New Developments in Alternating Least-Squares SAS Procedures

Warren F. Kuhfeld, The University of North Carolina at Chapel Hill
Thomas C. Eddy, The University of North Carolina at Chapel Hill
Douglas P. Kent, The University of North Carolina at Chapel Hill
Forrest W. Young, The University of North Carolina at Chapel Hill

ABSTRACT

A SAS software development project is currently in progress at the University of North Carolina at Chapel Hill. One of the purposes of this project is the development of SAS procedures that perform a variety of descriptive multivariate analyses. Three of these procedures, PROC PRINQUAL (principal components of qualitative data), PROC CORRESP (simple correspondence analysis) and PROC CONJOINT (analysis of variance and regression with a nonlinearly transformed dependent variable) will be described. PROC's CONJOINT and PRINQUAL fit a linear model to nonlinearly transformed variables using an alternating least-squares algorithm. PROC CORRESP can be used to summarize the information in a contingency table. The design and current state of development of these procedures will be discussed.

INTRODUCTION

A SAS software development project is currently underway at the University of North Carolina at Chapel Hill. In this paper we will report on the current state of three Version 6 SAS procedures that are being developed by this project. In addition, future directions will be discussed.

1) SAS is a registered trademark of SAS Institute Inc., Cary NC USA.

2) This paper describes experimental Version 6 SAS software. These procedures are being developed as a part of a SAS software development project at The Psychometric Laboratory of the University of North Carolina at Chapel Hill, not by SAS Institute. There is no commitment on the part of UNC or the Institute to support or distribute this software. If the software is distributed, no guarantee is made that it will be in accordance with the designs described herein. Questions concerning this paper, the software development project, or the procedures described in this paper should be directed to the authors at The Psychometric Laboratory, Davie Hall 013-A, The University of North Carolina, Chapel Hill NC USA, 27514, (919) 966-3548. Questions concerning the paper, project or procedures should not be directed to SAS Institute.

So far, the project has produced several programs. One program is a Version 5 plotting procedure, PROC IDPLOT, which is described elsewhere by Kuhfeld (1985, 1986a, 1986b, 1986c). The project has also produced the Version 5 PRINQUAL macro (Young and Kuhfeld, 1985) which is an enhancement of the 1982 PRINQUAL macro. Also, VISUALS, an interactive three-dimensional graphics system for the IBM PC AT with the Professional Graphics Device, has been developed and is being enhanced (Young et. al., 1986a, 1986b).

As of this writing, three Version 6 data analysis procedures have been completed, in preliminary form. These procedures, PROC PRINQUAL (principal components of qualitative data), PROC CORRESP (simple correspondence analysis), and PROC CONJOINT (conjoint and monotone multiple regression analysis) are the topics of this paper.

PROC PRINQUAL

The PRINQUAL procedure provides three methods of transforming a set of m variables. In all three methods, qualitative variables are transformed to decrease rank. One method uses a principal components algorithm (Young, Takane, and de Leeuw, 1978). The second method uses a multiple regression algorithm (Sarle, 1984). The third uses a regression-like algorithm where all regression weights are equal (de Leeuw, 1985). (Kuhfeld, Sarle, and Young (1985) discuss the first two methods as implemented in the Version 5, PRINQUAL macro, which was the prototype for the Version 6, PRINQUAL procedure.)

PROC PRINQUAL was designed to be used as a scoring procedure, like PROC STANDARD and PROC RANK. Its main function is to create an output data set. The results of the PRINQUAL analysis can be displayed with PROC's PLOT, PRINT, FACTOR, etc. The only printed output that PROC PRINQUAL produces is an iteration history summary table. Before the PRINQUAL algorithms are discussed in more detail, some of the uses of PROC PRINQUAL will be mentioned.

As the name PRINQUAL suggests, PROC PRINQUAL can be used to perform a principal components analysis of qualitative data. PROC PRINQUAL nonlinearly transforms the qualitative variables to maximize the fit of the data to the linear principal components model.
PROC PRINQUAL has powerful facilities for estimating missing data. Under the assumption of multivariate normality, with linear transformations, PRINQUAL becomes an EM algorithm for computing maximum likelihood estimates of the missing data.

PROC PRINQUAL can be used to perform a metric or nonmetric MDPREF analysis (Carroll, 1972). The output data set is constructed so that a MDPREF biplot can be produced by PROC PLOT or PROC IDPLOT without further manipulation of the data set.

PROC PRINQUAL can be used as a monotone multiple regression or conjoint analysis program. (However, PROC CONJOINT is much more efficient. In one test using the same data, PROC PRINQUAL required 5.21 seconds per iteration while PROC CONJOINT required 0.27 seconds per iteration.) If METHOD=MGV, one variable (the dependent variable) is monotonically transformed, and the remaining variables (continuous for monotone regression, a main effects design matrix for conjoint analysis, or any mixture for an arbitrary general linear models analysis) are linearly transformed, the dependent variable will be transformed to minimize the error sum of squares of the model.

PROC PRINQUAL METHODS

It was mentioned above that PROC PRINQUAL contains three algorithms or methods. The principal components based method constructs a representation of a set of data such that the qualitative variables are as much as possible (in a least-squares sense) like linear combinations of principal components, where n can be such smaller than m, the number of variables. The qualitative variables are transformed such that the total variance of the n components is maximized. Hence this method is called the maximum total variance (MTV) method.

The regression based method constructs a representation of a set of data such that each of the qualitative variables is as much as possible (in a least-squares sense) like linear combinations of principal components, where n can be such smaller than m, the number of variables. The qualitative variables are transformed such that the generalized variance of the transformed variables is minimized. Hence this method is called the minimum generalized variance (MGV) method.

The regression like method constructs a representation of a set of data such that each of the qualitative variables is as much as possible like an equally weighted average of the remaining variables. The qualitative variables are transformed such that the average of the correlations among the transformed variables is maximized. Hence this method is called the maximum average correlation (MAC) method.

All methods can be used to optimize some property of the correlation matrix among the variables before performing a final components or other data analysis. The MTV method maximizes the sum of the first n eigenvalues and minimizes the sum of the last (m - n) eigenvalues of the correlation matrix. The MGV method minimizes the product of the eigenvalues of the correlation matrix. The MAC method maximizes the average of the elements of the correlation matrix. In all cases, the sum of all m eigenvalues and the variance of each variable remains constant.

At the conclusion of the iterative process the PRINQUAL procedure has computed new scores for the observations on the qualitative variables (thus it has transformed these variables). The new scores are computed so that the measurement level of each variable is weakly maintained. This is done by a linear transformation of each quantitative variable, an order-preserving transformation of each ordinal variable, and a category-preserving transformation of each nominal variable. All variables are centered to mean zero. When correlations are analyzed, the variance of each variable is set to 1.0 throughout the analysis. When covariances are analyzed the variance of each variable is unchanged throughout the analysis.

PROC PRINQUAL TRANSFORMATIONS

Four types of transformations are currently available: OPSCORE, MONOTONE, UNTIE, and LINEAR. Each transformation places a different set of restrictions on the new variables's scorings.

OPSCORE specifies those variables which are optimally scored. This transformation assigns scores to each class (level) of the variable. Fisher's (1936) optimal scoring method is used. OPSCORE is appropriate for nominal variables. The final scoring maintains category membership.

MONOTONE specifies those variables which are to be transformed monotonically using the Kruskal & Shepard (1974) secondary least-squares monotonic transformation. This transformation is appropriate for ordinal variables. The final scoring weakly preserves order but not category membership (ties).

UNTIE specifies those variables which are to be transformed monotonically using the Kruskal & Shepard (1974) primary least-squares monotonic transformation. This transformation is appropriate for ordinal variables. The final scoring weakly preserves order but not category membership.

LINEAR specifies those variables which are subject to a linear transformation - change of origin (and scale if COV is not specified) only. This trans-
formation is appropriate for interval variables.

Ratio scale of measurement variables may be linearly transformed, and interval and ratio scale of measurement variables may be monotonically transformed, but these transformations will not preserve measurement level. There are many situations where nonlinear transformations are useful. For example, log and square root are two functional nonlinear transformations that are commonly used. PROC PRINQUAL can be used to hunt for appropriate functional transformations. The shape of the monotonic least-squares transformation of (say) a ratio scale of measurement variable may suggest that (for example) a log transformation of the original variable would be appropriate.

PROC PRINQUAL MISSING VALUE ESTIMATION

One of the most useful capabilities of PROC PRINQUAL is its powerful missing value estimation facilities. When the NOMISS option is used, observations with missing values are omitted from the analysis. Otherwise missing data are estimated.

No category or order restrictions are placed on the estimates of ordinary missing values (denoted 'A'). Missing value estimates within a variable can be forced to be identical by using special missing values. Up to 27 categories of missing values, where within category estimates must be the same, can be specified by coding the missing values using 'A', and 'Z'. A variable may have any combination of nonmissing, special missing, and ordinary missing values. However, no variable or observation may contain all missing values.

Missing data can be estimated without changing the mean and variance of the variables, or for linearly transformed variables, without changing the values of the nonmissing observations. The final mean and variance is controlled by the TSTANDARD= (transformation standardization) option. When TSTANDARD=ORIGINAL, the missing value estimation for a variable does not affect the mean and variance in the output data set. If it does, however, affect the number of observations that go into computing the mean and variance, and the actual values of the observations.

When TSTANDARD=NOMISS, the mean and variance of the variables in the output data set will not necessarily be the same as the mean and variance of the variables in the input data set, if there were missing values. But, if missing data are being estimated for linearly transformed variables, specifying TSTANDARD=NOMISS will result in variables whose original nonmissing values will not change. The difference between the original and transformed variables is that estimates will be substituted for the previously missing values.

The missing value estimation facilities can be used to allow for mixed type variables. For example, a variable can be considered to be part nominal and part ordinal, or part nominal and part interval. Nominal classes of ordinal and interval variables are coded with special missing values. This can be useful with survey research. The class 'unfamiliar with the product' in the variable: Rate your preference for "Brand X" on a 1 to 9 scale, or if you are unfamiliar with the product, check: "unfamiliar with the product", is an example.

"Unfamiliar with the product" can be coded as a special missing value, say 'A'. Then, if the variable is specified to be linear, the 1's to 9's will be linearly transformed, and all observations within the .A class will receive an optimal value based on the values in the entire data set. Or the "unfamiliar ..." class could be coded '.A'. Then each individual who checked "unfamiliar ..." could receive a different score. If the variable is specified to be ordinal, the 1's to 9's can be monotonically transformed, while no monotonic restrictions are placed on the quantification of the "unfamiliar ..." class.

PROC CORRESP

Correspondence analysis is a method of analyzing a two-way contingency table that finds the best simultaneous representation of the row and columns of the contingency table in a Euclidean space. The relationships among the row and column category values, as determined by the category frequencies, can be displayed in biplots. For example, the Table 1 artificial data appear in Greenacre (1984).

Table 1

<table>
<thead>
<tr>
<th>Smoking Level</th>
<th>None</th>
<th>Light</th>
<th>Medium</th>
<th>Heavy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Senior Manager</td>
<td>4</td>
<td>2</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Junior Manager</td>
<td>4</td>
<td>3</td>
<td>7</td>
<td>4</td>
</tr>
<tr>
<td>Senior Employee</td>
<td>25</td>
<td>10</td>
<td>12</td>
<td>4</td>
</tr>
<tr>
<td>Junior Employee</td>
<td>18</td>
<td>24</td>
<td>33</td>
<td>13</td>
</tr>
<tr>
<td>Secretary</td>
<td>0</td>
<td>6</td>
<td>7</td>
<td>2</td>
</tr>
</tbody>
</table>

The contingency table is a cross-tabulation of employee position by smoking level. Below is a result of a correspondence analysis of these data. All row and column categories are represented as points in the Figure 1 biplot. Correspondence analysis is related to principal components analysis. Since correspondence analysis is a newcomer to the SAS system, a few words that describe its purpose, and its relation to principal components analysis are in order. Principal components analysis is essen-
originally just a singular value decomposition (SVD). A raw multivariate data matrix (typically) interval scale of measurement variables defining the columns with replications on the rows, is (usually) centered, and (optionally) scaled so that each variable has unit variance. Then the resulting matrix $X$, is decomposed with a SVD: $X = UDV'$. The first column of $UD$ (the component score matrix) form a low rank approximate basis for the column space of $X$, and the first rows of $DV'$ (the component structure matrix) form a low rank approximate basis for the row space of $X$.

Now assume that the data matrix is a two-way contingency table. There would be several problems if such data were input to an ordinary principal components analysis. The idea of a set of variables and replications does not always apply to a contingency table. Centering is not reasonable since the numbers in the table represent ratio scale measurements, not interval scale measurements. Ordinary principal components analysis does not take marginal frequencies into account when it computes row and column coordinates.

Correspondence analysis might loosely be described as a modification of principal components analysis for a contingency table, that avoids the problems mentioned above. Rows and columns are treated in the same way, the columns are not centered, marginal frequencies are used as weights, and the main set of computations is still a SVD. Specifically, correspondence analysis involves a generalized singular value decomposition of departures from independence, (observed minus expected proportions), in a contingency table.

The formulas that underlie correspondence analysis are as follows. Let $C$ be an $n \times n$ contingency table of non-negative numbers with nonzero row and column sums. Let $P = (1/f)C$ where $f$ is the total frequency (grand sum) of the elements of $C$. Define $r = Pl$, $c = P'1$, $D_r = \text{diag}(r)$, and $D_c = \text{diag}(c)$, where $l$ is a vector of 1's of the appropriate order. Then the generalized singular value decomposition of $P - rc'$ is: $P - rc' = AGB'$ where $A' = B'D' = I$. $A$ is the matrix of left singular vectors, having $n_r$ rows and $n_c$ columns; $G$ is a diagonal matrix of singular values, having $n_r$ rows and columns; and $B$ is a square matrix of right singular vectors having $n_c$ rows and columns.

The above formulas can be expressed in terms of the ordinary SVD as follows: $D^{-1/2}(P - rc'D^{-1/2}) = UGV'$ where $U'U = V'V = I$. $UGV' = D^{-1/2}(AGB'D^{-1/2})$. $A = D^{-1/2}U$ and $B = D^{-1/2}V$. The columns of $A$ and $B$ define the principal axes of the column and row point clouds respectively, and the elements of $A$ and $B$ are the principal coordinates.

PROC CORRESP can be used to compute the coordinates for each row and column of the contingency table. These coordinates can be put into an output data set to serve as input to plotting procedures such as VISUALS, IDPLOT, G30, or PLOT.

PROC CORRESP provides you with four ways to standardize the coordinates, when STANDARD=NONE, the row coordinates are $A'$ and the column coordinates are $B'$. When STANDARD=ROWS, the row coordinates are $A'$ and the column coordinates are $A$. When STANDARD=COLUMNS, the row coordinates are $A'$ and the column coordinates are $B'$. When STANDARD=BOTH (the default), the row coordinates are $A'$ and the column coordinates are $B'$. PROC CORRESP can project supplemental variables and/or observations into the Euclidean space determined by the correspondence analysis. These variables and observations are also represented as points in this space according to their category frequencies.

PROC CORRESP can use as input, raw category responses on two classification variables, category frequency data concerning two classification variables, or a two-way contingency table. Output from PROC FREQ can serve as input to PROC CORRESP.

PROC CONJOINT

PROC CONJOINT is an alternating least-squares procedure that iteratively derives a monotone transformation of a dependent variable such that the fit of the transformed variable to a specified linear model is maximized. Like PROC PRINQUAL, PROC CONJOINT is a scoring program. It reads an input data set, creates an output data set, and produces little printed output (only an iteration history summary table). The output data set contains all of the variables in the input data set, and in addition contains a new variable which is the monotonic transformation of the dependent variable. The dependent variable is transformed to

849
minimize the error sums of squares of the model, under the constraint that the transformation must be monotone. The transformed dependent variable can be input to PROC GLM for the final analysis. However, PROC GLM p-values should generally be ignored, or interpreted only with extreme caution.

The current version of PROC CONJOINT is really just a monotone multiple regression program that allows for CLASS variables. The linear model is specified by a model statement of the form:

MODEL dependent variable = independent variable list;

All variables must be numeric. All variables named in the independent variable list are assumed to be continuous, unless named on the CLASS statement. All variables named on the CLASS statement are expanded to dummy variables for the analysis. The syntax does not currently allow for interactions. Eventually, PROC CONJOINT will allow the specification of any univariate linear model, just like PROC GLM.

The Table 2 set of data appeared in Green and Wind (1975) (except that the ranks here were reversed so that the most preferred combination was given a rank of 18 while the least preferred combination was given a rank of one). The data are ranked preferences of a carpet cleaner as a function of container design, brand name, Good Housekeeping seal, and money-back guarantee. The first five variables are levels from a five-way incomplete factorial design, and the sixth variable, RANK, is the rank order of the combinations that are defined by the five independent variables.

Table 2

<table>
<thead>
<tr>
<th>Design</th>
<th>Brand</th>
<th>Price</th>
<th>Seal</th>
<th>Guarantee</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>K2R</td>
<td>1.19</td>
<td>NO</td>
<td>NO</td>
<td>6</td>
</tr>
<tr>
<td>A</td>
<td>GLORY</td>
<td>1.39</td>
<td>NO</td>
<td>YES</td>
<td>8</td>
</tr>
<tr>
<td>A</td>
<td>BISSELL</td>
<td>1.59</td>
<td>YES</td>
<td>NO</td>
<td>2</td>
</tr>
<tr>
<td>B</td>
<td>K2R</td>
<td>1.39</td>
<td>YES</td>
<td>NO</td>
<td>17</td>
</tr>
<tr>
<td>B</td>
<td>GLORY</td>
<td>1.59</td>
<td>NO</td>
<td>NO</td>
<td>5</td>
</tr>
<tr>
<td>B</td>
<td>BISSELL</td>
<td>1.19</td>
<td>NO</td>
<td>NO</td>
<td>16</td>
</tr>
<tr>
<td>C</td>
<td>K2R</td>
<td>1.59</td>
<td>YES</td>
<td>YES</td>
<td>7</td>
</tr>
<tr>
<td>C</td>
<td>GLORY</td>
<td>1.19</td>
<td>YES</td>
<td>NO</td>
<td>12</td>
</tr>
<tr>
<td>C</td>
<td>BISSELL</td>
<td>1.39</td>
<td>NO</td>
<td>NO</td>
<td>10</td>
</tr>
<tr>
<td>A</td>
<td>K2R</td>
<td>1.59</td>
<td>YES</td>
<td>YES</td>
<td>1</td>
</tr>
<tr>
<td>A</td>
<td>GLORY</td>
<td>1.19</td>
<td>NO</td>
<td>YES</td>
<td>11</td>
</tr>
<tr>
<td>A</td>
<td>BISSELL</td>
<td>1.39</td>
<td>NO</td>
<td>NO</td>
<td>4</td>
</tr>
<tr>
<td>B</td>
<td>K2R</td>
<td>1.19</td>
<td>NO</td>
<td>NO</td>
<td>15</td>
</tr>
<tr>
<td>B</td>
<td>GLORY</td>
<td>1.39</td>
<td>YES</td>
<td>NO</td>
<td>13</td>
</tr>
<tr>
<td>B</td>
<td>BISSELL</td>
<td>1.59</td>
<td>NO</td>
<td>YES</td>
<td>14</td>
</tr>
<tr>
<td>C</td>
<td>K2R</td>
<td>1.39</td>
<td>NO</td>
<td>NO</td>
<td>9</td>
</tr>
<tr>
<td>C</td>
<td>GLORY</td>
<td>1.59</td>
<td>NO</td>
<td>NO</td>
<td>3</td>
</tr>
<tr>
<td>C</td>
<td>BISSELL</td>
<td>1.19</td>
<td>YES</td>
<td>YES</td>
<td>18</td>
</tr>
</tbody>
</table>

The PROC CONJOINT monotonic transformation of the variable RANK appears in Figure 2. The additive model fits the transformed ranks with a squared multiple correlation of 1.0. The squared multiple correlation for fitting the untransformed ranks is 0.98314. In this case, there was little room for improvement. Table 3 contains one-way marginal means computed by PROC SUMMARY. It can be seen that a guarantee is preferred to no guarantee, a cheaper price is preferred to a more expensive price, etc.

Table 3

<table>
<thead>
<tr>
<th>Condition</th>
<th>Marginal Means</th>
</tr>
</thead>
<tbody>
<tr>
<td>Guarantee</td>
<td>No 7.9872 12.5257</td>
</tr>
<tr>
<td>Seal</td>
<td>No 6.9957 Yes 10.5866</td>
</tr>
<tr>
<td>Brand</td>
<td>Bissell 10.6767 GLORY 8.6075 K2R 9.1358</td>
</tr>
<tr>
<td>Design</td>
<td>A 5.3257 B 13.3662 C 9.8982</td>
</tr>
</tbody>
</table>

FUTURE DIRECTIONS

PROC's CONJOINT, PRINQUAL, and CORRESP are finished in preliminary form. However, some final testing, polishing, and enhancements are needed before they will be released. It is hoped that in the next sixteen months, the following procedures will also be completed: PROC TRANSREG (transformation regression), PROC CDSAN (covariance structure analysis), PROC PROXSCAL (a multidimensional scaling procedure), and a Version 6 edition of PROC IDPLOT.
If you are interested in using or testing these procedures, contact us at the address listed in the footnote on the first page of this article. Do not contact SAS Institute about these procedures. Also, if you believe that you will use these procedures, and you have any comments or suggestions for options, please let us know.

REFERENCES


de Leeuw J. (1985), (Personal Communication).


