THE POWER TO KNOW.

SAS/STAT® 14.3 User's Guide

The SIMNORMAL Procedure

This document is an individual chapter from *SAS/STAT® 14.3 User's Guide*.

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2017. *SAS/STAT® 14.3 User's Guide*. Cary, NC: SAS Institute Inc.

SAS/STAT® 14.3 User's Guide

SAS Institute Inc., SAS Campus Drive, Cary, NC 27513-2414

September 2017

Chapter 107
The SIMNORMAL Procedure

Contents

Overview: SIMNORMAL Procedure **8913**
Getting Started: SIMNORMAL Procedure **8913**
Syntax: SIMNORMAL Procedure **8915**
PROC SIMNORMAL Statement 8915
BY Statement 8917
CONDITION Statement 8917
VAR Statement 8917
OUT= Output Data Set 8917
Details: SIMNORMAL Procedure **8918**
Introduction 8918
Unconditional Simulation 8918
Conditional Simulation 8919
Example: SIMNORM Procedure **8921**
References **8928**

Overview: SIMNORMAL Procedure

The SIMNORMAL procedure can perform conditional and unconditional simulation for a set of correlated normal or Gaussian random variables.

The means, variances, and covariances (or correlations) are read from an input TYPE=CORR or TYPE=COV data set. This data set is typically produced by the CORR procedure. Conditional simulations are performed by appending a special observation, identified by the value of 'COND' for the _TYPE_ variable, which contains the conditioning value.

The output data set from PROC SIMNORMAL contains simulated values for each of the analysis variables. Optionally, the output data set also contains the seed stream and the values of the conditioning variables. PROC SIMNORMAL produces no printed output.

Getting Started: SIMNORMAL Procedure

The following example illustrates the use of PROC SIMNORMAL to generate two normal random variates that have specified means and covariance.

In this example, the means and covariances are given; these might have come from previous experiments, observational studies, or other considerations.

First you create a _TYPE_=COV data set as the input data set, and then you run PROC SIMNORM with NUMREAL=5000, creating a sample that contains 5,000 observations. The simple statistics of this sample are checked using PROC CORR. The results are shown in Figure 107.1.

```
data scov(type=COV) ;
   input _TYPE_ $ 1-4 _NAME_ $ 9-10 S1 S2 ;
   datalines ;
COV     S1      1.915  0.3873
COV     S2      0.3873 4.321
MEAN            1.305  2.003
run;

proc simnorm data=scov outsim=ssim
               numreal = 5000
               seed = 54321 ;
   var s1 s2 ;
run;

proc corr data=ssim cov ;
   var s1 s2 ;
   title "Statistics for PROC SIMNORM Sample Using NUMREAL=5000" ;
run;
```

Figure 107.1 Statistics for PROC SIMNORM Sample Using NUMREAL=5000

Statistics for PROC SIMNORM Sample Using NUMREAL=5000

The CORR Procedure

2 Variables:	S1	S2

Covariance Matrix, DF = 4999		
	S1	S2
S1	1.895805499	0.424837163
S2	0.424837163	4.132974275

Simple Statistics						
Variable	N	Mean	Std Dev	Sum	Minimum	Maximum
S1	5000	1.30254	1.37688	6513	-3.90682	6.49864
S2	5000	1.98790	2.03297	9940	-5.69812	9.42833

Syntax: SIMNORMAL Procedure

The following statements are available in the SIMNORMAL procedure:

PROC SIMNORMAL DATA=*SAS-data-set < options >* **;**
VAR *variables* **;**
BY *variables* **;**
CONDITION *variables* **;**

Both the PROC SIMNORMAL and VAR statements are required. The following sections describe the PROC SIMNORMAL statement and then describe the other statements in alphabetical order.

PROC SIMNORMAL Statement

PROC SIMNORMAL DATA=*SAS-data-set < options >* **;**

The PROC SIMNORMAL statement invokes the SIMNORMAL procedure. Table 107.1 summarizes the options available in the PROC SIMNORMAL statement.

Table 107.1 Summary of PROC SIMNORMAL Statement Options

Option	Description
Specify Input and Output Data Sets	
DATA=	Specifies input data set (TYPE=CORR, COV, and so on)
OUT=	Creates output data set that contains simulated values
Seed Values	
SEED=	Specifies seed value (integer)
SEEDBY	Requests reinitialization of seed for each BY group
Control Contents of OUT= Data Set	
OUTSEED	Requests seed values written to OUT= data set
OUTCOND	Requests conditioning variable values written to OUT=data set
Control Number of Simulated Values	
NUMREAL=	Specifies the number of realizations for each BY group written to the OUT= data set
Singularity Criteria	
SINGULAR1=	Sets the singularity criterion for Cholesky decomposition
SINGULAR2=	Sets the singularity criterion for covariance matrix sweeping

The following options can be used with the PROC SIMNORMAL statement.

DATA=*SAS-data-set*

specifies the input data set that must be a specially structured TYPE=CORR, COV, UCORR, UCOV, or SSCP SAS data set. If the DATA= option is omitted, the most recently created SAS data set is used.

SEED=*seed-value*

specifies the seed to use for the random number generator. If the SEED= value is omitted, the system clock is used. If the system clock is used, a note is written to the log; the note gives the seed value based on the system clock. In addition, the random seed stream is copied to the OUT= data set if the OUTSEED option is specified.

SEEDBY

specifies that the seed stream be reinitialized for each BY group. By default, a single random stream is used over all BY groups. If you specify SEEDBY, the random stream starts again at the initial seed value. This initial value is from the SEED= value that you specify. If you do not specify a SEED=value, the system clock generates this initial seed.

For example, suppose you had a TYPE=CORR data set with BY groups, and the mean, variances, and covariance or correlation values were identical for each BY group. Then if you specified SEEDBY, the simulated values in each BY group in the OUT= data set would be identical.

OUT=*SAS-data-set*

specifies a SAS data set in which to store the simulated values for the VAR variables. If you omit the OUT=option, the output data set is created and given a default name by using the DATA*n* convention.

See the section "OUT= Output Data Set" on page 8917 for details.

NUMREAL=*n*

specifies the number of realizations to generate. A value of NUMREAL=500 generates 500 observations in the OUT=dataset, or 500 observations within each BY group if a BY statement is given.

NUMREAL can be abbreviated as NUMR or NR.

OUTSEED

requests that the seed values be included in the OUT= data set. The variable Seed is added to the OUT= data set. The first value of Seed is the SEED= value specified in the PROC SIMNORMAL statement (or obtained from the system clock); subsequent values are produced by the random number generator.

OUTCOND

requests that the values of the conditioning variables be included in the OUT= data set. These values are constant for the data set or within a BY group. Note that specifying OUTCOND can greatly increase the size of the OUT= data set. This increase depends on the number of conditioning variables.

SINGULAR1=*number*

specifies the first singularity criterion, which is applied to the Cholesky decomposition of the covariance matrix. The SINGULAR1= value must be in the range $(0, 1)$. The default value is 10^{-8}. SINGULAR1 can be abbreviated SING1.

SINGULAR2=*number*

specifies the second singularity criterion, which is applied to the sweeping of the covariance or correlation matrix to obtain the conditional covariance. The SINGULAR2=option is applicable only when a CONDITION statement is given. The SINGULAR2= value must be in the range $(0, 1)$. The default value is 10^{-8}. SINGULAR2 can be abbreviated SING2.

BY Statement

BY *variables* **;**

A BY statement can be used with the SIMNORMAL procedure to obtain separate simulations for each covariance structure defined by the BY variables. When a BY statement appears, the procedure expects the input DATA= data set to be sorted in the order of the BY variables. If a CONDITION statement is used along with a BY statement, there must be a _TYPE_='COND' observation within each BY group. Note that if a BY statement is specified, the number of realizations specified by the NUMREAL= option are produced for each BY group.

CONDITION Statement

CONDITION | COND *variables* **;**

A CONDITION statement specifies the conditioning variables. The presence of a CONDITION statement requests that a conditional simulation be performed.

The lack of a CONDITIONAL statement simply means that an unconditional simulation for the VAR variables is to be performed.

If a CONDITION statement is given, the variables listed must be numeric variables in the DATA= data set. This requires a conditioning value for each of the CONDITION variables. This value is supplied by adding a _TYPE_='COND' observation for each CONDITION variable. Such observations are added to the DATA= data set by a DATA step.

Note that a data set created by the CORR procedure is automatically given the TYPE=COV, UCOV, CORR, or UCORR attribute, so you do not have to specify the TYPE= option in the DATA= option in the PROC SIMNORMAL statement. However, when adding the conditioning values by using a DATA step with a SET statement, you must use the TYPE=COV, UCOV, CORR, or UCORR attribute in the new data set. See the section "Getting Started: SIMNORMAL Procedure" on page 8913 for an example in which the TYPE is set.

VAR Statement

VAR *variables* **;**

Use the VAR statement to specify the analysis variables. Only numeric variables can be specified. If a VAR statement is not given, all numeric variables in the DATA= data set that are not in the CONDITION or BY statement are used.

OUT= Output Data Set

The SIMNORMAL procedure produces a single output data set: the OUT=*SAS-data-set*.

The OUT= data set contains the following variables:

- all variables listed in the VAR statement

- all variables listed in the BY statement, if one is given
- Rnum, which is the realization number within the current BY group
- Seed, which is current seed value, if the OUTSEED option is specified
- all variables listed in the CONDITION statement, if a CONDITION statement is given and the OUTCOND option is specified

The number of observations is determined by the value of the NUMREAL= option. If there are no BY groups, the number of observations in the OUT= data set is equal to the value of the NUMREAL= option. If there are BY groups, there are number of observations equals the value of the NUMREAL= option for each BY group.

Details: SIMNORMAL Procedure

Introduction

There are a number of approaches to simulating a set of dependent random variables. In the context of spatial random fields, these include sequential indicator methods, turning bands, and the Karhunen-Loeve expansion. See Christakos (1992, Chapter 8) and Deutsch and Journel (1992, Chapter 5) for details.

In addition, there is the LU decomposition method, a particularly simple and computationally efficient for normal or Gaussian variates. For a given covariance matrix, the $\mathbf{LU} = \mathbf{LL}'$ decomposition is computed once, and the simulation proceeds by repeatedly generating a vector of independent $N(0, 1)$ random variables and multiplying by the $\mathbf{L}$ matrix.

One problem with this technique is that memory is required to hold the covariance matrix of all the analysis and conditioning variables in core.

Unconditional Simulation

It is a simple matter to produce an $N(0, 1)$ random number, and by stacking k such numbers in a column vector you obtain a vector with independent standard normal components $\mathbf{W} \sim N_k(\mathbf{0}, \mathbf{I})$. The meaning of the terms *independence* and *randomness* in the context of a deterministic algorithm required for the generation of these numbers is somewhat subtle; see Knuth (1973, Vol. 2, Chapter 3) for a discussion of these issues.

Rather than $\mathbf{W} \sim N_k(\mathbf{0}, \mathbf{I})$, what is required is the generation of a vector $\mathbf{Z} \sim N_k(\mathbf{0}, \mathbf{V})$—that is,

$$\mathbf{Z} = \begin{bmatrix} Z_1 \\ Z_2 \\ \vdots \\ Z_k \end{bmatrix}$$

with covariance matrix

$$\mathbf{V} = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1k} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2k} \\ & \ddots & & \\ \sigma_{k1} & \sigma_{k2} & \cdots & \sigma_{kk} \end{pmatrix}$$

where

$$\sigma_{ij} = \mathrm{Cov}(Z_i, Z_j)$$

If the covariance matrix is symmetric and positive definite, it has a Cholesky root $\mathbf{L}$ such that $\mathbf{V}$ can be factored as

$$\mathbf{V} = \mathbf{L}\mathbf{L}'$$

where $\mathbf{L}$ is lower triangular. See Ralston and Rabinowitz (1978, Chapter 9, Section 3-3) for details. This vector $\mathbf{Z}$ can be generated by the transformation $\mathbf{Z} = \mathbf{L}\mathbf{W}$. Note that this is where the assumption of multivariate normality is crucial. If $\mathbf{W} \sim N_k(\mathbf{0}, \mathbf{I}_k)$, then $\mathbf{Z} = \mathbf{L}\mathbf{W}$ is also normal or Gaussian. The mean of Z is

$$\mathrm{E}(\mathbf{Z}) = \mathbf{L}(\mathrm{E}(\mathbf{W})) = 0$$

and the variance is

$$\mathrm{Var}(\mathbf{Z}) = \mathrm{Var}(\mathbf{L}\mathbf{W}) = \mathrm{E}(\mathbf{L}\mathbf{W}\mathbf{W}'\mathbf{L}') = \mathbf{L}\,\mathrm{E}(\mathbf{W}\mathbf{W}')\,\mathbf{L}' = \mathbf{L}\mathbf{L}' = \mathbf{V}$$

Finally, let $Y_k = Z_k + \mu_k$; that is, you add a mean term to each variable Z_k. The covariance structure of the $Y_k's$ remains the same. Unconditional simulation is done by simply repeatedly generating k $N(0, 1)$ random numbers, stacking them, and performing the transformation

$$\mathbf{W} \longmapsto \mathbf{Z} = \mathbf{L}\mathbf{W} \longmapsto \mathbf{Y} = \mathbf{Z} + \boldsymbol{\mu}$$

Conditional Simulation

For a conditional simulation, this distribution of

$$\mathbf{Y} = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_k \end{bmatrix}$$

must be conditioned on the values of the CONDITION variables. The relevant general result concerning conditional distributions of multivariate normal random variables is the following. Let $\mathbf{X} \sim N_m(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{bmatrix}$$

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix}$$

$$\Sigma = \begin{pmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{pmatrix}$$

and where bX_1 is $k \times 1$, X_2 is $n \times 1$, $\boldsymbol{\Sigma}_{11}$ is $k \times k$, $\boldsymbol{\Sigma}_{22}$ is $n \times n$, and $\boldsymbol{\Sigma}_{12} = \boldsymbol{\Sigma}'_{21}$ is $k \times n$, with $k + n = m$. The full vector $\mathbf{X}$ has simply been partitioned into two subvectors, $\mathbf{X}_1$ and $\mathbf{X}_2$, and $\boldsymbol{\Sigma}$ has been similarly partitioned into covariances and cross covariances.

With this notation, the distribution of $\mathbf{X}_1$ conditioned on $\mathbf{X}_2 = x_2$ is $N_k(\tilde{\boldsymbol{\mu}}, \tilde{\boldsymbol{\Sigma}})$, with

$$\tilde{\mu} = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}(\mathrm{x}_2 - \boldsymbol{\mu}_2)$$

and

$$\tilde{\boldsymbol{\Sigma}} = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}$$

See Searle (1971, pp. 46–47) for details.

Using the SIMNORMAL procedure corresponds with the conditional simulation as follows. Let $Y_1, \ldots, Y_k$ be the VAR variables as before (k is the number of variables in the VAR list). Let the mean vector for $\mathbf{Y}$ be denoted by $\boldsymbol{\mu}_1 = \mathrm{E}(\mathbf{Y})$. Let the CONDITION variables be denoted by $C_1, \ldots, C_n$ (where n is the number of variables in the COND list). Let the mean vector for $\mathbf{C}$ be denoted by $\boldsymbol{\mu}_2 = \mathrm{E}(\mathbf{C})$ and the conditioning values be denoted by

$$\mathbf{c} = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix}$$

Then stacking

$$\mathbf{X} = \begin{bmatrix} \mathbf{Y} \\ \mathbf{C} \end{bmatrix}$$

the variance of $\mathbf{X}$ is

$$\mathbf{V} = \mathrm{Var}(\mathbf{X}) = \boldsymbol{\Sigma} = \begin{pmatrix} \mathbf{V}_{11} & \mathbf{V}_{12} \\ \mathbf{V}_{21} & \mathbf{V}_{22} \end{pmatrix}$$

where $\mathbf{V}_{11} = \mathrm{Var}(\mathbf{Y})$, $\mathbf{V}_{12} = \mathrm{Cov}(\mathbf{Y}, \mathbf{C})$, and $\mathbf{V}_{22} = \mathrm{Var}(\mathbf{C})$. By using the preceding general result, the relevant covariance matrix is

$$\tilde{\mathbf{V}} = \mathbf{V}_{11} - \mathbf{V}_{12}\mathbf{V}_{22}^{-1}\mathbf{V}_{21}$$

and the mean is

$$\tilde{\mu} = \boldsymbol{\mu}_1 + \mathbf{V}_{12}\mathbf{V}_{22}^{-1}(\mathbf{c} - \boldsymbol{\mu}_2)$$

By using $\tilde{\mathbf{V}}$ and $\tilde{\boldsymbol{\mu}}$, simulating $(\mathbf{Y}|\mathbf{C} = \mathbf{c}) \sim N_k(\tilde{\boldsymbol{\mu}}, \tilde{\mathbf{V}})$ now proceeds as in the unconditional case.

Example: SIMNORM Procedure

The following example illustrates the use of PROC SIMNORMAL to generate variable values conditioned on a set of related or correlated variables.

Suppose you are given a sample of size 50 from ten normally distributed, correlated random variables, $IN_{1,i}, \ldots, IN_{5,i}, OUT_{1,i}, \ldots, OUT_{5,i}, i = 1, \ldots, 50$. The first five variables represent input variables for a chemical manufacturing process, and the last five are output variables.

First, the data are input and the correlation structure is determined by using PROC CORR, as in the following statements. The results are shown in Figure 107.2.

```
data a ;
   input in1-in5 out1-out5 ;
   datalines ;
 9.3500    10.0964     7.3177    10.3617    10.3444     9.4612
10.7443     9.9026     9.0144    11.7968
 7.8599    10.4560    10.0075     8.5875    10.0014    10.3869

   ... more lines ...

 8.9174     9.9623     9.5742     9.9713
run ;

proc corr data=a cov nocorr outp=outcov ;
   var in1-in5 out1-out5 ;
run ;
```

Figure 107.2 Correlation of Chemical Process Variables

Statistics for PROC SIMNORM Sample Using NUMREAL=5000

The CORR Procedure

10 Variables:	in1	in2	in3	in4	in5	out1	out2	out3	out4	out5

Figure 107.2 *continued*

Covariance Matrix, DF = 49						
	in1	**in2**	**in3**	**in4**	**in5**	**out1**
in1	1.019198331	0.128086799	0.291646382	0.327014916	0.417546732	0.097650713
in2	0.128086799	1.056460818	0.143581799	0.095937707	0.104117743	0.056612934
in3	0.291646382	0.143581799	1.384051249	0.058853960	0.326107730	0.093498839
in4	0.327014916	0.095937707	0.058853960	1.023128678	0.347916864	0.022915645
in5	0.417546732	0.104117743	0.326107730	0.347916864	1.606858140	0.360270318
out1	0.097650713	0.056612934	0.093498839	0.022915645	0.360270318	0.807007554
out2	0.206698403	-0.121700731	0.078294087	0.125961491	0.297046593	0.217285879
out3	0.516271121	0.266581451	0.481576554	0.179627237	0.749212945	0.064816340
out4	0.118726106	0.092288067	0.057816322	0.075028230	0.220196337	-0.053931448
out5	0.261770905	-0.020971411	0.259053423	0.078147576	0.349618466	0.037758721

Covariance Matrix, DF = 49				
	out2	**out3**	**out4**	**out5**
in1	0.206698403	0.516271121	0.118726106	0.261770905
in2	-0.121700731	0.266581451	0.092288067	-0.020971411
in3	0.078294087	0.481576554	0.057816322	0.259053423
in4	0.125961491	0.179627237	0.075028230	0.078147576
in5	0.297046593	0.749212945	0.220196337	0.349618466
out1	0.217285879	0.064816340	-0.053931448	0.037758721
out2	0.929455806	0.206825664	0.138551008	0.054039499
out3	0.206825664	1.837505268	0.292963975	0.165910481
out4	0.138551008	0.292963975	0.832831377	-0.067396486
out5	0.054039499	0.165910481	-0.067396486	0.697717191

Simple Statistics						
Variable	**N**	**Mean**	**Std Dev**	**Sum**	**Minimum**	**Maximum**
in1	50	10.18988	1.00955	509.49400	7.63500	12.58860
in2	50	10.10673	1.02784	505.33640	8.12580	13.78310
in3	50	10.14888	1.17646	507.44420	7.31770	12.40080
in4	50	10.03884	1.01150	501.94200	7.40490	11.99060
in5	50	10.22587	1.26762	511.29340	7.23350	12.93360
out1	50	9.85347	0.89834	492.67340	8.01220	12.24660
out2	50	9.96857	0.96408	498.42840	7.76420	12.09450
out3	50	10.29588	1.35555	514.79410	7.29660	13.74200
out4	50	10.15856	0.91260	507.92780	8.43090	12.45230
out5	50	10.26023	0.83529	513.01130	7.86060	11.96000

After the mean and correlation structure are determined, any subset of these variables can be simulated. Suppose you are interested in a particular function of the output variables for two sets of values of the input variables for the process. In particular, you are interested in the mean and variability of the following function over 500 runs of the process conditioned on each set of input values:

$$f(out_1, \ldots, out_5) = \frac{out_1 - out_3}{out_1 + out_2 + out_3 + out_4 + out_5}$$

Although the distribution of these quantities could be determined theoretically, it is simpler to perform a conditional simulation by using PROC SIMNORMAL.

To do this, you first append a _TYPE_='COND' observation to the covariance data set produced by PROC CORR for each group of input values:

```
data cond1 ;
   _TYPE_='COND' ;
   in1 = 8    ;
   in2 = 10.5 ;
   in3 = 12   ;
   in4 = 13.5 ;
   in5 = 14.4 ;
   output ;
run ;

data cond2 ;
   _TYPE_='COND' ;
   in1 = 15.4 ;
   in2 = 13.7 ;
   in3 = 11 ;
   in4 = 7.9 ;
   in5 = 5.5 ;
   output ;
run ;
```

Next, each of these conditioning observations is appended to a copy of the OUTP=OUTCOV data from the CORR procedure, as in the following statements. A new variable, INPUT, is added to distinguish the sets of input values. This variable is used as a BY variable in subsequent steps.

```
data outcov1 ;
   input=1 ;
   set outcov cond1 ;
run ;

data outcov2 ;
   input=2 ;
   set outcov cond2 ;
run ;
```

Finally, these two data sets are concatenated:

```
data outcov ;
   set outcov1 outcov2 ;
run ;
proc print data=outcov ;
   where (_type_ ne 'COV') ;
run ;
```

Figure 107.3 shows the added observations.

Figure 107.3 OUTP= Data Set from PROC CORR with _TYPE_=COND Observations Appended

Statistics for PROC SIMNORM Sample Using NUMREAL=5000

Obs	input	_TYPE_	_NAME_	in1	in2	in3	in4	in5	out1	out2	out3	out4	out5
11	1	MEAN		10.1899	10.1067	10.1489	10.0388	10.2259	9.8535	9.9686	10.2959	10.1586	10.2602
12	1	STD		1.0096	1.0278	1.1765	1.0115	1.2676	0.8983	0.9641	1.3555	0.9126	0.8353
13	1	N		50.0000	50.0000	50.0000	50.0000	50.0000	50.0000	50.0000	50.0000	50.0000	50.0000
14	1	COND		8.0000	10.5000	12.0000	13.5000	14.4000	.	.	.	.	.
25	2	MEAN		10.1899	10.1067	10.1489	10.0388	10.2259	9.8535	9.9686	10.2959	10.1586	10.2602
26	2	STD		1.0096	1.0278	1.1765	1.0115	1.2676	0.8983	0.9641	1.3555	0.9126	0.8353
27	2	N		50.0000	50.0000	50.0000	50.0000	50.0000	50.0000	50.0000	50.0000	50.0000	50.0000
28	2	COND		15.4000	13.7000	11.0000	7.9000	5.5000	.	.	.	.	.

You now run PROC SIMNORMAL, specifying the input data set and the VAR and COND variables. Note that you must specify a TYPE=COV or TYPE=CORR for the input data set. PROC CORR automatically assigns a TYPE=COV or TYPE=CORR attribute for the OUTP= data set. However, since the intermediate DATA steps that appended the _TYPE_='COND' observations turned off this attribute, an explicit TYPE=CORR in the DATA= option in the PROC SIMNORMAL statement is needed.

The specification of PROC SIMNORMAL now follows from the problem description. The condition variables are IN1–IN5, the analysis variables are OUT1–OUT5, and 500 realizations are required. A seed value can be chosen arbitrarily, or the system clock can be used. Note that in the following statements, the simulation is done for each of the values of the BY variable INPUT:

```
proc simnormal data=outcov(type=cov)
      out = osim
      numreal = 500
      seed = 33179
      ;
   by input ;
   var out1-out5 ;
   cond in1-in5 ;
run;

data b;
   set osim ;
   denom = sum(of out1-out5) ;
   if abs(denom) < 1e-8 then ff = . ;
   else ff = (out1-out3)/denom ;
run ;
```

The DATA step that follows the simulation computes the function $f(out_1, \ldots, out_5)$; in the following statements the UNIVARIATE procedure computes the simple statistics for this function for each set of conditioning input values. This is shown in Figure 107.4, and Figure 107.5 shows the distribution of the function values for each set of input values by using the SGPANEL procedure.

```
proc univariate data=b ;
   by input ;
   var ff ;
run ;
title ;
```

```
proc sgpanel data=b ;
   panelby input ;
   REFLINE 0 / axis= x ;
   density ff ;
run ;
```

Figure 107.4 Simple Statistics for ff for Each Set of Input Values

Statistics for PROC SIMNORM Sample Using NUMREAL=5000

The UNIVARIATE Procedure
Variable: ff

input=1

Moments			
N	500	**Sum Weights**	500
Mean	-0.0134833	**Sum Observations**	-6.7416303
Std Deviation	0.02830426	**Variance**	0.00080113
Skewness	0.56773239	**Kurtosis**	1.31522925
Uncorrected SS	0.49066351	**Corrected SS**	0.39976435
Coeff Variation	-209.92145	**Std Error Mean**	0.0012658

input=1

Basic Statistical Measures			
Location		Variability	
Mean	-0.01348	**Std Deviation**	0.02830
Median	-0.01565	**Variance**	0.0008011
Mode	.	**Range**	0.21127
		Interquartile Range	0.03618

input=1

Tests for Location: Mu0=0				
Test	**Statistic**		**p Value**	
Student's t	**t**	-10.6519	**Pr > |t|**	<.0001
Sign	**M**	-106	**Pr >= |M|**	<.0001
Signed Rank	**S**	-33682	**Pr >= |S|**	<.0001

input=1

Quantiles (Definition 5)	
Level	**Quantile**
100% Max	0.11268600
99%	0.07245656
95%	0.03270269
90%	0.02064338
75% Q3	0.00370322
50% Median	-0.01564850
25% Q1	-0.03247389
10%	-0.04716239
5%	-0.05572806
1%	-0.07201126
0% Min	-0.09858350

Figure 107.4 *continued*

input=1

Extreme Observations			
Lowest		Highest	
Value	Obs	Value	Obs
-0.0985835	471	0.0750538	22
-0.0908179	472	0.0794747	245
-0.0802423	90	0.0840160	48
-0.0760645	249	0.1004812	222
-0.0756070	226	0.1126860	50

Statistics for PROC SIMNORM Sample Using NUMREAL=5000

The UNIVARIATE Procedure
Variable: ff

input=2

Moments			
N	500	**Sum Weights**	500
Mean	-0.0405913	**Sum Observations**	-20.295631
Std Deviation	0.03027008	**Variance**	0.00091628
Skewness	0.1033062	**Kurtosis**	-0.1458848
Uncorrected SS	1.28104777	**Corrected SS**	0.4572225
Coeff Variation	-74.57289	**Std Error Mean**	0.00135372

input=2

Basic Statistical Measures			
Location		Variability	
Mean	-0.04059	**Std Deviation**	0.03027
Median	-0.04169	**Variance**	0.0009163
Mode	.	**Range**	0.18332
		Interquartile Range	0.04339

input=2

Tests for Location: Mu0=0				
Test	Statistic		p Value	
Student's t	**t**	-29.985	**Pr > |t|**	<.0001
Sign	**M**	-203	**Pr >= |M|**	<.0001
Signed Rank	**S**	-58745	**Pr >= |S|**	<.0001

Figure 107.4 *continued*

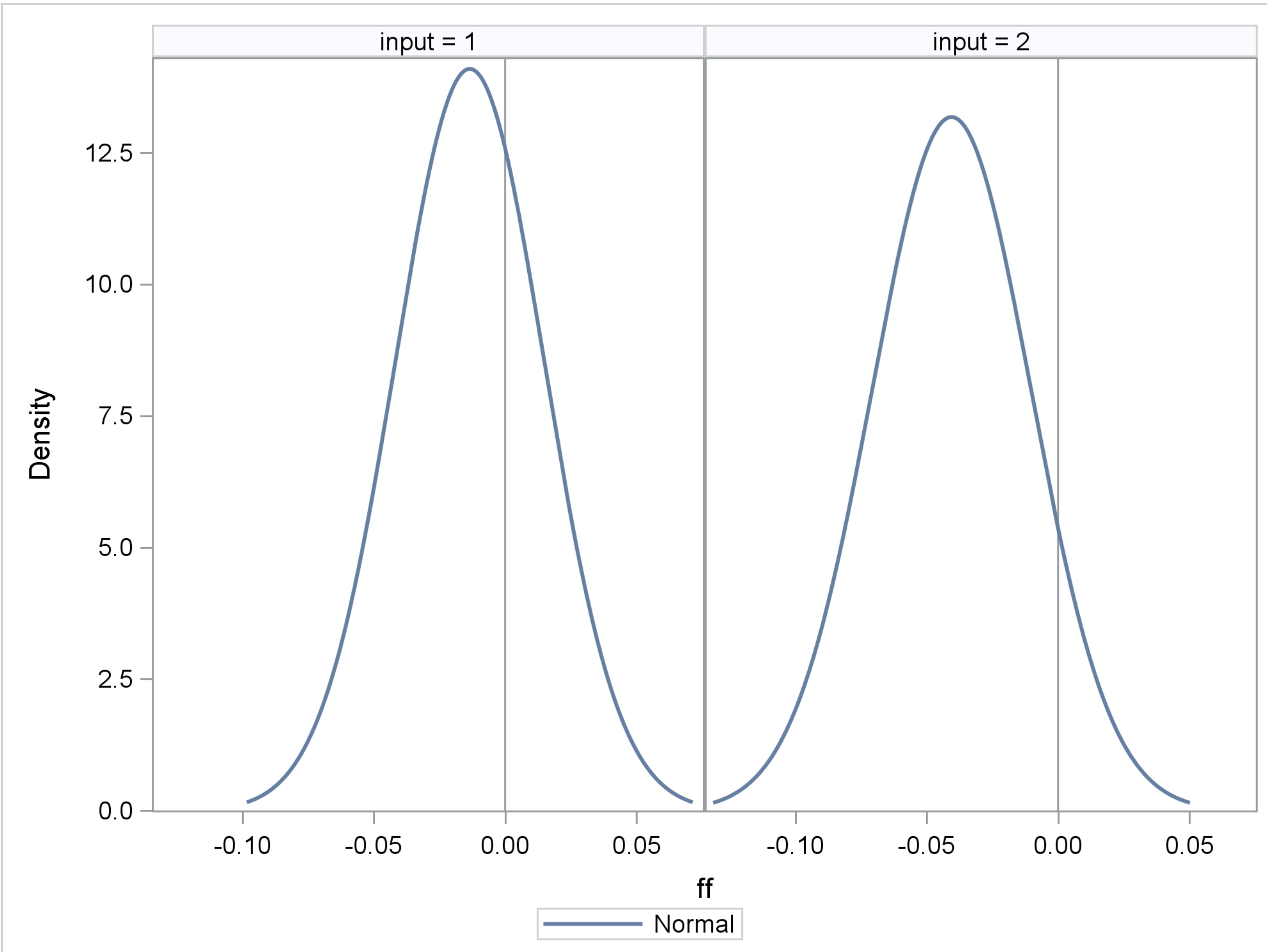
input=2

Quantiles (Definition 5)	
Level	**Quantile**
100% Max	0.06101208
99%	0.02693796
95%	0.01008202
90%	-0.00111776
75% Q3	-0.01847726
50% Median	-0.04169199
25% Q1	-0.06187039
10%	-0.07798499
5%	-0.08606522
1%	-0.11026564
0% Min	-0.12231183

input=2

Extreme Observations			
Lowest		**Highest**	
Value	**Obs**	**Value**	**Obs**
-0.122312	937	0.0272906	688
-0.119884	980	0.0291769	652
-0.113512	920	0.0388217	670
-0.112345	523	0.0477261	845
-0.110497	897	0.0610121	632

Figure 107.5 Frequency Plot for ff for Each Set of Input Values



References

Christakos, G. (1992). *Random Field Models in Earth Sciences*. New York: Academic Press.

Deutsch, C. V., and Journel, A. G. (1992). *GSLIB: Geostatistical Software Library and User's Guide*. New York: Oxford University Press.

Knuth, D. E. (1973). *Seminumerical Algorithms.* Vol. 2 of The Art of Computer Programming. Reading, MA: Addison-Wesley.

Ralston, A., and Rabinowitz, P. (1978). *A First Course in Numerical Analysis*. New York: McGraw-Hill.

Searle, S. R. (1971). *Linear Models*. New York: John Wiley & Sons.

Subject Index

SIMNORMAL procedure
- conditional simulation, 8919
- Gaussian random variables, 8913
- introductory example, 8913
- LU decomposition method, 8918
- normal random variables, 8913
- simulation, 8913
- unconditional simulation, 8918

Syntax Index

BY statement
SIMNORMAL procedure, 8917

CONDITION statement
CONDITION statement (SIMNORMAL), 8917

DATA= option
PROC SIMNORMAL statement, 8916

NUMREAL= option
PROC SIMNORMAL statement, 8916

OUT= option
PROC SIMNORMAL statement, 8916
OUTCOND option
PROC SIMNORMAL statement, 8916
OUTSEED option
PROC SIMNORMAL statement, 8916

PROC SIMNORM statement
SIMNORMAL procedure, 8915

SEED= option
PROC SIMNORMAL statement, 8916
SEEDBY option
PROC SIMNORMAL statement, 8916
SIMNORMAL procedure
syntax, 8915
SIMNORMAL procedure, BY statement, 8917
SINGULAR1= option
PROC SIMNORMAL statement, 8916
SINGULAR2= option
PROC SIMNORMAL statement, 8916

VAR statement
VAR statement (SIMNORMAL), 8917