



THE  
POWER  
TO KNOW.

# **SAS/STAT<sup>®</sup> 9.3 User's Guide**

## **The MIANALYZE Procedure**

### **(Chapter)**



This document is an individual chapter from *SAS/STAT® 9.3 User's Guide*.

The correct bibliographic citation for the complete manual is as follows: SAS Institute Inc. 2011. *SAS/STAT® 9.3 User's Guide*. Cary, NC: SAS Institute Inc.

Copyright © 2011, SAS Institute Inc., Cary, NC, USA

All rights reserved. Produced in the United States of America.

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

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

**U.S. Government Restricted Rights Notice:** Use, duplication, or disclosure of this software and related documentation by the U.S. government is subject to the Agreement with SAS Institute and the restrictions set forth in FAR 52.227-19, Commercial Computer Software-Restricted Rights (June 1987).

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

1st electronic book, July 2011

SAS® Publishing provides a complete selection of books and electronic products to help customers use SAS software to its fullest potential. For more information about our e-books, e-learning products, CDs, and hard-copy books, visit the SAS Publishing Web site at [support.sas.com/publishing](http://support.sas.com/publishing) or call 1-800-727-3228.

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

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

## Chapter 57

# The MIANALYZE Procedure

### Contents

---

|  |             |
|--|-------------|
| Overview: MIANALYZE Procedure . . . . .  | <b>4636</b> |
| Getting Started: MIANALYZE Procedure . . . . .   | <b>4637</b> |
| Syntax: MIANALYZE Procedure . . . . .  | <b>4640</b> |
| PROC MIANALYZE Statement . . . . .   | 4640        |
| BY Statement . . . . .   | 4643        |
| CLASS Statement . . . . .  | 4643        |
| MODELEFFECTS Statement . . . . .   | 4644        |
| STDERR Statement . . . . .   | 4644        |
| TEST Statement . . . . .   | 4644        |
| Details: MIANALYZE Procedure . . . . .   | <b>4646</b> |
| Input Data Sets . . . . .  | 4646        |
| Combining Inferences from Imputed Data Sets . . . . .  | 4650        |
| Multiple Imputation Efficiency . . . . .   | 4652        |
| Multivariate Inferences . . . . .  | 4652        |
| Testing Linear Hypotheses about the Parameters . . . . .                                     | 4654        |
| Examples of the Complete-Data Inferences . . . . .   | 4654        |
| ODS Table Names . . . . .  | 4656        |
| Examples: MIANALYZE Procedure . . . . .  | <b>4657</b> |
| Example 57.1: Reading Means and Standard Errors from Variables in a DATA= Data Set . . . . . | 4658        |
| Example 57.2: Reading Means and Covariance Matrices from a DATA= COV Data Set . . . . .      | 4661        |
| Example 57.3: Reading Regression Results from a DATA= EST Data Set . . . . .                 | 4663        |
| Example 57.4: Reading Mixed Model Results from PARMS= and COVB= Data Sets . . . . .          | 4665        |
| Example 57.5: Reading Generalized Linear Model Results . . . . .                             | 4668        |
| Example 57.6: Reading GLM Results from PARMS= and XPXI= Data Sets . . . . .                  | 4670        |
| Example 57.7: Reading Logistic Model Results from PARMS= and COVB= Data Sets . . . . .       | 4672        |
| Example 57.8: Reading Mixed Model Results with Classification Variables . . . . .            | 4674        |
| Example 57.9: Using a TEST statement . . . . .   | 4677        |
| Example 57.10: Combining Correlation Coefficients . . . . .                                  | 4679        |
| References . . . . .   | <b>4682</b> |

---

---

## Overview: MIANALYZE Procedure

The MIANALYZE procedure combines the results of the analyses of imputations and generates valid statistical inferences. Multiple imputation provides a useful strategy for analyzing data sets with missing values. Instead of filling in a single value for each missing value, Rubin's (1976, 1987) multiple imputation strategy replaces each missing value with a set of plausible values that represent the uncertainty about the right value to impute.

Multiple imputation inference involves three distinct phases:

1. The missing data are filled in  $m$  times to generate  $m$  complete data sets.
2. The  $m$  complete data sets are analyzed using standard statistical analyses.
3. The results from the  $m$  complete data sets are combined to produce inferential results.

A companion procedure, PROC MI, creates multiply imputed data sets for incomplete multivariate data. It uses methods that incorporate appropriate variability across the  $m$  imputations.

The analyses of imputations are obtained by using standard SAS procedures (such as PROC REG) for complete data. No matter which complete-data analysis is used, the process of combining results from different imputed data sets is essentially the same and results in valid statistical inferences that properly reflect the uncertainty due to missing values. These results of analyses are combined in the MIANALYZE procedure to derive valid inferences.

The MIANALYZE procedure reads parameter estimates and associated standard errors or covariance matrix that are computed by the standard statistical procedure for each imputed data set. The MIANALYZE procedure then derives valid univariate inference for these parameters. With an additional assumption about the population between and within imputation covariance matrices, multivariate inference based on Wald tests can also be derived.

The MODELEFFECTS statement lists the effects to be analyzed, and the CLASS statement lists the classification variables in the MODELEFFECTS statement. The variables in the MODELEFFECTS statement that are not specified in a CLASS statement are assumed to be continuous.

When each effect in the MODELEFFECTS statement is a continuous variable by itself, a STDERR statement specifies the standard errors when both parameter estimates and associated standard errors are stored as variables in the same data set.

For some parameters of interest, you can use TEST statements to test linear hypotheses about the parameters. For others, it is not straightforward to compute estimates and associated covariance matrices with standard statistical SAS procedures. Examples include correlation coefficients between two variables and ratios of variable means. These special cases are described in the section "[Examples of the Complete-Data Inferences](#)" on page 4654.

## Getting Started: MIANALYZE Procedure

The Fitness data described in the REG procedure are measurements of 31 individuals in a physical fitness course. See Chapter 76, “The REG Procedure,” for more information. The Fitness1 data set is constructed from the Fitness data set and contains three variables: Oxygen, RunTime, and RunPulse. Some values have been set to missing, and the resulting data set has an arbitrary pattern of missingness in these three variables.

```
*----- Data on Physical Fitness -----*
| These measurements were made on men involved in a physical |
| fitness course at N.C. State University.                    |
| Only selected variables of                                  |
| Oxygen (oxygen intake, ml per kg body weight per minute), |
| Runtime (time to run 1.5 miles in minutes), and            |
| RunPulse (heart rate while running) are used.              |
| Certain values were changed to missing for the analysis.   |
*-----*
data Fitness1;
    input Oxygen RunTime RunPulse @@;
    datalines;
44.609 11.37 178      45.313 10.07 185
54.297 8.65 156      59.571 . .
49.874 9.22 .        44.811 11.63 176
.      11.95 176      . 10.85 .
39.442 13.08 174     60.055 8.63 170
50.541 . .          37.388 14.03 186
44.754 11.12 176     47.273 . .
51.855 10.33 166     49.156 8.95 180
40.836 10.95 168     46.672 10.00 .
46.774 10.25 .       50.388 10.08 168
39.407 12.63 174     46.080 11.17 156
45.441 9.63 164      . 8.92 .
45.118 11.08 .       39.203 12.88 168
45.790 10.47 186     50.545 9.93 148
48.673 9.40 186      47.920 11.50 170
47.467 10.50 170
;
```

Suppose that the data are multivariate normally distributed and that the missing data are missing at random (see the “Statistical Assumptions for Multiple Imputation” section in the chapter “The MI Procedure” for a description of these assumptions). The following statements use the MI procedure to impute missing values for the Fitness1 data set:

```
proc mi data=Fitness1 seed=3237851 noprint out=outmi;
    var Oxygen RunTime RunPulse;
run;
```

The MI procedure creates imputed data sets, which are stored in the outmi data set. A variable named `_Imputation_` indicates the imputation numbers. Based on  $m$  imputations,  $m$  different sets of the point and variance estimates for a parameter can be computed. In this example,  $m = 5$  is the default.

The following statements generate regression coefficients for each of the five imputed data sets:

```
proc reg data=outmi outest=outreg covout noprint;
  model Oxygen= RunTime RunPulse;
  by _Imputation_;
run;
```

The following statements display (in [Figure 57.1](#)) output parameter estimates and covariance matrices from PROC REG for the first two imputed data sets:

```
proc print data=outreg (obs=8);
  var _Imputation_ _Type_ _Name_
      Intercept RunTime RunPulse;
  title 'Parameter Estimates from Imputed Data Sets';
run;
```

**Figure 57.1** Parameter Estimates

| Parameter Estimates from Imputed Data Sets |              |        |           |           |          |          |
|--|--------------|--------|-----------|-----------|----------|----------|
| Obs  | _Imputation_ | _TYPE_ | _NAME_    | Intercept | RunTime  | RunPulse |
| 1  | 1            | PARMS  |           | 86.544    | -2.82231 | -0.05873 |
| 2  | 1            | COV    | Intercept | 100.145   | -0.53519 | -0.55077 |
| 3  | 1            | COV    | RunTime   | -0.535    | 0.10774  | -0.00345 |
| 4  | 1            | COV    | RunPulse  | -0.551    | -0.00345 | 0.00343  |
| 5  | 2            | PARMS  |           | 83.021    | -3.00023 | -0.02491 |
| 6  | 2            | COV    | Intercept | 79.032    | -0.66765 | -0.41918 |
| 7  | 2            | COV    | RunTime   | -0.668    | 0.11456  | -0.00313 |
| 8  | 2            | COV    | RunPulse  | -0.419    | -0.00313 | 0.00264  |

The following statements combine the five sets of regression coefficients:

```
proc mianalyze data=outreg;
  modeleffects Intercept RunTime RunPulse;
run;
```

The “Model Information” table in [Figure 57.2](#) lists the input data set(s) and the number of imputations.

**Figure 57.2** Model Information Table

| The MIANALYZE Procedure |             |
|-------------------------|-------------|
| Model Information       |             |
| Data Set                | WORK.OUTREG |
| Number of Imputations   | 5           |

The “Variance Information” table in [Figure 57.3](#) displays the between-imputation, within-imputation, and total variances for combining complete-data inferences. It also displays the degrees of freedom for the total variance, the relative increase in variance due to missing values, the fraction of missing information, and the relative efficiency for each parameter estimate.

**Figure 57.3** Variance Information Table

| Variance Information |                    |           |            |        |
|----------------------|--------------------|-----------|------------|--------|
| Parameter            | -----Variance----- |           |            | DF     |
|                      | Between            | Within    | Total      |        |
| Intercept            | 45.529229          | 76.543614 | 131.178689 | 23.059 |
| RunTime              | 0.019390           | 0.106220  | 0.129487   | 123.88 |
| RunPulse             | 0.001007           | 0.002537  | 0.003746   | 38.419 |

  

| Variance Information |                               |                              |                     |
|----------------------|-------------------------------|------------------------------|---------------------|
| Parameter            | Relative Increase in Variance | Fraction Missing Information | Relative Efficiency |
|                      |                               |                              |                     |
| Intercept            | 0.713777                      | 0.461277                     | 0.915537            |
| RunTime              | 0.219051                      | 0.192620                     | 0.962905            |
| RunPulse             | 0.476384                      | 0.355376                     | 0.933641            |

The “Parameter Estimates” table in [Figure 57.4](#) displays a combined estimate and standard error for each regression coefficient (parameter). Inferences are based on  $t$  distributions. The table displays a 95% confidence interval and a  $t$  test with the associated  $p$ -value for the hypothesis that the parameter is equal to the value specified with the THETA0= option (in this case, zero by default). The minimum and maximum parameter estimates from the imputed data sets are also displayed.

**Figure 57.4** Parameter Estimates

| Parameter Estimates |           |           |                       |          |        |
|---------------------|-----------|-----------|-----------------------|----------|--------|
| Parameter           | Estimate  | Std Error | 95% Confidence Limits |          | DF     |
| Intercept           | 90.837440 | 11.453327 | 67.14779              | 114.5271 | 23.059 |
| RunTime             | -3.032870 | 0.359844  | -3.74511              | -2.3206  | 123.88 |
| RunPulse            | -0.068578 | 0.061204  | -0.19243              | 0.0553   | 38.419 |

  

| Parameter Estimates |           |            |
|---------------------|-----------|------------|
| Parameter           | Minimum   | Maximum    |
| Intercept           | 83.020730 | 100.839807 |
| RunTime             | -3.204426 | -2.822311  |
| RunPulse            | -0.112840 | -0.024910  |

  

| Parameter Estimates |           |                  |         |
|---------------------|-----------|------------------|---------|
| Parameter           | t for H0: |                  |         |
|                     | Theta0    | Parameter=Theta0 | Pr >  t |
| Intercept           | 0         | 7.93             | <.0001  |
| RunTime             | 0         | -8.43            | <.0001  |
| RunPulse            | 0         | -1.12            | 0.2695  |

## Syntax: MIANALYZE Procedure

The following statements are available in PROC MIANALYZE:

```
PROC MIANALYZE < options > ;
  BY variables ;
  CLASS variables ;
  MODELEFFECTS effects ;
  < label > TEST equation1 < , ... , < equationk > > < / options > ;
  STDERR variables ;
```

The BY statement specifies groups in which separate analyses are performed.

The CLASS statement lists the classification variables in the MODELEFFECTS statement. Classification variables can be either character or numeric.

The required MODELEFFECTS statement lists the effects to be analyzed. The variables in the statement that are not specified in a CLASS statement are assumed to be continuous.

The STDERR statement lists the standard errors associated with the effects in the MODELEFFECTS statement when both parameter estimates and standard errors are saved as variables in the same DATA= data set. The STDERR statement can be used only when each effect in the MODELEFFECTS statement is a continuous variable by itself.

The TEST statement tests linear hypotheses about the parameters. An  $F$  statistic is used to jointly test the null hypothesis ( $H_0 : \mathbf{L}_c = \mathbf{c}$ ) specified in a single TEST statement. Several TEST statements can be used.

The PROC MIANALYZE and MODELEFFECTS statements are required for the MIANALYZE procedure. The rest of this section provides detailed syntax information for each of these statements, beginning with the PROC MIANALYZE statement. The remaining statements are in alphabetical order.

## PROC MIANALYZE Statement

```
PROC MIANALYZE < options > ;
```

Table 57.1 summarizes the options in the PROC MIANALYZE statement.

**Table 57.1** Summary of PROC MIANALYZE Options

| Option                 | Description  |
|------------------------|--|
| <b>Input Data Sets</b> |  |
| DATA=                  | Specifies the COV, CORR, or EST type data set                      |
| DATA=                  | Specifies the data set for parameter estimates and standard errors |
| PARMS=                 | Specifies the data set for parameter estimates                     |
| PARMINFO=              | Specifies the data set for parameter information                   |
| COVB=                  | Specifies the data set for covariance matrices                     |

Table 57.1 *continued*

| Option                      | Description                                       |
|-----------------------------|---|
| XPXI=                       | Specifies the data set for $(X'X)^{-1}$ matrices  |
| <b>Statistical Analysis</b> |   |
| THETA0=                     | Specifies parameters under the null hypothesis    |
| ALPHA=                      | Specifies the level for the confidence interval   |
| EDF=                        | Specifies the complete-data degrees of freedom    |
| <b>Printed Output</b>       |   |
| WCOV                        | Displays the within-imputation covariance matrix  |
| BCOV                        | Displays the between-imputation covariance matrix |
| TCOV                        | Displays the total covariance matrix              |
| MULT                        | Displays multivariate inferences                  |

The following options can be used in the PROC MIANALYZE statement. They are listed in alphabetical order.

**ALPHA= $\alpha$** 

specifies that confidence limits are to be constructed for the parameter estimates with confidence level  $100(1 - \alpha)\%$ , where  $0 < \alpha < 1$ . The default is ALPHA=0.05.

**BCOV**

displays the between-imputation covariance matrix.

**COVB <(EFFECTVAR=STACKING | ROWCOL)> =SAS-data-set**

names an input SAS data set that contains covariance matrices of the parameter estimates from imputed data sets. If you provide a COVB= data set, you must also provide a PARMS= data set.

The EFFECTVAR= option identifies the variables for parameters displayed in the covariance matrix and is used only when the PARMINFO= option is not specified. The default is EFFECTVAR= STACKING.

See the section “[Input Data Sets](#)” on page 4646 for a detailed description of the COVB= option.

**DATA=SAS-data-set**

names an input SAS data set.

If the input DATA= data set is not a specially structured SAS data set, the data set contains both the parameter estimates and associated standard errors. The parameter estimates are specified in the MODELEFFECTS statement and the standard errors are specified in the STDERR statement.

If the data set is a specially structured input SAS data set, it must have a TYPE of EST, COV, or CORR that contains estimates from imputed data sets:

- If TYPE=EST, the data set contains the parameter estimates and associated covariance matrices.
- If TYPE=COV, the data set contains the sample means, sample sizes, and covariance matrices. Each covariance matrix for variables is divided by the sample size  $n$  to create the covariance matrix for parameter estimates.

- If TYPE=CORR, the data set contains the sample means, sample sizes, standard errors, and correlation matrices. The covariance matrices are computed from the correlation matrices and associated standard errors. Each covariance matrix for variables is divided by the sample size  $n$  to create the covariance matrix for parameter estimates.

If you do not specify an input data set with the DATA= or PARMS= option, then the most recently created SAS data set is used as an input DATA= data set. See the section “[Input Data Sets](#)” on page 4646 for a detailed description of the input data sets.

**EDF=number**

specifies the complete-data degrees of freedom for the parameter estimates. This is used to compute an adjusted degrees of freedom for each parameter estimate. By default, EDF= $\infty$  and the degrees of freedom for each parameter estimate are not adjusted.

**MULT**

**MULTIVARIATE**

requests multivariate inference for the parameters. It is based on Wald tests and is a generalization of the univariate inference. See the section “[Multivariate Inferences](#)” on page 4652 for a detailed description of the multivariate inference.

**PARMINFO=SAS-data-set**

names an input SAS data set that contains parameter information associated with variables PRM1, PRM2, . . . , and so on. These variables are used as variables for parameters in a COVB= data set. See the section “[Input Data Sets](#)” on page 4646 for a detailed description of the PARMINFO= option.

**PARMS <(CLASSVAR= ctype)> =SAS-data-set**

names an input SAS data set that contains parameter estimates computed from imputed data sets. When a COVB= data set is not specified, the input PARMS= data set also contains standard errors associated with these parameter estimates. If multivariate inference is requested, you must also provide a COVB= or XPXI= data set.

When the effects contain classification variables, the option CLASSVAR= *ctype* can be used to identify the associated classification variables when reading the classification levels from observations. The available types are FULL, LEVEL, and CLASSVAL. The default is CLASSVAR= FULL. See the section “[Input Data Sets](#)” on page 4646 for a detailed description of the PARMS= option.

**TCOV**

displays the total covariance matrix derived by assuming that the population between-imputation and within-imputation covariance matrices are proportional to each other.

**THETA0=numbers**

**MU0=numbers**

specifies the parameter values  $\theta_0$  under the null hypothesis  $\theta = \theta_0$  in the  $t$  tests for location for the effects. If only one number  $\theta_0$  is specified, that number is used for all effects. If more than one number is specified, the specified numbers correspond to effects in the MODELEFFECTS statement in the order in which they appear in the statement. When an effect contains classification variables, the corresponding value is not used and the test is not performed.

**WCOV**

displays the within-imputation covariance matrices.

**XPXI=SAS-data-set**

names an input SAS data set that contains the  $(X'X)^{-1}$  matrices associated with the parameter estimates computed from imputed data sets. If you provide an XPXI= data set, you must also provide a PARMS= data set. In this case, PROC MIANALYZE reads the standard errors of the estimates from the PARMS= data. The standard errors and  $(X'X)^{-1}$  matrices are used to derive the covariance matrices.

---

## BY Statement

**BY variables ;**

You can specify a BY statement with PROC MIANALYZE 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 MIANALYZE 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 ;**

The CLASS statement specifies the classification variables in the MODELEFFECTS statement. Classification variables can be either character or numeric. Classification levels are determined from the formatted values of the classification variables. See “The FORMAT Procedure” in the *Base SAS Procedures Guide* for details.

---

## MODELEFFECTS Statement

**MODELEFFECTS** *effects* ;

The MODELEFFECTS statement lists the effects in the data set to be analyzed. Each effect is a variable or a combination of variables, and is specified with a special notation that uses variable names and operators.

Each variable is either a classification (or CLASS) variable or a continuous variable. If a variable is not declared in the CLASS statement, it is assumed to be continuous. Crossing and nesting operators can be used in an effect to create crossed and nested effects.

One general form of an effect involving several variables is

$$X1 * X2 * A * B * C ( D E )$$

where A, B, C, D, and E are classification variables and X1 and X2 are continuous variables.

When the input DATA= data set is not a specially structured SAS data set, you must also specify standard errors of the parameter estimates in an STDERR statement.

---

## STDERR Statement

**STDERR** *variables* ;

The STDERR statement lists standard errors associated with effects in the MODELEFFECTS statement, when the input DATA= data set contains both parameter estimates and standard errors as variables in the data set.

With the STDERR statement, only continuous effects are allowed in the MODELEFFECTS statement. The specified standard errors correspond to parameter estimates in the order in which they appear in the MODELEFFECTS statement.

For example, you can use the following MODELEFFECTS and STDERR statements to identify both the parameter estimates and associated standard errors in a SAS data set:

```
proc mianalyze;
  modeleffects y1-y3;
  stderr sy1-sy3;
run;
```

---

## TEST Statement

*<label>* **TEST** *equation1 <, ..., <equationk> > </options>* ;

The TEST statement tests linear hypotheses about the parameters  $\beta$ . An  $F$  test is used to jointly test the null hypotheses ( $H_0 : L\beta = c$ ) specified in a single TEST statement in which the MULT option is specified.

Each *equation* specifies a linear hypothesis (a row of the **L** matrix and the corresponding element of the **c** vector); multiple *equations* are separated by commas. The label, which must be a valid SAS name, is used to identify the resulting output. You can submit multiple TEST statements. When a label is not included in a TEST statement, a label of “Test *j*” is used for the *j*th TEST statement.

The form of an *equation* is as follows:

$$\text{term} < \pm \text{term} \dots > < = \pm \text{term} < \pm \text{term} \dots > >$$

where *term* is a parameter of the model, or a constant, or a constant times a parameter. When no equal sign appears, the expression is set to 0. Only parameters for regressor effects (continuous variables by themselves) are allowed.

For each TEST statement, PROC MIANALYZE displays a “Test Specification” table of the **L** matrix and the **c** vector. The procedure also displays a “Variance Information” table of the between-imputation, within-imputation, and total variances for combining complete-data inferences, and a “Parameter Estimates” table of a combined estimate and standard error for each linear component. The linear components are labeled TestPrm1, TestPrm2, ... in the tables.

The following statements illustrate possible uses of the TEST statement:

```
proc mianalyze;
  modeleffects intercept a1 a2 a3;
  test1: test intercept + a2 = 0;
  test2: test intercept + a2;
  test3: test a1=a2=a3;
  test4: test a1=a2, a2=a3;
run;
```

The first and second TEST statements are equivalent and correspond to the specification in [Figure 57.5](#).

**Figure 57.5** Test Specification for test1 and test2

| The MIANALYZE Procedure |           |    |          |    |   |
|-------------------------|-----------|----|----------|----|---|
| Test: test1             |           |    |          |    |   |
| Test Specification      |           |    |          |    |   |
| -----L Matrix-----      |           |    |          |    |   |
| Parameter               | intercept | a1 | a2       | a3 | C |
| TestPrm1                | 1.000000  | 0  | 1.000000 | 0  | 0 |

The third and fourth TEST statements are also equivalent and correspond to the specification in [Figure 57.6](#).

**Figure 57.6** Test Specification for test3 and test4

| The MIANALYZE Procedure |           |          |           |           |   |
|-------------------------|-----------|----------|-----------|-----------|---|
| Test: test3             |           |          |           |           |   |
| Test Specification      |           |          |           |           |   |
| -----L Matrix-----      |           |          |           |           |   |
| Parameter               | intercept | a1       | a2        | a3        | C |
| TestPrm1                | 0         | 1.000000 | -1.000000 | 0         | 0 |
| TestPrm2                | 0         | 0        | 1.000000  | -1.000000 | 0 |

The ALPHA= and EDF options specified in the PROC MIANALYZE statement are also applied to the TEST statement. You can specify the following options in the TEST statement after a slash(/):

**BCOV**

displays the between-imputation covariance matrix.

**MULT**

displays the multivariate inference for parameters.

**TCOV**

displays the total covariance matrix.

**WCOV**

displays the within-imputation covariance matrix.

For more information, see the section “[Testing Linear Hypotheses about the Parameters](#)” on page 4654.

---

## Details: MIANALYZE Procedure

---

### Input Data Sets

You specify input data sets based on the type of inference you requested. For univariate inference, you can use one of the following options:

- a DATA= data set, which provides both parameter estimates and the associated standard errors
- a DATA=EST, COV, or CORR data set, which provides both parameter estimates and the associated standard errors either explicitly (type CORR) or through the covariance matrix (type EST, COV)
- PARMS= data set, which provides both parameter estimates and the associated standard errors

For multivariate inference, which includes the testing of linear hypotheses about parameters, you can use one of the following option combinations:

- a DATA=EST, COV, or CORR data set, which provides parameter estimates and the associated covariance matrix either explicitly (type EST, COV) or through the correlation matrix and standard errors (type CORR) in a single data set
- PARMS= and COVB= data sets, which provide parameter estimates in a PARMS= data set and the associated covariance matrix in a COVB= data set
- PARMS=, COVB=, and PARMINFO= data sets, which provide parameter estimates in a PARMS= data set, the associated covariance matrix in a COVB= data set with variables named PRM1, PRM2, ..., and the effects associated with these variables in a PARMINFO= data set
- PARMS= and XPXI= data sets, which provide parameter estimates and the associated standard errors in a PARMS= data set and the associated  $(X'X)^{-1}$  matrix in an XPXI= data set

The appropriate combination depends on the type of inference and the SAS procedure you used to create the data sets. For instance, if you used PROC REG to create an OUTEST= data set that contains the parameter estimates and covariance matrix, you would use the DATA= option to read the OUTEST= data set.

When the input DATA= data set is a specially structured SAS data set, the data set must contain the variable `_Imputation_` to identify the imputation by number. Otherwise, each observation corresponds to an imputation and contains both parameter estimates and associated standard errors.

If you do not specify an input data set with the DATA= or PARMS= option, then the most recently created SAS data set is used as an input DATA= data set. Note that with a DATA= data set, each effect represents a continuous variable; only regressor effects (continuous variables by themselves) are allowed in the MODEL-EFFECTS statement.

## DATA= SAS Data Set

The DATA= data set provides both parameter estimates and the associated standard errors computed from imputed data sets. Such data sets are typically created with an OUTPUT statement in procedures such as PROC MEANS and PROC UNIVARIATE.

The MIANALYZE procedure reads parameter estimates from observations with variables in the MODEL-EFFECTS statement, and standard errors for parameter estimates from observations with variables in the STDERR statement. The order of the variables for standard errors must match the order of the variables for parameter estimates.

## DATA=EST, COV, or CORR SAS Data Set

The specially structured DATA= data set provides both parameter estimates and the associated covariance matrix computed from imputed data sets. Such data sets are created by procedures such as PROC CORR (type COV, CORR) and PROC REG (type EST).

With TYPE=EST, the MIANALYZE procedure reads parameter estimates from observations with \_TYPE\_='PARM', \_TYPE\_='PARMS', \_TYPE\_='OLS', or \_TYPE\_='FINAL', and covariance matrices for parameter estimates from observations with \_TYPE\_='COV' or \_TYPE\_='COVB'.

With TYPE=COV, the procedure reads sample means from observations with \_TYPE\_='MEAN', sample size  $n$  from observations with \_TYPE\_='N', and covariance matrices for variables from observations with \_TYPE\_='COV'.

With TYPE=CORR, the procedure reads sample means from observations with \_TYPE\_='MEAN', sample size  $n$  from observations with \_TYPE\_='N', correlation matrices for variables from observations with \_TYPE\_='CORR', and standard errors for variables from observations with \_TYPE\_='STD'. The standard errors and correlation matrix are used to generate a covariance matrix for the variables.

Note that with TYPE=COV or CORR, each covariance matrix for the variables is divided by  $n$  to create the covariance matrix for the sample means.

## PARMS <(CLASSVAR= *ctype*)> = Data Set

The PARMS= data set contains both parameter estimates and the associated standard errors computed from imputed data sets. Such data sets are typically created with an ODS OUTPUT statement in procedures such as PROC GENMOD, PROC GLM, PROC LOGISTIC, and PROC MIXED.

The MIANALYZE procedure reads effect names from observations with the variable Parameter, Effect, Variable, or Parm. It then reads parameter estimates from observations with the variable Estimate and standard errors for parameter estimates from observations with the variable StdErr.

When the effects contain classification variables, the option CLASSVAR= *ctype* can be used to identify associated classification variables when reading the classification levels from observations. The available types are FULL, LEVEL, and CLASSVAL. The default is CLASSVAR= FULL.

With CLASSVAR=FULL, the data set contains the classification variables explicitly. PROC MIANALYZE reads the classification levels from observations with their corresponding classification variables. PROC MIXED generates this type of table.

With CLASSVAR=LEVEL, PROC MIANALYZE reads the classification levels for the effect from observations with variables Level1, Level2, and so on, where the variable Level1 contains the classification level for the first classification variable in the effect, and the variable Level2 contains the classification level for the second classification variable in the effect. For each effect, the variables in the crossed list are displayed before the variables in the nested list. The variable order in the CLASS statement is used for variables inside each list. PROC GENMOD generates this type of table.

For example, with the following statements, the variable Level1 has the classification level of the variable c2 for the effect c2:

```
proc mianalyze parms(classvar=Level)= dataparm;
  class c1 c2 c3;
  modeleffects c2 c3(c2 c1);
run;
```

For the effect c3(c2 c1), the variable Level1 has the classification level of the variable c3, Level2 has the level of c1, and Level3 has the level of c2.

Similarly, with CLASSVAR=CLASSVAL, PROC MIANALYZE reads the classification levels for the effect from observations with variables ClassVal0, ClassVal1, and so on, where the variable ClassVal0 contains the classification level for the first classification variable in the effect, and the variable ClassVal1 contains the classification level for the second classification variable in the effect. For each effect, the variables in the crossed list are displayed before the variables in the nested list. The variable order in the CLASS statement is used for variables inside each list. PROC LOGISTIC generates this type of tables.

## PARMS <(CLASSVAR= *ctype*)> = and COVB= Data Sets

The PARMS= data set contains parameter estimates, and the COVB= data set contains associated covariance matrices computed from imputed data sets. Such data sets are typically created with an ODS OUTPUT statement in procedures such as PROC LOGISTIC, PROC MIXED, and PROC REG.

With a PARMS= data set, the MIANALYZE procedure reads effect names from observations with the variable Parameter, Effect, Variable, or Parm. It then reads parameter estimates from observations with the variable Estimate.

When the effects contain classification variables, the option CLASSVAR= *ctype* can be used to identify the associated classification variables when reading the classification levels from observations. The available types are FULL, LEVEL, and CLASSVAL, and they are described in the section “PARMS <(CLASSVAR= *ctype*)> = Data Set” on page 4648. The default is CLASSVAR= FULL.

The option EFFECTVAR= *etype* identifies the variables for parameters displayed in the covariance matrix. The available types are STACKING and ROWCOL. The default is EFFECTVAR=STACKING.

With EFFECTVAR=STACKING, each parameter is displayed by stacking variables in the effect. Begin with the variables in the crossed list, followed by the continuous list, then followed by the nested list. Each classification variable is displayed with its classification level attached. PROC LOGISTIC generates this type of table.

When each effect is a continuous variable by itself, each stacked parameter name reduces to the effect name. PROC REG generates this type of table.

With EFFECTVAR=STACKING, the MIANALYZE procedure reads parameter names from observations with the variable Parameter, Effect, Variable, Parm, or RowName. It then reads covariance matrices from observations with the stacked variables in a COVB= data set.

With EFFECTVAR=ROWCOL, parameters are displayed by the variables Col1, Col2, ... The parameter associated with the variable Col1 is identified by the observation with value 1 for the variable Row. The parameter associated with the variable Col2 is identified by the observation with value 2 for the variable Row. PROC MIXED generates this type of table.

With EFFECTVAR=ROWCOL, the MIANALYZE procedure reads the parameter indices from observations with the variable Row and the effect names from observations with the variable Parameter, Effect, Variable, Parm, or RowName. It then reads covariance matrices from observations with the variables Col1, Col2, and so on in a COVB= data set.

When the effects contain classification variables, the data set contains the classification variables explicitly and the MIANALYZE procedure also reads the classification levels from their corresponding classification variables.

## PARMS <(CLASSVAR= *ctype*)> =, PARMINFO=, and COVB= Data Sets

The input PARMS= data set contains parameter estimates and the input COVB= data set contains associated covariance matrices computed from imputed data sets. Such data sets are typically created with an ODS OUTPUT statement using procedure such as PROC GENMOD.

With a PARMS= data set, the MIANALYZE procedure reads effect names from observations with the variable Parameter, Effect, Variable, or Parm. It then reads parameter estimates from observations with the variable Estimate.

When the effects contain classification variables, the option CLASSVAR= *ctype* can be used to identify the associated classification variables when reading the classification levels from observations. The available types are FULL, LEVEL, and CLASSVAL, and they are described in the section “PARMS <(CLASSVAR= *ctype*)> = Data Set” on page 4648. The default is CLASSVAR= FULL.

With a COVB= data set, the MIANALYZE procedure reads parameter names from observations with the variable Parameter, Effect, Variable, Parm, or RowName. It then reads covariance matrices from observations with the variables Prm1, Prm2, and so on.

The parameters associated with the variables Prm1, Prm2, and so on are identified in the PARMINFO= data set. PROC MIANALYZE reads the parameter names from observations with the variable Parameter and the corresponding effect from observations with the variable Effect. When the effects contain classification variables, the data set contains the classification variables explicitly and the MIANALYZE procedure also reads the classification levels from observations with their corresponding classification variables.

## PARMS= and XPXI= Data Sets

The input PARMS= data set contains parameter estimates, and the input XPXI= data set contains associated  $(X'X)^{-1}$  matrices computed from imputed data sets. Such data sets are typically created with an ODS OUTPUT statement in a procedure such as PROC GLM.

With a PARMS= data set, the MIANALYZE procedure reads parameter names from observations with the variable Parameter, Effect, Variable, or Parm. It then reads parameter estimates from observations with the variable Estimate and standard errors for parameter estimates from observations with the variable StdErr.

With a XPXI= data set, the MIANALYZE procedure reads parameter names from observations with the variable Parameter and  $(X'X)^{-1}$  matrices from observations with the parameter variables in the data set.

Note that this combination can be used only when each effect is a continuous variable by itself.

---

## Combining Inferences from Imputed Data Sets

With  $m$  imputations,  $m$  different sets of the point and variance estimates for a parameter  $Q$  can be computed. Suppose that  $\hat{Q}_i$  and  $\hat{W}_i$  are the point and variance estimates, respectively, from the  $i$ th imputed data set,  $i = 1, 2, \dots, m$ . Then the combined point estimate for  $Q$  from multiple imputation is the average of the  $m$

complete-data estimates:

$$\bar{Q} = \frac{1}{m} \sum_{i=1}^m \hat{Q}_i$$

Suppose that  $\bar{W}$  is the within-imputation variance, which is the average of the  $m$  complete-data estimates:

$$\bar{W} = \frac{1}{m} \sum_{i=1}^m \hat{W}_i$$

And suppose that  $B$  is the between-imputation variance:

$$B = \frac{1}{m-1} \sum_{i=1}^m (\hat{Q}_i - \bar{Q})^2$$

Then the variance estimate associated with  $\bar{Q}$  is the total variance (Rubin 1987)

$$T = \bar{W} + (1 + \frac{1}{m})B$$

The statistic  $(Q - \bar{Q})T^{-(1/2)}$  is approximately distributed as  $t$  with  $v_m$  degrees of freedom (Rubin 1987), where

$$v_m = (m-1) \left[ 1 + \frac{\bar{W}}{(1 + m^{-1})B} \right]^2$$

The degrees of freedom  $v_m$  depend on  $m$  and the ratio

$$r = \frac{(1 + m^{-1})B}{\bar{W}}$$

The ratio  $r$  is called the relative increase in variance due to nonresponse (Rubin 1987). When there is no missing information about  $Q$ , the values of  $r$  and  $B$  are both zero. With a large value of  $m$  or a small value of  $r$ , the degrees of freedom  $v_m$  will be large and the distribution of  $(Q - \bar{Q})T^{-(1/2)}$  will be approximately normal.

Another useful statistic is the fraction of missing information about  $Q$ :

$$\hat{\lambda} = \frac{r + 2/(v_m + 3)}{r + 1}$$

Both statistics  $r$  and  $\lambda$  are helpful diagnostics for assessing how the missing data contribute to the uncertainty about  $Q$ .

When the complete-data degrees of freedom  $v_0$  are small, and there is only a modest proportion of missing data, the computed degrees of freedom,  $v_m$ , can be much larger than  $v_0$ , which is inappropriate. For example, with  $m = 5$  and  $r = 10\%$ , the computed degrees of freedom  $v_m = 484$ , which is inappropriate for data sets with complete-data degrees of freedom less than 484.

Barnard and Rubin (1999) recommend the use of adjusted degrees of freedom

$$v_m^* = \left[ \frac{1}{v_m} + \frac{1}{\hat{v}_{obs}} \right]^{-1}$$

where  $\hat{v}_{obs} = (1 - \gamma) v_0(v_0 + 1)/(v_0 + 3)$  and  $\gamma = (1 + m^{-1})B/T$ .

If you specify the complete-data degrees of freedom  $v_0$  with the EDF= option, the MIANALYZE procedure uses the adjusted degrees of freedom,  $v_m^*$ , for inference. Otherwise, the degrees of freedom  $v_m$  are used.

## Multiple Imputation Efficiency

The relative efficiency (RE) of using the finite  $m$  imputation estimator, rather than using an infinite number for the fully efficient imputation, in units of variance, is approximately a function of  $m$  and  $\lambda$  (Rubin 1987, p. 114):

$$RE = \left(1 + \frac{\lambda}{m}\right)^{-1}$$

Table 57.2 shows relative efficiencies with different values of  $m$  and  $\lambda$ .

**Table 57.2** Relative Efficiencies

| m  | $\lambda$ |        |        |        |        |
|----|-----------|--------|--------|--------|--------|
|    | 10%       | 20%    | 30%    | 50%    | 70%    |
| 3  | 0.9677    | 0.9375 | 0.9091 | 0.8571 | 0.8108 |
| 5  | 0.9804    | 0.9615 | 0.9434 | 0.9091 | 0.8772 |
| 10 | 0.9901    | 0.9804 | 0.9709 | 0.9524 | 0.9346 |
| 20 | 0.9950    | 0.9901 | 0.9852 | 0.9756 | 0.9662 |

The table shows that for situations with little missing information, only a small number of imputations are necessary. In practice, the number of imputations needed can be informally verified by replicating sets of  $m$  imputations and checking whether the estimates are stable between sets (Horton and Lipsitz 2001, p. 246).

## Multivariate Inferences

Multivariate inference based on Wald tests can be done with  $m$  imputed data sets. The approach is a generalization of the approach taken in the univariate case (Rubin 1987, p. 137; Schafer 1997, p. 113). Suppose that  $\hat{\mathbf{Q}}_i$  and  $\hat{\mathbf{W}}_i$  are the point and covariance matrix estimates for a  $p$ -dimensional parameter  $\mathbf{Q}$  (such as a multivariate mean) from the  $i$ th imputed data set,  $i = 1, 2, \dots, m$ . Then the combined point estimate for  $\mathbf{Q}$  from the multiple imputation is the average of the  $m$  complete-data estimates:

$$\bar{\mathbf{Q}} = \frac{1}{m} \sum_{i=1}^m \hat{\mathbf{Q}}_i$$

Suppose that  $\overline{\mathbf{W}}$  is the within-imputation covariance matrix, which is the average of the  $m$  complete-data estimates:

$$\overline{\mathbf{W}} = \frac{1}{m} \sum_{i=1}^m \hat{\mathbf{W}}_i$$

And suppose that  $\mathbf{B}$  is the between-imputation covariance matrix:

$$\mathbf{B} = \frac{1}{m-1} \sum_{i=1}^m (\hat{\mathbf{Q}}_i - \overline{\mathbf{Q}})(\hat{\mathbf{Q}}_i - \overline{\mathbf{Q}})'$$

Then the covariance matrix associated with  $\overline{\mathbf{Q}}$  is the total covariance matrix

$$\mathbf{T}_0 = \overline{\mathbf{W}} + (1 + \frac{1}{m})\mathbf{B}$$

The natural multivariate extension of the  $t$  statistic used in the univariate case is the  $F$  statistic

$$F_0 = (\mathbf{Q} - \overline{\mathbf{Q}})' \mathbf{T}_0^{-1} (\mathbf{Q} - \overline{\mathbf{Q}})$$

with degrees of freedom  $p$  and

$$v = (m-1)(1 + 1/r)^2$$

where

$$r = (1 + \frac{1}{m}) \text{trace}(\mathbf{B}\overline{\mathbf{W}}^{-1})/p$$

is an average relative increase in variance due to nonresponse (Rubin 1987, p. 137; Schafer 1997, p. 114).

However, the reference distribution of the statistic  $F_0$  is not easily derived. Especially for small  $m$ , the between-imputation covariance matrix  $\mathbf{B}$  is unstable and does not have full rank for  $m \leq p$  (Schafer 1997, p. 113).

One solution is to make an additional assumption that the population between-imputation and within-imputation covariance matrices are proportional to each other (Schafer 1997, p. 113). This assumption implies that the fractions of missing information for all components of  $\mathbf{Q}$  are equal. Under this assumption, a more stable estimate of the total covariance matrix is

$$\mathbf{T} = (1 + r)\overline{\mathbf{W}}$$

With the total covariance matrix  $\mathbf{T}$ , the  $F$  statistic (Rubin 1987, p. 137)

$$F = (\mathbf{Q} - \overline{\mathbf{Q}})' \mathbf{T}^{-1} (\mathbf{Q} - \overline{\mathbf{Q}})/p$$

has an  $F$  distribution with degrees of freedom  $p$  and  $v_1$ , where

$$v_1 = \frac{1}{2}(p+1)(m-1)(1 + \frac{1}{r})^2$$

For  $t = p(m-1) \leq 4$ , PROC MIANALYZE uses the degrees of freedom  $v_1$  in the analysis. For  $t = p(m-1) > 4$ , PROC MIANALYZE uses  $v_2$ , a better approximation of the degrees of freedom given by Li, Raghunathan, and Rubin (1991):

$$v_2 = 4 + (t-4) \left[ 1 + \frac{1}{r} \left( 1 - \frac{2}{t} \right) \right]^2$$

## Testing Linear Hypotheses about the Parameters

Linear hypotheses for parameters  $\beta$  are expressed in matrix form as

$$H_0 : \mathbf{L}\beta = \mathbf{c}$$

where  $\mathbf{L}$  is a matrix of coefficients for the linear hypotheses and  $\mathbf{c}$  is a vector of constants.

Suppose that  $\hat{\mathbf{Q}}_i$  and  $\hat{\mathbf{U}}_i$  are the point and covariance matrix estimates, respectively, for a  $p$ -dimensional parameter  $\mathbf{Q}$  from the  $i$ th imputed data set,  $i=1, 2, \dots, m$ . Then for a given matrix  $\mathbf{L}$ , the point and covariance matrix estimates for the linear functions  $\mathbf{LQ}$  in the  $i$ th imputed data set are, respectively,

$$\begin{aligned} \mathbf{L}\hat{\mathbf{Q}}_i \\ \mathbf{L}\hat{\mathbf{U}}_i\mathbf{L}' \end{aligned}$$

The inferences described in the section “[Combining Inferences from Imputed Data Sets](#)” on page 4650 and the section “[Multivariate Inferences](#)” on page 4652 are applied to these linear estimates for testing the null hypothesis  $H_0 : \mathbf{L}\beta = \mathbf{c}$ .

For each TEST statement, the “Test Specification” table displays the  $\mathbf{L}$  matrix and the  $\mathbf{c}$  vector, the “Variance Information” table displays the between-imputation, within-imputation, and total variances for combining complete-data inferences, and the “Parameter Estimates” table displays a combined estimate and standard error for each linear component.

With the WCOV and BCOV options in the TEST statement, the procedure displays the within-imputation and between-imputation covariance matrices, respectively.

With the TCOV option, the procedure displays the total covariance matrix derived under the assumption that the population between-imputation and within-imputation covariance matrices are proportional to each other.

With the MULT option in the TEST statement, the “Multivariate Inference” table displays an  $F$  test for the null hypothesis  $\mathbf{L}\beta = \mathbf{c}$  of the linear components.

## Examples of the Complete-Data Inferences

For a given parameter of interest, it is not always possible to compute the estimate and associated covariance matrix directly from a SAS procedure. This section describes examples of parameters with their estimates and associated covariance matrices, which provide the input to the MIANALYZE procedure. Some are straightforward, and others require special techniques.

### Means

For a population mean vector  $\mu$ , the usual estimate is the sample mean vector

$$\bar{\mathbf{y}} = \frac{1}{n} \sum \mathbf{y}_i$$

A variance estimate for  $\bar{y}$  is  $\frac{1}{n}\mathbf{S}$ , where  $\mathbf{S}$  is the sample covariance matrix

$$\mathbf{S} = \frac{1}{n-1} \sum (\mathbf{y}_i - \bar{\mathbf{y}})(\mathbf{y}_i - \bar{\mathbf{y}})'$$

These statistics can be computed from a procedure such as PROC CORR. This approach is illustrated in [Example 57.2](#).

## Regression Coefficients

Many SAS procedures are available for regression analysis. Among them, PROC REG provides the most general analysis capabilities, and others like PROC LOGISTIC and PROC MIXED provide more specialized analyses.

Some regression procedures, such as REG and LOGISTIC, create an EST type data set that contains both the parameter estimates for the regression coefficients and their associated covariance matrix. You can read an EST type data set in the MIANALYZE procedure with the DATA= option. This approach is illustrated in [Example 57.3](#).

Other procedures, such as GLM, MIXED, and GENMOD, do not generate EST type data sets for regression coefficients. For PROC MIXED and PROC GENMOD, you can use ODS OUTPUT statement to save parameter estimates in a data set and the associated covariance matrix in a separate data set. These data sets are then read in the MIANALYZE procedure with the PARMS= and COVB= options, respectively. This approach is illustrated in [Example 57.4](#) for PROC MIXED and in [Example 57.5](#) for PROC GENMOD.

PROC GLM does not display tables for covariance matrices. However, you can use the ODS OUTPUT statement to save parameter estimates and associated standard errors in a data set and the associated  $(X'X)^{-1}$  matrix in a separate data set. These data sets are then read in the MIANALYZE procedure with the PARMS= and XPXI= options, respectively. This approach is illustrated in [Example 57.6](#).

For univariate inference, only parameter estimates and associated standard errors are needed. You can use the ODS OUTPUT statement to save parameter estimates and associated standard errors in a data set. This data set is then read in the MIANALYZE procedure with the PARMS= option. This approach is illustrated in [Example 57.4](#).

## Correlation Coefficients

For the population correlation coefficient  $\rho$ , a point estimate is the sample correlation coefficient  $r$ . However, for nonzero  $\rho$ , the distribution of  $r$  is skewed.

The distribution of  $r$  can be normalized through Fisher's  $z$  transformation

$$z(r) = \frac{1}{2} \log \left( \frac{1+r}{1-r} \right)$$

$z(r)$  is approximately normally distributed with mean  $z(\rho)$  and variance  $1/(n-3)$ .

With a point estimate  $\hat{z}$  and an approximate 95% confidence interval  $(z_1, z_2)$  for  $z(\rho)$ , a point estimate  $\hat{r}$  and a 95% confidence interval  $(r_1, r_2)$  for  $\rho$  can be obtained by applying the inverse transformation

$$r = \tanh(z) = \frac{e^{2z} - 1}{e^{2z} + 1}$$

to  $z = \hat{z}, z_1$ , and  $z_2$ .

This approach is illustrated in [Example 57.10](#).

## Ratios of Variable Means

For the ratio  $\mu_1/\mu_2$  of means for variables  $Y_1$  and  $Y_2$ , the point estimate is  $\bar{y}_1/\bar{y}_2$ , the ratio of the sample means. The Taylor expansion and delta method can be applied to the function  $y_1/y_2$  to obtain the variance estimate (Schafer 1997, p. 196)

$$\frac{1}{n} \left[ \left( \frac{\bar{y}_1}{\bar{y}_2} \right)^2 s_{22} - 2 \left( \frac{\bar{y}_1}{\bar{y}_2} \right) \left( \frac{1}{\bar{y}_2} \right) s_{12} + \left( \frac{1}{\bar{y}_2} \right)^2 s_{11} \right]$$

where  $s_{11}$  and  $s_{22}$  are the sample variances of  $Y_1$  and  $Y_2$ , respectively, and  $s_{12}$  is the sample covariance between  $Y_1$  and  $Y_2$ .

A ratio of sample means will be approximately unbiased and normally distributed if the coefficient of variation of the denominator (the standard error for the mean divided by the estimated mean) is 10% or less (Cochran 1977, p. 166; Schafer 1997, p. 196).

## ODS Table Names

PROC MIANALYZE assigns a name to each table it creates. You must use these names to reference tables when using the Output Delivery System (ODS). These names are listed in [Table 57.3](#). For more information about ODS, see Chapter 20, “Using the Output Delivery System.”

**Table 57.3** ODS Tables Produced by PROC MIANALYZE

| ODS Table Name         | Description   | Statement | Option |
|------------------------|---|-----------|--------|
| BCov                   | Between-imputation covariance matrix                                    |           | BCOV   |
| ModelInfo              | Model information   |           |        |
| MultStat               | Multivariate inference  |           | MULT   |
| ParameterEstimates     | Parameter estimates   |           |        |
| TCov                   | Total covariance matrix   |           | TCOV   |
| TestBCov               | Between-imputation covariance matrix for $\mathbf{L}\boldsymbol{\beta}$ | TEST      | BCOV   |
| TestMultStat           | Multivariate inference for $\mathbf{L}\boldsymbol{\beta}$               | TEST      | MULT   |
| TestParameterEstimates | Parameter estimates for $\mathbf{L}\boldsymbol{\beta}$                  | TEST      |        |
| TestSpec               | Test specification, $\mathbf{L}$ and $\mathbf{c}$                       | TEST      |        |
| TestTCov               | Total covariance matrix for $\mathbf{L}\boldsymbol{\beta}$              | TEST      | TCOV   |
| TestVarianceInfo       | Variance information for $\mathbf{L}\boldsymbol{\beta}$                 | TEST      |        |

**Table 57.3** *continued*

| ODS Table Name | Description                                      | Statement | Option |
|----------------|--|-----------|--------|
| TestWCov       | Within-imputation covariance matrix for $L\beta$ | TEST      | WCOV   |
| VarianceInfo   | Variance information                             |           |        |
| WCov           | Within-imputation covariance matrix              |           | WCOV   |

## Examples: MIANALYZE Procedure

The following statements generate five imputed data sets to be used in this section. The data set `Fitness1` was created in the section “[Getting Started: MIANALYZE Procedure](#)” on page 4637. See “[The MI Procedure](#)” chapter for details concerning the MI procedure.

```
proc mi data=Fitness1 seed=3237851 noprint out=outmi;
  var Oxygen RunTime RunPulse;
run;
```

The Fish data described in the STEPDISC procedure are measurements of 159 fish of seven species caught in Finland’s lake Laengelmavesi. For each fish, the length, height, and width are measured. See Chapter 85, “[The STEPDISC Procedure](#),” for more information.

The Fish2 data set is constructed from the Fish data set and contains two species of fish. Some values have been set to missing, and the resulting data set has a monotone missing pattern in the variables Length, Height, Width, and Species.

The following statements create the Fish2 data set. It contains two species of fish in the Fish data set.

```
*-----Fish2 Data-----*
| The data set contains two species of the fish (Bream and Pike) |
| and three measurements: Length, Height, Width.                |
| Some values have been set to missing, and the resulting data set |
| has a monotone missing pattern in the variables                 |
| Length, Height, Width, and Species.                             |
*-----*
data Fish2;
  title 'Fish Measurement Data';
  input Species $ Length Height Width @@;
  datalines;
Bream  30.0  11.520  4.020      .  31.2  12.480  4.306
Bream  31.1  12.378  4.696      Bream  33.5  12.730  4.456
.      34.0  12.444  .          Bream  34.7  13.602  4.927
Bream  34.5  14.180  5.279      Bream  35.0  12.670  4.690
Bream  35.1  14.005  4.844      Bream  36.2  14.227  4.959
.      36.2  14.263  .          Bream  36.2  14.371  4.815
Bream  36.4  13.759  4.368      Bream  37.3  13.913  5.073
Bream  37.2  14.954  5.171      Bream  37.2  15.438  5.580
Bream  38.3  14.860  5.285      Bream  38.5  14.938  5.198
.      38.6  15.633  5.134      Bream  38.7  14.474  5.728
```

```

Bream  39.5  15.129  5.570      .  39.2  15.994  .
Bream  39.7  15.523  5.280    Bream  40.6  15.469  6.131
.      40.5      .      .    Bream  40.9  16.360  6.053
Bream  40.6  16.362  6.090    Bream  41.5  16.517  5.852
Bream  41.6  16.890  6.198    Bream  42.6  18.957  6.603
Bream  44.1  18.037  6.306    Bream  44.0  18.084  6.292
Bream  45.3  18.754  6.750    Bream  45.9  18.635  6.747
Bream  46.5  17.624  6.371
Pike   34.8   5.568  3.376    Pike   37.8   5.708  4.158
Pike   38.8   5.936  4.384      .  39.8      .      .
Pike   40.5   7.290  4.577    Pike   41.0   6.396  3.977
.      45.5   7.280  4.323    Pike   45.5   6.825  4.459
Pike   45.8   7.786  5.130    Pike   48.0   6.960  4.896
Pike   48.7   7.792  4.870    Pike   51.2   7.680  5.376
Pike   55.1   8.926  6.171      .  59.7  10.686  .
Pike   64.0   9.600  6.144    Pike   64.0   9.600  6.144
Pike   68.0  10.812  7.480
;

```

The following statements generate five imputed data sets to be used in this section. The default regression method is used to impute missing values in continuous variables Height and Width, and the discriminant function method is used to impute the variable Species.

```

proc mi data=Fish2 seed=1305417 out=outfish;
  class Species;
  monotone discrim( Species= Length Height Width);
  var Length Height Width Species;
run;

```

Example 57.1 through Example 57.6 use different input option combinations to combine parameter estimates computed from different procedures. Example 57.7 and Example 57.8 combine parameter estimates with classification variables. Example 57.9 shows the use of a TEST statement, and Example 57.10 combines statistics that are not directly derived from procedures.

---

### Example 57.1: Reading Means and Standard Errors from Variables in a DATA= Data Set

This example creates an ordinary SAS data set that contains sample means and standard errors computed from imputed data sets. These estimates are then combined to generate valid univariate inferences about the population means.

The following statements use the UNIVARIATE procedure to generate sample means and standard errors for the variables in each imputed data set:

```

proc univariate data=outmi noprint;
  var Oxygen RunTime RunPulse;
  output out=outuni mean=Oxygen RunTime RunPulse
               stderr=SOxygen SRunTime SRunPulse;
  by _Imputation_;
run;

```

The following statements display the output data set from PROC UNIVARIATE shown in [Output 57.1.1](#):

```
proc print data=outuni;
  title 'UNIVARIATE Means and Standard Errors';
run;
```

**Output 57.1.1** UNIVARIATE Output Data Set

| UNIVARIATE Means and Standard Errors |              |         |         |              |         |              |               |
|--------------------------------------|--------------|---------|---------|--------------|---------|--------------|---------------|
| Obs                                  | _Imputation_ | Oxygen  | RunTime | Run<br>Pulse | SOxygen | SRun<br>Time | SRun<br>Pulse |
| 1                                    | 1            | 47.0120 | 10.4441 | 171.216      | 0.95984 | 0.28520      | 1.59910       |
| 2                                    | 2            | 47.2407 | 10.5040 | 171.244      | 0.93540 | 0.26661      | 1.75638       |
| 3                                    | 3            | 47.4995 | 10.5922 | 171.909      | 1.00766 | 0.26302      | 1.85795       |
| 4                                    | 4            | 47.1485 | 10.5279 | 171.146      | 0.95439 | 0.26405      | 1.75011       |
| 5                                    | 5            | 47.0042 | 10.4913 | 172.072      | 0.96528 | 0.27275      | 1.84807       |

The following statements combine the means and standard errors from imputed data sets, The EDF= option requests that the adjusted degrees of freedom be used in the analysis. For sample means based on 31 observations, the complete-data error degrees of freedom is 30.

```
proc mianalyze data=outuni edf=30;
  modeleffects Oxygen RunTime RunPulse;
  stderr SOxygen SRunTime SRunPulse;
run;
```

The “Model Information” table in [Output 57.1.2](#) lists the input data set(s) and the number of imputations. The “Variance Information” table in [Output 57.1.2](#) displays the between-imputation variance, within-imputation variance, and total variance for each univariate inference. It also displays the degrees of freedom for the total variance. The relative increase in variance due to missing values, the fraction of missing information, and the relative efficiency for each imputed variable are also displayed. A detailed description of these statistics is provided in the section “[Combining Inferences from Imputed Data Sets](#)” on page 4650 and the section “[Multiple Imputation Efficiency](#)” on page 4652.

**Output 57.1.2** Variance Information

| The MIANALYZE Procedure |             |
|-------------------------|-------------|
| Model Information       |             |
| Data Set                | WORK.OUTUNI |
| Number of Imputations   | 5           |

**Output 57.1.2** *continued*

| Variance Information |          |          |          |        |
|----------------------|----------|----------|----------|--------|
| -----Variance-----   |          |          |          |        |
| Parameter            | Between  | Within   | Total    | DF     |
| Oxygen               | 0.041478 | 0.930853 | 0.980626 | 26.298 |
| RunTime              | 0.002948 | 0.073142 | 0.076679 | 26.503 |
| RunPulse             | 0.191086 | 3.114442 | 3.343744 | 25.463 |

  

| Variance Information |                               |                              |                     |
|----------------------|-------------------------------|------------------------------|---------------------|
| Parameter            | Relative Increase in Variance | Fraction Missing Information | Relative Efficiency |
| Oxygen               | 0.053471                      | 0.051977                     | 0.989712            |
| RunTime              | 0.048365                      | 0.047147                     | 0.990659            |
| RunPulse             | 0.073626                      | 0.070759                     | 0.986046            |

The “Parameter Estimates” table in [Output 57.1.3](#) displays the estimated mean and corresponding standard error for each variable. The table also displays a 95% confidence interval for the mean and a  $t$  statistic with the associated  $p$ -value for testing the hypothesis that the mean is equal to the value specified. You can use the THETA0= option to specify the value for the null hypothesis, which is zero by default. The table also displays the minimum and maximum parameter estimates from the imputed data sets.

**Output 57.1.3** Parameter Estimates

| Parameter Estimates |            |           |                       |          |        |
|---------------------|------------|-----------|-----------------------|----------|--------|
| Parameter           | Estimate   | Std Error | 95% Confidence Limits |          | DF     |
| Oxygen              | 47.180993  | 0.990266  | 45.1466               | 49.2154  | 26.298 |
| RunTime             | 10.511906  | 0.276910  | 9.9432                | 11.0806  | 26.503 |
| RunPulse            | 171.517500 | 1.828591  | 167.7549              | 175.2801 | 25.463 |

  

| Parameter Estimates |            |            |
|---------------------|------------|------------|
| Parameter           | Minimum    | Maximum    |
| Oxygen              | 47.004201  | 47.499541  |
| RunTime             | 10.444149  | 10.592244  |
| RunPulse            | 171.146171 | 172.071730 |

  

| Parameter Estimates |           |                  |         |
|---------------------|-----------|------------------|---------|
| Parameter           | t for H0: |                  |         |
|                     | Theta0    | Parameter=Theta0 | Pr >  t |
| Oxygen              | 0         | 47.64            | <.0001  |
| RunTime             | 0         | 37.96            | <.0001  |
| RunPulse            | 0         | 93.80            | <.0001  |

Note that the results in this example could also have been obtained with the MI procedure.

## Example 57.2: Reading Means and Covariance Matrices from a DATA= COV Data Set

This example creates a COV-type data set that contains sample means and covariance matrices computed from imputed data sets. These estimates are then combined to generate valid statistical inferences about the population means.

The following statements use the CORR procedure to generate sample means and a covariance matrix for the variables in each imputed data set:

```
proc corr data=outmi cov nocorr noprint out=outcov(type=cov);
  var Oxygen RunTime RunPulse;
  by _Imputation_;
run;
```

The following statements display (in [Output 57.2.1](#)) output sample means and covariance matrices from PROC CORR for the first two imputed data sets:

```
proc print data=outcov(obs=12);
  title 'CORR Means and Covariance Matrices'
        ' (First Two Imputations)';
run;
```

**Output 57.2.1** COV Data Set

| CORR Means and Covariance Matrices (First Two Imputations) |              |        |          |          |         |          |
|--|--------------|--------|----------|----------|---------|----------|
| Obs  | _Imputation_ | _TYPE_ | _NAME_   | Oxygen   | RunTime | RunPulse |
| 1  | 1            | COV    | Oxygen   | 28.5603  | -7.2652 | -11.812  |
| 2  | 1            | COV    | RunTime  | -7.2652  | 2.5214  | 2.536    |
| 3  | 1            | COV    | RunPulse | -11.8121 | 2.5357  | 79.271   |
| 4  | 1            | MEAN   |          | 47.0120  | 10.4441 | 171.216  |
| 5  | 1            | STD    |          | 5.3442   | 1.5879  | 8.903    |
| 6  | 1            | N      |          | 31.0000  | 31.0000 | 31.000   |
| 7  | 2            | COV    | Oxygen   | 27.1240  | -6.6761 | -10.217  |
| 8  | 2            | COV    | RunTime  | -6.6761  | 2.2035  | 2.611    |
| 9  | 2            | COV    | RunPulse | -10.2170 | 2.6114  | 95.631   |
| 10   | 2            | MEAN   |          | 47.2407  | 10.5040 | 171.244  |
| 11   | 2            | STD    |          | 5.2081   | 1.4844  | 9.779    |
| 12   | 2            | N      |          | 31.0000  | 31.0000 | 31.000   |

Note that the covariance matrices in the data set outcov are estimated covariance matrices of variables,  $V(\mathbf{y})$ . The estimated covariance matrix of the sample means is  $V(\bar{\mathbf{y}}) = V(\mathbf{y})/n$ , where  $n$  is the sample size, and is not the same as an estimated covariance matrix for variables.

The following statements combine the results for the imputed data sets, and derive both univariate and multivariate inferences about the means. The EDF= option is specified to request that the adjusted degrees

of freedom be used in the analysis. For sample means based on 31 observations, the complete-data error degrees of freedom is 30.

```
proc mianalyze data=outcov edf=30;
  modeleffects Oxygen RunTime RunPulse;
run;
```

The “Variance Information” and “Parameter Estimates” tables display the same results as in [Output 57.1.2](#) and [Output 57.1.3](#), respectively, in [Example 57.1](#).

With the WCOV, BCOV, and TCOV options, as in the following statements, the procedure displays the between-imputation covariance matrix, within-imputation covariance matrix, and total covariance matrix assuming that the between-imputation covariance matrix is proportional to the within-imputation covariance matrix in [Output 57.2.2](#).

```
proc mianalyze data=outcov edf=30 wcov bcov tcov mult;
  modeleffects Oxygen RunTime RunPulse;
run;
```

#### Output 57.2.2 Covariance Matrices

| The MIANALYZE Procedure              |              |              |              |
|--------------------------------------|--------------|--------------|--------------|
| Within-Imputation Covariance Matrix  |              |              |              |
|                                      | Oxygen       | RunTime      | RunPulse     |
| Oxygen                               | 0.930852655  | -0.226506411 | -0.461022083 |
| RunTime                              | -0.226506411 | 0.073141598  | 0.080316017  |
| RunPulse                             | -0.461022083 | 0.080316017  | 3.114441784  |
| Between-Imputation Covariance Matrix |              |              |              |
|                                      | Oxygen       | RunTime      | RunPulse     |
| Oxygen                               | 0.0414778123 | 0.0099248946 | 0.0183701754 |
| RunTime                              | 0.0099248946 | 0.0029478891 | 0.0091684769 |
| RunPulse                             | 0.0183701754 | 0.0091684769 | 0.1910855259 |
| Total Covariance Matrix              |              |              |              |
|                                      | Oxygen       | RunTime      | RunPulse     |
| Oxygen                               | 1.202882661  | -0.292700068 | -0.595750001 |
| RunTime                              | -0.292700068 | 0.094516313  | 0.103787365  |
| RunPulse                             | -0.595750001 | 0.103787365  | 4.024598310  |

With the MULT option, the procedure assumes that the between-imputation covariance matrix is proportional to the within-imputation covariance matrix and displays a multivariate inference for all the parameters taken jointly.

**Output 57.2.3** Multivariate Inference

| Multivariate Inference   |        |        |                               |        |
|--|--------|--------|-------------------------------|--------|
| Assuming Proportionality of Between/Within Covariance Matrices |        |        |                               |        |
| Avg Relative<br>Increase<br>in Variance                        | Num DF | Den DF | F for H0:<br>Parameter=Theta0 | Pr > F |
| 0.292237   | 3      | 122.68 | 12519.7                       | <.0001 |

The “Multivariate Inference” table in [Output 57.2.3](#) shows a significant  $p$ -value for the null hypothesis that the population means are all equal to zero.

---

**Example 57.3: Reading Regression Results from a DATA= EST Data Set**

This example creates an EST-type data set that contains regression coefficients and their corresponding covariance matrices computed from imputed data sets. These estimates are then combined to generate valid statistical inferences about the regression model.

The following statements use the REG procedure to generate regression coefficients:

```
proc reg data=outmi outest=outreg covout noprint;
  model Oxygen= RunTime RunPulse;
  by _Imputation_;
run;
```

The following statements display (in [Output 57.3.1](#)) output regression coefficients and their covariance matrices from PROC REG for the first two imputed data sets:

```
proc print data=outreg(obs=8);
  var _Imputation_ _Type_ _Name_
      Intercept RunTime RunPulse;
  title 'REG Model Coefficients and Covariance Matrices'
        ' (First Two Imputations)';
run;
```

**Output 57.3.1** EST-Type Data Set

| REG Model Coefficients and Covariance Matrices (First Two Imputations) |              |        |           |           |          |          |
|--|--------------|--------|-----------|-----------|----------|----------|
| Obs  | _Imputation_ | _TYPE_ | _NAME_    | Intercept | RunTime  | RunPulse |
| 1  | 1            | PARMS  |           | 86.544    | -2.82231 | -0.05873 |
| 2  | 1            | COV    | Intercept | 100.145   | -0.53519 | -0.55077 |
| 3  | 1            | COV    | RunTime   | -0.535    | 0.10774  | -0.00345 |
| 4  | 1            | COV    | RunPulse  | -0.551    | -0.00345 | 0.00343  |
| 5  | 2            | PARMS  |           | 83.021    | -3.00023 | -0.02491 |
| 6  | 2            | COV    | Intercept | 79.032    | -0.66765 | -0.41918 |
| 7  | 2            | COV    | RunTime   | -0.668    | 0.11456  | -0.00313 |
| 8  | 2            | COV    | RunPulse  | -0.419    | -0.00313 | 0.00264  |

The following statements combine the results for the imputed data sets. The EDF= option is specified to request that the adjusted degrees of freedom be used in the analysis. For a regression model with three independent variables (including the Intercept) and 31 observations, the complete-data error degrees of freedom is 28.

```
proc mianalyze data=outreg edf=28;
  modeleffects Intercept RunTime RunPulse;
run;
```

### Output 57.3.2 Variance Information

| The MIANALYZE Procedure |                                     |                                    |                        |        |
|-------------------------|-------------------------------------|------------------------------------|------------------------|--------|
| Variance Information    |                                     |                                    |                        |        |
| -----Variance-----      |                                     |                                    |                        |        |
| Parameter               | Between                             | Within                             | Total                  | DF     |
| Intercept               | 45.529229                           | 76.543614                          | 131.178689             | 9.1917 |
| RunTime                 | 0.019390                            | 0.106220                           | 0.129487               | 18.311 |
| RunPulse                | 0.001007                            | 0.002537                           | 0.003746               | 12.137 |
| Variance Information    |                                     |                                    |                        |        |
| Parameter               | Relative<br>Increase<br>in Variance | Fraction<br>Missing<br>Information | Relative<br>Efficiency |        |
| Intercept               | 0.713777                            | 0.461277                           | 0.915537               |        |
| RunTime                 | 0.219051                            | 0.192620                           | 0.962905               |        |
| RunPulse                | 0.476384                            | 0.355376                           | 0.933641               |        |

The “Variance Information” table in [Output 57.3.2](#) displays the between-imputation, within-imputation, and total variances for combining complete-data inferences.

The “Parameter Estimates” table in [Output 57.3.3](#) displays the estimated mean and standard error of the regression coefficients. The inferences are based on the  $t$  distribution. The table also displays a 95% mean confidence interval and a  $t$  test with the associated  $p$ -value for the hypothesis that the regression coefficient is equal to zero. Since the  $p$ -value for RunPulse is 0.1597, this variable can be removed from the regression model.

**Output 57.3.3** Parameter Estimates

| Parameter Estimates |           |           |                       |          |        |
|---------------------|-----------|-----------|-----------------------|----------|--------|
| Parameter           | Estimate  | Std Error | 95% Confidence Limits |          | DF     |
| Intercept           | 90.837440 | 11.453327 | 65.01034              | 116.6645 | 9.1917 |
| RunTime             | -3.032870 | 0.359844  | -3.78795              | -2.2778  | 18.311 |
| RunPulse            | -0.068578 | 0.061204  | -0.20176              | 0.0646   | 12.137 |

  

| Parameter Estimates |           |            |
|---------------------|-----------|------------|
| Parameter           | Minimum   | Maximum    |
| Intercept           | 83.020730 | 100.839807 |
| RunTime             | -3.204426 | -2.822311  |
| RunPulse            | -0.112840 | -0.024910  |

  

| Parameter Estimates |        |                  |         |
|---------------------|--------|------------------|---------|
| Parameter           | Theta0 | t for H0:        |         |
|                     |        | Parameter=Theta0 | Pr >  t |
| Intercept           | 0      | 7.93             | <.0001  |
| RunTime             | 0      | -8.43            | <.0001  |
| RunPulse            | 0      | -1.12            | 0.2842  |

**Example 57.4: Reading Mixed Model Results from PARMS= and COVB= Data Sets**

This example creates data sets that contains parameter estimates and covariance matrices computed by a mixed model analysis for a set of imputed data sets. These estimates are then combined to generate valid statistical inferences about the parameters.

The following PROC MIXED statements generate the fixed-effect parameter estimates and covariance matrix for each imputed data set:

```
proc mixed data=outmi;
  model Oxygen= RunTime RunPulse RunTime*RunPulse/solution covb;
  by _Imputation_;
  ods output SolutionF=mixparms CovB=mixcovb;
run;
```

The following statements display (in [Output 57.4.1](#)) output parameter estimates from PROC MIXED for the first two imputed data sets:

```
proc print data=mixparms (obs=8);
  var _Imputation_ Effect Estimate StdErr;
  title 'MIXED Model Coefficients (First Two Imputations)';
run;
```

**Output 57.4.1** PROC MIXED Model Coefficients

| MIXED Model Coefficients (First Two Imputations) |              |                  |          |         |
|--|--------------|------------------|----------|---------|
| Obs  | _Imputation_ | Effect           | Estimate | StdErr  |
| 1  | 1            | Intercept        | 148.09   | 81.5231 |
| 2  | 1            | RunTime          | -8.8115  | 7.8794  |
| 3  | 1            | RunPulse         | -0.4123  | 0.4684  |
| 4  | 1            | RunTime*RunPulse | 0.03437  | 0.04517 |
| 5  | 2            | Intercept        | 64.3607  | 64.6034 |
| 6  | 2            | RunTime          | -1.1270  | 6.4307  |
| 7  | 2            | RunPulse         | 0.08160  | 0.3688  |
| 8  | 2            | RunTime*RunPulse | -0.01069 | 0.03664 |

The following statements display (in [Output 57.4.2](#)) the output covariance matrices associated with the parameter estimates from PROC MIXED for the first two imputed data sets:

```
proc print data=mixcovb (obs=8);
  var _Imputation_ Row Effect Col1 Col2 Col3 Col4;
  title 'Covariance Matrices (First Two Imputations)';
run;
```

**Output 57.4.2** PROC MIXED Covariance Matrices

| Covariance Matrices (First Two Imputations) |              |                    |          |         |          |          |
|---|--------------|--------------------|----------|---------|----------|----------|
| Obs   | _Imputation_ | Row Effect         | Col1     | Col2    | Col3     | Col4     |
| 1   | 1            | 1 Intercept        | 6646.01  | -637.40 | -38.1515 | 3.6542   |
| 2   | 1            | 2 RunTime          | -637.40  | 62.0842 | 3.6548   | -0.3556  |
| 3   | 1            | 3 RunPulse         | -38.1515 | 3.6548  | 0.2194   | -0.02099 |
| 4   | 1            | 4 RunTime*RunPulse | 3.6542   | -0.3556 | -0.02099 | 0.002040 |
| 5   | 2            | 1 Intercept        | 4173.59  | -411.46 | -23.7889 | 2.3441   |
| 6   | 2            | 2 RunTime          | -411.46  | 41.3545 | 2.3414   | -0.2353  |
| 7   | 2            | 3 RunPulse         | -23.7889 | 2.3414  | 0.1360   | -0.01338 |
| 8   | 2            | 4 RunTime*RunPulse | 2.3441   | -0.2353 | -0.01338 | 0.001343 |

Note that the variables Col1, Col2, Col3, and Col4 are used to identify the effects Intercept, RunTime, RunPulse, and RunTime\*RunPulse, respectively, through the variable Row.

For univariate inference, only parameter estimates and their associated standard errors are needed. The following statements use the MIANALYZE procedure with the input PARMS= data set to produce univariate results:

```
proc mianalyze parms=mixparms edf=28;
  modeleffects Intercept RunTime RunPulse RunTime*RunPulse;
run;
```

The “Variance Information” table in [Output 57.4.3](#) displays the between-imputation, within-imputation, and total variances for combining complete-data inferences.

**Output 57.4.3** Variance Information

| The MIANALYZE Procedure |                                     |                                    |                        |        |
|-------------------------|-------------------------------------|------------------------------------|------------------------|--------|
| Variance Information    |                                     |                                    |                        |        |
| Parameter               | -----Variance-----                  |                                    |                        | DF     |
|                         | Between                             | Within                             | Total                  |        |
| Intercept               | 1972.654530                         | 4771.948777                        | 7139.134213            | 11.82  |
| RunTime                 | 14.712602                           | 45.549686                          | 63.204808              | 13.797 |
| RunPulse                | 0.062941                            | 0.156717                           | 0.232247               | 12.046 |
| RunTime*RunPulse        | 0.000470                            | 0.001490                           | 0.002055               | 13.983 |
| Variance Information    |                                     |                                    |                        |        |
| Parameter               | Relative<br>Increase<br>in Variance | Fraction<br>Missing<br>Information | Relative<br>Efficiency |        |
| Intercept               | 0.496063                            | 0.365524                           | 0.931875               |        |
| RunTime                 | 0.387601                            | 0.305893                           | 0.942348               |        |
| RunPulse                | 0.481948                            | 0.358274                           | 0.933136               |        |
| RunTime*RunPulse        | 0.378863                            | 0.300674                           | 0.943276               |        |

The “Parameter Estimates” table in [Output 57.4.4](#) displays the estimated mean and standard error of the regression coefficients.

**Output 57.4.4** Parameter Estimates

| Parameter Estimates |            |           |                       |          |        |
|---------------------|------------|-----------|-----------------------|----------|--------|
| Parameter           | Estimate   | Std Error | 95% Confidence Limits |          | DF     |
| Intercept           | 136.071356 | 84.493397 | -48.3352              | 320.4779 | 11.82  |
| RunTime             | -7.457186  | 7.950145  | -24.5322              | 9.6178   | 13.797 |
| RunPulse            | -0.328104  | 0.481920  | -1.3777               | 0.7215   | 12.046 |
| RunTime*RunPulse    | 0.025364   | 0.045328  | -0.0719               | 0.1226   | 13.983 |

  

| Parameter Estimates |            |            |
|---------------------|------------|------------|
| Parameter           | Minimum    | Maximum    |
| Intercept           | 64.360719  | 186.549814 |
| RunTime             | -11.514341 | -1.127010  |
| RunPulse            | -0.602162  | 0.081597   |
| RunTime*RunPulse    | -0.010690  | 0.047429   |

  

| Parameter Estimates |        |                  |         |
|---------------------|--------|------------------|---------|
| Parameter           | Theta0 | t for H0:        |         |
|                     |        | Parameter=Theta0 | Pr >  t |
| Intercept           | 0      | 1.61             | 0.1337  |
| RunTime             | 0      | -0.94            | 0.3644  |
| RunPulse            | 0      | -0.68            | 0.5089  |
| RunTime*RunPulse    | 0      | 0.56             | 0.5846  |

Since each covariance matrix contains variables Row, Col1, Col2, Col3, and Col4 for parameters, the EFFECTVAR=ROWCOL option is needed when you specify the COVB= option. The following statements illustrate the use of the MIANALYZE procedure with input PARMS= and COVB(EFFECTVAR=ROWCOL)= data sets:

```
proc mianalyze parms=mixparms edf=28
    covb(effectvar=rowcol)=mixcovb;
    modeleffects Intercept RunTime RunPulse RunTime*RunPulse;
run;
```

**Example 57.5: Reading Generalized Linear Model Results**

This example creates data sets that contains parameter estimates and corresponding covariance matrices computed by a generalized linear model analysis for a set of imputed data sets. These estimates are then combined to generate valid statistical inferences about the model parameters.

The following statements use PROC GENMOD to generate the parameter estimates and covariance matrix for each imputed data set:

```
proc genmod data=outmi;
    model Oxygen= RunTime RunPulse/covb;
```

```

by _Imputation_;
ods output ParameterEstimates=gmparms
           ParmInfo=gmpinfo
           CovB=gmcovb;
run;

```

The following statements print (in [Output 57.5.1](#)) the output parameter estimates and covariance matrix from PROC GENMOD for the first two imputed data sets:

```

proc print data=gmparms (obs=8);
  var _Imputation_ Parameter Estimate StdErr;
  title 'GENMOD Model Coefficients (First Two Imputations)';
run;

```

**Output 57.5.1** PROC GENMOD Model Coefficients

| GENMOD Model Coefficients (First Two Imputations) |              |           |          |        |
|---|--------------|-----------|----------|--------|
| Obs   | _Imputation_ | Parameter | Estimate | StdErr |
| 1   | 1            | Intercept | 86.5440  | 9.5107 |
| 2   | 1            | RunTime   | -2.8223  | 0.3120 |
| 3   | 1            | RunPulse  | -0.0587  | 0.0556 |
| 4   | 1            | Scale     | 2.6692   | 0.3390 |
| 5   | 2            | Intercept | 83.0207  | 8.4489 |
| 6   | 2            | RunTime   | -3.0002  | 0.3217 |
| 7   | 2            | RunPulse  | -0.0249  | 0.0488 |
| 8   | 2            | Scale     | 2.5727   | 0.3267 |

The following statements display the parameter information table in [Output 57.5.2](#). The table identifies parameter names used in the covariance matrices. The parameters Prm1, Prm2, and Prm3 are used for the effects Intercept, RunTime, and RunPulse, respectively, in each covariance matrix.

```

proc print data=gmpinfo (obs=6);
  title 'GENMOD Parameter Information (First Two Imputations)';
run;

```

**Output 57.5.2** PROC GENMOD Model Information

| GENMOD Parameter Information (First Two Imputations) |              |           |           |
|--|--------------|-----------|-----------|
| Obs  | _Imputation_ | Parameter | Effect    |
| 1  | 1            | Prm1      | Intercept |
| 2  | 1            | Prm2      | RunTime   |
| 3  | 1            | Prm3      | RunPulse  |
| 4  | 2            | Prm1      | Intercept |
| 5  | 2            | Prm2      | RunTime   |
| 6  | 2            | Prm3      | RunPulse  |

The following statements display (in [Output 57.5.3](#)) the output covariance matrices from PROC GENMOD for the first two imputed data sets. Note that the GENMOD procedure computes maximum likelihood estimates for each covariance matrix.

```
proc print data=gmcovb (obs=8);
  var _Imputation_ RowName Prm1 Prm2 Prm3;
  title 'GENMOD Covariance Matrices (First Two Imputations)';
run;
```

### Output 57.5.3 PROC GENMOD Covariance Matrices

| GENMOD Covariance Matrices (First Two Imputations) |              |             |           |           |           |
|--|--------------|-------------|-----------|-----------|-----------|
| Obs  | _Imputation_ | Row<br>Name | Prm1      | Prm2      | Prm3      |
| 1  | 1            | Prm1        | 90.453923 | -0.483394 | -0.497473 |
| 2  | 1            | Prm2        | -0.483394 | 0.0973159 | -0.003113 |
| 3  | 1            | Prm3        | -0.497473 | -0.003113 | 0.0030954 |
| 4  | 1            | Scale       | 1.344E-15 | -1.09E-17 | -6.12E-18 |
| 5  | 2            | Prm1        | 71.383332 | -0.603037 | -0.378616 |
| 6  | 2            | Prm2        | -0.603037 | 0.1034766 | -0.002826 |
| 7  | 2            | Prm3        | -0.378616 | -0.002826 | 0.0023843 |
| 8  | 2            | Scale       | 1.602E-14 | 1.755E-16 | -1.02E-16 |

The following statements use the MIANALYZE procedure with input PARMS=, PARMINFO=, and COVB= data sets:

```
proc mianalyze parms=gmparms covb=gmcovb parminfo=gmpinfo;
  modeleffects Intercept RunTime RunPulse;
run;
```

Since the GENMOD procedure computes maximum likelihood estimates for the covariance matrix, the EDF= option is not used. The resulting model coefficients are identical to the estimates in [Output 57.3.3](#) in [Example 57.3](#). However, the standard errors are slightly different because in this example, maximum likelihood estimates for the standard errors are combined without the EDF= option, whereas in [Example 57.3](#), unbiased estimates for the standard errors are combined with the EDF= option.

## Example 57.6: Reading GLM Results from PARMS= and XPXI= Data Sets

This example creates data sets that contains parameter estimates and corresponding  $(X'X)^{-1}$  matrices computed by a general linear model analysis for a set of imputed data sets. These estimates are then combined to generate valid statistical inferences about the model parameters.

The following statements use PROC GLM to generate the parameter estimates and  $(X'X)^{-1}$  matrix for each imputed data set:

```
proc glm data=outmi;
  model Oxygen= RunTime RunPulse/inverse;
  by _Imputation_;
  ods output ParameterEstimates=glmparms
             InvXPX=glmxpxi;
quit;
```

The following statements display (in [Output 57.6.1](#)) the output parameter estimates and standard errors from PROC GLM for the first two imputed data sets:

```
proc print data=glmparms (obs=6);
  var _Imputation_ Parameter Estimate StdErr;
  title 'GLM Model Coefficients (First Two Imputations)';
run;
```

**Output 57.6.1** PROC GLM Model Coefficients

| GLM Model Coefficients (First Two Imputations) |              |           |            |             |  |
|--|--------------|-----------|------------|-------------|--|
| Obs  | _Imputation_ | Parameter | Estimate   | StdErr      |  |
| 1  | 1            | Intercept | 86.5440339 | 10.00726811 |  |
| 2  | 1            | RunTime   | -2.8223108 | 0.32824165  |  |
| 3  | 1            | RunPulse  | -0.0587292 | 0.05854109  |  |
| 4  | 2            | Intercept | 83.0207303 | 8.88996885  |  |
| 5  | 2            | RunTime   | -3.0002288 | 0.33847204  |  |
| 6  | 2            | RunPulse  | -0.0249103 | 0.05137859  |  |

The following statements display (in [Output 57.6.2](#))  $(X'X)^{-1}$  matrices from PROC GLM for the first two imputed data sets:

```
proc print data=glmxpxi (obs=8);
  var _Imputation_ Parameter Intercept RunTime RunPulse;
  title 'GLM X'X Inverse Matrices (First Two Imputations)';
run;
```

**Output 57.6.2** PROC GLM  $(X'X)^{-1}$  Matrices

| GLM X'X Inverse Matrices (First Two Imputations) |              |           |              |              |              |
|--|--------------|-----------|--------------|--------------|--------------|
| Obs  | _Imputation_ | Parameter | Intercept    | RunTime      | RunPulse     |
| 1  | 1            | Intercept | 12.696250656 | -0.067849956 | -0.069826009 |
| 2  | 1            | RunTime   | -0.067849956 | 0.0136594055 | -0.000436938 |
| 3  | 1            | RunPulse  | -0.069826009 | -0.000436938 | 0.0004344762 |
| 4  | 1            | Oxygen    | 86.544033929 | -2.822310769 | -0.058729234 |
| 5  | 2            | Intercept | 10.784620785 | -0.091107072 | -0.057201387 |
| 6  | 2            | RunTime   | -0.091107072 | 0.0156332765 | -0.000426902 |
| 7  | 2            | RunPulse  | -0.057201387 | -0.000426902 | 0.0003602208 |
| 8  | 2            | Oxygen    | 83.020730343 | -3.000228818 | -0.024910305 |

The standard errors for the estimates in the output glmparms data set are needed to create the covariance matrix from the  $(X'X)^{-1}$  matrix. The following statements use the MIANALYZE procedure with input PARMS= and XPXI= data sets to produce the same results as displayed in [Output 57.3.2](#) and [Output 57.3.3](#) in [Example 57.3](#):

```
proc mianalyze parms=glmparms xpxi=glmxpxi edf=28;
  modeleffects Intercept RunTime RunPulse;
run;
```

## Example 57.7: Reading Logistic Model Results from PARMS= and COVB= Data Sets

This example creates data sets that contains parameter estimates and corresponding covariance matrices computed by a logistic regression analysis for a set of imputed data sets. These estimates are then combined to generate valid statistical inferences about the model parameters.

The following statements use PROC LOGISTIC to generate the parameter estimates and covariance matrix for each imputed data set:

```
proc logistic data=outfish;
  class Species;
  model Species= Height Width Height*Width/ covb;
  by _Imputation_;
  ods output ParameterEstimates=lgsparms
             CovB=lgscovb;
run;
```

The following statements display (in [Output 57.7.1](#)) the output logistic regression coefficients from PROC LOGISTIC for the first two imputed data sets:

```
proc print data=lgsparms (obs=8);
  title 'LOGISTIC Model Coefficients (First Two Imputations)';
run;
```

**Output 57.7.1** PROC LOGISTIC Model Coefficients

| LOGISTIC Model Coefficients (First Two Imputations) |              |              |    |          |         |           |               |
|---|--------------|--------------|----|----------|---------|-----------|---------------|
| Obs   | _Imputation_ | Variable     | DF | Estimate | StdErr  | WaldChiSq | Prob<br>ChiSq |
| 1   | 1            | Intercept    | 1  | -28.2353 | 316.1   | 0.0080    | 0.9288        |
| 2   | 1            | Height       | 1  | 5.3362   | 28.1298 | 0.0360    | 0.8495        |
| 3   | 1            | Width        | 1  | -1.0812  | 60.8035 | 0.0003    | 0.9858        |
| 4   | 1            | Height*Width | 1  | -0.4304  | 5.1312  | 0.0070    | 0.9332        |
| 5   | 2            | Intercept    | 1  | -44.0620 | 262.5   | 0.0282    | 0.8667        |
| 6   | 2            | Height       | 1  | 7.3887   | 23.1824 | 0.1016    | 0.7499        |
| 7   | 2            | Width        | 1  | 1.6950   | 49.1462 | 0.0012    | 0.9725        |
| 8   | 2            | Height*Width | 1  | -0.7692  | 4.0205  | 0.0366    | 0.8483        |

The following statements displays the covariance matrices associated with parameter estimates derived from the first two imputations in [Output 57.7.2](#):

```
proc print data=lgscovb (obs=8);
  title 'LOGISTIC Model Covariance Matrices (First Two Imputations)';
run;
```

**Output 57.7.2** PROC LOGISTIC Covariance Matrices

| LOGISTIC Model Covariance Matrices (First Two Imputations) |              |             |           |          |          |                 |
|--|--------------|-------------|-----------|----------|----------|-----------------|
| Obs  | _Imputation_ | Parameter   | Intercept | Height   | Width    | Height<br>Width |
| 1  | 1            | Intercept   | 99938.75  | -8395.34 | -18879.9 | 1556.383        |
| 2  | 1            | Height      | -8395.34  | 791.2859 | 1535.382 | -142.121        |
| 3  | 1            | Width       | -18879.9  | 1535.382 | 3697.064 | -294.815        |
| 4  | 1            | HeightWidth | 1556.383  | -142.121 | -294.815 | 26.32931        |
| 5  | 2            | Intercept   | 68903.42  | -5586.74 | -12603.5 | 1000.283        |
| 6  | 2            | Height      | -5586.74  | 537.4232 | 958.5588 | -91.2266        |
| 7  | 2            | Width       | -12603.5  | 958.5588 | 2415.346 | -180.394        |
| 8  | 2            | HeightWidth | 1000.283  | -91.2266 | -180.394 | 16.16428        |

The following statements use the MIANALYZE procedure with input PARMS= and COVB= data sets:

```
proc mianalyze parms=lgsparms
               covb(effectvar=stacking)=lgscovb;
  modeleffects Intercept Height Width Height*Width;
run;
```

The “Variance Information” table in [Output 57.7.3](#) displays the between-imputation, within-imputation, and total variances for combining complete-data inferences.

**Output 57.7.3** Variance Information

| The MIANALYZE Procedure |                                     |                                    |                        |        |
|-------------------------|-------------------------------------|------------------------------------|------------------------|--------|
| Variance Information    |                                     |                                    |                        |        |
| -----Variance-----      |                                     |                                    |                        |        |
| Parameter               | Between                             | Within                             | Total                  | DF     |
| Intercept               | 283.306802                          | 93045                              | 93385                  | 301811 |
| Height                  | 4.985634                            | 751.535758                         | 757.518519             | 64127  |
| Width                   | 6.262249                            | 3331.888954                        | 3339.403653            | 789905 |
| Height*Width            | 0.113341                            | 23.797208                          | 23.933217              | 123858 |
| Variance Information    |                                     |                                    |                        |        |
| Parameter               | Relative<br>Increase<br>in Variance | Fraction<br>Missing<br>Information | Relative<br>Efficiency |        |
| Intercept               | 0.003654                            | 0.003647                           | 0.999271               |        |
| Height                  | 0.007961                            | 0.007929                           | 0.998417               |        |
| Width                   | 0.002255                            | 0.002253                           | 0.999550               |        |
| Height*Width            | 0.005715                            | 0.005699                           | 0.998862               |        |

The “Parameter Estimates” table in [Output 57.7.4](#) displays the combined parameter estimates with associated standard errors.

**Output 57.7.4** Parameter Estimates

| Parameter Estimates |            |            |                       |          |        |
|---------------------|------------|------------|-----------------------|----------|--------|
| Parameter           | Estimate   | Std Error  | 95% Confidence Limits |          | DF     |
| Intercept           | -45.536682 | 305.589037 | -644.483              | 553.4092 | 301811 |
| Height              | 7.452449   | 27.523054  | -46.493               | 61.3977  | 64127  |
| Width               | 1.548439   | 57.787574  | -111.713              | 114.8102 | 789905 |
| Height*Width        | -0.754088  | 4.892159   | -10.343               | 8.8345   | 123858 |

  

| Parameter Estimates |            |            |  |
|---------------------|------------|------------|--|
| Parameter           | Minimum    | Maximum    |  |
| Intercept           | -73.331892 | -28.235273 |  |
| Height              | 5.336231   | 11.217552  |  |
| Width               | -1.081173  | 5.645810   |  |
| Height*Width        | -1.313883  | -0.430377  |  |

  

| Parameter Estimates |        |                  |         |  |
|---------------------|--------|------------------|---------|--|
| Parameter           | Theta0 | t for H0:        |         |  |
|                     |        | Parameter=Theta0 | Pr >  t |  |
| Intercept           | 0      | -0.15            | 0.8815  |  |
| Height              | 0      | 0.27             | 0.7866  |  |
| Width               | 0      | 0.03             | 0.9786  |  |
| Height*Width        | 0      | -0.15            | 0.8775  |  |

## Example 57.8: Reading Mixed Model Results with Classification Variables

This example creates data sets that contains parameter estimates and corresponding covariance matrices with classification variables computed by a mixed regression model analysis for a set of imputed data sets. These estimates are then combined to generate valid statistical inferences about the model parameters.

The following statements use PROC MIXED to generate the parameter estimates and covariance matrix for each imputed data set:

```
proc mixed data=outfish;
  class Species;
  model Length= Species Height Width/ solution covb;
  by _Imputation_;
  ods output SolutionF=mxparms CovB=mxcovb;
run;
```

The following statements display (in [Output 57.8.1](#)) the output mixed model coefficients from PROC MIXED for the first two imputed data sets:

```
proc print data=mxparms (obs=10);
  var _Imputation_ Effect Species Estimate StdErr;
  title 'MIXED Model Coefficients (First Two Imputations)';
run;
```

**Output 57.8.1** PROC MIXED Model Coefficients

| MIXED Model Coefficients (First Two Imputations) |              |           |         |          |        |
|--|--------------|-----------|---------|----------|--------|
| Obs  | _Imputation_ | Effect    | Species | Estimate | StdErr |
| 1  | 1            | Intercept |         | 12.5356  | 2.7808 |
| 2  | 1            | Species   | Bream   | -11.9103 | 3.5386 |
| 3  | 1            | Species   | Pike    | 0        | .      |
| 4  | 1            | Height    |         | -0.1605  | 0.5158 |
| 5  | 1            | Width     |         | 7.3962   | 1.1365 |
| 6  | 2            | Intercept |         | 13.3607  | 2.7848 |
| 7  | 2            | Species   | Bream   | -10.5204 | 3.0517 |
| 8  | 2            | Species   | Pike    | 0        | .      |
| 9  | 2            | Height    |         | -0.3139  | 0.4384 |
| 10   | 2            | Width     |         | 7.4861   | 1.0005 |

The following statements use the MIANALYZE procedure with an input PARMS= data set:

```
proc mianalyze parms(classvar=full)=mxparms;
  class Species;
  modeleffects Intercept Species Height Width;
run;
```

The “Variance Information” table in [Output 57.8.2](#) displays the between-imputation, within-imputation, and total variances for combining complete-data inferences.

**Output 57.8.2** Variance Information

| The MIANALYZE Procedure |         |                                     |                                    |                        |        |
|-------------------------|---------|-------------------------------------|------------------------------------|------------------------|--------|
| Variance Information    |         |                                     |                                    |                        |        |
| Parameter               | Species | -----Variance-----                  |                                    |                        |        |
|                         |         | Between                             | Within                             | Total                  | DF     |
| Intercept               |         | 0.325023                            | 7.632716                           | 8.022743               | 1692.4 |
| Species                 | Bream   | 0.307202                            | 10.394843                          | 10.763486              | 3410   |
| Species                 | Pike    | 0                                   | .                                  | .                      | .      |
| Height                  |         | 0.003686                            | 0.217662                           | 0.222085               | 10085  |
| Width                   |         | 0.006488                            | 1.097103                           | 1.104888               | 80560  |
| Variance Information    |         |                                     |                                    |                        |        |
| Parameter               | Species | Relative<br>Increase<br>in Variance | Fraction<br>Missing<br>Information | Relative<br>Efficiency |        |
| Intercept               |         | 0.051099                            | 0.049738                           | 0.990150               |        |
| Species                 | Bream   | 0.035464                            | 0.034815                           | 0.993085               |        |
| Species                 | Pike    | .                                   | .                                  | .                      |        |
| Height                  |         | 0.020320                            | 0.020110                           | 0.995994               |        |
| Width                   |         | 0.007096                            | 0.007071                           | 0.998588               |        |

The “Parameter Estimates” table in [Output 57.8.3](#) displays the combined parameter estimates with associated standard errors.

**Output 57.8.3** Parameter Estimates

| Parameter Estimates |         |            |           |                       |          |        |
|---------------------|---------|------------|-----------|-----------------------|----------|--------|
| Parameter           | Species | Estimate   | Std Error | 95% Confidence Limits |          | DF     |
| Intercept           |         | 12.669835  | 2.832445  | 7.1144                | 18.22530 | 1692.4 |
| Species             | Bream   | -11.180159 | 3.280775  | -17.6126              | -4.74767 | 3410   |
| Species             | Pike    | 0          | .         | .                     | .        | .      |
| Height              |         | -0.246488  | 0.471259  | -1.1702               | 0.67727  | 10085  |
| Width               |         | 7.511074   | 1.051137  | 5.4509                | 9.57130  | 80560  |

| Parameter Estimates |           |         |            |            |
|---------------------|-----------|---------|------------|------------|
|                     | Parameter | Species | Minimum    | Maximum    |
|                     | Intercept |         | 12.004593  | 13.360690  |
|                     | Species   | Bream   | -11.910303 | -10.520395 |
|                     | Species   | Pike    | 0          | 0          |
|                     | Height    |         | -0.313882  | -0.160511  |
|                     | Width     |         | 7.396172   | 7.594860   |

| Parameter Estimates |         |        |                  |         |
|---------------------|---------|--------|------------------|---------|
| Parameter           | Species | Theta0 | t for H0:        |         |
|                     |         |        | Parameter=Theta0 | Pr >  t |
| Intercept           |         | 0      | 4.47             | <.0001  |
| Species             | Bream   | 0      | -3.41            | 0.0007  |
| Species             | Pike    | 0      | .                | .       |
| Height              |         | 0      | -0.52            | 0.6010  |
| Width               |         | 0      | 7.15             | <.0001  |

**Example 57.9: Using a TEST statement**

This example creates an EST-type data set that contains regression coefficients and their corresponding covariance matrices computed from imputed data sets. These estimates are then combined to generate valid statistical inferences about the regression model. A TEST statement is used to test linear hypotheses about the parameters.

The following statements use the REG procedure to generate regression coefficients:

```
proc reg data=outmi outest=outreg covout noprint;
  model Oxygen= RunTime RunPulse;
  by _Imputation_;
run;
```

The following statements combine the results for the imputed data sets. A TEST statement is used to test linear hypotheses of Intercept=0 and RunTime=RunPulse.

```
proc mianalyze data=outreg edf=28;
  modeleffects Intercept RunTime RunPulse;
  test Intercept, RunTime=RunPulse / mult;
run;
```

The “Test Specification” table in [Output 57.9.1](#) displays the **L** matrix and the **c** vector in a TEST statement. Since there is no label specified for the TEST statement, “Test 1” is used as the label.

**Output 57.9.1** Test Specification

| The MIANALYZE Procedure |           |          |           |   |
|-------------------------|-----------|----------|-----------|---|
| Test: Test 1            |           |          |           |   |
| Test Specification      |           |          |           |   |
| -----L Matrix-----      |           |          |           |   |
| Parameter               | Intercept | RunTime  | RunPulse  | C |
| TestPrm1                | 1.000000  | 0        | 0         | 0 |
| TestPrm2                | 0         | 1.000000 | -1.000000 | 0 |

The “Variance Information” table in [Output 57.9.2](#) displays the between-imputation variance, within-imputation variance, and total variance for each univariate inference. A detailed description of these statistics is provided in the section “[Combining Inferences from Imputed Data Sets](#)” on page 4650 and the section “[Multiple Imputation Efficiency](#)” on page 4652.

**Output 57.9.2** Variance Information

| Variance Information |                               |                              |                     |        |
|----------------------|-------------------------------|------------------------------|---------------------|--------|
| -----Variance-----   |                               |                              |                     |        |
| Parameter            | Between                       | Within                       | Total               | DF     |
| TestPrm1             | 45.529229                     | 76.543614                    | 131.178689          | 9.1917 |
| TestPrm2             | 0.014715                      | 0.114324                     | 0.131983            | 20.598 |
| Variance Information |                               |                              |                     |        |
| Parameter            | Relative Increase in Variance | Fraction Missing Information | Relative Efficiency |        |
| TestPrm1             | 0.713777                      | 0.461277                     | 0.915537            |        |
| TestPrm2             | 0.154459                      | 0.141444                     | 0.972490            |        |

The “Parameter Estimates” table in [Output 57.9.3](#) displays the estimated mean and standard error of the linear components. The inferences are based on the  $t$  distribution. The table also displays a 95% mean confidence interval and a  $t$  test with the associated  $p$ -value for the hypothesis that each linear component of  $L\beta$  is equal to zero.

**Output 57.9.3** Parameter Estimates

| Parameter Estimates |           |            |                       |                          |         |
|---------------------|-----------|------------|-----------------------|--------------------------|---------|
| Parameter           | Estimate  | Std Error  | 95% Confidence Limits |                          | DF      |
| TestPrm1            | 90.837440 | 11.453327  | 65.01034              | 116.6645                 | 9.1917  |
| TestPrm2            | -2.964292 | 0.363294   | -3.72070              | -2.2079                  | 20.598  |
| Parameter Estimates |           |            |                       |                          |         |
| Parameter           | Minimum   | Maximum    | C                     | t for H0:<br>Parameter=C | Pr >  t |
| TestPrm1            | 83.020730 | 100.839807 | 0                     | 7.93                     | <.0001  |
| TestPrm2            | -3.091586 | -2.763582  | 0                     | -8.16                    | <.0001  |

With the MULT option, the procedure assumes that the between-imputation covariance matrix is proportional to the within-imputation covariance matrix and displays a multivariate inference for all the linear components taken jointly in [Output 57.9.4](#).

**Output 57.9.4** Multivariate Inference

| Multivariate Inference   |        |        |                               |        |  |
|--|--------|--------|-------------------------------|--------|--|
| Assuming Proportionality of Between/Within Covariance Matrices |        |        |                               |        |  |
| Avg Relative<br>Increase<br>in Variance                        | Num DF | Den DF | F for H0:<br>Parameter=Theta0 | Pr > F |  |
| 0.419868   | 2      | 35.053 | 60.34                         | <.0001 |  |

## Example 57.10: Combining Correlation Coefficients

This example combines sample correlation coefficients computed from a set of imputed data sets by using Fisher’s  $z$  transformation.

Fisher’s  $z$  transformation of the sample correlation  $r$  is

$$z = \frac{1}{2} \log \left( \frac{1+r}{1-r} \right)$$

The statistic  $z$  is approximately normally distributed with mean

$$\log \left( \frac{1 + \rho}{1 - \rho} \right)$$

and variance  $1/(n-3)$ , where  $\rho$  is the population correlation coefficient and  $n$  is the number of observations.

The following statements use the CORR procedure to compute the correlation  $r$  and its associated Fisher's  $z$  statistic between variables Oxygen and RunTime for each imputed data set. The ODS statement is used to save Fisher's  $z$  statistic in an output data set.

```
proc corr data=outmi fisher(biasadj=no);
  var Oxygen RunTime;
  by _Imputation_;
  ods output FisherPearsonCorr= outz;
run;
```

The following statements display the number of observations and Fisher's  $z$  statistic for each imputed data set in [Output 57.10.1](#):

```
proc print data=outz;
  title 'Fisher's Correlation Statistics';
  var _Imputation_ NObs ZVal;
run;
```

**Output 57.10.1** Output  $z$  Statistics

| Fisher's Correlation Statistics |     |              |      |          |
|---------------------------------|-----|--------------|------|----------|
|                                 | Obs | _Imputation_ | NObs | ZVal     |
|                                 | 1   | 1            | 31   | -1.27869 |
|                                 | 2   | 2            | 31   | -1.30715 |
|                                 | 3   | 3            | 31   | -1.27922 |
|                                 | 4   | 4            | 31   | -1.39243 |
|                                 | 5   | 5            | 31   | -1.40146 |

The following statements generate the standard error associated with the  $z$  statistic,  $1/\sqrt{n-3}$ :

```
data outz;
  set outz;
  StdZ= 1. / sqrt(NObs-3);
run;
```

The following statements use the MIANALYZE procedure to generate a combined parameter estimate  $\hat{z}$  and its variance, as shown in [Output 57.10.2](#). The ODS statement is used to save the parameter estimates in an output data set.

```
proc mianalyze data=outz;
  ods output ParameterEstimates=parms;
  modeleffects ZVal;
  stderr StdZ;
run;
```

**Output 57.10.2** Combining Fisher's  $z$  Statistics

| The MIANALYZE Procedure |           |                  |                       |          |        |
|-------------------------|-----------|------------------|-----------------------|----------|--------|
| Parameter Estimates     |           |                  |                       |          |        |
| Parameter               | Estimate  | Std Error        | 95% Confidence Limits |          | DF     |
| ZVal                    | -1.331787 | 0.200327         | -1.72587              | -0.93771 | 330.23 |
| Parameter Estimates     |           |                  |                       |          |        |
| Parameter               | Minimum   | Maximum          |                       |          |        |
| ZVal                    | -1.401459 | -1.278686        |                       |          |        |
| Parameter Estimates     |           |                  |                       |          |        |
| Parameter               | Theta0    | t for H0:        |                       |          |        |
|                         |           | Parameter=Theta0 | Pr >  t               |          |        |
| ZVal                    | 0         | -6.65            | <.0001                |          |        |

In addition to the estimate for  $z$ , PROC MIANALYZE also generates 95% confidence limits for  $z$ ,  $\hat{z}_{.025}$  and  $\hat{z}_{.975}$ . The following statements print the estimate and 95% confidence limits for  $z$  in [Output 57.10.3](#):

```
proc print data=parms;
  title 'Parameter Estimates with 95% Confidence Limits';
  var Estimate LCLMean UCLMean;
run;
```

**Output 57.10.3** Parameter Estimates with 95% Confidence Limits

| Parameter Estimates with 95% Confidence Limits |           |          |          |  |
|--|-----------|----------|----------|--|
| Obs  | Estimate  | LCLMean  | UCLMean  |  |
| 1  | -1.331787 | -1.72587 | -0.93771 |  |

An estimate of the correlation coefficient with its corresponding 95% confidence limits is then generated from the following inverse transformation as described in the section “[Correlation Coefficients](#)” on page 4655:

$$r = \tanh(z) = \frac{e^{2z} - 1}{e^{2z} + 1}$$

for  $z = \hat{z}$ ,  $\hat{z}_{.025}$ , and  $\hat{z}_{.975}$ .

The following statements generate and display an estimate of the correlation coefficient and its 95% confidence limits, as shown in [Output 57.10.4](#):

```
data corr_ci;
  set parms;
  r=      tanh( Estimate);
  r_lower= tanh( LCLMean);
  r_upper= tanh( UCLMean);
run;
proc print data=corr_ci;
  title 'Estimated Correlation Coefficient'
        ' with 95% Confidence Limits';
  var r r_lower r_upper;
run;
```

**Output 57.10.4** Estimated Correlation Coefficient

| Estimated Correlation Coefficient with 95% Confidence Limits |     |          |          |          |
|--|-----|----------|----------|----------|
|  | Obs | r        | r_lower  | r_upper  |
|  | 1   | -0.86969 | -0.93857 | -0.73417 |

---

## References

- Allison, P. D. (2000), "Multiple Imputation for Missing Data: A Cautionary Tale," *Sociological Methods and Research*, 28, 301–309.
- Allison, P. D. (2001), "Missing Data," Thousand Oaks, CA: Sage Publications.
- Barnard, J. and Rubin, D. B. (1999), "Small-Sample Degrees of Freedom with Multiple Imputation," *Biometrika*, 86, 948–955.
- Cochran, W. J. (1977), *Sampling Techniques*, Second Edition, New York: John Wiley & Sons.
- Gadbury, G. L., Coffey, C. S., and Allison, D. B. (2003), "Modern Statistical Methods for Handling Missing Repeated Measurements in Obesity Trial Data: Beyond LOCF," *Obesity Reviews*, 4, 175–184.
- Horton, N. J. and Lipsitz, S. R. (2001), "Multiple Imputation in Practice: Comparison of Software Packages for Regression Models with Missing Variables," *Journal of the American Statistical Association*, 55, 244–254.
- Li, K. H., Raghunathan, T. E., and Rubin, D. B. (1991), "Large-Sample Significance Levels from Multiply Imputed Data Using Moment-Based Statistics and an F Reference Distribution," *Journal of the American Statistical Association*, 86, 1065–1073.
- Little, R. J. A. and Rubin, D. B. (2002), *Statistical Analysis with Missing Data*, Second Edition, New York: John Wiley & Sons.

Rubin, D. B. (1976), "Inference and Missing Data," *Biometrika*, 63, 581–592.

Rubin, D. B. (1987), *Multiple Imputation for Nonresponse in Surveys*, New York: John Wiley & Sons.

Rubin, D. B. (1996), "Multiple Imputation after 18+ Years," *Journal of the American Statistical Association*, 91, 473–489.

Schafer, J. L. (1997), *Analysis of Incomplete Multivariate Data*, New York: Chapman and Hall.

# Subject Index

## A

adjusted degrees of freedom  
    MIANALYZE procedure, [4652](#)  
average relative increase in variance  
    MIANALYZE procedure, [4653](#)

## B

between-imputation covariance matrix  
    MIANALYZE procedure, [4653](#)  
between-imputation variance  
    MIANALYZE procedure, [4651](#)

## C

combining inferences  
    MIANALYZE procedure, [4650](#)

## D

degrees of freedom  
    MIANALYZE procedure, [4651](#), [4653](#)

## F

fraction of missing information  
    MIANALYZE procedure, [4651](#)

## I

input data sets  
    MIANALYZE procedure, [4646](#)

## M

MIANALYZE procedure  
    adjusted degrees of freedom, [4652](#)  
    average relative increase in variance, [4653](#)  
    between-imputation covariance matrix, [4653](#)  
    between-imputation variance, [4651](#)  
    combining inferences, [4650](#)  
    degrees of freedom, [4651](#), [4653](#)  
    fraction of missing information, [4651](#)  
    input data sets, [4646](#)  
    introductory example, [4637](#)  
    multiple imputation efficiency, [4652](#)  
    multivariate inferences, [4652](#)

    ODS table names, [4656](#)  
    relative efficiency, [4652](#)  
    relative increase in variance, [4651](#)  
    syntax, [4640](#)  
    testing linear hypotheses, [4644](#), [4654](#)  
    total covariance matrix, [4653](#)  
    total variance, [4651](#)  
    within-imputation covariance matrix, [4653](#)  
    within-imputation variance, [4651](#)  
multiple imputation efficiency  
    MIANALYZE procedure, [4652](#)  
multiple imputations analysis, [4636](#)  
multivariate inferences  
    MIANALYZE procedure, [4652](#)

## R

relative efficiency  
    MIANALYZE procedure, [4652](#)  
relative increase in variance  
    MIANALYZE procedure, [4651](#)

## T

testing linear hypotheses  
    MIANALYZE procedure, [4644](#), [4654](#)  
total covariance matrix  
    MIANALYZE procedure, [4653](#)  
total variance  
    MIANALYZE procedure, [4651](#)

## W

within-imputation covariance matrix  
    MIANALYZE procedure, [4653](#)  
within-imputation variance  
    MIANALYZE procedure, [4651](#)



# Syntax Index

## A

ALPHA= option  
PROC MIANALYZE statement, [4641](#)

## B

BCOV option  
PROC MIANALYZE statement, [4641](#)  
TEST statement (MIANALYZE), [4646](#)  
BY statement  
MIANALYZE procedure, [4643](#)

## C

CLASS statement  
MIANALYZE procedure, [4643](#)  
CLASSVAR= option  
PROC MIANALYZE statement, [4642](#)  
COVB= option  
PROC MIANALYZE statement, [4641](#)

## D

DATA= option  
PROC MIANALYZE statement, [4641](#)

## E

EDF= option  
PROC MIANALYZE statement, [4642](#)  
EFFECTVAR= option  
PROC MIANALYZE statement, [4641](#)

## M

MIANALYZE procedure, BY statement, [4643](#)  
MIANALYZE procedure, CLASS statement, [4643](#)  
MIANALYZE procedure, MODELEFFECTS  
statement, [4644](#)  
MIANALYZE procedure, PROC MIANALYZE  
statement, [4640](#)  
ALPHA= option, [4641](#)  
BCOV option, [4641](#)  
CLASSVAR= option, [4642](#)  
COVB= option, [4641](#)  
DATA= option, [4641](#)  
EDF= option, [4642](#)

EFFECTVAR= option, [4641](#)  
MU0= option, [4642](#)  
MULT option, [4642](#)  
PARMINFO= option, [4642](#)  
PARMS= option, [4642](#)  
TCOV option, [4642](#)  
THETA0= option, [4642](#)  
WCOV option, [4642](#)  
XPXI= option, [4643](#)

MIANALYZE procedure, STDERR statement, [4644](#)  
MIANALYZE procedure, TEST statement, [4644](#)

BCOV option, [4646](#)  
MULT option, [4646](#)  
TCOV option, [4646](#)  
WCOV option, [4646](#)

MODELEFFECTS statement  
MIANALYZE procedure, [4644](#)

MU0= option  
PROC MIANALYZE statement, [4642](#)  
MULT option  
PROC MIANALYZE statement, [4642](#)  
TEST statement (MIANALYZE), [4646](#)

## P

PARMINFO= option  
PROC MIANALYZE statement, [4642](#)  
PARMS= option  
PROC MIANALYZE statement, [4642](#)  
PROC MIANALYZE statement, *see* MIANALYZE  
procedure

## S

STDERR statement  
MIANALYZE procedure, [4644](#)

## T

TCOV option  
PROC MIANALYZE statement, [4642](#)  
TEST statement (MIANALYZE), [4646](#)  
TEST statement  
MIANALYZE procedure, [4644](#)  
THETA0= option  
PROC MIANALYZE statement, [4642](#)

## W

WCOV option

PROC MIANALYZE statement, [4642](#)

TEST statement (MIANALYZE), [4646](#)

## X

XPXI= option

PROC MIANALYZE statement, [4643](#)

## 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](http://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](http://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](http://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](http://support.sas.com/spn)**



**THE  
POWER  
TO KNOW®**

