODE or RECODE command to recode the values of that variable.

- OVERALS ignores user-missing value specifications. Positive user-missing values less than the maximum value specified on the VARIABLES subcommand are treated as valid category values and are included in the analysis. If you do not want the category included, use COMPUTE or RECODE to change the value to something outside of the valid range. Values outside of the range (less than 1 or greater than the maximum value) are treated as system-missing and are excluded from the analysis.
- If one variable in a set has missing data, all variables in that set are missing for that object (case).
- Each set must have at least three valid (nonmissing, non-empty) cases.

Example

```
OVERALS VARIABLES=PRETEST1 PRETEST2 POSTEST1 POSTEST2(20)
                  SES(5) SCHOOL(3)
 /ANALYSIS=PRETEST1 TO POSTEST2 (NUME) SES (ORDI) SCHOOL (SNOM)
 /SETS=3(2,2,2)
 /PRINT=OBJECT FIT
 /PLOT=QUANT(PRETEST1 TO SCHOOL).
```

- VARIABLES defines the variables and their maximum values.
- ANALYSIS specifies that all of the variables from *PRETEST1* to *POSTEST2* are to be analyzed at the numeric level of optimal scaling, *SES* at the ordinal level, and *SCHOOL* as a single nominal. These are all of the variables that will be used in the analysis.
- SETS specifies that there are three sets of variables to be analyzed and two variables in each set.
- PRINT lists the object and fit scores.
- PLOT plots the single- and multiple-category coordinates of all of the variables in the analysis.

VARIABLES Subcommand

VARIABLES specifies all of the variables in the current OVERALS procedure.

- The VARIABLES subcommand is required and precedes all other subcommands. The actual word VARIABLES can be omitted.
- Each variable or variable list is followed by the maximum value in parentheses.

ANALYSIS Subcommand

ANALYSIS specifies the variables to be used in the analysis and the optimal scaling level at which each variable is to be analyzed.

- The ANALYSIS subcommand is required and follows the VARIABLES subcommand.
- The specification on ANALYSIS is a variable list and an optional keyword in parentheses indicating the level of optimal scaling.

- The variables on ANALYSIS must also be specified on the VARIABLES subcommand.
- Only active variables are listed on the ANALYSIS subcommand. **Active variables** are those used in the computation of the solution. **Passive variables**, those listed on the VARIABLES subcommand but not on the ANALYSIS subcommand, are ignored in the OVERALS solution. Object score plots can still be labeled by passive variables.

The following keywords can be specified to indicate the optimal scaling level:

MNOM *Multiple nominal.* The quantifications can be different for each dimension. When all variables are multiple nominal and there is only one variable in each set, OVERALS gives the same results as HOMALS.

SNOM *Single nominal*. OVERALS gives only one quantification for each category. Objects in the same category (cases with the same value on a variable) obtain the same quantification. When all variables are SNOM, ORDI, or NUME, and there is only one variable per set, OVERALS will give the same results as PRINCALS.

ORDI *Ordinal*. This is the default for variables listed without optimal scaling levels. The order of the categories of the observed variable is preserved in the quantified variable.

NUME *Numerical*. Interval or ratio scaling level. OVERALS assumes that the observed variable already has numerical values for its categories. When all variables are quantified at the numerical level and there is only one variable per set, the OVERALS analysis is analogous to classical principal components analysis.

These keywords can apply to a variable list as well as to a single variable. Thus, the default ORDI is not applied to a variable without a keyword if a subsequent variable on the list has a keyword.

SETS Subcommand

SETS specifies how many sets of variables there are and how many variables are in each set.

- SETS is required and must follow the ANALYSIS subcommand.
- SETS is followed by an integer to indicate the number of variable sets. Following this integer is a list of values in parentheses indicating the number of variables in each set.
- There must be at least two sets.
- The sum of the values in parentheses must equal the number of variables specified on the ANALYSIS subcommand. The variables in each set are read consecutively from the ANALYSIS subcommand.

For example,

```
/SETS=2(2,3)
```

indicates that there are two sets. The first two variables named on ANALYSIS are the first set, and the last three variables named on ANALYSIS are the second set.

NOBSERVATIONS Subcommand

NOBSERVATIONS specifies how many cases are used in the analysis.

- If NOBSERVATIONS is not specified, all available observations in the working data file are used.
- NOBSERVATIONS is followed by an integer, indicating that the first *n* cases are to be used.

DIMENSION Subcommand

DIMENSION specifies the number of dimensions you want OVERALS to compute.

- If you do not specify the DIMENSION subcommand, OVERALS computes two dimensions.
- DIMENSION is followed by an integer indicating the number of dimensions.
- If all the variables are SNOM (single nominal), ORDI (ordinal), or NUME (numerical), the maximum number of dimensions you can specify is the total number of variables on the ANALYSIS subcommand.
- If some or all of the variables are MNOM (multiple nominal), the maximum number of dimensions you can specify is the number of MNOM variable levels (categories) plus the number of nonMNOM variables, minus the number of MNOM variables.
- The maximum number of dimensions must be less than the number of observations minus 1.
- If the number of sets is two and all variables are SNOM, ORDI, or NUME, the number of dimensions should not be more than the number of variables in the smaller set.
- If the specified value is too large, OVERALS tries to adjust the number of dimensions to the allowable maximum. It might not be able to adjust if there are MNOM variables with missing data.

INITIAL Subcommand

The INITIAL subcommand specifies the method used to compute the initial configuration.

- The specification on INITIAL is keyword NUMERICAL or RANDOM. If the INITIAL subcommand is not specified, NUMERICAL is the default.

NUMERICAL *Treat all variables except multiple nominal as numerical*. This is usually best to use when there are no SNOM variables.

RANDOM *Compute a random initial configuration*. This should be used only when some or all of the variables are SNOM.

MAXITER Subcommand

MAXITER specifies the maximum number of iterations OVERALS can go through in its computations.

- If MAXITER is not specified, OVERALS will iterate up to 100 times.
- The specification on MAXITER is an integer indicating the maximum number of iterations.

CONVERGENCE Subcommand

CONVERGENCE specifies a convergence criterion value. OVERALS stops iterating if the difference in fit between the last two iterations is less than the CONVERGENCE value.

- The default CONVERGENCE value is 0.00001.
- The specification on CONVERGENCE is any value greater than 0.000001. (Values less than this might seriously affect performance.)

PRINT Subcommand

PRINT controls which statistics are included in your display output. The default output includes a table listing optimal scaling levels of each variable by set, eigenvalues and loss values by set by dimension, and the output produced by keywords FREQ, CENTROID, and WEIGHTS.

The following keywords are available:

FREQ *Marginal frequencies for the variables in the analysis.*

HISTORY *History of the iterations.*

FIT *Multiple fit, single fit, and single loss per variable.*

CENTROID *Category quantification scores, the projected centroids, and the centroids.*

OBJECT *Object scores.*

QUANT *Category quantifications and the single and multiple coordinates.*

WEIGHTS *Weights and component loadings.*

DEFAULT *FREQ, CENTROID, and WEIGHTS.*

NONE *Summary loss statistics.*

PLOT Subcommand

PLOT can be used to produce plots of transformations, object scores, coordinates, centroids, and component loadings.

- If PLOT is not specified, plots of the object scores and component loadings are produced.

The following keywords can be specified on PLOT:

LOADINGS *Plot of the component loadings.*

OBJECT *Plot of the object scores.*

TRANS *Plot of category quantifications.*

QUANT *Plot of all category coordinates.*

CENTROID *Plot of all category centroids.*

DEFAULT *OBJECT and LOADINGS.*

NONE *No plots.*

- Keywords OBJECT, QUANT, and CENTROID can each be followed by a variable list in parentheses to indicate that plots should be labeled with these variables. For QUANT and CENTROID, the variables must be specified on both the VARIABLES and the ANALYSIS subcommands. For OBJECT, the variables must be specified on VARIABLES but need not appear on ANALYSIS. This means that variables not used in the computations can still be used to label OBJECT plots. If the variable list is omitted, the default plots are produced.
- Object score plots use category labels corresponding to all categories within the defined range. Objects in a category that is outside the defined range are labeled with the label corresponding to the category immediately following the defined maximum category.
- If TRANS is followed by a variable list, only plots for those variables are produced. If a variable list is not specified, plots are produced for each variable.
- All of the keywords except NONE can be followed by an integer in parentheses to indicate how many characters of the variable or value label are to be used on the plot. (If you specified a variable list after OBJECT, CENTROID, TRANS, or QUANT, you can specify the value in parentheses after the list.) The value can range from 1 to 20. If the value is omitted, 12 characters are used. Spaces between words count as characters.
- If a variable label is missing, the variable name is used for that variable. If a value label is missing, the actual value is used.
- You should make sure that your variable and value labels are unique by at least one letter in order to distinguish them on the plots.
- When points overlap, the points involved are described in a summary following the plot.

In addition to the plot keywords, the following can be specified:

NDIM *Dimension pairs to be plotted.* NDIM is followed by a pair of values in parentheses. If NDIM is not specified, plots are produced for dimension 1 versus dimension 2.

- The first value indicates the dimension that is plotted against all higher dimensions. This value can be any integer from 1 to the number of dimensions minus 1.
- The second value indicates the highest dimension to be used in plotting the dimension pairs. This value can be any integer from 2 to the number of dimensions.
- Keyword ALL can be used instead of the first value to indicate that all dimensions are paired with higher dimensions.
- Keyword MAX can be used instead of the second value to indicate that plots should be produced up to and including the highest dimension fit by the procedure.

Example

```
OVERALS COLA1 COLA2 JUICE1 JUICE2 (4)
 /ANALYSIS=COLA1 COLA2 JUICE1 JUICE2 (SNOM)
 /SETS=2(2,2)
 /PLOT NDIM(1,3) QUANT(5).
```

- The NDIM(1,3) specification indicates that plots should be produced for two dimension pairs—dimension 1 versus dimension 2 and dimension 1 versus dimension 3.
- QUANT requests plots of the category quantifications. The (5) specification indicates that the first five characters of the value labels are to be used on the plots.

Example

```
OVERALS COLA1 COLA2 JUICE1 JUICE2 (4)
 /ANALYSIS=COLA1 COLA2 JUICE1 JUICE2 (SNOM)
 /SETS=2(2,2)
 /PLOT NDIM(ALL,3) QUANT(5).
```

- This plot is the same as above except for the ALL specification following NDIM. This indicates that all possible pairs up to the second value should be plotted, so QUANT plots will be produced for dimension 1 versus dimension 2, dimension 2 versus dimension 3, and dimension 1 versus dimension 3.

SAVE Subcommand

SAVE lets you add variables containing the object scores computed by OVERALS to the working data file.

- If SAVE is not specified, object scores are not added to the working data file.
- A variable rootname can be specified on the SAVE subcommand to which OVERALS adds the number of the dimension. Only one rootname can be specified, and it can contain up to six characters.
- If a rootname is not specified, unique variable names are automatically generated. The variable names are *OVEn_m*, where *n* is a dimension number and *m* is a set number. If three dimensions are saved, the first set of names are *OVE1_1*, *OVE2_1*, and *OVE3_1*. If another OVERALS is then run, the variable names for the second set are *OVE1_2*, *OVE2_2*, *OVE3_2*, and so on.
- Following the name, the number of dimensions for which you want object scores saved can be listed in parentheses. The number cannot exceed the value of the DIMENSION subcommand.
- The prefix should be unique for each OVERALS command in the same session. If it is not, OVERALS replaces the prefix with *DIM*, *OBJ*, or *OBSAVE*. If all of these already exist, SAVE is not executed.
- If the number of dimensions is not specified, the SAVE subcommand saves object scores for all dimensions.
- If you replace the working data file by specifying an asterisk (*) on a MATRIX subcommand, the SAVE subcommand is not executed.

Example

```
OVERALS CAR1 CAR2 CAR3(5) PRICE (10)
 /SET=2(3,1)
 /ANALYSIS=CAR1 TO CAR3(SNOM) PRICE(NUME)
 /DIMENSIONS=3
 /SAVE=DIM(2).
```

- Three single nominal variables, *CAR1*, *CAR2*, and *CAR3*, each with five categories, and one numeric level variable, with ten categories, are analyzed.
- The DIMENSIONS subcommand requests results for three dimensions.
- SAVE adds the object scores from the first two dimensions to the working data file. The names of these new variables will be *DIM00001* and *DIM00002*, respectively.

MATRIX Subcommand

The MATRIX subcommand is used to write category quantifications, coordinates, centroids, weights, and component loadings to a matrix data file.

- The specification on MATRIX is keyword OUT and a file enclosed in parentheses.
- You can specify the file with either an asterisk (*) to indicate that the working data file is to be replaced or with the name of an external file.
- All values are written to the same file.
- The matrix data file has one case for each value of each original variable.

The variables of the matrix data file and their values are:

ROWTYPE_ *String variable containing value QUANT for the category quantifications, SCOOR_ for the single-category coordinates, MCOOR_ for multiple-category coordinates, CENTRO_ for centroids, PCENTRO_ for projected centroids, WEIGHT_ for weights, and LOADING_ for the component scores.*

LEVEL *String variable containing the values (or value labels if present) of each original variable for category quantifications.* For cases with *ROWTYPE_=LOADING_* or *WEIGHT_*, the value of *LEVEL* is blank.

VARNAME_ *String variable containing the original variable names.*

VARTYPE_ *String variable containing values MULTIPLE, SINGLE N, ORDINAL, or NUMERICAL, depending on the level of optimal scaling specified for the variable.*

SET_ *The set number of the original variable.*

DIM1...DIMn *Numeric variables containing the category quantifications, the single-category coordinates, multiple-category coordinates, weights, centroids, projected centroids, and component loadings for each dimension.* Each one of these variables is labeled *DIMn*, where *n* represents the dimension number. If any of these values cannot be computed, they are assigned 0 in the file.

See the *SPSS Syntax Reference Guide* for more information on matrix data files.

PRINCALS

```
PRINCALS VARIABLES=varlist(max)

[/ANALYSIS=varlist[({ORDI**})]]
                    {SNOM  }
                    {MNOM  }
                    {NUME  }

[/NOBSERVATIONS=value]

[/DIMENSION={2**  }]
            {value}

[/MAXITER={100**}]
          {value}

[/CONVERGENCE={.00001**}]
              {value   }

[/PRINT=[DEFAULT] [FREQ**] [EIGEN**] [LOADINGS**] [QUANT]
        [HISTORY] [CORRELATION] [OBJECT] [ALL] [NONE]]

[/PLOT=[NDIM=({1    ,2    }**)]
              {value,value}
              {ALL  ,MAX  }
       [DEFAULT[(n)]] [OBJECT**[(varlist)][(n)]]
       [QUANT**[(varlist)][(n)]] [LOADINGS[(n)]]
       [ALL[(n)]] [NONE]]

[/SAVE=[rootname] [(value}]

[/MATRIX=OUT({*   })]
             {file}
```

**Default if subcommand or keyword is omitted.

Overview

PRINCALS (*princ*ipal *c*omponents analysis by means of *a*lternating *l*east *s*quares) analyzes a set of variables for major dimensions of variation. The variables can be of mixed optimal scaling levels, and the relationships among observed variables are not assumed to be linear.

Options

Optimal scaling level. You can specify the optimal scaling level for each variable to be used in the analysis.

Number of cases. You can restrict the analysis to the first *n* observations.

Number of dimensions. You can specify how many dimensions PRINCALS should compute.

Iterations and convergence. You can specify the maximum number of iterations and the value of a convergence criterion.

Display output. The output can include all available statistics, only the default statistics, or only the specific statistics you request. You can also control whether some of these statistics are plotted.

Saving scores. You can save object scores in the working data file.

Writing matrices. You can write a matrix data file containing category quantifications and loadings for use in further analyses.

Basic Specification

- The basic specification is command PRINCALS and the VARIABLES subcommand. PRINCALS performs the analysis assuming an ordinal level of optimal scaling for all variables and uses all cases to compute a two-dimensional solution. By default, marginal frequencies, eigenvalues, and summary measures of fit and loss are displayed, and quantifications and object scores are plotted.

Subcommand Order

- The VARIABLES subcommand must precede all others.
- Other subcommands can appear in any order.

Operations

- If the ANALYSIS subcommand is specified more than once, PRINCALS is not executed. For all other subcommands, only the last occurrence of each subcommand is executed.
- PRINCALS treats every value in the range of 1 to the maximum value specified on VARIABLES as a valid category. Use the AUTORECODE or RECODE command if you want to recode a categorical variable with nonsequential values or with a large number of categories to avoid unnecessary output. For variables treated as numeric, recoding is *not* recommended because the intervals between consecutive categories will not be maintained.

Limitations

- String variables are not allowed; use AUTORECODE to recode nominal string variables into numeric ones before using PRINCALS.
- The data must be positive integers. Zeros and negative values are treated as system-missing and are excluded from the analysis. Fractional values are truncated after the decimal and are included in the analysis. If one of the levels of a categorical variable has been coded 0 or a negative value and you want to treat it as a valid category, use the AUTORECODE or RECODE command to recode the values of that variable (see the *SPSS Syntax Reference Guide* for more information on AUTORECODE and RECODE).
- PRINCALS ignores user-missing value specifications. Positive user-missing values less than the maximum value on the VARIABLES subcommand are treated as valid category values and are included in the analysis. If you do not want the category included, you

can use COMPUTE or RECODE to change the value to something outside of the valid range. Values outside of the range (less than 1 or greater than the maximum value) are treated as system-missing.

Example

```
PRINCALS VARIABLES=ACOLA BCOLA(2) PRICEA PRICEB(5)
 /ANALYSIS=ACOLA BCOLA(SNOM) PRICEA PRICEB(NUME)
 /PRINT=QUANT OBJECT.
```

- VARIABLES defines the variables and their maximum number of levels.
- The ANALYSIS subcommand specifies that variables *ACOLA* and *BCOLA* are single nominal (SNOM) and that variables *PRICEA* and *PRICEB* are numeric (NUME).
- The PRINT subcommand lists the category quantifications and object scores.
- By default, plots of the category quantifications and the object scores are produced.

VARIABLES Subcommand

VARIABLES specifies all of the variables that will be used in the current PRINCALS procedure.

- The VARIABLES subcommand is required and precedes all other subcommands. The actual word VARIABLES can be omitted.
- Each variable or variable list is followed by the maximum number of categories (levels) in parentheses.
- The number specified in parentheses indicates the number of categories *and* the maximum category value. For example, *VAR1(3)* indicates that *VAR1* has three categories coded 1, 2, and 3. However, if a variable is not coded with consecutive integers, the number of categories used in the analysis will differ from the number of observed categories. For example, if a three category variable is coded {2, 4, 6}, the maximum category value is 6. The analysis treats the variable as having six categories, three of which are not observed and receive quantifications of 0.
- To avoid unnecessary output, use the AUTORECODE or RECODE command before PRINCALS to recode a categorical variable that was coded with nonsequential values. As noted in "Limitations," recoding is *not* recommended with variables treated as numeric (see the *SPSS Base Syntax Reference Guide* for more information on AUTORECODE and RECODE).

Example

```
DATA LIST FREE/V1 V2 V3.
BEGIN DATA
3 1 1
6 1 1
3 1 3
3 2 2
3 2 2
6 2 2
6 1 3
6 2 2
3 2 2
6 2 1
END DATA.
AUTORECODE V1 /INTO NEWVAR1.
PRINCALS VARIABLES=NEWVAR1 V2(2) V3(3).
```

- DATA LIST defines three variables, *V1*, *V2*, and *V3*.
- *V1* has two levels, coded 3 and 6, *V2* has two levels, coded 1 and 2, and *V3* has three levels, coded 1, 2, and 3.
- The AUTORECODE command creates *NEWVAR1* containing recoded values of *V1*. Values of 3 are recoded to 1 and values of 6 are recoded to 2.
- A maximum value of 2 can then be specified on the PRINCALS VARIABLES subcommand as the maximum category value for both *NEWVAR1* and *V2*. A maximum value of 3 is specified for *V3*.

ANALYSIS Subcommand

ANALYSIS specifies the variables to be used in the computations and the optimal scaling level used by PRINCALS to quantify each variable or variable list.

- If ANALYSIS is not specified, an ordinal level of optimal scaling is assumed for all variables.
- The specification on ANALYSIS is a variable list and an optional keyword in parentheses to indicate the optimal scaling level.
- The variables on the variable list must also be specified on the VARIABLES subcommand.
- Variables listed on the VARIABLES subcommand but not on the ANALYSIS subcommand can still be used to label object scores on the PLOT subcommand.

The following keywords can be specified to indicate the optimal scaling level:

MNOM *Multiple nominal*. The quantifications can be different for each dimension. When all variables are multiple nominal, PRINCALS gives the same results as HOMALS.

SNOM *Single nominal*. PRINCALS gives only one quantification for each category. Objects in the same category (cases with the same value on a variable) obtain the same quantification. When DIMENSION=1 and all variables are SNOM, this solution is the same as that of the first HOMALS dimension.

ORDI *Ordinal*. This is the default for variables listed without optimal scaling levels and for all variables if the ANALYSIS subcommand is not used. The order of the categories of the observed variable is preserved in the quantified variable.

NUME *Numerical*. This is the interval or ratio level of optimal scaling. PRINCALS assumes that the observed variable already has numerical values for its categories. When all variables are at the numerical level, the PRINCALS analysis is analogous to classical principal components analysis.

These keywords can apply to a variable list as well as to a single variable. Thus, the default ORDI is not applied to a variable without a keyword if a subsequent variable on the list has a keyword.

NOBSERVATIONS Subcommand

NOBSERVATIONS specifies how many cases are used in the analysis.

- If NOBSERVATIONS is not specified, all available observations in the working data file are used.
- NOBSERVATIONS is followed by an integer indicating that the first *n* cases are to be used.

DIMENSION Subcommand

DIMENSION specifies the number of dimensions you want PRINCALS to compute.

- If you do not specify the DIMENSION subcommand, PRINCALS computes two dimensions.
- DIMENSION is followed by an integer indicating the number of dimensions.
- If all of the variables are SNOM (single nominal), ORDI (ordinal), or NUME (numerical), the maximum number of dimensions you can specify is the smaller of the number of observations minus 1 *or* the total number of variables.
- If some or all of the variables are MNOM (multiple nominal), the maximum number of dimensions is the smaller of the number of observations minus 1 *or* the total number of valid MNOM variable levels (categories) plus the number of SNOM, ORDI, and NUME variables, minus the number of MNOM variables without missing values.
- PRINCALS adjusts the number of dimensions to the maximum if the specified value is too large.
- The minimum number of dimensions is 1.

MAXITER Subcommand

MAXITER specifies the maximum number of iterations PRINCALS can go through in its computations.

- If MAXITER is not specified, PRINCALS will iterate up to 100 times.
- MAXITER is followed by an integer indicating the maximum number of iterations allowed.

CONVERGENCE Subcommand

CONVERGENCE specifies a convergence criterion value. PRINCALS stops iterating if the difference in total fit between the last two iterations is less than the CONVERGENCE value.

- If CONVERGENCE is not specified, the default value is 0.00001.
- The specification on CONVERGENCE is a convergence criterion value.

PRINT Subcommand

PRINT controls which statistics are included in your output. The default output includes frequencies, eigenvalues, loadings, and summary measures of fit and loss.

PRINT is followed by one or more of the following keywords:

FREQ *Marginal frequencies for the variables in the analysis.*

HISTORY *History of the iterations.*

EIGEN *Eigenvalues.*

CORRELATION *Correlation matrix for the transformed variables in the analysis.* No correlation matrix is produced if there are any missing data.

OBJECT *Object scores.*

QUANT *Category quantifications and category coordinates for SNOM, ORDI, and NUME variables and category quantifications in each dimension for MNOM variables.*

LOADINGS *Component loadings for SNOM, ORDI, and NUME variables.*

DEFAULT *FREQ, EIGEN, LOADINGS, and QUANT.*

ALL *All of the available statistics.*

NONE *Summary measures of fit.*

PLOT Subcommand

PLOT can be used to produce plots of category quantifications, object scores, and component loadings.

- If PLOT is not specified, plots of the object scores and the quantifications are produced.
- No plots are produced for a one-dimensional solution.

PLOT is followed by one or more of the following keywords:

LOADINGS *Plots of the component loadings of SNOM, ORDI, and NUME variables.*

OBJECT *Plots of the object scores.*

QUANT *Plots of the category quantifications for MNOM variables and plots of the single-category coordinates for SNOM, ORDI, and NUME variables.*

DEFAULT *QUANT and OBJECT.*

ALL *All available plots.*

NONE *No plots.*

- Keywords OBJECT and QUANT can each be followed by a variable list in parentheses to indicate that plots should be labeled with these variables. For QUANT, the variables must be specified on both the VARIABLES and ANALYSIS subcommands. For OBJECT, the variables must be specified on VARIABLES but need not appear on the ANALYSIS subcommand. This means that variables not included in the computations can still be used to label OBJECT plots. If the variable list is omitted, only the default plots are produced.
- Object scores plots labeled with variables that appear on the ANALYSIS subcommand use category labels corresponding to all categories within the defined range. Objects in a category that is outside the defined range are labeled with the label corresponding to the next category greater than the defined maximum category.
- Object scores plots labeled with variables not included on the ANALYSIS subcommand use all category labels, regardless of whether or not the category value is inside the defined range.
- All of the keywords except NONE can be followed by an integer in parentheses to indicate how many characters of the variable or value label are to be used on the plot. (If you specify a variable list after OBJECT or QUANT, you can specify the value in parentheses after the list.) The value can range from 1 to 20. If the value is omitted, twelve characters are used. Spaces between words count as characters.
- The LOADINGS plots and one of the QUANT plots use variable labels; all other plots that use labels use value labels.
- If a variable label is missing, the variable name is used for that variable. If a value label is missing, the actual value is used.
- You should make sure that your variable and value labels are unique by at least one letter in order to distinguish them on the plots.
- When points overlap, the points involved are described in a summary following the plot.

Example

```
PRINCALS VARIABLES COLA1 (4) COLA2 (4) COLA3 (4) COLA4 (2)
 /ANALYSIS COLA1 COLA2 (SNOM) COLA3 (ORDI) COLA4 (ORDI)
 /PLOT OBJECT(COLA4).
```

- Four variables are included in the analysis.
- OBJECT requests a plot of the object scores labeled with the values of *COLA4*. Any object whose *COLA4* value is not 1 or 2 is labeled 3 (or the value label for category 3, if defined).

Example

```
PRINCALS VARIABLES COLA1 (4) COLA2 (4) COLA3 (4) COLA4 (2)
 /ANALYSIS COLA1 COLA2 (SNOM) COLA3 (ORDI)
 /PLOT OBJECT(COLA4).
```

- Three variables are included in the analysis.

- OBJECT requests a plot of the object scores labeled with the values of *COLA4,* a variable not included in the analysis. Objects are labeled using all values of *COLA4.*

In addition to the plot keywords, the following can be specified:

NDIM *Dimension pairs to be plotted.* NDIM is followed by a pair of values in parentheses. If NDIM is not specified, plots are produced for dimension 1 versus dimension 2.

- The first value indicates the dimension that is plotted against all higher dimensions. This value can be any integer from 1 to the number of dimensions minus 1.
- The second value indicates the highest dimension to be used in plotting the dimension pairs. This value can be any integer from 2 to the number of dimensions.
- Keyword ALL can be used instead of the first value to indicate that all dimensions are paired with higher dimensions.
- Keyword MAX can be used instead of the second value to indicate that plots should be produced up to, and including, the highest dimension fit by the procedure.

Example

```
PRINCALS COLA1 COLA2 COLA3 COLA4 (4)
 /PLOT NDIM(1,3) QUANT(5).
```

- The NDIM(1,3) specification indicates that plots should be produced for two dimension pairs—dimension 1 versus dimension 2 and dimension 1 versus dimension 3.
- QUANT requests plots of the category quantifications. The (5) specification indicates that the first five characters of the value labels are to be used on the plots.

Example

```
PRINCALS COLA1 COLA2 COLA3 COLA4 (4)
 /PLOT NDIM(ALL,3) QUANT(5).
```

- This plot is the same as above except for the ALL specification following NDIM. This indicates that all possible pairs up to the second value should be plotted, so QUANT plots will be produced for dimension 1 versus dimension 2, dimension 2 versus dimension 3, and dimension 1 versus dimension 3.

SAVE Subcommand

SAVE lets you add variables containing the object scores computed by PRINCALS to the working data file.

- If SAVE is not specified, object scores are not added to the working data file.
- A variable rootname can be specified on the SAVE subcommand to which PRINCALS adds the number of the dimension. Only one rootname can be specified, and it can contain up to six characters.
- If a rootname is not specified, unique variable names are automatically generated. The variable names are *PRIn_m*, where *n* is a dimension number and *m* is a set number. If three dimensions are saved, the first set of names is *PRI1_1*, *PRI2_1*, and *PRI3_1.* If another

PRINCALS is then run, the variable names for the second set are *PRI1_2*, *PRI2_2*, *PRI3_2*, and so on.

- Following the name, the number of dimensions for which you want to save object scores can be listed in parentheses. The number cannot exceed the value of the DIMENSION subcommand.
- If the number of dimensions is not specified, the SAVE subcommand saves object scores for all dimensions.
- If you replace the working data file by specifying an asterisk (*) on a MATRIX subcommand, the SAVE subcommand is not executed.
- The prefix should be unique for each PRINCALS command in the same session. If it is not, PRINCALS replaces the prefix with *DIM*, *OBJ*, or *OBSAVE*. If all of these already exist, SAVE is not executed.

Example

```
PRINCALS CAR1 CAR2 CAR3(5) PRICE (10)
 /ANALYSIS=CAR1 TO CAR3(SNOM) PRICE(NUM)
 /DIMENSIONS=3
 /SAVE=DIM(2).
```

- Three nominal variables, *CAR1*, *CAR2*, and *CAR3*, each with five categories, and one numerical (interval level) variable, with ten categories, are analyzed in this PRINCALS example.
- The DIMENSIONS subcommand requests results for three dimensions.
- SAVE adds the object scores from the first two dimensions to the working data file. The names of these new variables will be *DIM00001* and *DIM00002*, respectively.

MATRIX Subcommand

The MATRIX subcommand is used to write category quantifications, single-category coordinates, and component loadings to a matrix data file.

- The specification on MATRIX is keyword OUT and the file enclosed in parentheses.
- You can specify the file with either an asterisk (*) to indicate that the working data file is to be replaced or with the name of an external file.
- The category quantifications, coordinates, and component loadings are written to the same file.
- The matrix data file has one case for each value of each original variable.

The variables of the matrix data file and their values are:

ROWTYPE_ *String variable rowtype_ containing value QUANT for the category quantifications, SCOOR_ for single-category coordinates, MCOOR_ for multiple-category coordinates, and LOADING_ for the component scores.*

LEVEL *String variable containing the values (or value labels if present) of each original variable for category quantifications.* For cases with *ROWTYPE_=LOADING_*, the value of *LEVEL* is blank.

VARNAME_ *String variable containing the original variable names.*

VARTYPE_ *String variable containing values MULTIPLE, SINGLE N, ORDINAL, or NUMERICAL, depending on the optimal scaling level specified for the variable.*

DIM1...DIMn *Numeric variables containing category quantifications, the single-category coordinates, and component loadings for each dimension.* Each variable is labeled *DIMn*, where *n* represents the dimension number. The single-category coordinates and component loadings are written only for SNOM, ORDI, and NUME variables.

See the *SPSS Syntax Reference Guide* for more information on matrix data files.

PROXSCAL

```
PROXSCAL varlist

[/TABLE = {rowid BY columid [BY sourceid]}]
          {sourceid                       }

[/SHAPE = [{LOWER**}]]
           {UPPER  }
           {BOTH   }

[/INITIAL = [{SIMPLEX**        }]]
             {TORGERSON        }
             {RANDOM[({1})]    }
                      {n}
             {[(file)] [varlist] }

[/WEIGHTS = varlist]

[/CONDITION = [{MATRIX**      }]]
               {UNCONDITIONAL }

[/TRANSFORMATION = [{RATIO**                              }]]
                    {INTERVAL                             }
                    {ORDINAL[({UNTIE   })]                }
                             {KEEPTIES}
                    {SPLINE [DEGREE = {2}] [INKNOT = {1}]}
                                      {n}            {n}

[/PROXIMITIES = [{DISSIMILARITIES**}]]
                 {SIMILARITIES     }

[/MODEL = [{IDENTITY**    }]]
           {WEIGHTED      }
           {GENERALIZED   }
           {REDUCED[({2})]}
                    {n}

[/RESTRICTIONS = {COORDINATES(file) [{ALL    }]                                  }]
                                     {varlist}
                 {VARIABLES(file) [{ALL    }][({INTERVAL                       })]}
                                   {varlist}   {NOMINAL                        }
                                               {ORDINAL[({UNTIE   })]          }
                                                        {KEEPTIES}
                                               {SPLINE[DEGREE={2}][INKNOT={1}]}
                                                              {n}         {n}

[/ACCELERATION = NONE]

[/CRITERIA = [DIMENSIONS({2**      })]
                         {min[,max]}
             [MAXITER({100**})]
                      {n    }
             [DIFFSTRESS({0.0001**})]
                         {value   }
             [MINSTRESS({0.0001**}) ]]
                        {value   }
```

```
[/PRINT = [NONE][INPUT][RANDOM][HISTORY][STRESS**][DECOMPOSITION]
          [COMMON**][DISTANCES][WEIGHTS**][INDIVIDUAL]
          [TRANSFORMATIONS][VARIABLES**][CORRELATIONS**]]

[/PLOT = [NONE][STRESS][COMMON**][WEIGHTS**][CORRELATIONS**]
         [INDIVIDUAL({varlist})]
                     {ALL    }
         [TRANSFORMATIONS({varlist}) [({varlist})[...]] ]
                         {ALL    }   {ALL    }
         [RESIDUALS({varlist}) [({varlist})[...]] ]
                    {ALL    }   {ALL    }
         [VARIABLES({varlist})]]
                    {ALL    }

[/OUTFILE = [COMMON(file)] [WEIGHTS(file)] [DISTANCES(file)]
            [TRANSFORMATIONS(file)] [VARIABLES(file)] ]

[/MATRIX = IN({file})]].
```

** Default if the subcommand is omitted.

Overview

PROXSCAL performs multidimensional scaling of proximity data to find a least-squares representation of the objects in a low-dimensional space. Individual differences models are allowed for multiple sources. A majorization algorithm guarantees monotone convergence for optionally transformed metric and nonmetric data under a variety of models and constraints.

Options

Data input. You can read one or more square matrices of proximities that can either be symmetrical or asymmetrical. Alternatively, you can provide specifications with the TABLE subcommand for matrices with proximities in a stacked format. You can read proximity matrices created by PROXIMITIES and CLUSTER with the MATRIX subcommand. Additionally, you can read weights, initial configurations, fixed coordinates, and independent variables.

Methodological assumptions. You can specify transformations considering all sources (unconditional) or separate transformations for each source (matrix-conditional) on the CONDITION subcommand. You can treat proximities as nonmetric (ordinal) or as metric (numerical or splines) using the TRANSFORMATION subcommand. Ordinal transformations can treat tied observations as tied (discrete) and untied (continuous). You can specify whether your proximities are similarities or dissimilarities on the PROXIMITIES subcommand.

Model selection. You can specify multidimensional scaling models by selecting a combination of PROXSCAL subcommands, keywords, and criteria. The subcommand MODEL offers, besides the identity model, three individual differences models. You can specify other selections on the CRITERIA subcommand.

Constraints. You can specify fixed coordinates or independent variables to restrict the configuration(s) on the RESTRICTIONS subcommand. You can specify transformations (numerical, nominal, ordinal, and splines) for the independent variables on the same subcommand.

Output. You can produce output that includes the original and transformed proximities, history of iterations, common and individual configurations, individual space weights, distances, and decomposition of the stress. Plots can be produced of common and individual configurations, individual space weights, transformations, and residuals.

Basic Specification

The basic specification is PROXSCAL followed by a variable list. By default, PROXSCAL produces a two-dimensional metric Euclidean multidimensional scaling solution (identity model). Input is expected to contain one or more square matrices with proximities that are dissimilarities. The ratio transformation of the proximities is matrix-conditional. The analysis uses a simplex start as an initial configuration. By default, output includes fit and stress values, the coordinates of the common space, and a chart of the common space configuration.

Syntax Rules

- The number of dimensions (both minimum and maximum) may not exceed the number of proximities minus one.
- Dimensionality reduction is omitted if combined with multiple random starts.
- If there is only one source, then the model is always assumed to be identity.

Limitations

- PROXSCAL needs at least three objects, which means that at least three variables must be specified in the variable list. In the case of the TABLE subcommand, the minimum value for rowid and columnid must be at least three.
- PROXSCAL recognizes data weights created by the WEIGHT command but only in combination with the TABLE subcommand.
- Split-file has no implications for PROXSCAL.

Variable List Subcommand

The variable list identifies the columns in the proximity matrix or matrices that PROXSCAL reads. Each variable identifies one column of the proximity matrix, with each case in the working data file representing one row, unless specified otherwise with the TABLE subcommand. In this case, the variable list identifies whole matrices or sources.

- Only numeric variables may be specified.
- The total number of cases must be divisible by the number of variables. This is not applicable when the TABLE subcommand is used.
- PROXSCAL reads data row by row; the columns are represented by the variables on the variable list. The order of the variables on the list is crucial.

Example

```
DATA LIST
  /object01 object02 object03 object04.

BEGIN DATA
  0 2 6 3
  2 0 5 4
  6 5 0 1
  3 4 1 0
END DATA.

PROXSCAL VARIABLES=object01 TO object04.
```

- This example specifies an analysis on a 4×4 proximity matrix.
- The total number of cases must be divisible by 4.

TABLE Subcommand

The TABLE subcommand specifies the row identifier rowid and the column identifier columnid. Using TABLE, the proximities of separate sources are given in separate variables on the PROXSCAL variable list.

In the same manner, sources are identified by sourceid. In combination with rowid and columnid, the proximities are stacked in one single variable, containing the proximities of all sources, where sources are distinguished by the values of sourceid.

Using sourceid as the only variable on the TABLE subcommand indicates the use of stacked matrices, where individual stacked matrices are recognized by different values of sourceid.

- Rowid, columnid, and sourceid should not be specified on the variable list.
- When specifying both upper- and lower-triangular parts of the matrix, the SHAPE subcommand will determine the handling of the data.
- If a cell's value is specified multiple times, the final specification is used.
- Rowid, columnid, and sourceid must appear in that order.
- Omitting sourceid causes PROXSCAL to use the sources specified on the PROXSCAL variable list. Each variable is assumed to contain the proximities of one source.
- Specifying multiple sources on the PROXSCAL variable list in conjunction with specifying rowid, columnid, and sourceid is not possible and causes PROXSCAL to ignore sourceid.

rowid *Row identifying variable.* The values of this variable specify the row object of a proximity. The values must be integers between 1 and the number of objects, inclusive.

columnid *Column identifying variable.* The values specify the column object of a proximity. The values must be integers between 1 and the number of objects, inclusive.

sourceid *Source identifying variable.* The values specify the source number and must be integers between 1 and the number of sources, inclusive. The value labels of this variable are used to identify sources on other subcommands. These value labels must comply with SPSS variable name conventions. Omitting a value label causes PROXSCAL to use the default label *SRC_n* where *n* is the number of the source.

Example

```
DATA LIST
  /r_id c_id men women.

BEGIN DATA
2 1 1.08 1.14
3 1 0.68 1.12
3 2 0.95 0.75
4 1 0.96 0.32
4 2 0.76 0.98
4 3 0.47 0.69
. . .... ....
.. .. .... ....
13 10 0.55 0.86
13 11 0.61 0.97
13 12 0.46 0.83
END DATA.

PROXSCAL men women
  /TABLE=r_id BY c_id
  /PLOT = INDIVIDUAL (women).
```

- PROXSCAL reads two proximity matrices (*men* and *women*), where the row objects are specified by *r_id* and the column objects by *c_id*.
- A chart of the individual space for *women* is plotted.

This is one way to proceed. Another way is to add the proximities of the additional source below the proximities of the first source and specify sourceid on the TABLE subcommand, containing values distinguishing the first and the additional source (see the next example).

Example

```
DATA LIST
  /r_id c_id s_id prox.

BEGIN DATA
2 1 1 1.08
3 1 1 0.68
3 2 1 0.95
4 1 1 0.96
4 2 1 0.76
4 3 1 0.47
. . . ....
.. .. . ....
13 10 1 0.55
13 11 1 0.61
13 12 1 0.46
2 1 2 1.14
3 1 2 1.12
3 2 2 0.75
4 1 2 0.32
4 2 2 0.98
4 3 2 0.69
. . . ....
.. .. . ....
13 10 2 0.86
13 11 2 0.97
13 12 2 0.83
END DATA.

VALUE LABELS s_id 1 'men' 2 'women'.

PROXSCAL prox
  /TABLE=r_id BY c_id BY s_id
  /PLOT = INDIVIDUAL (women).
```

- PROXSCAL reads two proximity matrices. The row objects are identified by *r_id* and the column objects by *c_id*. The proximity matrices are gathered in one variable, *source01*, where each source is distinguished by a value of the source identifying variable *s_id*.
- A chart of the individual space for *women* is plotted.

Example

```
DATA LIST
  /obj_1 obj_2 obj_3 obj_4 s_id

BEGIN DATA
0 0 0 0 1
1 0 0 0 1
2 3 0 0 1
4 5 6 0 1
0 0 0 0 2
8 9 0 0 2
10 11 12 0 2
END DATA.

VALUE LABELS s_id 1 'women' 2 'men'.

PROXSCAL obj_1 obj_2 obj_3 obj_4
  /TABLE = s_id
  /PLOT = INDIVIDUAL (women).
```

- PROXSCAL reads two proximity matrices. The objects are given on the PROXSCAL variable list. Each source is distinguished by a value of the source identifying variable *s_id*, which is also used for labeling.
- A chart of the individual space for *women* is plotted.

SHAPE Subcommand

The SHAPE subcommand specifies the structure of the proximity matrix.

LOWER *Lower-triangular data matrix.* For a lower-triangular matrix, PROXSCAL expects a square matrix of proximities of which the lower-triangular elements are used under the assumption that the full matrix is symmetric. The diagonal is ignored but must be included.

UPPER *Upper-triangular data matrix.* For an upper-triangular matrix, PROXSCAL expects a square matrix of proximities of which the upper-triangular elements are used under the assumption that the full matrix is symmetric. The diagonal is ignored but must be included.

BOTH *Full data matrix.* The values in the corresponding cells in the upper and lower triangles may be different. PROXSCAL reads the complete square matrix and, after obtaining symmetry, continues with the lower-triangular elements. The diagonal is ignored but must be included.

- System or other missing values on the (virtual) diagonal are ignored.

Example

```
PROXSCAL object01 TO object07
  /SHAPE=UPPER.
```

- PROXSCAL reads square matrices of seven columns per matrix of which the upper-triangular parts are used in computations.
- Although specified, the diagonal and lower-triangular part of the matrix are not used.

INITIAL Subcommand

INITIAL defines the initial or starting configuration of the common space for the analysis. When a reduction in dimensionality is specified on the CRITERIA subcommand, a derivation of coordinates in the higher dimensionality is used as a starting configuration in the lower dimensionality.

- You can specify one of the three keywords listed below.
- You can specify a variable list containing the initial configuration.

SIMPLEX *Simplex start.* This specification is the default. PROXSCAL starts by placing the objects in the configuration all at the same distance of each other and taking one iteration to improve this high-dimensional configuration, followed by a dimension-reduction operation to obtain the user-provided maximum dimensionality specified in the CRITERIA subcommand with the keyword DIMENSIONS.

TORGERSON *Torgerson start.* A classical scaling solution is used as initial configuration.

RANDOM *(Multiple) random start.* You can specify the number of random starts (*n*). *n* is any positive integer. The random sequence can be controlled by the RANDOM SEED command and not by a subcommand within the PROXSCAL command. Each analysis starts with a different random configuration. In the output, all *n* final stress values are reported, as well as the initial seeds of each analysis (for reproduction purposes), followed by the full output of the analysis with the lowest stress value. The default number of random starts is 1. Reduction of dimensionality—that is, using a maximum dimensionality that is larger than the minimum dimensionality—is not allowed within this option and the minimum dimensionality is used, if reduction is specified anyway.

Instead of these keywords, a parenthesized SPSS data file can be specified containing the coordinates of the initial configuration. If the variable list is omitted, the first MAXDIM variables are automatically selected, where MAXDIM is the maximum number of dimensions requested for the analysis on the CRITERIA subcommand. Only nonmissing values are allowed as initial coordinates.

Example

```
PROXSCAL object01 TO object17
  /INITIAL=RANDOM(100).
```

- This example performs 100 analyses each, starting with different random configurations. The results of the analysis with the lowest final stress are displayed in the output.

WEIGHTS Subcommand

The WEIGHTS subcommand specifies non-negative weights on the proximities included in the working data file.

- The number and order of the variables in the variable list is important. The first variable on the WEIGHTS variable list corresponds to the first variable on the PROXSCAL variable list. This is repeated for all variables on the variable lists. Every proximity has its own weight. The number of variables on the WEIGHTS subcommand must therefore be equal to the number of variables on the PROXSCAL variable list.
- Negative weights are not allowed. If specified, a warning will be issued and the procedure will abort.

Example

```
DATA LIST FILE='cola.dat' FREE
  /object01 TO object14 weight01 TO weight14.

PROXSCAL object01 TO object14
  /WEIGHTS=weight01 TO weight14.
```

- In this example, the VARIABLES subcommand indicates that there are 14 columns per matrix of which the weights can be found in *weight01* to *weight14*.
- *weight01* contains the weights for *object01*, etc.

CONDITION Subcommand

CONDITION specifies how transformations among sources are compared. The TRANSFORMATION subcommand specifies the type of transformation.

MATRIX *Matrix conditional.* Only the proximities within each source are compared with each other. This is the default.

UNCONDITIONAL *Unconditional.* This specification is appropriate when the proximities in all sources can be compared with each other and result in a single transformation of all sources simultaneously.

- Note that if there is only one source, then MATRIX and UNCONDITIONAL give the same results.

Example

```
PROXSCAL object01 TO object15
  /CONDITION=UNCONDITIONAL
  /TRANSFORMATION=ORDINAL(UNTIE).
```

- In this example, the proximities are ordinally transformed, where tied proximities are allowed to be untied. The transformations are performed simultaneously over all possible sources.

TRANSFORMATION Subcommand

TRANSFORMATION offers four different options for optimal transformation of the original proximities. The resulting values are called transformed proximities. The distances between the objects in the configuration should match these transformed proximities as closely as possible.

RATIO *No transformation.* Omitting the entire subcommand is equivalent to using this keyword. In both cases, the transformed proximities are proportional to the original proximities. This "transformation" is only allowed for positive dissimilarities. In all other cases, a warning is issued and the transformation is set to INTERVAL.

INTERVAL *Numerical transformation.* In this case, the transformed proximities are proportional to the original proximities, including free estimation of the intercept. The inclusion of the intercept assures that all transformed proximities are positive.

ORDINAL *Ordinal transformation.* The transformed proximities have the same order as the original proximities. In parentheses, the approach to tied proximities can be specified. Keeping tied proximities tied, also known as secondary approach to ties, is default. Specification may be implicit, ORDINAL, or explicit, ORDINAL(KEEPTIES). Allowing tied proximities to be untied, also known as the primary approach to ties, is specified as ORDINAL (UNTIE).

SPLINE *Monotone spline transformation.* The transformed proximities are a smooth nondecreasing piecewise polynomial transformation of the original proximities of the chosen degree. The pieces are specified by the number and placement of the interior knots.

SPLINE Keyword

SPLINE has the following keywords:

DEGREE *The degree of the polynomial.* If DEGREE is not specified, the degree is assumed to be 2. The range of DEGREE is between 1 and 3 (inclusive).

INKNOT *The number of interior knots.* If INKNOT is not specified, the number of interior knots is assumed to be 1. The range of INKNOT is between 1 and the number of different proximities.

Example

```
PROXSCAL object01 TO object05
  /TRANSFORMATION=ORDINAL(UNTIE).
```

- In this example, the proximities are ordinally transformed, where tied proximities are allowed to be untied.
- The default conditionality (MATRIX) implies that the transformation is performed for each source separately.

PROXIMITIES Subcommand

The PROXIMITIES subcommand specifies the type of proximities used in the analysis. The term proximity is used for either similarity or dissimilarity data.

DISSIMILARITIES *Dissimilarity data.* This specification is the default when PROXIMITIES is not specified. Small dissimilarities correspond to small distances, and large dissimilarities correspond to large distances.

SIMILARITIES *Similarity data.* Small similarities correspond to large distances and large similarities correspond to small distances.

Example

```
PROXSCAL object01 TO object12
  /PROXIMITIES=SIMILARITIES.
```

- In this example, PROXSCAL expects the proximities to be similarities.

MODEL Subcommand

MODEL defines the scaling model for the analysis if more than one source is present. IDENTITY is the default model. The three other models are individual differences models.

IDENTITY *Identity model.* All sources have the same configuration. This is the default model, and it is not an individual differences model.

WEIGHTED *Weighted Euclidean model.* This model is an individual differences model and equivalent to the INDSCAL model in the ALSCAL procedure. Each source has an individual space, in which every dimension of the common space is weighted differentially.

GENERALIZED *Generalized Euclidean model.* This model is equivalent to the GEMSCAL model in the ALSCAL procedure. Each source has an individual space that is equal to a rotation of the common space, followed by a differential weighting of the dimensions.

REDUCED *Reduced rank model.* This model is similar to GENERALIZED, but the rank of the individual space is equal to *n*. This number is always smaller than the maximum number of dimensions and equal to or greater than 1. The default is 2.

- If IDENTITY is specified for only one source, this subcommand is silently ignored.
- If an individual differences model is specified for only one source, a warning is issued, and the model is set to IDENTITY.

Example

```
PROXSCAL object01 TO object07
  /MODEL=WEIGHTED.
```

- A weighted Euclidean model is fitted, but only when the number of cases in the working data file is a multiple of 7, starting from 14 (14, 21, 28, and so on). Otherwise, there is only one source, and the model is set to IDENTITY.

RESTRICTIONS Subcommand

PROXSCAL provides two types of restrictions for the user to choose from. The first type fixes (some) coordinates in the configuration. The second type specifies that the common space is a weighted sum of independent variables.

COORDINATES *Fixed coordinates.* A parenthesized SPSS data filename must be specified containing the fixed coordinates for the common space. A variable list may be given, if some specific variables need to be selected from the external file. If the variable list is omitted, the procedure automatically selects the first MAXDIM variables in the external file, where MAXDIM is the maximum number of dimensions requested for the analysis on the CRITERIA subcommand. A missing value indicates that a coordinate on a dimension is free. The coordinates of objects with nonmissing values are kept fixed during the analysis. The

number of cases for each variable must be equal to the number of objects.

VARIABLES *Independent variables.* The common space is restricted to be a linear combination of the independent variables in the variable list. A parenthesized SPSS data file must be specified containing the independent variables. If the variable list is omitted, the procedure automatically selects all variables in the external file. Instead of the variable list, the user may specify the keyword FIRST(n), where *n* is a positive integer, to select the first *n* variables in the external file. The number of cases for each variable must be equal to the number of objects. After the variable selection specification, we may provide a list of keywords (in number equal to the number of the independent variables) indicating the transformations for the independent variables.

VARIABLES Keyword

The following keywords may be specified:

INTERVAL *Numerical transformation.* In this case, the transformed values of a variable are proportional to the original values of the variable, including free estimation of the intercept.

NOMINAL *Nominal transformation.* The values are treated as unordered. The same values will obtain the same transformed values.

ORDINAL *Ordinal transformation.* The values of the transformed variable have the same order as the values of the original variable. In parenthesis, the approach to tied values can be specified. Keeping tied values tied, also known as secondary approach to ties, is default. Specification may be implicit, ORDINAL, or explicit, ORDINAL(KEEPTIES). Allowing tied values to be untied, also known as the primary approach to ties, is specified as ORDINAL (UNTIE).

SPLINE *Monotone spline transformation.* The transformed values of the variable are a smooth nondecreasing piecewise polynomial transformation of the original values of the chosen degree. The pieces are specified by the number and placement of the interior knots.

SPLINE Keyword

SPLINE has the following keywords:

DEGREE *The degree of the polynomial.* If DEGREE is not specified, the degree is assumed to be 2. The range of DEGREE is between 1 and 3 (inclusive).

INKNOT *The number of interior knots.* If INKNOT is not specified, the number of interior knots is assumed to be 1. The range of INKNOT is between 0 and the number of different values of the variable.

Example

```
PROXSCAL aunt TO uncle
  /RESTRICTIONS=VARIABLES(ivars.sav) degree generation gender
                          (ORDINAL ORDINAL NOMINAL).
```

- In this example, there are three independent variables specified, namely degree, generation, and gender.
- The variables are specified in the data file *ivars.sav*.
- On both degree and generation, ordinal transformations are allowed. By default, tied values in ordinal variables are kept tied. Gender is allowed to be nominally transformed.

ACCELERATION Subcommand

By default, a fast majorization method is used to minimize stress.

NONE *The standard majorization update.* This turns off the fast method.

- If the subcommand RESTRICTION is used with fixed coordinates or independent variables, ACCELERATION=NONE is in effect.
- If an individual differences model is specified on the MODEL subcommand, ACCELERATION=NONE is in effect.

Example

```
PROXSCAL VARIABLES=object01 TO object12
  /ACCELERATION=NONE.
```

- Here, relaxed updates are switched off through the specification of the keyword NONE after ACCELERATION.

CRITERIA Subcommand

Use CRITERIA to set the dimensionality and criteria for terminating the algorithm, or minimization process. You can specify one or more of the following keywords:

DIMENSIONS *Minimum and maximum number of dimensions.* By default, PROXSCAL computes a solution in two dimensions (min=2 and max=2). The minimum and maximum number of dimensions can be any integers inclusively between 1 and the number of objects minus 1, as long as the minimum is less than or equal to the maximum. PROXSCAL starts computing a solution in the largest dimensionality and reduces the dimensionality in steps, until the lowest dimensionality is reached. Specifying a single value represents both minimum and maximum number of dimensions, thus DIMENSIONS(4) is equivalent to DIMENSIONS(4,4).

MAXITER *Maximum number of iterations.* By default, *n*=100, specifying the maximum number of iterations that is performed while one of the convergence criterion below (CONVERGENCE and STRESSMIN) is not yet reached. Decreasing this number might give less accurate results but will take less time. *n* must have a positive integer value.

DIFFSTRESS *Convergence criterion.* PROXSCAL minimizes the goodness-of-fit index normalized raw stress. By default, PROXSCAL stops iterating when the difference in consecutive stress values is less than 0.0001 (*n*=0.0001). To obtain a more precise solution, you can specify a smaller value. The value specified must lie between 0.0 and 1.0, inclusively.

MINSTRESS *Minimum stress value.* By default, PROXSCAL stops iterating when the stress value itself is small, that is, less than 0.0001 (*n*=0.0001). To obtain an even more precise solution, you can specify a smaller value. The value specified must lie between 0.0 and 1.0, inclusively.

Example

```
PROXSCAL VARIABLES=object01 TO object24
  /CRITERIA=DIMENSIONS(2,4) MAXITER(200) DIFFSTRESS(0.00001).
```

- The maximum number of dimensions equals 4 and the minimum number of dimensions equals 2. PROXSCAL computes a four-, three-, and two-dimensional solution, respectively.
- The maximum number of iteration is raised to 200.
- The convergence criterion is sharpened to 0.00001.

PRINT Subcommand

PRINT specifies the optional output. By default, PROXSCAL displays the stress and fit values for each analysis, the coordinates of the common space, and, with appropriate specification on corresponding subcommands, the individual space weights and transformed independent variables, corresponding regression weights, and correlations.

- Omitting the PRINT subcommand or specifying PRINT without keywords is equivalent to specifying COMMON, WEIGHTS, and VARIABLES.
- If a keyword(s) is specified, only the output for that particular keyword(s) is displayed.
- In the case of duplicate or contradicting keyword specification, the last keyword applies.
- Inapplicable keywords are silently ignored. That is, specifying a keyword for which no output is available (for example, specifying INDIVIDUAL with only one source) will silently ignore this keyword.

NONE *No output.* Display only the normalized raw stress and corresponding fit values.

INPUT *Input data.* The display includes the original proximities, and, if present, the data weights, the initial configuration, and the fixed coordinates or the independent variables.

RANDOM *Multiple random starts.* Displays the random number seed and stress value of each random start.

HISTORY *History of iterations.* Displays the history of iterations of the main algorithm.

STRESS *Stress measures.* Displays different stress values. The table contains values for normalized raw stress, Stress-I, Stress-II, S-Stress, dispersion accounted for (D.A.F.), and Tucker's coefficient of congruence. This is specified by default.

DECOMPOSITION *Decomposition of stress.* Displays an object and source decomposition of stress, including row and column totals.

COMMON *Common space.* Displays the coordinates of the common space. This is specified by default.

DISTANCES *Distances.* Displays the distances between the objects in the configuration.

WEIGHTS *Individual space weights.* Displays the individual space weights, only if one of the individual differences models is specified on the MODEL subcommand. Depending on the model, the space weights are decomposed in rotation weights and dimension weights, which are also displayed. This is specified by default.

INDIVIDUAL *Individual spaces.* The coordinates of the individual spaces are displayed, only if one of the individual differences models is specified on the MODEL subcommand.

TRANSFORMATION *Transformed proximities.* Displays the transformed proximities between the objects in the configuration.

VARIABLES *Independent variables.* If VARIABLES was specified on the RESTRICTIONS subcommand, this keyword triggers the display of the transformed independent variables and the corresponding regression weights. This is specified by default.

CORRELATIONS *Correlations.* The correlations between the independent variables and the dimensions of the common space are displayed. This is specified by default.

Example

```
PROXSCAL VARIABLES=source01 TO source02
  /TABLE=row_id BY col_id
  /MODEL=WEIGHTED
  /PRINT=HISTORY COMMON STRESS.
```

- Here, a weighted Euclidean model is specified with two sources.
- The output consists of the history of iterations of the main algorithm, the coordinates of the common space, the individual space weights, and several measures of fit.

PLOT Subcommand

PLOT controls the display of plots. By default, PROXSCAL produces a scatterplot of object coordinates of the common space, the individual space weights, and the correlations between the independent variables (i.e., equivalent to specifying COMMON, WEIGHTS, and CORRELATIONS).

- Specifying a keyword overrides the default output and only output is generated for that keyword.
- Duplicate keywords are silently ignored.
- In case of contradicting keywords, only the last keyword is considered.
- Inapplicable keywords (for example, stress with equal minimum and maximum number of dimensions on the CRITERIA subcommand) are silently ignored.
- Multiple variable lists are allowed for TRANSFORMATIONS and RESIDUALS. For each variable list, a separate plot will be displayed.

NONE *No plots.* PROXSCAL does not produce any plots.

STRESS *Stress plot.* A plot is produced of stress versus dimensions. This plot is only produced if the maximum number of dimensions is larger than the minimum number of dimensions.

COMMON *Common space.* A scatterplot matrix of coordinates of the common space is displayed.

WEIGHTS *Individual space weights.* A scatterplot is produced of the individual space weights. This is only possible if one of the individual differences models is specified on the MODEL subcommand. For the weighted Euclidean model, the weights are printed in plots with one dimension on each axis. For the generalized Euclidean model, one plot is produced per dimension, indicating both rotation and weighting of that dimension. The reduced rank model produces the same plot as the generalized Euclidean model does but reduces the number of dimensions for the individual spaces.

INDIVIDUAL *Individual spaces.* For each source specified on the variable list, the coordinates of the individual spaces are displayed in scatterplot matrices. This is only possible if one of the individual differences models is specified on the MODEL subcommand.

TRANSFORMATIONS

Transformation plots. Plots are produced of the original proximities versus the transformed proximities. On the variable list, the sources can be specified of which the plot is to be produced.

RESIDUALS *Residuals plots.* The transformed proximities versus the distances are plotted. On the variable list, the sources can be specified of which the plot is to be produced.

VARIABLES *Independent variables.* Transformation plots are produced for the independent variables specified on the variable list.

CORRELATIONS *Correlations.* A plot of correlations between the independent variables and the dimensions of the common space is displayed.

Example

```
PROXSCAL VARIABLES=source01 TO source02
  /TABLE=row_id BY col_id
  /MODEL=WEIGHTED
  /CRITERIA=DIMENSIONS(3)
  /PLOT=COMMON INDIVIDUAL(source02).
```

- Here, the syntax specifies a weighted Euclidean model with two sources in three dimensions.
- COMMON produces a scatterplot matrix defined by dimensions 1, 2, and 3.
- For the individual spaces, a scatterplot matrix with 3 dimensions is only produced for the individual space of *source02*.

OUTFILE Subcommand

OUTFILE saves coordinates of the common space, individual space weights, distances, transformed proximities, and transformed independent variables to an SPSS data file. The only specification required is a name for the output file.

COMMON *Common space coordinates.* The coordinates of the common space are written to an SPSS data file. The columns (variables) represent the dimensions *DIM_1*, *DIM_2*, ..., *DIM_n* of the common space. The number of cases (rows) in the SPSS data file equals the number of objects.

WEIGHTS *Individual space weights.* The individual space weights are written to an SPSS data file. The columns represent the dimensions *DIM_1*, *DIM_2*, ..., *DIM_n* of the space weights. The number of cases depends on the individual differences model specified on the MODEL subcommand. The weighted Euclidean model uses diagonal weight matrices. Only the diagonals are written to file and the number of cases is equal to the number of dimensions. The generalized Euclidean model uses full-rank nonsingular weight matrices. The matrices are written to the SPSS data file row by row. The reduced rank model writes matrices to the SPSS data file in the same way as the generalized Euclidean model does but does not write the reduced part.

DISTANCES *Distances.* The matrices containing the distances for each source are stacked beneath each other and written to an SPSS data file. The number of variables in the data file are equal to the number of objects (*OBJ_1*, *OBJ_2*, ... *OBJ_n*) and the number of cases in the data file are equal to the number of objects times the number of sources.

TRANSFORMATION *Transformed proximities.* The matrices containing the transformed proximities for each source are stacked beneath each other and written to an SPSS data file. The number of variables in the file are equal to the number of objects (*OBJ_1*, *OBJ_2*, ... *OBJ_n*) and the number of

cases in the data file are equal to the number of objects times the number of sources.

VARIABLES *Independent variables.* The transformed independent variables are written to an SPSS data file. The variables are written to the columns (*VAR_1*, *VAR_2*, ..., *VAR_n*). The number of variables in the data file are equal to the number of independent variables and the number of cases are equal to the number of objects.

Example

```
PROXSCAL VARIABLES=source01 TO source04
  /TABLE=row_id BY col_id
  /OUTFILE=COMMON(start.dat).
```

- Here, the coordinates of the common space are written to the SPSS data file *start.dat*.

MATRIX Subcommand

MATRIX reads SPSS matrix data files. It can read a matrix written by either PROXIMITIES or CLUSTER.

- The specification on MATRIX is the keyword IN and the matrix file in parentheses.
- Generally, data read by PROXSCAL are already in matrix form, whether in square format, or in stacked format using the TABLE subcommand.
- The proximity matrices PROXSCAL reads have ROWTYPE_ values of PROX.
- Using MATRIX=IN, PROXSCAL will ignore variables specified on the main variable list. All numerical variables from the matrix data file are processed.
- PROXSCAL ignores variables specified in the WEIGHTS subcommand in combination with the use of MATRIX=IN.
- With MATRIX=IN, only a source identifying variable can be specified on the TABLE subcommand. The sources are created as a result of a split file action.

IN) *Read a matrix data file.* Specify the filename in parentheses. Data read through the MATRIX subcommand does not replace the working data file.

Example

```
GET FILE = 'PROXMTX.SAV'.

PROXSCAL
  /MATRIX=IN('MATRIX.SAV').
```

- MATRIX=IN specifies an external matrix data file called *matrix.sav*, of which all numerical variables are used for the current analysis.

Bibliography

Barlow, R. E., D. J. Bartholomew, D. J. Bremner, and H. D. Brunk. 1972. *Statistical inference under order restrictions*. New York: John Wiley and Sons.

Bell, E. H. 1961. *Social foundations of human behavior: Introduction to the study of sociology*. New York: Harper & Row.

Benzecri, J. P. 1969. Statistical analysis as a tool to make patterns emerge from data. In: *Methodologies of pattern recognition*, S. Watanabe, ed. New York: Academic Press.

Bishop, Y. M., S. E. Feinberg, and P. W. Holland. 1975. *Discrete multivariate analysis*. Cambridge, Mass.: MIT Press.

Breiman, L., and J. H. Friedman. 1985. Estimating optimal transformations for multiple regression and correlation. *Journal of the American Statistical Association*, 80: 580–598.

Buja, A. 1990. Remarks on functional canonical variates, alternating least squares methods and ACE. *Annals of Statistics*, 18: 1032–1069.

Carroll, J. D. 1968. Generalization of canonical correlation analysis to three or more sets of variables. *Proceedings of the 76th annual convention of the American Psychological Association*, 3: 227–228.

Carroll, J. D., and P. E. Green. 1988. An INDSCAL-based approach to multiple correspondence analysis. *Journal of Marketing Research*, 55: 193–203.

Commandeur, J. J. F, and W. J. Heiser. 1993. *Mathematical derivations in the proximity scaling (PROXSCAL) of symmetric data matrices*. Leiden: Department of Data Theory.

De Leeuw, J. 1982. Nonlinear principal components analysis. In: *COMPSTAT Proceedings in Computational Statistics*, 77–89. Vienna: Physica Verlag.

_____. 1984. *Canonical analysis of categorical data*. 2nd ed. Leiden: DSWO Press.

_____. 1984. The Gifi system of nonlinear multivariate analysis. In: *Data analysis and informatics*, E. Diday et al., eds. III: 415–424.

De Leeuw, J., and J. Van Rijckevorsel. 1980. HOMALS and PRINCALS—Some generalizations of principal components analysis. In: *Data analysis and informatics*, E. Diday et al., eds. Amsterdam: North-Holland.

De Leeuw, J., F. W. Young, and Y. Takane. 1976. Additive structure in qualitative data: An alternating least squares method with optimal scaling features. *Psychometrika*, 41: 471–503.

Eckart, C., and G. Young. 1936. The approximation of one matrix by another one of lower rank. *Psychometrika*, I: 211–218.

Fisher, R. A. 1938. *Statistical methods for research workers*. Edinburgh: Oliver and Boyd.

_____. 1940. The precision of discriminant functions. *Annals of Eugenics*, 10: 422–429.

Gabriel, K. R. 1971. The biplot graphic display of matrices with application to principal components analysis. *Biometrika*, 58: 453–467.

Gifi, A. 1985. *PRINCALS*. Research Report UG-85-02. Leiden: Department of Data Theory, University of Leiden.

_____. 1990. *Nonlinear multivariate analysis*. Chichester: John Wiley and Sons (First edition 1981, Department of Data Theory, University of Leiden).

Gilula, Z., and S. J. Haberman. 1988. The analysis of multivariate contingency tables by restricted canonical and restricted association models. *Journal of the American Statistical Association*, 83: 760–771.

Gower, J. C., and J. J. Meulman. 1993. The treatment of categorical information in physical anthropology. *International Journal of Anthropology*, 8: 43–51.

Green, P. E., and Y. Wind. 1973. *Multiattribute decisions in marketing: A measurement approach*. Hinsdale, Ill.: Dryden Press.

Greenacre, M. J. 1984. *Theory and applications of correspondence analysis*. London: Academic Press.

Guttman, L. 1941. The quantification of a class of attributes: A theory and method of scale construction. In: *The prediction of personal adjustment*, P. Horst et al., eds. New York: Social Science Research Council.

_____. 1968. A general nonmetric technique for finding the smallest coordinate space for configurations of points. *Psychometrika*, 33: 469–506.

Hartigan, J. A. 1975. *Clustering algorithms*. New York: John Wiley and Sons.

Hastie, T. J., and R. J. Tibshirani. 1990. *Generalized additive models*. London: Chapman and Hall.

Heiser, W. J. 1981. *Unfolding analysis of proximity data*. Leiden: Department of Data Theory, University of Leiden.

Heiser, W. J., and J. J. Meulman. 1994. Homogeneity analysis: Exploring the distribution of variables and their nonlinear relationships. In: *Correspondence analysis in the social sciences: Recent developments and applications*, M. Greenacre and J. Blasius, eds. New York: Academic Press, 179–209.

_____. 1995. Nonlinear methods for the analysis of homogeneity and heterogeneity. In: *Recent advances in descriptive multivariate analysis*, W. J. Krzanowski, ed. Oxford: Oxford University Press, 51–89.

Horst, P. 1935. Measuring complex attitudes. *Journal of Social Psychology*, 6: 369–374.

_____. 1963. *Matrix algebra for social scientists*. New York: Holt, Rinehart and Winston.

Israels, A. 1987. *Eigenvalue techniques for qualitative data*. Leiden: DSWO Press.

Kennedy, R., C. Riquier, and B. Sharp. 1996. Practical applications of correspondence analysis to categorical data in market research. *Journal of Targeting, Mearurement, and Analysis for Marketing*, 5(1): 56–70

Kruskal, J. B. 1964. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika*, 29: 1–28.

_____. 1964. Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29: 115–129.

_____. 1965. Analysis of factorial experiments by estimating monotone transformations of the data. *Journal of the Royal Statistical Society Series B*, 27: 251–263.

_____. 1978. Factor analysis and principal components analysis: Bilinear methods. In: *International encyclopedia of statistics*. W. H. Kruskal and J. M. Tanur, eds. New York: The Free Press, 307–330.

Kruskal, J. B., and R. N. Shepard. 1974. A nonmetric variety of linear factor analysis. *Psychometrika*, 39: 123–157.

Krzanowski W. J., and F. H. C. Marriott. 1994. *Multivariate analysis: Part I, distributions, ordination and inference*. London: Edward Arnold.

Lebart L., A. Morineau, and K. M. Warwick. 1984. *Multivariate descriptive statistical analysis*. New York: John Wiley and Sons.

Lingoes, J. C. 1968. The multivariate analysis of qualitative data. *Multivariate Behavioral Research*, 3: 61–94.

Meulman, J. 1982. *Homogeneity analysis of incomplete data*. Leiden: DSWO Press.

_____. 1986. *A distance approach to nonlinear multivariate analysis*. Leiden: DSWO Press.

_____. 1992. The integration of multidimensional scaling and multivariate analysis with optimal transformations of the variables. *Psychometrika*, 57: 539–565.

_____. 1993. Principal coordinates analysis with optimal transformations of the variables: Minimizing the sum of squares of the smallest eigenvalues. *British Journal of Mathematical and Statistical Psychology*, 46: 287–300.

_____. 1996. Fitting a distance model to homogeneous subsets of variables: Points of view analysis of categorical data. *Journal of Classification*, 13: 249–266.

Meulman, J. J., and W. J. Heiser. 1997. Graphical display of interaction in multiway contingency tables by use of homogeneity analysis. In: *Visual display of categorical data*, M. Greenacre and J. Blasius, eds. New York: Academic Press (in press).

Meulman, J. J., and P. Verboon. 1993. Points of view analysis revisited: Fitting multidimensional structures to optimal distance components with cluster restrictions on the variables. *Psychometrika*, 58: 7P35.

Nishisato, S. 1980. *Analysis of categorical data: Dual scaling and its applications*. Toronto: University of Toronto Press.

_____. 1994, *Elements of dual scaling: An introduction to practical data analysis*. Hillsdale: Lawrence Erlbaum Associates, Inc.

Pratt, J. W. 1987. Dividing the indivisible: Using smple symmetry to partition variance explained. In: *Proceedings of the Second International Conference in Statistics*. T. Pukkila and S. Puntanen, eds. Tampere, Finland: University of Tampere, 245–260.

Ramsay, J. O. 1989. Monotone regression splines in action. *Statistical Science*, 4: 425–441.

Rao, C. R. 1973. *Linear statistical inference and its applications*. New York: John Wiley and Sons.

_____. 1980. Matrix approximations and reduction of dimensionality in multivariate statistical analysis. In: *Multivariate Analysis*, Vol. 5, P. R. Krishnaiah, ed. Amsterdam: North-Holland.

Rosenberg, S., and M. P. Kim. 1975. The method of sorting as a data-gathering procedure in multivariate research. *Multivariate Behavioral Research*, 10: 489–502.

Roskam, E. E. 1968. *Metric analysis of ordinal data in psychology*. Voorschoten, VAM.

Shepard, R. N. 1962a. The analysis of proximities: Multidimensional scaling with an unknown distance function I. *Psychometrika*, 27: 125–140.

_____. 1962b. The analysis of proximities: Multidimensional scaling with an unknown distance function II. *Psychometrika*, 27: 219–246.

_____. 1966. Metric structures in ordinal data. *Journal of Mathematical Psychology*, 3: 287–315.

Tenenhaus, M., and F. W. Young. 1985. An analysis and synthesis of multiple correspondence analysis, optimal scaling, dual scaling, homogeneity analysis, and other methods for quantifying categorical multivariate data. *Psychometrika*, 50: 91–119.

Tucker, L. R. 1960. Intra-individual and inter-individual multidimensionality. In: *Psychological Scaling: Theory & Applications*, H. Gulliksen and S. Messick, eds. New York: Wiley.

Van der Burg, E. 1988. *Nonlinear canonical correlation and some related techniques*.

Leiden: DSWO Press.
Van der Burg, E., and J. De Leeuw. 1983. Nonlinear canonical correlation. *British Journal of Mathematical and Statistical Psychology*, 36: 54–80.
Van der Burg, E., J. De Leeuw, and R. Verdegaal. 1988. Homogeneity analysis with *k* sets of variables: An alternating least squares method with optimal scaling features. *Psychometrika*, 53: 177–197.
Van der Ham, T., J. J. Meulman, D. C. Van Strien, and H. Van Engeland. 1997. Empirically based subgrouping of eating disorders in adolescents: A longitudinal perspective. *British Journal of Psychiatry*, 170: 363–368.
Van der Kooij, A. J., and J. J. Meulman. 1997. MURALS: Multiple regression and optimal scaling using alternating least squares. In: *Softstat '97*: 99–106, F. Faulbaum and W. Bandilla, eds. Stuttgart: Gustav Fisher.
Van Rijckevorsel, J. 1987. *The application of fuzzy coding and horseshoes in multiple correspondence analysis.* Leiden: DSWO Press.
Verboon, P., and I. A. Van der Lans. 1994. Robust canonical discriminant analysis. *Psychometrika*, 59: 485–507.
Verdegaal, R. 1985. *Meer sets analyse voor kwalitatieve gegevens* (in Dutch). Leiden: Department of Data Theory, University of Leiden.
Vlek, C., and P. J. Stallen. 1981. Judging risks and benefits in the small and in the large. *Organizational Behavior and Human Performance*, 28: 235–271.
Wagenaar, W. A. 1988. *Paradoxes of gambling behaviour*. London: Lawrence Erlbaum Associates.
Winsberg, S., and J. O. Ramsay. 1983. Monotone spline transformations for dimension reduction. *Psychometrika*, 48: 575–595.
Wolter, K. M. 1985. *Introduction to variance estimation*. Berlin: Springer-Verlag.
Young, F. W. 1981. Quantitative analysis of qualitative data. *Psychometrika*, 40: 357–387.
Young, F. W., J. De Leeuw, and Y. Takane. 1976. Regression with qualitative and quantitative variables: An alternating least squares method with optimal scaling features. *Psychometrika*, 41: 505–528.
Young, F. W., Y. Takane, and J. De Leeuw. 1978. The principal components of mixed measurement level multivariate data: An alternating least squares method with optimal scaling features. *Psychometrika*, 43: 279–281.

Subject Index

active row
in Correspondence Analysis, 159
alternating least squares, 1
analyzing aggregated data
in Correspondence Analysis, 265
analyzing table data
in Correspondence Analysis, 258
ANOVA
in Categorical Regression, 24
aspect ratio, 16

biplots, 10, 168, 169–170, 177
in Categorical Principal Components Analysis, 35, 238
in Correspondence Analysis, 57, 263

canonical correlations, 136
cases
excluding from Homogeneity Analysis, 270
excluding from Nonlinear Canonical Correlation Analysis, 279
Categorical Principal Components Analysis, 7, 27, 107–130, 225–242
limitations, 227
optimal scaling level, 29
options, 226
syntax rules, 227
Categorical Regression, 7, 8, 17, 81–105, 243–254
compared to Nonlinear Canonical Correlation Analysis, 8
optimality of quantifications, 101–102
plots, 17, 90–92
residuals, 92
save, 25
statistics, 17
categorizing variables, 94–95
category plots
in Categorical Principal Components Analysis, 34
category points plots
in Categorical Principal Components Analysis, 238
category quantifications
in Categorical Principal Components Analysis, 37, 236
in Categorical Regression, 24
in Homogeneity Analysis, 63, 189–191
in Nonlinear Canonical Correlation Analysis, 46
CATREG. *See* Categorical Regression
centroid plots
in Nonlinear Canonical Correlation Analysis, 281
centroids
in Correspondence Analysis, 157
in Nonlinear Canonical Correlation Analysis, 46, 142–144, 148
chi-square distance
in Correspondence Analysis, 155, 260
coding categories, 5–7
coefficients
in Categorical Regression, 87, 96, 100
column scores
in Correspondence Analysis, 158–159, 181–182
common space
in Multidimensional Scaling, 203, 309
common space coordinates
in Multidimensional Scaling, 78
common space plots
in Multidimensional Scaling, 76, 310
component loadings
in Categorical Principal Components Analysis, 37, 114, 117, 123, 236
in Nonlinear Canonical Correlation Analysis, 46, 138
component loadings plots
in Categorical Principal Components Analysis, 36, 238
confidence statistics
in Correspondence Analysis, 56, 166–167
contributions
in Correspondence Analysis, 162–165, 173–175

correlation matrices
in Categorical Principal Components Analysis, 236
correlations
in Categorical Regression, 88–89
in Multidimensional Scaling, 78, 309
correlations plots
in Multidimensional Scaling, 76, 311
Correspondence Analysis, 7, 10–12, 49, 151–182, 255–266
compared to crosstabulation, 12, 151
compared to Homogeneity Analysis, 12
confidence statistics, 166–167
contributions, 162–165, 173–175
dimensions, 172, 259
distance measure, 260
distances, 156–157
equality constraints, 260
interaction variables, 11–12
normalization, 168–170, 177, 261
permutations, 165–166
plots, 49, 154–156, 263
profiles, 156–157
standardization, 261
statistics, 49
supplementary points, 259
correspondence tables, 10, 151, 178
crosstabulation
compared to Correspondence Analysis, 12, 151

decomposition of Stress
in Multidimensional Scaling, 202, 309
descriptive statistics
in Categorical Regression, 24
dimension reduction, 7
dimensions, 14–16
in 3-D scatterplots, 14–15
in Categorical Principal Components Analysis, 110, 115
in Correspondence Analysis, 54, 159–160, 172, 259
in Homogeneity Analysis, 185–186, 273
in Nonlinear Canonical Correlation Analysis, 135, 281–282
saving in Nonlinear Canonical Correlation Analysis, 282–283
discretization
in Categorical Principal Components Analysis, 32
in Categorical Regression, 20
discrimination measures
in Homogeneity Analysis, 63, 188–189, 195
dispersion accounted for
in Multidimensional Scaling, 309
dissimilarities, 179
distance measures
in Correspondence Analysis, 54, 260
distances
in Correspondence Analysis, 156–157, 158
in Multidimensional Scaling, 78

eigenvalues
in Homogeneity Analysis, 63, 185–186
in Nonlinear Canonical Correlation Analysis, 135, 136, 145–146
equality constraints
in Correspondence Analysis, 151, 159, 257, 260
Euclidean distance
in Correspondence Analysis, 260

fit
in Nonlinear Canonical Correlation Analysis, 46, 134–137

HOMALS. *See* Homogeneity Analysis
Homogeneity Analysis, 7, 12–13, 59, 183–196, 267–274
category quantifications, 189–191
compared to Correspondence Analysis, 12
compared to crosstabulation, 13
compared to Factor Analysis, 13
compared to Categorical Principal Components Analysis, 13
dimensions, 185–186, 273
discrimination measures, 188–189, 195
eigenvalues, 185–186
excluding cases, 270
labeling plots, 271
matrix output, 274
object scores, 186–188, 191–194, 195–196
plots, 59
saving object scores, 273
statistics, 59
value labels, 272
variable labels, 272

importance
in Categorical Regression, 89, 100
INCLUDE files, 213
individual space weights
in Multidimensional Scaling, 78, 309
individual space weights plots
in Multidimensional Scaling, 76, 310
individual spaces
in Multidimensional Scaling, 309
individual spaces plots
in Multidimensional Scaling, 76, 310
inertia
in Correspondence Analysis, 56, 157, 159
initial configuration
in Categorical Regression, 23
in Multidimensional Scaling, 75, 200
in Nonlinear Canonical Correlation Analysis, 46, 132
intercorrelations
in Categorical Regression, 86
isotropic scaling, 16
iteration criteria
in Multidimensional Scaling, 75
iteration history
in Categorical Principal Components Analysis, 37, 236
in Multidimensional Scaling, 78, 308

joint category plots
in Categorical Principal Components Analysis, 34
joint category points plots
in Categorical Principal Components Analysis, 239

linear canonical correlation analysis
compared to Nonlinear Canonical Correlation Analysis, 131
linear regression, 82–84
coefficients, 82–83
model fit, 82
residuals, 83
loss
in Nonlinear Canonical Correlation Analysis, 134–137

mass
in Correspondence Analysis, 157
matrix output
in Homogeneity Analysis, 274
in Nonlinear Canonical Correlation Analysis, 283
measurement level
compared to optimal scaling level, 2
missing values, 5
in Categorical Principal Components Analysis, 33, 232
in Categorical Regression, 22
model fit
in Categorical Principal Components Analysis, 122
in Categorical Regression, 87, 96, 100, 102, 104
monotone spline
in Multidimensional Scaling, 304
multicollinearity
in Categorical Regression, 86, 90
Multidimensional Scaling, 7, 65, 295–312
limitations, 297
model
options, 75, 296
output, 78
plots, 65, 76, 78
restrictions, 74
statistics, 65
multiple category coordinates
in Nonlinear Canonical Correlation Analysis, 140–142
multiple correlation
in Nonlinear Canonical Correlation Analysis, 135
multiple correspondence analysis, 12
multiple fit
in Nonlinear Canonical Correlation Analysis, 137, 146
multiple *R*
in Categorical Regression, 24

Nonlinear Canonical Correlation Analysis, 7, 9–10, 41, 131–149, 275–283
centroid plots, 281
centroids, 142–144, 148
dimensions, 281–282
excluding cases, 279
matrix output, 283
optimal scaling level, 278

plots, 41
saving dimensions, 282–283
saving object scores, 282
statistics, 41
transformation plots, 146–147, 281
value labels, 281–282
variable labels, 281–282
Nonlinear Principal Components Analysis, 9
compared to Homogeneity Analysis, 9
normalization, 155–156, 168–170
column principal, 168
custom, 170
in Correspondence Analysis, 54, 261
principal, 168, 172
row principal, 155–156, 168
symmetrical, 168, 177
normalized raw Stress
in Multidimensional Scaling, 309
numerical scaling level
in Multidimensional Scaling, 303

object points plots
in Categorical Principal Components Analysis, 35, 237
object scores
in Categorical Principal Components Analysis, 37, 112, 116, 124, 236
in Homogeneity Analysis, 63, 186–188, 191–194, 195–196
in Nonlinear Canonical Correlation Analysis, 46, 133
saving in Homogeneity Analysis, 273
saving in Nonlinear Canonical Correlation Analysis, 282
optimal scaling level, 2
choosing, 84–85, 95–100
compared to measurement level, 2
in Categorical Principal Components Analysis, 29, 229
in Nonlinear Canonical Correlation Analysis, 278
multiple nominal, 12
nominal, 2, 3, 5, 10, 91
numerical, 2, 3, 6–7, 90, 303
ordinal, 2, 3, 5, 91, 100, 303
ordinal scaling level
in Multidimensional Scaling, 303
outliers
in Nonlinear Canonical Correlation Analysis, 133

OVERALS. *See* Nonlinear Canonical Correlation Analysis

perceptual mapping, 7
permutations
in Correspondence Analysis, 165–166
plots
in Categorical Regression, 26
in Correspondence Analysis, 57, 263
in Homogeneity Analysis, 63
in Multidimensional Scaling, 76, 78
in Nonlinear Canonical Correlation Analysis, 46
PRINCALS. *See* Nonlinear Principal Components Analysis
profiles
in Correspondence Analysis, 156–157
projected centroids
in Nonlinear Canonical Correlation Analysis, 142–144, 148
projected centroids plots
in Categorical Principal Components Analysis, 34, 239
proximities, 197

quantifications
in Categorical Principal Components Analysis, 111
in Nonlinear Canonical Correlation Analysis, 141
optimality of, 101–102

regression coefficients
in Categorical Regression, 24
Regression with Optimal Scaling. *See* Categorical Regression
relaxed updates
in Multidimensional Scaling, 75
residuals
in Categorical Regression, 92
residuals plots
in Categorical Principal Components Analysis, 238
in Multidimensional Scaling, 310
restrictions
in Multidimensional Scaling, 74

row scores
in Correspondence Analysis, 158–159, 180–182
row scores plots
in Correspondence Analysis, 154–156

scatterplot matrices, 15–16, 116
scree plot, 198
in Multidimensional Scaling, 199
similarities, 179
single category coordinates
in Nonlinear Canonical Correlation Analysis, 140–142
single fit
in Nonlinear Canonical Correlation Analysis, 137, 145–146
single loss
in Nonlinear Canonical Correlation Analysis, 137
singular values, 160
spline
in Multidimensional Scaling, 304
standardization
in Correspondence Analysis, 54, 151, 261
standardized regression coefficients
in Categorical Regression, 87, 96, 100
strain, 13
Stress
in Multidimensional Scaling, 201
Stress decomposition
in Multidimensional Scaling, 202
Stress measures
in Multidimensional Scaling, 78, 309
Stress plots
in Multidimensional Scaling, 76, 310
sunflower plots, 187
supplementary objects
in Categorical Regression, 23
supplementary points
in Correspondence Analysis, 152, 159, 160–162, 259
syntax
diagrams, 211–212
INCLUDE files, 213
rules, 211–212

tolerance
in Categorical Regression, 90
transformation plots, 4
creating, 4
in Categorical Principal Components Analysis, 34, 238
in Multidimensional Scaling, 76, 310
in Nonlinear Canonical Correlation Analysis, 138–140, 146–147, 281
nominal scaling level, 92, 97, 98, 99, 105, 139, 146, 147, 149
numerical scaling level, 91
ordinal scaling level, 101, 121–122, 140
transformation type. *See* optimal scaling level
transformations
effects of, 102–105
transformed independent variables
in Multidimensional Scaling, 78
transformed proximities
in Multidimensional Scaling, 78, 309
triplots
in Categorical Principal Components Analysis, 35, 239
Tucker's coefficient of congruence
in Multidimensional Scaling, 309

value labels
as point labels in Homogeneity Analysis, 271
as point labels in Nonlinear Canonical Correlation Analysis, 281–282
in Homogeneity Analysis, 272
variable labels
as plot labels in Homogeneity Analysis, 271
as plot labels in Nonlinear Canonical Correlation Analysis, 281–282
in Homogeneity Analysis, 272
variable weight
in Categorical Principal Components Analysis, 29, 229
variance
in Homogeneity Analysis, 188
variance accounted for
in Categorical Principal Components Analysis, 37, 236

weights
in Nonlinear Canonical Correlation Analysis, 46

Syntax Index

ACCELERATION (subcommand)
PROXSCAL command, 307

ACTIVE (keyword)
CATPCA command, 232, 233

aggregate data
ANACOR command, 222–223

ALL (keyword)
ANACOR command, 217–218, 221
CORRESPONDENCE command, 258
HOMALS command, 271, 272
OVERALS command, 281–282
PRINCALS command, 290

ANACOR (command), 215–223
aggregate data, 222–223
DIMENSION subcommand, 218
MATRIX subcommand, 221–222
NORMALIZATION subcommand, 218–219
PLOT subcommand, 220–221
PRINT subcommand, 219–220
TABLE subcommand, 216–218
value labels, 220
VARIANCES subcommand, 219
with WEIGHT command, 222–223

ANALYSIS (subcommand)
CATPCA command, 229
CATREG command, 247
HOMALS command, 270
OVERALS command, 277–278
PRINCALS command, 288–289
with SETS subcommand, 278
with VARIABLES subcommand, 270, 277–278

ANOVA (keyword)
CATREG command, 251

APPROX (keyword)
CATPCA command, 240, 242

AUTORECODE (command)
with HOMALS command, 268, 269
with OVERALS command, 276–277
with PRINCALS command, 286–287, 287–288

BIPLOT (keyword)
CATPCA command, 238, 239
CORRESPONDENCE command, 263

BOTH (keyword)
PROXSCAL command, 301

CANONICAL (keyword)
ANACOR command, 218–219

CATEGORY (keyword)
CATPCA command, 238

CATPCA (command), 225
ANALYSIS subcommand, 229
CONFIGURATION subcommand, 233
CRITITER subcommand, 235
DIMENSION subcommand, 234
DISCRETIZATION subcommand, 231
MAXITER subcommand, 235
MISSING subcommand, 232
NORMALIZATION subcommand, 234
OPRINCIPAL keyword, 234
OUTFILE subcommand, 242
PLOT subcommand, 237
PRINT subcommand, 235
SAVE subcommand, 240
SUPPLEMENTARY subcommand, 233
SYMMETRICAL keyword, 234
VARIABLES subcommand, 229
VPRINCIPAL keyword, 234

CATREG (command), 243–254
ANALYSIS subcommand, 247
CRITITER subcommand, 250
DISCRETIZATION subcommand, 248
INITIAL subcommand, 250
MAXITER subcommand, 250
MISSING subcommand, 249
OUTFILE subcommand, 254
PLOT subcommand, 252
PRINT subcommand, 251
SUPPLEMENTARY subcommand, 250
VARIABLES subcommand, 246, 253

CCONF (keyword)
CORRESPONDENCE command, 262

CENTR (keyword)
CATPCA command, 239
with BIPLOT keyword, 239

CENTROID (keyword)
OVERALS command, 280, 280–282
CHISQ (keyword)
CORRESPONDENCE command, 260
CMEAN (keyword)
CORRESPONDENCE command, 261
COEFF (keyword)
CATREG command, 251
COLUMNS (keyword)
ANACOR command, 219, 220–221
COMMON (keyword)
PROXSCAL command, 309, 310, 311
CONDITION (subcommand)
PROXSCAL command, 303
CONFIGURATION (subcommand)
CATPCA command, 233
CONTRIBUTIONS (keyword)
ANACOR command, 220
CONVERGENCE (subcommand)
HOMALS command, 271
OVERALS command, 280
PRINCALS command, 290
COORDINATES (keyword)
PROXSCAL command, 305
CORR (keyword)
CATPCA command, 236
CATREG command, 251
CORRELATION (keyword)
PRINCALS command, 290
CORRELATIONS (keyword)
PROXSCAL command, 309, 311
CORRESPONDENCE (command), 255–266
DIMENSION subcommand, 259
EQUAL subcommand, 260
MEASURE subcommand, 260
NORMALIZATION subcommand, 261
OUTFILE subcommand, 264
PLOT subcommand, 263
PRINT subcommand, 262
STANDARDIZE subcommand, 261
SUPPLEMENTARY subcommand, 259
TABLE subcommand, 257
CPOINTS (keyword)
CORRESPONDENCE command, 262, 263
CPRINCIPAL (keyword)
ANACOR command, 219
CORRESPONDENCE command, 262
CPROFILES (keyword)
CORRESPONDENCE command, 262
CRITERIA (subcommand)
PROXSCAL command, 307
CRITITER (subcommand)
CATPCA command, 235
CATREG command, 250
CSUM (keyword)
CORRESPONDENCE command, 261

DECOMPOSITION (keyword)
PROXSCAL command, 309
DEFAULT (keyword)
ANACOR command, 220, 220–221
CORRESPONDENCE command, 263
HOMALS command, 271, 272
OVERALS command, 280, 281–282
PRINCALS command, 290, 291–292
DEGREE (keyword)
CATPCA command, 231
CATREG command, 248
PROXSCAL command, 304, 306
with SPLINE keyword, 304, 306
DESCRIP (keyword)
CATPCA command, 235
CATREG command, 251
DIFFSTRESS (keyword)
PROXSCAL command, 308
DIM variable
ANACOR command, 222
HOMALS command, 274
OVERALS command, 283
PRINCALS command, 294
DIMENSION (subcommand)
ANACOR command, 218
CATPCA command, 234
CORRESPONDENCE command, 259
HOMALS command, 270
OVERALS command, 279
PRINCALS command, 289
with SAVE subcommand, 274, 282, 292–293
DIMENSIONS (keyword)
PROXSCAL command, 307
DIM*n* variable
CORRESPONDENCE command, 265
DISCRDATA (keyword)
CATREG command, 254
DISCRDATA (keyword)
CATPCA command, 242

DISCRETIZATION (subcommand)
CATPCA command, 231
CATREG command, 248
DISCRIM (keyword)
HOMALS command, 271, 272
DISSIMILARITIES (keyword)
PROXSCAL command, 304
DISTANCES (keyword)
PROXSCAL command, 309, 311
DISTR (keyword)
CATPCA command, 232
CATREG command, 249

EIGEN (keyword)
HOMALS command, 271
PRINCALS command, 290
EQINTV (keyword)
CATPCA command, 231
CATREG command, 249
with GROUPING keyword, 231
EQUAL (subcommand)
CORRESPONDENCE command, 260
EUCLID (keyword)
CORRESPONDENCE command, 260
EXTRACAT (keyword)
CATPCA command, 232, 233
CATREG command, 249
with ACTIVE keyword, 233
with PASSIVE keyword, 232

FIRST (keyword)
PROXSCAL command, 306
with VARIABLES keyword, 306
FIT (keyword)
OVERALS command, 280
FIXED (keyword)
CATPCA command, 233
FREQ (keyword)
HOMALS command, 271
OVERALS command, 280
PRINCALS command, 290

GENERALIZED (keyword)
PROXSCAL command, 305
GROUPING (keyword)
CATPCA command, 231
CATREG command, 248

HISTORY (keyword)
CATPCA command, 236
CATREG command, 251
HOMALS command, 271
OVERALS command, 280
PRINCALS command, 290
PROXSCAL command, 308
HOMALS (command), 267–274
ANALYSIS subcommand, 270
compared to OVERALS, 278
CONVERGENCE subcommand, 271
DIMENSION subcommand, 270
MATRIX subcommand, 274
MAXITER subcommand, 271
NOBSERVATIONS subcommand, 270
PLOT subcommand, 271
PRINT subcommand, 271
SAVE subcommand, 273
value labels, 272
variable labels, 272
VARIABLES subcommand, 269
with AUTORECODE command, 268, 269
with RECODE command, 268

IDENTITY (keyword)
PROXSCAL command, 305
IN (keyword)
PROXSCAL command, 312
INDEPENDENT (keyword)
CATPCA command, 234
INDIVIDUAL (keyword)
PROXSCAL command, 309, 310
INITIAL (keyword)
CATPCA command, 233
INITIAL (subcommand)
CATREG command, 250
OVERALS command, 279
PROXSCAL command, 301
INKNOT (keyword)
CATPCA command, 231
CATREG command, 248

PROXSCAL command, 304, 306
with SPLINE keyword, 304, 306
INPUT (keyword)
PROXSCAL command, 308
INTERVAL (keyword)
PROXSCAL command, 303, 306
with VARIABLES keyword, 306

JOINT (keyword)
ANACOR command, 220–221
JOINTCAT (keyword)
CATPCA command, 239

KEEPTIES (keyword)
PROXSCAL command, 306
with ORDINAL keyword, 306

LEVEL (keyword)
CATPCA command, 229, 230
CATREG command, 247
LEVEL variable
ANACOR command, 221–222
HOMALS command, 274
OVERALS command, 283
PRINCALS command, 293
LEVEL_ variable
CORRESPONDENCE command, 264, 265
LISTWISE (keyword)
CATPCA command, 232
CATREG command, 249
LOADING (keyword)
CATPCA command, 236, 238, 239
with BIPLOT keyword, 239
LOADINGS (keyword)
OVERALS command, 280–282
PRINCALS command, 290, 290–292
LOWER (keyword)
PROXSCAL command, 301

MATRIX (keyword)
PROXSCAL command, 303
MATRIX (subcommand)
ANACOR command, 221–222
HOMALS command, 274
OVERALS command, 283
PRINCALS command, 293–294
PROXSCAL command, 312
with SAVE subcommand, 274, 282, 293
MAX (keyword)
ANACOR command, 221
CORRESPONDENCE command, 264
HOMALS command, 273
OVERALS command, 281–282
PRINCALS command, 292
MAXITER (keyword)
PROXSCAL command, 307
MAXITER (subcommand)
CATPCA command, 235
CATREG command, 250
HOMALS command, 271
OVERALS command, 279
PRINCALS command, 289
MEASURE (subcommand)
CORRESPONDENCE command, 260
MINSTRESS (keyword)
PROXSCAL command, 308
MISSING (subcommand)
CATPCA command, 232
CATREG command, 249
missing values
with OVERALS command, 277
with PRINCALS command, 286
MNOM (keyword)
CATPCA command, 230
OVERALS command, 278
PRINCALS command, 288
MODEIMPU (keyword)
CATPCA command, 232, 233
CATREG command, 249
with ACTIVE keyword, 233
with PASSIVE keyword, 232
MODEL (subcommand)
PROXSCAL command, 305
MULTIPLYING (keyword)
CATPCA command, 231
CATREG command, 248

NCAT (keyword)
CATPCA command, 231
CATREG command, 249
with GROUPING keyword, 231
NDIM (keyword)
ANACOR command, 220–221
CORRESPONDENCE command, 263
HOMALS command, 273

OVERALS command, 281–282
PRINCALS command, 292
NOBSERVATIONS (subcommand)
HOMALS command, 270
OVERALS command, 279
PRINCALS command, 289
NOMI (keyword)
CATPCA command, 230
CATREG command, 247
NOMINAL (keyword)
PROXSCAL command, 306
with VARIABLES keyword, 306
NONE (keyword)
ANACOR command, 220, 220–221
CATPCA command, 236, 239
CATREG command, 252
CORRESPONDENCE command, 263
HOMALS command, 271, 272
OVERALS command, 280, 281–282
PRINCALS command, 290, 291–292
PROXSCAL command, 307, 308, 310
NORMAL (keyword)
CATPCA command, 232
CATREG command, 249
with DISTR keyword, 232
NORMALIZATION (subcommand)
ANACOR command, 218–219
CATPCA command, 234
CORRESPONDENCE command, 261
with PLOT subcommand, 220
NUME (keyword)
CATPCA command, 230
CATREG command, 248
OVERALS command, 278
PRINCALS command, 289
NUMERICAL (keyword)
CATREG command, 250
OVERALS command, 279

OBJECT (keyword)
CATPCA command, 233, 236, 237, 240, 242
CATREG command, 250
HOMALS command, 271, 272
OVERALS command, 280, 280–282
PRINCALS command, 290, 290–292
OCORR (keyword)
CATPCA command, 236
CATREG command, 251
OPRINCIPAL (keyword)
CATPCA command, 234
ORDI (keyword)
CATPCA command, 230
CATREG command, 247
OVERALS command, 278
PRINCALS command, 288
ORDINAL (keyword)
PROXSCAL command, 303, 306
with VARIABLES keyword, 306
OUT (keyword)
ANACOR command, 221
HOMALS command, 274
OUTFILE (subcommand)
CATPCA command, 242
CATREG command, 254
CORRESPONDENCE command, 264
PROXSCAL command, 311
OVERALS (command), 275–283
active variables, 278
ANALYSIS subcommand, 277–278
compared to HOMALS, 278
compared to PRINCALS, 278
CONVERGENCE subcommand, 280
DIMENSION subcommand, 279
INITIAL subcommand, 279
MATRIX subcommand, 283
MAXITER subcommand, 279
NOBSERVATIONS subcommand, 279
passive variables, 277–278
PLOT subcommand, 280–282
PRINT subcommand, 280
SAVE subcommand, 282–283
SETS subcommand, 278
value labels, 281–282
variable labels, 281–282
VARIABLES subcommand, 277
with AUTORECODE command, 276–277
with RECODE command, 276–277

PASSIVE (keyword)
CATPCA command, 232
PERMUTATION (keyword)
ANACOR command, 220
CORRESPONDENCE command, 262
PLOT (subcommand)
ANACOR command, 220–221
CATPCA command, 237

CATREG command, 252
CORRESPONDENCE command, 263
HOMALS command, 271
OVERALS command, 280–282
PRINCALS command, 290–292
PROXSCAL command, 310
with NORMALIZATION subcommand, 220
PRED (keyword)
CATREG command, 253
PRINCALS (command), 285–294
ANALYSIS subcommand, 288–289
compared to OVERALS, 278
DIMENSION subcommand, 289
MATRIX subcommand, 293–294
MAXITER subcommand, 289
NOBSERVATIONS subcommand, 289
PLOT subcommand, 290–292
PRINT subcommand, 290
SAVE subcommand, 292–293
value labels, 291–292
variable labels, 291–292
VARIABLES subcommand, 287–288
with AUTORECODE command, 286–287, 287–288
with RECODE command, 286–287, 287–288
PRINCIPAL (keyword)
ANACOR command, 218–219
CORRESPONDENCE command, 261
PRINT (subcommand)
ANACOR command, 219–220
CATPCA command, 235
CATREG command, 251
CORRESPONDENCE command, 262
HOMALS command, 271
OVERALS command, 280
PRINCALS command, 290
PROXSCAL command, 308
PROFILES (keyword)
ANACOR command, 219
PROJCENTR (keyword)
CATPCA command, 239
PROXIMITIES (subcommand)
PROXSCAL command, 304
PROXSCAL (command), 295–312
ACCELERATION subcommand, 307
CONDITION subcommand, 303
CRITERIA subcommand, 307
INITIAL subcommand, 301
MATRIX subcommand, 312
OUTFILE subcommand, 311
PLOT subcommand, 310
PRINT subcommand, 308
PROXIMITIES subcommand, 304
RESTRICTIONS subcommand, 305
SHAPE subcommand, 301
TABLE subcommand, 298
TRANSFORMATION subcommand, 303
WEIGHTS subcommand, 302

QUANT (keyword)
CATPCA command, 236
CATREG command, 251
HOMALS command, 271, 272
OVERALS command, 280, 280–282
PRINCALS command, 290, 290–292

R (keyword)
CATREG command, 251
RANDOM (keyword)
CATREG command, 250
OVERALS command, 279
PROXSCAL command, 302, 308
RANKING (keyword)
CATPCA command, 231
CATREG command, 248
RATIO (keyword)
PROXSCAL command, 303
RCMEAN (keyword)
CORRESPONDENCE command, 261
RCONF (keyword)
CORRESPONDENCE command, 262
RECODE (command)
with HOMALS command, 268
with OVERALS command, 276–277
with PRINCALS command, 286–287, 287–288
REDUCED (keyword)
PROXSCAL command, 305
RES (keyword)
CATREG command, 253
RESID (keyword)
CATPCA command, 238
CATREG command, 252
RESIDUALS (keyword)
PROXSCAL command, 310
RESTRICTIONS (subcommand)
PROXSCAL command, 305
RMEAN (keyword)
CORRESPONDENCE command, 261

ROWS (keyword)
ANACOR command, 219, 220–221
ROWTYPE_ variable
ANACOR command, 221–222
CORRESPONDENCE command, 264, 265
HOMALS command, 274
OVERALS command, 283
PRINCALS command, 293
RPOINTS (keyword)
CORRESPONDENCE command, 262, 263
RPRINCIPAL (keyword)
ANACOR command, 219
CORRESPONDENCE command, 261
RPROFILES (keyword)
CORRESPONDENCE command, 262
RSUM (keyword)
CORRESPONDENCE command, 261

SAVE (subcommand)
CATPCA command, 240
HOMALS command, 273
OVERALS command, 282–283
PRINCALS command, 292–293
with DIMENSION subcommand, 274, 282, 292–293
with MATRIX subcommand, 274, 282, 293
SCORE (keyword)
ANACOR command, 221–222
CORRESPONDENCE command, 264
SCORE variable
ANACOR command, 222
SCORE_ variable
CORRESPONDENCE command, 265
SCORES (keyword)
ANACOR command, 219
SET_ variable
OVERALS command, 283
SETS (subcommand)
OVERALS command, 278
with ANALYSIS subcommand, 278
SHAPE (subcommand)
PROXSCAL command, 301
SIMILARITIES (keyword)
PROXSCAL command, 304
SIMPLEX (keyword)
PROXSCAL command, 301
SINGULAR (keyword)
ANACOR command, 219
SNOM (keyword)
OVERALS command, 278
PRINCALS command, 288
SPLINE (keyword)
PROXSCAL command, 304, 306
with VARIABLES keyword, 306
SPNOM (keyword)
CATPCA command, 230, 231
CATREG command, 247
SPORD (keyword)
CATPCA command, 230, 231
CATREG command, 247
STANDARDIZE (subcommand)
CORRESPONDENCE command, 261
STRESS (keyword)
PROXSCAL command, 309, 310
SUPPLEMENTARY (subcommand)
CATPCA command, 233
CATREG command, 250
CORRESPONDENCE command, 259
SYMMETRICAL (keyword)
CATPCA command, 234
CORRESPONDENCE command, 261

TABLE (keyword)
ANACOR command, 219
CORRESPONDENCE command, 262
TABLE (subcommand)
ANACOR command, 216–218
casewise data, 217
CORRESPONDENCE command, 257
PROXSCAL command, 298
table data, 217–218
TORGERSON (keyword)
PROXSCAL command, 302
TRANS (keyword)
CATPCA command, 238
CATREG command, 252
OVERALS command, 280–282
TRANSFORMATION (keyword)
PROXSCAL command, 309, 311
TRANSFORMATION (subcommand)
PROXSCAL command, 303
TRANSFORMATIONS (keyword)
PROXSCAL command, 310
TRCOLUMNS (keyword)
ANACOR command, 220–221
CORRESPONDENCE command, 263

TRDATA (keyword)
CATPCA command, 240, 242
CATREG command, 253, 254
TRIPLOT (keyword)
CATPCA command, 239
TRROWS (keyword)
ANACOR command, 220–221
CORRESPONDENCE command, 263

UNCONDITIONAL (keyword)
PROXSCAL command, 303
UNIFORM (keyword)
CATPCA command, 232
CATREG command, 249
with DISTR keyword, 232
UNTIE (keyword)
PROXSCAL command, 306
with ORDINAL keyword, 306
UPPER (keyword)
PROXSCAL command, 301

VAF (keyword)
CATPCA command, 236
value labels
ANACOR command, 220
VARIABLE (keyword)
CATPCA command, 233
VARIABLES (keyword)
PROXSCAL command, 306, 309, 310, 312
VARIABLES (subcommand)
CATPCA command, 229
CATREG command, 246, 253
HOMALS command, 269
OVERALS command, 277
PRINCALS command, 287–288
with ANALYSIS subcommand, 270, 277–278
VARIANCE (keyword)
ANACOR command, 222
CORRESPONDENCE command, 264
VARIANCES (subcommand)
ANACOR command, 219
VARNAME_ variable
ANACOR command, 222
CORRESPONDENCE command, 264, 265
HOMALS command, 274
OVERALS command, 283
PRINCALS command, 294
VARTYPE_ variable
OVERALS command, 283
PRINCALS command, 294
VPRINCIPAL (keyword)
CATPCA command, 234

WEIGHT (command)
with ANACOR command, 222–223
with CORRESPONDENCE command, 265
WEIGHT (keyword)
CATPCA command, 229
WEIGHTED (keyword)
PROXSCAL command, 305
WEIGHTS (keyword)
OVERALS command, 280
PROXSCAL command, 309, 310, 311
WEIGHTS (subcommand)
PROXSCAL command, 302