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.

```
Example:

ANACOR  TABLE=MENTAL(1,4) BY SES(1,6)
 /PRINT=SCORES CONTRIBUTIONS
 /PLOT=ROWS COLUMNS.
```

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. 8).

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 5 levels and the other has 4 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 Base 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 below, 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 equals 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

Annotated Example

This example of ANACOR uses the crosstabular data on staff category and smoking habits from Greenacre (1984).

```
DATA LIST FREE /STAFF SMOKE FREQ.
WEIGHT BY FREQ.
VALUE LABELS
 STAFF 1 'SENIOR MANAGERS' 2 'JUNIOR MANAGERS'
 3 'SENIOR EMPLOYEES' 4 'JUNIOR EMPLOYEES' 5 'SECRETARIES'
 /SMOKE 1 'none' 2 'light' 3 'medium' 4 'heavy'.
BEGIN DATA
1 1  4   1 2  2   1 3  3   1 4  2
2 1  4   2 2  3   2 3  7   2 4  4
3 1 25   3 2 10   3 3 12   3 4  4
4 1 18   4 2 24   4 3 33   4 4 13
5 1 10   5 2  6   5 3  7   5 4  2
END DATA.
ANACOR TABLE=STAFF(1,5) BY SMOKE(1,4)
 /DIMENSION=2
 /NORMALIZATION=PRINCIPAL
 /PRINT=SCORES
 /PLOT TRROWS TRCOLUMNS ROWS(15) COLUMNS(15).
```

- The DATA LIST command specifies three variables in freefield format. The data from this example are taken from a crosstabulation. Variable *STAFF* represents the category values of the row variable, variable *SMOKE* represents the category values of the column variable, and variable *FREQ* is the cell count.
- The WEIGHT command counts each "case" as many times as the value of the cell count variable *FREQ*.

- The VALUE LABELS command provides descriptive labels for each category of variables *STAFF* and *SMOKE*. Since value labels are used in plots generated by ANACOR, it is usually helpful to assign short, unique labels. It is also a good idea to try to distinguish value labels for the different variables. In this example, the value labels for *STAFF* appear in upper case, and the labels for *SMOKE* are in lower case to further distinguish between variables on the plots.
- The TABLE subcommand on ANACOR specifies the row and column variables, along with their value ranges in parentheses.
- The DIMENSION subcommand tells ANACOR to compute two dimensions. (This is the default if the DIMENSION subcommand is omitted.)
- The NORMALIZATION subcommand indicates that the PRINCIPAL method should be used. The distances between categories of the row variable and between categories of the column variable are approximations of chi-square distances. (The distances between variables, however, are not.)
- The PRINT subcommand requests a display of marginal proportions and scores for each row and column. This subcommand produces the output shown in Figure 2. The singular values (eigenvalues) shown in Figure 1 are displayed automatically, even if you specify the keyword NONE on the PRINT subcommand.
- The PLOT subcommand requests plots of values calculated by ANACOR. The keywords TRROWS and TRCOLUMNS produce the plots of transformed row and column scores shown in Figure 4 and Figure 5. The keywords ROWS and COLUMNS produce the two-dimensional plots of row scores and column scores shown in Figure 6 and Figure 7. (With the principal normalization method, combined plots are not appropriate, and the keyword JOINT is not available for the PLOT subcommand. Joint plots are available with all other normalization techniques.) The optional integer values in parentheses after the keywords ROWS and COLUMNS specify the number of characters from the value labels that should be used on the plots.

The singular values shown in Figure 1 can be interpreted as correlations between the row and column scores displayed in Figure 2. For example, the singular value for dimension 1, which is 0.273, is the correlation between the row and column scores for dimension 1. If you replace the category values with the scores and calculate the Pearson correlation coefficient, the resulting *r* value is the same as the singular value (see Figure 3).

Figure 1 Singular values

Dimension	Singular Value	Inertia	Proportion Explained	Cumulative Proportion
1	.27342	.07476	.878	.878
2	.10009	.01002	.118	.995
3	.02034	.00041	.005	1.000
Total		.08519	1.000	1.000

Figure 2 Row and column scores

```
Row Scores:

STAFF          Marginal    Dim
                Profile       1        2

   1 SENIOR M     .057    -.066     .194
   2 JUNIOR M     .093     .259     .243
   3 SENIOR E     .264    -.381     .011
   4 JUNIOR E     .456     .233    -.058
   5 SECRETAR     .130    -.201    -.079

Column Scores:

SMOKE          Marginal    Dim
                Profile       1        2

   1 none         .316    -.393     .030
   2 light        .233     .099    -.141
   3 medium       .321     .196    -.007
   4 heavy        .130     .294     .198
```

Figure 3 Correlation between row and column scores

```
RECODE
 STAFF (1=-.066) (2=.259) (3=-.381) (4=.233) (5=-.201) INTO STAFFP
 /SMOKE (1=-.393) (2=.099) (3=.196) (4=.294) INTO SMOKEP.
CORRELATION STAFFP SMOKEP.


                        - -  Correlation Coefficients  - -

              STAFFP      SMOKEP

STAFFP        1.0000       .2734
             (  193)     (  193)
             P= .        P= .000

SMOKEP         .2734      1.0000
             (  193)     (  193)
             P= .000     P= .

(Coefficient / (Cases) / 2-tailed Significance)

" . " is printed if a coefficient cannot be computed
```

Figure 4 and Figure 5 show the separate plots of transformed row and column scores. From Figure 4, we can see that the first dimension differentiates between juniors—both managers and employees—and other groups. The second dimension separates managers from employees and secretaries. In Figure 5, dimension 1 distinguishes smokers from nonsmokers.

Figure 4 Plots of transformed row scores

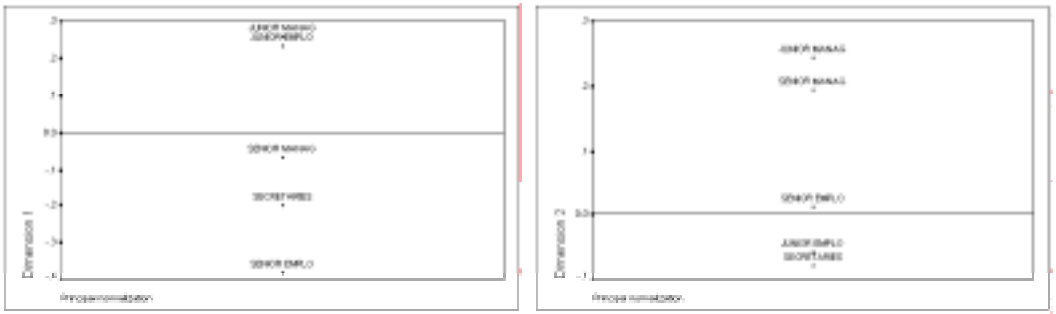
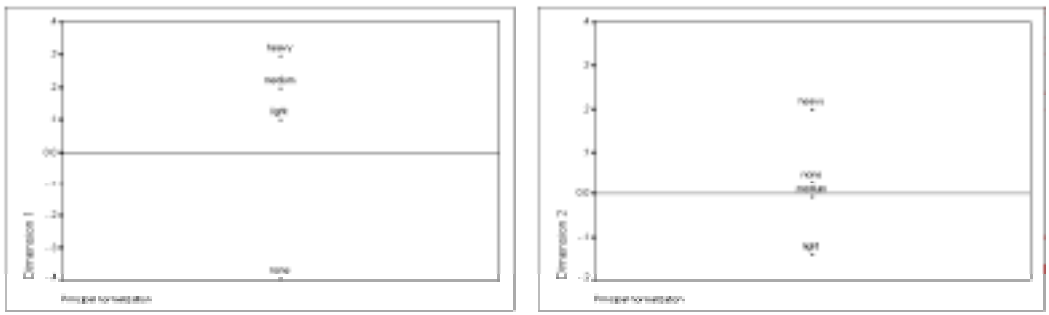


Figure 5 Plots of transformed column scores



In Figure 6, the two dimensions are combined. The plot of row scores shows that dimension 1 separates seniors from juniors (both employees and managers) and dimension 2 distinguishes between managers and employees. The plot of column scores (see Figure 7) shows that dimension 1 separates smokers from nonsmokers, and dimension 2 differentiates between categories of smokers.

Figure 6 Two-dimensional plot of row scores

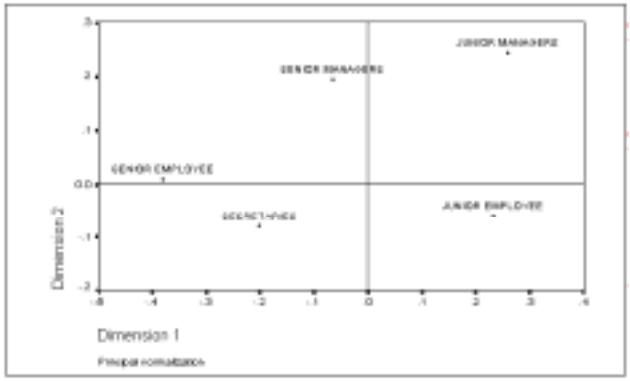


Figure 7 Two-dimensional plot of column scores

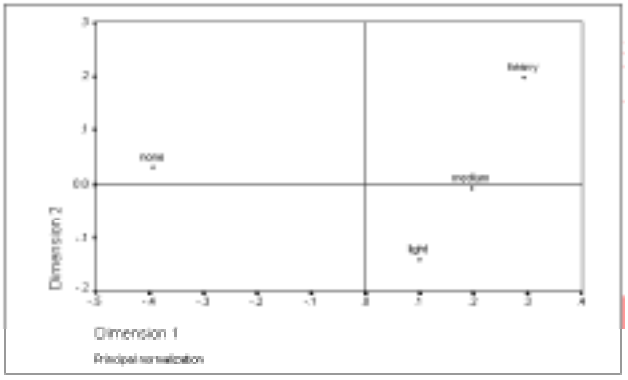
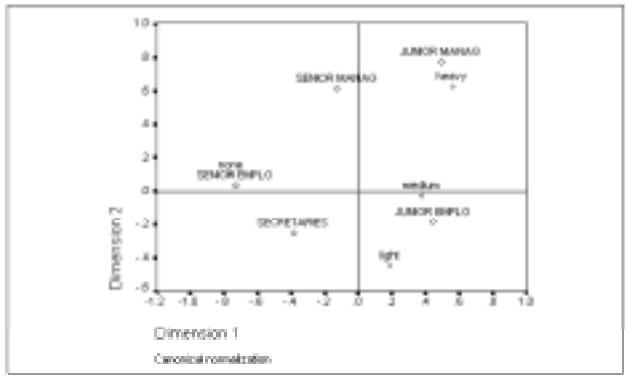


Figure 8 shows the joint plot of row and column scores produced with canonical normalization. This plot shows the relationship between the two variables. From this plot, we can see that heavy smoking is most common among managers (both junior and senior) and nonsmoking is most prevalent among senior employees. Junior employees tend to be either medium or light smokers, and secretaries tend to be either nonsmokers or light to medium smokers.

Figure 8 Joint plot with canonical normalization



CATREG

```
CATREG [VARIABLES =] varlist (max)

/ANALYSIS depvar[({ORDI**})] WITH indvarlist[({ORDI**})]

{NOMI  }              {NOMI  }

{NUME  }              {NUME  }

[/INITIAL = {NUMERICAL**}]

{RANDOM    }

[/MAXITER = {100**  }]

{value }

[/CONVERGENCE = {.00001**}]

{value }

[/MISSING = {LISTWISE**}]

{MODEIMPU  }

[/PRINT = [R**] [COEFF**] [FREQ**] [HISTORY] [ANOVA**]

[CORR] [OCORR] [QUANT] [TRANS[(value)]]
[DEFAULT] [NONE]]

[/PLOT = QUANT({varlist}) [value]]

{ALL }

[/SAVE = [ {TRANS**  } ]]

{rootname}

[/OUTFILE = `filename'].
```

** Default if subcommand or keyword is omitted.

Overview

CATREG (*Cat*egorical *reg*ression 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 of mixed optimal scaling levels and no distributional assumptions about the variables are made.

Options

Transformation type. You can specify the transformation type (nominal, ordinal, numerical) at which you want to analyze each variable.

Initial solution. You can specify the kind of initial solution through the INITIAL subcommand.

Tuning the algorithm. You can control the values of algorithm-tuning parameters with the MAXITER and CONVERGENCE subcommands.

Missing data. You can specify the treatment of missing data with the MISSING subcommand.

Display output. You can request optional output through the PRINT subcommand.

Quantification plot. For each variable, you can request a plot of the quantifications against the category numbers.

Writing transformed data. You can write the quantifications to an outfile for use in further analyses.

Saving scores. You can save the transformed variables in the working data file.

Basic Specification

- The basic specification is the command CATREG with the VARIABLES and ANALYSIS subcommands. By default, CATREG assumes all variables are ordinal and displays tables listing fit statistics, regression coefficients, descriptive statistics, and analysis of variance results.

Subcommand Order

- The VARIABLES and ANALYSIS subcommands must appear.
- The VARIABLES subcommand must be the first subcommand specified.
- Other subcommands, if specified, can be in any order.

Operations

- 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.

- If more than one dependent variable is specified in the ANALYSIS subcommand, CATREG is not executed.
- The words WITH and TO may not be used as variable names.
- If a subcommand is specified more than once, the last one is executed.
- CATREG treats every positive integer as a valid category indicator. 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 continuous variables, recoding is not recommended, since the characteristic of equal intervals in the data will not be maintained. (See the *SPSS Base Syntax Reference Guide* or the *SPSS Base User's Guide* for more information on AUTORECODE and RECODE).

Limitations

- String variables are not allowed; use AUTORECODE to recode nominal string variables.
- CATREG-missing values for a variable are values less than 1 or greater than or equal to *max*+1, where *max* is the largest valid category number defined for the variable on the VARIABLES subcommand. If one of the category values of a variable has been coded 0 or some negative value and you want to treat it as a valid category, use AUTORECODE or RECODE to recode that variable.
- The data must be positive integers. Fractional values are truncated after the decimal. Thus, note that a value larger than 0 but less than 1 becomes 0 and is regarded as a missing value. On the other hand, if a variable takes a value larger than its *max* but less than (*max* + 1), then the value is still valid because of the truncation.
- CATREG ignores user-missing value specifications. User-missing values larger than or equal to 1 are treated as valid category values (after truncation if necessary) and are included in the analysis. If you do not want the category included, use COMPUTE or RECODE to change the value to a CATREG-missing value.
- There must be at least three valid cases.
- The number of valid cases must be greater than the number of independent variables plus one.
- The maximum number of independent variables is 200.

Example

```
CATREG VARIABLES = TEST1 TEST3 (2) TEST2 TEST4 TEST5 (3) TEST6 (4)
                   TEST7 TO TEST9 (5) STATUS01 STATUS02 (4)
      /ANALYSIS TEST4 (ORDI)
           WITH TEST1 TO TEST2 (NUME) TEST5 TEST7 (NOMI) TEST8 (ORDI)
                STATUS01 STATUS02 (NOMI)
      /INITIAL = RANDOM
      /MAXITER = 100
      /CONVERGENCE = .000001
      /MISSING = MODEIMPU
      /PRINT = R COEFF FREQ ANOVA QUANT TRANS
      /PLOT = QUANT (TEST2 TO TEST7 TEST4)
      /SAVE
      /OUTFILE = 'c:\data\qdata.sav'.
```

- VARIABLES defines the variables and their maximum number of categories. 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, to be treated as ordinal 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 variables *TEST1*, *TEST2*, and *TEST3* are treated as numerical, the variables *TEST5*, *TEST7*, *STATUS01*, and *STATUS02* as nominal and variable *TEST8* as ordinal.
- 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.
- CONVERGENCE sets the convergence criterion to a value smaller than the default value.
- To include cases with CATREG-missing values, the MISSING subcommand specifies that for each variable missing values are replaced with the most frequent category (the mode).
- PRINT lists the correlations, the coefficients, the descriptive statistics, the ANOVA table, the category quantifications, 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.
- Each variable or variable list is followed by the maximum category number in parentheses. This value can be any positive integer, but in practice, if it is 1, then the procedure stops because of the 0 variance of such variable.
- Each maximum category number applies to all variables preceding the number but after the previous number if given.
- The keyword TO on the VARIABLES subcommand refers to the order of variables in the working data file.

ANALYSIS Subcommand

ANALYSIS specifies the dependent variable, the keyword WITH, and the independent variables (in that order).

- 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, whereas the variable list following WITH contains at least one variable as an independent variable. For each variable, you may add an optional keyword in parentheses indicating the transformation type of the variable.
- The keyword TO in the independent variable list honors the order of variables in the VARIABLES subcommand.

The following keywords can be specified to indicate the transformation type:

NOMI *Nominal*. Categories are treated as unordered. Objects in the same category obtain the same quantification.

ORDI *Ordinal*. This is the default for variables listed without any effective transformation type; that is, those without a transformation type specification after them. Categories are treated as ordered. The order of the categories of the observed variable is preserved in the optimally transformed variable.

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. When all variables are scaled at the numerical level, the CATREG analysis is analogous to standard multiple regression analysis.

- Keywords indicating transformation type can apply to a variable list as well as to a single variable. The default ORDI is not applied to a variable without a keyword if a subsequent variable on the list has a keyword.

INITIAL Subcommand

INITIAL specifies the method used to compute the initial solution/configuration.

- The specification on INITIAL is keyword NUMERICAL or RANDOM.
- If INITIAL is not specified, NUMERICAL is the default.

The following keywords can be specified:

NUMERICAL *Treat all variables as numerical.* This is usually best to use when there are only numerically and/or ordinally scaled variables.

RANDOM *Compute a random initial solution.* This should be used only when some or all of the variables are scaled at the nominal level.

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.

CONVERGENCE Subcommand

CONVERGENCE specifies a convergence criterion value. CATREG stops iterating if the difference in fit between the last two iterations is less than this value.

- If CONVERGENCE is not specified, the convergence value is 0.00001.
- The specification on CONVERGENCE is any value between 0.01 and 0.000001, inclusive.

MISSING Subcommand

CATREG ignores user-missing value specifications. User-missing values between 1 and the maximum value specified on the VARIABLES subcommand are treated as valid category values and are included in the analysis. Missing values in CATREG are values outside the valid range. By default, cases with CATREG-missing values on any of the variables are excluded from the analysis. The MISSING subcommand allows you to have CATREG-missing values on a variable replaced with the most frequent category (mode) of the variable.

The following keywords can be specified:

LISTWISE *Exclude cases with* CATREG-*missing value.* Only cases with valid values for all variables are included. This is the default when MISSING is not specified.

MODEIMPU *Replace* CATREG-*missing value with mode.* All cases are included and the imputations are treated as valid observations. When there are multiple modes, the smallest mode is used.

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 by the keyword OCORR. The default keywords are R, COEFF, FREQ, and ANOVA. If a keyword is duplicated or contradicted (for example, /PRINT = R R NONE), then the last occurence is executed.

The following keywords can be specified for PRINT:

R *Multiple R. R* includes R^2 and adjusted R^2.

COEFF *Standardized regression coefficients (beta).* COEFF includes the standard errors of the beta's, zero-order, part, and partial correlation, Pratt's relative importance measure for the transformed predictors, the tolerance before and after transformation and *F*. If the tolerance for a transformed predictor is lower than 0.0001, or if the tolerance is 0, then the computation for this variable is not done.

FREQ A table with *descriptive statistics* (transformation type, number of missing values, mode) including *marginal frequencies* for the variables in the analysis.

HISTORY *History of iterations.* For each iteration (including 0), the multiple *R*, and the regression error (square root of $(1 - R^2)$) are shown. The increase in multiple *R* is listed from iteration number 1.

ANOVA *Analysis of variance table.* This option includes regression and residual sums of squares, mean squares and *F*.

CORR *Correlations of the transformed predictors.*

OCORR *Correlations of the original predictors.*

QUANT *Category quantifications.*

TRANS (n) *Transformed data list. n* is a non-negative integer. If *n* is specified, only the first *n* cases are shown in the list, assuming that *n* is less than the total number of cases. When *n* is 0, the table is empty and does not appear. If *n* is not specified or if *n* is larger than or equal to the total number of cases, then all cases are shown.

DEFAULT *R, COEFF, FREQ, and ANOVA.*

NONE *No PRINT output is shown.*

PLOT Subcommand

This subcommand produces quantification plots (optimal quantifications against original category numbers). The following keyword must be specified:

QUANT(varlist)(l) *Quantification plots.* A list of variables must be given in parentheses following the keyword. A keyword ALL is allowed. Also, after the variables list, you can specify an optional parameter (l) in parentheses to control the length of all category value label lengths.

- If the plot keyword QUANT is omitted or if the variable list is empty, no plot is created.
- The category value label length parameter (|) must be an integer between 0 and 20, inclusive. If | = 0, instead of value labels, variable values are displayed on the horizontal axis. If the length parameter is omitted, the first 20 characters of the value label are used for each category.
- For categories without a value label, the 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 along the category axis if the length parameter is 2.
- The keyword TO in the variable list honors the order of the variables in the ANALYSIS subcommand.

SAVE Subcommand

The SAVE subcommand is used to add the transformed variables to the working data file. If on the MISSING subcommand LISTWISE is specified, excluded cases are represented by a dot on every variable.

- A variable rootname can be specified on the SAVE subcommand to which CATREG adds a number corresponding to the position of the variables on 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. Only one rootname can be specified and it can contain up to five characters (if more than one rootname is specified, the first name is used. If a rootname contains more than five characters, the first five characters are used) .
- If a rootname is not specified, unique variable names are automatically generated. The formula is *TRANSm_n*, where *m* increments from 1 to create unique variable names using the source variable's position numbers in the ANALYSIS subcommand and *n* increments from 1 to create unique rootnames for CATREG procedures with the SAVE subcommand. For example, the first set of default names, if they do not exist in the data file, would be *TRANS1_1*, *TRANS2_1*, *TRANS3_1*, and so forth. The next set of default names, if they do not exist in the data file, would be *TRANS1_2*, *TRANS2_2*, *TRANS3_2*, and so forth. If, for example, *TRANS2_2* already exists in the data file, then the default names would be *TRANS1_3*, *TRANS2_3*, *TRANS3_3*, and so forth.
- As *m* and/or *n* increase, the prefix *TRANS* is truncated to keep variable names within eight characters. For example, the variable after *TRANS9_1* would be *TRAN10_1*. The initial character *T* is required.

OUTFILE Subcommand

The OUTFILE subcommand is used to write the transformed data to an external data file. If on the MISSING subcommand LISTWISE is specified, excluded cases are represented by a dot on every variable.

- The specification on OUTFILE is a filename enclosed by single quotation marks.
- A working data file, in principle, should not be replaced by this subcommand, and the asterisk (*) file specification is not supported.

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 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, 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.

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, 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 5 rows and 4 columns where 2 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 necesary 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 CROSSSTABS.

CPROFILES *The column profiles*. PRINT=CPROFILES is analogous to the CELLS=COLUMN subcommand in CROSSSTABS.

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 file names.

- 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 Base 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 below, 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.

Example:

```
HOMALS  VARIABLES=ACOLA(2) BCOLA(2) CCOLA(2) DCOLA(2)
 /PRINT=FREQ EIGEN QUANT OBJECT.
```

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 Base 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 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.
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 2 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 which appear on the ANALYSIS subcommand use category labels corresponding to all categories within the defined range. Objects in a category which 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 twelve 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 Base 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.
```

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).
```

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 Base 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 (non-missing, 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 numerical 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 non-MNOM 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.

- If CONVERGENCE is not specified, the 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 which 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, twelve 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 5 categories, and one numerical level variable, with 10 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 Base 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.

Example:

```
PRINCALS VARIABLES=ACOLA BCOLA(2) PRICEA PRICEB(5)
 /ANALYSIS=ACOLA BCOLA(SNOM) PRICEA PRICEB(NUME)
 /PRINT=EIGEN QUANT OBJECT.
```

Overview

PRINCALS (*prin*cipal *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, 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 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 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 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. (See the *SPSS Base 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" above, 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*. 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 criteria 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 of 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 on 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 score plots labeled with variables which appear on the ANALYSIS subcommand use category labels corresponding to all categories within the defined range. Objects in a category which is outside the defined range are labeled with the label corresponding to the next category greater than the defined maximum category.
- 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 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 5 categories, and one numerical (interval level) variable, with 10 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 Base Syntax Reference Guide* for more information on matrix data files.

Subject Index

Syntax Index