Chapter 10
The MIANALYZE Procedure

Chapter Table of Contents

Chapter 10
The MIANALYZE Procedure

Overview

The experimental 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. You can implement the strategy with two SAS procedures: PROC MI, which generates imputed data sets, and PROC MIANALYZE, which combines the results of analyses carried out on the data sets. These two procedures are available in experimental form in Release 8.2 of the SAS System.

These 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. This results in valid statistical inferences that properly reflect the uncertainty due to missing values.

The MIANALYZE procedure reads the parameter estimates and associated covariance matrix that are computed by the standard statistical procedure for each imputed data set. The MIANALYZE procedure then derives valid univariate and multivariate inferences for these parameters.

For some parameters of interest, 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. Special cases such as these are described in the "Examples of the Complete-Data Inferences" section on page 214.

Getting Started

The Fitness data set has been altered to contain an arbitrary missing pattern:

```
*----------------- 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 FitMiss;
   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     49.091  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    .
  .     10.25    .     50.388  10.08  168
39.407  12.63  174     46.080  11.17  156
45.441   9.63  164       .      8.92  146
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
;
```

Assume 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 MI Procedure" chapter for a description of these assumptions). The following statements use the MI procedure to impute missing values for the FitMiss data set.

```
proc mi data=FitMiss noprint out=outmi seed=37851;
   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;

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

Parameter Estimates from Imputed Data Sets

Obs	_Imputation_	_TYPE_	_NAME_	Intercept	RunTime	RunPulse
1	1	PARMS		97.2874	-2.98892	-0.10684
2	1	COV	Intercept	55.7516	-0.73348	-0.27870
3	1	COV	RunTime	-0.7335	0.15167	-0.00509
4	1	COV	RunPulse	-0.2787	-0.00509	0.00194
5	2	PARMS		90.9324	-2.93338	-0.07391
6	2	COV	Intercept	37.5576	-0.25970	-0.20442
7	2	COV	RunTime	-0.2597	0.13978	-0.00722
8	2	COV	RunPulse	-0.2044	-0.00722	0.00166

Figure 10.1. Parameter Estimates

The following statements combine the five sets of regression coefficients:

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

The MIANALYZE Procedure

Model Information

Data Set	WORK.OUTREG
Number of Imputations	5

Multiple Imputation Variance Information

	Variance			
Parameter	Between	Within	Total	DF
Intercept	74.179857	57.287519	146.303348	10.805
RunTime	0.034202	0.142151	0.183193	79.694
RunPulse	0.001533	0.002304	0.004144	20.292

Multiple Imputation Variance Information

Parameter	Relative Increase in Variance	Fraction Missing Information
Intercept	1.553843	0.665161
RunTime	0.288719	0.242803
RunPulse	0.798522	0.491731

Figure 10.2. Model Information and Variance Information Tables

The "Model Information" table lists the input data set(s) and the number of imputations. The "Multiple Imputation Variance Information" table 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, and the fraction of missing information for each parameter estimate.

```
                        The MIANALYZE Procedure

                  Multiple Imputation Parameter Estimates

Parameter        Estimate      Std Error    95% Confidence Limits        DF

Intercept       92.156840      12.095592     65.47596     118.8377   10.805
RunTime         -2.955317       0.428011     -3.80714      -2.1035   79.694
RunPulse        -0.079851       0.064376     -0.21401       0.0543   20.292

                  Multiple Imputation Parameter Estimates

                  Parameter         Minimum        Maximum

                  Intercept       77.939497      99.920480
                  RunTime         -3.159663      -2.660085
                  RunPulse        -0.112277      -0.015111

                  Multiple Imputation Parameter Estimates

                                          t for H0:
          Parameter          Theta0   Parameter=Theta0   Pr > |t|

          Intercept               0              7.62     <.0001
          RunTime                 0             -6.90     <.0001
          RunPulse                0             -1.24     0.2290
```

Figure 10.3. Multiple Imputation Parameter Estimates

The "Multiple Imputation Parameter Estimates" table 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.

Syntax

The following statements are available in PROC MIANALYZE.

PROC MIANALYZE < *options* > **;**

BY *variables* **;**
VAR *variables* **;**

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

The VAR statement lists the numeric variables to be analyzed. The statement is required.

The rest of this section gives detailed syntax information for each of these statements. The PROC MIANALYZE statement and the VAR statement are required for the MIANALYZE procedure.

PROC MIANALYZE Statement

PROC MIANALYZE < *options* > **;**

The following table summarizes the options in the PROC MIANALYZE statement.

Table 10.1. Summary of PROC MIANALYZE Options

Tasks	Options
Specify input data sets	
a COV, CORR, or EST type data set	DATA=
parameter estimates and covariance matrices	PARMS=, COVB=
parameter estimates and $(X'X)^{-1}$ matrices	PARMS=, XPXI=
Specify statistical analysis	
parameters under the null hypothesis	THETA0=
level for the confidence interval	ALPHA=
complete-data degrees of freedom	EDF=
multivariate inferences	MULT

The following are explanations of the options that can be used in the PROC MIANALYZE statement (in alphabetical order):

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

COVB=*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.

DATA=*SAS-data-set*
names a specially structured input SAS data set that contains estimates from imputed data sets. This data set must have a TYPE of EST, COV, or CORR:

- 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=, COVB=, or XPXI= option, then the most recently created SAS data set is used as an input DATA= data set.

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=∞ and the degrees of freedom for each parameter estimate is not adjusted.

MULT
MULTIVARIATE

requests multivariate inference for the variables.

PARMS=*SAS-data-set*

names an input SAS data set that contains parameter estimates computed from imputed data sets. If you provide a PARMS= data set, you must also provide a COVB= or XPXI= data set.

THETA0=*numbers*
MU0=*numbers*

specifies the parameter values $\boldsymbol{\theta}_0$ under the null hypothesis $\boldsymbol{\theta} = \boldsymbol{\theta}_0$ in the *t* tests for location for the variables. If only one number $\boldsymbol{\theta}_0$ is specified, that number is used for all variables. If more than one number is specified, the specified numbers correspond to variables in the VAR statement in the order in which they appear in the VAR statement.

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

- Sort the data using the SORT procedure with a similar BY statement.
- Specify the BY statement option NOTSORTED or DESCENDING in the BY statement for the MI 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 using the DATASETS procedure.

For more information on the BY statement, refer to the discussion in *SAS Language Reference: Concepts, Version 8*. For more information on the DATASETS procedure, refer to the discussion in the *SAS Procedures Guide, Version 8*.

VAR Statement

VAR *variables* **;**

The VAR statement lists the variables to be analyzed. The statement is required, and the variables must be numeric.

Details

Input Data Sets

You can specify input data sets using one of the following option combinations:

- DATA=, which provides both parameter estimates and the associated covariance matrix in a single input data set.
- PARMS= and COVB=, which provide parameter estimates and the associated covariance matrix in separate data sets, respectively.
- PARMS= and XPXI=, 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 SAS procedure you used to create the parameter estimates and associated covariance matrix. For instance, if you used PROC REG to create an OUTEST= data set containing the parameter estimates and covariance matrix, you would use the DATA= option to read the OUTEST= data set. Each input data set contains the variable _Imputation_ to identify the imputation by number.

If you do not specify an input data set with the DATA=, COVB=, or XPXI= option, then the most recently created SAS data set is used as an input DATA= data set.

DATA= data set

The input DATA= data set is a specially structured SAS data set created by statistical procedures available with SAS software. The data set must have a TYPE of EST, COV, or CORR.

With TYPE=EST, the MIANALYZE procedure reads parameter estimates from observations with _TYPE_='PARM' or _TYPE_='PARMS', 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= 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 procedures such as PROC MIXED and PROC GENMOD.

The MIANALYZE procedure uses a PARMS= data set to read parameter names from the variable Parameter or Effect, and it reads parameter estimates from the variable Estimate.

The MIANALYZE procedure uses a COVB= data set to read parameter names from the variable Parameter, Effect, or RowName, and it reads covariance matrices from the subsequent variables Col1, Col2, ... or Prm1, Prm2, ... in the data set.

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 using a procedure such as PROC GLM.

The MIANALYZE procedure uses a PARMS= data set to read parameter names from the variable Parameter; it reads parameter estimates from the variable Estimate, and it reads standard errors for parameter estimates from the variable StdErr.

The MIANALYZE procedure uses an XPXI= data set to read parameter names from the variable Parameter, and it reads $(X'X)^{-1}$ matrices from the subsequent variables in the data set.

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. Let $\hat{Q}_i$ and $\hat{U}_i$ be the point and variance estimates from the ith imputed data set, i=1, 2, ..., m. Then the combined point estimate for Q from multiple imputation is the average of the m complete-data estimates:

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

Let $\overline{U}$ be the within-imputation variance, which is the average of the m complete-data estimates:

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

and B be the between-imputation variance

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

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

$$T = \overline{U} + (1 + \frac{1}{m})B$$

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

$$v_m = (m-1)[1 + \frac{\overline{U}}{(1+m^{-1})B}]^2$$

When the complete-data degrees of freedom v_0 is 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. Barnard and Rubin (1999) recommend the use of an 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 is used.

The degrees of freedom v_m depends on m and the ratio

$$r = \frac{(1 + m^{-1})B}{\overline{U}}$$

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 will be large and the distribution of $(Q - \overline{Q})T^{-(1/2)}$ will be approximately normal.

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

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

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

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 λ (Rubin 1987, p. 114).

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

The following table shows relative efficiencies with different values of m and λ. For cases with little missing information, only a small number of imputations are necessary.

Table 10.2. Relative Efficiency

	λ				
m	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

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 $\hat{\mathbf{U}}_i$ are the point and covariance matrix estimates for a vector-valued parameter $\mathbf{Q}$ (such as a multivariate mean) from the ith imputed data set, i=1, 2, ..., m. Then the combined point estimate for $\mathbf{Q}$ from the multiple imputation is the average of the m complete-data estimates:

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

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

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

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

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

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

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

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

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

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).

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

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 gives 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 $\boldsymbol{\mu}$, the usual estimate is the sample mean vector

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

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

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

These statistics can be computed from a procedure such as CORR. This approach is illustrated in Example 10.1.

Regression Coefficients

Many SAS procedures are available for regression analysis. Among them, REG provides the most general analysis capabilities, and others like LOGISTIC and 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 10.2.

Other procedures, such as GLM, MIXED, and GENMOD, do not generate EST type data sets for regression coefficients. For MIXED and 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 10.3 for PROC MIXED and in Example 10.4 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 10.5.

Correlation Coefficients

For the population correlation coefficient ρ, a point estimate is the sample correlation coefficient r. However, for nonzero ρ, 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 ρ can be obtained by applying the inverse transformation

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

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

This approach is illustrated in Example 10.3.

Ratios of Variable Means

For the ratio μ_1/μ_2 of means for variables Y_1 and Y_2, the point estimate is $\overline{y}_1/\overline{y}_2$, 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)

$$\frac{1}{n}\left[\left(\frac{\overline{y}_1}{\overline{y}_2^2}\right)^2 s_{22} - 2\left(\frac{\overline{y}_1}{y_2^2}\right)\left(\frac{1}{\overline{y}_2}\right)s_{12} + \left(\frac{1}{\overline{y}_2}\right)^2 s_{11}\right]$$

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). This approach is illustrated in Example 10.7.

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 the following table. For more information on ODS, refer to the chapter "Using the Output Delivery System" in the *SAS/STAT User's Guide, Version 8.*

Table 10.3. ODS Tables Produced in PROC MIANALYZE

ODS Table Name	Description	Option
ModelInfo	Model information	
VarianceInfo	Variance information	
ParmEst	Parameter estimates	
BetweenCov	Between-imputation covariance matrix	MULT
WithinCov	Within-imputation covariance matrix	MULT
TotalCov	Total covariance matrix	MULT
MultInf	Multivariate inference	MULT

Examples

The following statements generate five imputed data sets to be used in this section. The data set FitMiss was created in the section "Getting Started" on page 203. See "The MI Procedure" chapter for details concerning the MI procedure.

```
proc mi data=FitMiss seed=37851 noprint out=outmi;
   var Oxygen RunTime RunPulse;
run;
```

Example 10.1. 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 out=outcov(type=cov) nocorr noprint;
   var Oxygen RunTime RunPulse;
   by _Imputation_;
run;

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

Output 10.1.1. COV Data Set

CORR Means and Covariance Matrices (First Two Imputations)

Obs	_Imputation_	_TYPE_	_NAME_	Oxygen	RunTime	RunPulse
1	1	COV	Oxygen	28.2139	-5.9997	-29.700
2	1	COV	RunTime	-5.9997	1.8353	4.812
3	1	COV	RunPulse	-29.7002	4.8121	143.368
4	1	MEAN		47.3458	10.5859	171.301
5	1	STD		5.3117	1.3547	11.974
6	1	N		31.0000	31.0000	31.000
7	2	COV	Oxygen	28.7419	-6.6895	-39.006
8	2	COV	RunTime	-6.6895	2.0553	8.938
9	2	COV	RunPulse	-39.0060	8.9385	172.998
10	2	MEAN		47.3316	10.6004	169.204
11	2	STD		5.3612	1.4336	13.153
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 $V(\overline{\mathbf{y}}) = V(\mathbf{y})/n$, 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 mult;
   var Oxygen RunTime RunPulse;
run;
```

Output 10.1.2. Multiple Imputation Variance Information

The MIANALYZE Procedure

Multiple Imputation Variance Information

Parameter	Variance: Between	Within	Total	DF
Oxygen	0.007552	0.969527	0.978590	27.904
RunTime	0.001577	0.070505	0.072397	27.317
RunPulse	1.363982	4.469865	6.106642	15.052

Multiple Imputation Variance Information

Parameter	Relative Increase in Variance	Fraction Missing Information
Oxygen	0.009348	0.009304
RunTime	0.026834	0.026466
RunPulse	0.366181	0.292981

The "Multiple Imputation Variance Information" table 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 and the fraction of missing information for each variable are also displayed. A detailed description of these statistics is provided in the "Combining Inferences from Imputed Data Sets" section on page 211 and the "Multiple Imputation Efficiency" section on page 212.

Output 10.1.3. Multiple Imputation Parameter Estimates

The MIANALYZE Procedure

Multiple Imputation Parameter Estimates

Parameter	Estimate	Std Error	95% Confidence Limits		DF
Oxygen	47.360514	0.989237	45.3338	49.3872	27.904
RunTime	10.557687	0.269067	10.0059	11.1095	27.317
RunPulse	169.970152	2.471162	164.7046	175.2357	15.052

Multiple Imputation Parameter Estimates

Parameter	Minimum	Maximum
Oxygen	47.273244	47.506505
RunTime	10.505186	10.600445
RunPulse	169.053679	171.301061

Multiple Imputation Parameter Estimates

Parameter	Theta0	t for H0: Parameter=Theta0	Pr > |t|
Oxygen	0	47.88	<.0001
RunTime	0	39.24	<.0001
RunPulse	0	68.78	<.0001

The "Multiple Imputation Parameter Estimates" table 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.

With the MULT option, the procedure displays the between-imputation covariance matrix, within-imputation covariance matrix, and total covariance matrix for multivariate inference.

Output 10.1.4. Within-Imputation Covariance Matrices

The MIANALYZE Procedure

Within-Imputation Covariance Matrix

	Oxygen	RunTime	RunPulse
Oxygen	0.969526960	-0.222851446	-0.997345212
RunTime	-0.222851446	0.070505098	0.222928579
RunPulse	-0.997345212	0.222928579	4.469864543

Between-Imputation Covariance Matrix

	Oxygen	RunTime	RunPulse
Oxygen	0.007552454	-0.002444417	0.068595335
RunTime	-0.002444417	0.001576638	-0.010973127
RunPulse	0.068595335	-0.010973127	1.363981626

Total Covariance Matrix

	Oxygen	RunTime	RunPulse
Oxygen	1.160483133	-0.266743840	-1.193780414
RunTime	-0.266743840	0.084391647	0.266836165
RunPulse	-1.193780414	0.266836165	5.350240500

Assuming that the between-imputation covariance matrix is proportional to the within-imputation covariance matrix, the procedure also displays a multivariate inference for all the parameters taken jointly.

Output 10.1.5. Multiple Imputation Multivariate Inference

The MIANALYZE Procedure

Multiple Imputation Multivariate Inference
Assuming Proportionality of Between/Within Covariance Matrices

Avg Relative Increase in Variance	Num DF	Den DF	F for H0: Parameter=Theta0	Pr > F
0.196958	3	222.91	11408.0	<.0001

Assuming that the within-imputation covariance matrix is proportional to the between-imputation covariance matrix, the table shows a significant p-value for the null hypothesis that the population means are all equal to zero.

With the exception of the multivariate inference, the preceding results could also have been obtained with the MI procedure.

Example 10.2. 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;

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 10.2.1. EST Type Data Set

REG Model Coefficients and Covariance matrices (First Two Imputations)

Obs	_Imputation_	_TYPE_	_NAME_	Intercept	RunTime	RunPulse
1	1	PARMS		97.2874	-2.98892	-0.10684
2	1	COV	Intercept	55.7516	-0.73348	-0.27870
3	1	COV	RunTime	-0.7335	0.15167	-0.00509
4	1	COV	RunPulse	-0.2787	-0.00509	0.00194
5	2	PARMS		90.9324	-2.93338	-0.07391
6	2	COV	Intercept	37.5576	-0.25970	-0.20442
7	2	COV	RunTime	-0.2597	0.13978	-0.00722
8	2	COV	RunPulse	-0.2044	-0.00722	0.00166

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;
   var Intercept RunTime RunPulse;
run;
```

Output 10.2.2. Multiple Imputation Variance Information

The MIANALYZE Procedure

Multiple Imputation Variance Information

Parameter	Variance: Between	Variance: Within	Variance: Total	DF
Intercept	74.179857	57.287519	146.303348	5.2619
RunTime	0.034202	0.142151	0.183193	16.195
RunPulse	0.001533	0.002304	0.004144	8.4786

Multiple Imputation Variance Information

Parameter	Relative Increase in Variance	Fraction Missing Information
Intercept	1.553843	0.665161
RunTime	0.288719	0.242803
RunPulse	0.798522	0.491731

The "Multiple Imputation Variance Information" table displays the between-imputation, within-imputation, and total variances for combining complete-data inferences.

Output 10.2.3. Multiple Imputation Parameter Estimates

The MIANALYZE Procedure

Multiple Imputation Parameter Estimates

Parameter	Estimate	Std Error	95% Confidence Limits		DF
Intercept	92.156840	12.095592	61.52383	122.7898	5.2619
RunTime	-2.955317	0.428011	-3.86177	-2.0489	16.195
RunPulse	-0.079851	0.064376	-0.22686	0.0672	8.4786

Multiple Imputation Parameter Estimates

Parameter	Minimum	Maximum
Intercept	77.939497	99.920480
RunTime	-3.159663	-2.660085
RunPulse	-0.112277	-0.015111

Multiple Imputation Parameter Estimates

Parameter	Theta0	t for H0: Parameter=Theta0	Pr > |t|
Intercept	0	7.62	0.0005
RunTime	0	-6.90	<.0001
RunPulse	0	-1.24	0.2481

The "Multiple Imputation Parameter Estimates" table 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.

Example 10.3. Reading Mixed Model Results from PARMS= and COVB= Data Sets

This example creates data sets containing 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/solution covb;
   by _Imputation_;
   ods output SolutionF=mixparms CovB=mixcovb;
run;

proc print data=mixparms (obs=6);
   var _Imputation_ Effect Estimate;
   title 'MIXED Model Coefficients (First Two Imputations)';
run;

proc print data=mixcovb (obs=6);
   var _Imputation_ Effect Col1 Col2 Col3;
   title 'MIXED Covariance Matrices (First Two Imputations)';
run;
```

Output 10.3.1. PROC MIXED Model Coefficients

MIXED Model Coefficients (First Two Imputations)

Obs	_Imputation_	Effect	Estimate
1	1	Intercept	97.2874
2	1	RunTime	-2.9889
3	1	RunPulse	-0.1068
4	2	Intercept	90.9324
5	2	RunTime	-2.9334
6	2	RunPulse	-0.07391

Output 10.3.2. PROC MIXED Covariance Matrices

Covariance Matrices (First Two Imputations)

Obs	_Imputation_	Effect	Col1	Col2	Col3
1	1	Intercept	55.7516	-0.7335	-0.2787
2	1	RunTime	-0.7335	0.1517	-0.00509
3	1	RunPulse	-0.2787	-0.00509	0.001942
4	2	Intercept	37.5576	-0.2597	-0.2044
5	2	RunTime	-0.2597	0.1398	-0.00722
6	2	RunPulse	-0.2044	-0.00722	0.001661

The following statements use the MIANALYZE procedure with PARMS= and COVB= input data sets to produce the same results as in Example 10.2:

```
proc mianalyze parms=mixparms covb=mixcovb edf=28;
   var Intercept RunTime RunPulse;
run;
```

Example 10.4. Reading Generalized Linear Model Results from PARMS= and COVB= Data Sets

This example creates data sets containing 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
              CovB=gmcovb;
run;

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

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

Output 10.4.1. PROC GENMOD Model Coefficients

GENMOD Model Coefficients (First Two Imputations)

Obs	_Imputation_	Parameter	Estimate
1	1	Intercept	97.2874
2	1	RunTime	-2.9889
3	1	RunPulse	-0.1068
4	1	Scale	2.6227
5	2	Intercept	90.9324
6	2	RunTime	-2.9334
7	2	RunPulse	-0.0739
8	2	Scale	2.4567

The following table displays the covariance matrices for the first two imputed data sets. Note that the GENMOD procedure computes maximum likelihood estimates for the covariance matrix.

Output 10.4.2. PROC GENMOD Covariance Matrices

GENMOD Covariance Matrices (First Two Imputations)

Obs	_Imputation_	Row Name	Prm1	Prm2	Prm3
1	1	Prm1	50.356306	-0.662498	-0.251728
2	1	Prm2	-0.662498	0.1369885	-0.004598
3	1	Prm3	-0.251728	-0.004598	0.0017536
4	1	Scale	-4.14E-16	3.635E-16	-1.53E-17
5	2	Prm1	33.923008	-0.234572	-0.184639
6	2	Prm2	-0.234572	0.1262507	-0.006523
7	2	Prm3	-0.184639	-0.006523	0.0014999
8	2	Scale	1.858E-14	8.886E-16	-1.58E-16

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

```
proc mianalyze parms=gmparms covb=gmcovb;
   var 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 from Example 10.2, but 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 10.2, unbiased estimates for the standard errors are combined with the EDF= option.

Example 10.5. Reading GLM Results from PARMS= and XPXI= Data Sets

This example creates data sets containing 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;
run;

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

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

Output 10.5.1. PROC GLM Model Coefficients

GLM Model Coefficients (First Two Imputations)

Obs	_Imputation_	Parameter	Estimate
1	1	Intercept	97.28741708
2	1	RunTime	-2.98892274
3	1	RunPulse	-0.10683710
4	2	Intercept	90.93235575
5	2	RunTime	-2.93337517
6	2	RunPulse	-0.07390872

Output 10.5.2. PROC GLM $(X'X)^{-1}$ Matrices

GLM X'X Inverse Matrices (First Two Imputations)

Obs	_Imputation_	Parameter	Intercept	RunTime	RunPulse
1	1	Intercept	7.3205589063	-0.096310799	-0.036595038
2	1	RunTime	-0.096310799	0.0199147383	-0.000668435
3	1	RunPulse	-0.036595038	-0.000668435	0.0002549371
4	1	Oxygen	97.28741708	-2.988922744	-0.106837096
5	2	Intercept	5.620924071	-0.038867743	-0.030594091
6	2	RunTime	-0.038867743	0.0209193054	-0.00108086
7	2	RunPulse	-0.030594091	-0.00108086	0.0002485262
8	2	Oxygen	90.932355745	-2.933375175	-0.073908724

The following statements use the MIANALYZE procedure with PARMS= and XPXI= input data sets to produce the same results as in Example 10.2:

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

Example 10.6. Combining Correlation Coefficients

This example combines sample correlation coefficients and associated variances computed from a set of imputed data sets.

The following statements use the CORR procedure to compute the correlation r between variables Oxygen and RunTime for each imputed data set:

```
proc corr data=outmi out=outcorr;
   var Oxygen RunTime;
   by _Imputation_;
run;

proc print data=outcorr (obs=10);
   title 'Correlations (First Two Imputations)';
run;
```

Output 10.6.1. CORR Type Data Set

Correlations (First Two Imputations)

Obs	_Imputation_	_TYPE_	_NAME_	Oxygen	RunTime
1	1	MEAN		47.3458	10.5859
2	1	STD		5.3117	1.3547
3	1	N		31.0000	31.0000
4	1	CORR	Oxygen	1.0000	-0.8338
5	1	CORR	RunTime	-0.8338	1.0000
6	2	MEAN		47.3316	10.6004
7	2	STD		5.3612	1.4336
8	2	N		31.0000	31.0000
9	2	CORR	Oxygen	1.0000	-0.8704
10	2	CORR	RunTime	-0.8704	1.0000

The following statements compute Fisher's Z transformation of r

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

and the variance estimate corresponding to z, $1/(n-3)$.

```
data ztrans(type=EST);
   set outcorr (drop= RunTime rename= Oxygen= Z);
   if (_type_ = 'N' or _name_ = 'RunTime');

   if (_type_ = 'CORR')  then do;
      _type_= 'PARMS';
      _name_='';
      Z= 0.5 * log((1+Z)/(1-Z));
      end;


else if (_type_ = 'N')  then do;
   _type_='COVB';
   _name_='Z';
   Z= 1. / (Z-3);
   end;
run;

proc print data=ztrans;
   title 'EST Type Data Set with Fisher''s Z Transformation';
run;
```

Output 10.6.2. Fisher's Z Transformation

EST Type Data Set with Fisher's Z Transformation

Obs	_Imputation_	_TYPE_	_NAME_	Z
1	1	COVB	Z	0.03571
2	1	PARMS		-1.20037
3	2	COVB	Z	0.03571
4	2	PARMS		-1.33458
5	3	COVB	Z	0.03571
6	3	PARMS		-1.34657
7	4	COVB	Z	0.03571
8	4	PARMS		-1.20194
9	5	COVB	Z	0.03571
10	5	PARMS		-1.29004

The following statements use the MIANALYZE procedure to generate a combined parameter estimate $\hat{z}$ and variance for Fisher's z. The ODS statement is used to save the parameter estimates in an output data set.

```
proc mianalyze data=ztrans;
   ods output ParmEst=parms;
   var z;
run;
```

Output 10.6.3. Inferences Based on Fisher's Z

The MIANALYZE Procedure

Multiple Imputation Parameter Estimates

Parameter	Estimate	Std Error	95% Confidence Limits		DF
z	-1.274701	0.204097	-1.67720	-0.87220	196.63

Multiple Imputation Parameter Estimates

Parameter	Minimum	Maximum
z	-1.346574	-1.200373

Multiple Imputation Parameter Estimates

Parameter	Theta0	t for H0: Parameter=Theta0	Pr > |t|
z	0	-6.25	<.0001

In addition to the estimate for z, PROC MIANALYZE also generates 95% confidence limits for z, $\hat{z}_{.025}$ and $\hat{z}_{.975}$. An estimate of the correlation coefficient and 95% confidence limits can then be generated from the inverse transformation

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

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

The following statements print the estimate for z and 95% confidence limits for z:

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

Output 10.6.4. Parameter Estimates with 95% Confidence Limits

Parameter Estimates with 95% Confidence Limits

Obs	Estimate	LCLMean	UCLMean
1	-1.274701	-1.67720	-0.87220

The following statements generate an estimate of the correlation coefficient and 95% confidence limits:

```
data corr_ci;
   set parms;
   r=     (exp(2*Estimate)-1)/(exp(2*Estimate)+1);
   r_low= (exp(2*LCLMean)-1) /(exp(2*LCLMean)+1);
   r_upp= (exp(2*UCLMean)-1) /(exp(2*UCLMean)+1);
;

proc print data=corr_ci;
   title 'Estimated Correlation Coefficient'
         ' with 95% Confidence Limits';
   var r r_low r_upp;
run;
```

Output 10.6.5. Estimated Correlation Coefficient

Estimated Correlation Coefficient with 95% Confidence Limits

Obs	r	r_low	r_upp
1	-0.85507	-0.93250	-0.70249

Example 10.7. Combining Ratios of Variable Means

This example combines ratios of variable means and associated variances computed from imputed data sets.

The following statements use the CORR procedure to compute the means and covariance matrix of variables Oxygen and RunTime for each imputed data set. Within each imputation, the data set is sorted so that observations with _TYPE_='MEAN' and _TYPE_='N' are read before observations with _TYPE_='COV'.

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

proc sort data=outcov;
   by _Imputation_ _name_;
run;

proc print data=outcov (obs=10);
   title 'Means and Covariance Matrices (First Two Imputations)';
run;
```

Output 10.7.1. Means and Covariance Matrices

Means and Covariance Matrices (First Two Imputations)

Obs	_Imputation_	_TYPE_	_NAME_	RunTime	Run Pulse
1	1	MEAN		10.5859	171.301
2	1	STD		1.3547	11.974
3	1	N		31.0000	31.000
4	1	COV	RunPulse	4.8121	143.368
5	1	COV	RunTime	1.8353	4.812
6	2	MEAN		10.6004	169.204
7	2	STD		1.4336	13.153
8	2	N		31.0000	31.000
9	2	COV	RunPulse	8.9385	172.998
10	2	COV	RunTime	2.0553	8.938

For each imputation, the following statements compute a ratio estimate of the means for the variables RunPulse and RunTime, and a corresponding variance estimate:

```
data vratio (type=EST);
   set outcov;
   keep _Imputation_ _type_ _name_ ratio;
   retain TMean PMean N VarP;

   if (_type_ = 'N')  then N= RunTime;

   if (_type_ = 'MEAN')  then do;
      TMean= RunTime;
      PMean= RunPulse;
      Ratio= RunPulse / RunTime;
      _type_= 'PARMS';
      end;

   if (_type_ = 'COV' and _name_ = 'RunPulse')
      then VarP= RunPulse;

   if (_type_ = 'COV' and _name_ = 'RunTime')  then do;
      Ratio= ( PMean**2/TMean**4 * RunTime
                     - 2 * PMean/TMean**3 * RunPulse
                     + (1./ TMean**2) * VarP ) / N;
      _name_= 'Ratio';
      end;

   if ( _type_= 'STD' or _type_= 'N' or _name_= 'RunPulse')
      then delete;
run;

proc print data=vratio;
  title 'EST Type Data Set with Ratios of Variable Means';
run;
```

Output 10.7.2. Ratio of Variable Means

EST Type Data Set with Ratios of Variable Means

Obs	_Imputation_	_TYPE_	_NAME_	Ratio
1	1	PARMS		16.1821
2	1	COV	Ratio	0.1348
3	2	PARMS		15.9620
4	2	COV	Ratio	0.1181
5	3	PARMS		16.0602
6	3	COV	Ratio	0.1376
7	4	PARMS		15.9968
8	4	COV	Ratio	0.1409
9	5	PARMS		16.2961
10	5	COV	Ratio	0.1678

The following statements use the MIANALYZE procedure to generate a combined point estimate for the ratio of variable means and a variance estimate:

```
proc mianalyze data=vratio theta0=1;
   var ratio;
run;
```

Output 10.7.3. Inferences for a Ratio of Variable Means

The MIANALYZE Procedure

Multiple Imputation Parameter Estimates

Parameter	Estimate	Std Error	95% Confidence Limits		DF
ratio	16.099445	0.403440	15.30394	16.89495	201.33

Multiple Imputation Parameter Estimates

Parameter	Minimum	Maximum
ratio	15.961997	16.296124

Multiple Imputation Parameter Estimates

Parameter	Theta0	t for H0: Parameter=Theta0	Pr > |t|
ratio	1.000000	37.43	<.0001

The variable RunTime has an estimated mean of about 10.6 and a standard error of about 1.4 in each imputed data set. An estimated coefficient of variation is $(1.4/\sqrt{31})/10.6 = 2.37\%$. Since the coefficient of variation is less than 10%, the ratio of sample means is approximately unbiased and normally distributed.

References

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, Inc.

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.

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, Inc.

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

SAS Institute Inc. (1999), *SAS Language Reference: Concepts, Version 8*, Cary, NC: SAS Institute Inc.

SAS Institute Inc. (1999), *SAS Procedures Guide, Version 8*, Cary, NC: SAS Institute Inc.

SAS Institute Inc. (1999), *SAS/STAT User's Guide, Version 8*, Cary, NC: SAS Institute Inc.

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