SAS/STAT® 12.1 User's Guide
The VARCOMP Procedure (Chapter)

SAS® Documentation

This document is an individual chapter from *SAS/STAT® 12.1 User's Guide*.

The correct bibliographic citation for the complete manual is as follows: SAS Institute Inc. 2012. *SAS/STAT® 12.1 User's Guide*. Cary, NC: SAS Institute Inc.

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

Electronic book 1, August 2012

Chapter 101
The VARCOMP Procedure

Contents

Overview: VARCOMP Procedure

The VARCOMP procedure handles general linear models that have random effects. Random effects are classification effects with levels that are assumed to be randomly selected from an infinite population of possible levels. PROC VARCOMP estimates the contribution of each of the random effects to the variance of the dependent variable.

A single MODEL statement specifies the dependent variables and the effects: main effects, interactions, and nested effects. The effects must be composed of classification variables; no continuous variables are allowed on the right side of the equal sign.

You can specify certain effects as fixed (nonrandom) by putting them first in the MODEL statement and indicating the number of fixed effects with the FIXED= option. An intercept is always fitted and assumed fixed. Except for the effects specified as fixed, all other effects are assumed to be random. Their contribution to the model can be thought of as an observation from a distribution that is normally and independently distributed.

The dependent variables are grouped based on the similarity of their missing values. Each group of dependent variables is then analyzed separately. The columns of the design matrix **X** are formed in the same order in which the effects are specified in the MODEL statement. A singular parameterization involving just 0–1 dummy variables is used, as in the GLM procedure.

You can specify four general methods of estimation in the PROC VARCOMP statement by using the METHOD= option. They are TYPE1 (based on computation of Type I sum of squares for each effect), MIVQUE0, maximum likelihood (METHOD=ML), and restricted maximum likelihood (METHOD=REML). A fifth method, METHOD=GRR, provides a specialized analysis for gauge repeatability and reproducibility (R&R) studies. See the section "Gauge Repeatability and Reproducibility Analysis" on page 8542 for further details. Note that this method, along with the CL option in the MODEL statement for confidence limits, applies only to certain designs, namely balanced one-way or two-way designs. The other four general methods apply to any random-effects model and design.

Other procedures, such as PROC GLM, PROC MIXED, and PROC GLIMMIX, fit similar random effects models. The VARCOMP procedure is usually more computationally efficient for certain special designs and models. See the section "Relationship to PROC MIXED" on page 8547 for a more precise comparison with the MIXED procedure in particular.

The GAUGE application in SAS/QC software provides a graphical interface for computing many of the same statistics as METHOD=GRR in PROC VARCOMP.

Getting Started: VARCOMP Procedure

Analyzing the Cure Rate of Rubber

This example, using data from Hicks (1973), concerns an experiment to determine the sources of variability in cure rates of rubber. The goal of the experiment was to find out if the different laboratories contributed more to the variance of cure rates than did the different batches of raw materials. This information would be useful in trying to control the cure rate of the final product because it would provide insight into the sources of the variability in cure rates. The rubber used was cured at three temperatures, which were taken to be fixed. Three laboratories were chosen at random, and three different batches of raw material were tested at each combination of temperature and laboratory. The following statements read the data into the SAS data set Cure.

```
data Cure;
   input Lab Temp Batch $ Cure @@;
   datalines;
1 145 A 18.6   1 145 A 17.0   1 145 A 18.7   1 145 A 18.7
1 145 B 14.5   1 145 B 15.8   1 145 B 16.5   1 145 B 17.6
1 145 C 21.1   1 145 C 20.8   1 145 C 21.8   1 145 C 21.0
1 155 A  9.5   1 155 A  9.4   1 155 A  9.5   1 155 A 10.0
1 155 B  7.8   1 155 B  8.3   1 155 B  8.9   1 155 B  9.1
1 155 C 11.2   1 155 C 10.0   1 155 C 11.5   1 155 C 11.1
1 165 A  5.4   1 165 A  5.3   1 165 A  5.7   1 165 A  5.3
1 165 B  5.2   1 165 B  4.9   1 165 B  4.3   1 165 B  5.2
1 165 C  6.3   1 165 C  6.4   1 165 C  5.8   1 165 C  5.6
2 145 A 20.0   2 145 A 20.1   2 145 A 19.4   2 145 A 20.0
2 145 B 18.4   2 145 B 18.1   2 145 B 16.5   2 145 B 16.7
2 145 C 22.5   2 145 C 22.7   2 145 C 21.5   2 145 C 21.3
2 155 A 11.4   2 155 A 11.5   2 155 A 11.4   2 155 A 11.5
2 155 B 10.8   2 155 B 11.1   2 155 B  9.5   2 155 B  9.7
2 155 C 13.3   2 155 C 14.0   2 155 C 12.0   2 155 C 11.5
2 165 A  6.8   2 165 A  6.9   2 165 A  6.0   2 165 A  5.7
2 165 B  6.0   2 165 B  6.1   2 165 B  5.0   2 165 B  5.2
2 165 C  7.7   2 165 C  8.0   2 165 C  6.6   2 165 C  6.3
3 145 A 19.7   3 145 A 18.3   3 145 A 16.8   3 145 A 17.1
3 145 B 16.3   3 145 B 16.7   3 145 B 14.4   3 145 B 15.2
3 145 C 22.7   3 145 C 21.9   3 145 C 19.3   3 145 C 19.3
3 155 A  9.3   3 155 A 10.2   3 155 A  9.8   3 155 A  9.5
3 155 B  9.1   3 155 B  9.2   3 155 B  8.0   3 155 B  9.0
3 155 C 11.3   3 155 C 11.0   3 155 C 10.9   3 155 C 11.4
3 165 A  6.7   3 165 A  6.0   3 165 A  5.0   3 165 A  4.8
3 165 B  5.7   3 165 B  5.5   3 165 B  4.6   3 165 B  5.4
3 165 C  6.6   3 165 C  6.5   3 165 C  5.9   3 165 C  5.8
;
```

The variables Lab, Temp, and Batch contain levels of laboratory, temperature, and batch, respectively. The Cure variable contains the response values.

The following SAS statements perform a restricted maximum likelihood variance component analysis.

```
title 'Analyzing the Cure Rate of Rubber';
proc varcomp method=reml data=cure;
   class temp lab batch;
   model cure=temp|lab batch(lab temp) / fixed=1;
run;
```

The FIXED=1 option indicates that the first factor, Temp, is fixed. The effect specification Temp|Lab is equivalent to putting the three terms Temp, Lab, and Temp*Lab in the model. Batch(Lab Temp) is equivalent to putting Batch(Temp*Lab) in the MODEL statement. The results of this analysis are displayed in Figure 101.1 through Figure 101.4.

Figure 101.1 Class Level Information

```
                    Analyzing the Cure Rate of Rubber

                 Variance Components Estimation Procedure

                         Class Level Information

                   Class          Levels    Values

                   Temp                3    145 155 165

                   Lab                 3    1 2 3

                   Batch               3    A B C


                 Number of Observations Read         108
                 Number of Observations Used         108


                       Dependent Variable:    Cure
```

Figure 101.1 provides information about the variables used in the analysis and the number of observations and specifies the dependent variable.

Figure 101.2 Iteration History

```
                                            REML Iterations

                                                                    Var(Batch(Temp*
Iteration         Objective          Var(Lab)    Var(Temp*Lab)               Lab))        Var(Error)

        0     13.4500060254      0.5094464340                0        2.4004888633      0.5787185225
        1     13.0898262160      0.3194348317                0        2.0869636935      0.6016005334
        2     13.0893125570      0.3176048001                0        2.0738906134      0.6026217204
        3     13.0893125555      0.3176017115                0        2.0738685461      0.6026234568


                 Convergence criteria met.
```

The “REML Iterations” table in Figure 101.2 displays the iteration history, which includes the value of the objective function associated with REML and the values of the variance components at each iteration.

Figure 101.3 REML Estimates

```
                         REML Estimates

          Variance Component            Estimate

          Var(Lab)                       0.31760
          Var(Temp*Lab)                        0
          Var(Batch(Temp*Lab))           2.07387
          Var(Error)                     0.60262
```

Figure 101.3 displays the REML estimates of the variance components.

Figure 101.4 Covariance Matrix for REML Estimates

```
                  Asymptotic Covariance Matrix of Estimates

                                       Var(Lab)             Var(Temp*Lab)

   Var(Lab)                             0.32452                         0
   Var(Temp*Lab)                              0                         0
   Var(Batch(Temp*Lab))                -0.04998                         0
   Var(Error)                         1.026E-12                         0

                  Asymptotic Covariance Matrix of Estimates

                           Var(Batch(Temp*Lab))                Var(Error)

   Var(Lab)                            -0.04998                 1.026E-12
   Var(Temp*Lab)                              0                         0
   Var(Batch(Temp*Lab))                 0.45042                -0.0022417
   Var(Error)                        -0.0022417                 0.0089668
```

The “Asymptotic Covariance Matrix of Estimates” table in Figure 101.4 displays the asymptotic covariance matrix of the REML estimates.

The results of the analysis show that the variance attributable to Batch(Temp*Lab) (with a variance component of 2.0739) is considerably larger than the variance attributable to Lab (0.3176). Therefore, attempts to reduce the variability of cure rates should concentrate on improving the homogeneity of the batches of raw material used rather than standardizing the practices or equipment within the laboratories. Also, note that since the Batch(Temp*Lab) variance is considerably larger than the experimental error (Var(Error)=0.6026), the Batch(Temp*Lab) variability plays an important part in the overall variability of the cure rates.

Syntax: VARCOMP Procedure

The following statements are available in the VARCOMP procedure:

PROC VARCOMP < *options* > ;
  CLASS *variables* ;
  MODEL *dependent* = < *effects* > < / *options* > ;
  BY *variables* ;

Only one MODEL statement is allowed. The BY, CLASS, and MODEL statements are described after the PROC VARCOMP statement.

PROC VARCOMP Statement

PROC VARCOMP < *options* > ;

The PROC VARCOMP statement invokes the VARCOMP procedure. Table 101.1 summarizes the options available in the VARCOMP statement.

Table 101.1 PROC VARCOMP Statement Options

Option	Description
DATA=	Specifies the input SAS data set to use
EPSILON=	Specifies the convergence value of the objective function
MAXITER=	Specifies the maximum number of iterations for METHOD=ML or METHOD=REML
METHOD=	Specifies which of the five methods to use
SEED=	Specifies an unsigned integer used to start the pseudo-random number generator

You can specify the following options in the PROC VARCOMP statement.

DATA=*SAS-data-set*
specifies the input SAS data set to use. If this option is omitted, the most recently created SAS data set is used.

EPSILON=*number*
specifies the convergence value of the objective function for METHOD=ML or METHOD=REML. By default, EPSILON=1E–8.

MAXITER=*number*
specifies the maximum number of iterations for METHOD=ML or METHOD=REML. By default, MAXITER=50.

METHOD=TYPE1 | MIVQUE0 | ML | REML | GRR < *(options)* >
specifies which of the five methods (TYPE1, MIVQUE0, ML, REML, or GRR) you want to use. By default, METHOD=MIVQUE0. METHOD=GRR provides a specialized analysis only for certain

designs, whereas the other four methods apply to any random-effects model and design. You can specify the following options in parentheses after METHOD=GRR.

SPECLIMITS=*(LSL,USL,< k>)*

SL=*(LSL,USL,< k>)*

specifies the specification limits for the first random factor, which is regarded as the product being tested in the gauge R&R study. The lower limit *(LSL)* must be smaller than the upper limit *(USL)*. The value *k* is optional. The default value is 6, which corresponds to the number of standard deviations between the "natural" tolerance limits containing the middle 99.73% of a normal process. *SPECLIMITS=(LSL,USL,k)* requests the estimates of the parameters PTR(LSL,USL,k) and Cp(LSL,USL,k) to be displayed.

RATIO

specifies that certain additional ratios of variance components should also be computed and displayed, such as proportion of total variance due to the process. These ratios are listed in Table 101.5.

For more information see the section "Computational Methods" on page 8540.

SEED=*n*

specifies an unsigned integer used to start the pseudo-random number generator. If you do not specify a seed or if you specify zero, the seed is generated from reading the time of day from the computer clock. You can use a SAS date as a seed. The random number generation is used in the computation of generalized confidence limits; see the section "Confidence Limits" on page 8544.

BY Statement

BY *variables* **;**

You can specify a BY statement with PROC VARCOMP to obtain separate analyses of 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 VARCOMP 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 to be used in the analysis. All effects in the MODEL statement must be composed of effects that appear in the CLASS statement. Classification variables can be either numeric or character; if they are character, only the first 16 characters are used.

Numeric classification variables are not restricted to integers since a variable's format determines the levels. For more information, see the discussion of the FORMAT statement in *SAS Formats and Informats: Reference*.

MODEL Statement

MODEL *dependent* **=** < *effects* > < / *options* > **;**

The MODEL statement gives the dependent variables and independent effects. If you specify more than one dependent variable, a separate analysis is performed for each one. The independent effects are limited to main effects, interactions, and nested effects; no continuous effects are allowed. All independent effects must be composed of effects that appear in the CLASS statement. Effects are specified in the VARCOMP procedure in the same way as described for the ANOVA procedure. Only one MODEL statement is allowed.

The following options are available in the MODEL statement.

FIXED=*n*

specifies that the first *n* effects in the MODEL statement are fixed effects. The remaining effects are assumed to be random. By default, PROC VARCOMP assumes that all effects are random in the model. Keep in mind that if you use bar notation and, for example, specify Y=A|B / FIXED=2, then A*B is considered a random effect.

CL=

CL=MLS

CL=GCL< *(options)* >

specifies that confidence limits for all of the parameters of interest be computed and displayed. It also optionally specifies the method to use for computing the confidence limits. There are two methods: the modified large-sample (MLS) method and the generalized confidence limits (GCL) method. The default method is MLS. For more information about these two methods, see the section "Confidence Limits" on page 8544.

You can specify the following options in parentheses after CL=GCL.

NSAMPLE=*n*

specifies the sample size for generalized pivot quantities (GPQ) sampling. The default value is 12,605.

EPSILON=*number*

specifies a small positive value used in some GPQ computations. The default value is 0.001.

The CL option applies only to balanced one-way or two-way designs for METHOD=TYPE1 or GRR.

ALPHA=α

specifies the level of significance α for $(1-\alpha)100\%$ two-sided confidence limits. The value of α must be between 0 and 1. By default, α is equal to 0.05.

Details: VARCOMP Procedure

Missing Values

If an observation has a missing value for any variable used in the independent effects, then the analyses of all dependent variables omit this observation. An observation is deleted from the analysis of a given dependent variable if the observation's value for that dependent variable is missing. Note that a missing value in one dependent variable does not eliminate an observation from the analysis of the other dependent variables.

During processing, PROC VARCOMP groups the dependent variables on their missing values across observations so that sums of squares and crossproducts can be computed in the most efficient manner.

Fixed and Random Effects

Central to the idea of variance components models is the idea of fixed and random effects. Each effect in a variance components model must be classified as either a fixed or a random effect. Fixed effects arise when the levels of an effect constitute the entire population in which you are interested. For example, if a plant scientist is comparing the yields of three varieties of soybeans, then Variety would be a fixed effect, providing that the scientist was concerned about making inferences about only these three varieties of soybeans. Similarly, if an industrial experiment focused on the effectiveness of two brands of a machine, Machine would be a fixed effect only if the experimenter's interest did not go beyond the two machine brands.

On the other hand, an effect is classified as a random effect when you want to make inferences about an entire population, and the levels in your experiment represent only a sample from that population. Psychologists comparing test results between different groups of subjects would consider Subject as a random effect. Depending on the psychologists' particular interest, the Group effect might be either fixed or random. For example, if the groups are based on the sex of the subject, then Sex would be a fixed effect. But if the psychologists are interested in the variability in test scores due to different teachers, then they might choose a random sample of teachers as being representative of the total population of teachers, and Teacher would be a random effect. Note that, in the soybean example presented earlier, if the scientists are interested in making inferences about the entire population of soybean varieties and randomly choose three varieties for testing, then Variety would be a random effect.

If all the effects in a model (except for the intercept) are considered random effects, then the model is called a *random-effects model*; likewise, a model with only fixed effects is called a *fixed-effects model*. The more common case, where some factors are fixed and others are random, is called a *mixed model*. In PROC VARCOMP, by default, effects are assumed to be random. You specify which effects are fixed by using the FIXED= option in the MODEL statement. In general, if an interaction or nested effect contains any effect that is random, then the interaction or nested effect should be considered a random effect as well.

In the linear model, each level of a fixed effect contributes a fixed amount to the expected value of the dependent variable. What makes a random effect different is that each level of a random effect contributes an amount that is viewed as a sample from a population of normally distributed variables, each with mean 0, and an unknown variance, much like the usual random error term that is a part of all linear models. The estimate of the variance associated with the random effect is known as the *variance component* because it measures the part of the overall variance contributed by that effect. Thus, PROC VARCOMP estimates the variance of the random variables that are associated with the random effects in your model, and the variance components tell you how much each of the random factors contributes to the overall variability in the dependent variable.

Negative Variance Component Estimates

The variance components estimated by PROC VARCOMP should theoretically be nonnegative because they are assumed to represent the variance of a random variable. Nevertheless, when you are using METHOD=MIVQUE0, TYPE1, or GRR, some estimates of variance components might become negative. (Due to the nature of the algorithms used for METHOD=ML and METHOD=REML, negative estimates are constrained to zero.) These negative estimates might arise for a variety of reasons:

- The variability in your data might be large enough to produce a negative estimate, even though the true value of the variance component is positive.
- Your data might contain outliers. See Hocking (1983) for a graphical technique for detecting outliers in variance components models by using the SAS System.
- A different model for interpreting your data might be appropriate. Under some statistical models for variance components analysis, negative estimates are an indication that observations in your data are negatively correlated. See Hocking (1984) for further information about these models.

Assuming you are satisfied that the model that PROC VARCOMP is using is appropriate for your data, it is common practice to treat negative variance components as if they are zero.

Computational Methods

Four methods of estimation can be specified in the PROC VARCOMP statement by using the METHOD= option. They are described in the following sections.

The Type I Method

This method (METHOD=TYPE1) computes the Type I sum of squares for each effect, equates each mean square involving only random effects to its expected value, and solves the resulting system of equations (Gaylor, Lucas, and Anderson 1970). The $\mathbf{X}'\mathbf{X} \mid \mathbf{X}'\mathbf{Y}$ matrix is computed and adjusted in segments whenever memory is not sufficient to hold the entire matrix.

The MIVQUE0 Method

Based on the technique suggested by Hartley, Rao, and LaMotte (1978), the MIVQUE0 method (METHOD=MIVQUE0) produces unbiased estimates that are invariant with respect to the fixed effects of the model and that are locally best quadratic unbiased estimates given that the true ratio of each component to the residual error component is zero. The technique is similar to TYPE1 except that the random effects are adjusted only for the fixed effects. This affords a considerable timing advantage over the TYPE1 method; thus, MIVQUE0 is the default method used in PROC VARCOMP. The $\mathbf{X}'\mathbf{X}|\mathbf{X}'\mathbf{Y}$ matrix is computed and adjusted in segments whenever memory is not sufficient to hold the entire matrix. Each element (i, j) of the form

$$\mathrm{SSQ}(\mathbf{X}_i'\mathbf{M}\mathbf{X}_j)$$

is computed, where

$$\mathbf{M} = \mathbf{I} - \mathbf{X}_0(\mathbf{X}_0'\mathbf{X}_0)^-\mathbf{X}_0'$$

and where $\mathbf{X}_0$ is part of the design matrix for the fixed effects, $\mathbf{X}_i$ is part of the design matrix for one of the random effects, and SSQ is an operator that takes the sum of squares of the elements. For more information see Rao (1971, 1972) and Goodnight (1978).

The Maximum Likelihood Method

The maximum likelihood method (METHOD=ML) computes maximum likelihood estimates of the variance components; see Searle, Casella, and McCulloch (1992). The computing algorithm makes use of the W-transformation developed by Hemmerle and Hartley (1973) and Goodnight and Hemmerle (1979). The procedure uses a Newton-Raphson algorithm, iterating until the log-likelihood objective function converges.

The objective function for METHOD=ML is $\ln(|\mathbf{V}|) + \mathbf{r}'\mathbf{V}^{-1}\mathbf{r}$, where

$$\mathbf{V} = \sigma_0^2\mathbf{I} + \sum_{i=1}^{n_r} \sigma_i^2\mathbf{X}_i\mathbf{X}_i'$$

and where σ_0^2 is the residual variance, n_r is the number of random effects in the model, σ_i^2 represents the variance components, $\mathbf{X}_i$ is part of the design matrix for one of the random effects, and

$$\mathbf{r} = \mathbf{y} - \mathbf{X}_0(\mathbf{X}_0'\mathbf{V}^{-1}\mathbf{X}_0)^-\mathbf{X}_0'\mathbf{V}^{-1}\mathbf{y}$$

is the vector of residuals.

The Restricted Maximum Likelihood Method

The restricted maximum likelihood method (METHOD=REML) is similar to the maximum likelihood method, but it first separates the likelihood into two parts: one that contains the fixed effects and one that does not (Patterson and Thompson 1971). The procedure uses a Newton-Raphson algorithm, iterating until convergence is reached for the log-likelihood objective function of the portion of the likelihood that does not contain the fixed effects. Using notation from earlier methods, the objective function for METHOD=REML is $\ln(|\mathbf{V}|) + \mathbf{r}'\mathbf{V}^{-1}\mathbf{r} + \ln(|\mathbf{X}_0'\mathbf{V}^{-1}\mathbf{X}_0|)$. See Searle, Casella, and McCulloch (1992) for additional details.

The GRR Method

Based on the technique suggested by Burdick, Borror, and Montgomery (2005), the GRR method (METHOD=GRR) produces minimum variance unbiased estimators.

Gauge Repeatability and Reproducibility Analysis

In a typical gauge R&R experiment, each operator (O_j) makes multiple observations on each of several similar parts (P_i) from a monitored process. The statistical model used to describe the response variable is the balanced two-factor crossed random model with interaction

$$y_{ijk} = \mu_y + P_i + O_j + (PO)_{ij} + E_{ijk}$$

where $i = 1, \ldots, p$, $j = 1, \ldots, o$, $k = 1, \ldots, r$, μ_y is an unknown constant, and $P_i, O_j, (PO)_{ij}, E_{ijk}$ are jointly independent normal random variables with means of zero and variances $\mathrm{Var}(P), \mathrm{Var}(O), \mathrm{Var}(PO)$, and $\mathrm{Var}(E)$, respectively. The corresponding SAS statements are as follows:

```
proc varcomp method=grr;
   class P O;
   model y = P|O;
run;
```

The first random effect in the MODEL statement is assumed to be the “Part” effect and the second is “Operator.”

The ANOVA table for the preceding model is shown in Table 101.2.

Table 101.2 GRR Analysis of Variance

Source	DF	Mean Square	Expected Mean Square
Parts(P)	$p-1$	S_P^2	$\mathrm{Var}(E) + r\mathrm{Var}(PO) + or\mathrm{Var}(P)$
Operators(O)	$o-1$	S_O^2	$\mathrm{Var}(E) + r\mathrm{Var}(PO) + pr\mathrm{Var}(O)$
P×O	$(p-1)(o-1)$	S_{PO}^2	$\mathrm{Var}(E) + r\mathrm{Var}(PO)$
Error(E)	$po(r-1)$	S_E^2	$\mathrm{Var}(E)$

The gauge R&R parameters of interest are given in Table 101.3 in terms of $\mathrm{Var}(P)$, $\mathrm{Var}(O)$, $\mathrm{Var}(PO)$, and $\mathrm{Var}(E)$.

Table 101.3 Gauge R&R Parameters

Parameter	Formula
Mean of population of measurements	$\mu_y = \bar{y}_{\ldots} = \Sigma_{ijk} y_{ijk}/por$
Variance of the monitored process	$\gamma_P = \mathrm{Var}(P)$
Variance of the measurement system	$\gamma_M = \mathrm{Var}(O) + \mathrm{Var}(PO) + \mathrm{Var}(E)$
Total variance of the response variable	$\gamma_y = \mathrm{Var}(y) = \gamma_P + \gamma_M$
Ratio of process variance to measurement variance	$\gamma_R = \gamma_P/\gamma_M$

Table 101.3 (continued)

Parameter	Formula
Proportion of total variance due to the process	$\rho_P = \gamma_P/\gamma_y = \frac{\gamma_R}{1+\gamma_R}$
Proportion of total variance due to the measurement	$\rho_M = \gamma_M/\gamma_y = 1 - \rho_P$
Signal-to-noise ratio	$\mathrm{SNR} = \sqrt{2 \times \gamma_R}$
Discrimination ratio	$\mathrm{DR} = 1 + 2\gamma_R$

For a one-way model, $\gamma_M = \mathrm{Var}(E)$, and for a two-way model with no interaction, $\gamma_M = \mathrm{Var}(O) + \mathrm{Var}(E)$.

If you use the SPECLIMITS option to give specification limits, the two parameters in Table 101.4 will also be estimated and displayed.

Table 101.4 Gauge R&R Parameters Related to Specification Limits

Parameter	Formula
Precision-to-tolerance ratio	$\mathrm{PTR(LSL, USL, k)} = k\sqrt{\gamma_M}/(\mathrm{USL} - \mathrm{LSL})$
Process capability ratio	$\mathrm{Cp(LSL, USL, k)} = (\mathrm{USL} - \mathrm{LSL})/(k\sqrt{\gamma_P})$

Here, USL and LSL are the specification limits, and the value k corresponds to the number of standard deviations between the "natural" tolerance limits of a normal process.

If you use the RATIO option, the ratios in Table 101.5 will also be estimated and displayed.

Table 101.5 Gauge R&R Ratios

Ratio	Formula
Ratio of process variance to total variance	$\mathrm{Var}(P)/\gamma_y$
Ratio of operator variance to total variance	$\mathrm{Var}(O)/\gamma_y$
Ratio of process by operator variance to total variance	$\mathrm{Var}(PO)/\gamma_y$
Ratio of process variance to residual variance	$\mathrm{Var}(P)/\mathrm{Var}(E)$
Ratio of operator variance to residual variance	$\mathrm{Var}(O)/\mathrm{Var}(E)$
Ratio of process by operator variance to residual variance	$\mathrm{Var}(PO)/\mathrm{Var}(E)$

Confidence Limits

When no exact confidence limits exist, it is common practice to use approximate confidence limits. Two such approximations are the modified large-sample (MLS) method and the generalized confidence limit (GCL) method as discussed in Burdick, Borror, and Montgomery (2005). When analyzing a balanced one-way or two-way design, if you specify the CL= option with METHOD=TYPE1 or GRR, the VARCOMP procedure computes confidence limits by using either the MLS method (the default) or the GCL method. Generalized confidence limits are obtained by specifying the CL=GCL option in the MODEL statement.

MLS Confidence Limits

The method of MLS confidence limits was first introduced by Graybill and Wang (1980). It starts with approximate large-sample confidence limits; then it modifies the limits to be exact under certain parameter conditions.

For a balanced two-way crossed random model with interaction, formulas for the MLS method are given in Table 101.6. See Burdick, Borror, and Montgomery (2005) for the formulas for one-way or balanced two-way with no interaction models.

Confidence limits for parameters such as variances and their ratios might not contain the corresponding point estimates, because negative confidence bounds are increased to zero.

Table 101.6 $100(1-\alpha)\%$ MLS Confidence Limits

Parameter	Lower Bound	Upper Bound
μ_y	$\bar{y}_{\dots} - C\sqrt{\frac{K}{por}}$	$\bar{y}_{\dots} + C\sqrt{\frac{K}{por}}$
γ_P	$\hat{\gamma}_P - \sqrt{V_{LP}}/(or)$	$\hat{\gamma}_P + \sqrt{V_{UP}}/(or)$
γ_M	$\hat{\gamma}_M - \sqrt{V_{LM}}/(pr)$	$\hat{\gamma}_M + \sqrt{V_{UM}}/(pr)$
γ_y	$\hat{\gamma}_y - \sqrt{V_{LT}}/(por)$	$\hat{\gamma}_y + \sqrt{V_{UT}}/(por)$
γ_R	L_R	U_R
ρ_P	$L_R/(1+L_R)$	$U_R/(1+U_R)$
ρ_M	$1/(1+U_R)$	$1/(1+L_R)$

The terms in Table 101.6 are defined as follows:

$$
\begin{aligned}
V_{LP} &= G_1^2 S_P^4 + H_3^2 S_{PO}^4 + G_{13} S_P^2 S_{PO}^2 \\
V_{UP} &= H_1^2 S_P^4 + G_3^2 S_{PO}^4 + H_{13} S_P^2 S_{PO}^2 \\
V_{LM} &= G_2^2 S_O^4 + G_3^2 (p-1)^2 S_{PO}^4 + G_4^2 p^2 (r-1)^2 S_E^4 \\
V_{UM} &= H_2^2 S_O^4 + H_3^2 (p-1)^2 S_{PO}^4 + H_4^2 p^2 (r-1)^2 S_E^4 \\
V_{LT} &= G_1^2 p^2 S_P^4 + G_2^2 o^2 S_O^4 + G_3^2 (po-p-o)^2 S_{PO}^4 + G_4^2 (po)^2 (r-1)^2 S_E^4 \\
V_{UT} &= H_1^2 p^2 S_P^4 + H_2^2 o^2 S_O^4 + H_3^2 (po-p-o)^2 S_{PO}^4 + H_4^2 (po)^2 (r-1)^2 S_E^4 \\
L_R &= \frac{p(1-G_1)(S_P^2 - F_1 S_{PO}^2)}{po(r-1)S_E^2 + o(1-G_1)F_3 S_O^2 + o(p-1)S_{PO}^2} \\
U_R &= \frac{p(1+H_1)(S_P^2 - F_2 S_{PO}^2)}{po(r-1)S_E^2 + o(1+H_1)F_4 S_O^2 + o(p-1)S_{PO}^2} \\
G_1 &= 1 - F_{\alpha/2:\infty,p-1} \\
G_2 &= 1 - F_{\alpha/2:\infty,o-1} \\
G_3 &= 1 - F_{\alpha/2:\infty,(p-1)(o-1)} \\
G_4 &= 1 - F_{\alpha/2:\infty,po(r-1)} \\
H_1 &= F_{1-\alpha/2:\infty,p-1} - 1 \\
H_2 &= F_{1-\alpha/2:\infty,o-1} - 1 \\
H_3 &= F_{1-\alpha/2:\infty,(p-1)(o-1)} - 1 \\
H_4 &= F_{1-\alpha/2:\infty,po(r-1)} - 1 \\
F_1 &= F_{1-\alpha/2:p-1,(p-1)(o-1)} \\
F_2 &= F_{\alpha/2:p-1,(p-1)(o-1)} \\
F_3 &= F_{1-\alpha/2:p-1,o-1} \\
F_4 &= F_{\alpha/2:p-1,o-1} \\
G_{13} &= \frac{(F_1-1)^2 - G_1^2 F_1^2 - H_3^2}{F_1} \\
H_{13} &= \frac{(1-F_2)^2 - H_1^2 F_2^2 - G_3^2}{F_2} \\
K &= s_P^2 + s_O^2 - s_{PO}^2 \\
C &= \frac{s_P^2 \sqrt{F_{1-\alpha:1,p-1}} + s_O^2 \sqrt{F_{1-\alpha:1,o-1}} - s_{PO}^2 \sqrt{F_{1-\alpha:1,(p-1)(o-1)}}}{K}
\end{aligned}
$$

The symbol $F_{\alpha:df1,df2}$ represents the percentile of an *F* distribution with *df1* and *df2* degrees of freedom and area α to the left.

Generalized Confidence Limits

The method of generalized confidence limits was first introduced by Weerahandi (1993). The 100(1-α)% generalized confidence limits are determined as follows:

1. Initialize the random number generator with the seed. The seed value is specified by the SEED= option.
2. Sample N generalized pivot quantities (GPQ), defined to have a distribution that is independent of the parameters under study. The value N is specified by the NSAMPLE= option.
3. Define the lower and upper limits as the $\alpha/2$ and $1-\alpha/2$ quantiles of the sampled GPQ values.

Formulas for generalized confidence limits are given in Table 101.7, where Z denotes a standard normal random variable and W_1, W_2, W_3, and W_4 denote jointly independent chi-square random variables that are independent of Z with degrees of freedom $p-1, o-1, (p-1)(o-1)$ and $po(r-1)$, respectively. The value of ϵ in Table 101.7 is specified by the EPSILON= option.

Table 101.7 100$(1-\alpha)$% Generalized Confidence Limits

Parameter	GPQ
μ_y	$\bar{y}_{\dots} - Z\sqrt{\max\left[\epsilon, \frac{(p-1)s_p^2}{porW_1} + \frac{(o-1)s_O^2}{porW_2} - \frac{(p-1)(o-1)s_{PO}^2}{porW_3}\right]}$
γ_P	$\max\left[0, \frac{(p-1)s_p^2}{orW_1} - \frac{(p-1)(o-1)s_{po}^2}{prW_3}\right]$
γ_M	$\frac{(o-1)s_o^2}{prW_2} + \frac{(p-1)^2(0-1)s_p^2o}{prW_3} + \frac{po(r-1)^2s_E^2}{rW_4}$
γ_y	$\frac{(p-1)s_p^2}{orW_1} + \frac{(o-1)s_O^2}{prW_2} + \frac{(po-p-o)(p-1)(o-1)s_{PO}^2}{porW_3} + \frac{po(r-1)^2s_E^2}{rW_4}$
γ_R	$\frac{\text{GPQ}(\gamma_P)}{\text{GPQ}(\gamma_M)}$

In general, the GCL method provides a more accurate confidence interval with a shorter interval width than the MLS method. However, the greater accuracy comes at the cost of being somewhat nondeterministic, because of the reliance on simulation.

Displayed Output

PROC VARCOMP displays the following items:

- Class Level Information for verifying the levels in your data
- Number of observations read from the data set and number of observations used in the analysis
- for METHOD=TYPE1, an analysis-of-variance table with Source, DF, Type I Sum of Squares, Type I Mean Square, and Expected Mean Square, and a table of Type I variance component estimates
- for METHOD=MIVQUE0, the SSQ Matrix containing sums of squares of partitions of the $\mathbf{X}'\mathbf{X}$ crossproducts matrix adjusted for the fixed effects

- for METHOD=ML and METHOD=REML, the iteration history, including the objective function, a table of variance component estimates, and the estimated Asymptotic Covariance Matrix of the variance components
- for METHOD=GRR, an analysis-of-variance table with Source, DF, GRR Sum of Squares, GRR Mean Square, and Expected Mean Square, and a table of GRR parameter estimates. If the CL option is specified, confidence limits for each parameter estimate will also be displayed.

ODS Table Names

PROC VARCOMP assigns a name to each table it creates. You can use these names to reference the table when using the Output Delivery System (ODS) to select tables and create output data sets. These names are listed in Table 101.8. For more information about ODS, see Chapter 20, “Using the Output Delivery System.”

Table 101.8 ODS Tables Produced by PROC VARCOMP

ODS Table Name	Description	Statement
ANOVA	Type 1 analysis of variance	METHOD=TYPE1 or GRR
AsyCov	Asymptotic covariance matrix of estimates	METHOD=ML or REML
ClassLevels	Class level information	default
ConvergenceStatus	Convergence status	METHOD=ML or REML
DepVar	Dependent variable	METHOD=TYPE1, REML, ML, or GRR
DependentInfo	Dependent variable info (multiple variables)	
Estimates	Variance component estimates	default
IterHistory	Iteration history	METHOD=ML or REML
NObs	Number of observations	default
SSCP	Sum of squares matrix	METHOD=MIVQUE0

In situations where multiple dependent variables are analyzed that differ in their missing value pattern, separate names for ANOVA*n*, AsyCov*n*, Estimates*n*, IterHistory*n*, and SSCP*n* tables are no longer required. The results are combined into a single output data set. For METHOD=TYPE1, ML, or REML, the variable Dependent in the output data set identifies the dependent variable. For METHOD=MIVQUE0, a variable is added to the output data set for each dependent variable.

Relationship to PROC MIXED

The MIXED procedure effectively performs the same analyzes as PROC VARCOMP and many others, including Type I, Type II, and Type III tests of fixed effects, confidence limits, customized contrasts, and least squares means. Furthermore, continuous variables are permitted as both fixed and random effects in PROC MIXED, and numerous other covariance structures besides variance components are available. The VARCOMP procedure is more computationally efficient for some special designs and models.

To translate PROC VARCOMP code into PROC MIXED code, move all random effects to the RANDOM statement in PROC MIXED. For example, the syntax for the example in the section "Getting Started: VARCOMP Procedure" on page 8532 is as follows:

```
proc mixed;
   class Temp Lab Batch;
   model Cure = Temp;
   random Lab Temp*Lab Batch(Lab Temp);
run;
```

REML is the default estimation method in PROC MIXED, and you can specify other methods by using the METHOD= option.

Examples: VARCOMP Procedure

Example 101.1: Using the Four General Estimation Methods

In this example, a and b are classification variables and y is the dependent variable. a is declared fixed, and b and a*b are random. Note that this design is unbalanced because the cell sizes are not all the same. PROC VARCOMP is invoked four times, once for each of the general estimation methods. The data are from Hemmerle and Hartley (1973). The following statements produce Output 101.1.1.

```
data a;
   input a b y @@;
   datalines;
1 1 237   1 1 254   1 1 246   1 2 178   1 2 179
2 1 208   2 1 178   2 1 187   2 2 146   2 2 145   2 2 141
3 1 186   3 1 183   3 2 142   3 2 125   3 2 136
;

proc varcomp method=type1 data=a;
   class a b;
   model y=a|b / fixed=1;
run;
```

Output 101.1.1 VARCOMP Procedure: Method=TYPE1

```
                    Variance Components Estimation Procedure

                            Class Level Information

                        Class          Levels    Values

                        a                   3    1 2 3

                        b                   2    1 2
```

Output 101.1.1 *continued*

```
                            Number of Observations Read          16
                            Number of Observations Used          16


                                    Dependent Variable:    y


                                  Type 1 Analysis of Variance

                               Sum of
Source               DF       Squares   Mean Square  Expected Mean Square

a                     2         11736   5868.218750  Var(Error) + 2.725 Var(a*b)
                                                     + 0.1 Var(b) + Q(a)
b                     1         11448         11448  Var(Error) + 2.6308 Var(a*b) + 7.8 Var(b)
a*b                   2    299.041026    149.520513  Var(Error) + 2.5846 Var(a*b)
Error                10    786.333333     78.633333  Var(Error)
Corrected Total      15         24270


                                        Type 1 Estimates

                               Variance Component        Estimate

                               Var(b)                      1448.4
                               Var(a*b)                  27.42659
                               Var(Error)                78.63333
```

The “Class Level Information” table in Output 101.1.1 displays the levels of each variable specified in the CLASS statement. You can check this table to make sure the data are input correctly.

The Type I analysis of variance in Output 101.1.1 consists of a sequential partition of the total sum of squares. The mean square is the sum of squares divided by the degrees of freedom, and the expected mean square is the expected value of the mean square under the mixed model. The “Q” notation in the expected mean squares refers to a quadratic form in parameters of the parenthesized effect.

The Type I estimates of the variance components in Output 101.1.1 result from solving the linear system of equations established by equating the observed mean squares to their expected values.

The following statements are the same as before, except that the estimation method is MIVQUE0 instead of the default TYPE1. They produce Output 101.1.2.

```
proc varcomp method=mivque0 data=a;
   class a b;
   model y=a|b / fixed=1;
run;
```

Output 101.1.2 VARCOMP Procedure: Method=MIVQUE0

```
                  Variance Components Estimation Procedure

                           MIVQUE(0) SSQ Matrix

Source                 b              a*b            Error                y

b               60.84000         20.52000          7.80000          89295.4
a*b             20.52000         20.52000          7.80000          30181.3
Error            7.80000          7.80000         13.00000          12533.5


                           MIVQUE(0) Estimates

                       Variance Component             y

                       Var(b)                    1466.1
                       Var(a*b)               -35.49170
                       Var(Error)             105.73660
```

The MIVQUE0 estimates in Output 101.1.2 result from solving the equations established by the MIVQUE0 SSQ matrix. Note that the estimate of the variance component for the interaction effect, Var(a*b), is negative for this example.

The following statements use METHOD=ML to invoke maximum likelihood estimation. They produce Output 101.1.3.

```
proc varcomp method=ml data=a;
   class a b;
   model y=a|b / fixed=1;
run;
```

Output 101.1.3 VARCOMP Procedure: Method=ML

```
                  Variance Components Estimation Procedure

                       Maximum Likelihood Iterations

Iteration         Objective            Var(b)         Var(a*b)        Var(Error)

        0     78.3850371200        1031.49070                0     74.3909717935
        1     78.2637043807    732.3606453635                0     77.4011688154
        2     78.2635471161    723.6867470850                0     77.5301774839
        3     78.2635471152    723.6658365289                0     77.5304926877


       Convergence criteria met.
```

Output 101.1.3 *continued*

```
                      Maximum Likelihood
                          Estimates

                   Variance
                   Component       Estimate

                   Var(b)         723.66584
                   Var(a*b)               0
                   Var(Error)      77.53049


          Asymptotic Covariance Matrix of Estimates

                      Var(b)         Var(a*b)      Var(Error)

 Var(b)             537826.1                0      -107.33905
 Var(a*b)                  0                0               0
 Var(Error)       -107.33905                0       858.71104
```

The “Maximum Likelihood Iterations” table in Output 101.1.3 shows that the Newton-Raphson algorithm used by PROC VARCOMP requires three iterations to converge.

The ML estimate of Var(a*b) is zero for this example, and the other two estimates are smaller than their Type I and MIVQUE0 counterparts.

One benefit of using likelihood-based methods is that an approximate covariance matrix is available from the matrix of second derivatives evaluated at the ML solution. This covariance matrix is valid asymptotically and can be unreliable in small samples.

Here the variance component estimates for B and the Error are negatively correlated, and the elements for Var(a*b) are set to zero because the estimate equals zero. Also, the very large variance for Var(b) indicates a lot of uncertainty about the estimate for Var(b), and one contributing explanation is that B has only two levels in this data set.

Finally, the following statements use the restricted maximum likelihood (REML) for estimation. They produce Output 101.1.4.

```
proc varcomp method=reml data=a;
   class a b;
   model y=a|b / fixed=1;
run;
```

Output 101.1.4 VARCOMP Procedure: Method=REML

```
                    Variance Components Estimation Procedure

                                REML Iterations

 Iteration        Objective            Var(b)          Var(a*b)        Var(Error)

         0    63.4134144942        1269.52701                 0     91.5581191305
         1    63.0446869787        1601.84199     32.7632417174     76.9355562461
         2    63.0311530508        1468.82932     27.2258186561     78.7548276319
         3    63.0311265148        1464.33646     26.9564053003     78.8431476502
         4    63.0311265127        1464.36727     26.9588525177     78.8423898761


        Convergence criteria met.


                                 REML Estimates

                             Variance
                             Component        Estimate

                             Var(b)             1464.4
                             Var(a*b)         26.95885
                             Var(Error)       78.84239


                   Asymptotic Covariance Matrix of Estimates

                                Var(b)        Var(a*b)       Var(Error)

           Var(b)          4401703.8         1.29359       -273.39651
           Var(a*b)          1.29359          3559.1       -502.85157
           Var(Error)     -273.39651      -502.85157           1249.7
```

The “REML Iterations” table in Output 101.1.4 shows that the REML optimization requires four iterations to converge.

The REML estimates in Output 101.1.4 are all larger than the corresponding ML estimates (adjusting for potential downward bias) and are fairly similar to the Type I estimates.

The “Asymptotic Covariance Matrix of Estimates” table in Output 101.1.4 shows that the Error variance component estimate is negatively correlated with the other two variance component estimates, and the estimated variances are all larger than their ML counterparts.

Example 101.2: Using the GRR Method

In this example from Houf and Burman (1988), the response variable is the thermal performance of a module measured in Celsius degrees per watt. Each of three operators measures 10 parts three times. It is assumed that parts and operators are selected at random from larger populations. The following statements produce Output 101.2.1.

```
data Houf;
   input a b y @@;
   datalines;
1  1 37    1  1 38    1 1 37
1  2 41    1  2 41    1 2 40
1  3 41    1  3 42    1 3 41
2  1 42    2  1 41    2 1 43
2  2 42    2  2 42    2 2 42
2  3 43    2  3 42    2 3 43
3  1 30    3  1 31    3 1 31
3  2 31    3  2 31    3 2 31
3  3 29    3  3 30    3 3 28
4  1 42    4  1 43    4 1 42
4  2 43    4  2 43    4 2 43
4  3 42    4  3 42    4 3 42
5  1 28    5  1 30    5 1 29
5  2 29    5  2 30    5 2 29
5  3 31    5  3 29    5 3 29
6  1 42    6  1 42    6 1 43
6  2 45    6  2 45    6 2 45
6  3 44    6  3 46    6 3 45
7  1 25    7  1 26    7 1 27
7  2 28    7  2 28    7 2 30
7  3 29    7  3 27    7 3 27
8  1 40    8  1 40    8 1 40
8  2 43    8  2 42    8 2 42
8  3 43    8  3 43    8 3 41
9  1 25    9  1 25    9 1 25
9  2 27    9  2 29    9 2 28
9  3 26    9  3 26    9 3 26
10 1 35   10  1 34   10 1 34
10 2 35   10  2 35   10 2 34
10 3 35   10  3 34   10 3 35
;

proc varcomp data=Houf method=grr (speclimits=(18,58) ratio);
   class a b;
   model y=a|b/cl;
run;
```

You specify METHOD=GRR in this example to drive the VARCOMP procedure to produce a gauge repeatability and reproducibility analysis. With the option speclimits=(18 58), the parameters PTR and Cp are estimated and displayed. With the RATIO option, certain additional ratios of variance components are also estimated and displayed. Finally, the CL= option in the MODEL statement specifies that estimates of GRR quantities should have the corresponding confidence limits.

Output 101.2.1 Class Level Information Using Method=GRR

```
                 Variance Components Estimation Procedure

                         Class Level Information

              Class         Levels    Values

              a                 10    1 2 3 4 5 6 7 8 9 10

              b                  3    1 2 3


                 Number of Observations Read          90
                 Number of Observations Used          90


                         Dependent Variable:    y
```

The “Class Level Information” table in Output 101.2.1 displays the levels of each variable specified in the CLASS statement.

Output 101.2.2 Analysis of Variance Using Method=GRR

```
                          GRR Analysis of Variance

                                            Sum of
        Source                 DF          Squares     Mean Square

        a                       9      3935.955556      437.328395
        b                       2        39.266667       19.633333
        a*b                    18        48.511111        2.695062
        Error                  60        30.666667        0.511111
        Corrected Total        89      4054.400000

                          GRR Analysis of Variance

       Source              Expected Mean Square

       a                   Var(Error) + 3 Var(a*b) + 9 Var(a)
       b                   Var(Error) + 3 Var(a*b) + 30 Var(b)
       a*b                 Var(Error) + 3 Var(a*b)
       Error               Var(Error)
       Corrected Total
```

The GRR analysis of variance in Output 101.2.2 is the same as for the Type I analysis when the design is balanced.

Finally, the estimates of the GRR parameters of interest and their confidence limits are displayed in Output 101.2.3.

Output 101.2.3 Parameter Estimates Using Method=GRR

```
                              GRR Estimates

Parameter                  Estimate      95% Confidence Limits

Mu Y                       35.80000      30.49477      41.10523
Var(a)                     48.29259      22.69452     161.63918
Var(b)                      0.56461       0.07296      25.75077
Var(a*b)                    0.72798       0.33273       1.79272
Var(Error)                  0.51111       0.36816       0.75754
Gamma Y                    50.09630      24.48844     166.22217
Gamma P                    48.29259      22.69452     161.63918
Gamma M                     1.80370       1.20623      27.01724
Gamma R                    26.77413       1.69168     105.60895
SNR                         7.31767       1.83939      14.53334
PTR(18,58,6)                0.20145       0.16474       0.77967
Cp(18,58,6)                 0.95933       0.52437       1.39942
DR                         54.54825       4.38336     212.21791
Rho P                       0.96400       0.62848       0.99062
Rho M                       0.03600     0.0093801       0.37152
Var(a)/Gamma Y              0.96400       0.62848       0.99062
Var(b)/Gamma Y              0.01127     0.0008700       0.34151
Var(a*b)/Gamma Y            0.01453     0.0027083       0.04744
Var(a)/Var(Error)          94.48551      40.19199     327.32469
Var(b)/Var(Error)           1.10467       0.13662      50.37744
Var(a*b)/Var(Error)         1.42432       0.55232       3.74691
```

You can draw the following inferences from the results of the analysis. Most of the variation is due to differences between parts because of the relative larger value of Gamma R. The measurement system is nearly inadequate because the PTR exceeds 20%. However, the measurement system is of value in monitoring the process since the SNR is greater than five. See Burdick, Borror, and Montgomery (2003) for more information about interpreting gauge R&R studies.

The confidence limits in Output 101.2.3 are based on large-sample asymptotic approximation. You can alternatively compute more accurate and usually smaller confidence intervals by using CL=GCL for generalized confidence limits. The following statements produce Output 101.2.4:

```
proc varcomp data=Houf method=grr (speclimits=(18,58) ratio) seed=104;
   class a b;
   model y=a|b/cl=gcl;
run;
```

Output 101.2.4 Generalized Confidence Limits

```
                 Variance Components Estimation Procedure

                              GRR Estimates

        Parameter                Estimate    95% Generalized Confidence Limits

        Mu Y                     35.80000      30.48351      41.31148
        Var(a)                   48.29259      22.79316     168.91421
        Var(b)                    0.56461       0.07157      24.28846
        Var(a*b)                  0.72798       0.33476       1.75806
        Var(Error)                0.51111       0.36816       0.75754
        Gamma Y                  50.09630      25.47092     180.85535
        Gamma P                  48.29259      22.79316     168.91421
        Gamma M                   1.80370       1.18494      25.76890
        Gamma R                  26.77413       1.91286      87.60026
        SNR                       7.31767       1.95594      13.23633
        PTR(18,58,6)              0.20145       0.16328       0.76145
        Cp(18,58,6)               0.95933       0.51295       1.39639
        DR                       54.54825       4.82572     176.20052
        Rho P                     0.96400       0.65669       0.98871
        Rho M                     0.03600       0.01129       0.34331
        Var(a)/Gamma Y            0.96400       0.65669       0.98871
        Var(b)/Gamma Y            0.01127     0.0010082       0.32122
        Var(a*b)/Gamma Y          0.01453     0.0032088       0.04300
        Var(a)/Var(Error)        94.48551      40.44585     336.50782
        Var(b)/Var(Error)         1.10467       0.12886      47.19043
        Var(a*b)/Var(Error)       1.42432       0.55232       3.74691
```

Note that the generalized confidence interval widths from Output 101.2.4 for parameters γ_R and DR are 85.7 and 171.4, respectively. These widths are much shorter than the MLS-based widths, which are 103.9 and 207.8 from Output 101.2.3.

In general, the GCL method provides a more accurate confidence interval with a shorter interval width than the MLS method. However, as discussed in the section "Generalized Confidence Limits" on page 8546, they are computationally intensive and somewhat nondeterministic, because they are based on an underlying Monte Carlo simulation.

References

Burdick, R. K., Borror, C. M., and Montgomery, D. C. (2003), "A Review of Methods for Measurement Systems Capability Analysis," *Journal of Quality Technology*, 35, 342–354.

Burdick, R. K., Borror, C. M., and Montgomery, D. C. (2005), *Design and Analysis of Gauge R&R Studies: Making Decisions with Confidence Intervals in Random and Mixed ANOVA Models*, Philadelphia, PA and Alexandria, VA: SIAM and ASA.

Gaylor, D. W., Lucas, H. L., and Anderson, R. L. (1970), "Calculation of Expected Mean Squares by the Abbreviated Doolittle and Square Root Methods," *Biometrics*, 26, 641–655.

Goodnight, J. H. (1978), *Computing MIVQUE0 Estimates of Variance Components*, Technical report, SAS Institute Inc, Cary, NC, SAS Technical Report R-105 Edition.

Goodnight, J. H. and Hemmerle, W. J. (1979), “A Simplified Algorithm for the W-Transformation in Variance Component Estimation,” *Technometrics*, 21, 265–268.

Graybill, F. A. and Wang, C. M. (1980), “Confidence Intervals on Nonnegative Linear Combinations of Variances,” *Journal of the American Statistical Association*, 75, 869–873.

Hartley, H. O., Rao, J. N. K., and LaMotte, L. (1978), “A Simple Synthesis-Based Method of Variance Component Estimation,” *Biometrics*, 34, 233–244.

Hemmerle, W. J. and Hartley, H. O. (1973), “Computing Maximum Likelihood Estimates for the Mixed AOV Model Using the W-Transformation,” *Technometrics*, 15, 819–831.

Hicks, C. R. (1973), *Fundamental Concepts in the Design of Experiments*, New York: Holt, Rinehart and Winston.

Hocking, R. R. (1983), “A Diagnostic Tool for Mixed Models with Applications to Negative Estimates of Variance Components,” in *Proceedings of the Eighth Annual SAS Users Group International Conference*, Cary, NC: SAS Institute Inc.

Hocking, R. R. (1984), *Analysis of Linear Models*, Monterey, CA: Brooks/Cole.

Houf, R. E. and Burman, D. B. (1988), “Statistical Analysis of Power Module Thermal Test Equipment Performance,” *IEEE Transactions on Components Hybrids, and Manufacturing Technology*, 11, 516–520.

Patterson, H. D. and Thompson, R. (1971), “Recovery of Inter-Block Information When Block Sizes Are Unequal,” *Biometrika*, 58, 545–554.

Rao, C. R. (1971), “Minimum Variance Quadratic Unbiased Estimation of Variance Components,” *Journal of Multivariate Analysis*, 1, 445–456.

Rao, C. R. (1972), “Estimation of Variance and Covariance Components in Linear Models,” *Journal of the American Statistical Association*, 67, 112–115.

Searle, S. R., Casella, G., and McCulloch, C. E. (1992), *Variance Components*, New York: John Wiley & Sons.

Weerahandi, S. (1993), “Generalized Confidence Intervals,” *Journal of the American Statistical Association*, 88, 899–905.

Subject Index

Syntax Index

Your Turn

We welcome your feedback.

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