SAS/STAT® 9.22 User's Guide

The ORTHOREG Procedure

(Book Excerpt)

SAS® Documentation

This document is an individual chapter from *SAS/STAT® 9.22 User's Guide*.

The correct bibliographic citation for the complete manual is as follows: SAS Institute Inc. 2010. *SAS/STAT® 9.22 User's Guide*. Cary, NC: SAS Institute Inc.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st electronic book, May 2010

Chapter 63

The ORTHOREG Procedure

Contents

Overview: ORTHOREG Procedure

The ORTHOREG procedure fits general linear models by the method of least squares. Other SAS/STAT software procedures, such as the GLM and REG procedures, fit the same types of models, but PROC ORTHOREG can produce more accurate estimates than other regression procedures when your data are ill-conditioned. Instead of collecting crossproducts, PROC ORTHOREG uses Gentleman-Givens transformations to update and compute the upper triangular matrix **R** of the QR decomposition of the data matrix, with special care for scaling (Gentleman 1972, 1973). This method has the advantage over other orthogonalization methods (for example, Householder transformations) of not requiring the data matrix to be stored in memory.

The standard SAS regression procedures (PROC REG and PROC GLM) are very accurate for most problems. However, if you have very ill-conditioned data, these procedures can produce estimates that yield an error sum of squares very close to the minimum but still different from the exact least squares estimates. Normally, this coincides with estimates that have very high standard errors. In other words, the numerical error is much smaller than the statistical standard error.

PROC ORTHOREG fits models by the method of linear least squares, minimizing the sum of the squared residuals for predicting the responses—that is, the distance between the regression line and the observed Ys. The “ORTHO” in the name of the procedure refers to the orthogonalization approach to solving the least squares equations. In particular, PROC ORTHOREG does *not* perform the modeling method known as “orthogonal regression,” which minimizes a different criterion (namely, the distance between the regression line and the X/Y points taken together.)

Getting Started: ORTHOREG Procedure

Longley Data

The labor statistics data set of Longley (1967) is noted for being ill-conditioned. Both the ORTHOREG and GLM procedures are applied for comparison (only portions of the PROC GLM results are shown).

NOTE: The results from this example vary from machine to machine, depending on floating-point configuration.

The following statements read the data into the SAS data set Longley:

```
title 'PROC ORTHOREG used with Longley data';
data Longley;
   input Employment Prices GNP Jobless Military PopSize Year;
   datalines;
60323  83.0 234289 2356 1590 107608 1947
61122  88.5 259426 2325 1456 108632 1948
```

```
60171  88.2 258054 3682 1616 109773 1949
61187  89.5 284599 3351 1650 110929 1950
63221  96.2 328975 2099 3099 112075 1951
63639  98.1 346999 1932 3594 113270 1952
64989  99.0 365385 1870 3547 115094 1953
63761 100.0 363112 3578 3350 116219 1954
66019 101.2 397469 2904 3048 117388 1955
67857 104.6 419180 2822 2857 118734 1956
68169 108.4 442769 2936 2798 120445 1957
66513 110.8 444546 4681 2637 121950 1958
68655 112.6 482704 3813 2552 123366 1959
69564 114.2 502601 3931 2514 125368 1960
69331 115.7 518173 4806 2572 127852 1961
70551 116.9 554894 4007 2827 130081 1962
;
```

The data set contains one dependent variable, Employment (total derived employment), and six independent variables: Prices (GNP implicit price deflator normalized to the value 100 in 1954), GNP (gross national product), Jobless (unemployment), Military (size of armed forces), PopSize (noninstitutional population aged 14 and over), and Year (year).

The following statements use the ORTHOREG procedure to model the Longley data by using a quadratic model in each independent variable, without interaction:

```
proc orthoreg data=Longley;
   model Employment = Prices   Prices*Prices
                      GNP      GNP*GNP
                      Jobless  Jobless*Jobless
                      Military Military*Military
                      PopSize  PopSize*PopSize
                      Year     Year*Year;
run;
```

Figure 63.1 shows the resulting analysis.

Figure 63.1 PROC ORTHOREG Results

PROC ORTHOREG used with Longley data

The ORTHOREG Procedure

Dependent Variable: Employment

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	12	184864508.5	15405375.709	320.24	0.0003
Error	3	144317.49568	48105.831895		
Corrected Total	15	185008826			

Root MSE	219.33041717
R-Square	0.9992199426

Figure 63.1 *continued*

Parameter	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	186931078.640216	154201839.66	1.21	0.3122
Prices	1	1324.50679362506	916.17455832	1.45	0.2440
Prices**2	1	-6.61923922845539	4.7891445654	-1.38	0.2609
GNP	1	-0.12768642156232	0.0738897784	-1.73	0.1824
GNP**2	1	3.1369569286212E-8	8.7167753E-8	0.36	0.7428
Jobless	1	-4.35507653558708	1.3851792402	-3.14	0.0515
Jobless**2	1	0.00022132944101	0.0001763541	1.26	0.2983
Military	1	4.91162014560828	1.826715856	2.69	0.0745
Military**2	1	-0.00113707146734	0.0003539971	-3.21	0.0489
PopSize	1	-0.0303997234299	5.9272538242	-0.01	0.9962
PopSize**2	1	-1.212511414607E-6	0.0000237262	-0.05	0.9625
Year	1	-194907.139041839	157739.28757	-1.24	0.3045
Year**2	1	50.8067603538501	40.279878943	1.26	0.2963

The estimates in Figure 63.1 compare very well with the best estimates available; for additional information, see Longley (1967) and Beaton, Rubin, and Barone (1976).

The following statements request the same analysis from the GLM procedure:

```
proc glm data=Longley;
   model Employment = Prices   Prices*Prices
                      GNP      GNP*GNP
                      Jobless  Jobless*Jobless
                      Military Military*Military
                      PopSize  PopSize*PopSize
                      Year     Year*Year;
   ods select OverallANOVA
              FitStatistics
              ParameterEstimates
              Notes;
run;
```

Figure 63.2 contains the overall ANOVA table and the parameter estimates produced by PROC GLM. Notice that the PROC ORTHOREG fit achieves a somewhat smaller root mean square error (RMSE) and also that the GLM procedure detects spurious singularities.

Figure 63.2 Partial PROC GLM Results

```
                        PROC ORTHOREG used with Longley data

                                 The GLM Procedure

Dependent Variable: Employment

                                              Sum of
 Source                      DF             Squares     Mean Square    F Value    Pr > F

 Model                       11         184791061.6      16799187.4     308.58    <.0001

 Error                        4            217764.4         54441.1

 Corrected Total             15         185008826.0

            R-Square     Coeff Var      Root MSE     Employment Mean

            0.998823      0.357221      233.3262            65317.00

                                                     Standard
  Parameter                  Estimate                   Error    t Value    Pr > |t|

  Intercept              -3598851.899 B       1327335.652      -2.71      0.0535
  Prices                      523.802             688.979       0.76      0.4894
  Prices*Prices                -2.326               3.507      -0.66      0.5434
  GNP                          -0.138               0.078      -1.76      0.1526
  GNP*GNP                       0.000               0.000       0.24      0.8218
  Jobless                      -4.599               1.459      -3.15      0.0344
  Jobless*Jobless               0.000               0.000       1.14      0.3183
  Military                      4.994               1.942       2.57      0.0619
  Military*Military            -0.001               0.000      -3.15      0.0346
  PopSize                      -4.246               5.156      -0.82      0.4565
  PopSize*PopSize               0.000 B             0.000       0.81      0.4655
  Year                          0.000 B              .           .          .
  Year*Year                     1.038               0.419       2.48      0.0683

NOTE: The X'X matrix has been found to be singular, and a generalized inverse
      was used to solve the normal equations.  Terms whose estimates are
      followed by the letter 'B' are not uniquely estimable.
```

Syntax: ORTHOREG Procedure

The following statements are available in PROC ORTHOREG:

PROC ORTHOREG < *options* > **;**
CLASS *variables* < / *option* > **;**
MODEL *dependent-variable=independent-effects* < / *option* > **;**
BY *variables* **;**
EFFECT *name = effect-type (variables* < / *options* > *)* **;**
EFFECTPLOT < *plot-type* < *(plot-definition-options)* > >< / *options* > **;**
ESTIMATE < *'label'* > *estimate-specification* < / *options* > **;**
LSMEANS < *model-effects* > < / *options* > **;**
LSMESTIMATE *model-effect lsmestimate-specification* < / *options* > **;**
SLICE *model-effect* < / *options* > **;**
STORE < **OUT=** >*item-store-name* < / **LABEL=***'label'* > **;**
TEST < *model-effects* > < / *options* > **;**
WEIGHT *variable* **;**

The BY, CLASS, MODEL, and WEIGHT statements are described in full after the PROC ORTHOREG statement in alphabetical order. The EFFECT, EFFECTPLOT, ESTIMATE, LSMEANS, LSMESTIMATE, SLICE, STORE, and TEST statements are common to many procedures. Summary descriptions of functionality and syntax for these statements are also given after the PROC ORTHOREG statement in alphabetical order, and full documentation about them is available in Chapter 19, "Shared Concepts and Topics."

PROC ORTHOREG Statement

PROC ORTHOREG < *options* > **;**

The PROC ORTHOREG statement has the following options:

DATA=*SAS-data-set*
specifies the input SAS data set to use. By default, the procedure uses the most recently created SAS data set. The data set specified cannot be a TYPE=CORR, TYPE=COV, or TYPE=SSCP data set.

NOPRINT
suppresses the normal display of results. This option temporarily disables the Output Delivery System (ODS); see Chapter 20, "Using the Output Delivery System" for more information.

ORDER=DATA | FORMATTED | FREQ | INTERNAL
specifies the order in which to sort the levels of the classification variables (which are specified in the CLASS statement). This ordering determines which parameters in the model correspond to each level in the data, so the ORDER= option may be useful when you use ESTIMATE statement. This option applies to the levels for all classification variables, except when you use

the (default) ORDER=FORMATTED option with numeric classification variables that have no explicit format. With this option, the levels of such variables are ordered by their internal value.

The ORDER= option can take the following values:

Value of ORDER=	**Levels Sorted By**
DATA	Order of appearance in the input data set
FORMATTED	External formatted value, except for numeric variables with no explicit format, which are sorted by their unformatted (internal) value
FREQ	Descending frequency count; levels with the most observations come first in the order
INTERNAL	Unformatted value

By default, ORDER=FORMATTED. For FORMATTED and INTERNAL, the sort order is machine-dependent.

For more information about sorting order, see the chapter on the SORT procedure in the *Base SAS Procedures Guide* and the discussion of BY-group processing in *SAS Language Reference: Concepts*.

OUTEST=*SAS-data-set*
produces an output data set that contains the parameter estimates, the BY variables, and the special variables _TYPE_ (value "**PARMS**"), _NAME_ (blank), and _RMSE_ (root mean squared error).

SINGULAR=*s*
specifies a singularity criterion ($s \geq 0$) for the inversion of the triangular matrix **R**. By default, SINGULAR=10E−12.

BY Statement

BY *variables* **;**

You can specify a BY statement with PROC ORTHOREG to obtain separate analyses on observations in groups that are defined by the BY variables. When a BY statement appears, the procedure expects the input data set to be sorted in order of the BY variables. If you specify more than one BY statement, only the last one specified is used.

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

- Sort the data by using the SORT procedure with a similar BY statement.

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

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

CLASS Statement

CLASS *variables* < / **TRUNCATE** > **;**

The CLASS statement names the classification variables to be used in the model. Typical classification variables are Treatment, Sex, Race, Group, and Replication. If you use the CLASS statement, it must appear before the MODEL statement.

Classification variables can be either character or numeric. By default, class levels are determined from the entire set of formatted values of the CLASS variables.

NOTE: Prior to SAS 9, class levels were determined by using no more than the first 16 characters of the formatted values. To revert to this previous behavior, you can use the TRUNCATE option in the CLASS statement.

In any case, you can use formats to group values into levels. See the discussion of the FORMAT procedure in the *Base SAS Procedures Guide* and the discussions of the FORMAT statement and SAS formats in *SAS Language Reference: Dictionary*. You can adjust the order of CLASS variable levels with the ORDER= option in the PROC ORTHOREG statement. You can specify the following option in the CLASS statement after a slash (/):

TRUNCATE

specifies that class levels should be determined by using only up to the first 16 characters of the formatted values of CLASS variables. When formatted values are longer than 16 characters, you can use this option to revert to the levels as determined in releases prior to SAS 9.

EFFECT Statement (Experimental)

EFFECT *name = effect-type (variables < / options>)* **;**

The EFFECT statement enables you to construct special collections of columns for design matrices. These collections are referred to as *constructed effects* to distinguish them from the usual model effects formed from continuous or classification variables, as discussed in the section "GLM Parameterization of Classification Variables and Effects" on page 410 of Chapter 19, "Shared Concepts and Topics."

The following *effect-types* are available:

COLLECTION	is a collection effect that defines one or more variables as a single effect with multiple degrees of freedom. The variables in a collection are considered as a unit for estimation and inference.
LAG	is a classification effect in which the level that is used for a given period corresponds to the level in the preceding period.
MULTIMEMBER | MM	is a multimember classification effect whose levels are determined by one or more variables that appear in a CLASS statement.
POLYNOMIAL | POLY	is a multivariate polynomial effect in the specified numeric variables.
SPLINE	is a regression spline effect whose columns are univariate spline expansions of one or more variables. A spline expansion replaces the original variable with an expanded or larger set of new variables.

Table 63.1 summarizes important options for each type of EFFECT statement.

Table 63.1 Important EFFECT Statement Options

Option	Description
Options for Collection Effects	
DETAILS	Displays the constituents of the collection effect
Options for Lag Effects	
DESIGNROLE=	Names a variable that controls to which lag design an observation is assigned
DETAILS	Displays the lag design of the lag effect
NLAG=	Specifies the number of periods in the lag
PERIOD=	Names the variable that defines the period
WITHIN=	Names the variable or variables that define the group within which each period is defined
Options for Multimember Effects	
NOEFFECT	Specifies that observations with all missing levels for the multimember variables should have zero values in the corresponding design matrix columns
WEIGHT=	Specifies the weight variable for the contributions of each of the classification effects

Table 63.1 *continued*

Option	Description
Options for Polynomial Effects	
DEGREE=	Specifies the degree of the polynomial
MDEGREE=	Specifies the maximum degree of any variable in a term of the polynomial
STANDARDIZE=	Specifies centering and scaling suboptions for the variables that define the polynomial
Options for Spline Effects	
BASIS=	Specifies the type of basis (B-spline basis or truncated power function basis) for the spline expansion
DEGREE=	Specifies the degree of the spline transformation
KNOTMETHOD=	Specifies how to construct the knots for spline effects

For further details about the syntax of these *effect-types* and how columns of constructed effects are computed, see the section "EFFECT Statement (Experimental)" on page 418 of Chapter 19, "Shared Concepts and Topics."

EFFECTPLOT Statement

EFFECTPLOT < *plot-type* < *(plot-definition-options)* > > < / *options* > **;**

The EFFECTPLOT statement produces a display of the fitted model and provides options for changing and enhancing the displays. Table 63.2 describes the available *plot-types* and their *plot-definition-options*.

Table 63.2 *Plot-Types* and *Plot-Definition-Options*

Description	Plot-Definition-Options
BOX *plot-type*	
Displays a box plot of continuous response data at each level of a CLASS effect, with predicted values superimposed and connected by a line. This is an alternative to the INTERACTION *plot-type*.	PLOTBY= variable or CLASS effect X= CLASS variable or effect
CONTOUR *plot-type*	
Displays a contour plot of predicted values against two continuous covariates.	PLOTBY= variable or CLASS effect X= continuous variable Y= continuous variable
FIT *plot-type*	
Displays a curve of predicted values versus a continuous variable.	PLOTBY= variable or CLASS effect X= continuous variable

Table 63.2 *continued*

Description	Plot-Definition-Options
INTERACTION *plot-type*	
Displays a plot of predicted values (possibly with error bars) versus the levels of a CLASS effect. The predicted values are connected with lines and can be grouped by the levels of another CLASS effect.	PLOTBY= variable or CLASS effect SLICEBY= variable or CLASS effect X= CLASS variable or effect
SLICEFIT *plot-type*	
Displays a curve of predicted values versus a continuous variable grouped by the levels of a CLASS effect.	PLOTBY= variable or CLASS effect SLICEBY= variable or CLASS effect X= continuous variable

For full details about the syntax and options of the EFFECTPLOT statement, see the section "EFFECTPLOT Statement" on page 436 of Chapter 19, "Shared Concepts and Topics."

ESTIMATE Statement

ESTIMATE < *'label'*> *estimate-specification* < **(***divisor***=***n***)** >
< , . . . < *'label'*> *estimate-specification* < **(***divisor***=***n***)** > >
< / *options* > **;**

The ESTIMATE statement provides a mechanism for obtaining custom hypothesis tests. Estimates are formed as linear estimable functions of the form $\mathbf{L}\boldsymbol{\beta}$. You can perform hypothesis tests for the estimable functions, construct confidence limits, and obtain specific nonlinear transformations.

Table 63.3 summarizes important options in the ESTIMATE statement.

Table 63.3 Important ESTIMATE Statement Options

Option	Description
Construction and Computation of Estimable Functions	
DIVISOR=	Specifies a list of values to divide the coefficients
NOFILL	Suppresses the automatic fill-in of coefficients for higher-order effects
SINGULAR=	Tunes the estimability checking difference

Table 63.3 *continued*

Option	Description
Degrees of Freedom and *p*-values	
ADJUST=	Determines the method for multiple comparison adjustment of estimates
ALPHA=α	Determines the confidence level $(1 - \alpha)$
LOWER	Performs one-sided, lower-tailed inference
STEPDOWN	Adjusts multiplicity-corrected *p*-values further in a step-down fashion
TESTVALUE=	Specifies values under the null hypothesis for tests
UPPER	Performs one-sided, upper-tailed inference
Statistical Output	
CL	Constructs confidence limits
CORR	Displays the correlation matrix of estimates
COV	Displays the covariance matrix of estimates
E	Prints the **L** matrix
JOINT	Produces a joint F or chi-square test for the estimable functions
SEED=	Specifies the seed for computations that depend on random numbers

For details about the syntax of the ESTIMATE statement, see the section "ESTIMATE Statement" on page 462 of Chapter 19, "Shared Concepts and Topics."

LSMEANS Statement

LSMEANS < *model-effects* > < / *options* > **;**

The LSMEANS statement computes and compares least squares means (LS-means) of fixed effects. LS-means are *predicted population margins*—that is, they estimate the marginal means over a balanced population. In a sense, LS-means are to unbalanced designs as class and subclass arithmetic means are to balanced designs.

Table 63.4 summarizes important options in the LSMEANS statement.

Table 63.4 Important LSMEANS Statement Options

Option	Description
Construction and Computation of LS-Means	
AT	Modifies the covariate value in computing LS-means
BYLEVEL	Computes separate margins
DIFF	Requests differences of LS-means
OM=	Specifies the weighting scheme for LS-means computation as determined by the input data set
SINGULAR=	Tunes estimability checking

Table 63.4 *continued*

Option	Description
Degrees of Freedom and *p*-values	
ADJUST=	Determines the method for multiple comparison adjustment of LS-means differences
ALPHA=α	Determines the confidence level $(1-\alpha)$
STEPDOWN	Adjusts multiple comparison *p*-values further in a step-down fashion
Statistical Output	
CL	Constructs confidence limits for means and mean differences
CORR	Displays the correlation matrix of LS-means
COV	Displays the covariance matrix of LS-means
E	Prints the **L** matrix
LINES	Produces a "Lines" display for pairwise LS-means differences
MEANS	Prints the LS-means
PLOTS=	Requests ODS statistical graphics of means and mean comparisons
SEED=	Specifies the seed for computations that depend on random numbers

For details about the syntax of the LSMEANS statement, see the section "LSMEANS Statement" on page 479 of Chapter 19, "Shared Concepts and Topics."

LSMESTIMATE Statement

LSMESTIMATE *model-effect* < *'label'*> *values* < *divisor*=*n* >
< , . . . < *'label'*> *values* < *divisor*=*n* > >
< / *options*> **;**

The LSMESTIMATE statement provides a mechanism for obtaining custom hypothesis tests among least squares means.

Table 63.5 summarizes important options in the LSMESTIMATE statement.

Table 63.5 Important LSMESTIMATE Statement Options

Option	Description
Construction and Computation of LS-Means	
AT	Modifies covariate values in computing LS-means
BYLEVEL	Computes separate margins
DIVISOR=	Specifies a list of values to divide the coefficients
OM=	Specifies the weighting scheme for LS-means computation as determined by a data set
SINGULAR=	Tunes estimability checking

Table 63.5 *continued*

Option	Description
Degrees of Freedom and *p*-values	
ADJUST=	Determines the method for multiple comparison adjustment of LS-means differences
ALPHA=α	Determines the confidence level $(1 - \alpha)$
LOWER	Performs one-sided, lower-tailed inference
STEPDOWN	Adjusts multiple comparison p-values further in a step-down fashion
TESTVALUE=	Specifies values under the null hypothesis for tests
UPPER	Performs one-sided, upper-tailed inference
Statistical Output	
CL	Constructs confidence limits for means and mean differences
CORR	Displays the correlation matrix of LS-means
COV	Displays the covariance matrix of LS-means
E	Prints the **L** matrix
ELSM	Prints the **K** matrix
JOINT	Produces a joint F or chi-square test for the LS-means and LS-means differences
SEED=	Specifies the seed for computations that depend on random numbers

For details about the syntax of the LSMESTIMATE statement, see the section "LSMESTIMATE Statement" on page 496 of Chapter 19, "Shared Concepts and Topics."

MODEL Statement

MODEL *dependent-variable=independent-effects* < */ option* > **;**

The MODEL statement names the dependent variable and the independent effects. Only one MODEL statement is allowed. The specification of effects and the parameterization of the linear model are the same as in the GLM procedure; see Chapter 39, "The GLM Procedure" for further details.

The following option can be used in the MODEL statement:

NOINT

omits the intercept term from the model.

SLICE Statement

SLICE *model-effect* < / *options* > **;**

The SLICE statement provides a general mechanism for performing a partitioned analysis of the LS-means for an interaction. This analysis is also known as an analysis of simple effects.

The SLICE statement uses the same options as the LSMEANS statement, which are summarized in Table 19.19. For details about the syntax of the SLICE statement, see the section "SLICE Statement" on page 526 of Chapter 19, "Shared Concepts and Topics."

STORE Statement

STORE < **OUT=** >*item-store-name* < / **LABEL=**'*label*'> **;**

The STORE statement requests that the procedure save the context and results of the statistical analysis. The resulting *item store* is a binary file format that cannot be modified. The contents of the item store can be processed with the PLM procedure.

For details about the syntax of the STORE statement, see the section "STORE Statement" on page 529 of Chapter 19, "Shared Concepts and Topics."

TEST Statement

TEST < *model-effects* > < / *options* > **;**

The TEST statement enables you to perform *F* tests for model effects that test Type I, II, or Type III hypotheses. See Chapter 15, "The Four Types of Estimable Functions," for details about the construction of Type I, II, and III estimable functions.

Table 63.6 summarizes options in the TEST statement.

Table 63.6 TEST Statement Options

Option	Description
CHISQ	Requests chi-square tests
DDF=	Specifies denominator degrees of freedom for fixed effects
E	Requests Type I, Type II, and Type III coefficients
E1	Requests Type I coefficients
E2	Requests Type II coefficients
E3	Requests Type III coefficients
HTYPE=	Indicates the type of hypothesis test to perform
INTERCEPT	Adds a row that corresponds to the overall intercept

For details about the syntax of the TEST statement, see the section "TEST Statement" on page 530 of Chapter 19, "Shared Concepts and Topics."

WEIGHT Statement

WEIGHT *variable* **;**

A WEIGHT statement names a variable in the input data set whose values are relative weights for a weighted least squares regression. If the weight value is proportional to the reciprocal of the variance for each observation, the weighted estimates are the best linear unbiased estimates (BLUE). For a more complete description of the WEIGHT statement, see the section "WEIGHT Statement" on page 3042 in the GLM procedure.

Details: ORTHOREG Procedure

Missing Values

If there is a missing value for any model variable in an observation, the entire observation is dropped from the analysis.

Output Data Set

The OUTEST= option produces a TYPE=EST output SAS data set that contains the BY variables, parameter estimates, and four special variables. For each new value of the BY variables, PROC ORTHOREG outputs an observation to the OUTEST= data set. The variables in the data set are as follows:

- parameter estimates for all variables listed in the MODEL statement
- BY variables
- _TYPE_, which is a character variable with the value PARMS for every observation
- _NAME_, which is a character variable left blank for every observation
- _RMSE_, which is the root mean square error (the estimate of the standard deviation of the true errors)
- Intercept, which is the estimated intercept. This variable does not exist in the OUTEST= data set if the NOINT option is specified.

Displayed Output

PROC ORTHOREG displays the parameter estimates and associated statistics. These include the following:

- overall model analysis of variance, including the error mean square, which is an estimate of σ^2 (the variance of the true errors), and the overall F test for a model effect.
- root mean square error, which is an estimate of the standard deviation of the true errors. It is calculated as the square root of the mean squared error.
- R square (R^2) measures how much variation in the dependent variable can be accounted for by the model. R square, which can range from 0 to 1, is the ratio of the sum of squares for the model to the corrected total sum of squares. In general, the larger the value of R square, the better the model's fit.
- estimates for the parameters in the linear model

The table of parameter estimates consists of the following:

- the terms used as regressors, including the intercept.
- degrees of freedom (DF) for the variable. There is one degree of freedom for each parameter being estimated unless the model is not full rank.
- estimated linear coefficients.
- estimates of the standard errors of the parameter estimates.
- the critical t values for testing whether the parameters are zero. This is computed as the parameter estimate divided by its standard error.
- the two-sided p-value for the t test, which is the probability that a t statistic would obtain a greater absolute value than that observed given that the true parameter is zero.

ODS Table Names

PROC ORTHOREG assigns a name to each table it creates. You can use these names to reference the table when you use the Output Delivery System (ODS) to select tables and create output data sets. These names are listed in Table 63.7. For more information about ODS, see Chapter 20, "Using the Output Delivery System."

Each of the EFFECT, ESTIMATE, LSMEANS, LSMESTIMATE, and SLICE statements also creates tables, which are not listed in Table 63.7. For information about these tables, see the corresponding sections of Chapter 19, "Shared Concepts and Topics."

Table 63.7 ODS Tables Produced by PROC ORTHOREG

ODS Table Name	Description	Statement
ANOVA	Analysis of variance	Default
FitStatistics	Overall statistics for fit	Default
Levels	Table of class levels	CLASS statement
ParameterEstimates	Parameter estimates	Default

ODS Graph Names

When the ODS Graphics are in effect, then each of the EFFECT, ESTIMATE, LSMEANS, LSMESTIMATE, and SLICE statements can produce plots associated with their analyses. For information about these plots, see the corresponding sections of Chapter 19, "Shared Concepts and Topics."

Examples: ORTHOREG Procedure

Example 63.1: Precise Analysis of Variance

The data for the following example are from Powell, Murphy, and Gramlich (1982). In order to calibrate an instrument for measuring atomic weight, 24 replicate measurements of the atomic weight of silver (chemical symbol Ag) are made with the new instrument and with a reference instrument.

NOTE: The results from this example vary from machine to machine, depending on floating-point configuration.

The following statements read the measurements for the two instruments into the SAS data set AgWeight:

```
title 'Atomic Weight of Silver by Two Different Instruments';
data AgWeight;
   input Instrument AgWeight @@;
   datalines;
1 107.8681568   1 107.8681465   1 107.8681572   1 107.8681785
1 107.8681446   1 107.8681903   1 107.8681526   1 107.8681494
1 107.8681616   1 107.8681587   1 107.8681519   1 107.8681486
1 107.8681419   1 107.8681569   1 107.8681508   1 107.8681672
1 107.8681385   1 107.8681518   1 107.8681662   1 107.8681424
1 107.8681360   1 107.8681333   1 107.8681610   1 107.8681477
2 107.8681079   2 107.8681344   2 107.8681513   2 107.8681197
2 107.8681604   2 107.8681385   2 107.8681642   2 107.8681365
2 107.8681151   2 107.8681082   2 107.8681517   2 107.8681448
2 107.8681198   2 107.8681482   2 107.8681334   2 107.8681609
2 107.8681101   2 107.8681512   2 107.8681469   2 107.8681360
2 107.8681254   2 107.8681261   2 107.8681450   2 107.8681368
;
```

Notice that the variation in the atomic weight measurements is several orders of magnitude less than their mean. This is a situation that can be difficult for standard, regression-based analysis-of-variance procedures to handle correctly.

The following statements invoke the ORTHOREG procedure to perform a simple one-way analysis of variance, testing for differences between the two instruments:

```
proc orthoreg data=AgWeight;
   class Instrument;
   model AgWeight = Instrument;
run;
```

Output 63.1.1 shows the resulting analysis.

Output 63.1.1 PROC ORTHOREG Results for Atomic Weight Example

```
                 Atomic Weight of Silver by Two Different Instruments

                              The ORTHOREG Procedure

                              Class Level Information
                            Factor        Levels    -Values-

                            Instrument         2    1 2

                 Atomic Weight of Silver by Two Different Instruments

                              The ORTHOREG Procedure

                           Dependent Variable: AgWeight

                                        Sum of
Source                      DF         Squares     Mean Square    F Value    Pr > F

Model                        1     3.6383419E-9    3.6383419E-9      15.95    0.0002
Error                       46     1.0495173E-8    2.281559E-10
Corrected Total             47     1.4133515E-8

                              Root MSE     0.0000151048
                              R-Square     0.2574265445


                                                        Standard
Parameter           DF   Parameter Estimate               Error   t Value   Pr > |t|

Intercept            1     107.868136354166    3.0832608E-6   3.499E7     <.0001
(Instrument='1')     1     0.00001741249999    4.3603893E-6      3.99     0.0002
(Instrument='2')     0                    0               .         .          .
```

The mean difference between instruments is about 1.74×10^{-5} (the value of the `(Instrument='1')` parameter in the parameter estimates table), whereas the level of background variation in the measurements is about 1.51×10^{-5} (the value of the root mean square error). At this level of error, the difference is significant, with a p-value of 0.0002.

The National Institute of Standards and Technology (1998) has provided certified ANOVA values for this data set. The following statements use ODS to examine the ANOVA values produced by both the ORTHOREG and GLM procedures more precisely for comparison with the NIST-certified values:

```
ods listing close;
ods output ANOVA         = OrthoregANOVA
           FitStatistics = OrthoregFitStat;

proc orthoreg data=AgWeight;
   class Instrument;
   model AgWeight = Instrument;
run;

ods output OverallANOVA  = GLMANOVA
           FitStatistics = GLMFitStat;
proc glm data=AgWeight;
   class Instrument;
   model AgWeight = Instrument;
run;
ods listing;

data _null_; set OrthoregANOVA  (in=inANOVA)
                 OrthoregFitStat(in=inFitStat);
   if (inANOVA) then do;
      if (Source = 'Model') then put "Model SS: " ss e20.;
      if (Source = 'Error') then put "Error SS: " ss e20.;
   end;
   if (inFitStat) then do;
      if (Statistic = 'Root MSE') then
                            put "Root MSE: " nValue1 e20.;
      if (Statistic = 'R-Square') then
                         put "R-Square: " nValue1 best20.;
   end;
data _null_; set GLMANOVA  (in=inANOVA)
                 GLMFitStat(in=inFitStat);
   if (inANOVA) then do;
      if (Source = 'Model') then put "Model SS: " ss e20.;
      if (Source = 'Error') then put "Error SS: " ss e20.;
   end;
   if (inFitStat) then      put "Root MSE: " RootMSE e20.;
   if (inFitStat) then   put "R-Square: " RSquare best20.;
run;
```

In SAS/STAT software prior to SAS 8, PROC GLM gave much less accurate results than PROC ORTHOREG. Table 63.8 and Table 63.9 compare the ANOVA values certified by NIST with those produced by the two procedures.

Table 63.8 Accuracy Comparison for Sums of Squares

Values	Model SS	Error SS
NIST-certified	3.6383418750000E−09	1.0495172916667E−08
ORTHOREG	3.6383418747907E−09	1.0495172916797E−08
GLM, since SAS 8	3.6383418747907E−09	1.0495172916797E−08
GLM, before SAS 8	0	1.0331496763990E−08

Table 63.9 Accuracy Comparison for Fit Statistics

Values	Root MSE	R Square
NIST-certified	1.5104831444641E−05	0.25742654453832
ORTHOREG	1.5104831444735E−05	0.25742654452494
GLM, since SAS 8	1.5104831444735E−05	0.25742654452494
GLM, before SAS 8	1.4986585859992E−05	0

Although the PROC ORTHOREG values and the PROC GLM values for the current version are quite close to the certified ones, the PROC GLM values for releases prior to SAS 8 are not. In fact, since the model sum of squares is so small, in prior releases the GLM procedure set it (and consequently R square) to zero.

Example 63.2: Wampler Data

This example applies the ORTHOREG procedure to a collection of data sets noted for being ill-conditioned. The OUTEST= data set is used to collect the results for comparison with values certified to be correct by the National Institute of Standards and Technology (1998).

NOTE: The results from this example vary from machine to machine, depending on floating-point configuration.

The data are from Wampler (1970). The independent variates for all five data sets are x^i, $i = 1, \ldots 5$, for $x = 0, 1, \ldots, 20$. Two of the five dependent variables are exact linear functions of the independent terms:

$$
\begin{aligned}
y_1 &= 1 + x + x^2 + x^3 + x^4 + x^5 \\
y_2 &= 1 + 0.1x + 0.01x^2 + 0.001x^3 + 0.0001x^4 + 0.00001x^5
\end{aligned}
$$

The other three dependent variables have the same mean value as y_1, but with nonzero errors:

$$\begin{aligned} y_3 &= y_1 + \mathbf{e} \\ y_4 &= y_1 + 100\mathbf{e} \\ y_5 &= y_1 + 10000\mathbf{e} \end{aligned}$$

where **e** is a vector of values with standard deviation ~2044, chosen to be orthogonal to the mean model for y_1.

The following statements create a SAS data set Wampler that contains the Wampler data, run a SAS macro program that uses PROC ORTHOREG to fit a fifth-order polynomial in x to each of the Wampler dependent variables, and collect the results in a data set named ParmEst:

```
data Wampler;
   do x=0 to 20;
      input e @@;
      y1 = 1 +       x    +        x**2 +      x**3
             +       x**4 +        x**5;
      y2 = 1 + .1   *x    + .01   *x**2 + .001*x**3
             + .0001*x**4 + .00001*x**5;
      y3 = y1 +       e;
      y4 = y1 +   100*e;
      y5 = y1 + 10000*e;
      output;
   end;
   datalines;
759 -2048 2048 -2048 2523 -2048 2048 -2048 1838 -2048 2048
-2048 1838 -2048 2048 -2048 2523 -2048 2048 -2048 759
;

%macro WTest;
   data ParmEst; if (0); run;
   %do i = 1 %to 5;
      proc orthoreg data=Wampler outest=ParmEst&i noprint;
         model y&i = x x*x x*x*x x*x*x*x x*x*x*x*x;
      data ParmEst&i; set ParmEst&i; Dep = "y&i";
      data ParmEst; set ParmEst ParmEst&i;
         label Col1='x'     Col2='x**2'  Col3='x**3'
               Col4='x**4'  Col5='x**5';
      run;
   %end;
%mend;
%WTest;
```

Instead of displaying the raw values of the RMSE and parameter estimates, use an additional DATA step as follows to compute the deviations from the values certified to be correct by the National Institute of Standards and Technology (1998):

```
data ParmEst; set ParmEst;
   if      (Dep = 'y1') then
      _RMSE_ = _RMSE_ - 0.00000000000000;
   else if (Dep = 'y2') then
      _RMSE_ = _RMSE_ - 0.00000000000000;
   else if (Dep = 'y3') then
      _RMSE_ = _RMSE_ - 2360.14502379268;
   else if (Dep = 'y4') then
      _RMSE_ = _RMSE_ - 236014.502379268;
   else if (Dep = 'y5') then
      _RMSE_ = _RMSE_ - 23601450.2379268;
   if (Dep ^= 'y2') then do;
      Intercept = Intercept - 1.00000000000000;
      Col1      = Col1      - 1.00000000000000;
      Col2      = Col2      - 1.00000000000000;
      Col3      = Col3      - 1.00000000000000;
      Col4      = Col4      - 1.00000000000000;
      Col5      = Col5      - 1.00000000000000;
   end;
   else do;
      Intercept = Intercept - 1.00000000000000;
      Col1      = Col1      - 0.100000000000000;
      Col2      = Col2      - 0.100000000000000e-1;
      Col3      = Col3      - 0.100000000000000e-2;
      Col4      = Col4      - 0.100000000000000e-3;
      Col5      = Col5      - 0.100000000000000e-4;
   end;
run;

proc print data=ParmEst label noobs;
   title 'Wampler data: Deviations from Certified Values';
   format _RMSE_ Intercept Col1-Col5 e9.;
   var Dep _RMSE_ Intercept Col1-Col5;
run;
```

The results, shown in Output 63.2.1, indicate that the values computed by PROC ORTHOREG are quite close to the NIST-certified values.

Output 63.2.1 Wampler Data: Deviations from Certified Values

Wampler data: Deviations from Certified Values

Dep	_RMSE_	Intercept	x	x**2	x**3	x**4	x**5
y1	0.00E+00	5.46E-12	-9.82E-11	1.55E-11	-5.68E-13	3.55E-14	-6.66E-16
y2	0.00E+00	8.88E-16	-3.19E-15	1.24E-15	-1.88E-16	1.20E-17	-2.57E-19
y3	-2.09E-11	-7.73E-11	1.46E-11	-2.09E-11	2.50E-12	-1.28E-13	2.66E-15
y4	-4.07E-10	-5.38E-10	8.99E-10	-3.29E-10	4.23E-11	-2.27E-12	4.35E-14
y5	-3.35E-08	-4.10E-08	8.07E-08	-2.77E-08	3.54E-09	-1.90E-10	3.64E-12

Example 63.3: Fitting Polynomials

The extra accuracy of the regression algorithm used by PROC ORTHOREG is most useful when the model contains near-singularities that you want to be able to distinguish from true singularities. This example demonstrates this usefulness in the context of fitting polynomials of high degree.

NOTE: The results from this example vary from machine to machine, depending on floating-point configuration.

The following DATA step computes a response y as an exact ninth-degree polynomial function of a predictor x evaluated at 0, 0.01, 0.02, ..., 1.

```
title 'Polynomial Data';
data Polynomial;
   do i = 1 to 101;
      x = (i-1)/(101-1);
      y = 10**(9/2);
      do j = 0 to 8;
         y = y * (x - j/8);
      end;
      output;
   end;
run;
```

The polynomial is constructed in such a way that its zeros lie at $x = i/8$ for $i = 0, \ldots, 8$. The following statements use the EFFECT statement to fit a ninth-degree polynomial to this data with PROC ORTHOREG. The EFFECT statement makes it easy to specify complicated polynomial models.

```
ods graphics on;

proc orthoreg data=Polynomial;
   effect xMod = polynomial(x / degree=9);
   model y = xMod;
   effectplot fit / obs;
   store OStore;
run;

ods graphics off;
```

The effect xMod defined by the EFFECT statement refers to all nine degrees of freedom in the ninth-degree polynomial (excluding the intercept term). The resulting output is shown in Output 63.3.1. Note that the R square for the fit is 1, indicating that the ninth-degree polynomial has been correctly fit.

Output 63.3.1 PROC ORTHOREG Results for Ninth-Degree Polynomial

Polynomial Data

The ORTHOREG Procedure

Dependent Variable: y

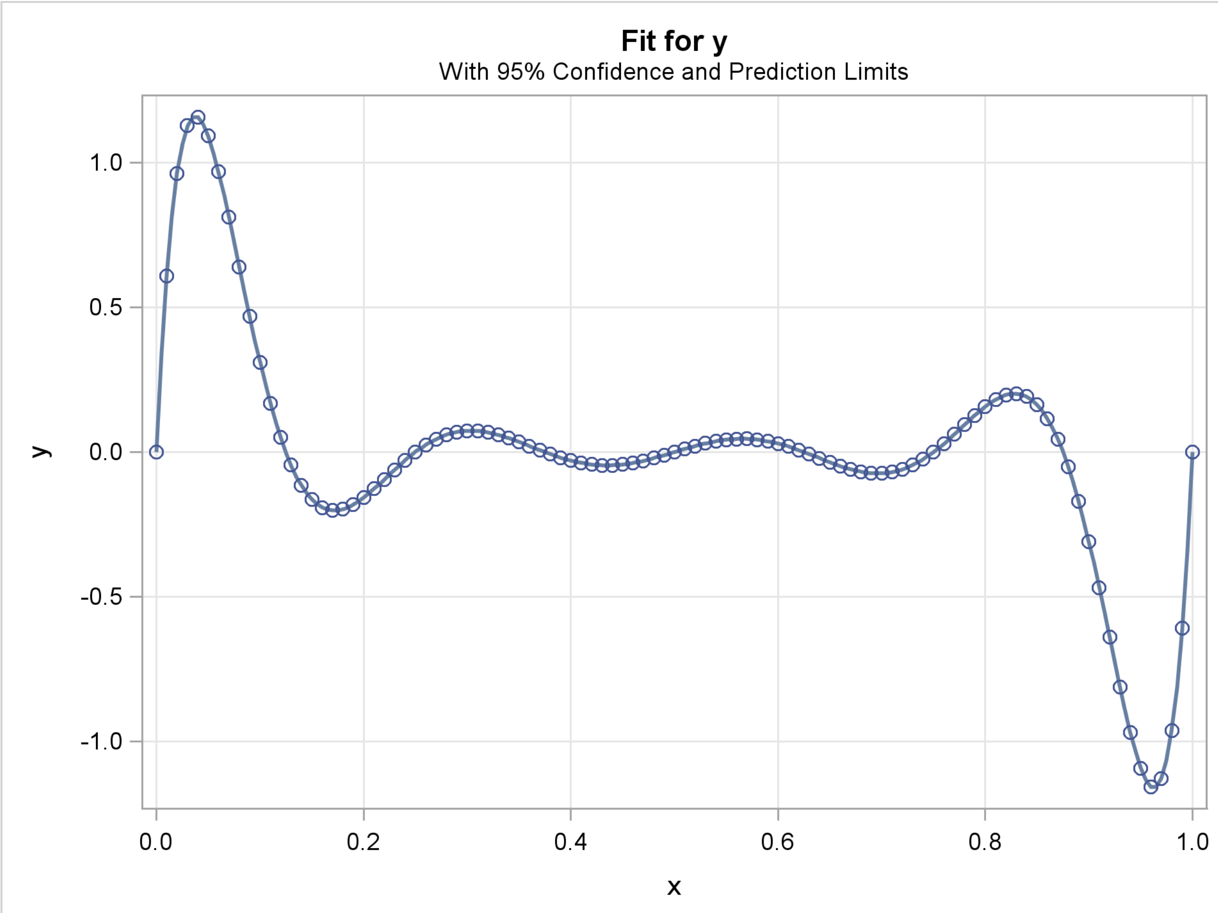
Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	9	15.527180055	1.7252422284	1.65E22	<.0001
Error	91	9.496616E-21	1.043584E-22		
Corrected Total	100	15.527180055			

Root MSE	1.02156E-11
R-Square	1

Parameter	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	-3.24572035915E-11	8.114115E-12	-4.00	0.0001
x	1	75.9977312440678	4.898326E-10	1.55E11	<.0001
x_2	1	-1652.40781362191	9.5027919E-9	-174E9	<.0001
x_3	1	14249.4539769783	8.3110512E-8	1.71E11	<.0001
x_4	1	-64932.461575205	3.8997072E-7	-167E9	<.0001
x_5	1	173315.359360779	1.066611E-6	1.62E11	<.0001
x_6	1	-280158.03646002	1.7523078E-6	-16E10	<.0001
x_7	1	269781.812887653	1.7021134E-6	1.58E11	<.0001
x_8	1	-142302.494710055	9.0027891E-7	-158E9	<.0001
x_9	1	31622.7766022468	1.997493E-7	1.58E11	<.0001

The fit plot produced by the EFFECTPLOT statement, Output 63.3.2, also demonstrates the perfect fit.

Output 63.3.2 PROC ORTHOREG Fit Plot for Ninth-Degree Polynomial



Finally, you can use the PLM procedure with the fit model saved by the STORE statement in the item store OStore to check the predicted values for the known zeros of the polynomial, as shown in the following statements:

```
data Zeros(keep=x);
   do j = 0 to 8;
      x = j/8;
      output;
   end;
run;

proc plm restore=OStore noprint;
   score data=Zeros out=OZeros pred=OPred;
run;

proc print noobs;
run;
```

The predicted values of the zeros, shown in Output 63.3.3, are again all miniscule.

Output 63.3.3 Predicted Zeros for Ninth-Degree Polynomial

Polynomial Data

Obs	x	OPred
1	0.000	-3.2457E-11
2	0.125	-2.1262E-11
3	0.250	-9.5867E-12
4	0.375	-2.2895E-11
5	0.500	-5.2154E-11
6	0.625	-1.2329E-10
7	0.750	-2.5329E-10
8	0.875	-3.9836E-10
9	1.000	-5.9663E-10

To compare these results with those from a least squares fit produced by an alternative algorithm, consider fitting a polynomial to this data using the GLM procedure. PROC GLM does not have an EFFECT statement, but the familiar bar notation can still be used to specify a ninth-degree polynomial fairly succinctly, as shown in the following statements:

```
proc glm data=Polynomial;
   model y = x|x|x|x|x|x|x|x|x;
   store GStore;
run;
```

Partial results are shown in Output 63.3.4. In this case, the R square for the fit is only about 0.83, indicating that the full ninth-degree polynomial was not correctly fit.

Output 63.3.4 PROC GLM for Ninth-Degree Polynomial

Polynomial Data

The GLM Procedure

Dependent Variable: y

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	8	12.91166643	1.61395830	56.77	<.0001
Error	92	2.61551363	0.02842950		
Corrected Total	100	15.52718006			

R-Square	Coeff Var	Root MSE	y Mean
0.831553	-6.6691E17	0.168610	-0.000000

The following statements, which use the PLM procedure to compute predictions based on the GLM fit at the true zeros of the polynomial, also confirm that PROC GLM is not able to correctly fit a polynomial of this degree, as shown in Output 63.3.5.

```
proc plm restore=GStore noprint;
   score data=Zeros out=GZeros pred=GPred;
run;

data Zeros;
   merge OZeros GZeros;
run;

proc print noobs;
run;
```

Output 63.3.5 Predicted Zeros for Ninth-Degree Polynomial

Polynomial Data

Obs	x	OPred	GPred
1	0.000	-3.2457E-11	0.44896
2	0.125	-2.1262E-11	0.22087
3	0.250	-9.5867E-12	-0.19037
4	0.375	-2.2895E-11	0.12710
5	0.500	-5.2154E-11	0.00000
6	0.625	-1.2329E-10	-0.12710
7	0.750	-2.5329E-10	0.19037
8	0.875	-3.9836E-10	-0.22087
9	1.000	-5.9663E-10	-0.44896

References

Beaton, A. E., Rubin, D. B., and Barone, J. L. (1976), "The Acceptability of Regression Solutions: Another Look at Computational Accuracy," *Journal of the American Statistical Association*, 71, 158–168.

Gentleman, W. M. (1972), *Basic Procedures for Large, Sparse, or Weighted Least Squares Problems*, Technical Report CSRR-2068, University of Waterloo, Ontario.

Gentleman, W. M. (1973), "Least Squares Computations by Givens Transformations without Square Roots," *J. Inst. Math. Appl.*, 12, 329–336.

Lawson, C. L. and Hanson, R. J. (1974), *Solving Least Squares Problems*, Englewood Cliffs, NJ: Prentice-Hall.

Longley, J. W. (1967), "An Appraisal of Least Squares Programs for the Electronic Computer from the Point of View of the User," *Journal of the American Statistical Association*, 62, 819–841.

National Institute of Standards and Technology (1998), "Statistical Reference Data Sets," http://www.nist.gov/srd/index.htm, last accessed January 29, 2010.

Powell, L. J., Murphy, T. J., and Gramlich, J. W. (1982), "The Absolute Isotopic Abundance and Atomic Weight of a Reference Sample of Silver," *NBS Journal of Research*, 87, 9–19.

Wampler, R. H. (1970), "A Report of the Accuracy of Some Widely Used Least Squares Computer Programs," *Journal of the American Statistical Association*, 65, 549–563.

Subject Index

Syntax Index

Your Turn

We welcome your feedback.

- If you have comments about this book, please send them to **`yourturn@sas.com`**. Include the full title and page numbers (if applicable).
- If you have comments about the software, please send them to **`suggest@sas.com`**.

SAS® Publishing Delivers!

Whether you are new to the work force or an experienced professional, you need to distinguish yourself in this rapidly changing and competitive job market. SAS® Publishing provides you with a wide range of resources to help you set yourself apart. Visit us online at support.sas.com/bookstore.

SAS® Press

Need to learn the basics? Struggling with a programming problem? You'll find the expert answers that you need in example-rich books from SAS Press. Written by experienced SAS professionals from around the world, SAS Press books deliver real-world insights on a broad range of topics for all skill levels.

support.sas.com/saspress

SAS® Documentation

To successfully implement applications using SAS software, companies in every industry and on every continent all turn to the one source for accurate, timely, and reliable information: SAS documentation. We currently produce the following types of reference documentation to improve your work experience:

- Online help that is built into the software.
- Tutorials that are integrated into the product.
- Reference documentation delivered in HTML and PDF – **free** on the Web.
- Hard-copy books.

support.sas.com/publishing

SAS® Publishing News

Subscribe to SAS Publishing News to receive up-to-date information about all new SAS titles, author podcasts, and new Web site features via e-mail. Complete instructions on how to subscribe, as well as access to past issues, are available at our Web site.

support.sas.com/spn

sas® THE POWER TO KNOW®