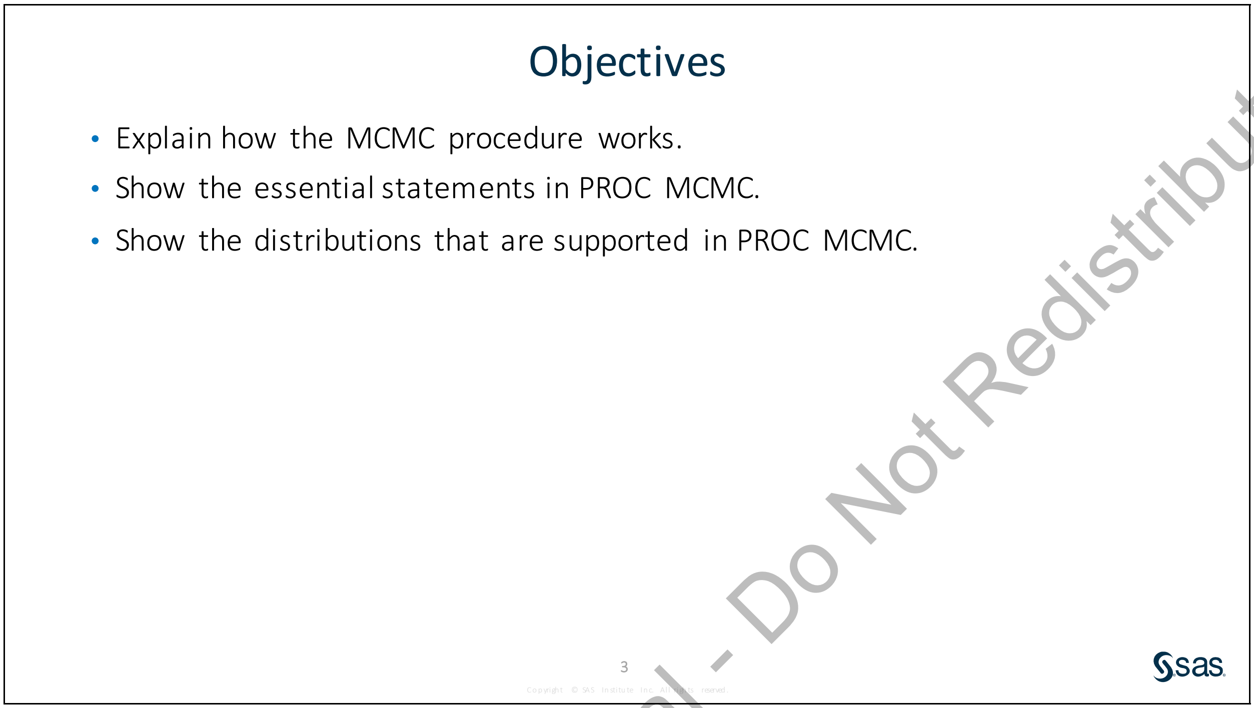
Chapter 2 Fitting Models with the MCMC Procedure

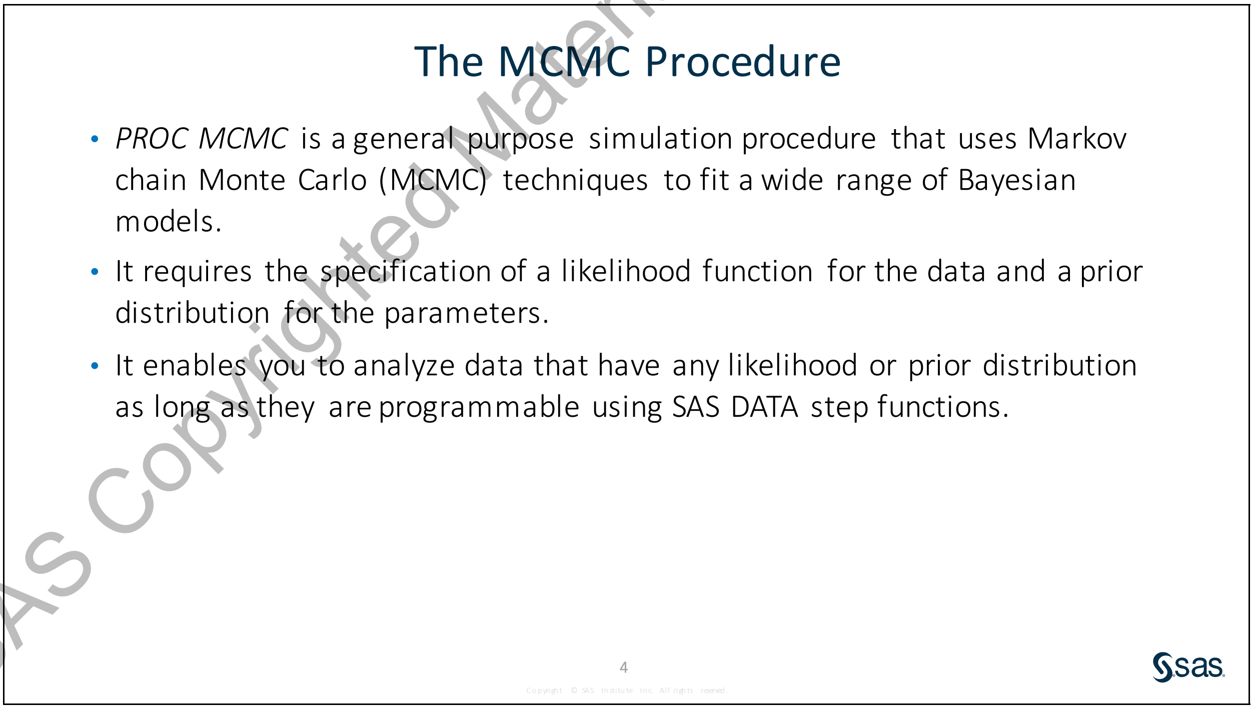
2.1 Introduction to the MCMC Procedure



Objectives

- Explain how the MCMC procedure works.
- Show the essential statements in PROC MCMC.
- Show the distributions that are supported in PROC MCMC.

3

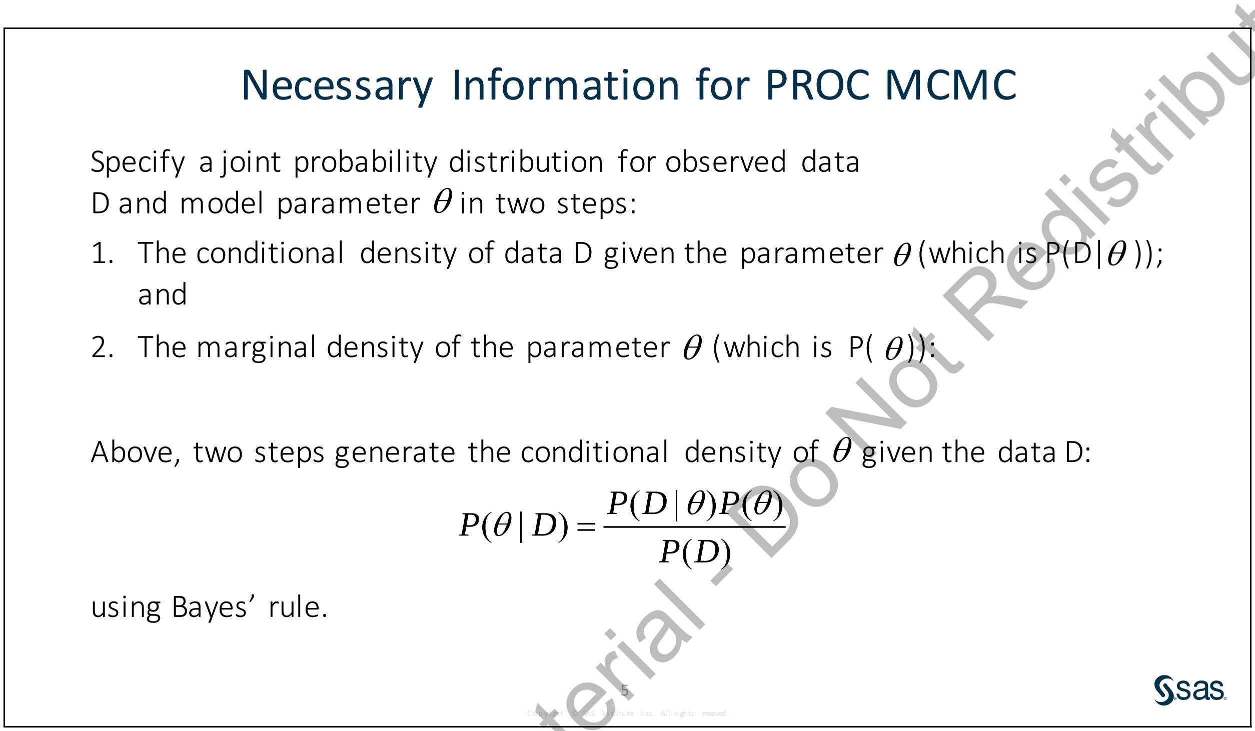


The MCMC Procedure

- *PROC MCMC* is a general purpose simulation procedure that uses Markov chain Monte Carlo (MCMC) techniques to fit a wide range of Bayesian models.
- It requires the specification of a likelihood function for the data and a prior distribution for the parameters.
- It enables you to analyze data that have any likelihood or prior distribution as long as they are programmable using SAS DATA step functions.

4

PROC MCMC is a flexible simulation-based procedure that derives inferences from simulation rather than from analytical or numerical integration methods. To use the procedure, you need to specify a likelihood function for the data and a prior distribution for the parameters. PROC MCMC then obtains samples from the corresponding posterior distributions, produces summary and diagnostic statistics, and, when requested, saves the posterior samples in an output data set that can be used for further inferential analysis.



Necessary Information for PROC MCMC

Specify a joint probability distribution for observed data D and model parameter θ in two steps:

1. The conditional density of data D given the parameter θ (which is P(D|θ)); and
2. The marginal density of the parameter θ (which is P(θ)).

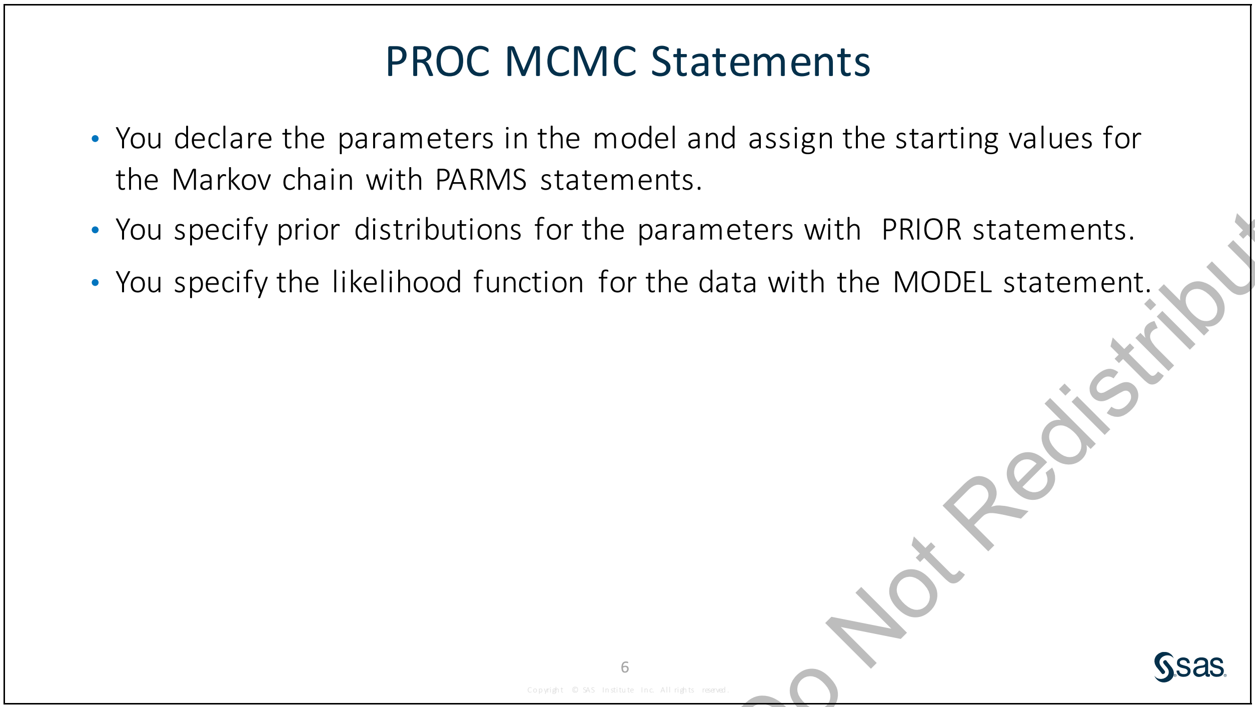
Above, two steps generate the conditional density of θ given the data D:

$$P(\theta \mid D) = \frac{P(D \mid \theta)P(\theta)}{P(D)}$$

using Bayes' rule.

5

The likelihood function of θ is proportional to $P(D \mid \theta)$.

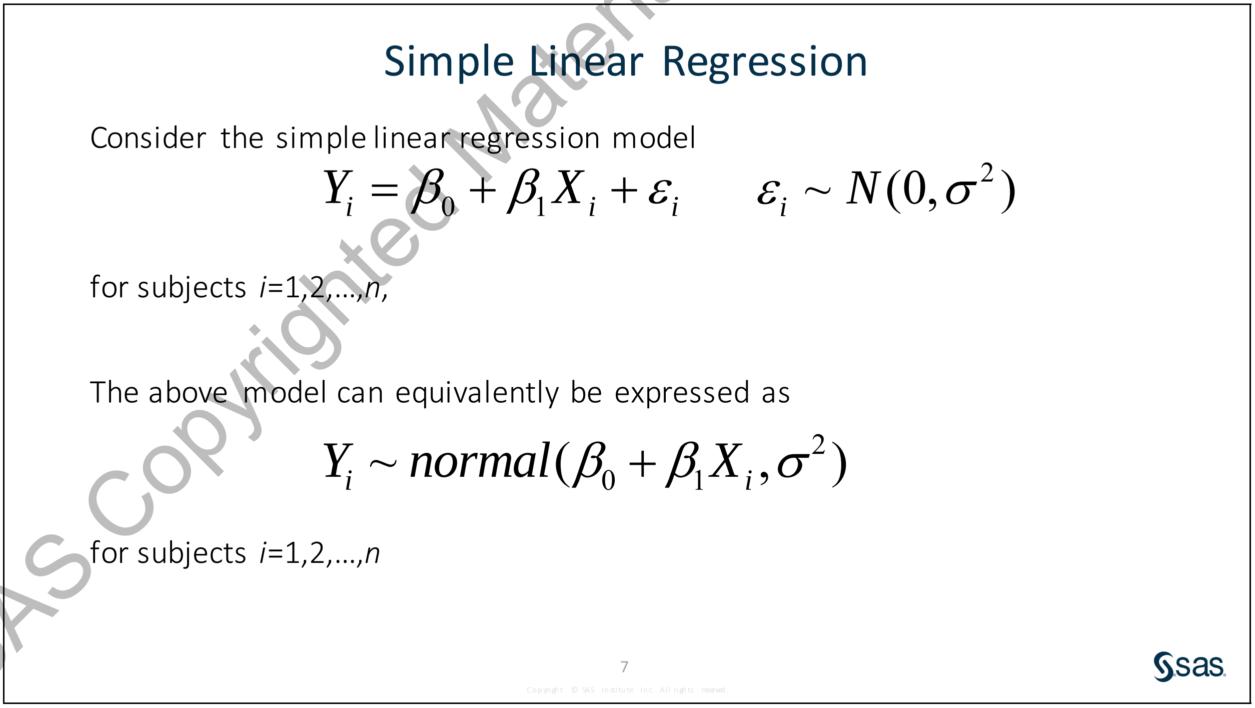


PROC MCMC Statements

- You declare the parameters in the model and assign the starting values for the Markov chain with PARMS statements.
- You specify prior distributions for the parameters with PRIOR statements.
- You specify the likelihood function for the data with the MODEL statement.

6

The statements in PROC MCMC are in many ways like DATA step statements; PROC MCMC evaluates every statement in order for each observation. The model specification is similar to PROC NLIN and shares much of the same syntax as PROC NLMIXED.



Simple Linear Regression

Consider the simple linear regression model

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i \qquad \varepsilon_i \sim N(0,\sigma^2)$$

for subjects *i*=1,2,…,*n*,

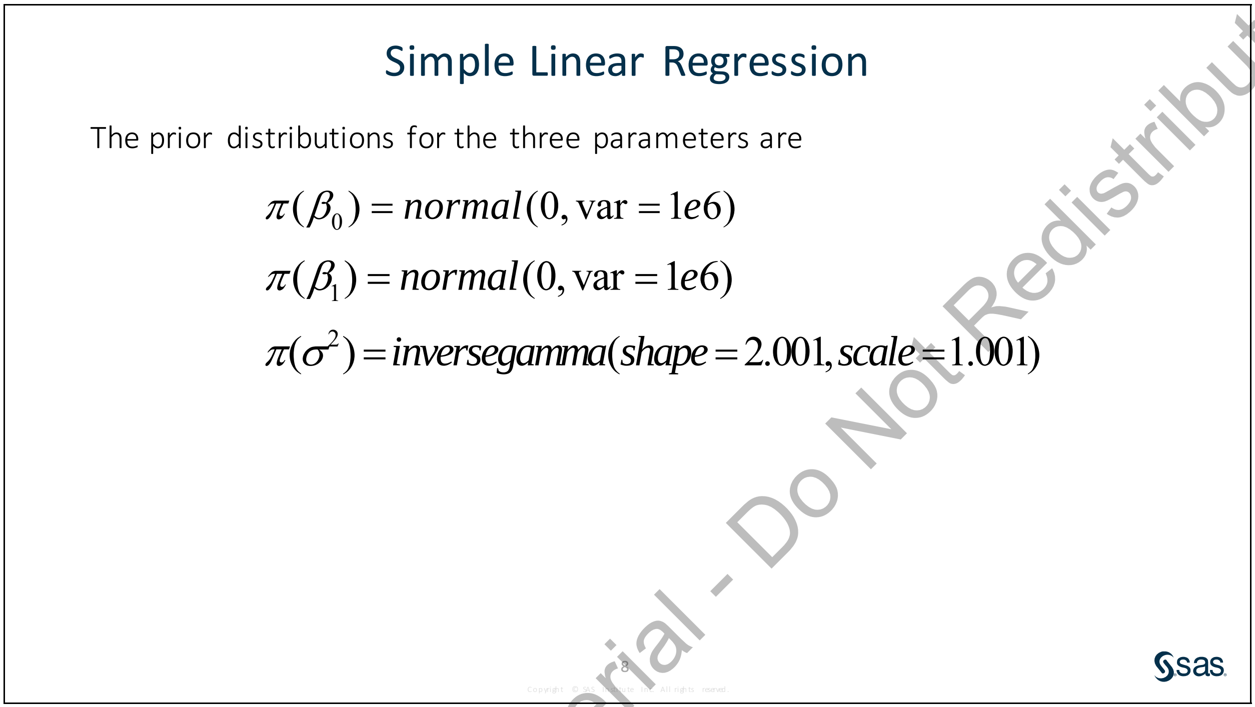
The above model can equivalently be expressed as

$$Y_i \sim normal(\beta_0 + \beta_1 X_i, \sigma^2)$$

for subjects *i*=1,2,…,*n*

7

To illustrate the functionality of PROC MCMC, start with a simple linear regression model. The observation errors are assumed to be independent and normally distributed with a mean of 0 and a constant variance, which is the same as assuming that observations are assumed independent and normally distributed with mean $\beta_0+\beta_1X_i$ and variance σ^2.



Simple Linear Regression

The prior distributions for the three parameters are

$$\pi(\beta_0) = normal(0, \mathrm{var} = 1e6)$$

$$\pi(\beta_1) = normal(0, \mathrm{var} = 1e6)$$

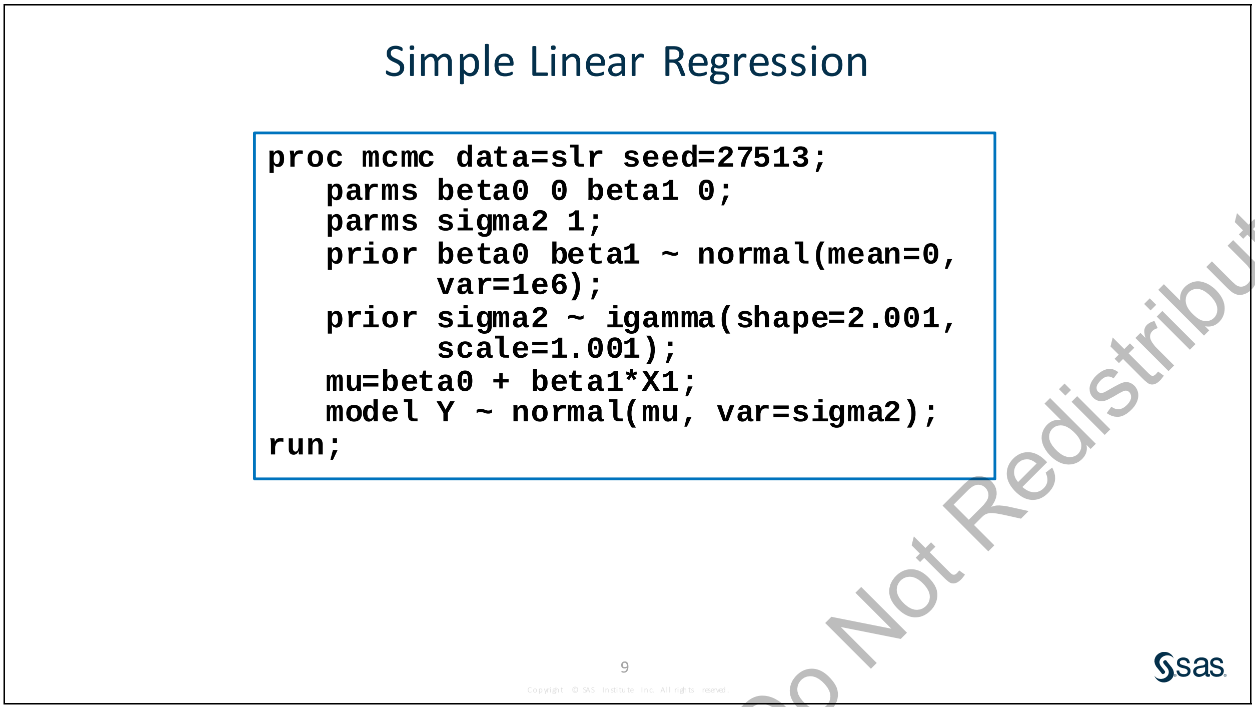
$$\pi(\sigma^2) = inversegamma(shape = 2.001, scale = 1.001)$$

8

Copyright © SAS Institute Inc. All rights reserved.

The normal priors on β_0 and β_1 have large variances, expressing your lack of knowledge about the regression coefficients. These priors have equal-tail 95% credible intervals of approximately (-2000, 2000). A frequently used diffuse prior for the variance parameter is the inverse gamma distribution. The mean of the inverse gamma distribution is $\frac{b}{a-1}$ if a>1 (*a* is the shape and *b* is the scale). The variance of the inverse gamma distribution is $\frac{b^2}{(a-1)^2(a-2)}$. Setting the shape to 2.001 and the scale to 1.001 gives a mean of 1 and a variance of 1000, which corresponds to conjugate noninformative priors.

Note: The formulas for the gamma distribution were obtained from the SAS/STAT 14.2 documentation (SAS Institute, Inc. 2016).



Simple Linear Regression

```
proc mcmc data=slr seed=27513;
   parms beta0 0 beta1 0;
   parms sigma2 1;
   prior beta0 beta1 ~ normal(mean=0,
         var=1e6);
   prior sigma2 ~ igamma(shape=2.001,
         scale=1.001);
   mu=beta0 + beta1*X1;
   model Y ~ normal(mu, var=sigma2);
run;
```

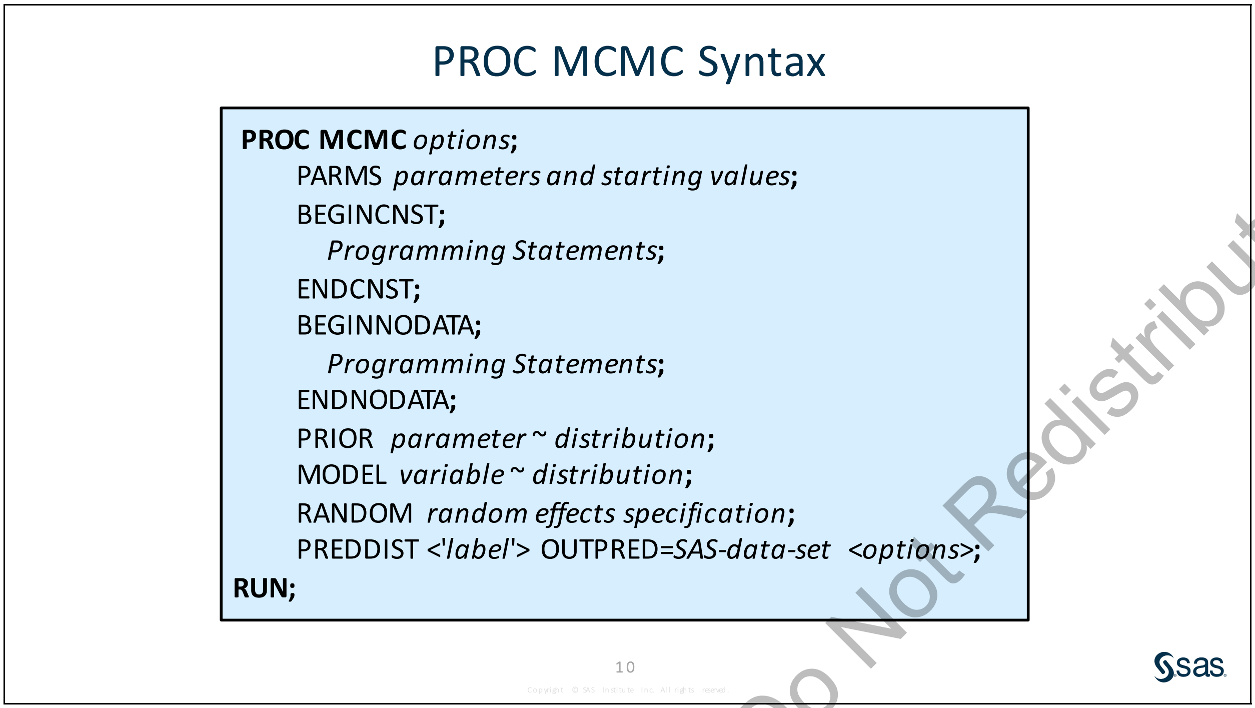
The above program fits the simple linear regression model with the noninformative prior information. The PROC MCMC statement invokes the procedure and specifies the input data set **slr**. The SEED= option specifies a seed for the random number generator, which enables the reproducibility of the random stream.

The PARMS statements identify the three parameters in the model: **beta0**, **beta1**, and **sigma2**. These statements also assign initial values to the parameters. The regression coefficients are given 0 as their initial values, and the scale parameter **sigma2** starts at value 1. If you do not provide initial values, the procedure chooses starting values for every parameter using the prior distribution.

The PRIOR statements specify prior distributions for the parameters. The parameters **beta0** and **beta1** both share the same prior—a normal prior with mean 0 and variance 1e6. The parameter **sigma2** has an inverse gamma distribution with a shape parameter of 2.001 and a scale parameter of 1.001.

The **mu** assignment statement calculates the conditional expected value of *Y* as a linear function of *X* for a given value x. The MODEL statement specifies the log-likelihood functions for the response variable. It uses the normal density function to indicate that the response variable, *Y*, is normally distributed with parameters **mu** and **sigma2**. The functional argument MEAN= in the normal distribution is optional, but you have to indicate whether **sigma2** is a variance (VAR=), a standard deviation (SD=), or a precision (PRECISION=) parameter.

You do not need to know the form of the posterior distribution when you use PROC MCMC. PROC MCMC automatically obtains samples from the desired posterior distribution, which is determined by the prior and the likelihood that you specify.

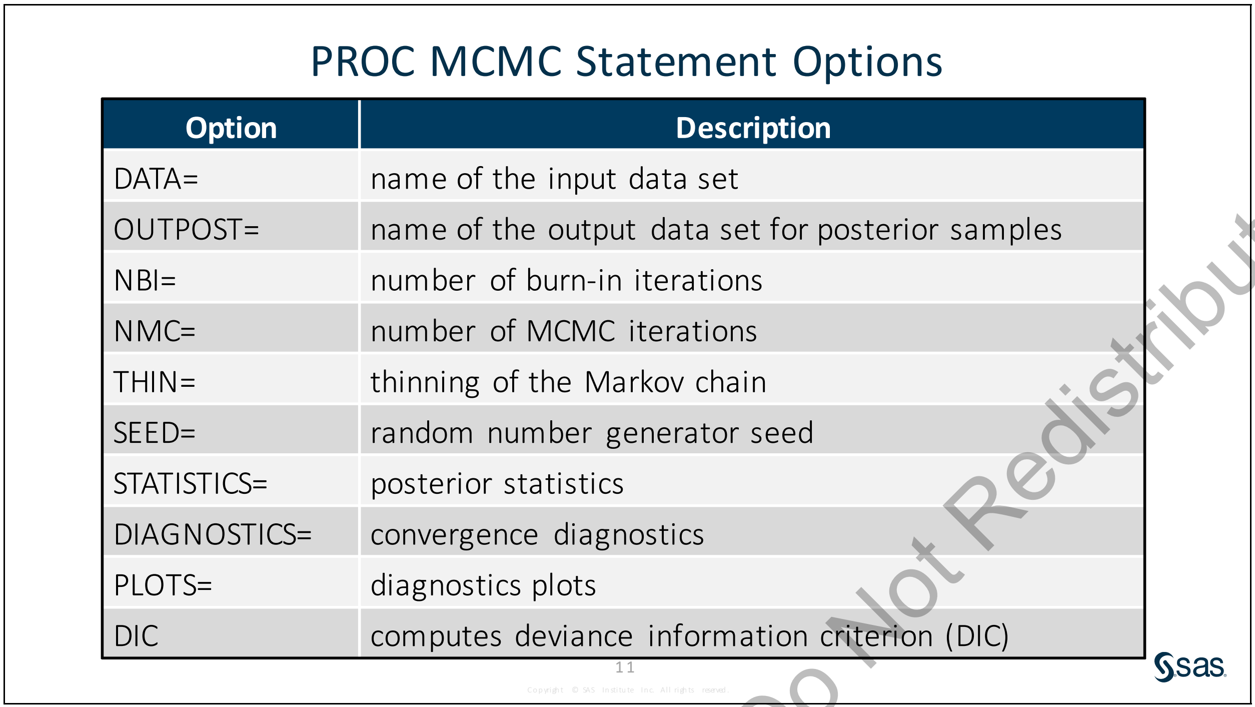


PROC MCMC Syntax

```
PROC MCMC options;
    PARMS  parameters and starting values;
    BEGINCNST;
      Programming Statements;
    ENDCNST;
    BEGINNODATA;
      Programming Statements;
    ENDNODATA;
    PRIOR  parameter ~ distribution;
    MODEL  variable ~ distribution;
    RANDOM  random effects specification;
    PREDDIST <'label'> OUTPRED=SAS-data-set  <options>;
RUN;
```

10

The PARMS, PRIOR, and MODEL statements form the basis of every Bayesian model. The BEGINCNST/ENDCNST and BEGINNODATA/ENDNODATA statements are used to save unnecessary evaluation and reduce simulation time. The PREDDIST statement is used to generate samples from the posterior predictive distribution. In addition, you can use the ARRAY statement to define constant or parameter arrays, programming statements to specify more complicated models, the RANDOM statement to define random effects and their prior distributions, and the UDS statement, which enables you to use a separate algorithm, other than the default random walk Metropolis, to update parameters in the model.

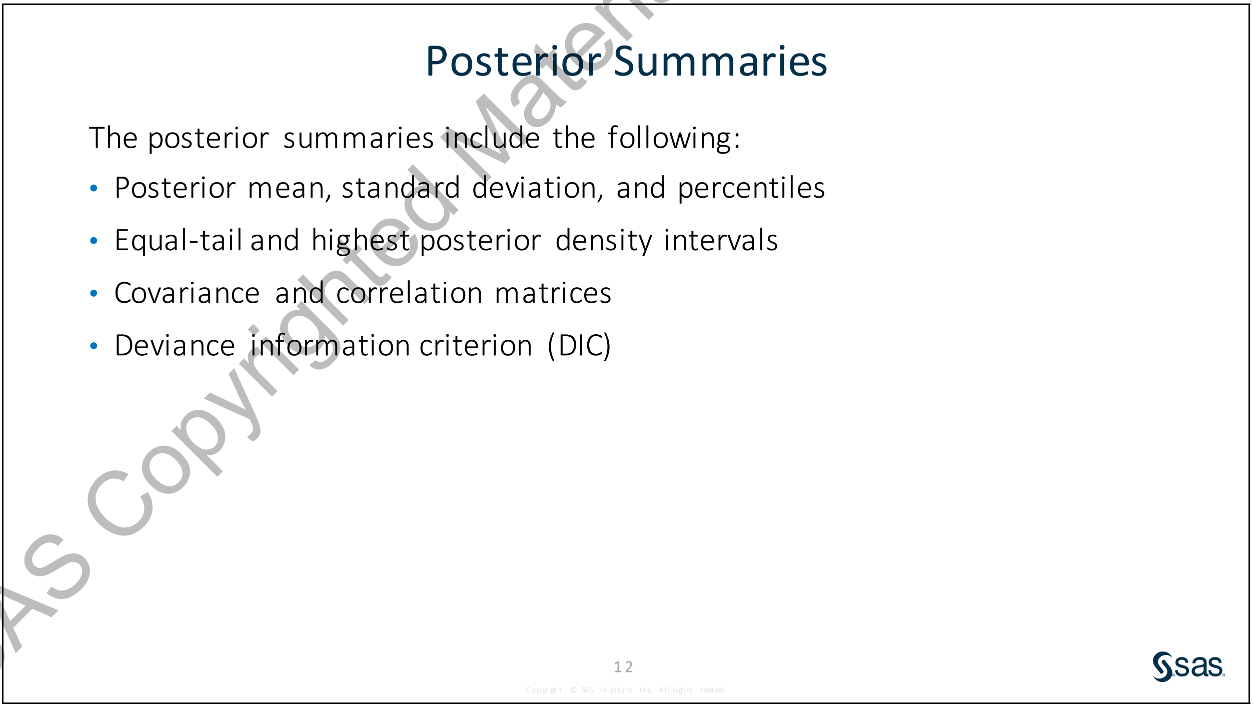


PROC MCMC Statement Options

Option	Description
DATA=	name of the input data set
OUTPOST=	name of the output data set for posterior samples
NBI=	number of burn-in iterations
NMC=	number of MCMC iterations
THIN=	thinning of the Markov chain
SEED=	random number generator seed
STATISTICS=	posterior statistics
DIAGNOSTICS=	convergence diagnostics
PLOTS=	diagnostics plots
DIC	computes deviance information criterion (DIC)

11

Some of the PROC MCMC statement options are shown above. PROC MCMC performs only posterior analyses (such as plots, diagnostics, and summaries) on the parameters selected with the MONITOR= option. By default, only the model parameters are selected for analysis.



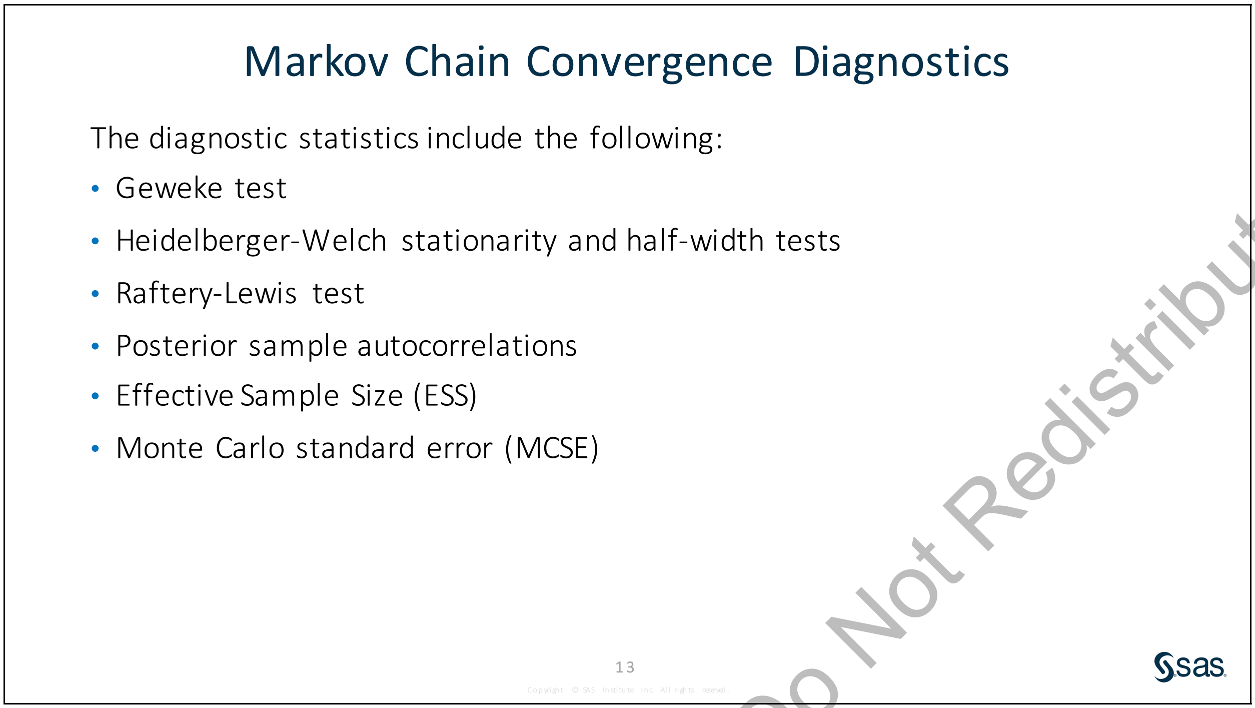
Posterior Summaries

The posterior summaries include the following:

- Posterior mean, standard deviation, and percentiles
- Equal-tail and highest posterior density intervals
- Covariance and correlation matrices
- Deviance information criterion (DIC)

12

PROC MCMC computes the posterior mean, standard deviation, quantiles, and two 95% credible intervals: equal-tail and highest posterior density (HPD). Other available summaries include the posterior correlation and covariance.



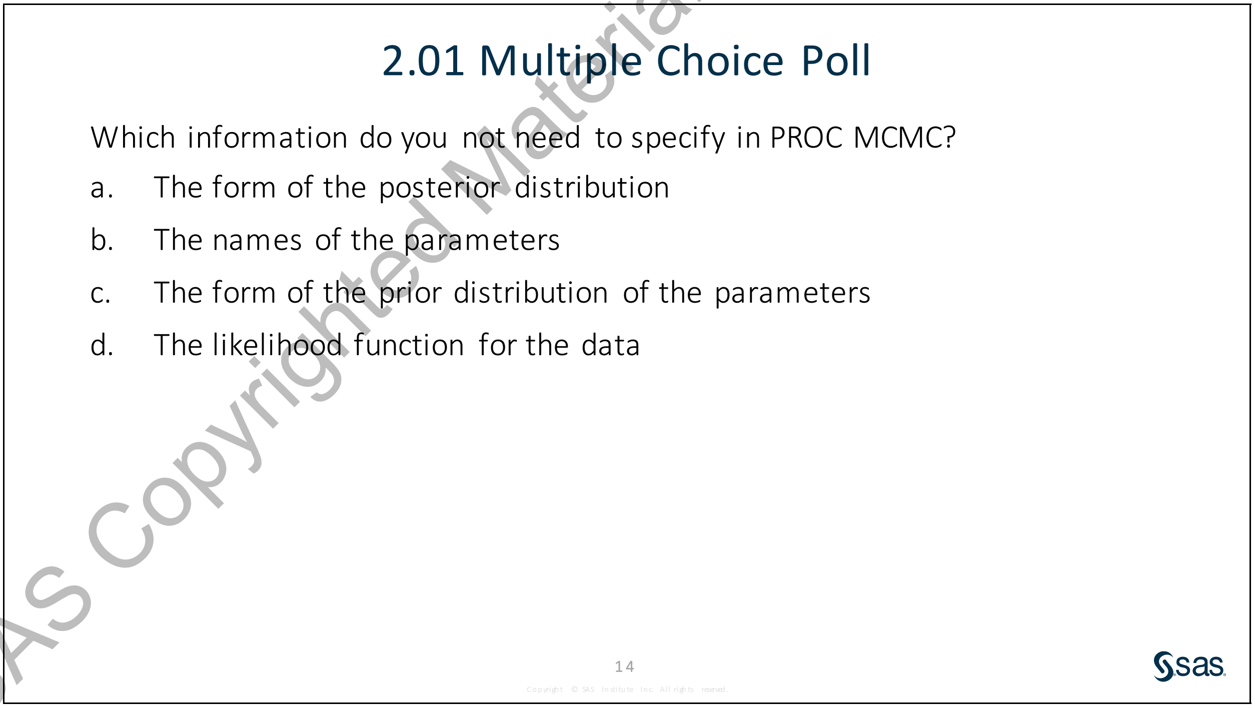
Markov Chain Convergence Diagnostics

The diagnostic statistics include the following:

- Geweke test
- Heidelberger-Welch stationarity and half-width tests
- Raftery-Lewis test
- Posterior sample autocorrelations
- Effective Sample Size (ESS)
- Monte Carlo standard error (MCSE)

13

PROC MCMC computes the Geweke test, sample autocorrelations, effective sample sizes, and Monte Carlo standard errors.



2.01 Multiple Choice Poll

Which information do you not need to specify in PROC MCMC?

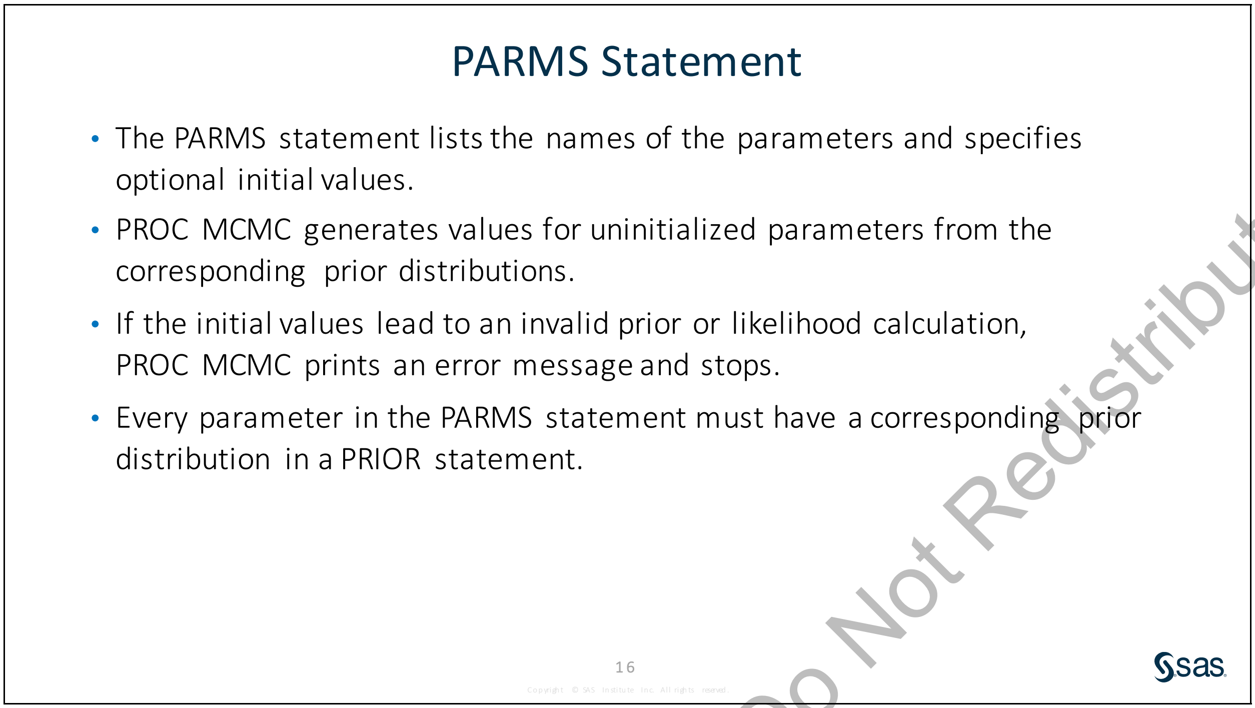
a. The form of the posterior distribution

b. The names of the parameters

c. The form of the prior distribution of the parameters

d. The likelihood function for the data

14

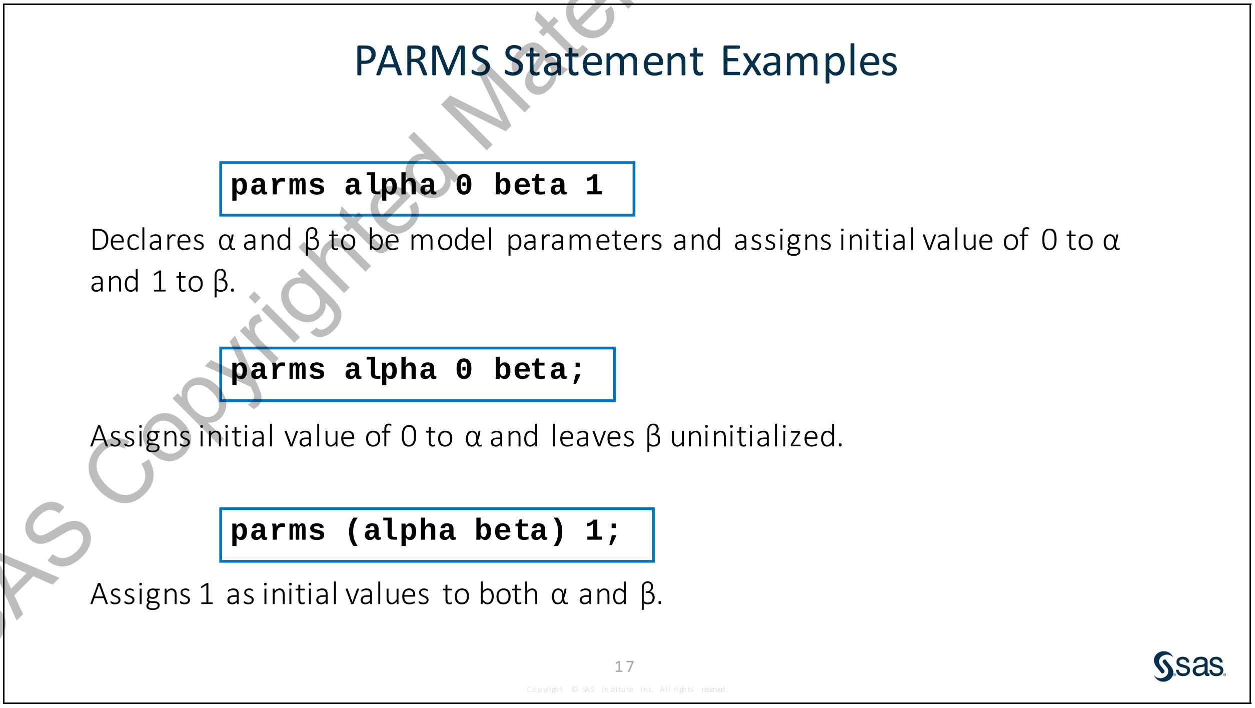


PARMS Statement

- The PARMS statement lists the names of the parameters and specifies optional initial values.
- PROC MCMC generates values for uninitialized parameters from the corresponding prior distributions.
- If the initial values lead to an invalid prior or likelihood calculation, PROC MCMC prints an error message and stops.
- Every parameter in the PARMS statement must have a corresponding prior distribution in a PRIOR statement.

16

Each PARMS statement defines a block of parameters, and the blocked Metropolis algorithm updates the parameters in each block simultaneously. The program exits if every parameter specified in the PARMS statement does not have a corresponding prior distribution specified in the PRIOR statement.



PARMS Statement Examples

```
parms alpha 0 beta 1
```

Declares α and β to be model parameters and assigns initial value of 0 to α and 1 to β.

```
parms alpha 0 beta;
```

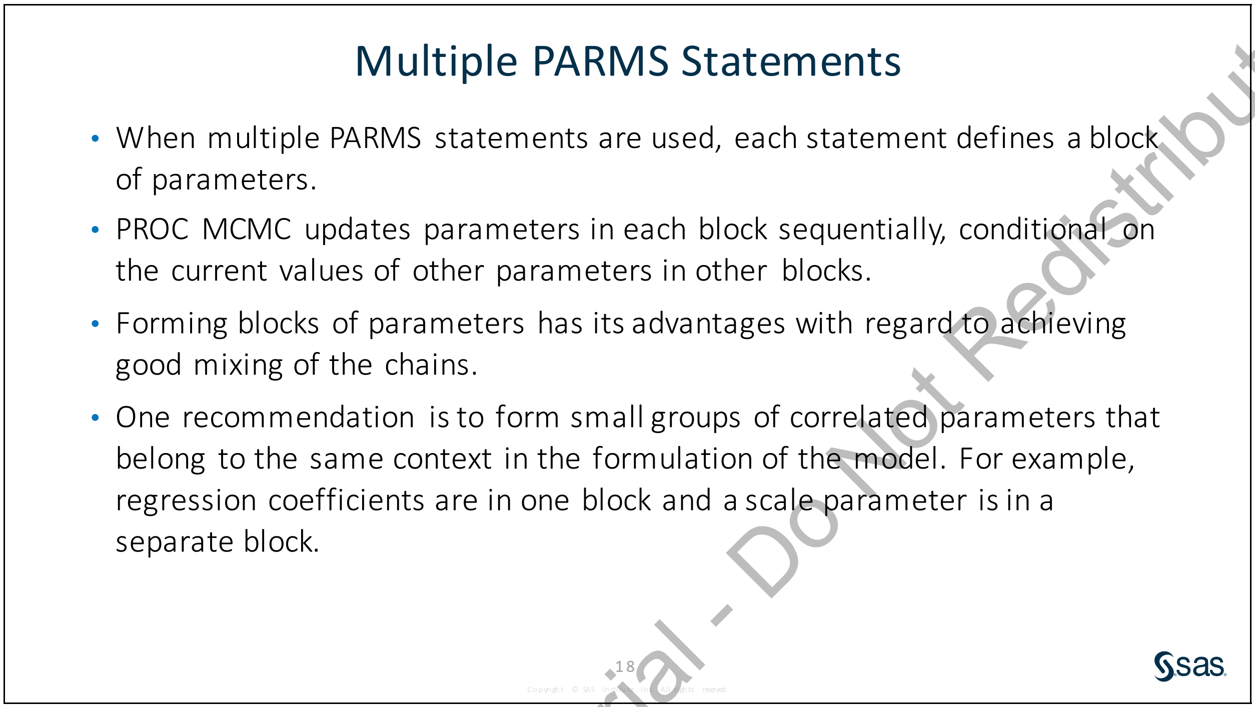
Assigns initial value of 0 to α and leaves β uninitialized.

```
parms (alpha beta) 1;
```

Assigns 1 as initial values to both α and β.

17

Multiple PARMS statements are allowed. With PROC MCMC, you can sample all parameters simultaneously by putting them all in a single PARMS statement, you can sample parameters individually by putting each parameter in its own PARMS statement, or you can sample certain subsets of parameters together by grouping each subset in its own PARMS statements.

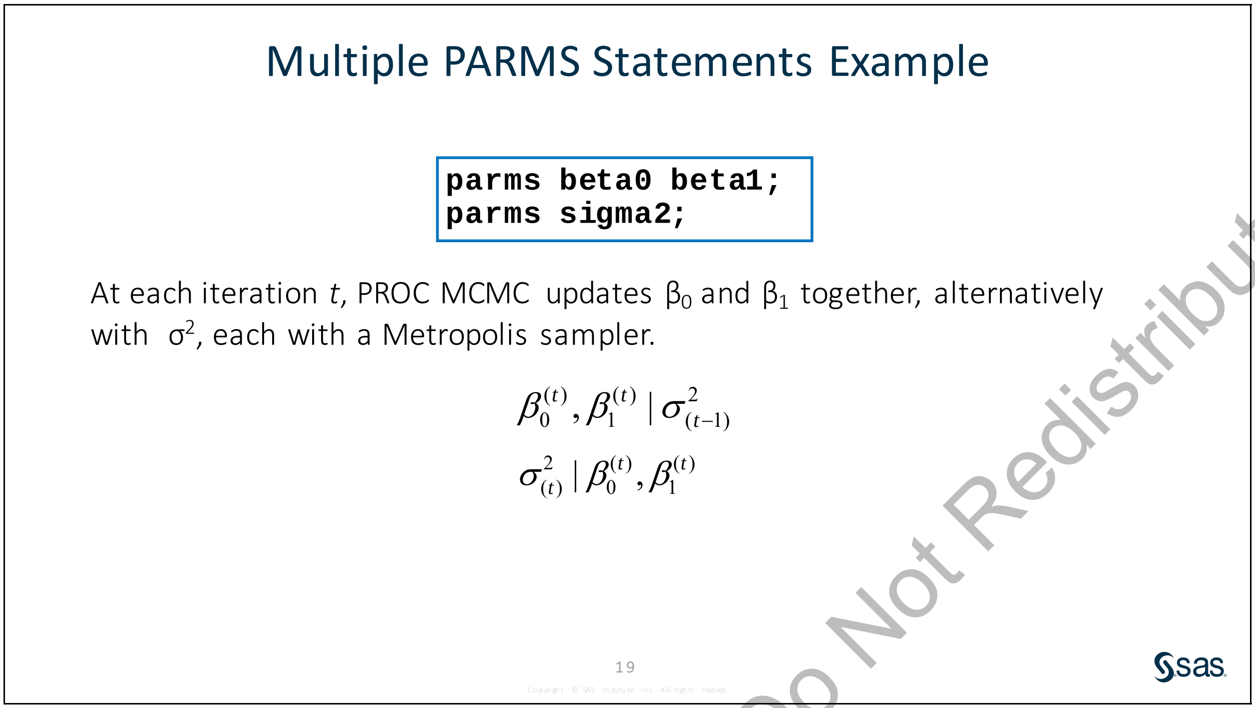


Multiple PARMS Statements

- When multiple PARMS statements are used, each statement defines a block of parameters.
- PROC MCMC updates parameters in each block sequentially, conditional on the current values of other parameters in other blocks.
- Forming blocks of parameters has its advantages with regard to achieving good mixing of the chains.
- One recommendation is to form small groups of correlated parameters that belong to the same context in the formulation of the model. For example, regression coefficients are in one block and a scale parameter is in a separate block.

18

When simultaneously sampling a large number of parameters, the algorithm might find it difficult to achieve good mixing. As the number of parameters gets large, it is much more likely to have (proposal) samples that fall well into the tails of the target distribution, producing too small a test ratio. As a result, few proposed values are accepted and convergence is slow. On the other hand, when sampling each parameter individually, the chain might mix far too slowly because the conditional distributions might be very "narrow." Hence, it takes a long time for the chain to explore fully that dimension alone. The best mixing is usually obtained with a blocking strategy somewhere between the all-at-once and one-at-a-time strategies.



Multiple PARMS Statements Example

```
parms beta0 beta1;
parms sigma2;
```

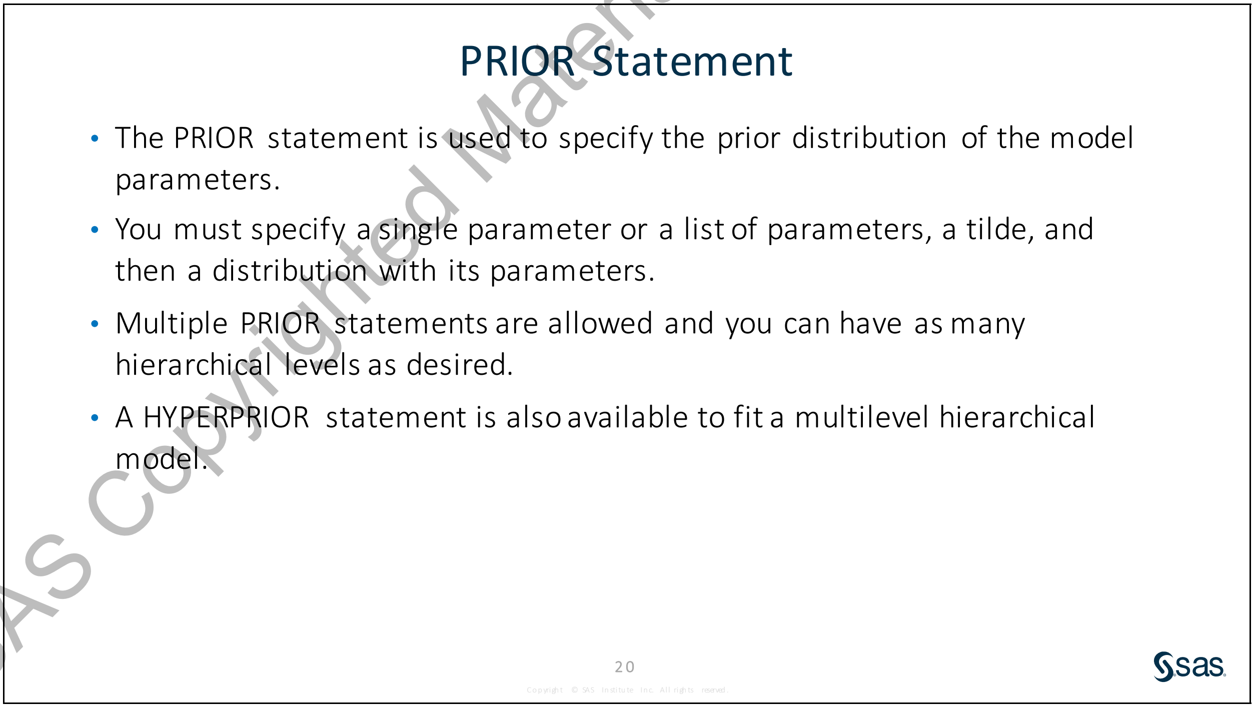
At each iteration *t*, PROC MCMC updates β_0 and β_1 together, alternatively with σ^2, each with a Metropolis sampler.

$$\beta_0^{(t)}, \beta_1^{(t)} \mid \sigma^2_{(t-1)}$$

$$\sigma^2_{(t)} \mid \beta_0^{(t)}, \beta_1^{(t)}$$

19

In the simple linear regression example, you expect the parameters **beta0** and **beta1** to have high posterior correlations, and placing them both in the same block improves the mixing of the chain. This will improve the efficiency that the posterior parameter space is explored by the Markov chain.

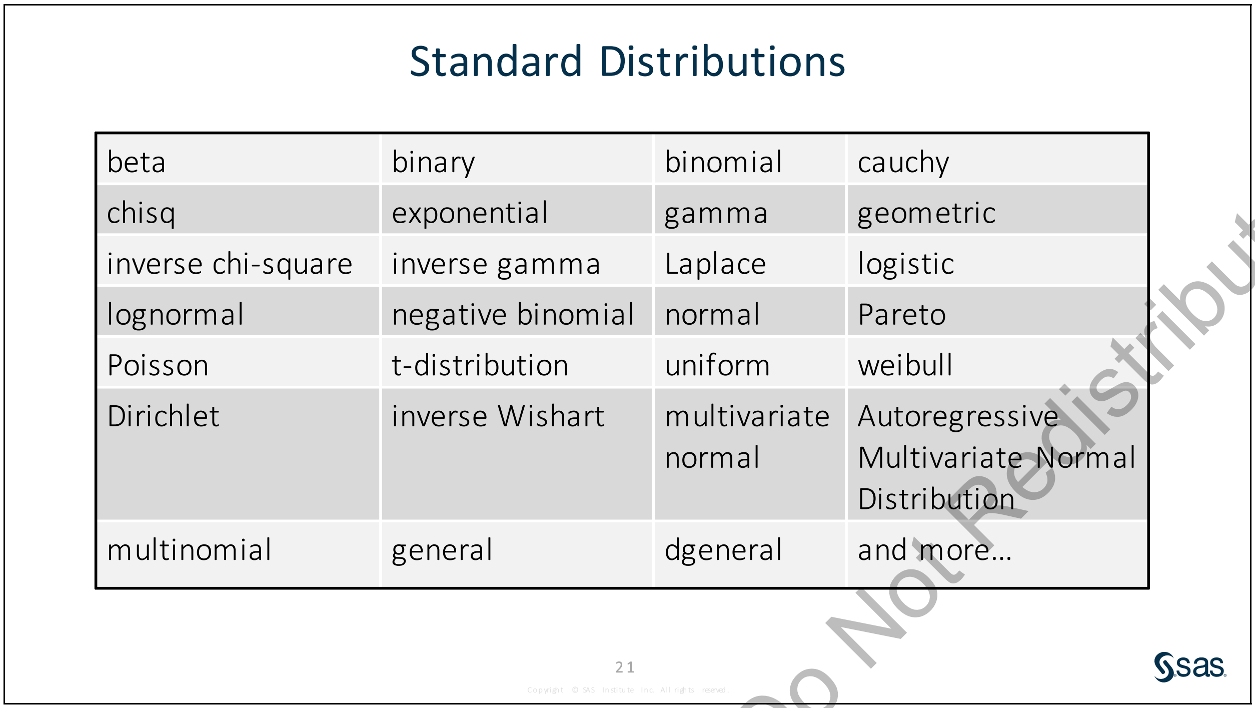


PRIOR Statement

- The PRIOR statement is used to specify the prior distribution of the model parameters.
- You must specify a single parameter or a list of parameters, a tilde, and then a distribution with its parameters.
- Multiple PRIOR statements are allowed and you can have as many hierarchical levels as desired.
- A HYPERPRIOR statement is also available to fit a multilevel hierarchical model.

20

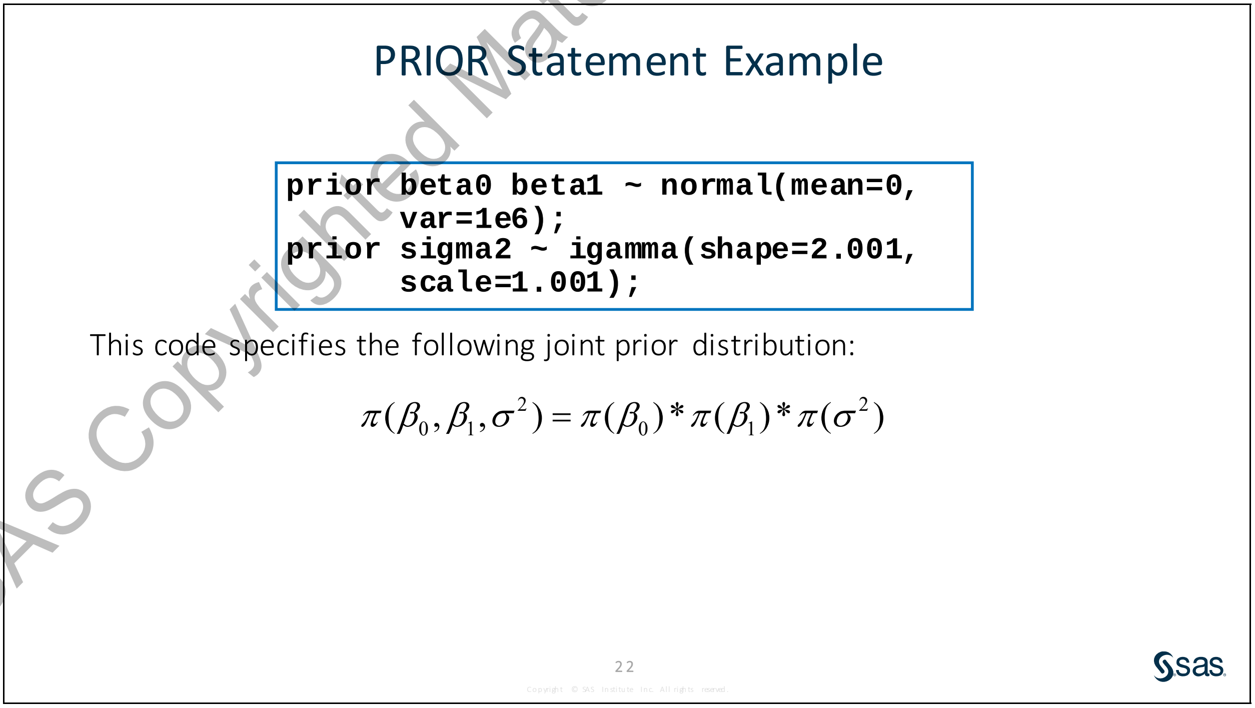
The PRIOR statements are processed twice at every Markov chain simulation (twice per pass through the data set). The statements are called at the first and the last observation of the data set.



Standard Distributions

beta	binary	binomial	cauchy
chisq	exponential	gamma	geometric
inverse chi-square	inverse gamma	Laplace	logistic
lognormal	negative binomial	normal	Pareto
Poisson	t-distribution	uniform	weibull
Dirichlet	inverse Wishart	multivariate normal	Autoregressive Multivariate Normal Distribution
multinomial	general	dgeneral	and more…

Standard distributions listed in the above slide can be used only in the MODEL, PRIOR, and HYPERPRIOR statements to specify either a prior distribution or a conditional distribution of the data, given parameters. They do not return any values, and you cannot use them in the programming statements. PROC MCMC also has a number of internally defined log-density functions.



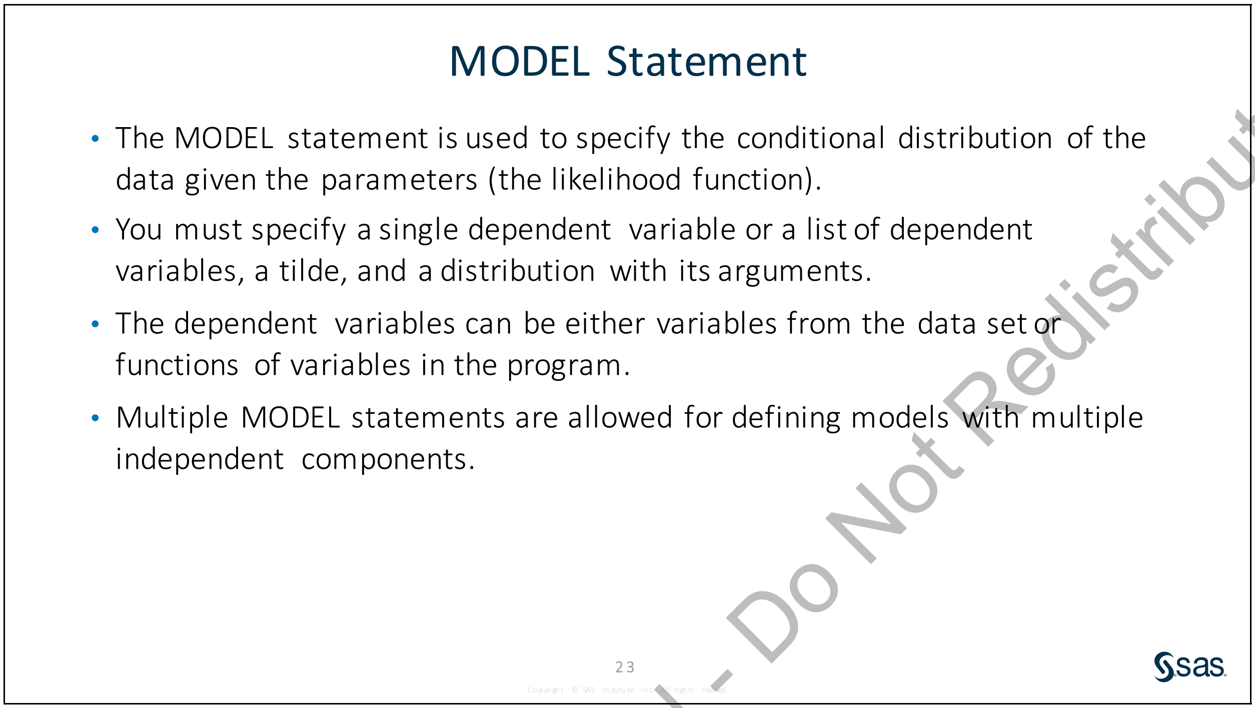
PRIOR Statement Example

```
prior beta0 beta1 ~ normal(mean=0,
          var=1e6);
prior sigma2 ~ igamma(shape=2.001,
          scale=1.001);
```

This code specifies the following joint prior distribution:

$$\pi(\beta_0,\beta_1,\sigma^2)=\pi(\beta_0)*\pi(\beta_1)*\pi(\sigma^2)$$

When constructing the posterior density, PROC MCMC takes the sum of the log prior density values from each of the PRIOR statements.

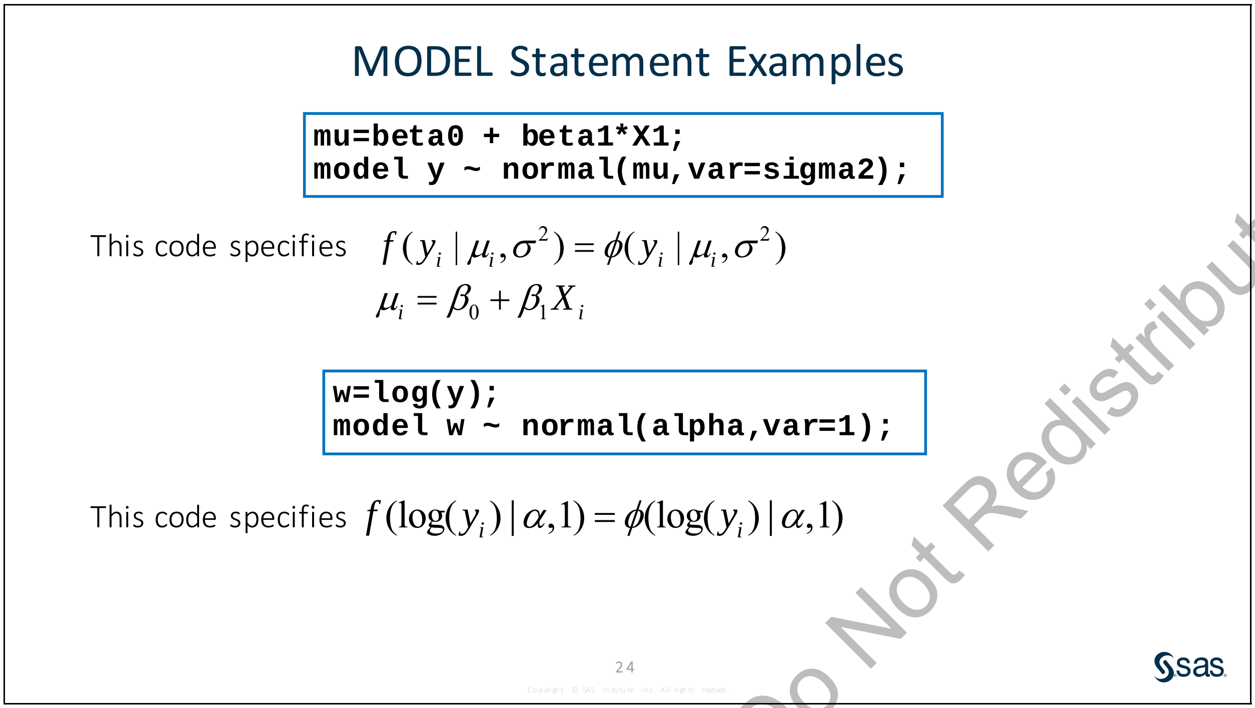


MODEL Statement

- The MODEL statement is used to specify the conditional distribution of the data given the parameters (the likelihood function).
- You must specify a single dependent variable or a list of dependent variables, a tilde, and a distribution with its arguments.
- The dependent variables can be either variables from the data set or functions of variables in the program.
- Multiple MODEL statements are allowed for defining models with multiple independent components.

23

PROC MCMC is a programming language that is similar to the DATA step, and the order of statement evaluation is important. For example, the MODEL statement must come after any SAS programming statements that define or modify arguments used in the construction of the log likelihood. In PROC MCMC, a symbol is allowed to be defined multiple times and used at different places. Using an expression out of order produces erroneous results that can also be hard to detect.



MODEL Statement Examples

```
mu=beta0 + beta1*X1;
model y ~ normal(mu,var=sigma2);
```

This code specifies $f(y_i \mid \mu_i, \sigma^2) = \phi(y_i \mid \mu_i, \sigma^2)$

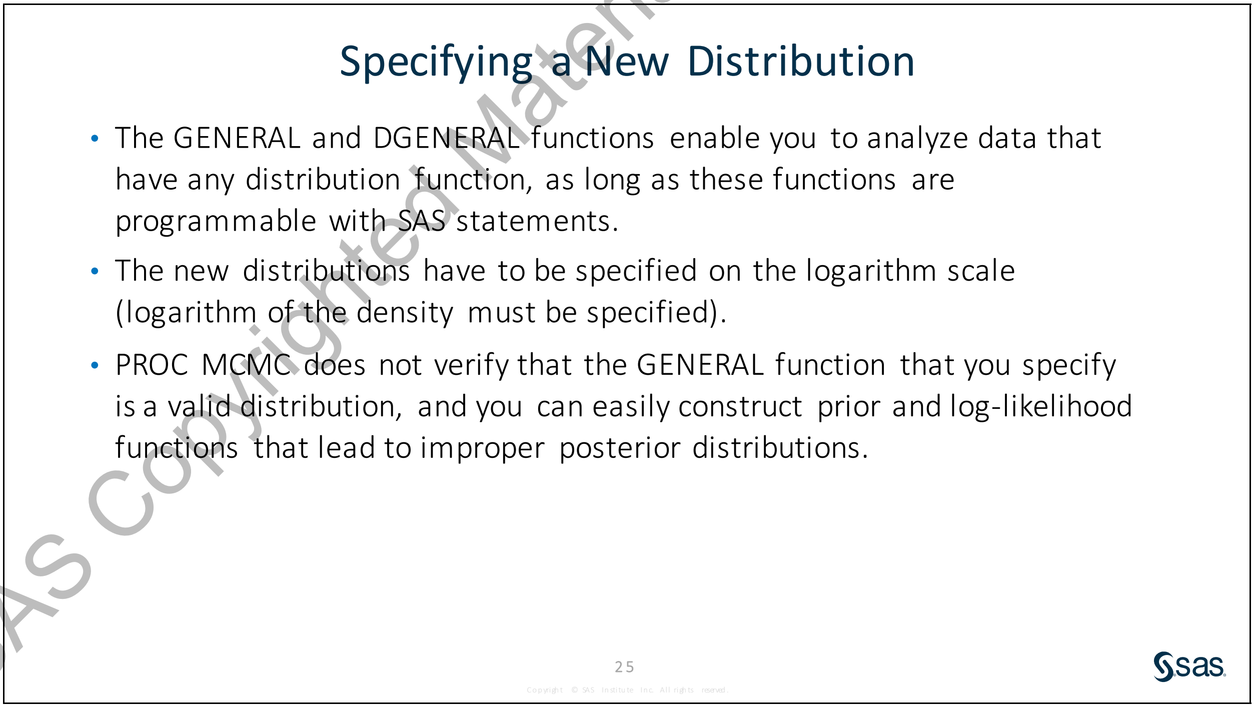
$$\mu_i = \beta_0 + \beta_1 X_i$$

```
w=log(y);
model w ~ normal(alpha,var=1);
```

This code specifies $f(\log(y_i) \mid \alpha, 1) = \phi(\log(y_i) \mid \alpha, 1)$

24

PROC MCMC allows some distributions supported in the MODEL statement to be parameterized in multiple ways. For example, you can specify a normal distribution with variance (VAR=), standard deviation (SD=), or precision (PRECISION=) parameter.



Specifying a New Distribution

- The GENERAL and DGENERAL functions enable you to analyze data that have any distribution function, as long as these functions are programmable with SAS statements.
- The new distributions have to be specified on the logarithm scale (logarithm of the density must be specified).
- PROC MCMC does not verify that the GENERAL function that you specify is a valid distribution, and you can easily construct prior and log-likelihood functions that lead to improper posterior distributions.

25

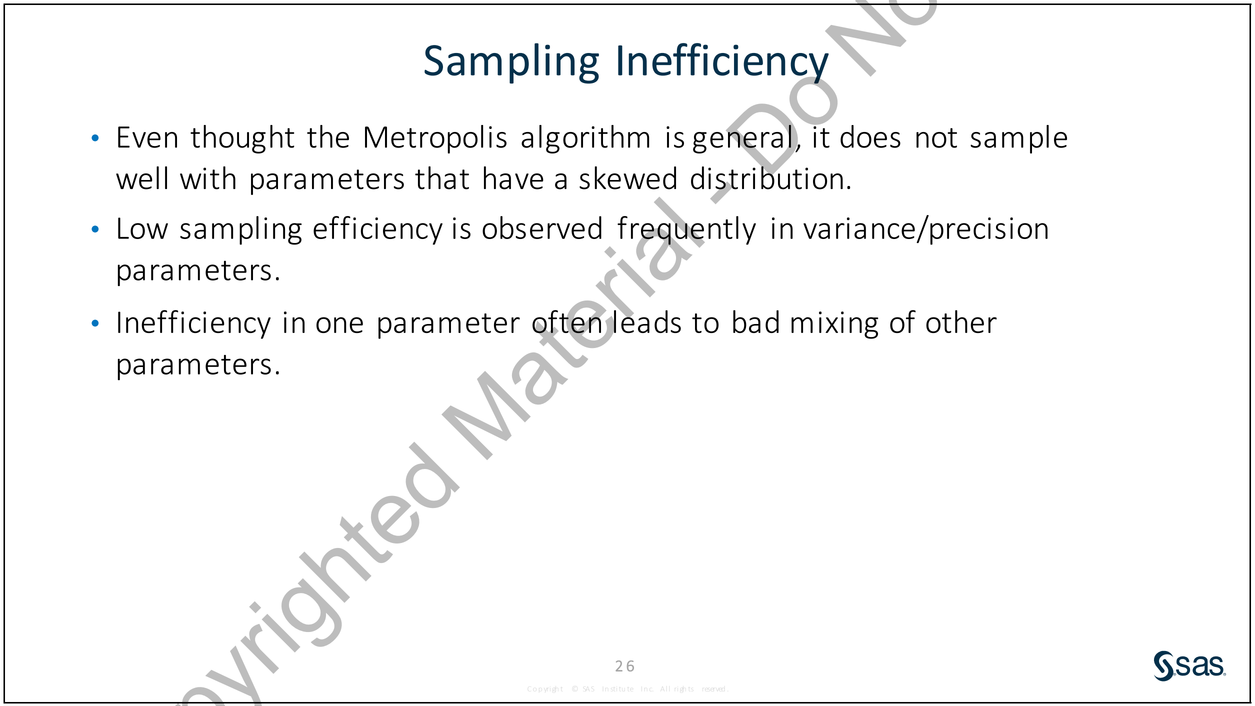
When you use the GENERAL function in the MODEL statement, you do not need to specify the dependent variable on the left of the tilde. The log-likelihood function takes the dependent variable into account. Hence, there is no need to explicitly state the dependent variable in the MODEL statement. However, in the PRIOR statements, you need to explicitly state the parameter names and a tilde with the GENERAL and DGENERAL functions. Furthermore, when the function is used in the PRIOR statements, you must supply initial values.

The new distributions have to be specified on the logarithm scale because PROC MCMC computes the logarithm of the posterior density.

$$\log(\pi(\theta \mid y)) = \log(\pi(\theta)) + \sum_{i=1}^{n} \log(f(y_i \mid \theta)) + \text{constant}$$

At each Markov chain iteration, the procedure computes the log of the posterior density by stepping through the input data set, performing the computations for each response variable value, and cumulatively adding the log-likelihood values. At the last observation, the log of the prior density is added to the sum of the log likelihood to obtain the log of the posterior density up to an additive constant.

Note: The DGENERAL function is for discrete distributions.



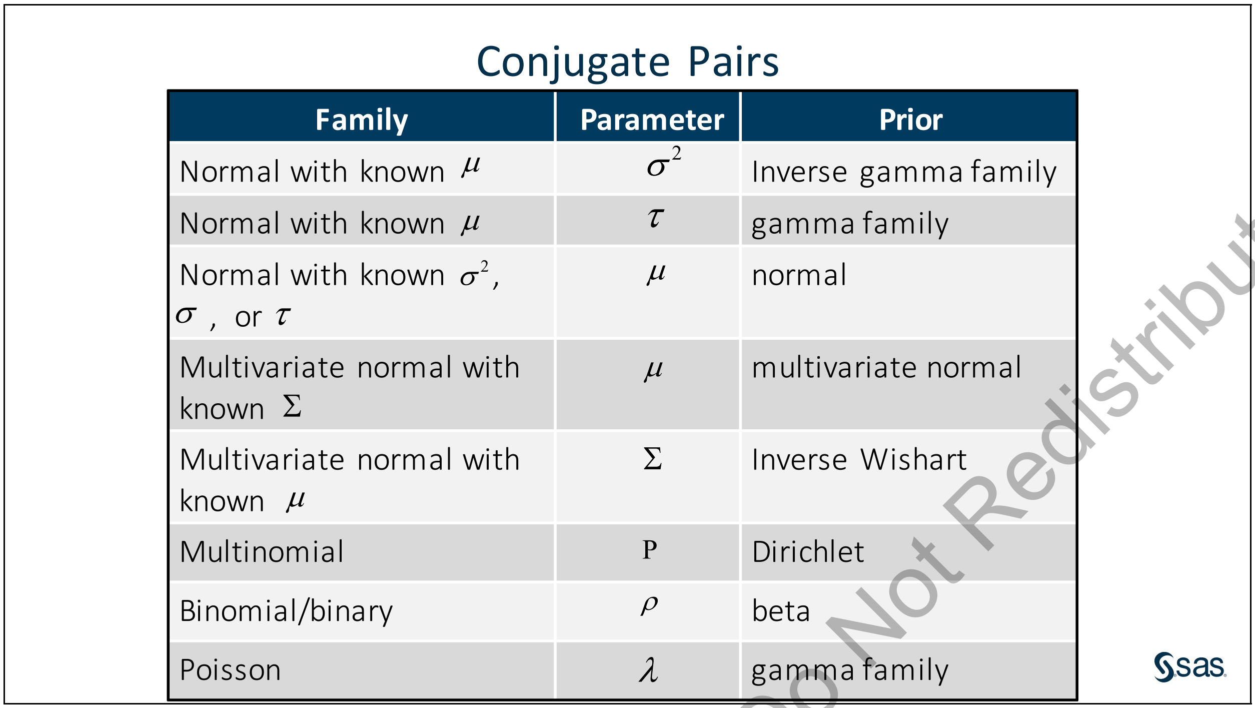
Sampling Inefficiency

- Even thought the Metropolis algorithm is general, it does not sample well with parameters that have a skewed distribution.
- Low sampling efficiency is observed frequently in variance/precision parameters.
- Inefficiency in one parameter often leads to bad mixing of other parameters.

26

Copyright © SAS Institute Inc. All rights reserved.

PROC MCMC automatically obtains samples from the desired posterior distribution, which is determined by the prior (through the PRIOR statement) and the likelihood (through the MODEL statement) you specify. The main sampling mechanism is a self-tuned random walk Metropolis algorithm. However, this sampling algorithm does not sample well with parameters that have skewed distributions such as the variance parameters. One alternative is to sample on a transformed parameter space, which works well in some but not all cases.



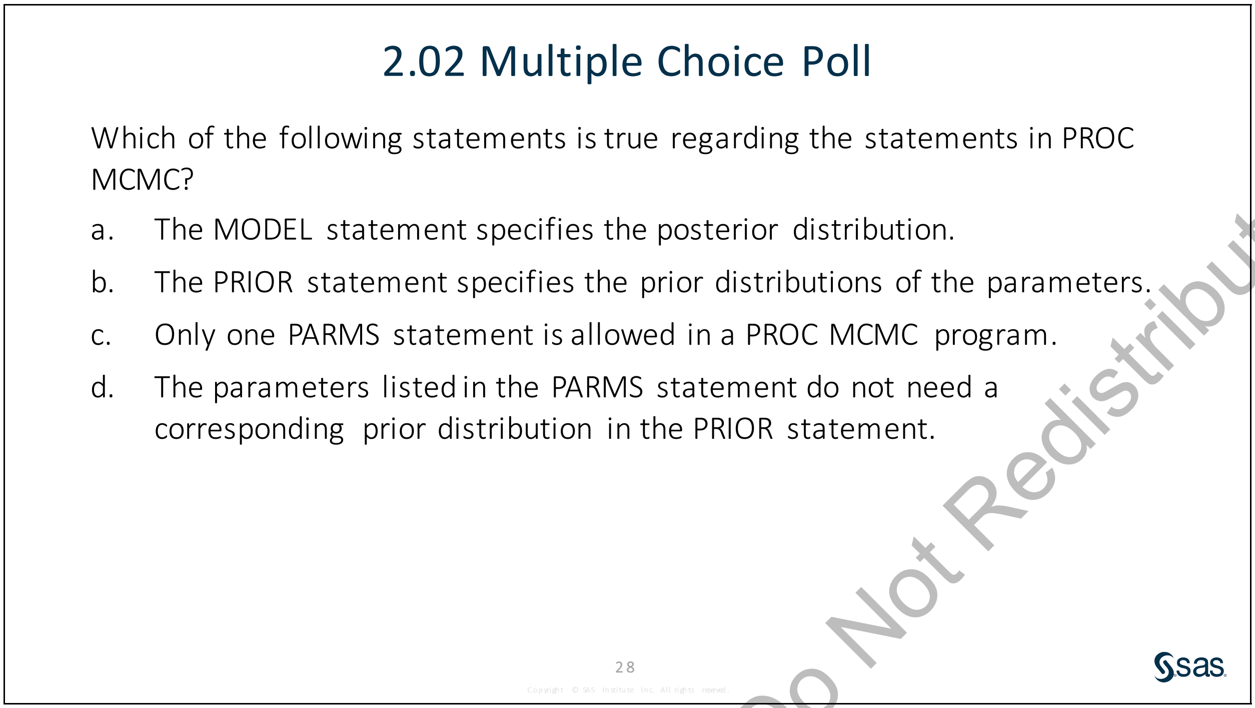
Conjugate Pairs

Family	Parameter	Prior
Normal with known μ	σ^2	Inverse gamma family
Normal with known μ	τ	gamma family
Normal with known σ^2, σ , or τ	μ	normal
Multivariate normal with known Σ	μ	multivariate normal
Multivariate normal with known μ	Σ	Inverse Wishart
Multinomial	P	Dirichlet
Binomial/binary	ρ	beta
Poisson	λ	gamma family

PROC MCMC recognizes certain configurations of the statistical models and applies conjugate sampling when appropriate. As was stated before, if the posterior distributions are in the same family as the prior probability distribution, the prior and the posterior are called conjugate distributions and it is possible to obtain closed-form solutions for the posterior distribution. Therefore, conjugate sampling is efficient because the Markov chain can obtain samples from the target distribution directly. The table above lists scenarios that lead to conjugate sampling in PROC MCMC.

In most cases, the distributions shown in the Family column of the table refers to the likelihood function. However, it does not necessarily have to be the case. The family is a distribution that is conditional on the parameter of interest, and it can appear in any level of the hierarchical model, including on the random-effects level. An example of a hierarchical model is shown later in the course.

PROC MCMC can detect conjugacy only if the model parameter (not a function or a transformation of the model parameter) is used in the prior and family distributions. If the parameter enters the likelihood function through a symbol or a transformation, then PROC MCMC resorts to the default sampling algorithm even though conjugacy still holds in theory. The sampling algorithm information can be found in the Parameters table, which is part of the analysis output.



2.02 Multiple Choice Poll

Which of the following statements is true regarding the statements in PROC MCMC?

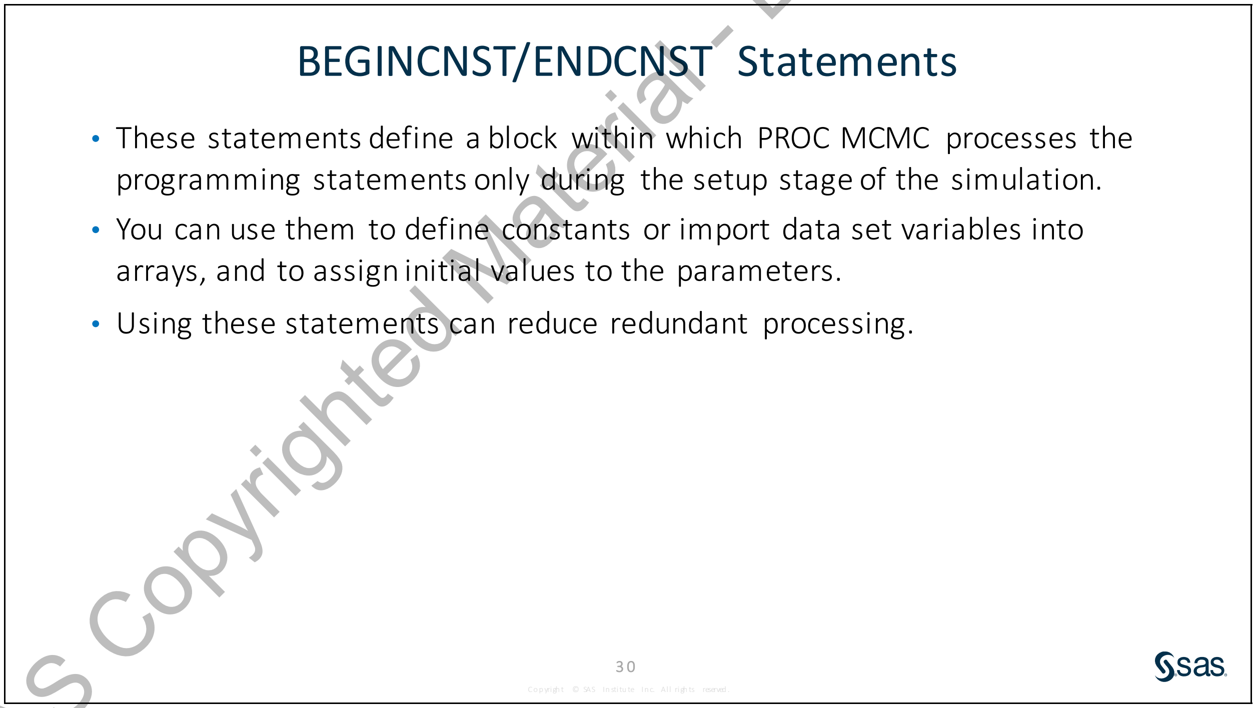
a. The MODEL statement specifies the posterior distribution.

b. The PRIOR statement specifies the prior distributions of the parameters.

c. Only one PARMS statement is allowed in a PROC MCMC program.

d. The parameters listed in the PARMS statement do not need a corresponding prior distribution in the PRIOR statement.

28

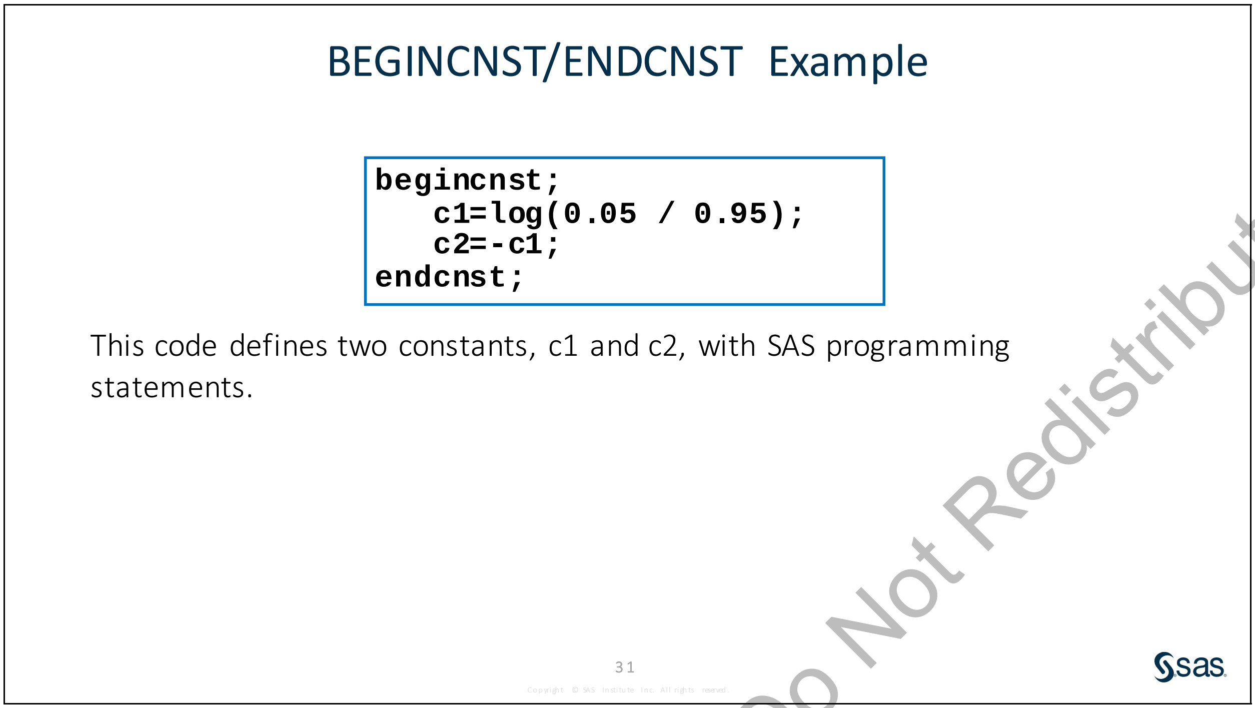


BEGINCNST/ENDCNST Statements

- These statements define a block within which PROC MCMC processes the programming statements only during the setup stage of the simulation.
- You can use them to define constants or import data set variables into arrays, and to assign initial values to the parameters.
- Using these statements can reduce redundant processing.

30

The MCMC procedure evaluates the programming statements with the BEGINCNST/ENDCNST block once and ignores them in the rest of the simulation.



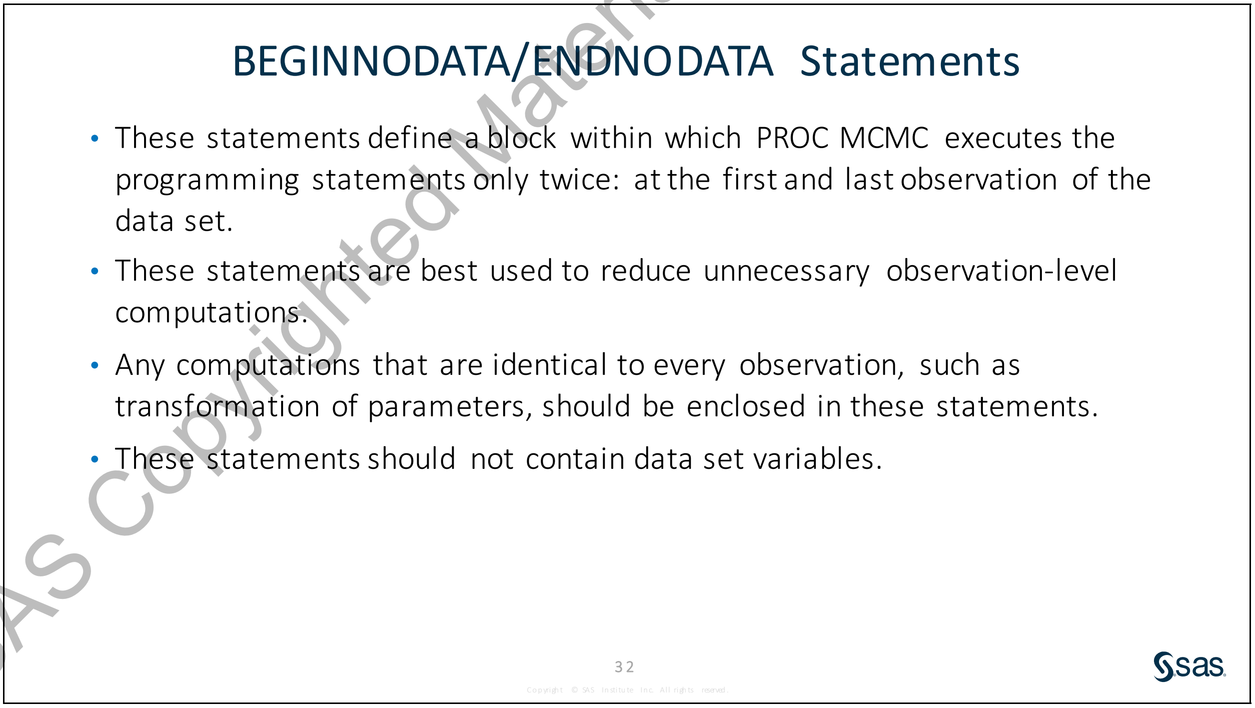
BEGINCNST/ENDCNST Example

```
begincnst;
   c1=log(0.05 / 0.95);
   c2=-c1;
endcnst;
```

This code defines two constants, c1 and c2, with SAS programming statements.

31

Whenever you have programming statements that calculate constants that do not need to be evaluated multiple times throughout the simulation, you should put them within the BEGINCNST and ENDCNST statements. Using these statements can reduce redundant processing.

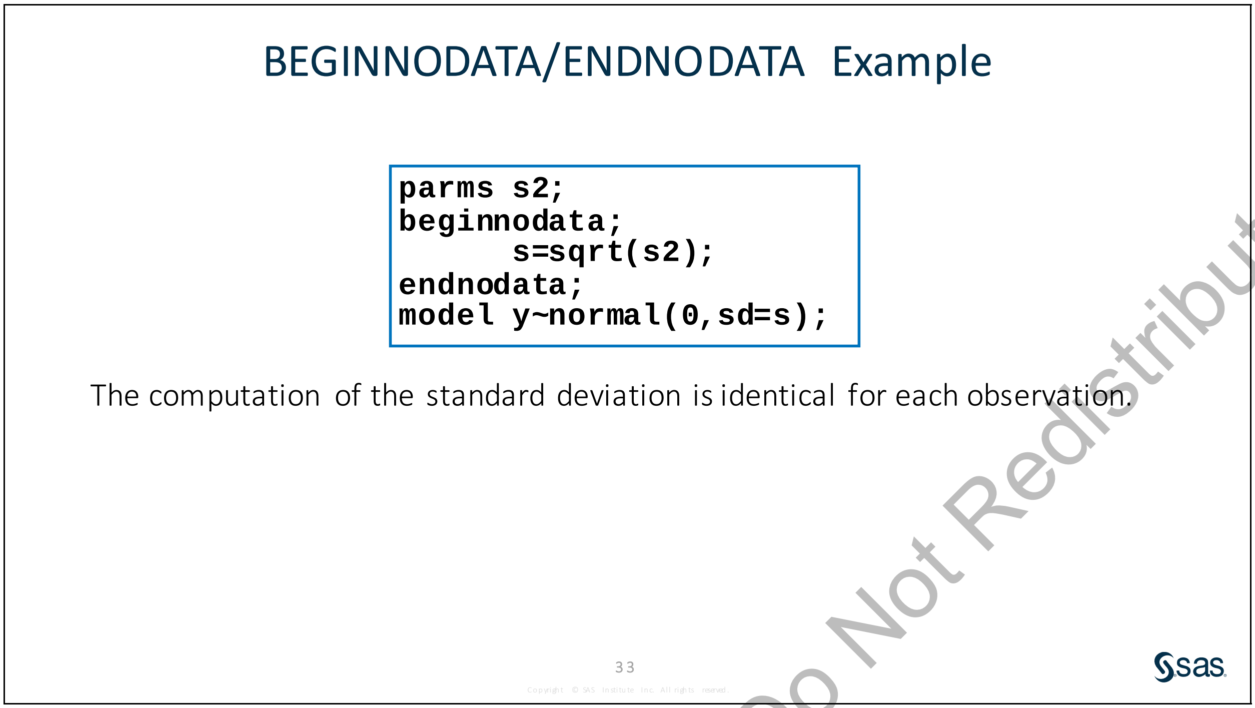


BEGINNODATA/ENDNODATA Statements

- These statements define a block within which PROC MCMC executes the programming statements only twice: at the first and last observation of the data set.
- These statements are best used to reduce unnecessary observation-level computations.
- Any computations that are identical to every observation, such as transformation of parameters, should be enclosed in these statements.
- These statements should not contain data set variables.

32

The BEGINNODATA and ENDNODATA statements are best reserved for calculations that relate to parameters only.



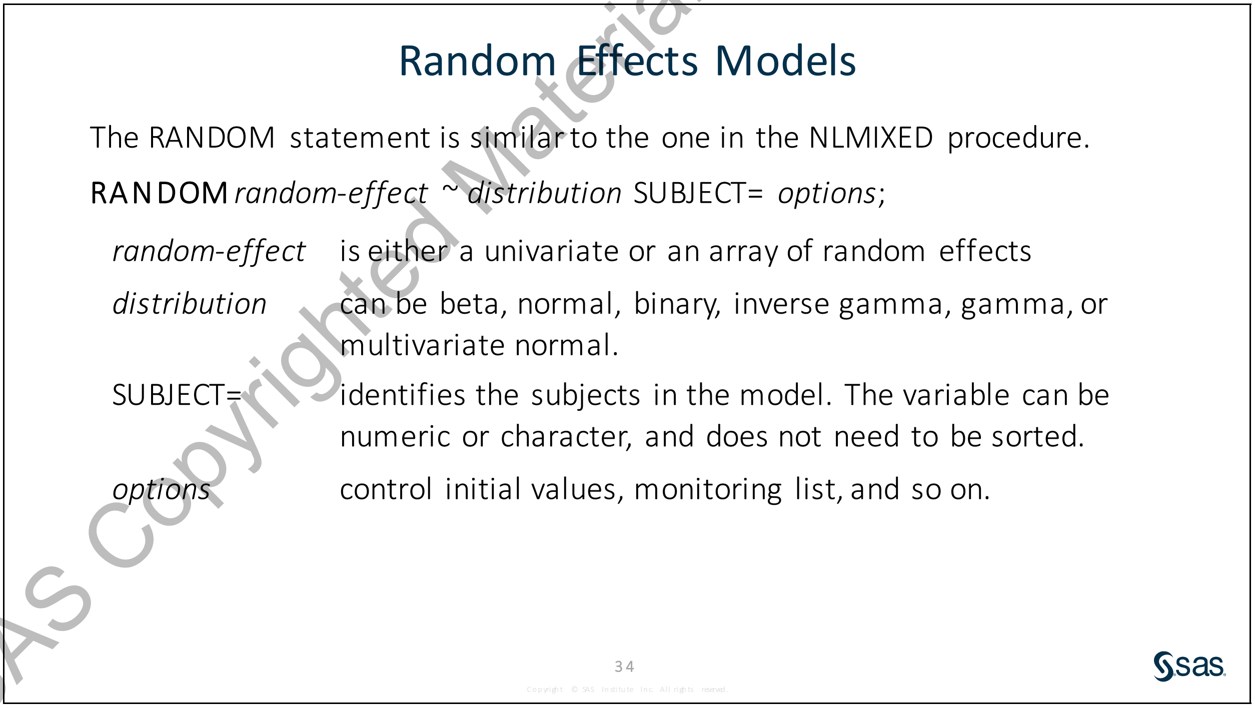
BEGINNODATA/ENDNODATA Example

```
parms s2;
beginnodata;
      s=sqrt(s2);
endnodata;
model y~normal(0,sd=s);
```

The computation of the standard deviation is identical for each observation.

33

The above slide shows code where you want to parameterize your model on the standard deviation but want to sample of the variance scale.



Random Effects Models

The RANDOM statement is similar to the one in the NLMIXED procedure.

RANDOM *random-effect* ~ *distribution* SUBJECT= *options*;

random-effect is either a univariate or an array of random effects

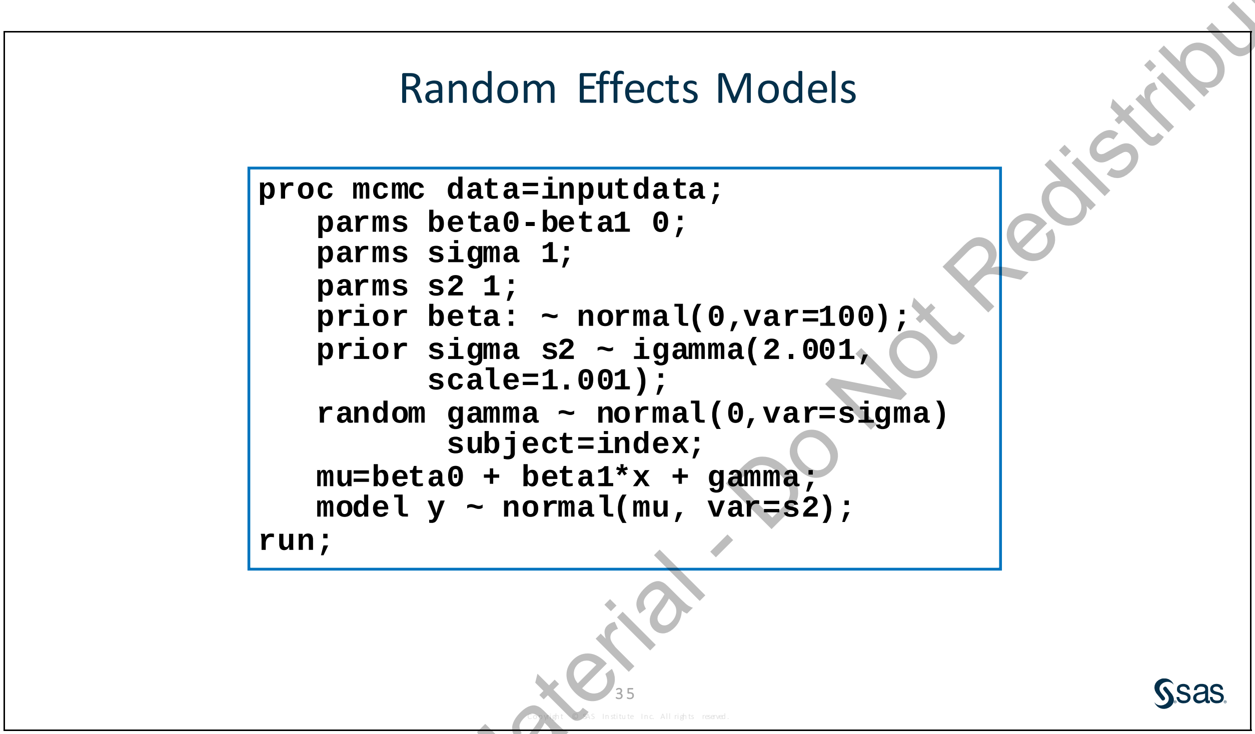
distribution can be beta, normal, binary, inverse gamma, gamma, or multivariate normal.

SUBJECT= identifies the subjects in the model. The variable can be numeric or character, and does not need to be sorted.

options control initial values, monitoring list, and so on.

34

The RANDOM statement defines a single random effect and its prior distribution or an array of random effects and their prior distribution. The statement must consist of a symbol for a random effect (or an array for multivariate random effects), a tilde, the distribution for the random effect, and then a SUBJECT= variable. You can use an arbitrary name to define the random effects as long as it does not conflict with names of the model parameters or data set variables used in the program. You can specify multiple RANDOM statements. Not all distributions supported in the MODEL statement are available for the RANDOM statement.



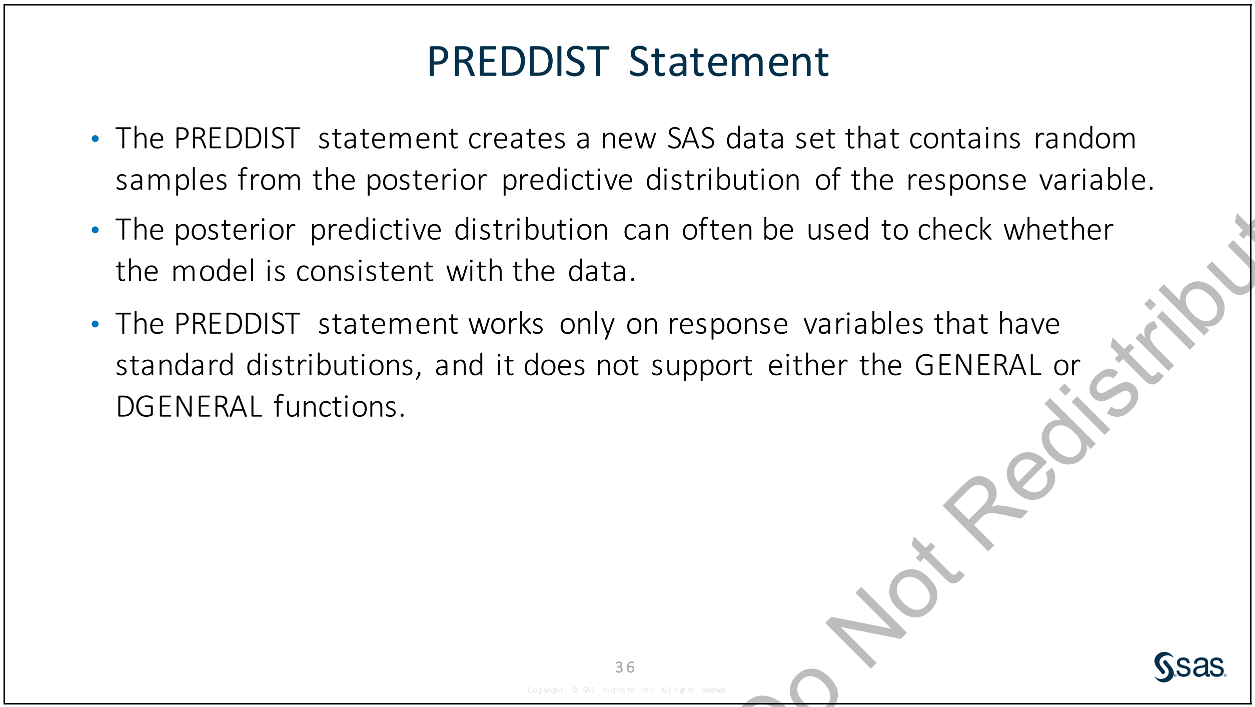
Random Effects Models

```
proc mcmc data=inputdata;
   parms beta0-beta1 0;
   parms sigma 1;
   parms s2 1;
   prior beta: ~ normal(0,var=100);
   prior sigma s2 ~ igamma(2.001,
         scale=1.001);
   random gamma ~ normal(0,var=sigma)
          subject=index;
   mu=beta0 + beta1*x + gamma;
   model y ~ normal(mu, var=s2);
run;
```

35

In the slide above, the RANDOM statement defines gamma as the random effect with a normal prior distribution with a mean of 0 and a variance defined by sigma. The SUBJECT=option, which is required, declares a data set variable (in this case **index**) that indicates the random effects' cluster membership. PROC MCMC assumes a priori that the random effects parameters within a cluster share the same prior distribution and that they are conditionally independent of each other, given the data and other parameters in the model. You can have multiple random effects in a model with each having its own prior distribution with the use of multiple RANDOM statements. Conditional independence is not assumed between random effect parameters that are defined by different RANDOM statements.

During the setup stage of the simulation, PROC MCMC first determines the number of random effects parameters, which is the number of unique values in the SUBJECT=variable. Then PROC MCMC creates the random effects parameters and updates them conditionally in the simulation. Using the SUBJECT=option bypasses the need to account for the number of clusters in the data set variable ahead of time.



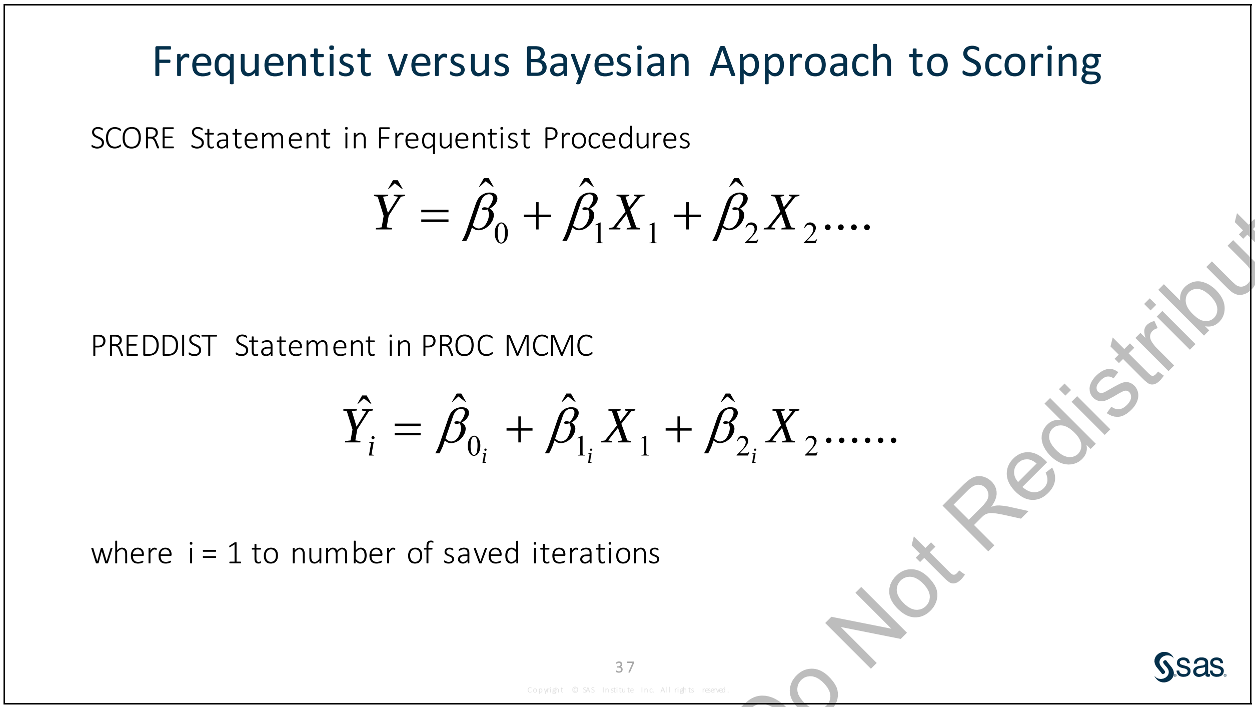
PREDDIST Statement

- The PREDDIST statement creates a new SAS data set that contains random samples from the posterior predictive distribution of the response variable.
- The posterior predictive distribution can often be used to check whether the model is consistent with the data.
- The PREDDIST statement works only on response variables that have standard distributions, and it does not support either the GENERAL or DGENERAL functions.

36

The posterior predictive distribution is the distribution of the unobserved predicted observations, conditional on the observed data. You can use the posterior predictive distribution to make predictions using new covariates and to check whether the model is consistent with the data. For example, you can generate samples from the posterior predictive distribution and see whether they differ systematically from the observed data.

Note: The posterior predictive distribution is an integral of the likelihood function $p(y_{pred} \mid \theta)$ with respect to the posterior distribution $p(\theta \mid y)$. The PREDDIST statement generates samples from a posterior predictive distribution based on draws from the posterior distribution of θ.



Frequentist versus Bayesian Approach to Scoring

SCORE Statement in Frequentist Procedures

$$\hat{Y} = \hat{\beta}_0 + \hat{\beta}_1 X_1 + \hat{\beta}_2 X_2 ....$$

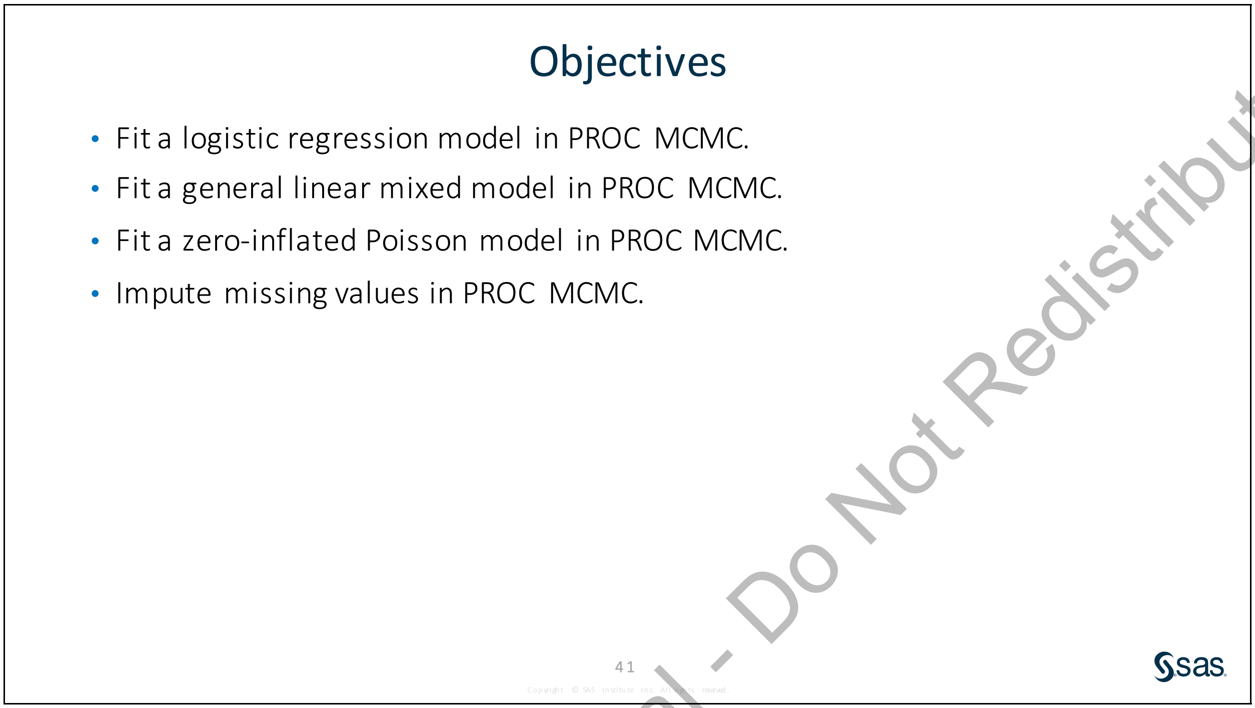
PREDDIST Statement in PROC MCMC

$$\hat{Y}_i = \hat{\beta}_{0_i} + \hat{\beta}_{1_i} X_1 + \hat{\beta}_{2_i} X_2 ......$$

where i = 1 to number of saved iterations

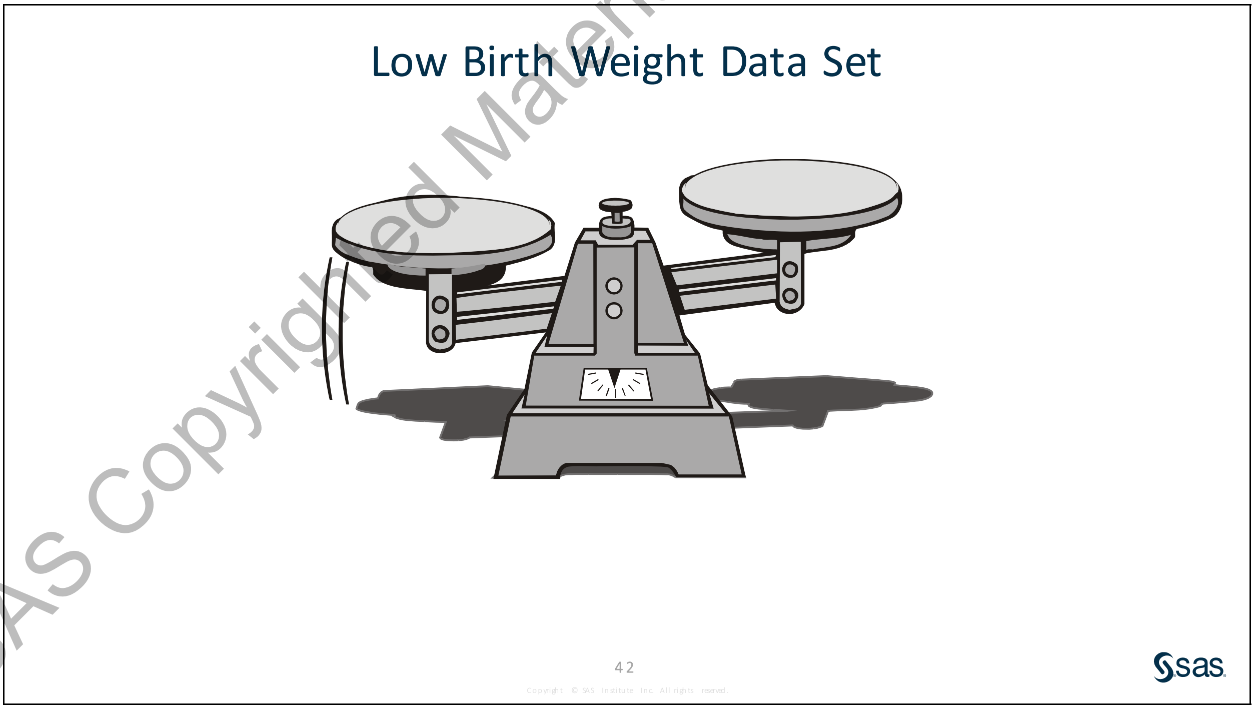
Scoring a data set, which is especially important for predictive modeling, means applying a previously fitted model to a new data set in order to compute the conditional values of each response category given the values of the explanatory variables in each observation. If the response variable is a binary variable, the conditional value is a probability. In the frequentist procedures such as PROC LOGISTIC and PROC GLMSELECT, the SCORE statement is used to score new data sets and output the scored values. However, only one value for each response category is generated. With the PREDDIST statement, multiple samples are drawn from the likelihood conditional on the posterior samples of the parameters. Therefore, a distribution of predicted values is generated and you can request the mean, standard deviation, and percentiles of that distribution.

2.2 Applications of the MCMC Procedure



Objectives

- Fit a logistic regression model in PROC MCMC.
- Fit a general linear mixed model in PROC MCMC.
- Fit a zero-inflated Poisson model in PROC MCMC.
- Impute missing values in PROC MCMC.



Low Birth Weight Data Set

Example: Babies with low birth weights (defined to be less than 2500 grams) are a concern because of their potential medical problems. Health researchers want to identify possible contributing factors to low birth weight and recommend strategies to reduce the number of low birth weight babies.

These are the variables in the data set:

low	low birth weight (1=yes, 0=no)
mother_wt	mother's weight at last menstrual period
alcohol	drinking status during pregnancy (1=yes, 0=no)
prev_pretrm	history of preterm labor (0=none, 1=one or more)
hist_hyp	history of hypertension (1=yes, 0=no).

The data are stored in a SAS data set named **sasuser.birth**.

Note: The data were modified from an example in Hosmer and Lemeshow (2000).

Fitting a Logistic Regression Model in PROC MCMC

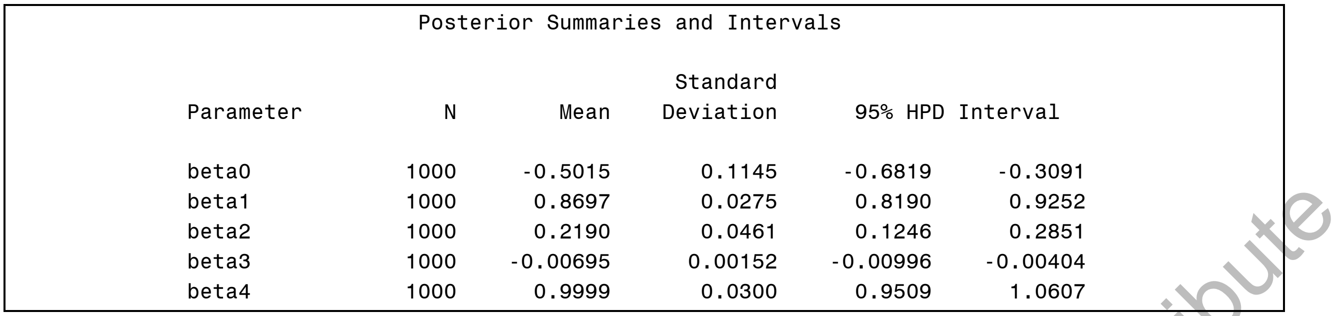
Example: Fit a logistic regression model on the low birth weight data set in PROC MCMC. Specify the initial values for the parameters as 0 and specify a noninformative prior with a normal distribution and a mean of 0 and a variance of 100. Display a fitted penalized B-spline curve for each trace plot and request all the diagnostic statistics and the DIC criterion.

```
/*  stbay02d01.sas  */
proc mcmc data=sasuser.birth diag=all dic
     plots(smooth)=all seed=27513;
   parms (beta0 beta1 beta2 beta3 beta4) 0;
   prior beta: ~ normal(0, var=100);
   p=logistic(beta0+beta1*alcohol+beta2*hist_hyp+
     beta3*mother_wt+beta4*prev_pretrm);
   model low ~ binary(p);
   title "Bayesian Analysis of Low Birth Weight Data";
run;
```

The PARMS statement specifies the five parameters with initial values of 0. The PRIOR statement specifies a normal prior distribution with a mean of 0 and a variance of 100 for each parameter. The *p* assignment statement computes the probability of low birth weight using the parameter estimates, data values, and the logit link transformation (with the SAS function LOGISTIC). The MODEL statement specifies that the response variable **low** has a binary distribution with a parameter *p*.

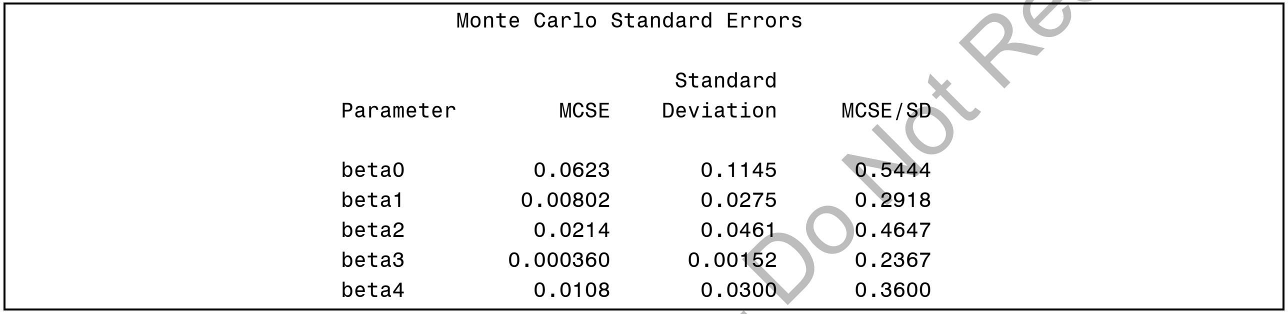
```
                       Bayesian Analysis of Low Birth Weight Data

                                  The MCMC Procedure

                        Number of Observations Read         189
                        Number of Observations Used         189

                                      Parameters

                             Sampling         Initial
      Block    Parameter     Method             Value    Prior Distribution

          1    beta0         N-Metropolis           0    normal(0, var=100)
               beta1                                0    normal(0, var=100)
               beta2                                0    normal(0, var=100)
               beta3                                0    normal(0, var=100)
               beta4                                0    normal(0, var=100)
```

The first table that PROC MCMC produces is the Number of Observations table. This table lists the number of observations read from the input data set and the number of nonmissing observations used in the analysis. The Parameters table lists the names of the parameters, the blocking information, the sampling method used, the starting values, and the prior distributions. You should to check this table to ensure that you have specified the parameters correctly, especially for complicated models.



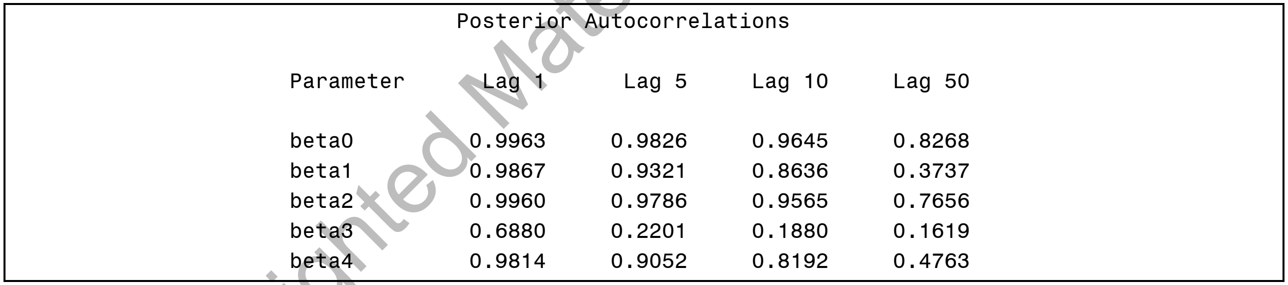
```
                               Posterior Summaries and Intervals

                                                   Standard
             Parameter           N        Mean    Deviation      95% HPD Interval

             beta0            1000     -0.5015       0.1145     -0.6819     -0.3091
             beta1            1000      0.8697       0.0275      0.8190      0.9252
             beta2            1000      0.2190       0.0461      0.1246      0.2851
             beta3            1000    -0.00695      0.00152    -0.00996    -0.00404
             beta4            1000      0.9999       0.0300      0.9509      1.0607
```

For each posterior distribution, PROC MCMC also reports summary statistics (posterior means and standard deviations) and interval statistics (95% highest posterior density credible intervals). The results show that the parameter estimate for alcohol use (**beta1**) is 0.8697 with a highest posterior density credible interval of [0.8190, 0.9252].



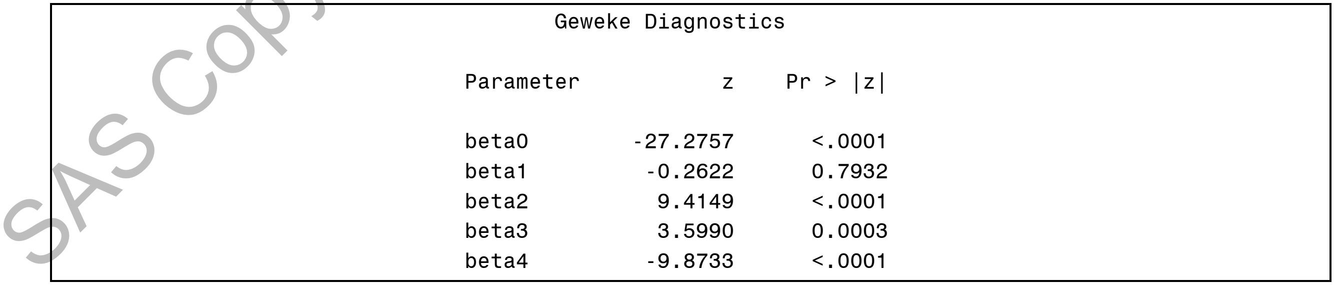
```
                                  Monte Carlo Standard Errors

                                                   Standard
                         Parameter        MCSE    Deviation     MCSE/SD

                         beta0          0.0623       0.1145      0.5444
                         beta1         0.00802       0.0275      0.2918
                         beta2          0.0214       0.0461      0.4647
                         beta3        0.000360      0.00152      0.2367
                         beta4          0.0108       0.0300      0.3600
```

The Monte Carlo Standard Errors table indicates that the standard errors of the mean estimates for each of the parameters are relatively large, with respect to the posterior standard deviations. The values in the MCSE/SD column (ratios of the standard errors and the standard deviations) are large. This means that a large fraction of the posterior variability is due to the simulation.



```
                                  Posterior Autocorrelations

                     Parameter      Lag 1      Lag 5     Lag 10     Lag 50

                     beta0         0.9963     0.9826     0.9645     0.8268
                     beta1         0.9867     0.9321     0.8636     0.3737
                     beta2         0.9960     0.9786     0.9565     0.7656
                     beta3         0.6880     0.2201     0.1880     0.1619
                     beta4         0.9814     0.9052     0.8192     0.4763
```

The Posterior Autocorrelations table shows that the autocorrelations among posterior samples are large even after lag 50.



```
                                      Geweke Diagnostics

                               Parameter           z    Pr > |z|

                               beta0        -27.2757      <.0001
                               beta1         -0.2622      0.7932
                               beta2          9.4149      <.0001
                               beta3          3.5990      0.0003
                               beta4         -9.8733      <.0001
```

The Geweke Diagnostics table indicates that four parameters failed the test.

```
                       Raftery-Lewis Diagnostics
      Quantile=0.025 Accuracy=+/-0.005 Probability=0.95 Epsilon=0.001

                         Number of Samples          Dependence
       Parameter     Burn-In       Total     Minimum        Factor

       beta0               .           .        3746         .
       beta1               .           .        3746         .
       beta2               .           .        3746         .
       beta3               .           .        3746         .
       beta4               .           .        3746         .

       NOTE: The minimum required sample size of 3746 is larger
       than the available chain length of 1000.
```

The Raftery-Lewis Diagnostics table shows that the Markov chain sample size of 1000 is not enough to perform the diagnostic test. The minimum sample size needed is 3746. One solution is to increase the number of iterations.

```
                            Heidelberger-Welch Diagnostics

                       Stationarity Test                         Half-Width Test
            Cramer-von             Test      Iterations     Half-              Relative  Test
Parameter   Mises Stat   p-Value   Outcome    Discarded     Width       Mean  Half-Width  Outcome

beta0            .         .       Failed             .         .          .           .
beta1            .         .       Failed             .         .          .           .
beta2            .         .       Failed             .         .          .           .
beta3            .         .       Failed             .         .          .           .
beta4            .         .       Failed             .         .          .           .

                                 Effective Sample Sizes

                                         Autocorrelation
                      Parameter        ESS              Time    Efficiency

                      beta0            3.4             296.4        0.0034
                      beta1           11.7           85.1609        0.0117
                      beta2            4.6             216.0        0.0046
                      beta3           17.8           56.0348        0.0178
                      beta4            7.7             129.6        0.0077
```

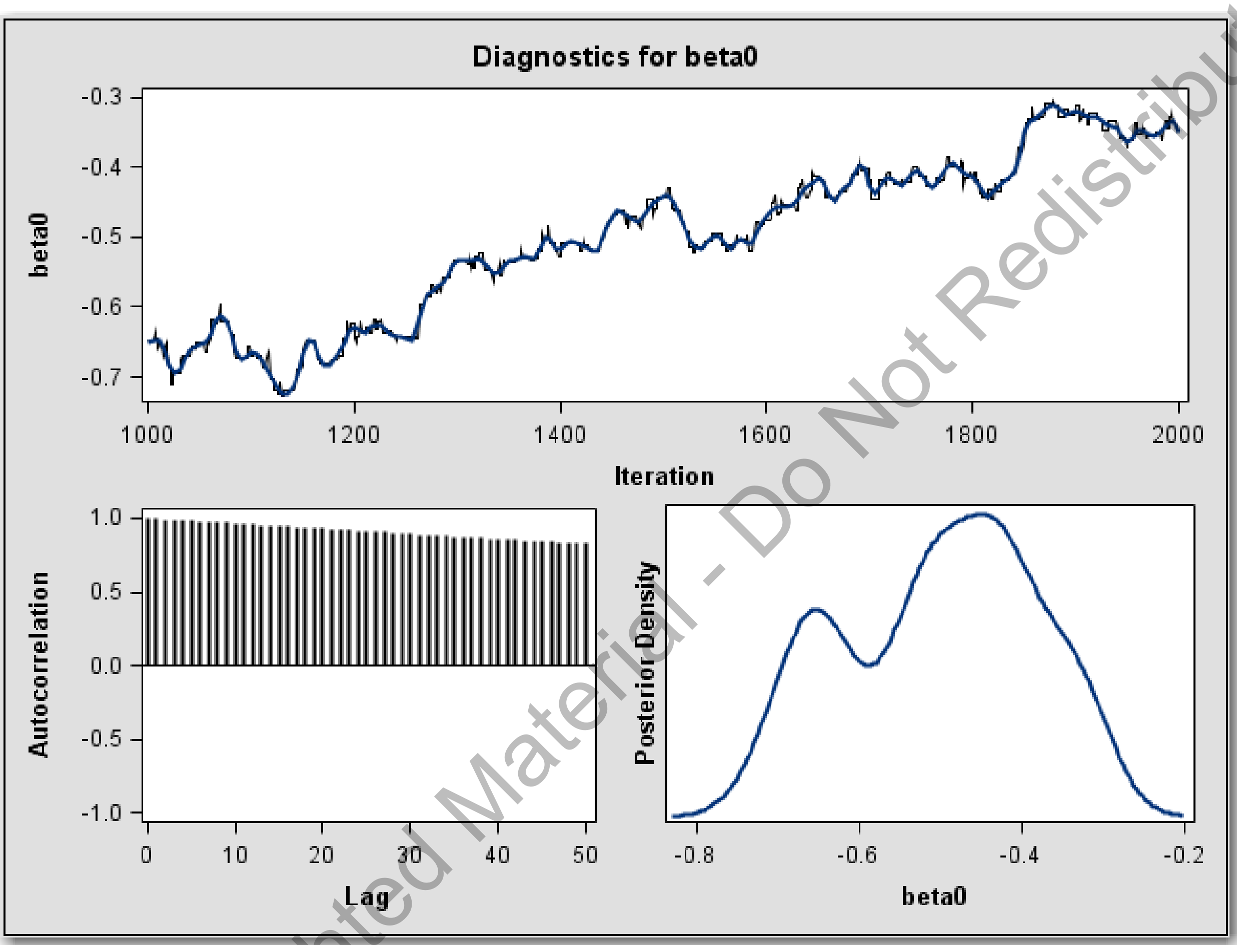
The Heidelberger-Welch Diagnostics all indicate rejection and the effective sample sizes are much lower than the actual sample size, indicating slower mixing of the Markov chain.

```
                           Deviance Information Criterion

         Dbar (posterior mean of deviance)                 217.014
         Dmean (deviance evaluated at posterior mean)      215.799
         pD (effective number of parameters)                 1.216
         DIC (smaller is better)                           218.230
```

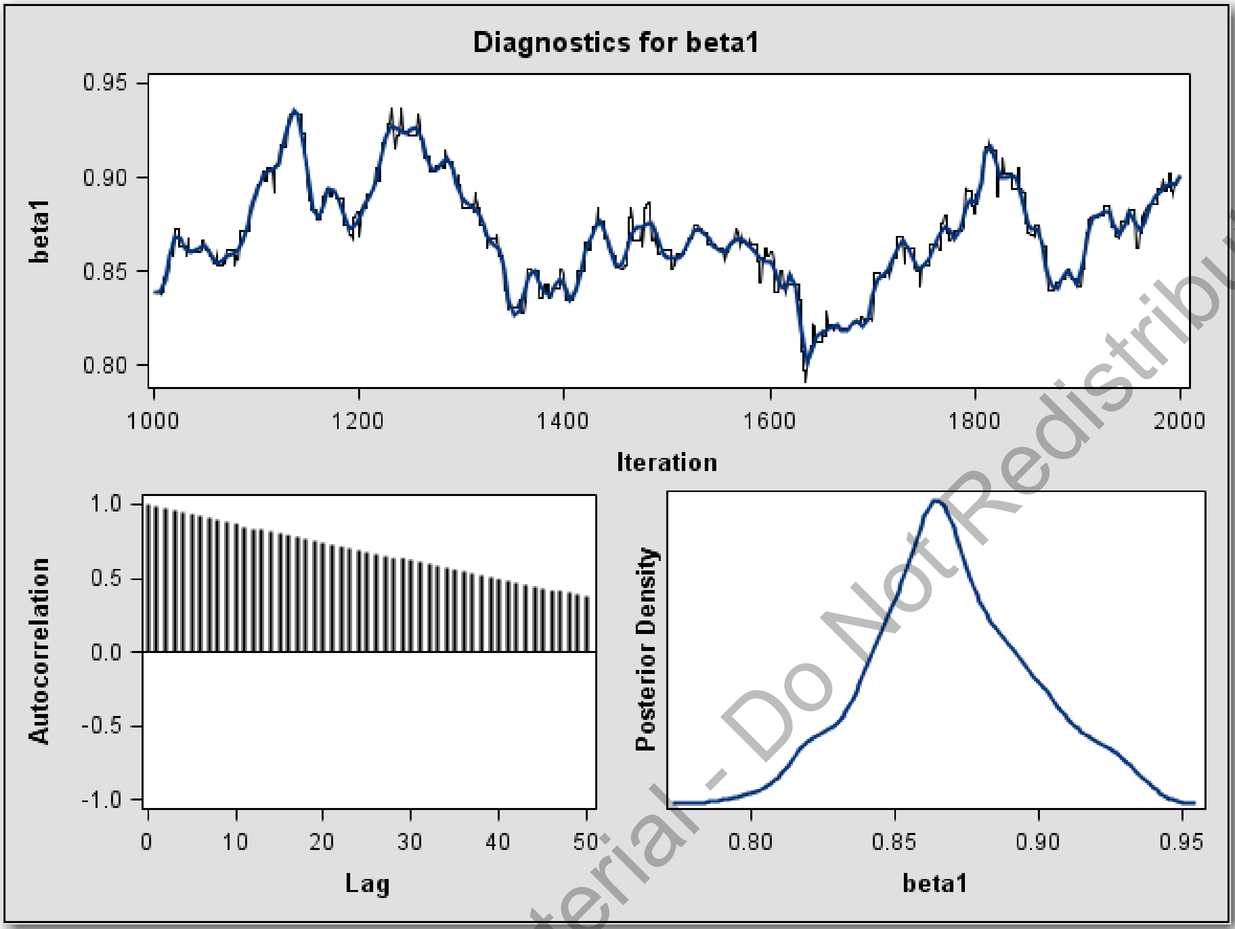
The deviance information criterion (DIC) (Spiegelhalter et al. 2002) is a model assessment tool, and it is a Bayesian alternative to Akaike's information criterion (AIC) and the Bayesian information criterion (BIC, also known as the Schwarz criterion). The DIC uses the posterior summaries, which means that it takes the prior information into account. A smaller DIC indicates a better fit to the data set.

The effective number of parameters describes the complexity of the model, and it serves as a penalization term that corrects deviance’s propensity toward models with more parameters. In this model, the effective number of parameters should be around 5. A value of 1.216 indicates a possible convergence problem with the model.

Partial Graphics Output:



The trace plot for **beta0** shows very poor mixing of the Markov chain and very high autocorrelations.



The trace plot for **beta1** shows very poor mixing and high autocorrelations.

Note: The remaining diagnostic plots (not shown) show patterns of non-convergence.

With the convergence diagnostic statistics and the diagnostic plots all showing poor mixing, the results of this model are inadequate for Bayesian analysis.

Example: Refit the Bayesian model for the low birth weight data but increase the number of iterations to 400,000, the number of iterations to use in each proposal tuning phase to 5,000, the number of burn-in iterations to 5000, the thinning rate to 10, and specify the quasi-Newton optimization in constructing the initial covariance matrix for the Metropolis-Hastings algorithm. Furthermore, create an output data set of the posterior samples of all model parameters, display all the posterior statistics and the DIC, and display the Markov chain sampling history.

```
proc mcmc data=sasuser.birth outpost=birthout diag=all dic
     propcov=quanew nbi=5000 ntu=5000 nmc=400000 thin=10
     mchistory=brief plots(smooth)=all seed=27513 stats=all;
   parms (beta0 beta1 beta2 beta3 beta4) 0;
   prior beta: ~ normal(0, var=100);
   p=logistic(beta0+beta1*alcohol+beta2*hist_hyp+
              beta3*mother_wt+beta4*prev_pretrm);
   model low ~ binary(p);
   title "Bayesian Analysis of Low Birth Weight Data";
run;
```

Selected PROC MCMC statement options:

OUTPOST= specifies an output data set that contains the posterior samples of all model parameters.

PROPCOV= specifies the method used in constructing the initial covariance matrix for the Metropolis-Hastings algorithm. The quasi-Newton optimization (QUANEW) and the Nelder-Mead simplex optimization (NMSIMP) methods find numerically approximated covariance matrices at the optimum of the posterior density function with respect to all continuous parameters. The optimization does not apply to discrete parameters. The tuning phase starts at the optimized values; in some problems, this can greatly increase convergence performance.

NTU= specifies the number of iterations to use in each proposal tuning phase. By default, NTU=500.

NMC= specifies the number of iterations in the main simulation loop. This is the MCMC sample size if there is no thinning. By default, NMC=1000.

THIN=n controls the thinning rate of the simulation. PROC MCMC keeps every nth simulation sample and discards the rest. All of the posterior statistics and diagnostics are calculated using the thinned samples. By default, THIN=1.

NBI=n specifies the number of burn-in iterations to perform before beginning to save parameter estimate chains. By default, NBI=1000.

MCHISTORY= controls the display of the Markov chain sampling history. The keyword BRIEF produces a summary output for the tuning, burn-in, and sampling history tables.

STATS= specifies options for posterior statistics. You can request all of the posterior statistics by specifying STATS=ALL. You can suppress all the calculations by specifying STATS=NONE.

The number of iterations to use in each proposal tuning phase should be increased to a large number (for example, 5,000) so that the Metropolis algorithm has a reasonably well fine-tuned proposal density.

```
                         Bayesian Analysis of Low Birth Weight Data

                                   The MCMC Procedure

                         Number of Observations Read         189
                         Number of Observations Used         189

                                       Parameters

                          Sampling         Initial
    Block    Parameter    Method             Value    Prior Distribution

        1    beta0        N-Metropolis           0    normal(0, var=100)
             beta1                               0    normal(0, var=100)
             beta2                               0    normal(0, var=100)
             beta3                               0    normal(0, var=100)
             beta4                               0    normal(0, var=100)
```

```
                                          Tuning History

                                                             RWM
                                                      Acceptance
                                 Phase       Scale          Rate

                                     1      2.3800        0.2842
                                     2      2.3800        0.2888

                                          Burn-In History

                                                          RWM
                                                   Acceptance
                                         Scale           Rate

                                        2.3800         0.2990

                                         Sampling History

                                                          RWM
                                                   Acceptance
                                         Scale           Rate

                                        2.3800         0.2992
```

Note: The numeric difference seen between the course notes and the output will be discussed at the conclusion of this demonstration.

One key factor in achieving high efficiency of a Metropolis-based Markov chain is finding a good proposal distribution for each block of parameters. This process is referred to as tuning. The tuning of the Markov chain is broken into a number of phases. In each phase, PROC MCMC generates trial samples and automatically modifies the proposal distribution as a result of the acceptance rate. The acceptance probability is the percentage of iterations in each proposal tuning phase that have been accepted. If the probability falls within the acceptance tolerance range, the current configuration is kept. Otherwise, these parameters are modified before the next tuning loop.

The acceptance rate is closely related to the sampling efficiency of a Metropolis chain. For a random walk Metropolis, high acceptance rate means that most new samples occur right around the current data point. Their frequent acceptance means that the Markov chain is moving rather slowly and not exploring the parameter space fully. On the other hand, a low acceptance rate means that the proposed samples are often rejected. Hence, the chain is not moving much because the chain is repeating the same values for each iteration for an extended length of time. An efficient Metropolis sampler has an acceptance rate that is neither too high nor too low. Roberts, Gelman, and Gilks (1997) showed that if both the target and proposal densities are normal, the optimal acceptance probability for the Markov chain should be around 0.45 in a single dimensional problem, and asymptotically approach 0.234 in higher dimensions. The corresponding optimal scale is 2.38, which is the initial scale set for each block.

The scale value effectively controls this acceptance probability. If the observed acceptance rate in a given tuning phase is less than the lower bound defined by the PROC MCMC statement options, the scale is reduced. If the observed acceptance rate is greater than the upper bound defined by the PROC MCMC statement options, the scale is increased.

In this example, PROC MCMC found an acceptable proposal distribution after 2 phases, and this distribution is used in both the burn-in and sampling stages of the simulation.

The Burn-In History table shows the acceptance rate for the burn-in phase, and the Sampling History table shows the acceptance rate for the main phase sampling.

```
                                   Posterior Summaries

                                             Standard             Percentiles
       Parameter            N        Mean    Deviation          25          50          75

       beta0            40000      0.8982       0.8825      0.2975      0.8753      1.4802
       beta1            40000      0.5154       0.3519      0.2805      0.5163      0.7527
       beta2            40000      1.9367       0.7326      1.4400      1.9155      2.4174
       beta3            40000     -0.0181      0.00689     -0.0227     -0.0178     -0.0134
       beta4            40000      1.3319       0.4475      1.0306      1.3266      1.6305

                                   Posterior Intervals

               Parameter    Alpha     Equal-Tail Interval        HPD Interval

               beta0        0.050     -0.7755      2.6950     -0.7810      2.6844
               beta1        0.050     -0.1807      1.1996     -0.1702      1.2081
               beta2        0.050      0.5437      3.4370      0.4691      3.3448
               beta3        0.050     -0.0323    -0.00527     -0.0317    -0.00478
               beta4        0.050      0.4587      2.2197      0.4327      2.1897
```

The parameter estimate for alcohol use (**beta1**) is now 0.5154 with a highest posterior density credible interval of -0.1702 and 1.2081. These results are very different from the first model, which shows the results from models with poor Markov chain convergence cannot be trusted.

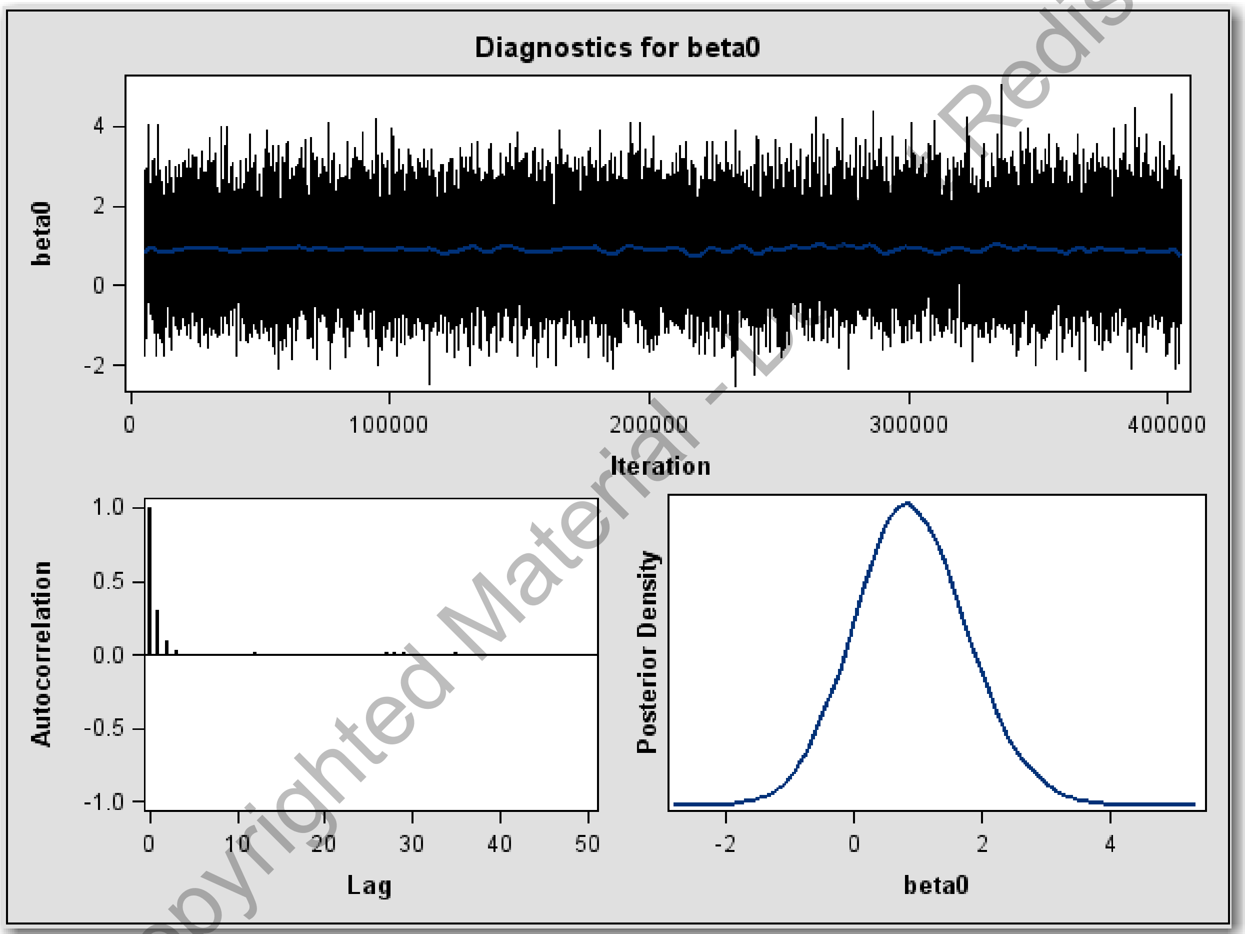
```
                               Posterior Correlation Matrix

           Parameter         beta0         beta1         beta2         beta3         beta4

           beta0            1.0000       -0.1956        0.2802       -0.9609       -0.0889
           beta1           -0.1956        1.0000       -0.0141        0.0339       -0.1495
           beta2            0.2802       -0.0141        1.0000       -0.3451        0.0447
           beta3           -0.9609        0.0339       -0.3451        1.0000        0.0184
           beta4           -0.0889       -0.1495        0.0447        0.0184        1.0000

                               Posterior Covariance Matrix

        Parameter         beta0          beta1          beta2          beta3          beta4

        beta0            0.7788        -0.0607         0.1812       -0.00584        -0.0351
        beta1           -0.0607         0.1238       -0.00363       0.000082        -0.0235
        beta2            0.1812       -0.00363         0.5367       -0.00174         0.0147
        beta3          -0.00584       0.000082       -0.00174       0.000048       0.000057
        beta4           -0.0351        -0.0235         0.0147       0.000057         0.2003

                               Monte Carlo Standard Errors

                                                Standard
                          Parameter        MCSE    Deviation     MCSE/SD

                          beta0         0.00600       0.8825     0.00680
                          beta1         0.00241       0.3519     0.00685
                          beta2         0.00505       0.7326     0.00690
                          beta3        0.000047      0.00689     0.00682
                          beta4         0.00306       0.4475     0.00683
```

```
                                 Posterior Autocorrelations

                    Parameter      Lag 1      Lag 5     Lag 10     Lag 50

                    beta0         0.3005     0.0010     0.0070    -0.0053
                    beta1         0.3116     0.0120    -0.0039     0.0012
                    beta2         0.3027     0.0036    -0.0025    -0.0038
                    beta3         0.3044    -0.0029     0.0059    -0.0035
                    beta4         0.3053    -0.0020    -0.0043    -0.0061

                                     Geweke Diagnostics

                              Parameter           z    Pr > |z|

                              beta0         -0.5959      0.5512
                              beta1         -0.2568      0.7974
                              beta2         -0.9862      0.3240
                              beta3          1.3847      0.1661
                              beta4         -0.6149      0.5386

                                  Raftery-Lewis Diagnostics
               Quantile=0.025 Accuracy=+/-0.005 Probability=0.95 Epsilon=0.001

                                      Number of Samples           Dependence
                 Parameter     Burn-In       Total     Minimum        Factor

                 beta0               4        4622        3746        1.2338
                 beta1               4        4872        3746        1.3006
                 beta2               3        4502        3746        1.2018
                 beta3               4        5038        3746        1.3449
                 beta4               5        8035        3746        2.1450

                                Heidelberger-Welch Diagnostics

                       Stationarity Test                           Half-Width Test
           Cramer-von           Test      Iterations     Half-              Relative  Test
Parameter  Mises Stat  p-Value  Outcome    Discarded     Width      Mean  Half-Width  Outcome

beta0          0.0569   0.8335  Passed             0    0.0127    0.8982      0.0141  Passed
beta1          0.1898   0.2880  Passed             0   0.00498    0.5154     0.00967  Passed
beta2          0.1284   0.4626  Passed             0   0.00816    1.9367     0.00422  Passed
beta3          0.0930   0.6203  Passed             0  0.000099   -0.0181    -0.00549  Passed
beta4          0.0328   0.9660  Passed             0   0.00643    1.3319     0.00483  Passed


                                    Effective Sample Sizes

                                             Autocorrelation
                    Parameter         ESS               Time    Efficiency

                    beta0         21627.7             1.8495        0.5407
                    beta1         21298.7             1.8781        0.5325
                    beta2         21018.8             1.9031        0.5255
                    beta3         21499.4             1.8605        0.5375
                    beta4         21431.6             1.8664        0.5358
```

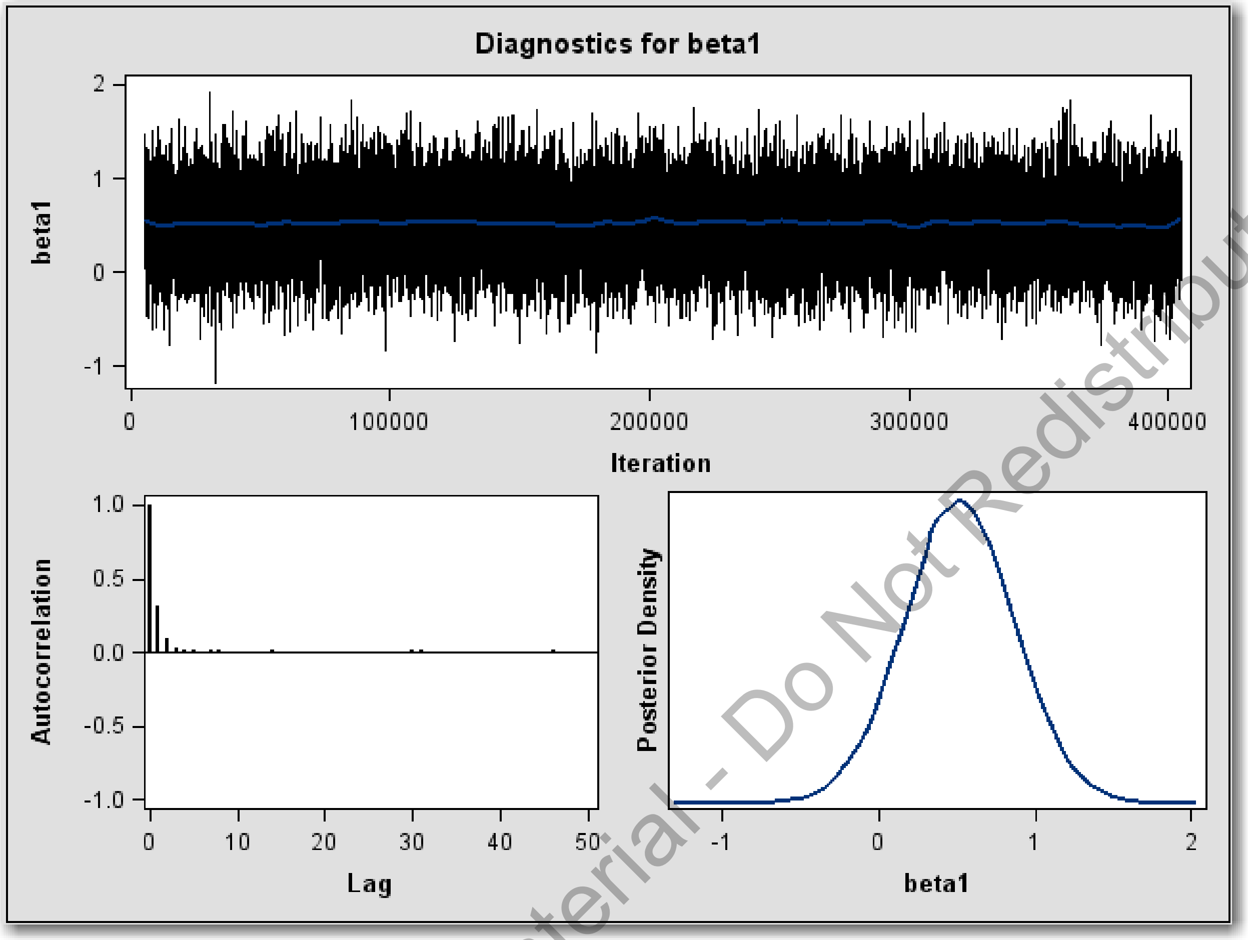
```
                         Deviance Information Criterion

             Dbar (posterior mean of deviance)                212.936
             Dmean (deviance evaluated at posterior mean)     207.885
             pD (effective number of parameters)                5.051
             DIC (smaller is better)                          217.988
```

The diagnostic statistics show no indication that the Markov chain has not reached convergence. The posterior autocorrelations are small, the Geweke diagnostics are not significant, and effective number of parameters is much closer to the actual number of parameters (5).

Partial Graphics Output:



The diagnostic plots show a huge improvement over the last model for **beta0**. The trace of the samples centers on 0.90 with a relatively constant mean and variance, and the autocorrelations are quite small after about 2 lags.



The diagnostic plots show a huge improvement over the last model for **beta1**. The trace of the samples centers on 0.52 with a relatively constant mean and variance, and the autocorrelations are quite small after about 2 lags.

Note: The remaining diagnostic plots (not shown) show patterns of convergence.

Example: Using the output data set with the generated posterior samples from the Bayesian model of the low birth weight data set, generate the probability that the parameter estimates are greater than 0.

```
data birthout;
   set birthout;
   alc=(beta1 gt 0);
   hist=(beta2 gt 0);
   wt=(beta3 gt 0);
   pretrm=(beta4 gt 0);
run;

proc means data=birthout mean maxdec=8;
   var alc hist wt pretrm;
   title "Proportion of Parameter Estimates Greater than Zero";
run;
```

```
                  Proportion of Parameter Estimates Greater than Zero

                                     The MEANS Procedure

                                   Variable            Mean
                                   ________________________
                                   alc           0.92885000
                                   hist          0.99682500
                                   wt            0.00245000
                                   pretrm        0.99870000
                                   ________________________
```

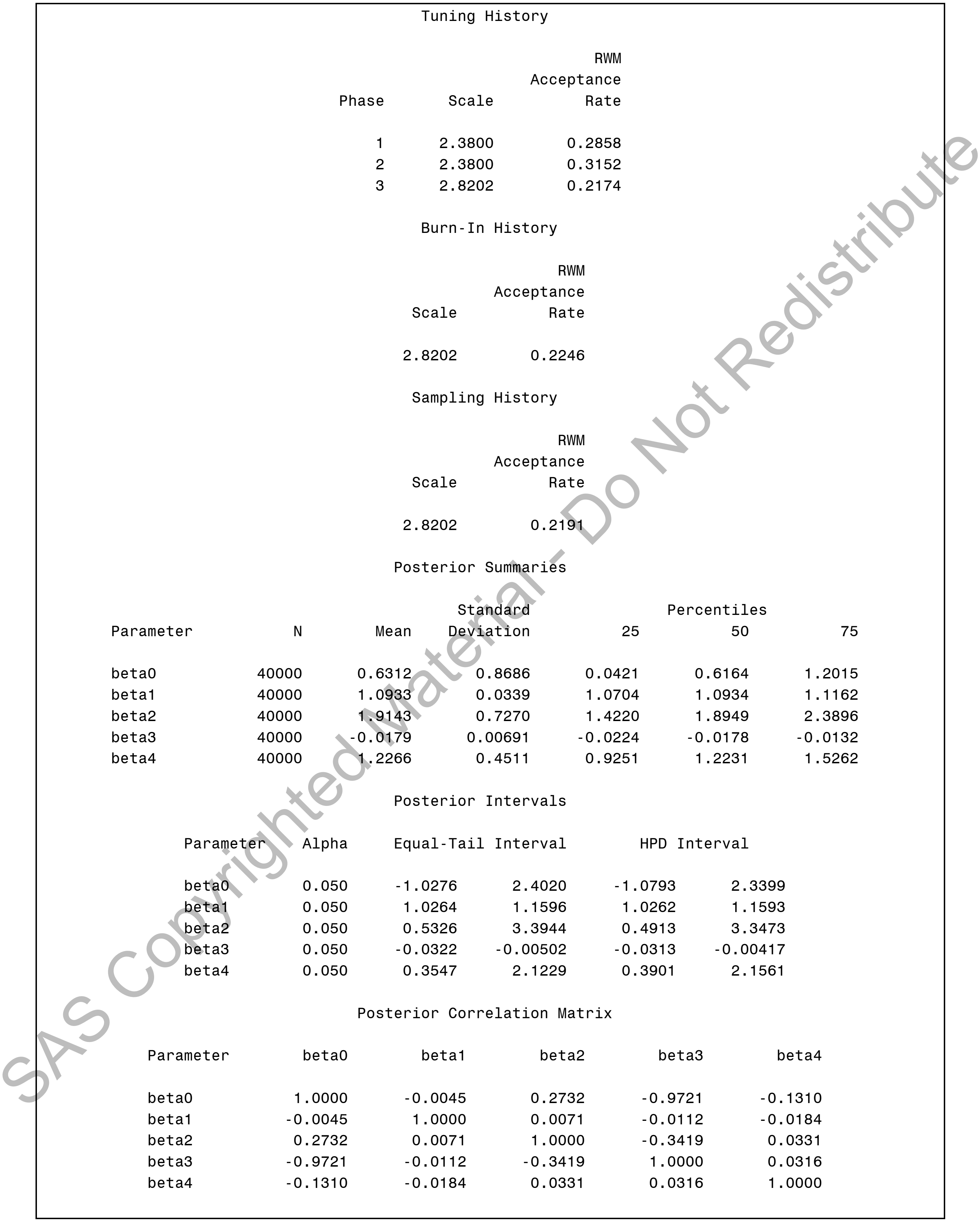
The probabilities are very similar to the ones generated by the model fit in PROC GENMOD.

Example: Use an informative prior for the parameter estimate for alcohol with a mean of 1.0986 and a variance of 0.00116. Use noninformative priors for the other parameter estimates. Use the same options as the previous model with regard to model tuning, create an output data set with the posterior samples, display a fitted penalized B-spline curve for each trace plot, and request all the diagnostic convergence statistics along with the DIC.

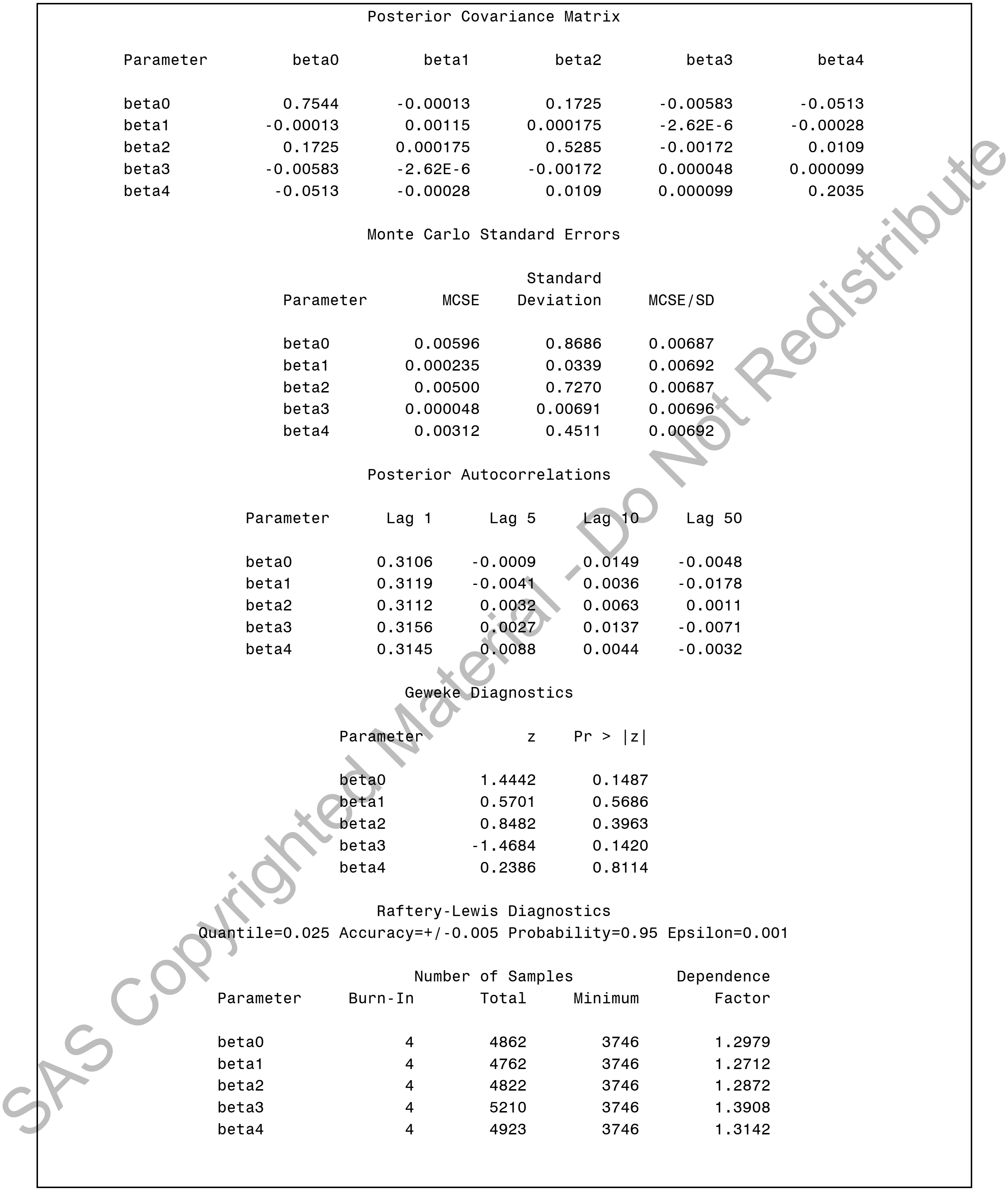
```
proc mcmc data=sasuser.birth outpost=birthout1 diag=all dic
     propcov=quanew nbi=5000 ntu=5000 nmc=400000 thin=10
     mchistory=brief plots(smooth)=all seed=27513 stats=all;
   parms (beta0 beta1 beta2 beta3 beta4) 0;
   prior beta1 ~ normal(1.0986,var=0.00116);
   prior beta0 beta2 beta3 beta4 ~ normal(0, var=100);
   p=logistic(beta0+beta1*alcohol+beta2*hist_hyp+
              beta3*mother_wt+beta4*prev_pretrm);
   model low ~ binary(p);
   title "Bayesian Analysis of Low Birth Weight Data";
run;
```

```
                           Bayesian Analysis of Low Birth Weight Data

                                      The MCMC Procedure

                            Number of Observations Read         189
                            Number of Observations Used         189

                                          Parameters

                             Sampling         Initial
      Block    Parameter     Method             Value    Prior Distribution

          1    beta0         N-Metropolis           0    normal(0, var=100)
               beta1                                0    normal(1.0986,var=0.00116)
               beta2                                0    normal(0, var=100)
               beta3                                0    normal(0, var=100)
               beta4                                0    normal(0, var=100)
```

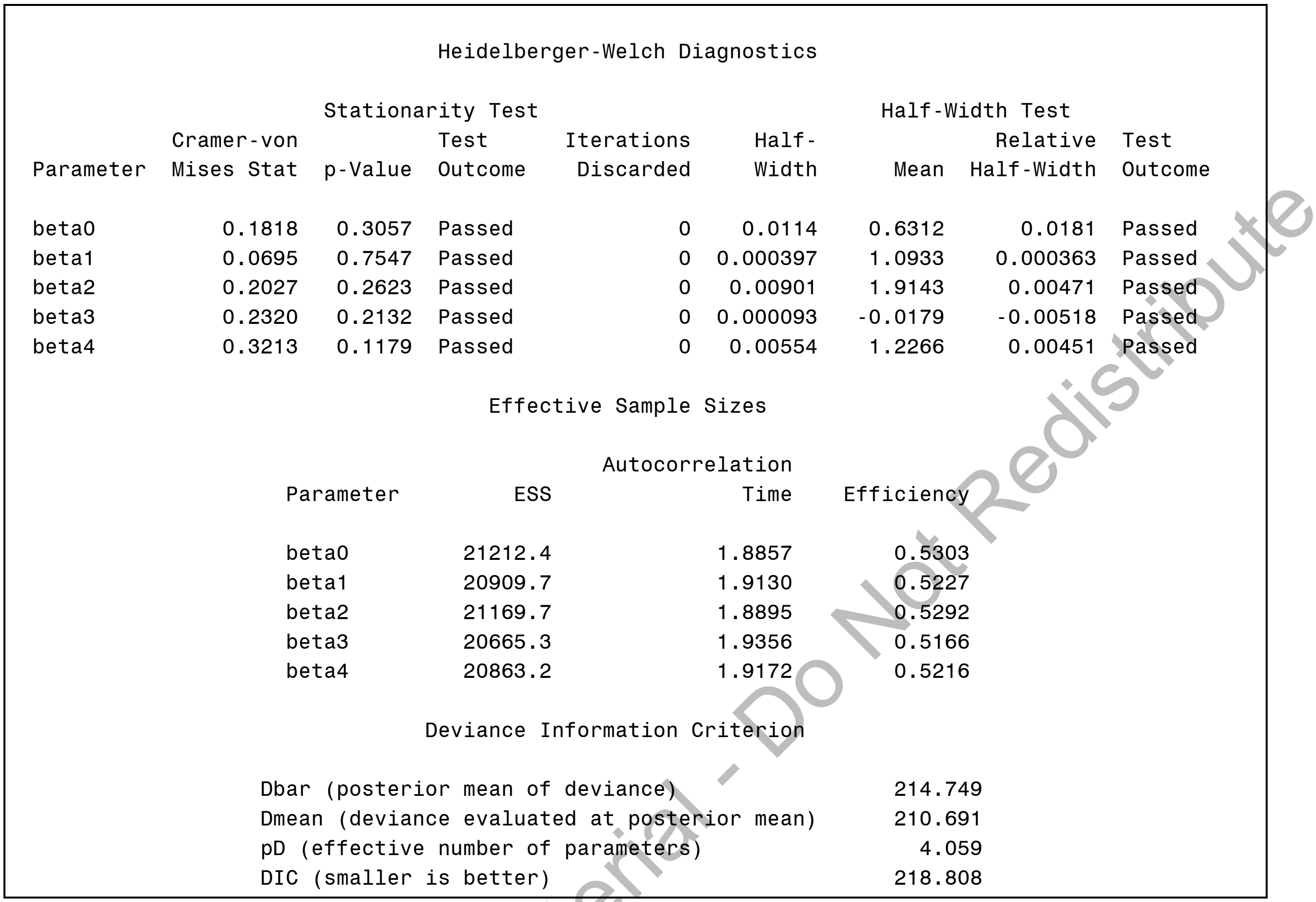
The Parameters table shows the informative prior for **beta1**.



```
                                 Tuning History

                                                  RWM
                                           Acceptance
                        Phase       Scale        Rate

                            1      2.3800      0.2858
                            2      2.3800      0.3152
                            3      2.8202      0.2174

                                 Burn-In History

                                              RWM
                                       Acceptance
                             Scale           Rate

                            2.8202         0.2246

                                Sampling History

                                              RWM
                                       Acceptance
                             Scale           Rate

                            2.8202         0.2191

                               Posterior Summaries

                                          Standard            Percentiles
        Parameter          N        Mean    Deviation        25          50          75

        beta0          40000      0.6312       0.8686    0.0421      0.6164      1.2015
        beta1          40000      1.0933       0.0339    1.0704      1.0934      1.1162
        beta2          40000      1.9143       0.7270    1.4220      1.8949      2.3896
        beta3          40000     -0.0179      0.00691   -0.0224     -0.0178     -0.0132
        beta4          40000      1.2266       0.4511    0.9251      1.2231      1.5262

                               Posterior Intervals

                Parameter   Alpha     Equal-Tail Interval        HPD Interval

                beta0       0.050     -1.0276      2.4020     -1.0793      2.3399
                beta1       0.050      1.0264      1.1596      1.0262      1.1593
                beta2       0.050      0.5326      3.3944      0.4913      3.3473
                beta3       0.050     -0.0322    -0.00502     -0.0313    -0.00417
                beta4       0.050      0.3547      2.1229      0.3901      2.1561

                            Posterior Correlation Matrix

            Parameter        beta0        beta1        beta2        beta3        beta4

            beta0           1.0000      -0.0045       0.2732      -0.9721      -0.1310
            beta1          -0.0045       1.0000       0.0071      -0.0112      -0.0184
            beta2           0.2732       0.0071       1.0000      -0.3419       0.0331
            beta3          -0.9721      -0.0112      -0.3419       1.0000       0.0316
            beta4          -0.1310      -0.0184       0.0331       0.0316       1.0000
```



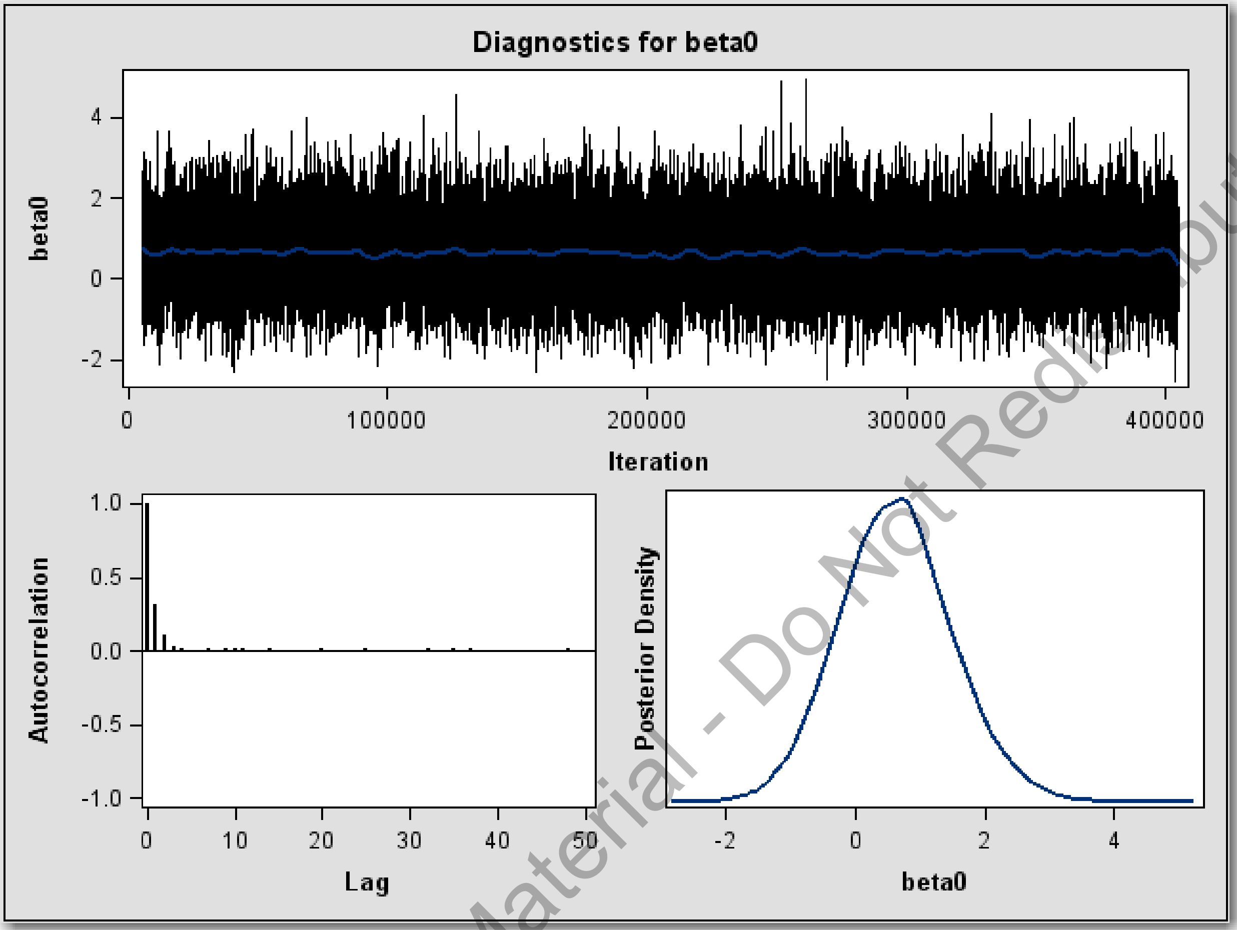
```
                           Posterior Covariance Matrix

    Parameter         beta0         beta1         beta2         beta3         beta4

    beta0            0.7544      -0.00013        0.1725      -0.00583       -0.0513
    beta1          -0.00013       0.00115      0.000175       -2.62E-6      -0.00028
    beta2            0.1725      0.000175        0.5285      -0.00172        0.0109
    beta3          -0.00583       -2.62E-6     -0.00172      0.000048      0.000099
    beta4           -0.0513      -0.00028        0.0109      0.000099        0.2035

                           Monte Carlo Standard Errors

                                                Standard
                     Parameter        MCSE    Deviation     MCSE/SD

                     beta0         0.00596       0.8686     0.00687
                     beta1        0.000235       0.0339     0.00692
                     beta2         0.00500       0.7270     0.00687
                     beta3        0.000048      0.00691     0.00696
                     beta4         0.00312       0.4511     0.00692

                           Posterior Autocorrelations

                 Parameter      Lag 1      Lag 5     Lag 10     Lag 50

                 beta0         0.3106    -0.0009     0.0149    -0.0048
                 beta1         0.3119    -0.0041     0.0036    -0.0178
                 beta2         0.3112     0.0032     0.0063     0.0011
                 beta3         0.3156     0.0027     0.0137    -0.0071
                 beta4         0.3145     0.0088     0.0044    -0.0032

                               Geweke Diagnostics

                           Parameter          z    Pr > |z|

                           beta0         1.4442      0.1487
                           beta1         0.5701      0.5686
                           beta2         0.8482      0.3963
                           beta3        -1.4684      0.1420
                           beta4         0.2386      0.8114

                            Raftery-Lewis Diagnostics
              Quantile=0.025 Accuracy=+/-0.005 Probability=0.95 Epsilon=0.001

                                  Number of Samples          Dependence
               Parameter     Burn-In        Total     Minimum        Factor

               beta0               4         4862        3746        1.2979
               beta1               4         4762        3746        1.2712
               beta2               4         4822        3746        1.2872
               beta3               4         5210        3746        1.3908
               beta4               4         4923        3746        1.3142
```



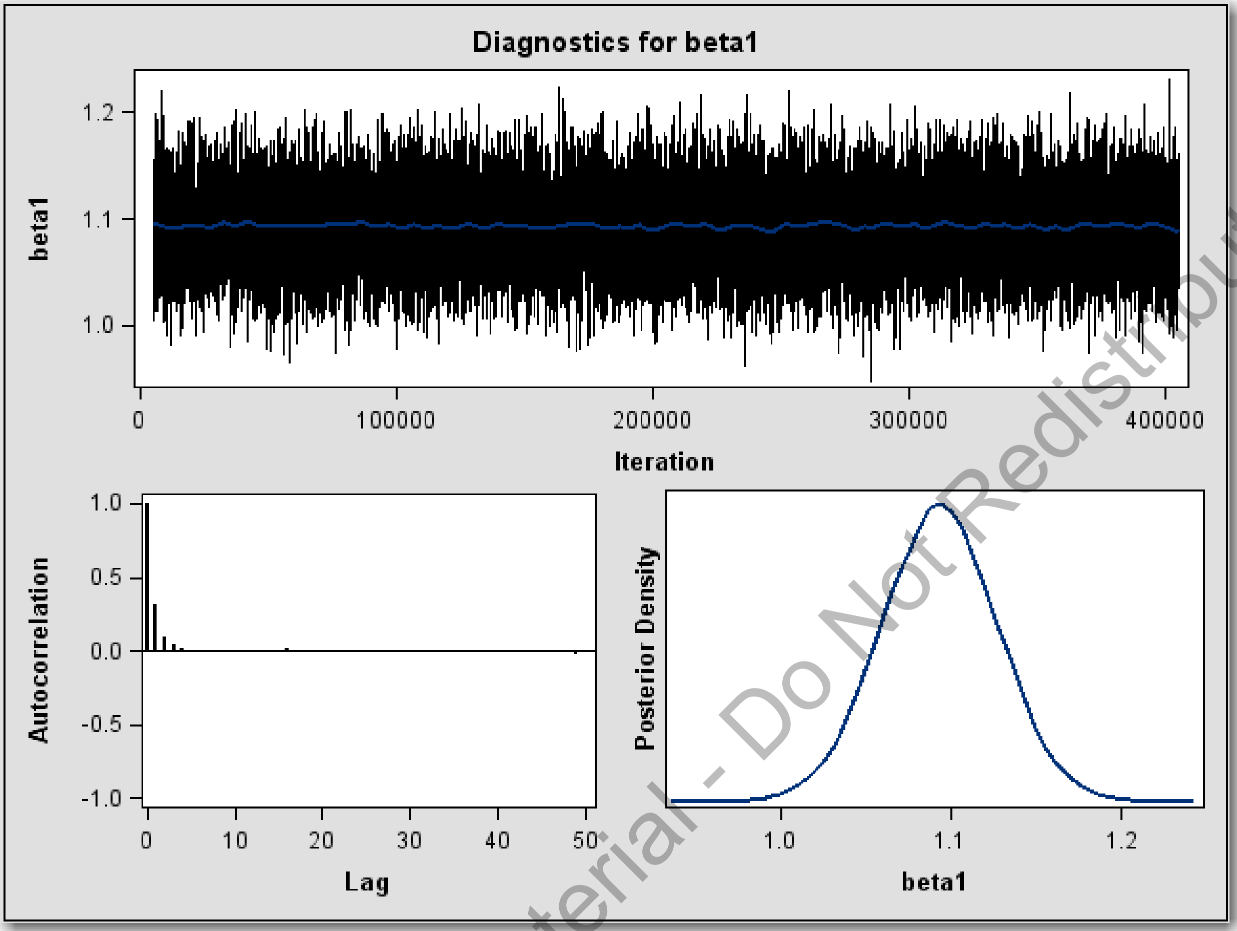
```
                                  Heidelberger-Welch Diagnostics

                         Stationarity Test                              Half-Width Test
           Cramer-von              Test      Iterations     Half-              Relative  Test
Parameter  Mises Stat  p-Value  Outcome    Discarded      Width       Mean  Half-Width  Outcome

beta0          0.1818   0.3057  Passed             0     0.0114     0.6312      0.0181  Passed
beta1          0.0695   0.7547  Passed             0   0.000397     1.0933    0.000363  Passed
beta2          0.2027   0.2623  Passed             0    0.00901     1.9143     0.00471  Passed
beta3          0.2320   0.2132  Passed             0   0.000093    -0.0179    -0.00518  Passed
beta4          0.3213   0.1179  Passed             0    0.00554     1.2266     0.00451  Passed

                                      Effective Sample Sizes

                                               Autocorrelation
                     Parameter          ESS               Time    Efficiency

                     beta0          21212.4             1.8857        0.5303
                     beta1          20909.7             1.9130        0.5227
                     beta2          21169.7             1.8895        0.5292
                     beta3          20665.3             1.9356        0.5166
                     beta4          20863.2             1.9172        0.5216

                                 Deviance Information Criterion

                   Dbar (posterior mean of deviance)                  214.749
                   Dmean (deviance evaluated at posterior mean)       210.691
                   pD (effective number of parameters)                  4.059
                   DIC (smaller is better)                            218.808
```

The diagnostic statistics show no indication that the Markov chain has not reached convergence. The posterior autocorrelations are small, and the Geweke diagnostics are not significant. The DIC statistic is slightly higher than the last model, but this might be due to the informative prior. Usage of the DIC to evaluate a prior is not recommended.

Partial Graphics Output:



The diagnostic plots for **beta0** show no problem with convergence of the Markov chain.

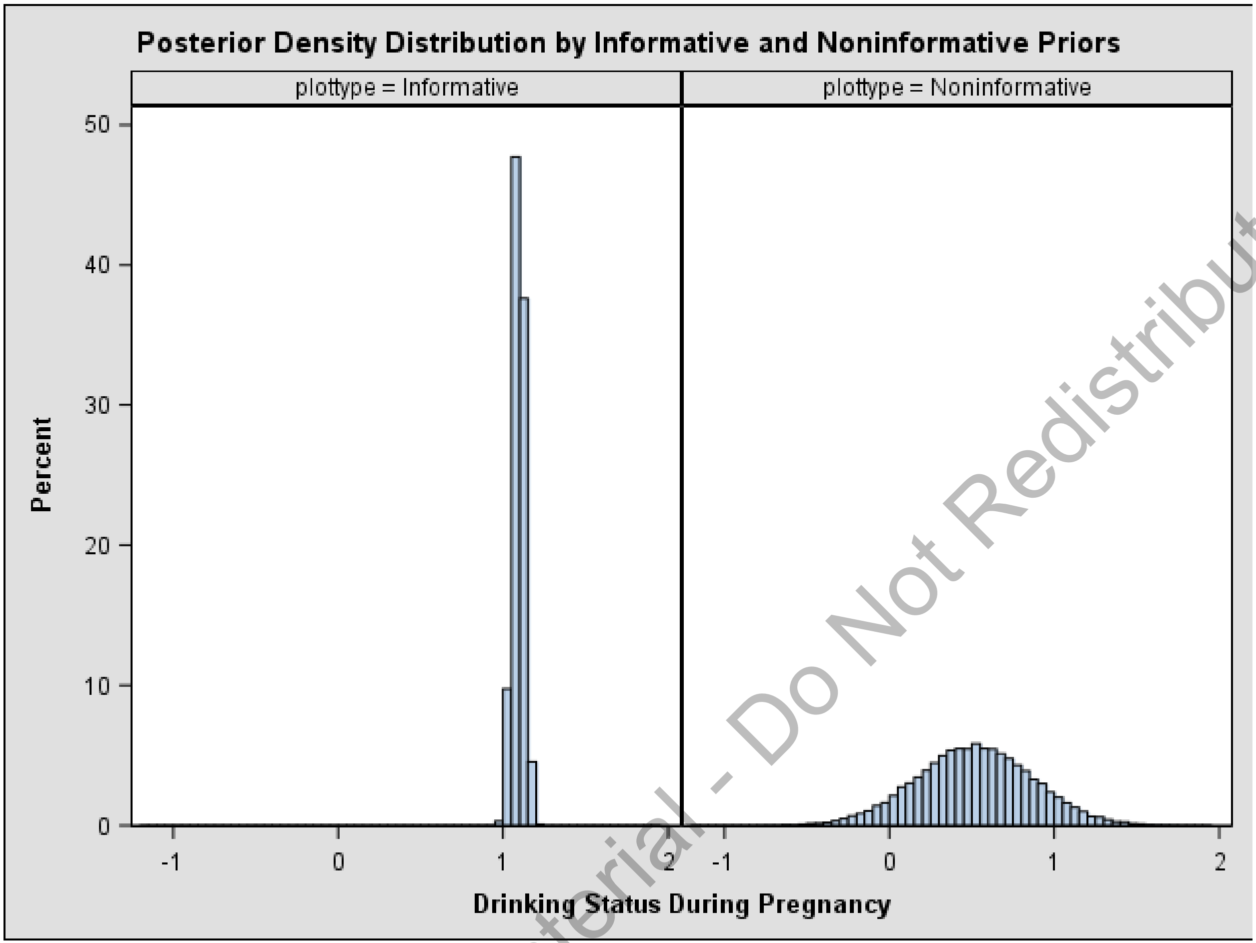


The diagnostic plots for **beta1** show no problem with convergence of the Markov chain.

Note: The remaining diagnostic plots (not shown) show patterns of convergence.

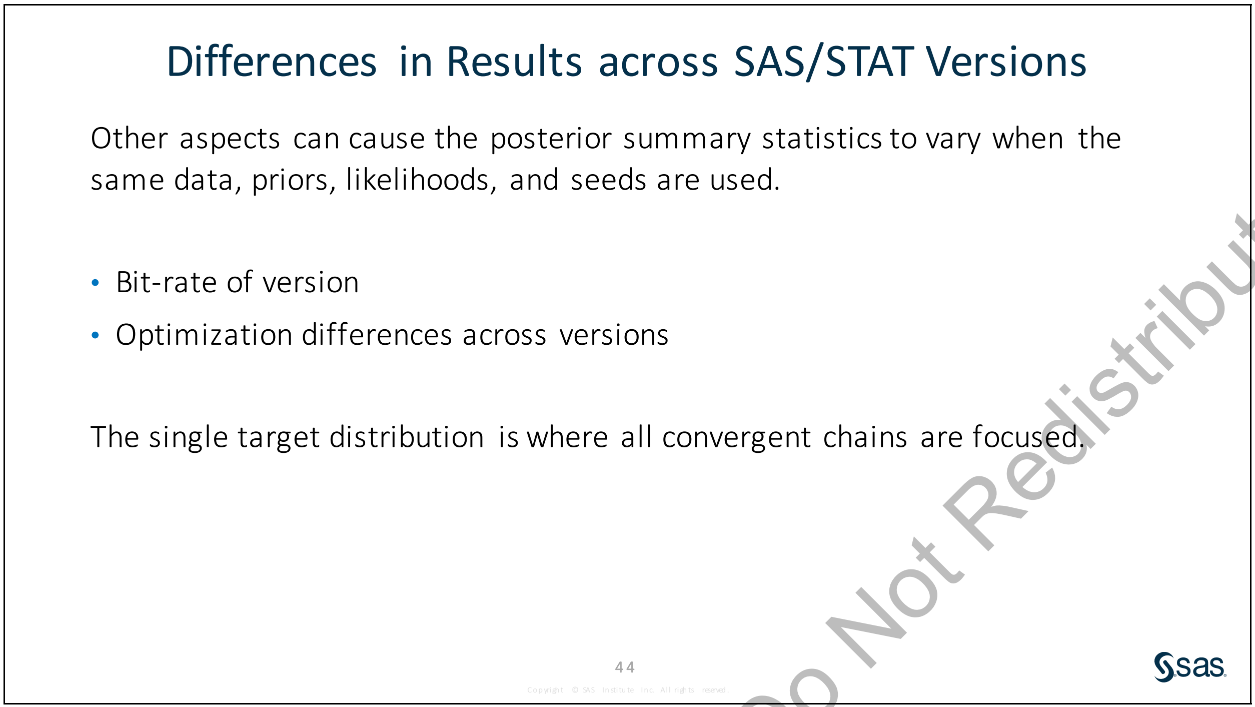
Example: Create side-by-side histograms of the posterior density distributions with one based on the noninformative prior distribution and the other based on the informative prior distribution.

```
data plot;
   length plottype $ 14;
   set birthout birthout1(in=inform);
   if inform=1 then plottype="Informative";
   else plottype="Noninformative";
run;

proc sgpanel data=plot;
   panelby plottype;
   histogram beta1;
   rowaxis label="Percent";
   colaxis label="Drinking Status During Pregnancy";
   title "Posterior Density Distribution by Informative and "
         "Noninformative Priors";
run;
```



The histograms are very similar to the ones produced from the data sets created from PROC GENMOD.

End of Demonstration



Differences in Results across SAS/STAT Versions

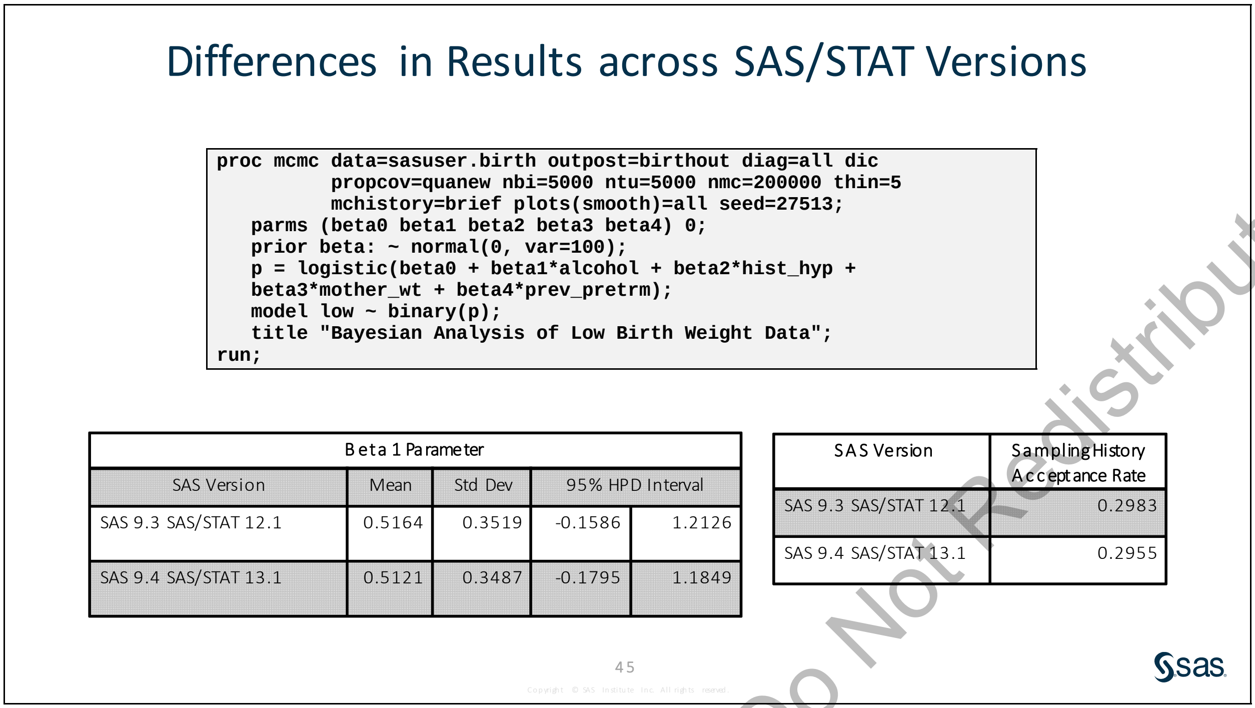
Other aspects can cause the posterior summary statistics to vary when the same data, priors, likelihoods, and seeds are used.

- Bit-rate of version
- Optimization differences across versions

The single target distribution is where all convergent chains are focused.

44

When an analyst is using linear regression, the concept of convergence of parameter estimates is not important. The solving of the normal equations provides a closed-form solution such that everyone arrives at the same answer. If you move to logistic regression, this is no longer the case. Due to the logit transformation, we lose our closed form and must reach a solution by the maximum likelihood method. This is an iterative process where we must achieve convergence to a single point estimate for each parameter. If an analyst were to use a Bayesian approach, convergence is still a necessity. However, recall that a Bayesian statistician is now concerned with the posterior distribution of the parameters in their problem. For this reason, Bayesian's focus on convergence to a distribution rather than a point estimate. Point estimates such as the posterior mean, posterior median, or posterior mode are all highly influenced by the chain of iterations generated and kept by the MCMC algorithm. However, if the diagnostic statistics and plots indicate that the chain has converged, each chain applied to the same priors, likelihood, and data should have converged to the same distribution. This is true even if the chains are not exactly alike.



Differences in Results across SAS/STAT Versions

```
proc mcmc data=sasuser.birth outpost=birthout diag=all dic
          propcov=quanew nbi=5000 ntu=5000 nmc=200000 thin=5
          mchistory=brief plots(smooth)=all seed=27513;
   parms (beta0 beta1 beta2 beta3 beta4) 0;
   prior beta: ~ normal(0, var=100);
   p = logistic(beta0 + beta1*alcohol + beta2*hist_hyp +
   beta3*mother_wt + beta4*prev_pretrm);
   model low ~ binary(p);
   title "Bayesian Analysis of Low Birth Weight Data";
run;
```

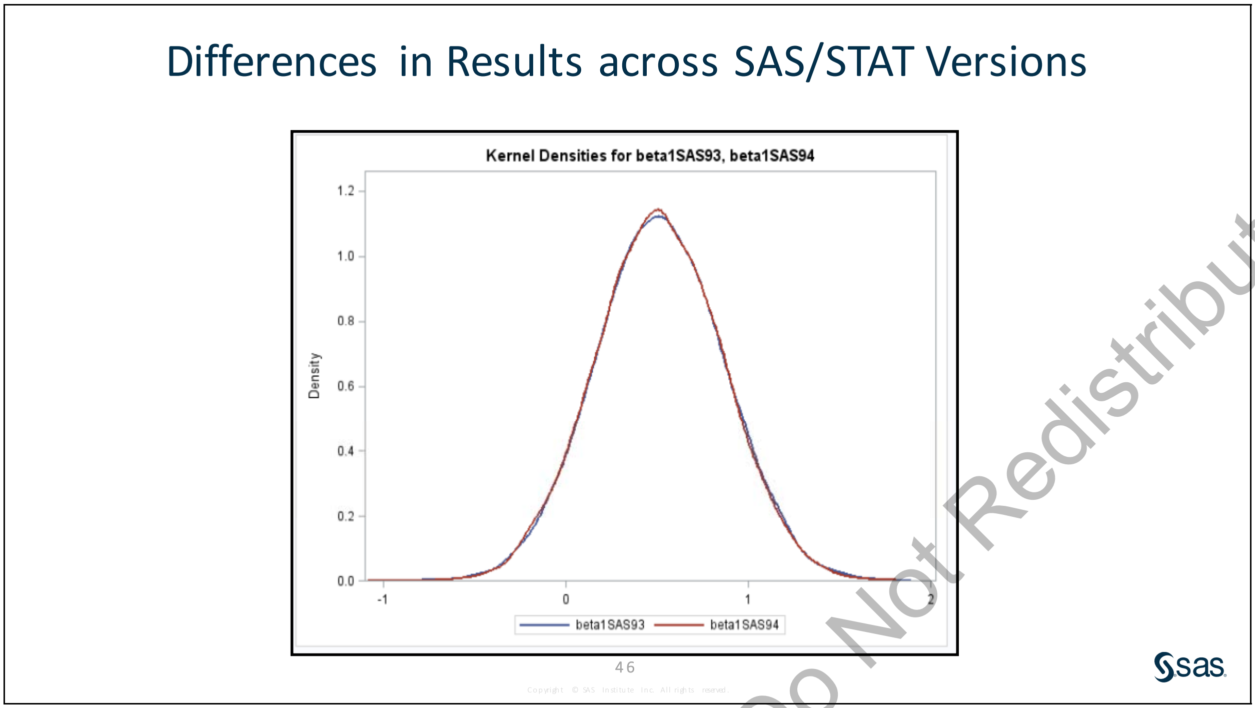
Beta 1 Parameter				
SAS Version	Mean	Std Dev	95% HPD Interval	
SAS 9.3 SAS/STAT 12.1	0.5164	0.3519	-0.1586	1.2126
SAS 9.4 SAS/STAT 13.1	0.5121	0.3487	-0.1795	1.1849

SAS Version	Sampling History Acceptance Rate
SAS 9.3 SAS/STAT 12.1	0.2983
SAS 9.4 SAS/STAT 13.1	0.2955

45

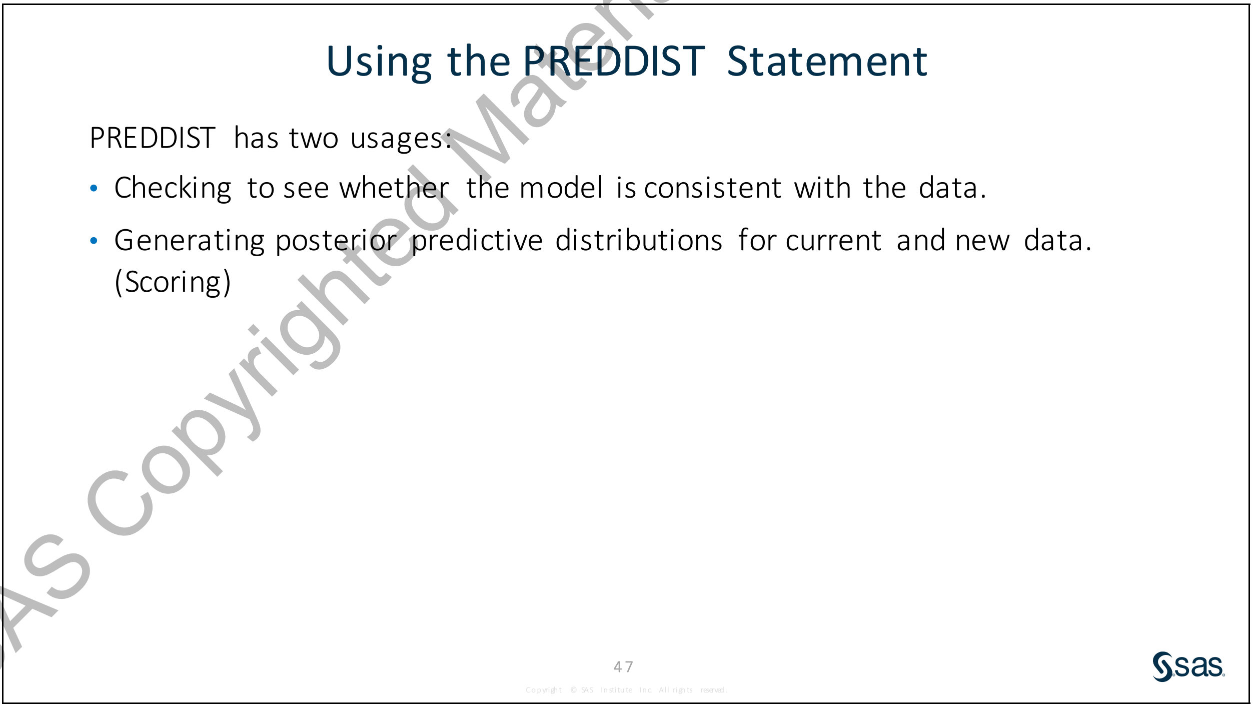
Here is an example of what we mean. The following code is an excerpt from file stbay02d01.sas. This code was executed on different versions of SAS/STAT, 12.1 and 13.1. Notice that the all options within the MCMC procedure are the same including the seed. You would expect that setting this seed would cause the chain to be replicated in both versions of SAS/STAT yielding matching results. However, improvements made to the quasi-newton algorithm in the PROPCOV option will cause a change to the initial values selected when starting the chain. These changes at the start will propagate through the chain causing them to be different.

Notice that the acceptance rate for the sampling history are different as well as the point estimates for the posterior means for the parameters. This is to be expected due to the differences in the chains. But are these changes vastly different? Did we still converge to the same distribution?



Differences in Results across SAS/STAT Versions

Yes we did! The image shows the kernel density estimates for the beta1 term from both runs. Notice that these kernel densities are essentially the same. Despite the differences in the chain, we still converged to the same distribution because we were using the same priors, likelihoods, and data.



Using the PREDDIST Statement

PREDDIST has two usages:

- Checking to see whether the model is consistent with the data.
- Generating posterior predictive distributions for current and new data. (Scoring)

47
Copyright © SAS Institute Inc. All rights reserved.
sas

The upcoming demonstration will present both usages of the PREDDIST statement.

Using PREDDIST in PROC MCMC

Example: Fit the same model with the informative prior as before, but create a data set with a posterior predictive distribution and a data set of the posterior summaries of the predicted values of the response variable. Do not create any diagnostic plots or output. Then create a histogram of the posterior predictive distribution and create a reference line at the mean of the data and a tail-area probability of the test statistics (based on the observed data) with respect to the estimated posterior predictive distribution.

```
/*  stbay02d02.sas  */
ods select none;
proc mcmc data=sasuser.birth plots=none seed=27513
     propcov=quanew nbi=5000 ntu=5000 nmc=400000 thin=10 stats=all;
   parms (beta0 beta1 beta2 beta3 beta4) 0;
   prior beta1 ~ normal(1.0986,var=0.00116);
   prior beta0 beta2 beta3 beta4 ~ normal(0, var=100);
   p=logistic(beta0+beta1*alcohol+beta2*hist_hyp+
              beta3*mother_wt+beta4*prev_pretrm);
   model low ~ binary(p);
   preddist outpred=pred stats=summary;
   ods output predsummaries=prediction_summaries;
run;
ods select all;
```

Selected PROC MCMC statements:

PREDDIST creates a new SAS data set that contains random samples from the posterior predictive distribution of the response variable.

Selected PREDDIST statement options:

OUTPRED= creates an output data set of the predicted samples.

STATS= specifies options for calculating posterior statistics. The summary keyword computes the posterior means, standard deviations, and percentile points for each variable. By default, the 25th, 50th, and 75th percentile points are produced, but you can use the global PERCENT= option to request specific percentile points.

```
proc print data=prediction_summaries(obs=10);
   title "Posterior Summaries of Predicted Values of the Response "
         "Variable";
run;
```

```
              Posterior Summaries of Predicted Values of the Response Variable

 Obs    Parameter           N        Mean      StdDev         P25         P50         P75

   1     low_1          40000      0.6833      0.4652           0      1.0000      1.0000
   2     low_2          40000      0.1550      0.3619           0           0           0
   3     low_3          40000      0.5656      0.4957           0      1.0000      1.0000
   4     low_4          40000      0.8416      0.3651      1.0000      1.0000      1.0000
   5     low_5          40000      0.2937      0.4555           0           0      1.0000
   6     low_6          40000      0.1161      0.3203           0           0           0
   7     low_7          40000      0.2536      0.4351           0           0      1.0000
```

```
       8      low_8           40000       0.3953       0.4889            0            0       1.0000
       9      low_9           40000       0.5376       0.4986            0       1.0000       1.0000
      10      low_10          40000       0.6524       0.4762            0       1.0000       1.0000
```

The **prediction_summaries** data set contains for each observation the posterior mean, standard deviations, and percentile points for the response variable. By default, the 25th, 50th, and 75th percentile points are produced, but you can use the global PERCENT= option to request specific percentile points in the PREDDIST statement. You can also specify the number of simulated predicted values by using the NSIM= option. By default, the number of MCMC iterations is used.

```
proc print data=pred(obs=10);
   var low_1-low_10;
   title "Random Samples from the Posterior Predictive Distribution";
run;
```

```
                 Random Samples from the Posterior Predictive Distribution

  Obs    low_1    low_2    low_3    low_4    low_5    low_6    low_7    low_8    low_9    low_10

    1      1        0        1        1        0        0        1        0        1        1
    2      1        0        1        1        0        1        0        1        1        0
    3      1        1        1        1        0        1        0        0        1        0
    4      1        0        1        1        1        0        1        1        1        1
    5      1        0        1        1        0        0        0        1        1        0
    6      1        0        0        1        0        1        0        0        0        1
    7      1        0        1        1        0        1        0        1        0        1
    8      1        0        1        1        0        0        1        0        0        1
    9      1        0        1        1        0        0        0        1        1        1
   10      0        0        0        1        0        0        1        0        1        1
```

The **pred** data set contains random samples from the posterior predictive distribution of the response variable. The number of variables is equal to the number of observations (in this case, 189).

```
data pred;
   set pred;
   iter_mean=mean(of low:);
run;

proc means data=sasuser.birth noprint;
   var low;
   output out=stat mean=sample_mean;
run;

data _null_;
   set stat;
   call symput('sample_mean',sample_mean);
run;
```

The above statements compute the mean of the response variable from the simulated data and the mean of the response variable from the actual data, and creates a macro variable with the value of the mean from the actual data.

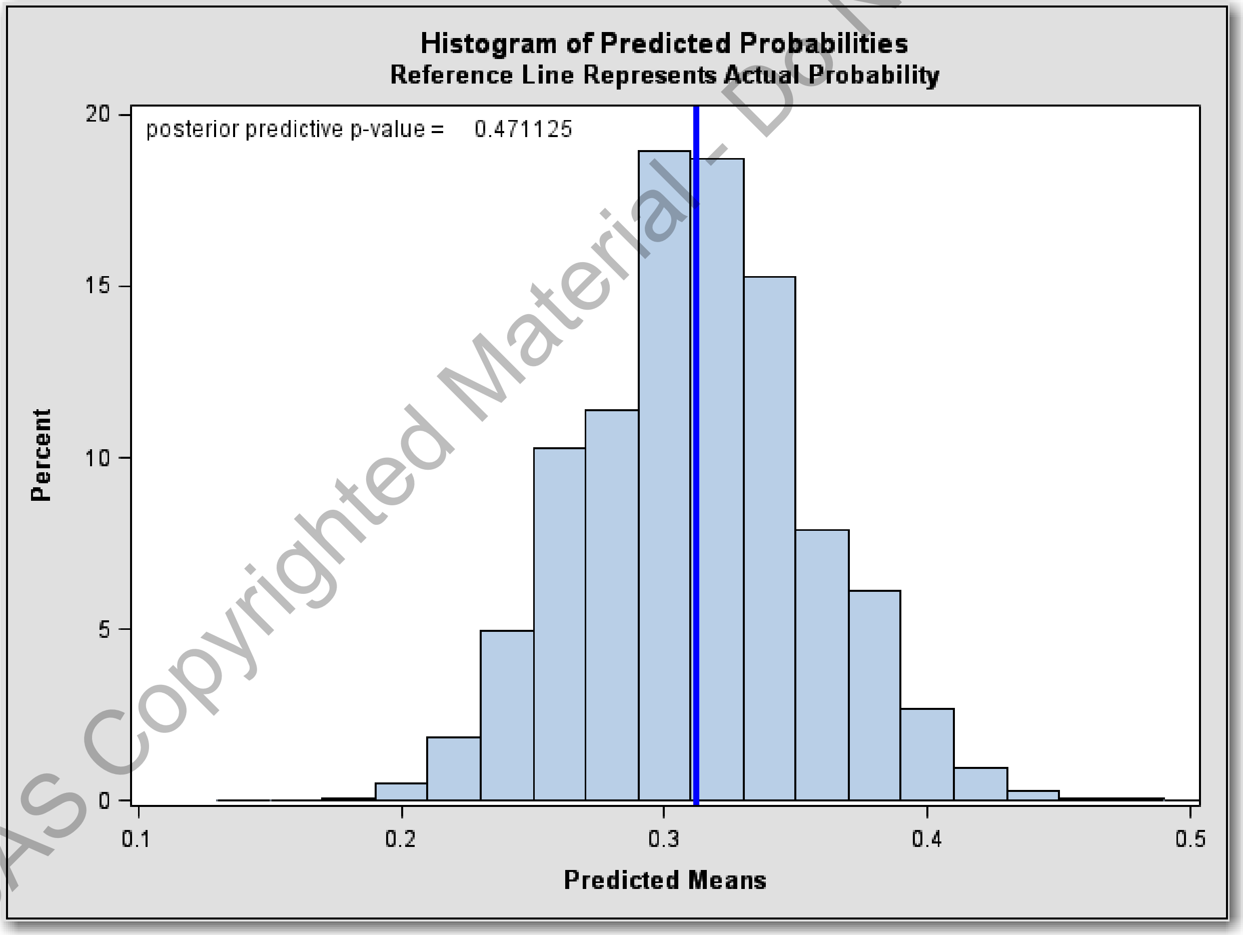
```
data _null_;
   set pred end=eof nobs=nobs;
   ctmean+(iter_mean>&sample_mean);
   if eof then do;
```

```
      pmean=ctmean/nobs;
      call symput('pmean',pmean);
   end;
run;
```

The tail-area probabilities are computed by counting the number of samples in the data set **pred** that are greater than the observed test statistic (the mean) based on the actual data. This quantity is the posterior *p*-value.

```
proc sgplot data=pred;
   histogram iter_mean / nbins=20;
   refline &sample_mean / axis=x lineattrs=(color=blue thickness=3);
   xaxis values=(.1 to .5 by .1) label="Predicted Means";
   inset "posterior predictive p-value=&pmean";
   title1 "Histogram of Predicted Probabilities";
   title2 "Reference Line Represents Actual Probability";
run;
```

The reference line is drawn at the mean of the data and the INSET statement puts the posterior *p*-value in the graph.



The reference line is in the middle of the histogram and the posterior predictive *p*-value is not extreme. This supports the notion that the predicted results are similar to the actual observations and that the model fits the data.

Example: Fit the same model with the informative prior as before, but create a data set that contains random samples from the posterior predictive distribution from a new data set with observations with no response values. Create a data set of the posterior summaries of the predicted values of the response variable and do not create any diagnostic plots or output.

```
data new_birth;
   input mother_wt alcohol prev_pretrm hist_hyp;
datalines;
 125  1  1  0
 130  0  0  0
 187  1  0  0
 175  0  1  1
 185  0  0  0
;
```

The **new_birth** data set has observations for which the predictions are established. Notice there is no response variable in this data set.

```
ods select none;
proc mcmc data=sasuser.birth plots=none seed=27513
     propcov=quanew nbi=5000 ntu=5000 nmc=200000 thin=5;
   parms (beta0 beta1 beta2 beta3 beta4) 0;
   prior beta1 ~ normal(1.0986,var=0.00116);
   prior beta0 beta2 beta3 beta4 ~ normal(0, var=100);
   p=logistic(beta0+beta1*alcohol+beta2*hist_hyp+
              beta3*mother_wt+beta4*prev_pretrm);
   model low ~ binary(p);
   preddist outpred=scored covariates=new_birth
         stats(percent=50)=summary;
   ods output predsummaries=scored_summaries;
run;
ods select all;
```

Selected PREDDIST statement options:

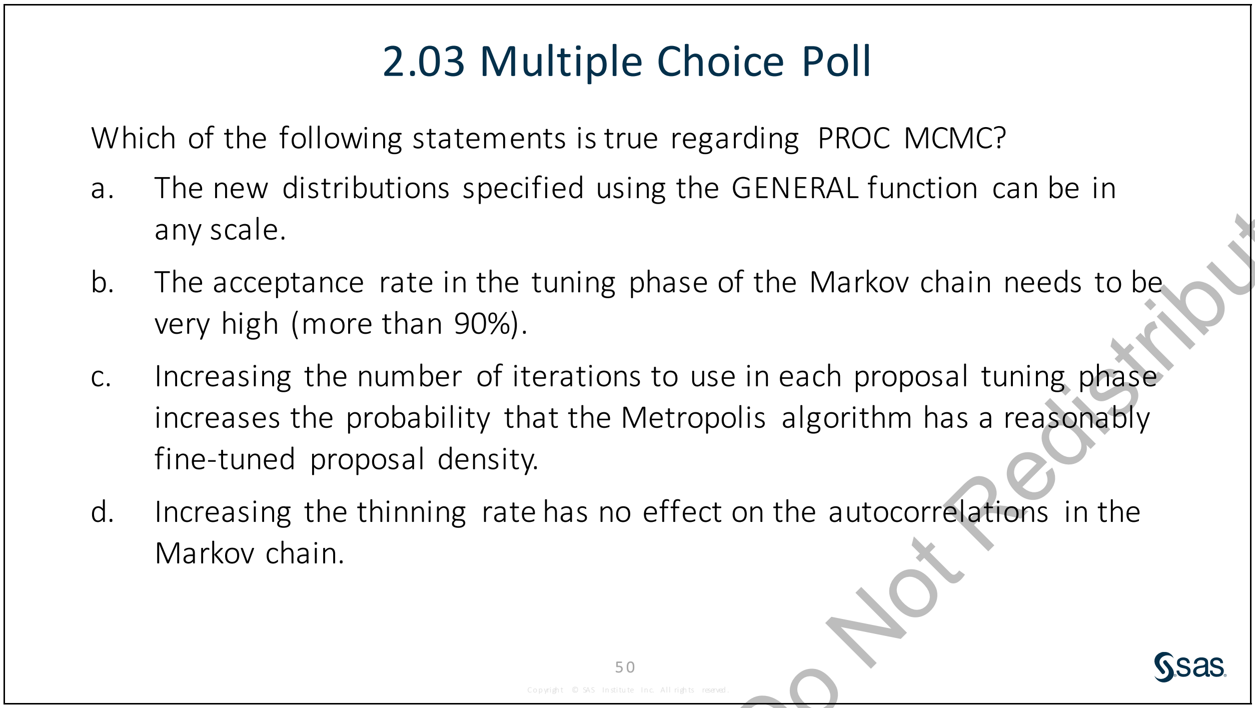
COVARIATES= names the SAS data set that contains the sets of explanatory variable values for which the predictions are established. This data set must contain data with the same variable names as are used in the likelihood function.

```
proc print data=scored_summaries;
   title "Posterior Summaries of Scored Observations";
run;
```

```
                    Posterior Summaries of Scored Observations

          Obs    Parameter         N        Mean      StdDev         P50

            1      low_1       40000      0.6628      0.4728      1.0000
            2      low_2       40000      0.1593      0.3660           0
            3      low_3       40000      0.1731      0.3783           0
            4      low_4       40000      0.6317      0.4823      1.0000
            5      low_5       40000      0.0696      0.2544           0
```

The mean represents the predicted probability of the event for the scored observations.

End of Demonstration



2.03 Multiple Choice Poll

Which of the following statements is true regarding PROC MCMC?

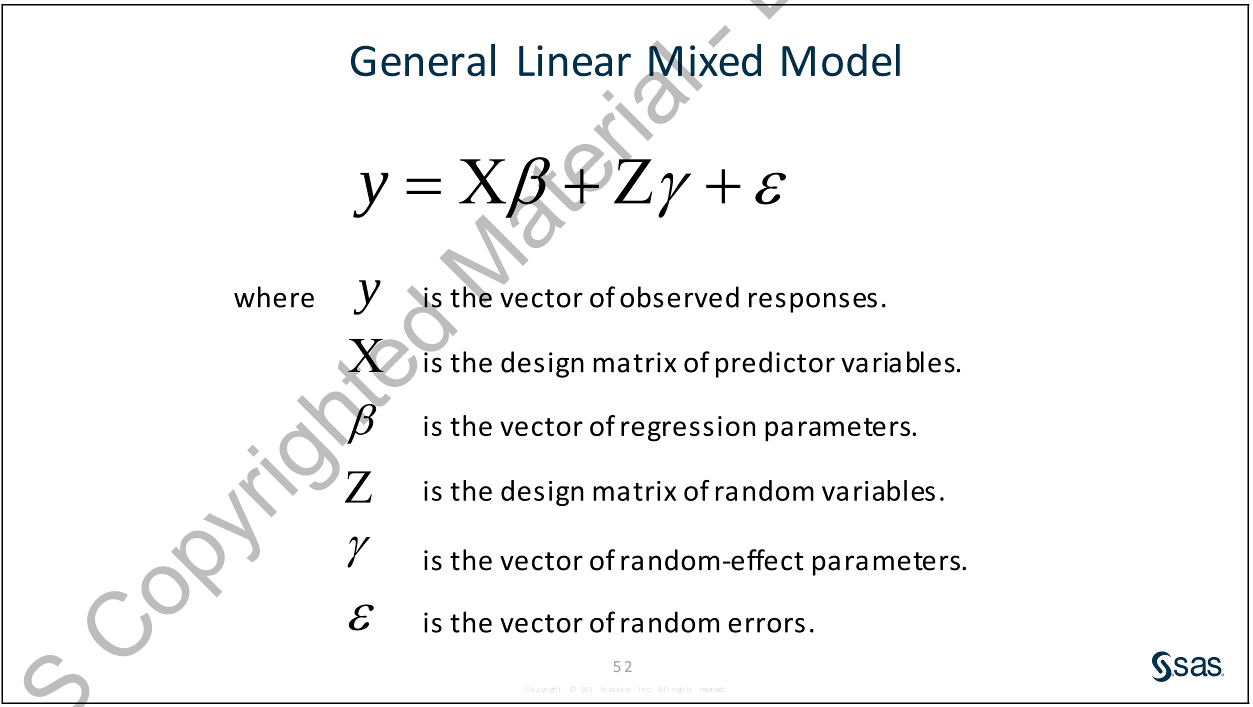
a. The new distributions specified using the GENERAL function can be in any scale.

b. The acceptance rate in the tuning phase of the Markov chain needs to be very high (more than 90%).

c. Increasing the number of iterations to use in each proposal tuning phase increases the probability that the Metropolis algorithm has a reasonably fine-tuned proposal density.

d. Increasing the thinning rate has no effect on the autocorrelations in the Markov chain.

50



General Linear Mixed Model

$$y = X\beta + Z\gamma + \varepsilon$$

where y is the vector of observed responses.

X is the design matrix of predictor variables.

β is the vector of regression parameters.

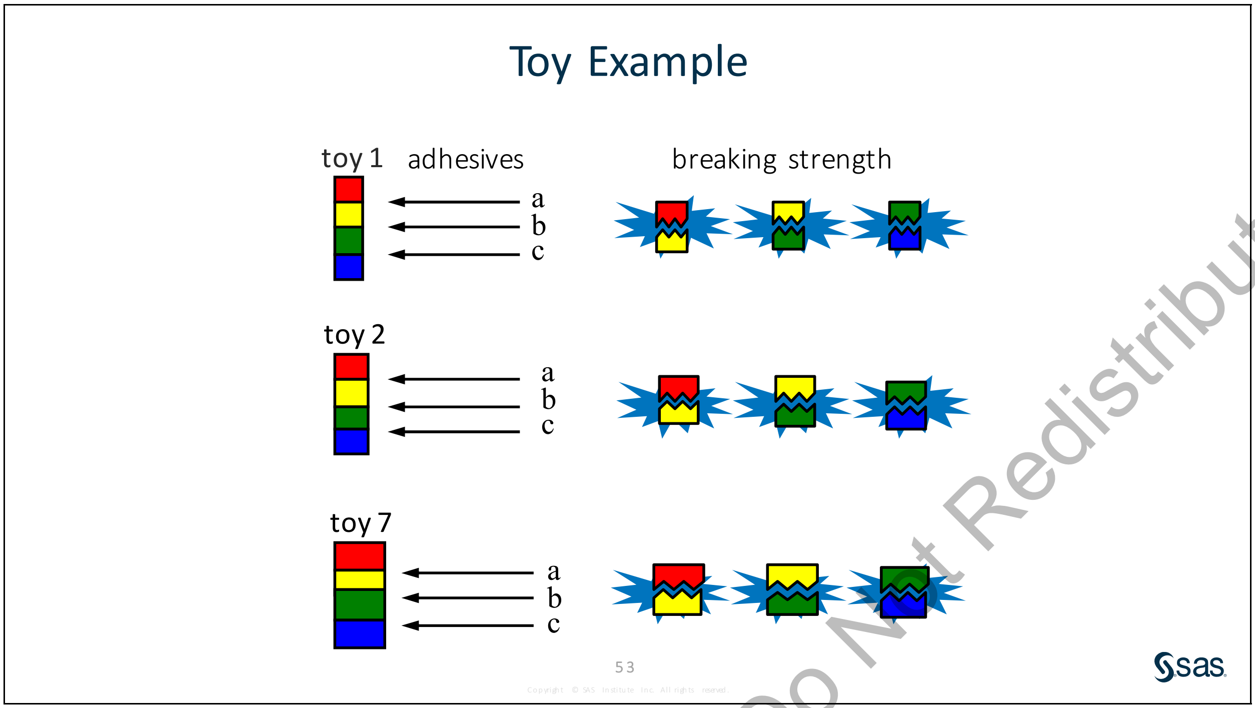
Z is the design matrix of random variables.

γ is the vector of random-effect parameters.

ε is the vector of random errors.

52

The general linear mixed model extends the general linear model with the addition of random effect parameters and by allowing a more flexible specification of the covariance matrix of the random errors. For example, general linear mixed models allow for both correlated error terms and error terms with heterogeneous variances. The matrix **Z** can contain continuous or dummy predictor variables, just like the matrix **X**. The name mixed model indicates that the model contains both fixed-effect parameters and random-effect parameters.



Toy Example

Example: An engineer wants to test the strength of three adhesives used as bonding agents. Seven toys are randomly selected from a population of toys and are used for this strength test. Three different brands of adhesives, a, b, and c, are used to glue parts from each toy. The amount of pressure required to break the bond is then recorded. Data are stored in the SAS data set **sasuser.toy**.

This *randomized complete block (RCB)* design consists of the following effects:

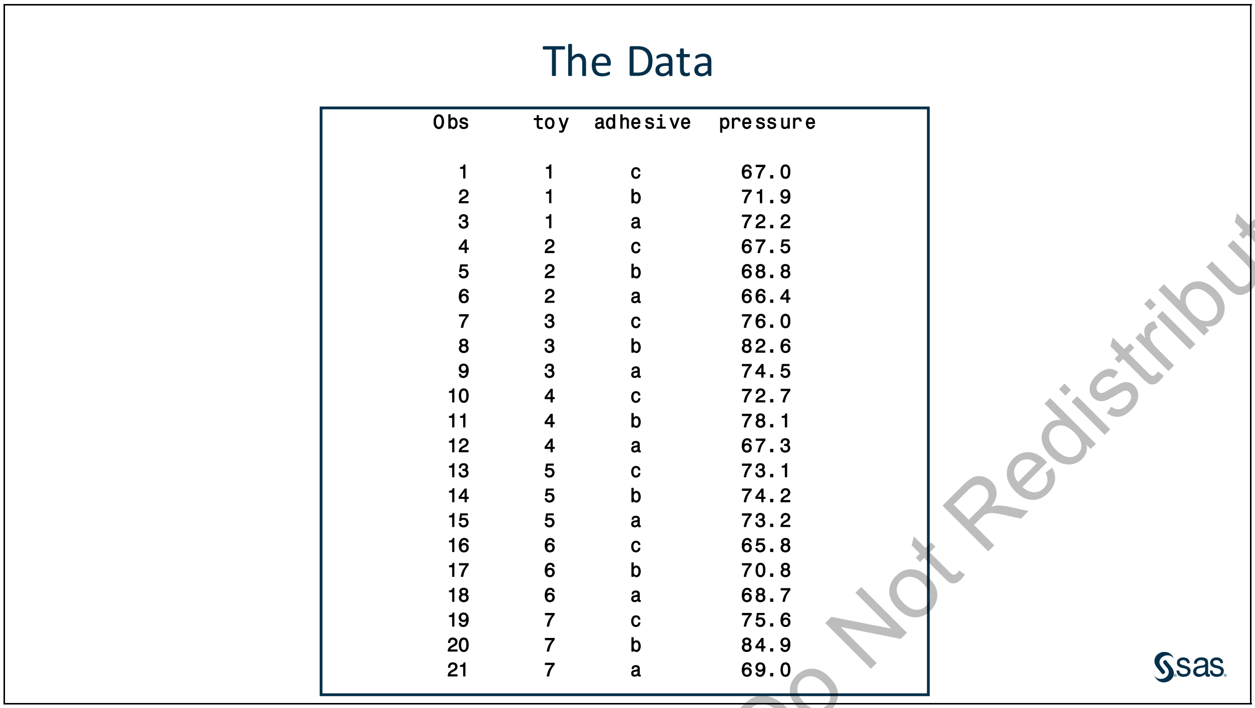
adhesive a treatment effect. This is a **fixed** effect because only three adhesives (**a**, **b**, and **c**) are used in the study, and the engineer is interested only in making inference about these three adhesives.

toy a blocking effect. This is a **random** effect because the seven toys are randomly selected from a population of toys, and the inference about the treatment means is made over the entire population of toys.

Note: The treatments are assumed not to interact with the blocking variable.

The purpose of such an experiment is to

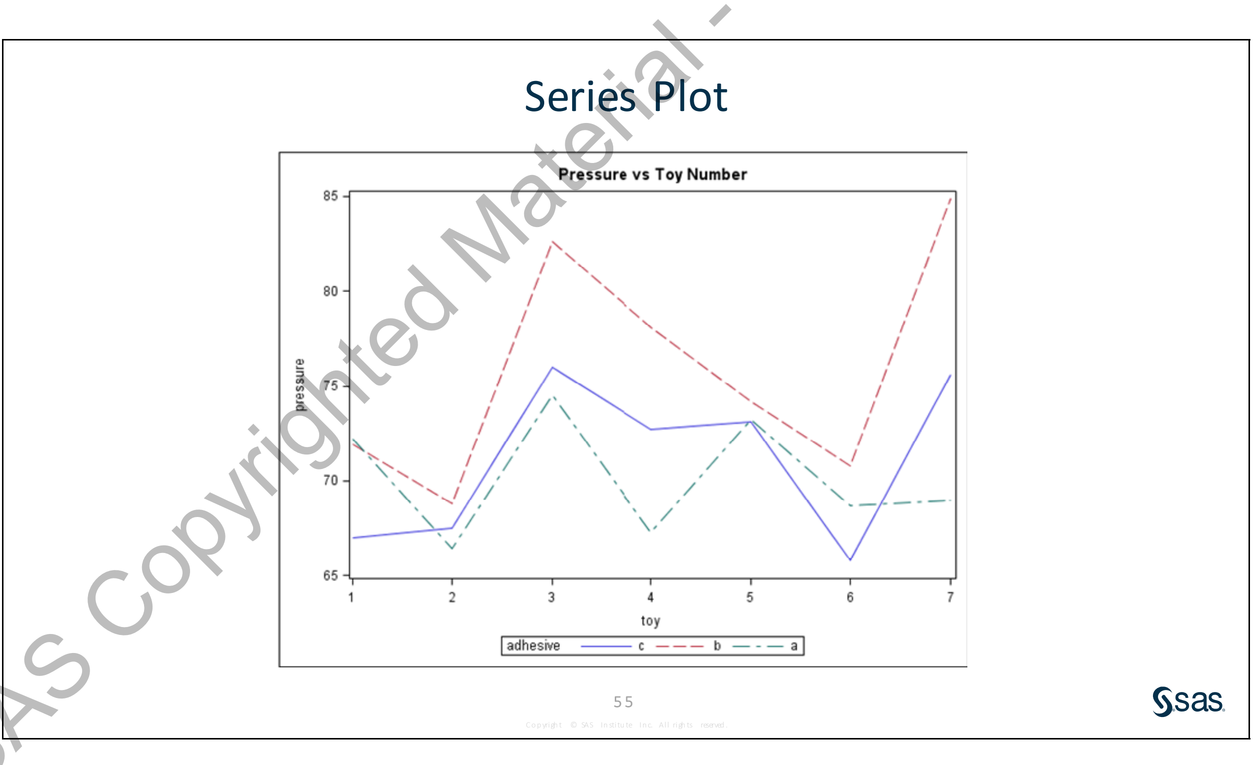
- estimate and compare the treatment means over the entire population of blocks
- account for the variability in the response variable due to the blocks.



The Data

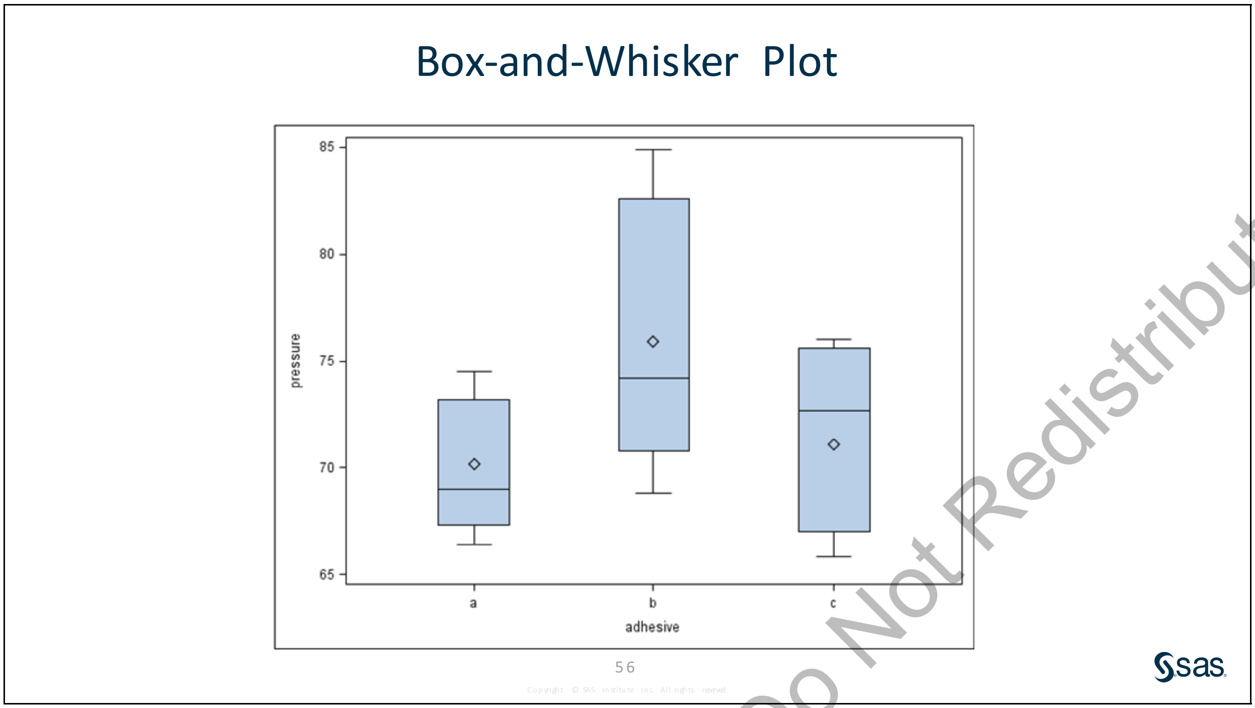
```
 Obs      toy  adhesive  pressure

   1       1      c        67.0
   2       1      b        71.9
   3       1      a        72.2
   4       2      c        67.5
   5       2      b        68.8
   6       2      a        66.4
   7       3      c        76.0
   8       3      b        82.6
   9       3      a        74.5
  10       4      c        72.7
  11       4      b        78.1
  12       4      a        67.3
  13       5      c        73.1
  14       5      b        74.2
  15       5      a        73.2
  16       6      c        65.8
  17       6      b        70.8
  18       6      a        68.7
  19       7      c        75.6
  20       7      b        84.9
  21       7      a        69.0
```

The variables **toy** and **adhesive** are categorical variables. The variable **pressure** is continuous.



Series Plot

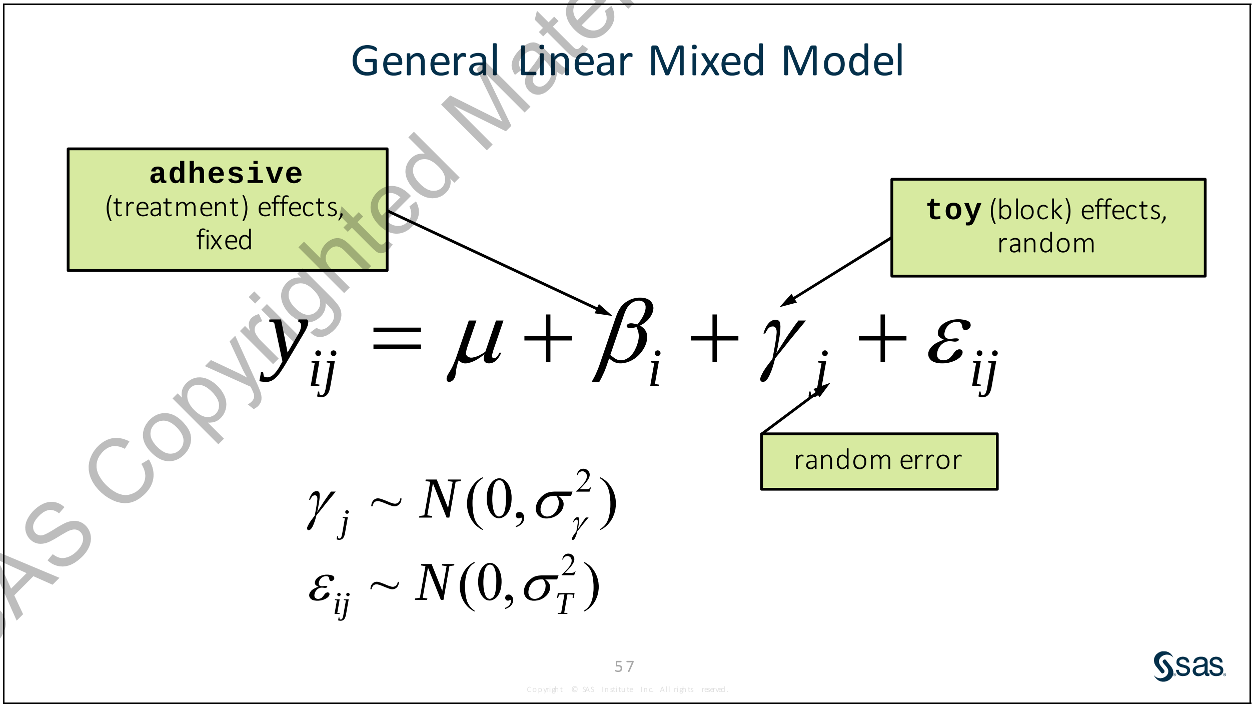
The series plot shows a substantial amount of variability among the seven blocks. In addition, **b** seems to be the strongest adhesive brand in six of the seven toys.



Box-and-Whisker Plot

56

The box-and-whisker plot shows that there are some differences among the three treatment means. The mean for **adhesive b** appears to be bigger than that for the other two adhesives. The variation for **b** seems to be bigger as well. The variability in **pressure** for each adhesive is largely due to the block (**toy**) variability.



General Linear Mixed Model

adhesive (treatment) effects, fixed

toy (block) effects, random

$$y_{ij} = \mu + \beta_i + \gamma_j + \varepsilon_{ij}$$

random error

$$\gamma_j \sim N(0, \sigma_\gamma^2)$$

$$\varepsilon_{ij} \sim N(0, \sigma_T^2)$$

57

To determine whether there is a significant difference in the mean breaking pressure of bonds made using the three adhesives, you can use a general linear mixed model treating the block effect **toy** as a random effect. The model equation is shown above, where:

y_{ij} breaking strength for the i^{th} **adhesive** and j^{th} **toy**, $i=1,\ldots,t$ (treatments), and $j=1,\ldots,r$ (blocks).

μ overall mean.

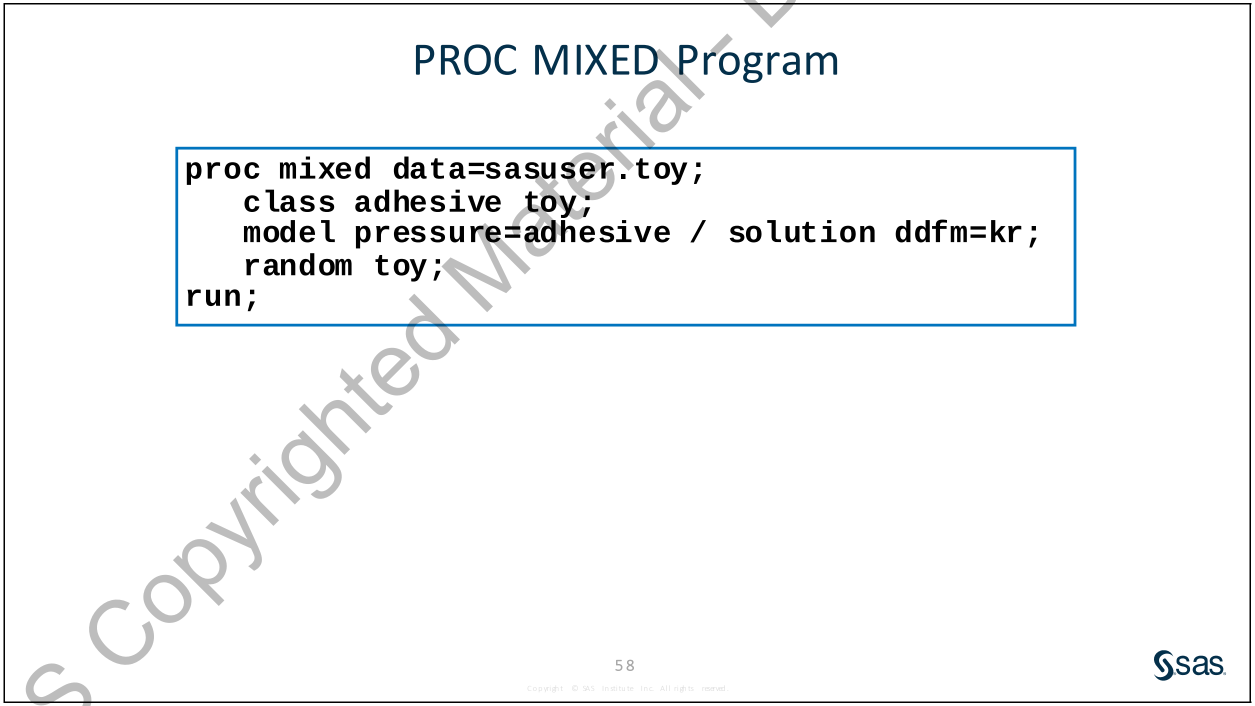
β_i fixed effect associated with the i^{th} **adhesive** (treatment).

γ_j random effect associated with the j^{th} **toy** (block), γ_j ~ i.i.d. N(0, σ_γ^2). These random effects γ_j's are assumed to be independently and normally distributed with mean zero and variance σ_γ^2. The variance σ_γ^2 is the parameter to be estimated in the mixed model for this effect.

ε_{ij} experimental error associated with samples within blocks, ε_{ij}~ i.i.d. N(0, σ_T^2). The random errors are assumed to follow a normal distribution with mean zero and variance σ_T^2. The variance σ_T^2 is the parameter to be estimated in the mixed model for random error.

The effects γ_j and ε_{ij} are assumed to be independent random variables. Therefore,

- E(y_{ij})=$\mu+\beta_i$ is the mean pressure for **adhesive** *i* averaged across all toys in the population.
- *var*(y_{ij})=$\sigma_\gamma^2+\sigma_T^2$. The variance of an observation is the sum of the variances due to blocks (often referred to as between-block variation) and random errors (within-block variation).



PROC MIXED Program

```
proc mixed data=sasuser.toy;
   class adhesive toy;
   model pressure=adhesive / solution ddfm=kr;
   random toy;
run;
```

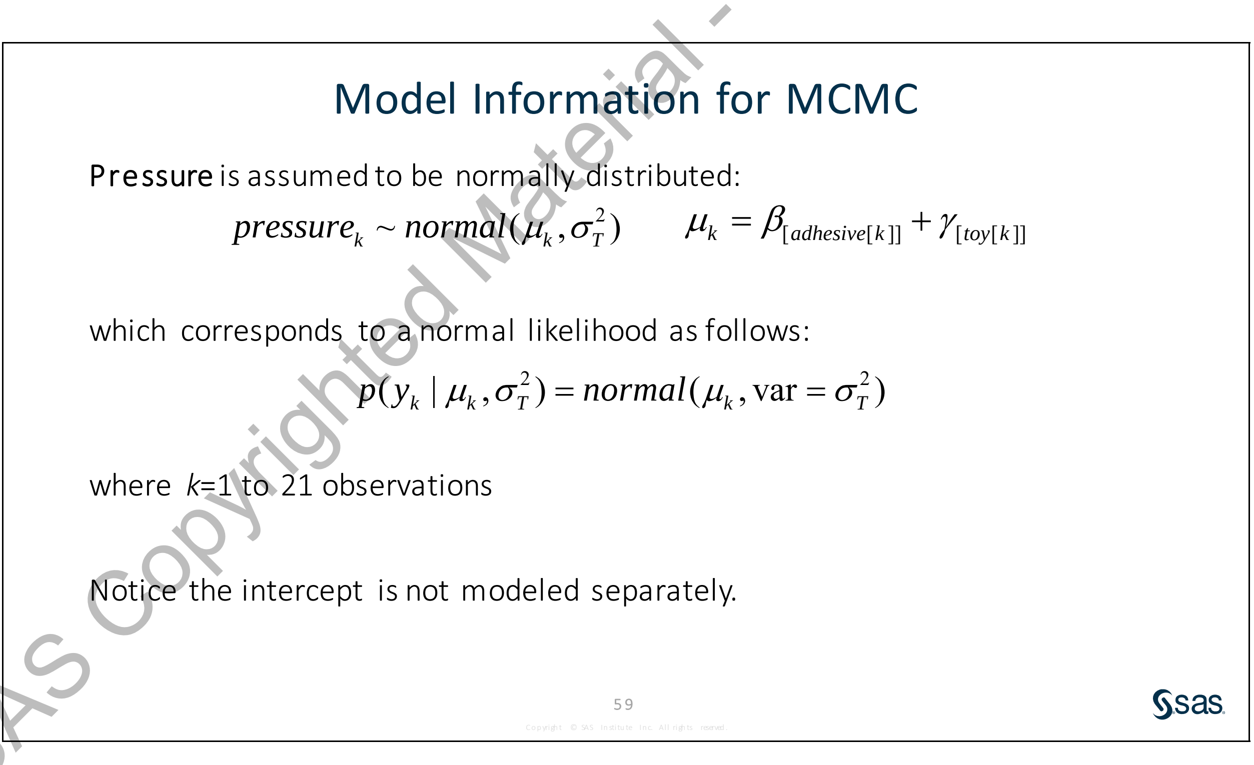
58

You can use the MIXED procedure to analyze the toy data set, treating the block effect **toy** as a random effect. This would be the frequentist approach to analyzing this data.

The output from the Covariance Estimates table and the Parameter Estimates table is shown below:

```
                                    Covariance Parameter
                                          Estimates

                                    Cov Parm      Estimate

                                    toy            11.4478
                                    Residual       10.3716

                                  Solution for Fixed Effects

                                                  Standard
          Effect       adhesive    Estimate          Error       DF    t Value    Pr > |t|

          Intercept                 71.1000         1.7655     11.6      40.27      <.0001
          adhesive     a            -0.9143         1.7214       12      -0.53      0.6050
          adhesive     b             4.8000         1.7214       12       2.79      0.0164
          adhesive     c                  0              .        .          .           .
```

The output shows that the estimated between-block variation is 11.4 while the overall variance is 21.8. The estimated difference between adhesives **a** and **c** is -0.9143 and the estimated difference between adhesives b and c is 4.800. The estimated difference between adhesives **a** and **b** is -5.7143=(-0.9143–4.8000). The estimated mean for adhesive **a** is 70.187=(71.1000–0.9143), the estimated mean for adhesive **b** is 75.900=(71.1000+4.8000), and the estimated mean for adhesive **c** is 71.1000.



Model Information for MCMC

Pressure is assumed to be normally distributed:

$$pressure_k \sim normal(\mu_k, \sigma_T^2) \qquad \mu_k = \beta_{[adhesive[k]]} + \gamma_{[toy[k]]}$$

which corresponds to a normal likelihood as follows:

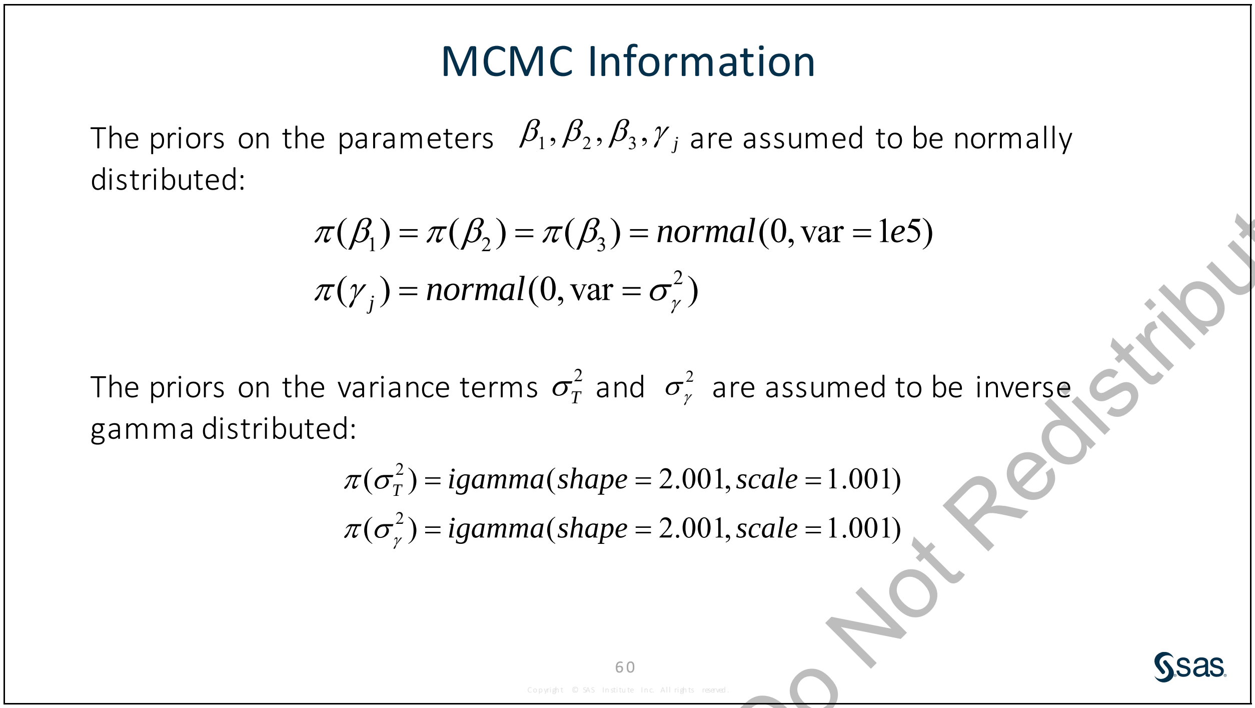
$$p(y_k \mid \mu_k, \sigma_T^2) = normal(\mu_k, \mathrm{var} = \sigma_T^2)$$

where k=1 to 21 observations

Notice the intercept is not modeled separately.

59

PROC MCMC enables you to fit a general linear mixed model, and you can model as many levels of random effects as are needed. The likelihood function for each observation of **pressure**, which is specified in the MODEL statement, is shown above. The above model has **adhesive** recoded as 1 through 3.



MCMC Information

The priors on the parameters $\beta_1, \beta_2, \beta_3, \gamma_j$ are assumed to be normally distributed:

$$\pi(\beta_1) = \pi(\beta_2) = \pi(\beta_3) = normal(0, \text{var} = 1e5)$$
$$\pi(\gamma_j) = normal(0, \text{var} = \sigma_\gamma^2)$$

The priors on the variance terms σ_T^2 and σ_γ^2 are assumed to be inverse gamma distributed:

$$\pi(\sigma_T^2) = igamma(shape = 2.001, scale = 1.001)$$
$$\pi(\sigma_\gamma^2) = igamma(shape = 2.001, scale = 1.001)$$

60

The normal priors on the parameters for the fixed effects have large variances, reflecting the lack of knowledge about the regression coefficients. The normal prior for the random effects has a mean of zero and a variance to be estimated in the mixed model. The priors for the variance terms use the inverse gamma distribution with a shape and scale parameter that reflects the lack of knowledge about the variance coefficients.

Fitting a Mixed Model in PROC MCMC

Example: Fit a general linear mixed model in PROC MCMC for the toy data set. First, create a new variable called **adhesivebeta**, which has values of **1** when **adhesive** is **a**, **2** when **adhesive** is **b**, and **3** when **adhesive** is **c**. In PROC MCMC, use the ARRAY statement to define the fixed effects for **adhesive**, use a RANDOM statement to define the random effect **gamma** for **toy** with a normal prior distribution with a mean of 0 and a variance of s2g, use the NAMESUFFIX=POSITION option in the RANDOM statement to construct the random effect parameter names using position number, and use the BEGINNODATA and ENDNODATA statements to estimate the contrast between the fixed effect parameters of adhesive **a** versus **b**, **a** versus **c**, and **b** versus **c**. Also use the monitor option to monitor the parameters of interest (including the contrasts) and create an output data set for the posterior samples.

```
/* stbay02d03.sas  */
data toy;
   set sasuser.toy;
   if adhesive='a' then adhesivebeta=1;
   if adhesive='b' then adhesivebeta=2;
   if adhesive='c' then adhesivebeta=3;
run;
```

Because PROC MCMC does not have a CLASS statement, the categorical variable for the fixed effect is used in an ARRAY statement with its values corresponding to the elements in the array.

```
proc mcmc data=toy seed=27513 diag=all dic outpost=mixed
     propcov=quanew thin=25 nbi=5000 ntu=5000 nmc=500000
     plots(smooth)=all mchistory=brief stats=all
     monitor=(a_vs_b a_vs_c b_vs_c beta1 beta2 beta3 s2t s2g);
   array beta[3];
   parms beta: 0;
   parms s2t 1;
   parms s2g 1;
   prior beta: ~ normal(0, var=1e5);
   prior s2: ~ igamma(2.001, scale=1.001);
   beginnodata;
        a_vs_b=beta[1]-beta[2];
        a_vs_c=beta[1]-beta[3];
        b_vs_c=beta[2]-beta[3];
   endnodata;
   random gamma ~normal(0,var=s2g) subject=toy monitor=(gamma)
          namesuffix=position;
   mu=beta[adhesivebeta]+gamma;
   model pressure ~ normal(mu, var=s2t);
   title "Bayesian Analysis of the Toy Data Set";
run;
```

The values of the PROC MCMC options were chosen based on trial and error trying to find a good mixing of the Markov chain. The MONITOR= option exhibits analysis for selected parameters of interest in the program. PROC MCMC performs only posterior analyses (such as plotting, diagnostics, and summaries) on the parameters selected with the MONITOR= option.

Note: Within the MONITOR= option, all parameters defined with the PARMS statements can be easily included using the _PARMS_ keyword. The wildcard variable reference operand, beta:, is not permitted within the MONITOR= option. However, the variable range operand, beta1-beta3, is permitted.

The ARRAY statements define a one-dimensional array **beta**, with 3 elements. You can refer to the array elements with variable names (**beta1** to **beta3** by default) or with subscripts, such as **beta**[3]. To indicate subscripts, you must use either brackets [ ] or braces { }, but not parentheses ( ). Note that this is different from the way subscripts are indicated in the DATA step.

The PRIOR statement notation **beta***:* is shorthand for all symbols that start with the letters 'beta'. In this example, there are 3 **beta** parameters. Similarly, **s2***:* stands for both *s2t* (total variance of the likelihood function referred to in PROC MIXED as the residual variance) and *s2g* (random effects variance).

The BEGINNODATA and ENDNODATA statements are designed to reduce unnecessary observation-level computations. They jointly define a block, and the enclosed programming statements are not executed for every data set observation. They are used to calculate the parameters that estimate the contrast between adhesive **a** and **b**, between **a** and **c**, and between **b** and **c**.

The RANDOM statement defines the random effect **gamma** and its prior distribution (normal with a mean of 0 and a variance of s2g). The SUBJECT= option identifies the subjects in the random-effects model (in this example, toy). The SUBJECT= variable can be either a numeric variable or character literal, and it does not need to be sorted. The MONITOR= option exhibits analysis for selected random-effects parameters. You can choose either to monitor all random-effects parameters by specifying MONITOR=(*random effect*) or to monitor a subset of the parameters by specifying a variable list. The NAMESUFFIX= option specifies how the names of the random-effects parameters are internally created.

PROC MCMC creates an output data set with all the posterior samples of random effect parameters with the OUTPOST= option.

The **mu** assignment statement calculates the expected value of **pressure** in the mixed model. The symbol **adhesivebeta** is a data set variable that indexes **adhesivebeta**. In this example, **beta**[**adhesivebeta**] is the fixed effect for the value of **adhesivebeta**.

The MODEL statement specifies the likelihood function for **pressure**.

Partial Output:

```
                        Bayesian Analysis of the Toy Data Set

                                  The MCMC Procedure

                        Number of Observations Read          21
                        Number of Observations Used          21

                                      Parameters

                             Sampling         Initial
   Block    Parameter     Method             Value    Prior Distribution

       1    beta1         N-Metropolis           0    normal(0, var = 1e5)
            beta2                                0    normal(0, var = 1e5)
            beta3                                0    normal(0, var = 1e5)
       2    s2t           Conjugate         1.0000    igamma(2.001, scale = 1.001)
```

```
         3    s2g           Conjugate        1.0000    igamma(2.001, scale = 1.001)
```

When the prior probability distribution and the posterior probability distribution are in the same distributional family (they are conjugate distributions), it is possible to obtain closed-form solutions for the posterior distribution. If PROC MCMC can detect conjugacy, then the procedure uses conjugate sampling methods to draw conditional posterior samples. Conjugate sampling is efficient because it enables the Markov chain to obtain samples from the target distribution directly. In this example, conjugate sampling methods were used to estimate the two variance parameters.

As was stated before, PROC MCMC can detect conjugacy only if the model parameter (not a function or a transformation of the model parameter) is used in the prior and family distributions (a distribution that is conditional on the parameter of interest). In most cases, the family distribution is the likelihood function.

```
                                   Random Effect Parameters

          Sampling              Number of   Subject                    Prior
 Parameter Method        Subject  Subjects   Values                     Distribution

 gamma     N-Metropolis      toy         7   1 2 3 4 5 6 7              normal(0,var=s2g)
```

The Random Effects Parameter table shows the number of levels of the random effect and its prior distribution.

```
                                         Tuning History

                                      RWM Scale      RWM Acceptance Rate
                            Phase       Low     High       Low     High

                                1     2.380    2.380    0.318    0.471
                                2     2.839    3.928    0.181    0.286

                                         Burn-In History

                                   RWM Scale      RWM Acceptance Rate
                                  Low     High       Low     High

                                2.839    3.928    0.176    0.283

                                        Sampling History

                                   RWM Scale      RWM Acceptance Rate
                                  Low     High       Low     High

                                2.839    3.928    0.182    0.287
```

The Tuning History, Burn-In History, and Sampling History tables show the range of scales and the range of the acceptance rates for each random walk Metropolis block. Roberts and Rosenthal (2001) empirically demonstrated that acceptance rates between 0.15 and 0.50 are at least 80% efficient, and the acceptance rates for this model are in that range.

```
                                      Posterior Summaries

                                               Standard               Percentiles
       Parameter            N         Mean     Deviation           25           50           75

       a_vs_b           20000      -5.6952        2.0286      -6.9840      -5.7054      -4.4066
       a_vs_c           20000      -0.8953        2.0197      -2.1683      -0.9070       0.3831
       b_vs_c           20000       4.7998        2.0141       3.5206       4.7995       6.0783
```

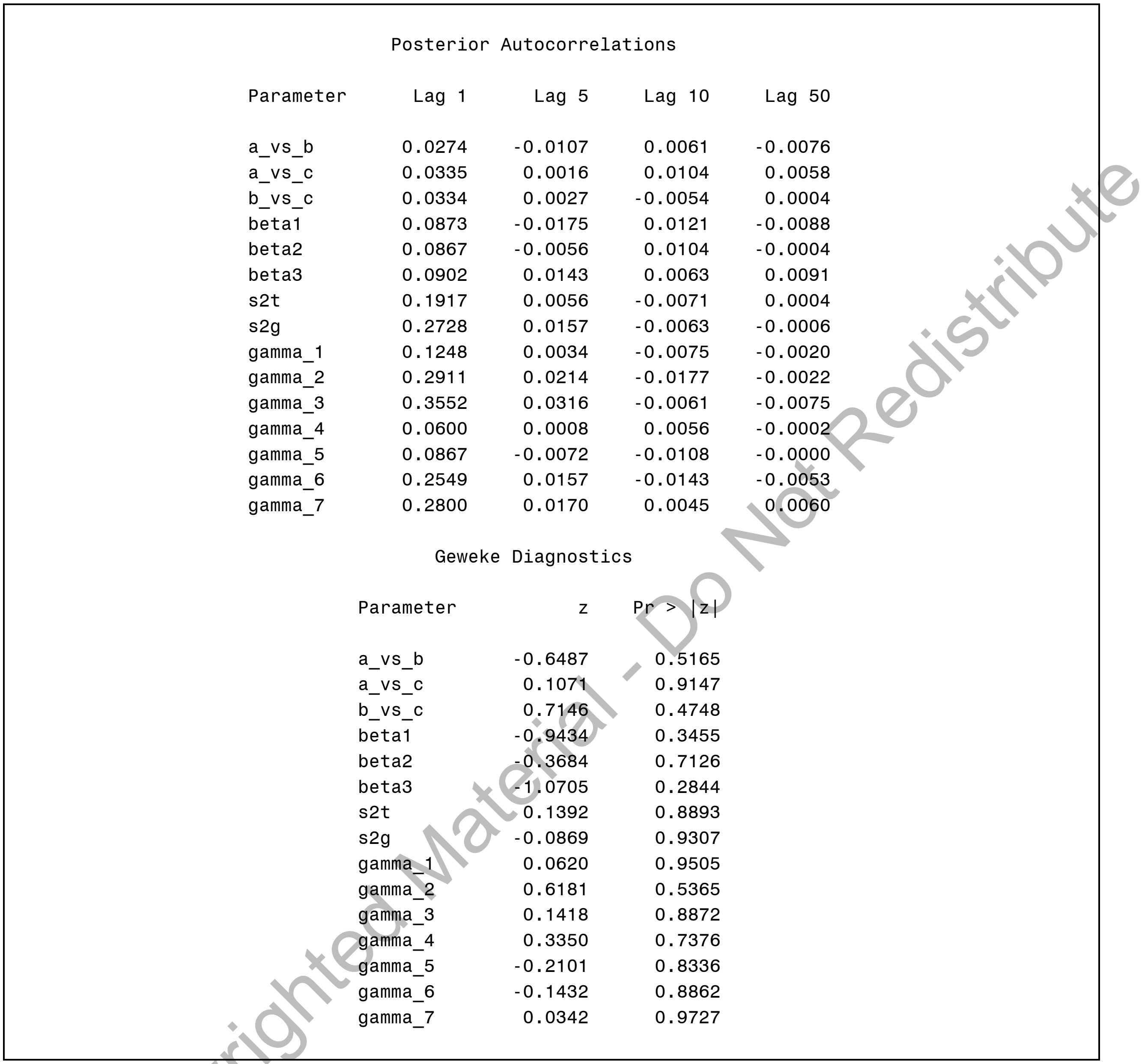
```
        beta1          20000     70.1879        1.5808     69.1521     70.1894     71.2130
        beta2          20000     75.8831        1.5699     74.8354     75.8939     76.9161
        beta3          20000     71.0832        1.5810     70.0584     71.0798     72.1224
        s2t            20000     14.2274        6.4917      9.5105     13.1112     17.5082
        s2g            20000      3.2025        4.1222      0.6552      1.6222      4.2575
        gamma_1        20000     -0.6930        1.3214     -1.3873     -0.5176      0.1483
        gamma_2        20000     -1.6253        1.7441     -2.6478     -1.2370     -0.3520
        gamma_3        20000      1.8119        1.8488      0.4219      1.3775      2.9505
        gamma_4        20000      0.1061        1.2233     -0.5954      0.0866      0.7826
        gamma_5        20000      0.3893        1.2629     -0.3810      0.2873      1.0563
        gamma_6        20000     -1.3482        1.6077     -2.2698     -1.0147     -0.2054
        gamma_7        20000      1.3874        1.6360      0.2131      1.0485      2.3208
```

The results of the posterior summaries for the parameter estimates are very similar to the results from PROC MIXED. However, the variance estimates are different in PROC MCMC.

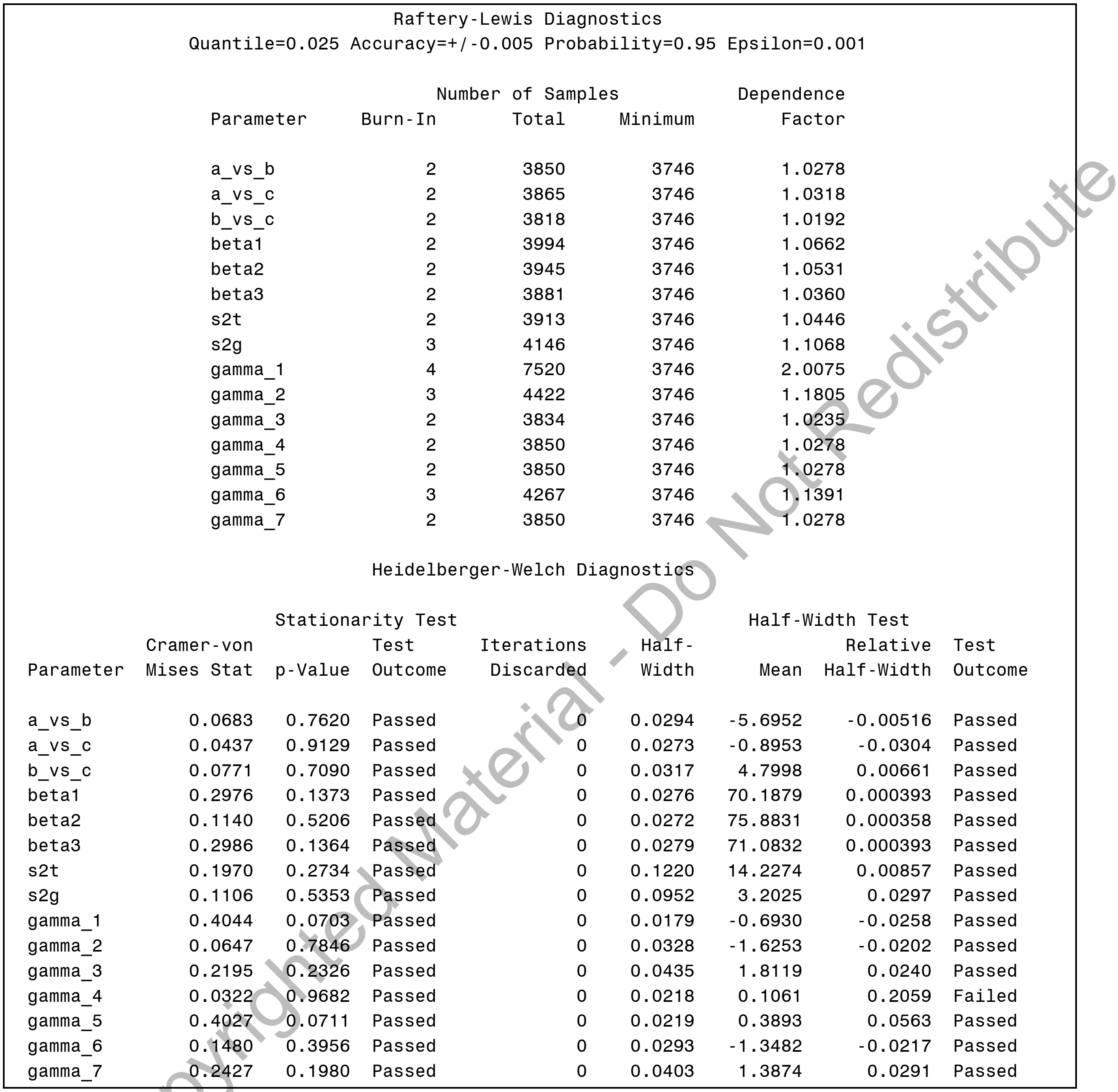
```
                                       Posterior Intervals

                Parameter    Alpha     Equal-Tail Interval        HPD Interval

                a_vs_b       0.050     -9.7426     -1.6006     -9.7378     -1.5994
                a_vs_c       0.050     -4.9271      3.1750     -4.9383      3.1435
                b_vs_c       0.050      0.7982      8.7788      0.7876      8.7574
                beta1        0.050     67.0399     73.3180     67.0083     73.2743
                beta2        0.050     72.7695     78.9626     72.7635     78.9523
                beta3        0.050     67.9350     74.1671     67.9183     74.1456
                s2t          0.050      5.4081     30.3946      4.0841     26.6922
                s2g          0.050      0.2280     14.2690      0.1015     10.8483
                gamma_1      0.050     -3.7735      1.5786     -3.4676      1.8251
                gamma_2      0.050     -5.7270      0.8781     -5.2461      1.1956
                gamma_3      0.050     -0.7634      6.1215     -0.9982      5.6812
                gamma_4      0.050     -2.3644      2.7012     -2.3261      2.7291
                gamma_5      0.050     -1.9193      3.2199     -1.9071      3.2276
                gamma_6      0.050     -5.1327      1.0354     -4.8354      1.2439
                gamma_7      0.050     -1.0093      5.2530     -1.1950      4.8841
```

The contrast between adhesives **a** and **b** and the contrast between adhesives **b** and **c** seem to be important because 0 is not in the credible interval.

```
                                   Monte Carlo Standard Errors

                                                 Standard
                          Parameter        MCSE    Deviation     MCSE/SD

                          a_vs_b         0.0147       2.0286     0.00726
                          a_vs_c         0.0148       2.0197     0.00730
                          b_vs_c         0.0147       2.0141     0.00730
                          beta1          0.0121       1.5808     0.00766
                          beta2          0.0120       1.5699     0.00766
                          beta3          0.0125       1.5810     0.00792
                          s2t            0.0596       6.4917     0.00919
                          s2g            0.0422       4.1222      0.0102
                          gamma_1        0.0112       1.3214     0.00845
                          gamma_2        0.0180       1.7441      0.0103
                          gamma_3        0.0204       1.8488      0.0110
                          gamma_4       0.00953       1.2233     0.00779
                          gamma_5       0.00989       1.2629     0.00783
                          gamma_6        0.0159       1.6077     0.00986
                          gamma_7        0.0166       1.6360      0.0102
```



```
                      Posterior Autocorrelations

 Parameter       Lag 1       Lag 5      Lag 10      Lag 50

 a_vs_b         0.0274     -0.0107      0.0061     -0.0076
 a_vs_c         0.0335      0.0016      0.0104      0.0058
 b_vs_c         0.0334      0.0027     -0.0054      0.0004
 beta1          0.0873     -0.0175      0.0121     -0.0088
 beta2          0.0867     -0.0056      0.0104     -0.0004
 beta3          0.0902      0.0143      0.0063      0.0091
 s2t            0.1917      0.0056     -0.0071      0.0004
 s2g            0.2728      0.0157     -0.0063     -0.0006
 gamma_1        0.1248      0.0034     -0.0075     -0.0020
 gamma_2        0.2911      0.0214     -0.0177     -0.0022
 gamma_3        0.3552      0.0316     -0.0061     -0.0075
 gamma_4        0.0600      0.0008      0.0056     -0.0002
 gamma_5        0.0867     -0.0072     -0.0108     -0.0000
 gamma_6        0.2549      0.0157     -0.0143     -0.0053
 gamma_7        0.2800      0.0170      0.0045      0.0060

                          Geweke Diagnostics

           Parameter            z    Pr > |z|

           a_vs_b         -0.6487      0.5165
           a_vs_c          0.1071      0.9147
           b_vs_c          0.7146      0.4748
           beta1          -0.9434      0.3455
           beta2          -0.3684      0.7126
           beta3          -1.0705      0.2844
           s2t             0.1392      0.8893
           s2g            -0.0869      0.9307
           gamma_1         0.0620      0.9505
           gamma_2         0.6181      0.5365
           gamma_3         0.1418      0.8872
           gamma_4         0.3350      0.7376
           gamma_5        -0.2101      0.8336
           gamma_6        -0.1432      0.8862
           gamma_7         0.0342      0.9727
```



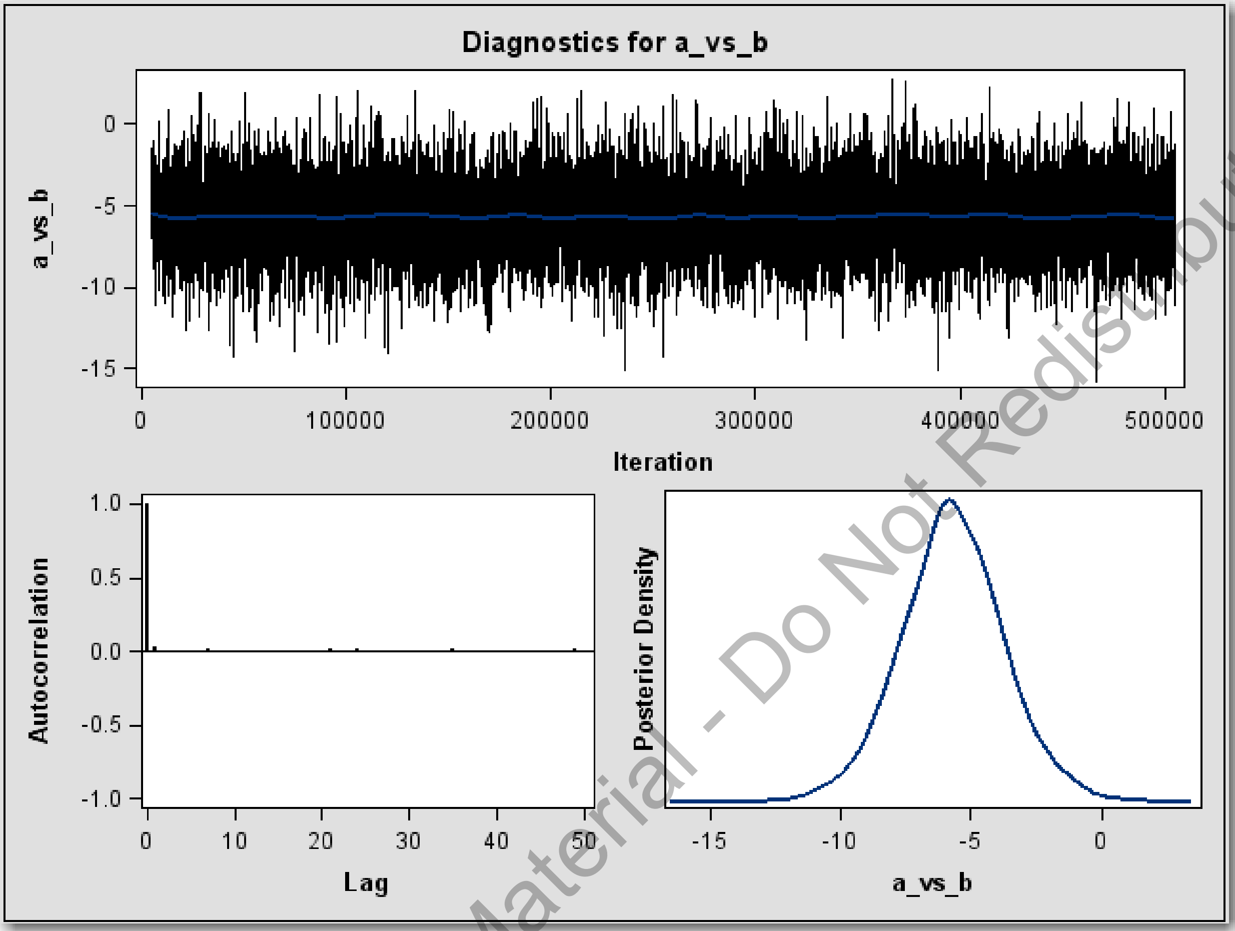
```
                                  Raftery-Lewis Diagnostics
                 Quantile=0.025 Accuracy=+/-0.005 Probability=0.95 Epsilon=0.001

                                      Number of Samples          Dependence
                   Parameter     Burn-In        Total     Minimum        Factor

                   a_vs_b              2         3850        3746        1.0278
                   a_vs_c              2         3865        3746        1.0318
                   b_vs_c              2         3818        3746        1.0192
                   beta1               2         3994        3746        1.0662
                   beta2               2         3945        3746        1.0531
                   beta3               2         3881        3746        1.0360
                   s2t                 2         3913        3746        1.0446
                   s2g                 3         4146        3746        1.1068
                   gamma_1             4         7520        3746        2.0075
                   gamma_2             3         4422        3746        1.1805
                   gamma_3             2         3834        3746        1.0235
                   gamma_4             2         3850        3746        1.0278
                   gamma_5             2         3850        3746        1.0278
                   gamma_6             3         4267        3746        1.1391
                   gamma_7             2         3850        3746        1.0278

                                Heidelberger-Welch Diagnostics

                        Stationarity Test                           Half-Width Test
           Cramer-von            Test      Iterations      Half-              Relative  Test
Parameter  Mises Stat  p-Value  Outcome     Discarded     Width       Mean  Half-Width  Outcome

a_vs_b         0.0683   0.7620  Passed              0    0.0294    -5.6952    -0.00516  Passed
a_vs_c         0.0437   0.9129  Passed              0    0.0273    -0.8953     -0.0304  Passed
b_vs_c         0.0771   0.7090  Passed              0    0.0317     4.7998     0.00661  Passed
beta1          0.2976   0.1373  Passed              0    0.0276    70.1879    0.000393  Passed
beta2          0.1140   0.5206  Passed              0    0.0272    75.8831    0.000358  Passed
beta3          0.2986   0.1364  Passed              0    0.0279    71.0832    0.000393  Passed
s2t            0.1970   0.2734  Passed              0    0.1220    14.2274     0.00857  Passed
s2g            0.1106   0.5353  Passed              0    0.0952     3.2025      0.0297  Passed
gamma_1        0.4044   0.0703  Passed              0    0.0179    -0.6930     -0.0258  Passed
gamma_2        0.0647   0.7846  Passed              0    0.0328    -1.6253     -0.0202  Passed
gamma_3        0.2195   0.2326  Passed              0    0.0435     1.8119      0.0240  Passed
gamma_4        0.0322   0.9682  Passed              0    0.0218     0.1061      0.2059  Failed
gamma_5        0.4027   0.0711  Passed              0    0.0219     0.3893      0.0563  Passed
gamma_6        0.1480   0.3956  Passed              0    0.0293    -1.3482     -0.0217  Passed
gamma_7        0.2427   0.1980  Passed              0    0.0403     1.3874      0.0291  Passed
```

One random effect parameter failed the Heidelberger-Welch half-width test, which indicates that there are not enough data to accurately estimate the mean with 95% confidence under a predetermined accuracy value. In this test, the relative half-width quantifies accuracy of the 95% confidence interval of the mean estimate by measuring the ratio between the sample standard error of the mean and the mean itself. In other words, you can stop the Markov chain if the variability of the mean stabilizes with respect to the mean. An implicit assumption is that large means are often accompanied by large variances. If this assumption is not met, then this test can produce false rejections such as a small mean around 0 and a large standard deviation. The random effect parameter that failed the test did indeed have a relatively large standard deviation and a small mean. As with any diagnostic statistic, it is important to examine the diagnostic plots to see whether there truly is a problem.

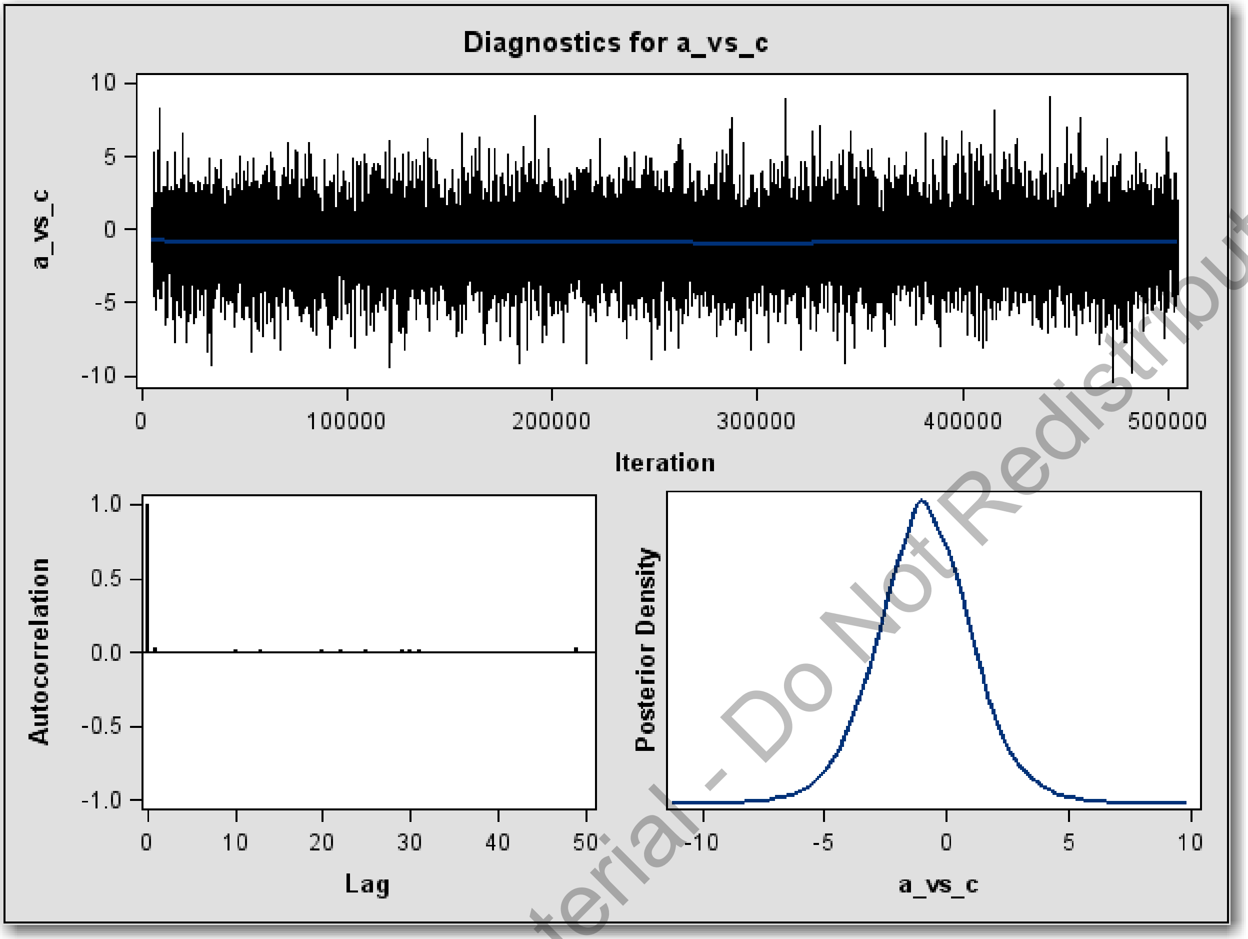
```
                              Effective Sample Sizes

                                       Autocorrelation
                    Parameter          ESS           Time    Efficiency

                    a_vs_b         18959.3         1.0549        0.9480
                    a_vs_c         18744.7         1.0670        0.9372
                    b_vs_c         18749.2         1.0667        0.9375
                    beta1          17026.6         1.1746        0.8513
                    beta2          17045.3         1.1733        0.8523
                    beta3          15937.6         1.2549        0.7969
                    s2t            11849.0         1.6879        0.5925
                    s2g             9542.1         2.0960        0.4771
                    gamma_1        13999.2         1.4287        0.7000
                    gamma_2         9372.7         2.1339        0.4686
                    gamma_3         8209.2         2.4363        0.4105
                    gamma_4        16461.7         1.2149        0.8231
                    gamma_5        16298.3         1.2271        0.8149
                    gamma_6        10280.2         1.9455        0.5140
                    gamma_7         9690.9         2.0638        0.4845

                           Deviance Information Criterion

                  Dbar (posterior mean of deviance)              117.124
                  Dmean (deviance evaluated at posterior mean)   111.328
                  pD (effective number of parameters)              5.796
                  DIC (smaller is better)                        122.921
```

The diagnostic statistics all indicate convergence of the Markov chain.

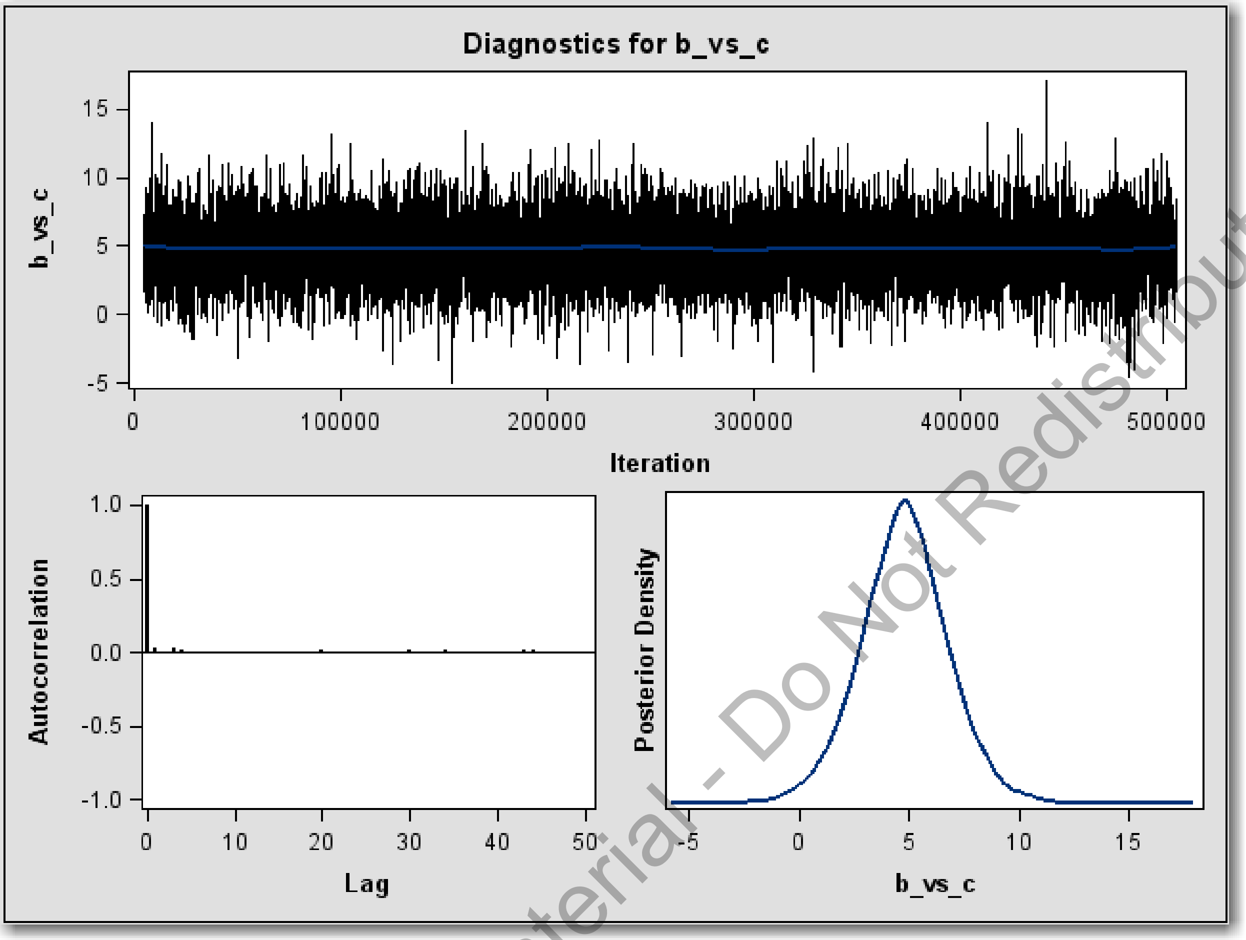
Partial Graphics Output:



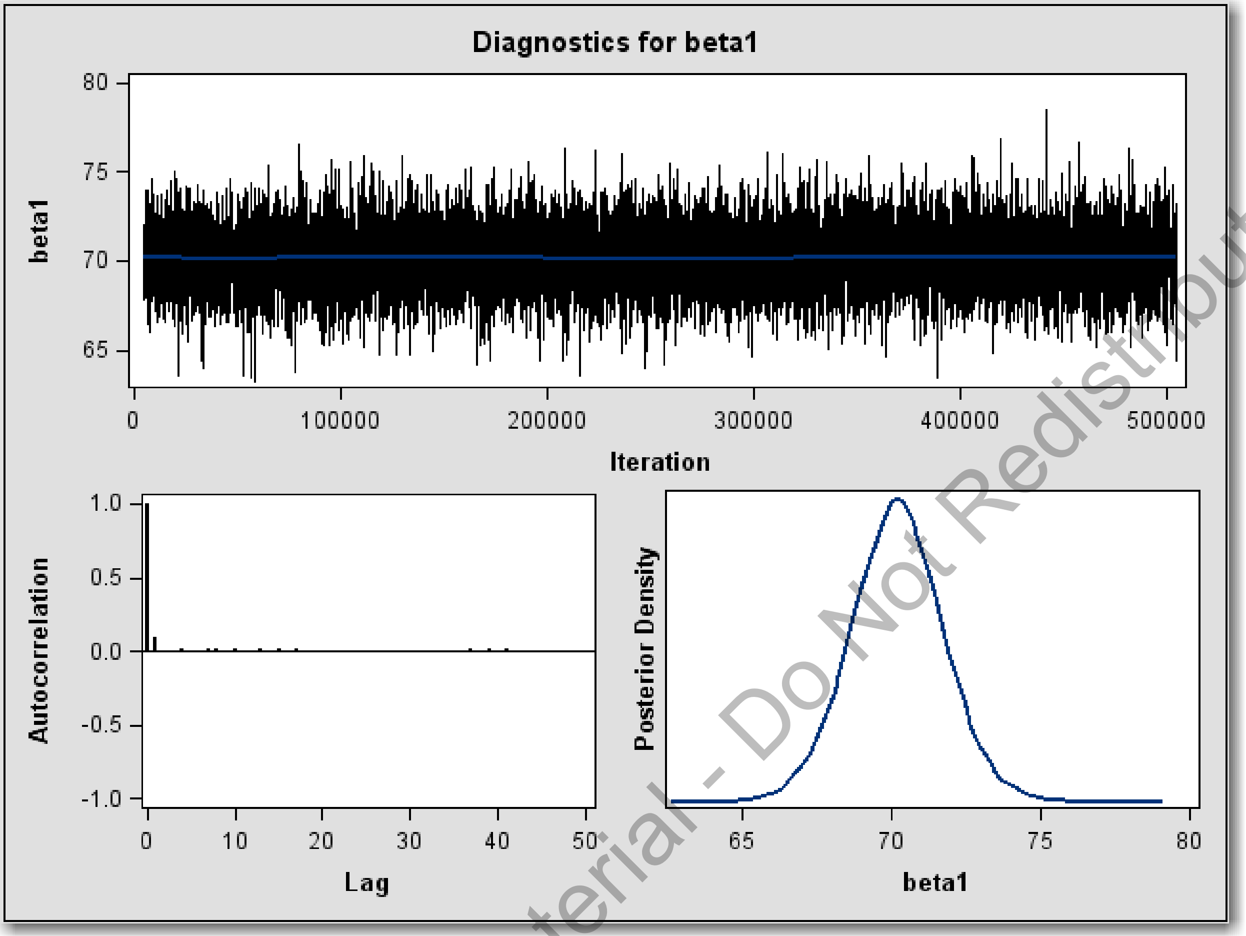
The diagnostic plots for the contrast between adhesives **a** and **b** show a converged Markov chain. Furthermore, 0 seems to be relatively far in the right tail, so the contrast seems to be important.



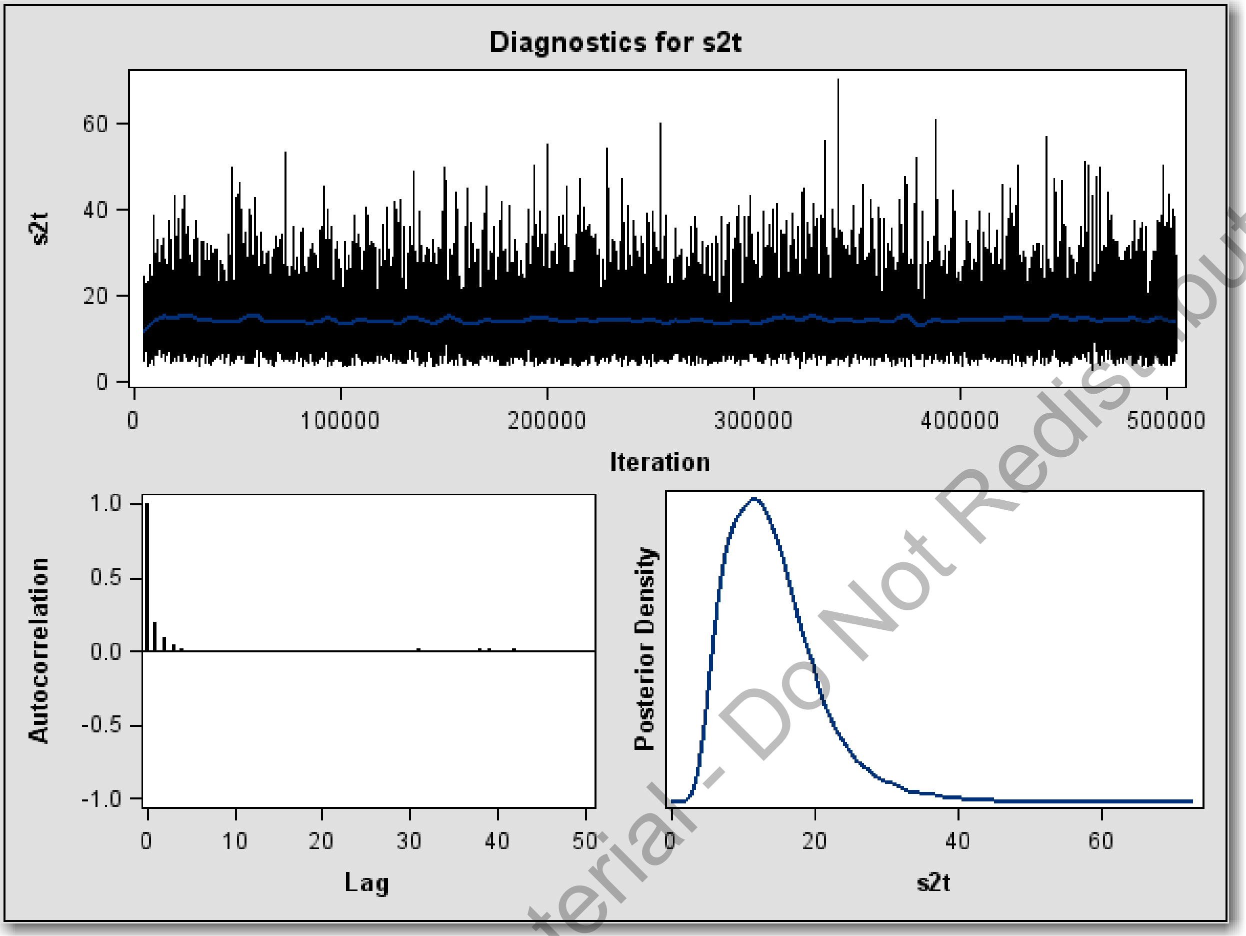
The diagnostic plots for the contrast between **a** and **c** show a converged Markov chain. Notice the distribution of the contrast clearly covers 0, so the difference between adhesive **a** versus **c** is not significant.



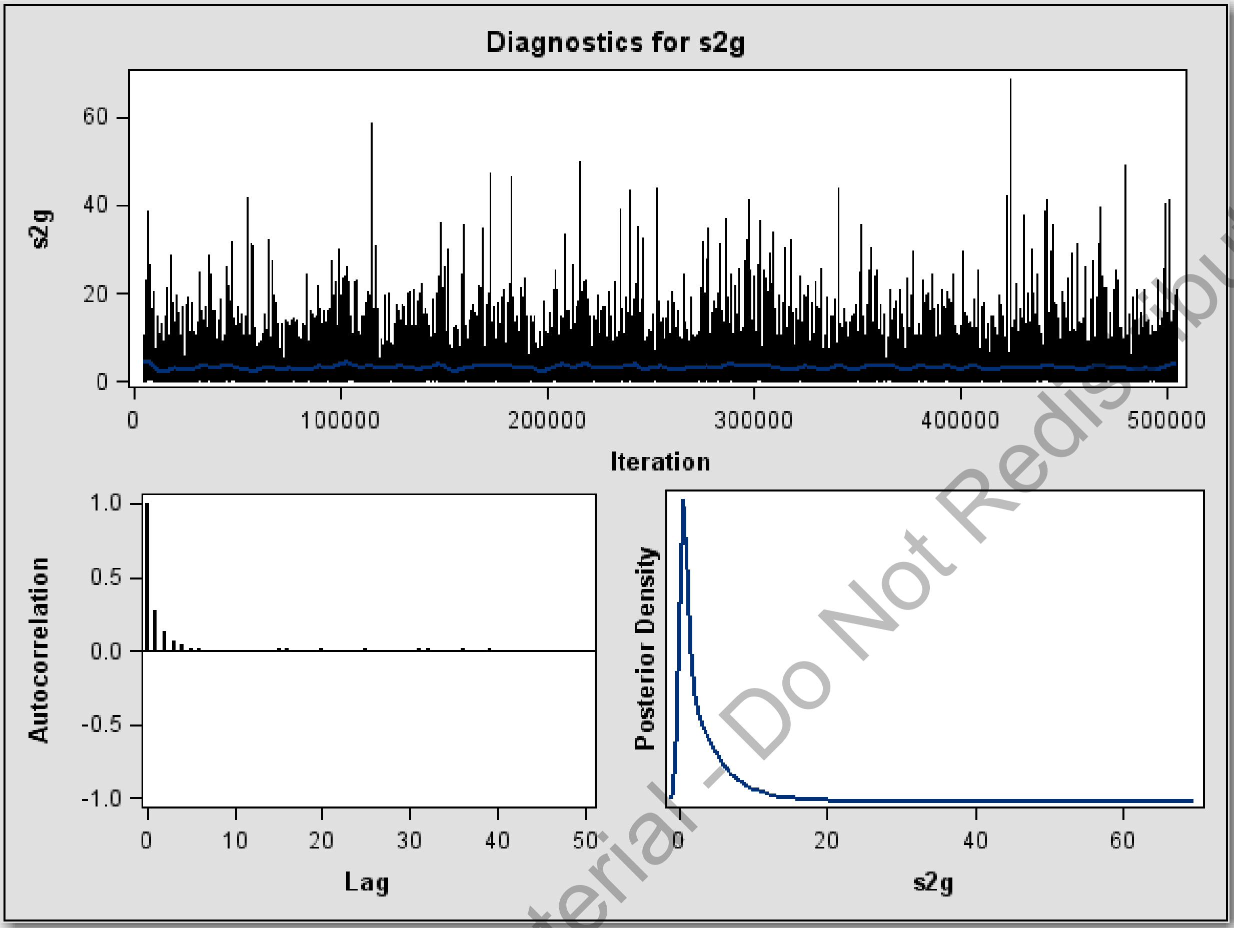
The diagnostic plots for the contrast between **b** and **c** show a converged Markov chain. The distribution of the contrast shows that 0 is relatively far in the left tail, so the contrast seems to be important.



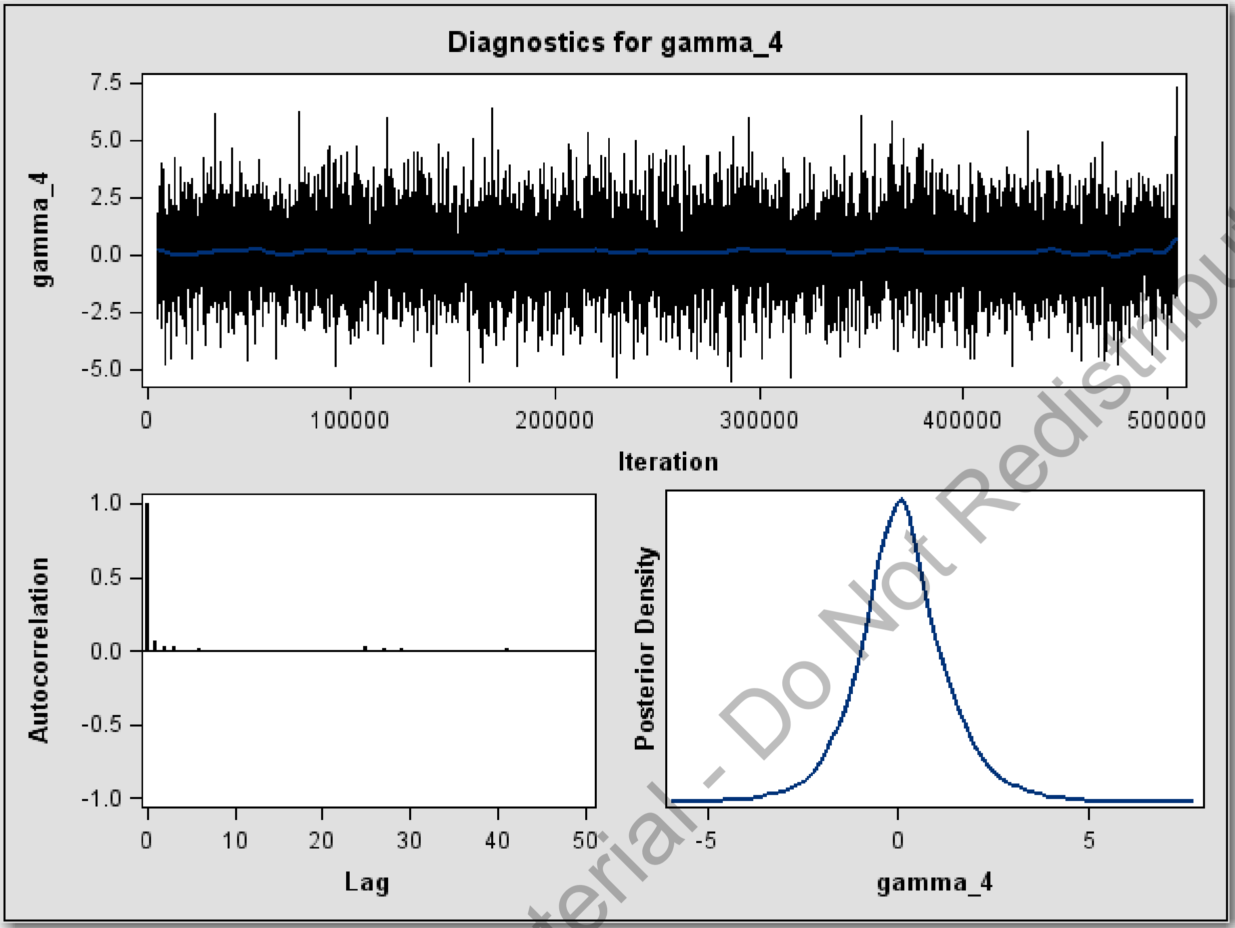
The diagnostic plots for **beta1** (the mean of adhesive a) show a converged Markov chain.



The diagnostic plots for **s2t** (the total variance of the likelihood function) show a converged Markov chain.



The diagnostic plots for **s2g** (the between-block variation) show a converged Markov chain.



The diagnostic plots for the random effect parameter for toy number 4 shows a converged Markov chain.

Note: The other diagnostic plots (not shown) all shown patterns of convergence.

Example: Use the output data set for the posterior samples and create a box plot of the population pressure means by adhesive.

```
data meana (keep=beta1) meanb (keep=beta2) meanc (keep=beta3);
   set mixed;
run;
```

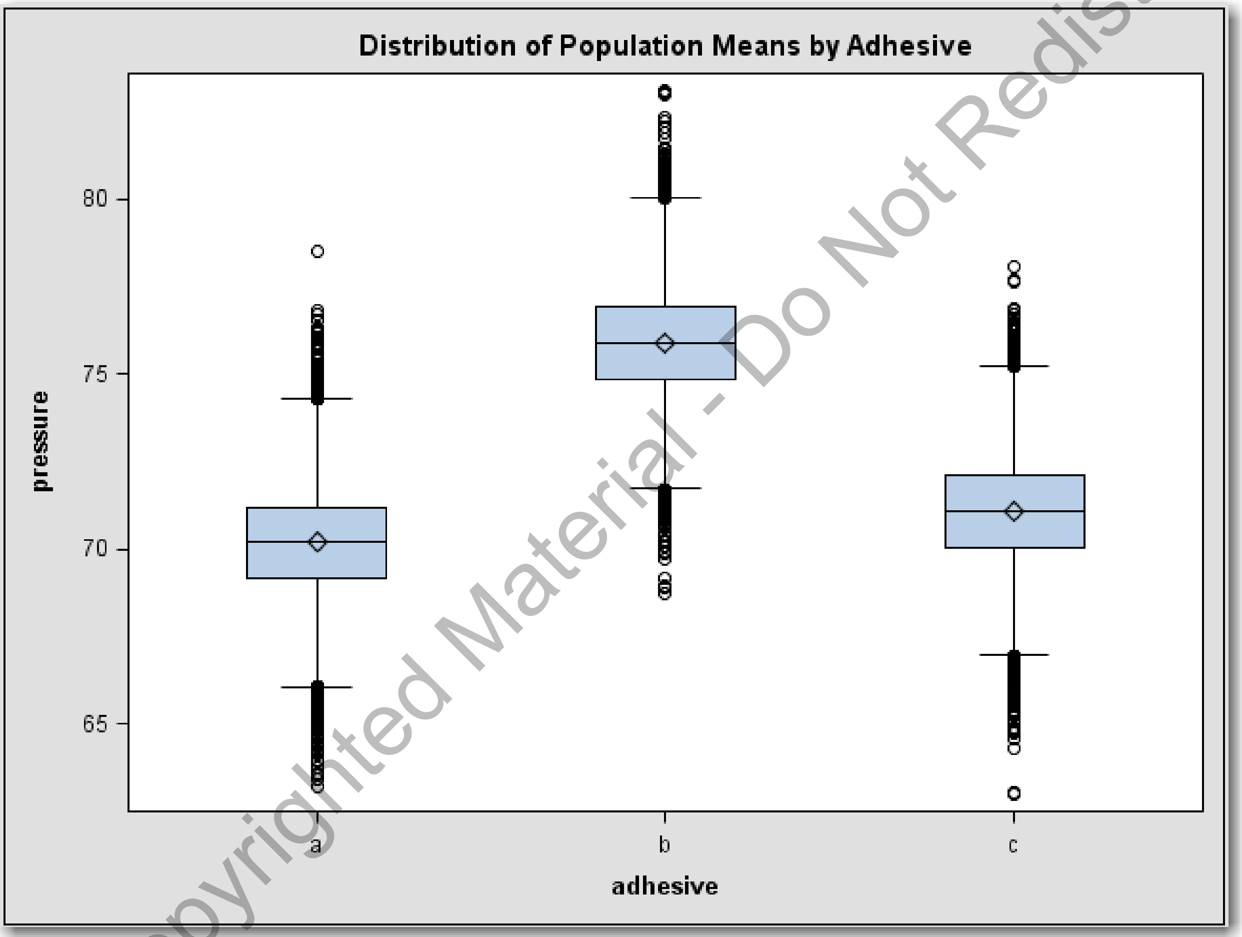
The first step is to create three data sets, each one with a separate parameter estimate.

```
data boxplot;
   set meana (in=a) meanb (in=b) meanc (in=c);
   if a then do;
      pressure=beta1;
      adhesive='a';
   end;
   if b then do;
      pressure=beta2;
      adhesive='b';
   end;
   if c then do;
```

```
      pressure=beta3;
      adhesive='c';
   end;
run;
```

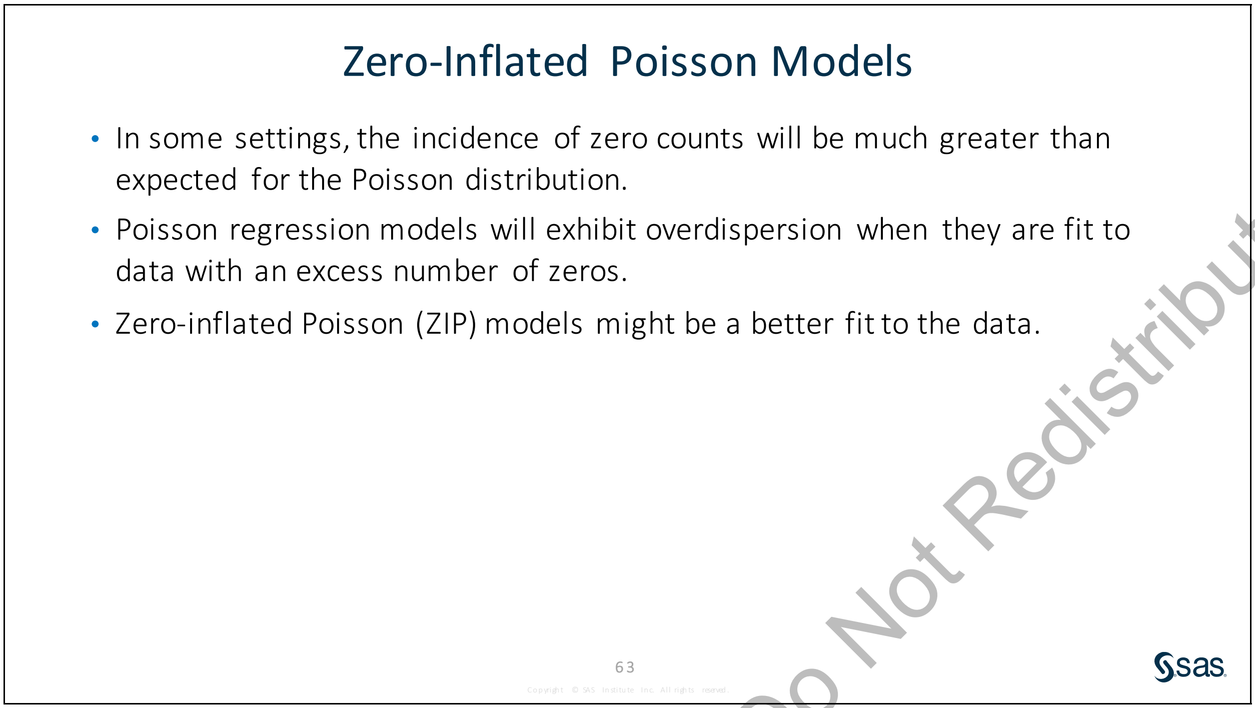
The next step is to concatenate the three data sets and create a variable called **pressure** that has the population mean values and a variable called **adhesive** that has the type of adhesive.

```
proc sgplot data=boxplot;
   vbox pressure / category=adhesive;
   title "Distribution of Population Means by Adhesive";
run;
```



The box plot shows the distribution of the population means by adhesive based on the Bayesian simulations.

End of Demonstration

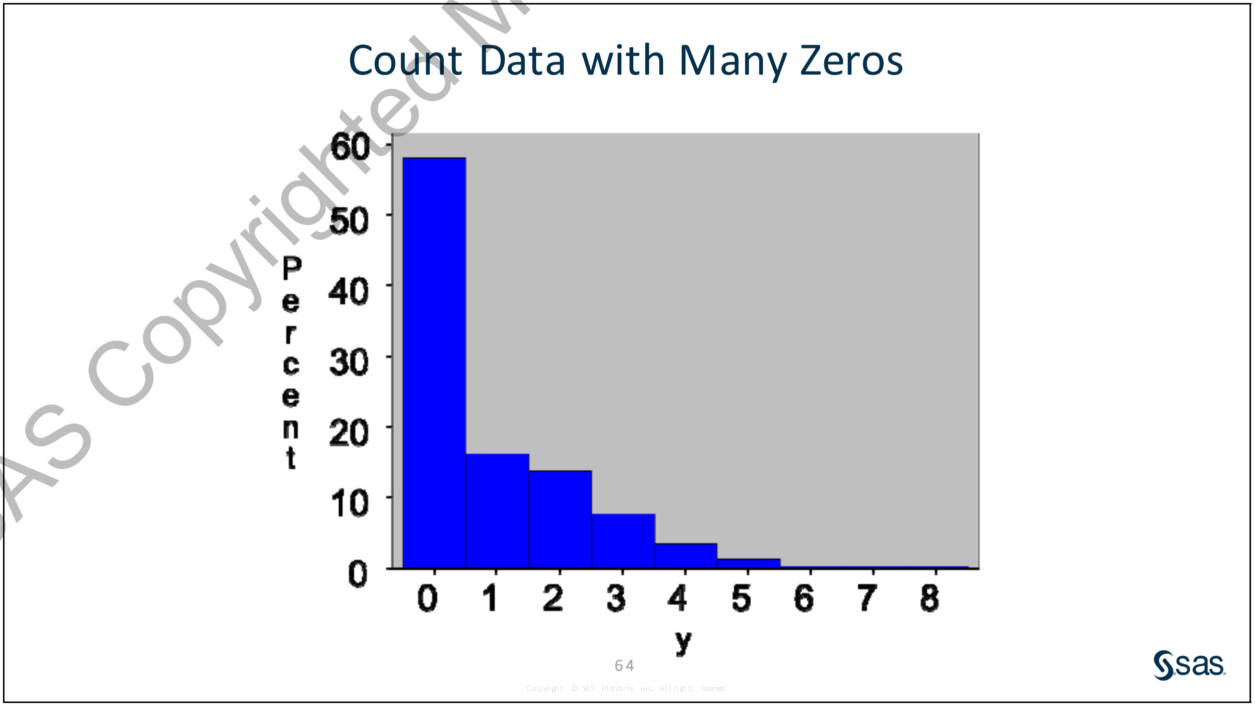


Zero-Inflated Poisson Models

- In some settings, the incidence of zero counts will be much greater than expected for the Poisson distribution.
- Poisson regression models will exhibit overdispersion when they are fit to data with an excess number of zeros.
- Zero-inflated Poisson (ZIP) models might be a better fit to the data.

63

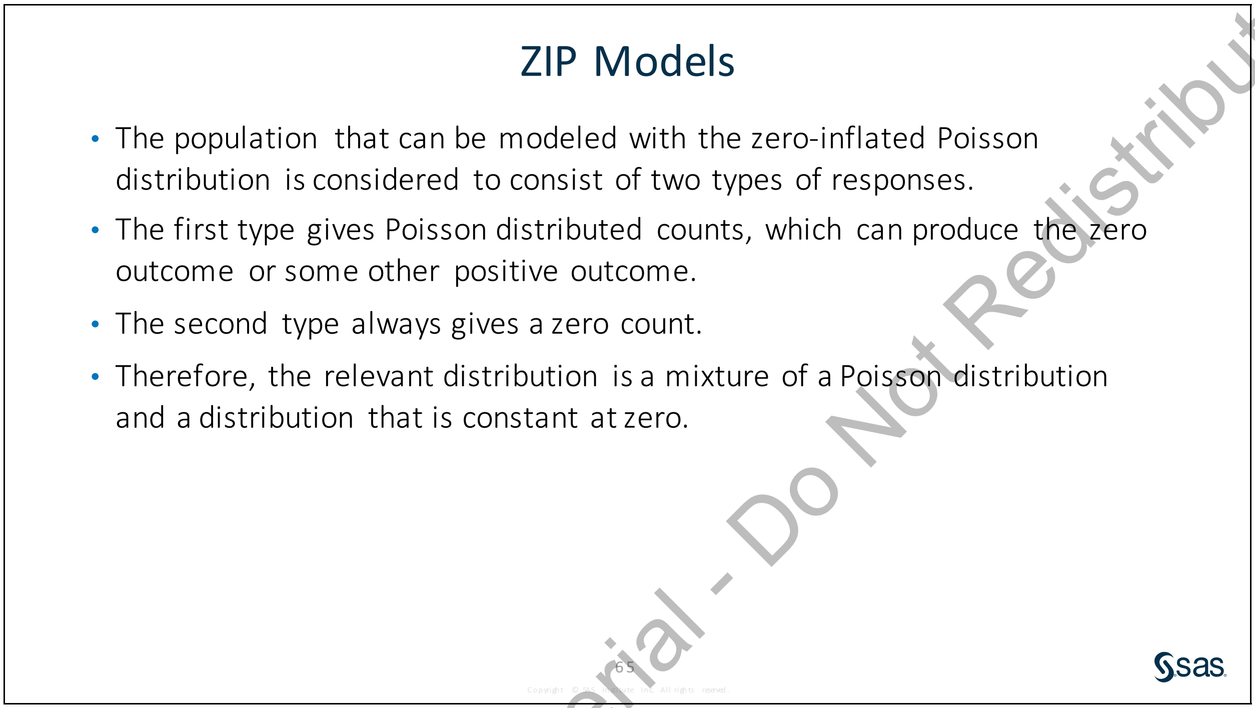
One of the properties of the Poisson distribution is that the mean and variance are equal. However, count data are often overdispersed relative to the Poisson distribution. One possible reason for overdispersion is that the incidence of zero counts is greater than expected for the Poisson distribution. Models that account for overdispersion, such as the negative binomial model, concentrate on modeling the variance-mean relationship correctly. However, if the overdispersion is due to an excess number of zeros, then models that allow for excess zeros such as the zero-inflated Poisson model would be more appropriate.



Count Data with Many Zeros

64

There are many examples of count data with an excess number of zeros. For example, the number of insurance claims for a policy holder at a given year might have zero counts well above what is expected from a Poisson distribution. Another example is the number of disease lesions on plants where a plant might have no lesions either because it is resistant to the disease or because no disease spores have landed on it (Ridout et al. 1998).



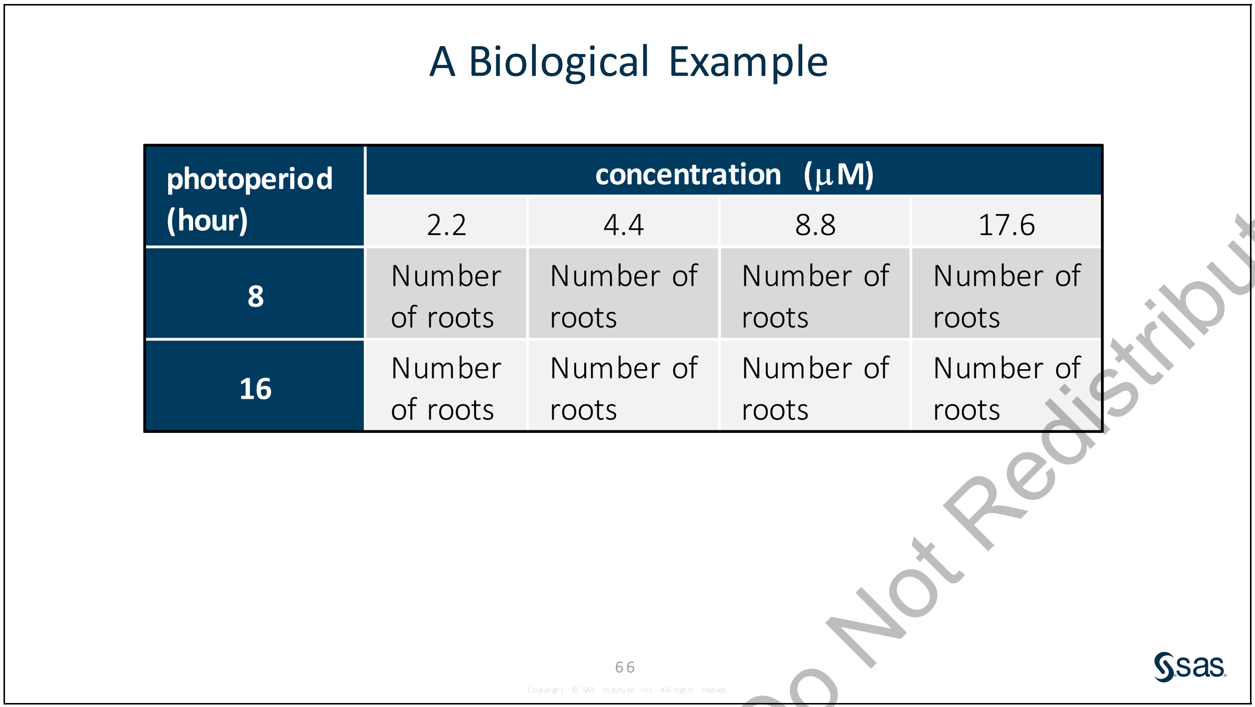
ZIP Models

- The population that can be modeled with the zero-inflated Poisson distribution is considered to consist of two types of responses.
- The first type gives Poisson distributed counts, which can produce the zero outcome or some other positive outcome.
- The second type always gives a zero count.
- Therefore, the relevant distribution is a mixture of a Poisson distribution and a distribution that is constant at zero.

65

Ridout et al. (1998) points out that there are two types of zero outcomes, which are structural zeros, which have an expected value of zero, and the sampling zeros, which occur by chance. Therefore, ZIP models have a distribution that is a mixture of a Poisson distribution (which contributes to the sampling zeros) and a distribution that is constant at zero (which contributes to the structural zeros). An example of a mixture distribution was shown in Lambert (1992) where the outcome is the number of defective items produced by a manufacturing process in a given time interval. If the process is under control, the outcome is always zero. If the process is not under control, the number of defective items is distributed as Poisson and might be zero or positive in any period.

The property of the mean equaling the variance does not hold in the mixture distribution as the excess number of zeros induces overdispersion. A problem for modelers is to decide whether the overdispersion arises from subject heterogeneity or as a result of the nature of the process generating the zero outcomes.



A Biological Example

photoperiod (hour)	concentration (µM)			
	2.2	4.4	8.8	17.6
8	Number of roots	Number of roots	Number of roots	Number of roots
16	Number of roots	Number of roots	Number of roots	Number of roots

Example: Data are collected for 270 micropropagated shoots (young growth arising from a germinating seed) of the columnar apple cultivar *Trajan*. During the rooting period, all shoots were maintained under identical conditions, but the shoots themselves were cultured on media containing different concentration of the cytokinin BAP (plant hormones), in growth cabinets with an 8- or 16-hour photoperiod.

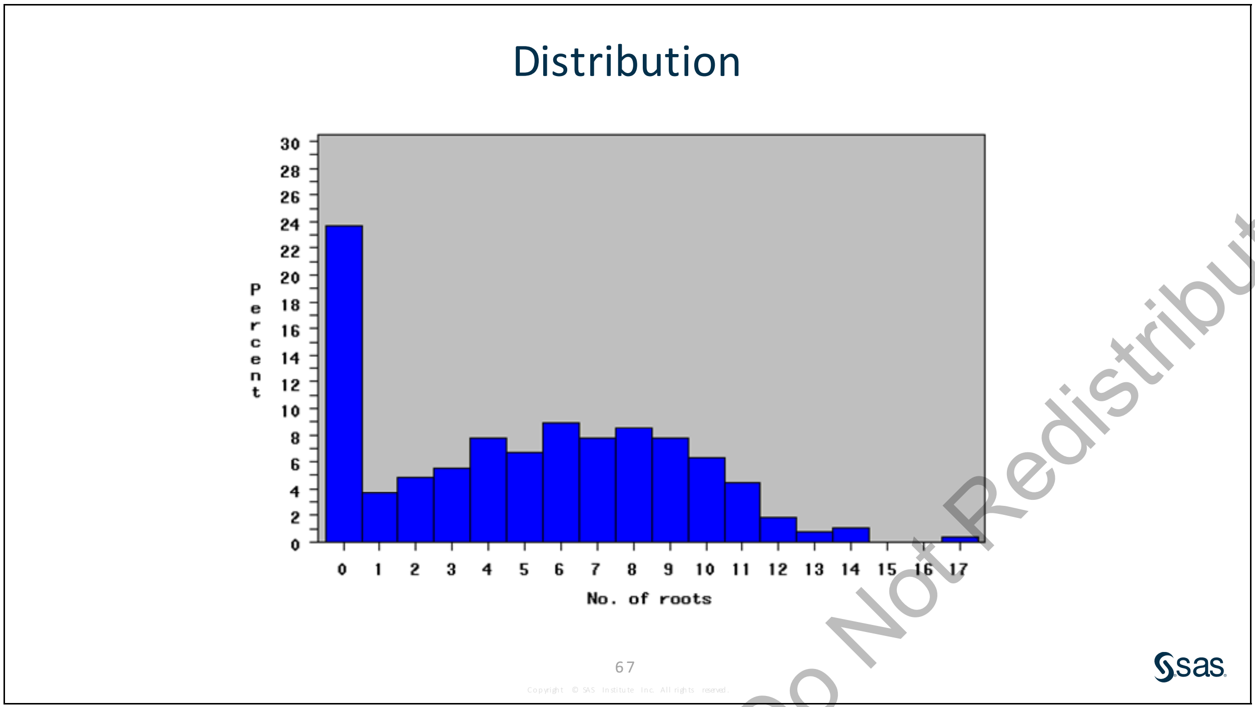
The data are stored in a SAS data set **sasuser.roots** and contains the following variables:

photo the photoperiod, 8 hours, or 16 hours

bap the concentrations of the cytokinin BAP, 2.2, 4.4, 8.8, or 17.6 µm

roots the number of roots

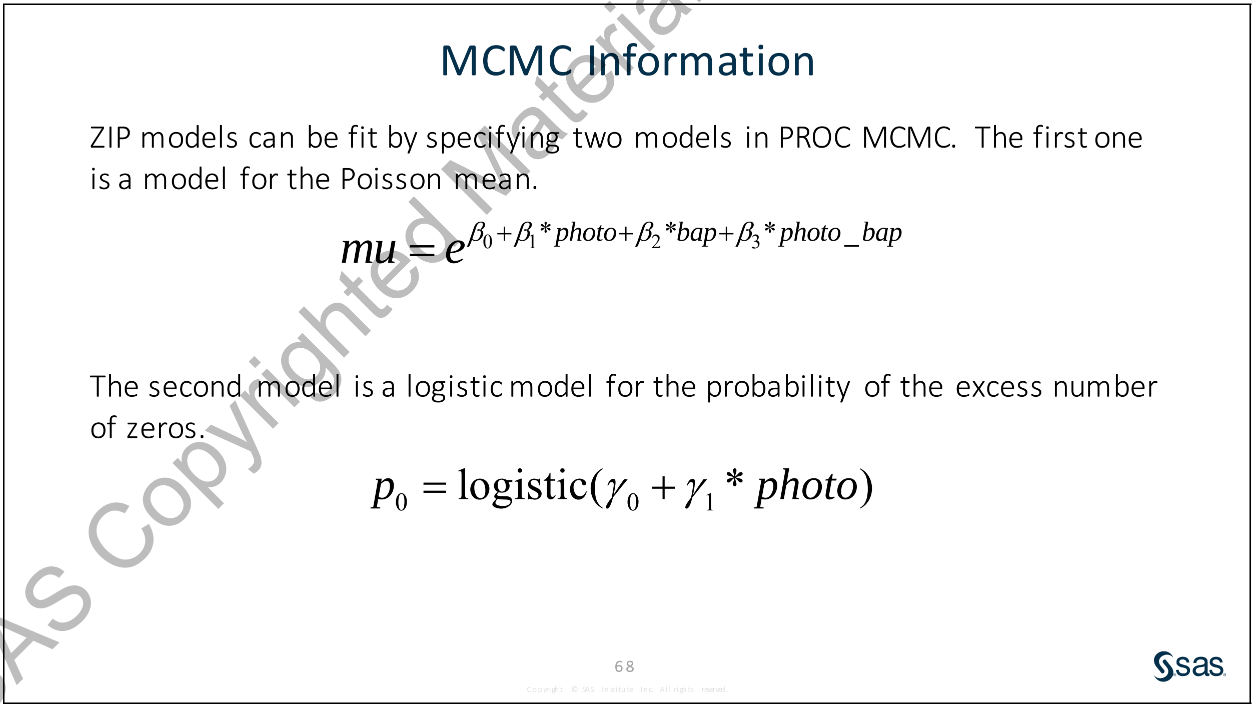
Note: See Ridout et al. (1998) for more information about the data.



Distribution

67

The number of zero counts appears to be more that what you would expect from a Poisson distribution.



MCMC Information

ZIP models can be fit by specifying two models in PROC MCMC. The first one is a model for the Poisson mean.

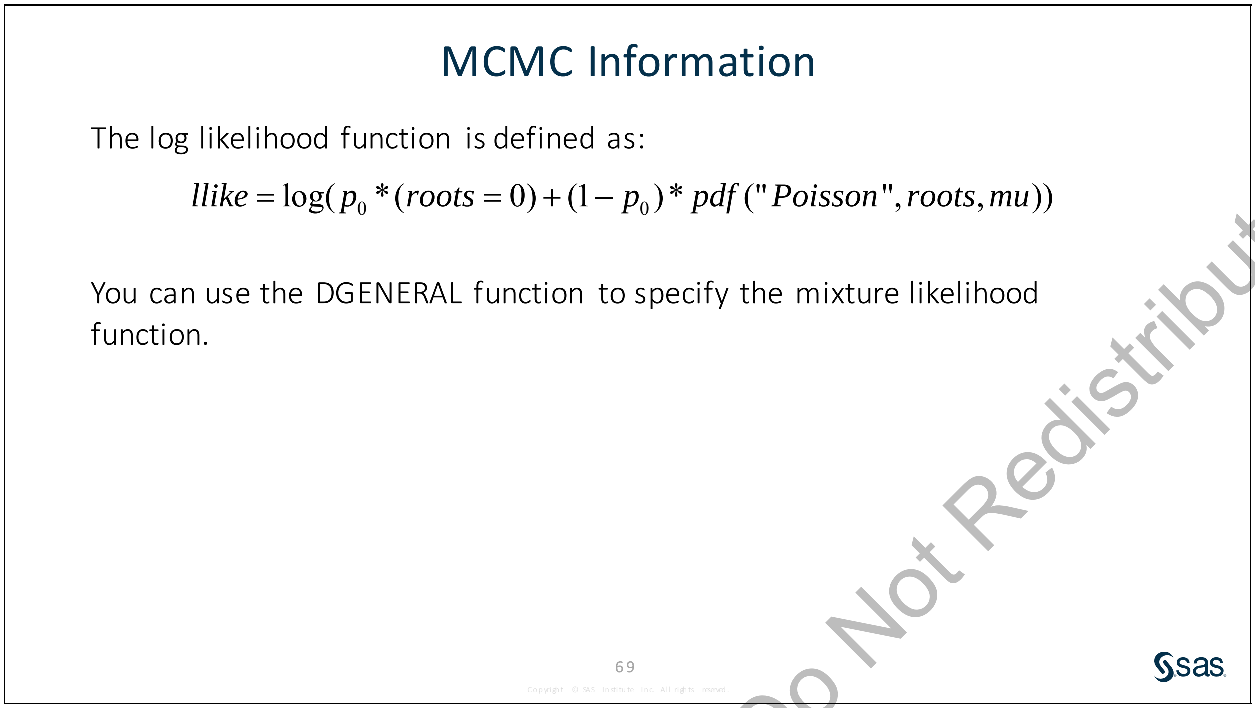
$$mu = e^{\beta_0 + \beta_1 * photo + \beta_2 * bap + \beta_3 * photo\_bap}$$

The second model is a logistic model for the probability of the excess number of zeros.

$$p_0 = \text{logistic}(\gamma_0 + \gamma_1 * photo)$$

68

In the Poisson model, the variables deemed important are the photoperiod, the concentration of the cytokinin BAP, and the interaction of the two. In the logistic model, only the photoperiod was deemed important. The SAS function LOGISTIC does the logit link transformation for the logistic model.



MCMC Information

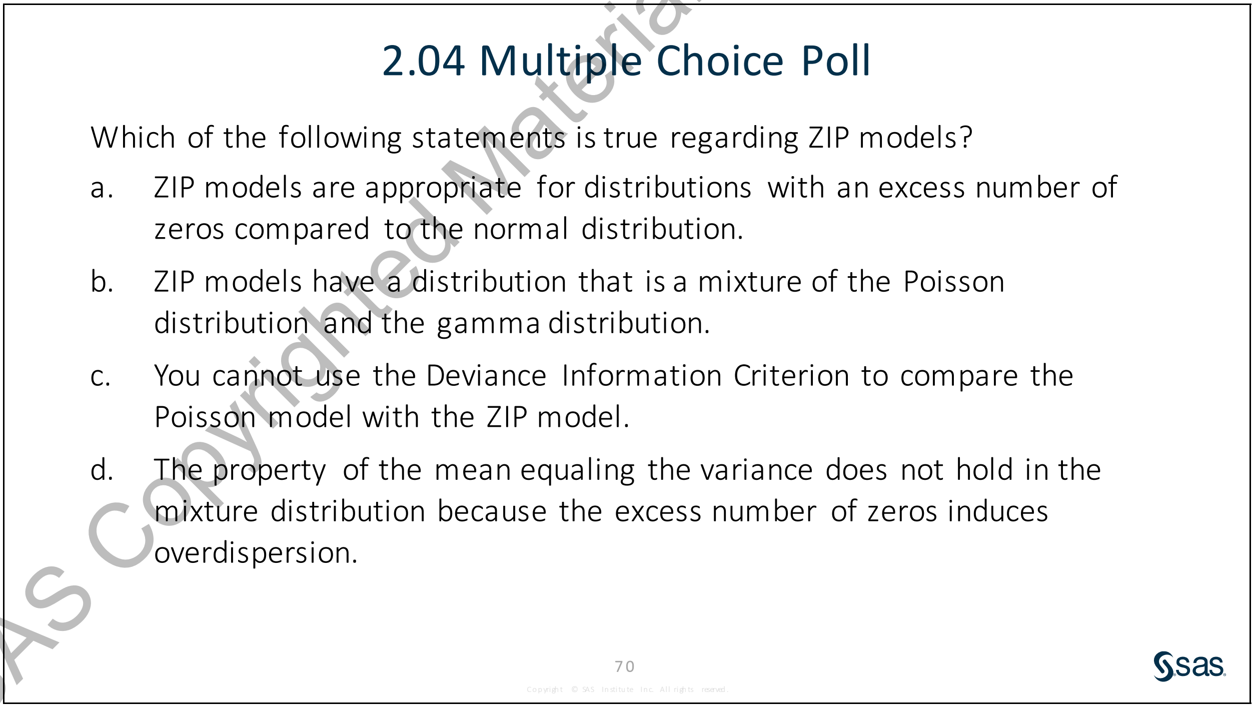
The log likelihood function is defined as:

$$llike = \log(p_0 * (roots = 0) + (1 - p_0) * pdf("Poisson", roots, mu))$$

You can use the DGENERAL function to specify the mixture likelihood function.

69

The log likelihood function is derived from the mixture of two distributions to model the zero-inflated data.



2.04 Multiple Choice Poll

Which of the following statements is true regarding ZIP models?

a. ZIP models are appropriate for distributions with an excess number of zeros compared to the normal distribution.

b. ZIP models have a distribution that is a mixture of the Poisson distribution and the gamma distribution.

c. You cannot use the Deviance Information Criterion to compare the Poisson model with the ZIP model.

d. The property of the mean equaling the variance does not hold in the mixture distribution because the excess number of zeros induces overdispersion.

70

Fitting a Zero-Inflated Poisson Model in PROC MCMC

Example: Fit a ZIP model in PROC MCMC. First, create an interaction between **photo** and **bap** in a DATA step. Specify parameters for the Poisson model and the logistic model and use uninformative normal priors for all the parameters. Specify the Poisson model, the logistic model, and the log likelihood and use the DGENERAL function in the MODEL statement.

```
/* stbay02d04.sas */
data roots;
   set sasuser.roots;
   photo_bap=photo*bap;
run;

proc mcmc data=roots diag=all dic propcov=quanew mchistory=brief
     stats=all nbi=5000 ntu=5000 nmc=500000 thin=10 plots(smooth)=all
     seed=27513;
   parms (beta0 beta1 beta2 beta3) 0;
   parms (gamma0 gamma1) 0;
   prior beta: ~ normal(0,var=1000);
   prior gamma: ~ normal(0,var=10);
   mu=exp(beta0+beta1*photo+beta2*bap+beta3*photo_bap);
   p0=logistic(gamma0+gamma1*photo);
   llike=log(p0*(roots eq 0)+(1-p0)*pdf("poisson",roots,mu));
   model dgeneral(llike);
   title "Bayesian Analysis of Roots Data Set";
run;
```

The PROC MCMC option values were chosen on a trial and error basis to improve the convergence of the Markov chain. The letter ‘D’ in the DGENERAL function stands for discrete. The new distributions have to be specified on the logarithm scale.

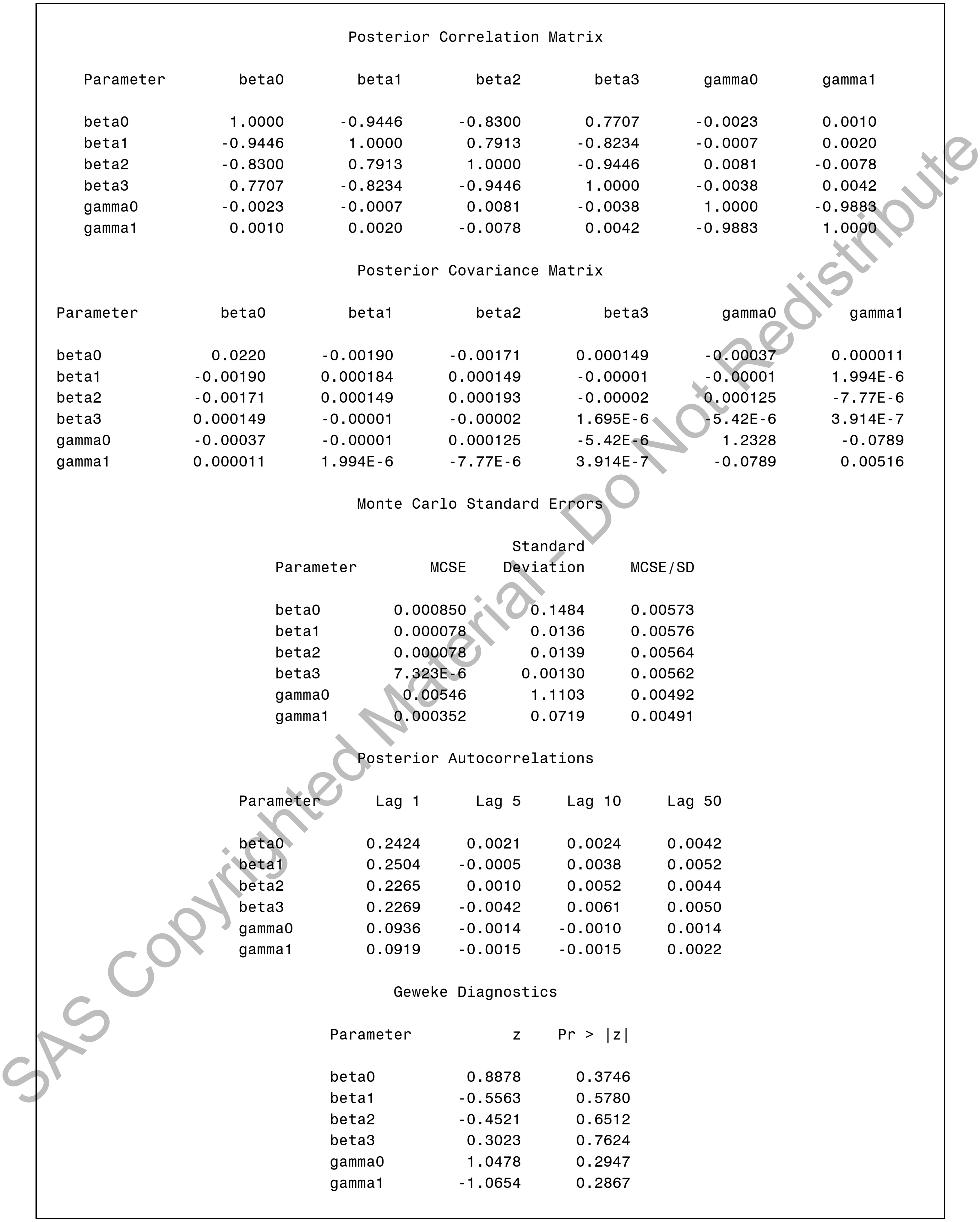
When you use the DGENERAL function in the MODEL statement, you do not need to specify the dependent variable on the left of the tilde. The log-likelihood function takes the dependent variable into account. Hence, there is no need to explicitly state the dependent variable.

```
                         Bayesian Analysis of Roots Data Set

                                 The MCMC Procedure

                       Number of Observations Read         270
                       Number of Observations Used         270

                                     Parameters

                        Sampling         Initial
      Block    Parameter     Method             Value    Prior Distribution

          1    beta0         N-Metropolis           0    normal(0,var=1000)
               beta1                                0    normal(0,var=1000)
               beta2                                0    normal(0,var=1000)
               beta3                                0    normal(0,var=1000)
          2    gamma0        N-Metropolis           0    normal(0,var=10)
               gamma1                               0    normal(0,var=10)
```

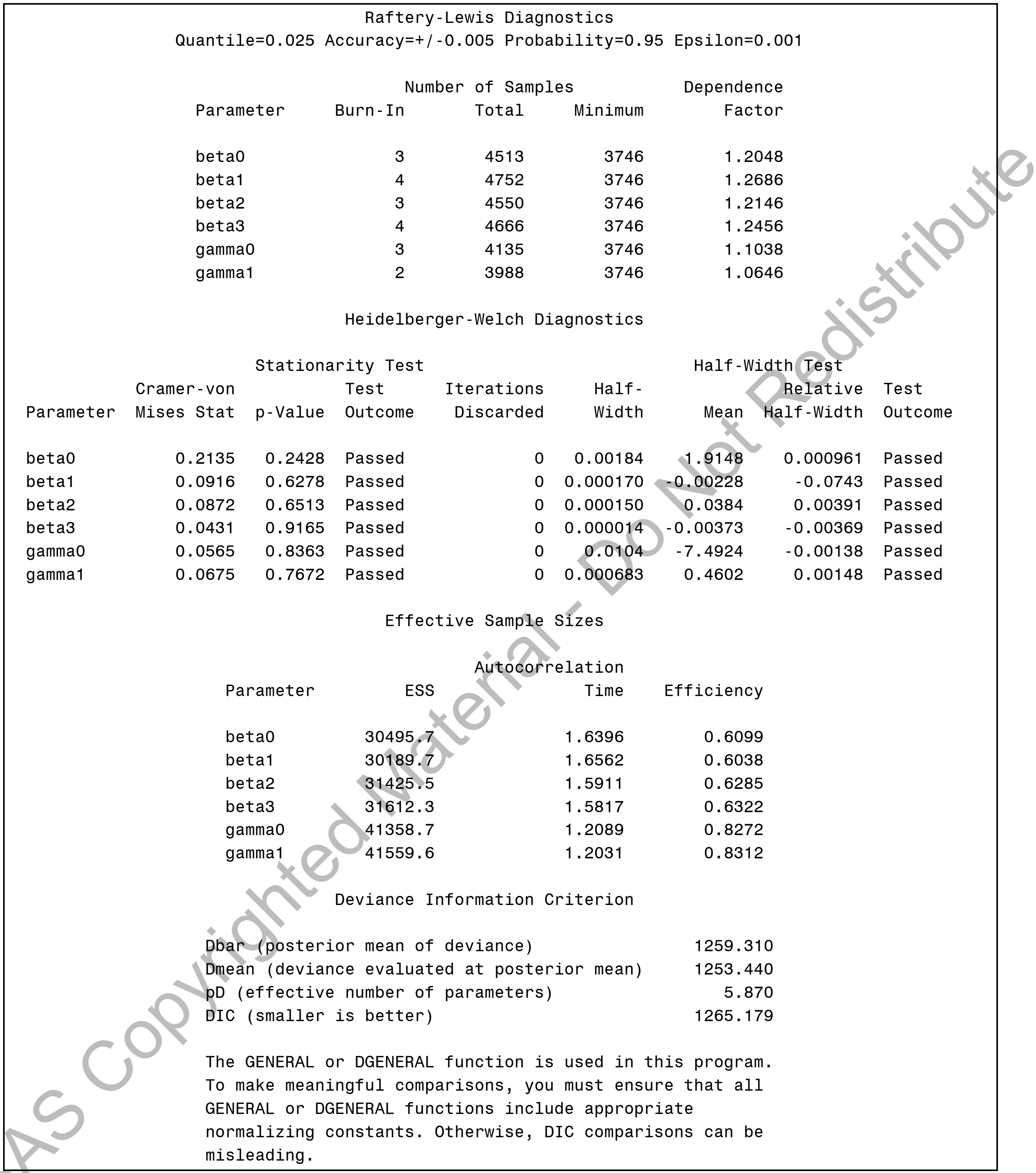
```
                               Tuning History

                             RWM Scale      RWM Acceptance Rate
                  Phase      Low     High      Low     High

                      1    2.380    2.380    0.380    0.575
                      2    3.224    5.054    0.256    0.278

                               Burn-In History

                           RWM Scale      RWM Acceptance Rate
                          Low     High      Low     High

                        3.224    5.054    0.259    0.273

                              Sampling History

                           RWM Scale      RWM Acceptance Rate
                          Low     High      Low     High

                        3.224    5.054    0.256    0.279

                             Posterior Summaries

                                         Standard                 Percentiles
Parameter           N         Mean     Deviation          25           50           75

beta0           50000       1.9148        0.1484      1.8142       1.9147       2.0154
beta1           50000     -0.00228        0.0136     -0.0115     -0.00220      0.00693
beta2           50000       0.0384        0.0139      0.0289       0.0384       0.0478
beta3           50000     -0.00373       0.00130    -0.00461     -0.00373     -0.00285
gamma0          50000      -7.4924        1.1103     -8.1874      -7.4113      -6.7109
gamma1          50000       0.4602        0.0719      0.4095       0.4556       0.5053
```

The results from PROC MCMC closely match the results from the same model fit in PROC GENMOD. **Beta0** was 1.92, **beta1** was -0.002, **beta2** was 0.038, **beta3** was -0.004, **gamma0** was -8.43, and **gamma1** was 0.52 in PROC GENMOD.

```
                             Posterior Intervals

        Parameter    Alpha     Equal-Tail Interval        HPD Interval

        beta0        0.050      1.6249      2.2054      1.6222      2.2016
        beta1        0.050     -0.0290      0.0241     -0.0286      0.0245
        beta2        0.050      0.0113      0.0656      0.0111      0.0654
        beta3        0.050    -0.00629    -0.00121    -0.00625    -0.00118
        gamma0       0.050     -9.8834     -5.5500     -9.6794     -5.3940
        gamma1       0.050      0.3335      0.6139      0.3229      0.6011
```



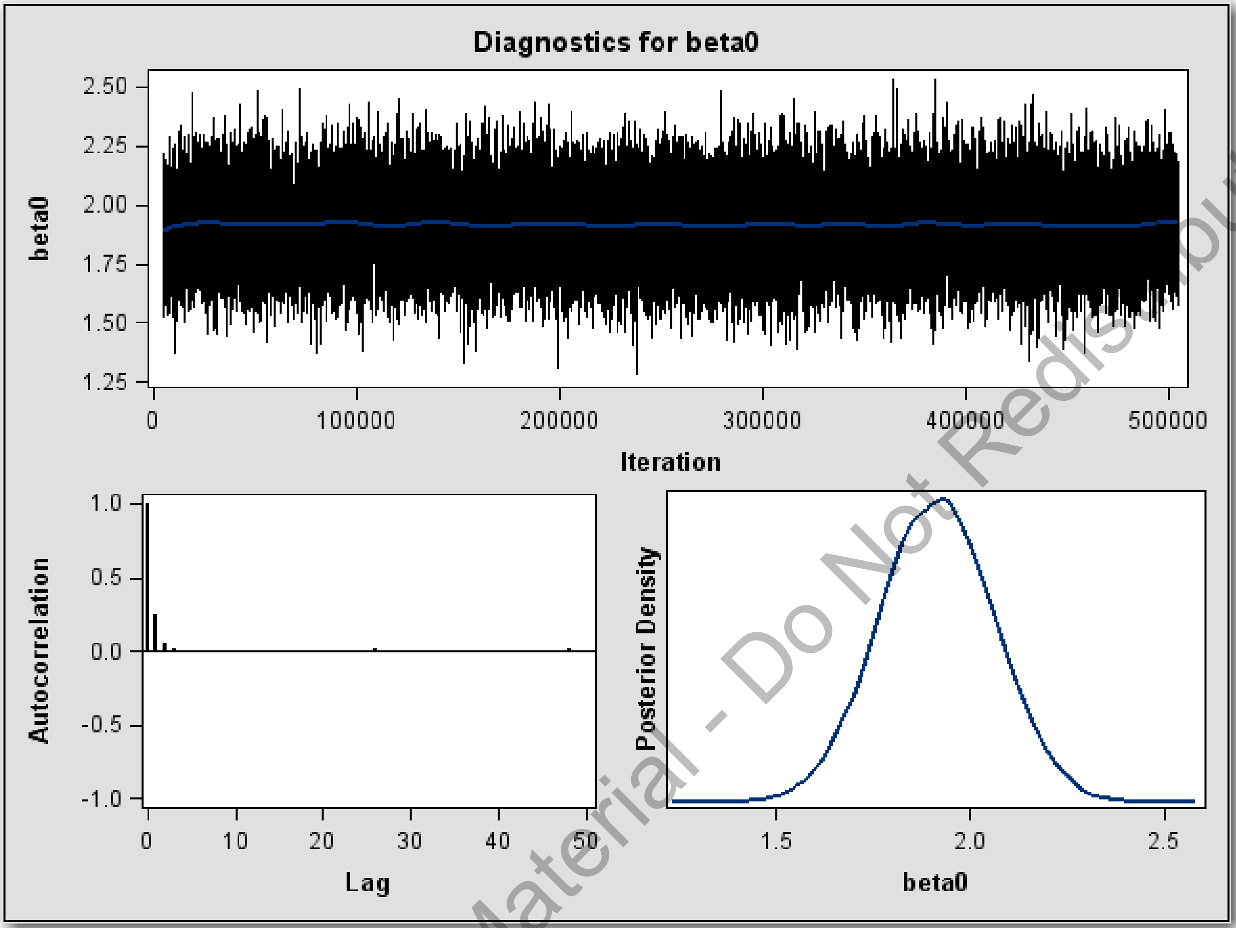
```
                              Posterior Correlation Matrix

   Parameter        beta0        beta1        beta2        beta3       gamma0       gamma1

   beta0           1.0000      -0.9446      -0.8300       0.7707      -0.0023       0.0010
   beta1          -0.9446       1.0000       0.7913      -0.8234      -0.0007       0.0020
   beta2          -0.8300       0.7913       1.0000      -0.9446       0.0081      -0.0078
   beta3           0.7707      -0.8234      -0.9446       1.0000      -0.0038       0.0042
   gamma0         -0.0023      -0.0007       0.0081      -0.0038       1.0000      -0.9883
   gamma1          0.0010       0.0020      -0.0078       0.0042      -0.9883       1.0000

                               Posterior Covariance Matrix

Parameter        beta0         beta1         beta2         beta3        gamma0        gamma1

beta0           0.0220      -0.00190      -0.00171      0.000149      -0.00037      0.000011
beta1         -0.00190      0.000184      0.000149      -0.00001      -0.00001      1.994E-6
beta2         -0.00171      0.000149      0.000193      -0.00002      0.000125      -7.77E-6
beta3         0.000149      -0.00001      -0.00002      1.695E-6      -5.42E-6      3.914E-7
gamma0        -0.00037      -0.00001      0.000125      -5.42E-6        1.2328       -0.0789
gamma1        0.000011      1.994E-6      -7.77E-6      3.914E-7       -0.0789       0.00516

                               Monte Carlo Standard Errors

                                                 Standard
                        Parameter        MCSE    Deviation     MCSE/SD

                        beta0        0.000850       0.1484     0.00573
                        beta1        0.000078       0.0136     0.00576
                        beta2        0.000078       0.0139     0.00564
                        beta3        7.323E-6      0.00130     0.00562
                        gamma0        0.00546       1.1103     0.00492
                        gamma1       0.000352       0.0719     0.00491

                               Posterior Autocorrelations

                    Parameter       Lag 1       Lag 5      Lag 10      Lag 50

                    beta0          0.2424      0.0021      0.0024      0.0042
                    beta1          0.2504     -0.0005      0.0038      0.0052
                    beta2          0.2265      0.0010      0.0052      0.0044
                    beta3          0.2269     -0.0042      0.0061      0.0050
                    gamma0         0.0936     -0.0014     -0.0010      0.0014
                    gamma1         0.0919     -0.0015     -0.0015      0.0022

                                   Geweke Diagnostics

                              Parameter            z    Pr > |z|

                              beta0           0.8878      0.3746
                              beta1          -0.5563      0.5780
                              beta2          -0.4521      0.6512
                              beta3           0.3023      0.7624
                              gamma0          1.0478      0.2947
                              gamma1         -1.0654      0.2867
```



```
                              Raftery-Lewis Diagnostics
               Quantile=0.025 Accuracy=+/-0.005 Probability=0.95 Epsilon=0.001

                                  Number of Samples          Dependence
                 Parameter     Burn-In        Total     Minimum        Factor

                 beta0               3         4513        3746        1.2048
                 beta1               4         4752        3746        1.2686
                 beta2               3         4550        3746        1.2146
                 beta3               4         4666        3746        1.2456
                 gamma0              3         4135        3746        1.1038
                 gamma1              2         3988        3746        1.0646

                              Heidelberger-Welch Diagnostics

                     Stationarity Test                              Half-Width Test
           Cramer-von             Test      Iterations     Half-                Relative  Test
Parameter  Mises Stat  p-Value  Outcome      Discarded     Width       Mean  Half-Width  Outcome

beta0          0.2135   0.2428  Passed               0   0.00184     1.9148    0.000961  Passed
beta1          0.0916   0.6278  Passed               0  0.000170   -0.00228     -0.0743  Passed
beta2          0.0872   0.6513  Passed               0  0.000150     0.0384     0.00391  Passed
beta3          0.0431   0.9165  Passed               0  0.000014   -0.00373    -0.00369  Passed
gamma0         0.0565   0.8363  Passed               0    0.0104    -7.4924    -0.00138  Passed
gamma1         0.0675   0.7672  Passed               0  0.000683     0.4602     0.00148  Passed

                                  Effective Sample Sizes

                                         Autocorrelation
                    Parameter         ESS           Time    Efficiency

                    beta0         30495.7         1.6396        0.6099
                    beta1         30189.7         1.6562        0.6038
                    beta2         31425.5         1.5911        0.6285
                    beta3         31612.3         1.5817        0.6322
                    gamma0        41358.7         1.2089        0.8272
                    gamma1        41559.6         1.2031        0.8312

                              Deviance Information Criterion

                  Dbar (posterior mean of deviance)                 1259.310
                  Dmean (deviance evaluated at posterior mean)      1253.440
                  pD (effective number of parameters)                  5.870
                  DIC (smaller is better)                           1265.179

                  The GENERAL or DGENERAL function is used in this program.
                  To make meaningful comparisons, you must ensure that all
                  GENERAL or DGENERAL functions include appropriate
                  normalizing constants. Otherwise, DIC comparisons can be
                  misleading.
```

The diagnostic statistics all show that the Markov chain reached convergence. The note regarding the DGENERAL function alerts users to make sure that the prior and the likelihood are proper in the sense that π(θ) is a probability density function and the likelihood function is constructed using the proper probability densities of the data given the parameters.

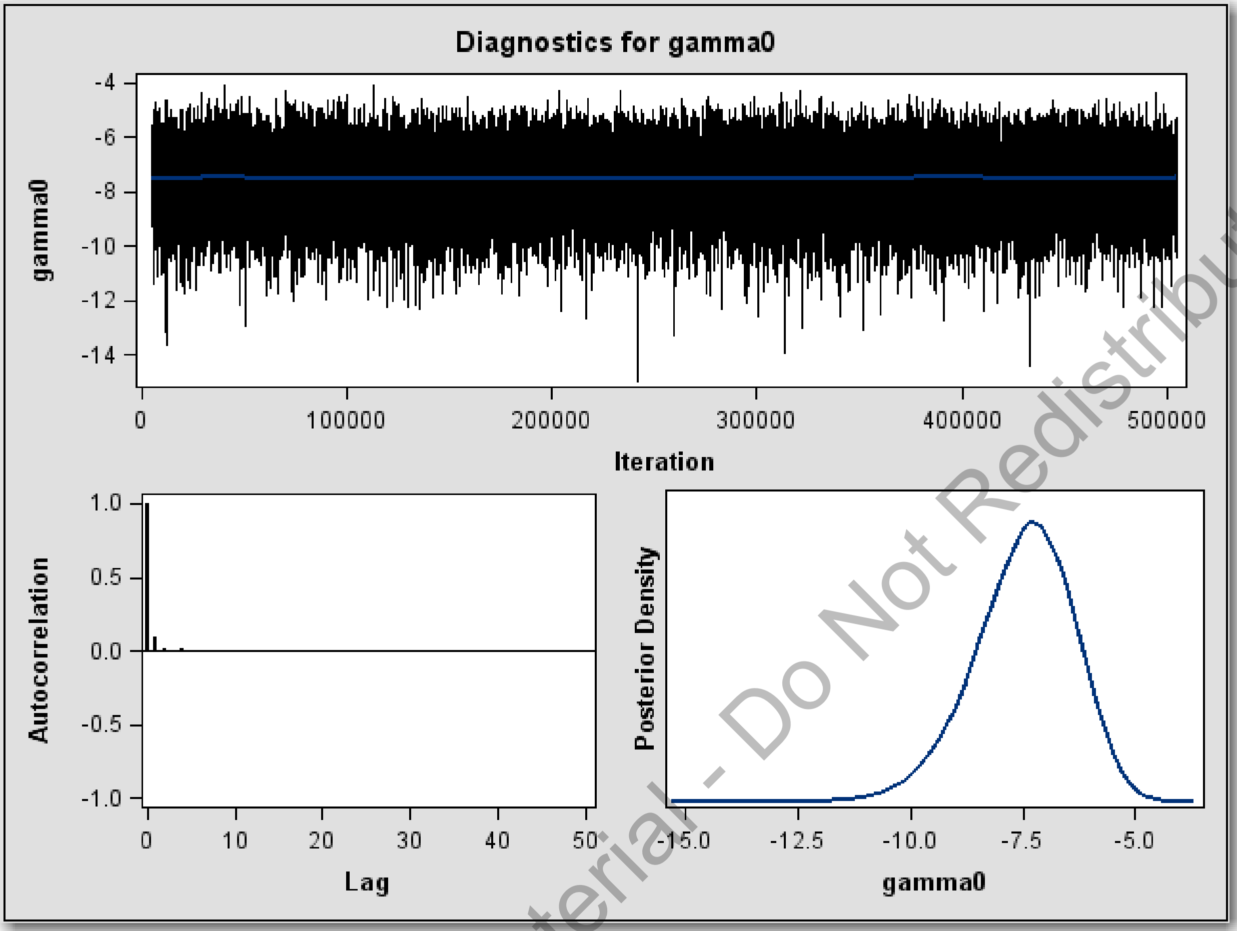
Partial Graphics Output:



The diagnostic plots for **beta0** show no problems with Markov chain convergence.

Note: The other diagnostic plots (not shown) show patterns of convergence.

The diagnostic plots for **beta3,** not shown, show a converged Markov chain. The posterior density distribution also shows the parameter for **photo** by **bap** interaction in the Poisson regression model seems important because 0 is not in the center of the distribution.



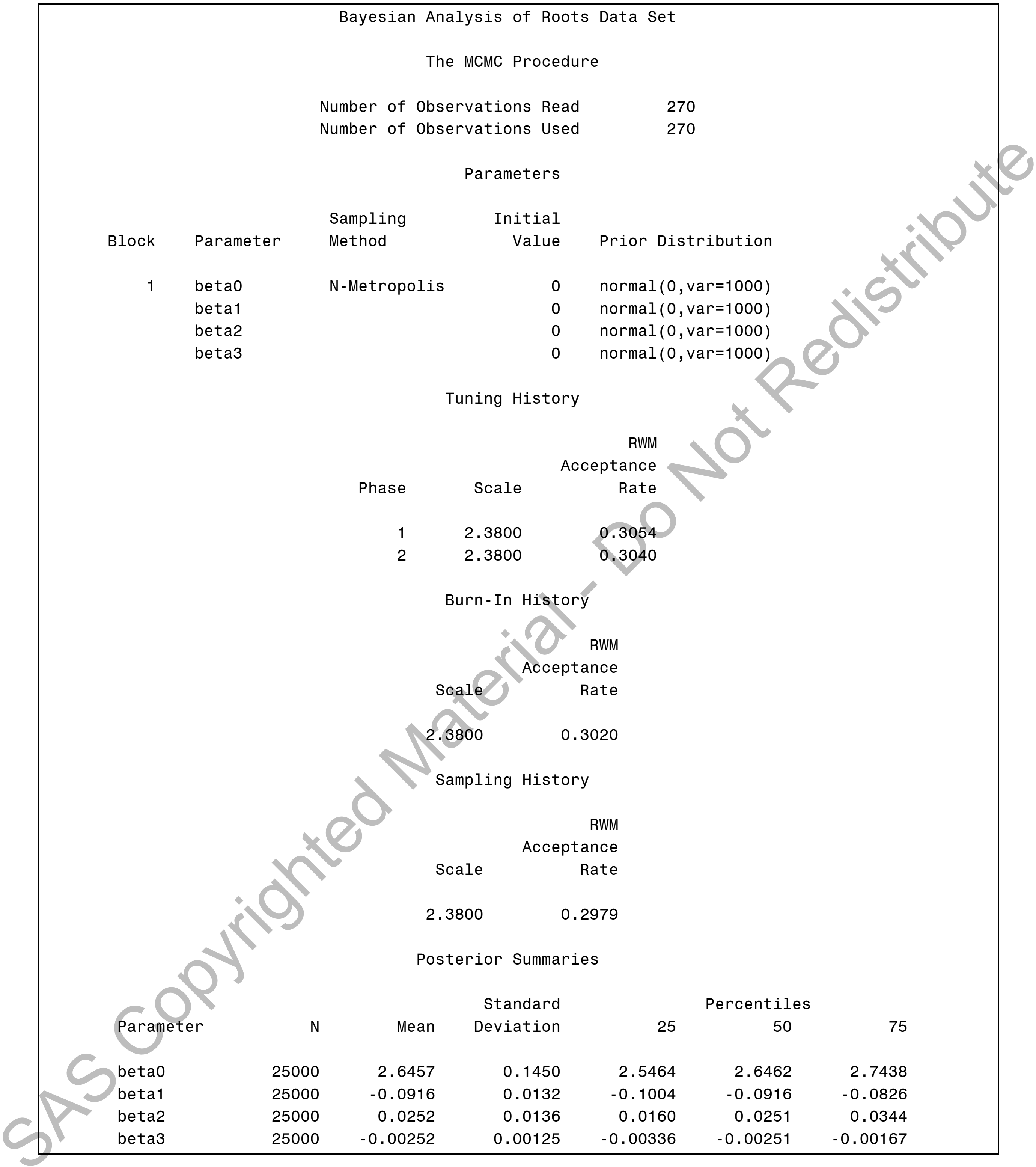
The diagnostic plots for **gamma0** show a converged Markov chain.

The diagnostic plots for **gamma1**, not shown, show a converged Markov chain. The posterior density distribution also shows the parameter for photo in the logistic regression model seems important because 0 is far from the center of the distribution.

Example: Fit a Poisson regression model on the **roots** data set and compare the DIC for the Poisson model with the DIC with the ZIP model. Use noninformative normal priors for all of the parameters and use the Poisson distribution in the MODEL statement.

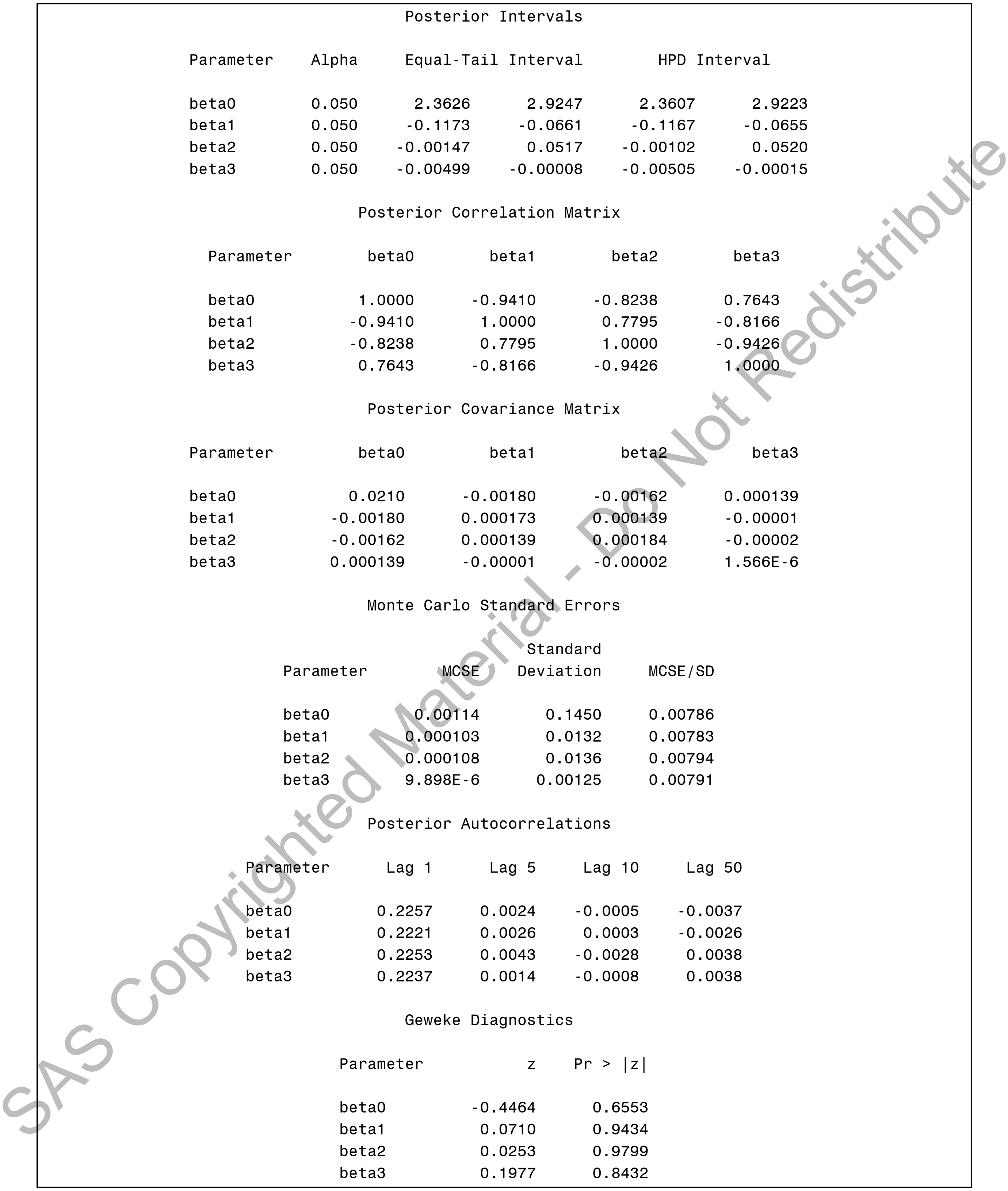
```
proc mcmc data=roots diag=all dic propcov=quanew ntu=5000 nmc=250000
     mchistory=brief thin=10 plots(smooth)=all seed=27513 stats=all;
   parms (beta0 beta1 beta2 beta3) 0;
   prior beta: ~ normal(0,var=1000);
   mu=exp(beta0+beta1*photo+beta2*bap+beta3*photo_bap);
   model roots ~ Poisson(mu);
   title "Bayesian Analysis of Roots Data Set";
run;
```

The PROC MCMC option values were chosen on a trial and error basis to improve the convergence of the Markov chain.

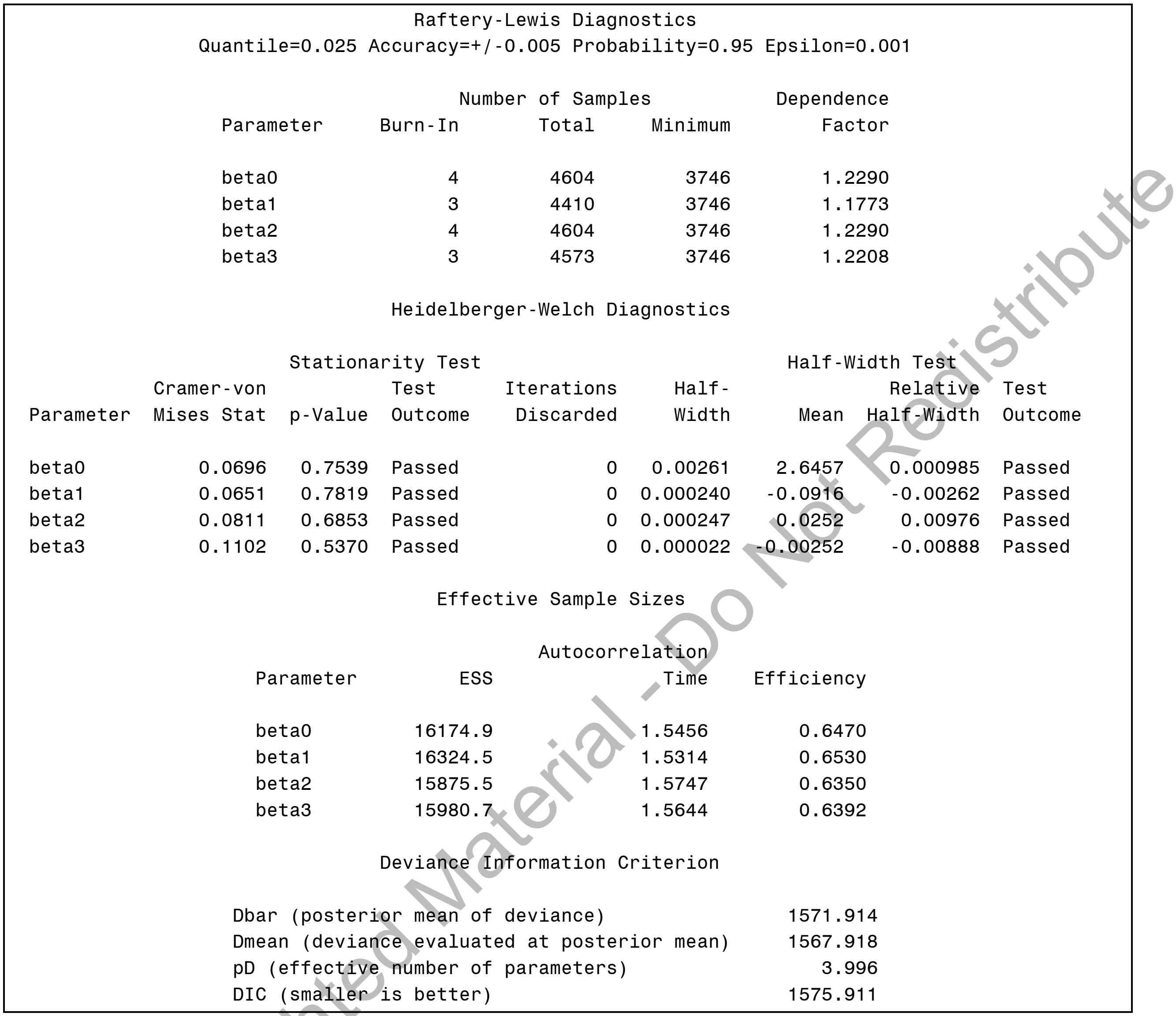


```
                        Bayesian Analysis of Roots Data Set

                                 The MCMC Procedure

                      Number of Observations Read         270
                      Number of Observations Used         270

                                     Parameters

                       Sampling         Initial
Block    Parameter     Method             Value    Prior Distribution

    1    beta0         N-Metropolis           0    normal(0,var=1000)
         beta1                                0    normal(0,var=1000)
         beta2                                0    normal(0,var=1000)
         beta3                                0    normal(0,var=1000)

                                   Tuning History

                                                      RWM
                                               Acceptance
                          Phase       Scale          Rate

                              1      2.3800        0.3054
                              2      2.3800        0.3040

                                   Burn-In History

                                                  RWM
                                           Acceptance
                                  Scale          Rate

                                 2.3800        0.3020

                                  Sampling History

                                                  RWM
                                           Acceptance
                                  Scale          Rate

                                 2.3800        0.2979

                                Posterior Summaries

                                       Standard              Percentiles
 Parameter           N        Mean    Deviation          25          50          75

 beta0           25000      2.6457       0.1450      2.5464      2.6462      2.7438
 beta1           25000     -0.0916       0.0132     -0.1004     -0.0916     -0.0826
 beta2           25000      0.0252       0.0136      0.0160      0.0251      0.0344
 beta3           25000    -0.00252      0.00125    -0.00336    -0.00251    -0.00167
```

The parameter estimates are almost identical to the results of the Poisson model fit in PROC GENMOD. The intercept (**beta0**) was 2.6474, the coefficient for photo (**beta1**) was -.0916, the coefficient for BAP (**beta2**) was .0251, and the coefficient for the interaction (**beta3**) was -.0025.



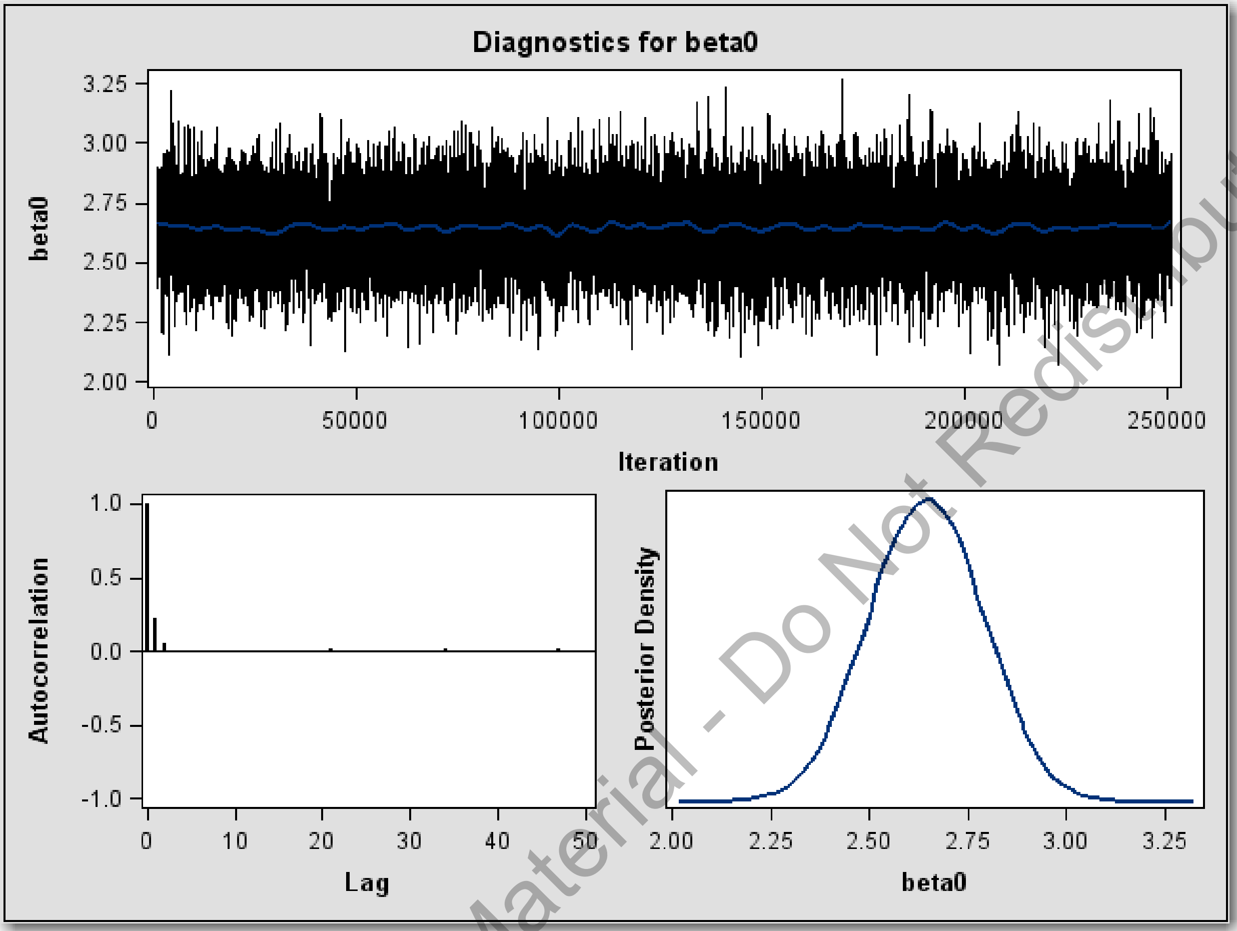
```
                          Posterior Intervals

Parameter    Alpha     Equal-Tail Interval        HPD Interval

beta0        0.050      2.3626      2.9247      2.3607      2.9223
beta1        0.050     -0.1173     -0.0661     -0.1167     -0.0655
beta2        0.050    -0.00147      0.0517    -0.00102      0.0520
beta3        0.050    -0.00499    -0.00008    -0.00505    -0.00015

                     Posterior Correlation Matrix

  Parameter        beta0        beta1        beta2        beta3

  beta0           1.0000      -0.9410      -0.8238       0.7643
  beta1          -0.9410       1.0000       0.7795      -0.8166
  beta2          -0.8238       0.7795       1.0000      -0.9426
  beta3           0.7643      -0.8166      -0.9426       1.0000

                      Posterior Covariance Matrix

Parameter         beta0         beta1         beta2         beta3

beta0            0.0210      -0.00180      -0.00162      0.000139
beta1          -0.00180      0.000173      0.000139      -0.00001
beta2          -0.00162      0.000139      0.000184      -0.00002
beta3          0.000139      -0.00001      -0.00002      1.566E-6

                      Monte Carlo Standard Errors

                                     Standard
          Parameter        MCSE     Deviation     MCSE/SD

          beta0         0.00114        0.1450     0.00786
          beta1        0.000103        0.0132     0.00783
          beta2        0.000108        0.0136     0.00794
          beta3        9.898E-6       0.00125     0.00791

                      Posterior Autocorrelations

      Parameter      Lag 1      Lag 5     Lag 10     Lag 50

      beta0         0.2257     0.0024    -0.0005    -0.0037
      beta1         0.2221     0.0026     0.0003    -0.0026
      beta2         0.2253     0.0043    -0.0028     0.0038
      beta3         0.2237     0.0014    -0.0008     0.0038

                          Geweke Diagnostics

                Parameter           z    Pr > |z|

                beta0         -0.4464      0.6553
                beta1          0.0710      0.9434
                beta2          0.0253      0.9799
                beta3          0.1977      0.8432
```



```
                              Raftery-Lewis Diagnostics
                 Quantile=0.025 Accuracy=+/-0.005 Probability=0.95 Epsilon=0.001

                                   Number of Samples          Dependence
                   Parameter     Burn-In       Total     Minimum        Factor

                   beta0               4        4604        3746        1.2290
                   beta1               3        4410        3746        1.1773
                   beta2               4        4604        3746        1.2290
                   beta3               3        4573        3746        1.2208

                               Heidelberger-Welch Diagnostics

                      Stationarity Test                           Half-Width Test
           Cramer-von            Test      Iterations     Half-              Relative  Test
Parameter  Mises Stat  p-Value  Outcome     Discarded     Width      Mean  Half-Width  Outcome

beta0          0.0696   0.7539  Passed              0   0.00261    2.6457    0.000985  Passed
beta1          0.0651   0.7819  Passed              0  0.000240   -0.0916    -0.00262  Passed
beta2          0.0811   0.6853  Passed              0  0.000247    0.0252     0.00976  Passed
beta3          0.1102   0.5370  Passed              0  0.000022  -0.00252    -0.00888  Passed

                                   Effective Sample Sizes

                                         Autocorrelation
                      Parameter         ESS             Time    Efficiency

                      beta0         16174.9           1.5456        0.6470
                      beta1         16324.5           1.5314        0.6530
                      beta2         15875.5           1.5747        0.6350
                      beta3         15980.7           1.5644        0.6392

                              Deviance Information Criterion

                    Dbar (posterior mean of deviance)               1571.914
                    Dmean (deviance evaluated at posterior mean)    1567.918
                    pD (effective number of parameters)                3.996
                    DIC (smaller is better)                         1575.911
```

The diagnostic statistics all show that the Markov chain reached convergence. The DIC for the Poisson model is much larger than the DIC for the ZIP model (1575.911 versus 1265.179). This indicates that the ZIP model is a better fitting model than the Poisson model for the **roots** data set.

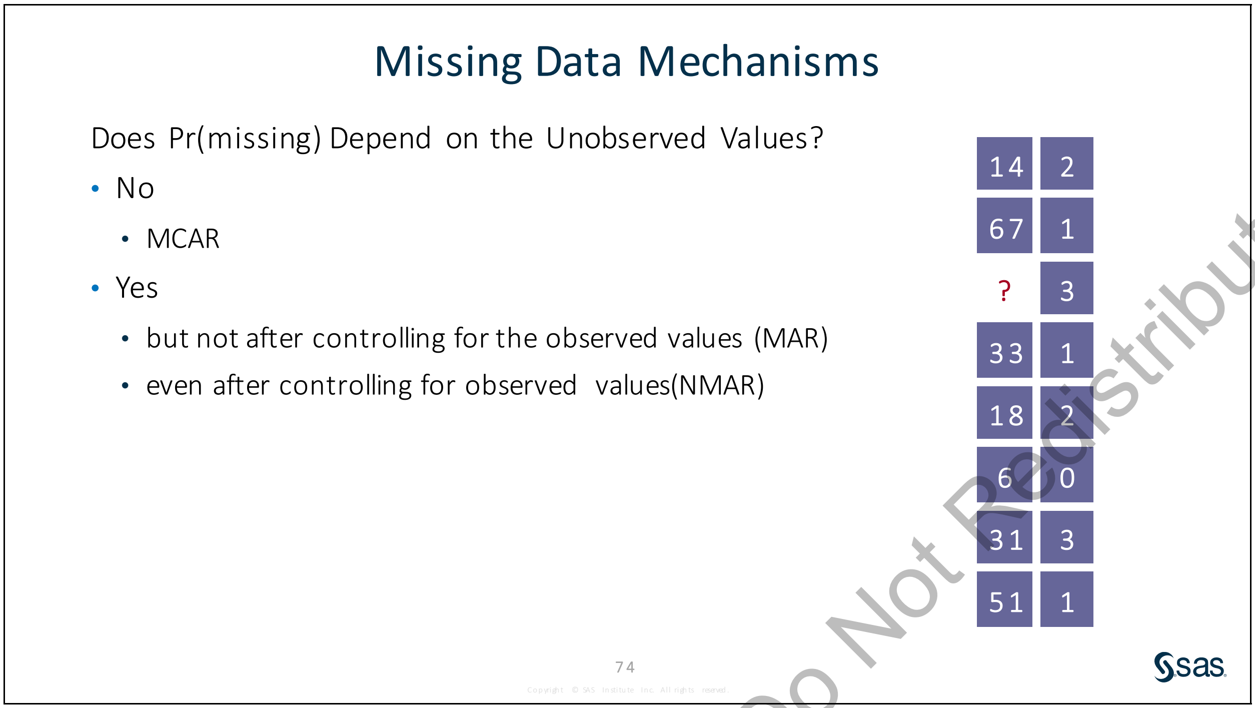
Partial Graphics Output:



The diagnostic plots for **beta0** show no problems with Markov chain convergence.

Note: The other diagnostic plots (not shown) show patterns of convergence.

End of Demonstration



Missing Data Mechanisms

Does Pr(missing) Depend on the Unobserved Values?

- No
  - MCAR
- Yes
  - but not after controlling for the observed values (MAR)
  - even after controlling for observed values(NMAR)

14	2
67	1
?	3
33	1
18	2
6	0
31	3
51	1

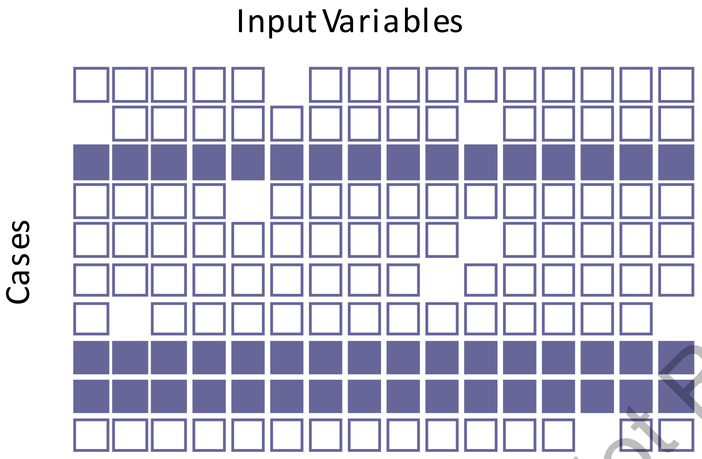
74

Missing values of the input variables can arise from several mechanisms (Little 1992). A value is missing completely at random (MCAR) if the probability that it is missing is independent of the unobserved values. The formal definition of MCAR requires that the probability of missing data on a variable X is unrelated to the values of X itself. In other words, the observed data values are a simple random sample of the values that you would have observed if the data had been complete.

A value is missing at random (MAR) if the probability that it is missing on a variable X is related to some other measured variable (or variables) in the model but does not depend on any unobserved data after controlling for the observed data. With the MAR assumption, a systematic relationship exists between one or more measured variables and the probability of missing data. MAR is sometimes referred to as ignorable missing because the missing data mechanism can be ignored and does not need to be taken into account as part of the modeling process.

A value is not missing at random (NMAR) if the probability of missing for a variable X is related to the values of X itself, even after controlling for the other variables. In other words, the probability of missing depends on the unobserved values. This is a very general scenario that assumes that the missing data mechanism is no longer ignorable and that a model for the missing data mechanism is required in order to make correct inferences about the model parameters.

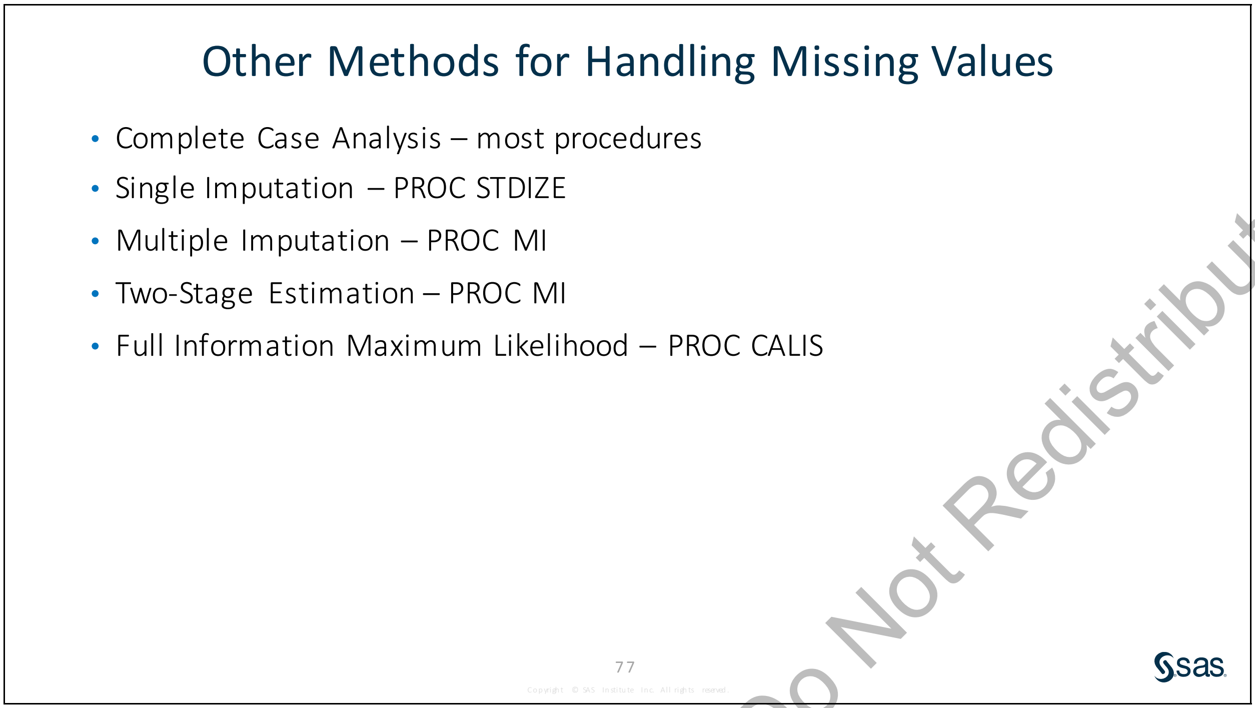
Complete Case Analysis



76

The default method for treating missing values in most SAS modeling procedures (including the MCMC procedure in SAS/STAT 14.2 and earlier releases) is complete-case analysis. In complete-case analysis, only those cases without any missing values are used in the analysis.

The shortcoming of complete-case analysis is that the missing data mechanism has to be MCAR for the parameter estimates to be unbiased. If the missing data mechanism is MAR, the parameter estimates can be biased. Furthermore, with complete case analysis even a smattering of missing values can cause an enormous loss of data in high dimensions. This sample size reduction will inflate the standard errors
of the parameters.

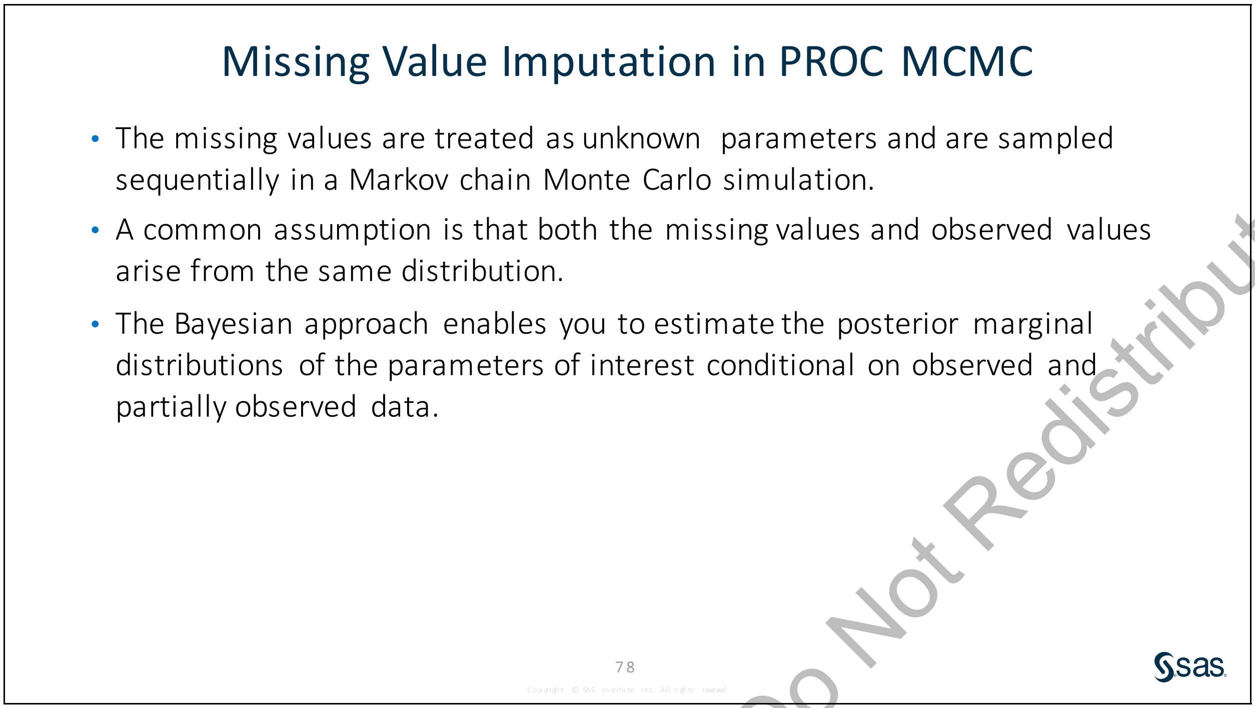


Other Methods for Handling Missing Values

- Complete Case Analysis – most procedures
- Single Imputation – PROC STDIZE
- Multiple Imputation – PROC MI
- Two-Stage Estimation – PROC MI
- Full Information Maximum Likelihood – PROC CALIS

77

Single imputation, such as replacing missing values with the observed sample mean, downwardly biases the standard errors because there is an artificially inflated distribution of values at the mean. Multiple imputation imputes missing values with several values from the estimated distribution of the variable. This method requires the use of several versions of the data, multiple analyses, and recombining parameter estimates to obtain correct inferences. The two-stage estimation approach first obtains maximum likelihood estimates of the mean vector and covariance matrix. The approach then uses these estimates as the input to the analysis. Finally, the full information maximum likelihood method uses a modified log-likelihood function that maximizes the likelihood over cases with both complete and incomplete data. This maximum likelihood estimation method accounts for the missingness patterns in the data.

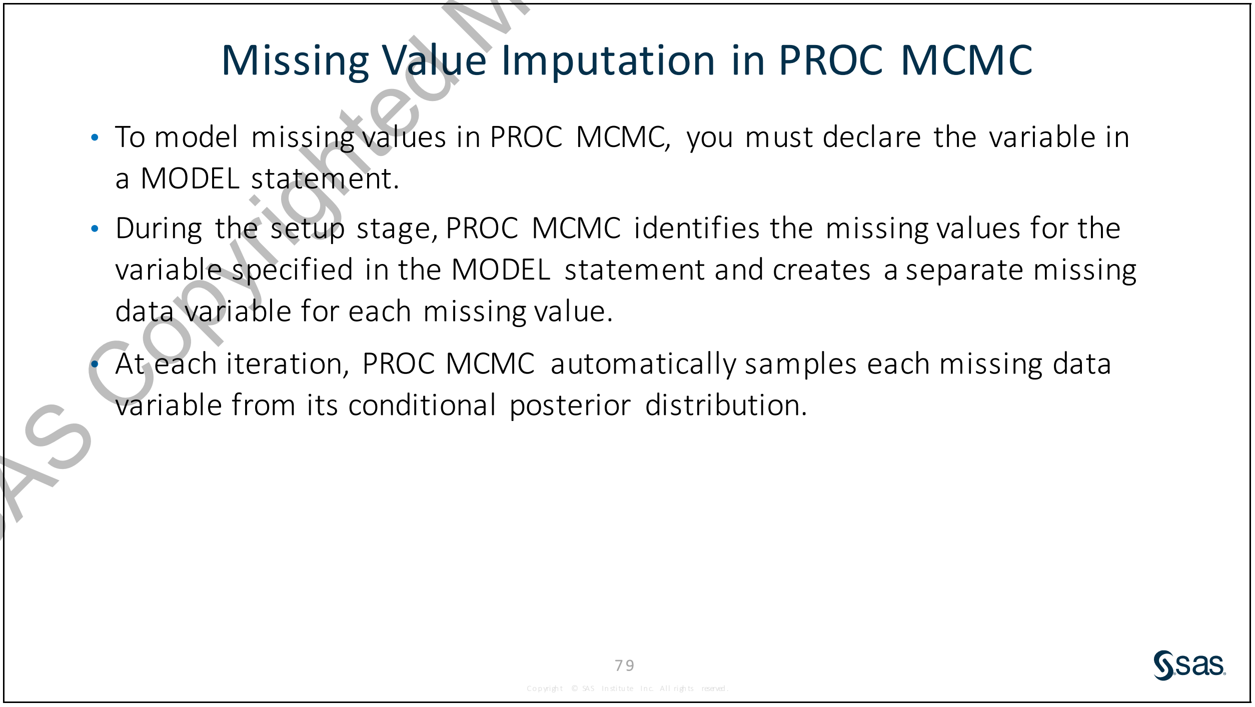


Missing Value Imputation in PROC MCMC

- The missing values are treated as unknown parameters and are sampled sequentially in a Markov chain Monte Carlo simulation.
- A common assumption is that both the missing values and observed values arise from the same distribution.
- The Bayesian approach enables you to estimate the posterior marginal distributions of the parameters of interest conditional on observed and partially observed data.

78

In SAS/STAT 14.2, PROC MCMC automatically samples all the missing values and incorporates them in the Markov chain for the parameters. You can now obtain the posterior distributions of the incomplete data given the observed data. Furthermore, you can take into account the uncertainty about the missing values and estimate the posterior marginal distributions of the parameters of interest conditional on observed (and partially observed) data. PROC MCMC can also handle various types of missing data, including data that are missing at random (MAR) and data that are not missing at random (NMAR).



Missing Value Imputation in PROC MCMC

- To model missing values in PROC MCMC, you must declare the variable in a MODEL statement.
- During the setup stage, PROC MCMC identifies the missing values for the variable specified in the MODEL statement and creates a separate missing data variable for each missing value.
- At each iteration, PROC MCMC automatically samples each missing data variable from its conditional posterior distribution.

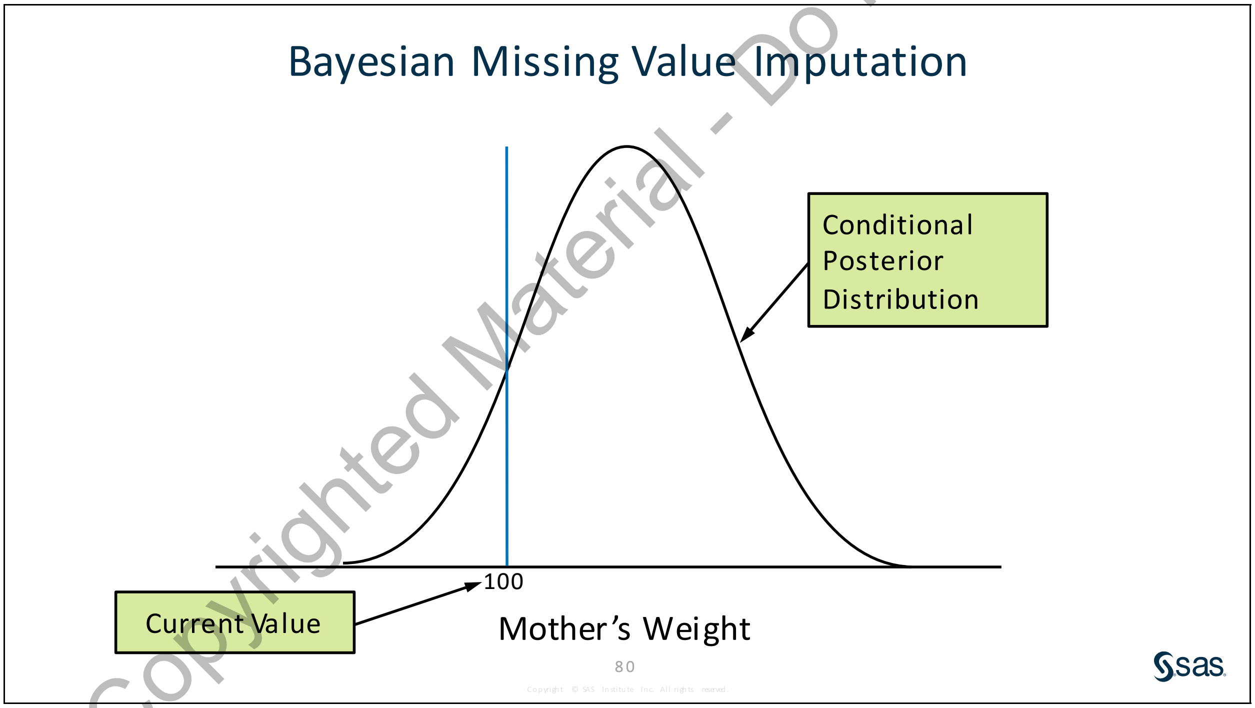
79

The distribution in the MODEL statement is the usual likelihood function for the variable with the missing values. You can think of it as the sampling distribution that generates the covariate of interest. The distribution can be a stand-alone distribution, such as a binary distribution with an unknown probability, or a more complex model that involves additional regression covariates.

The missing data variables become additional parameters in the model. The name of the missing data variable is created by concatenating the data set variable name with the observation index. Introducing these additional parameters adds limited complexity to the problem because the parameters are simply an additional layer of variables that can be sampled sequentially in an MCMC simulation. This enables you to obtain the posterior distributions of the incomplete data given the observed data.

For a response variable with missing values, the posterior distribution is the same as the likelihood function. Direct sampling algorithms are often used to draw these samples. For covariates with missing values, the posterior distribution is the product of its prior distribution and its contribution to the likelihood function. PROC MCMC resorts to scenario-specific sampling algorithms to draw these samples.

Note: If a covariate has missing values and it has not been specified in the MODEL statement, observations that contain missing values for that covariate are discarded before the analysis.



Bayesian Missing Value Imputation

Conditional Posterior Distribution

Current Value

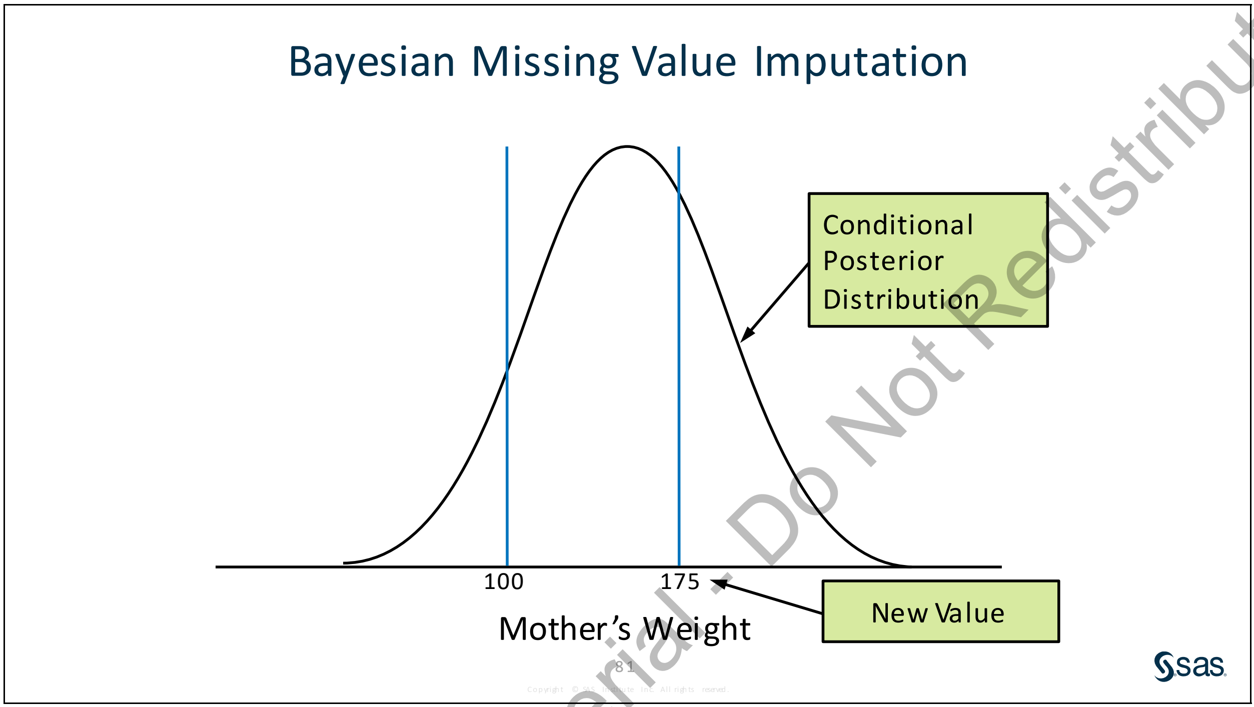
100

Mother's Weight

80

As was stated before, at each iteration of the MCMC simulation, PROC MCMC samples each missing data variable from its conditional posterior distribution just as the procedure does for all the parameters in the model. The default starting value for the missing data variable is the average of the nonmissing values of the variable. However, you can specify your own starting values with the INITIAL= option in the MODEL statement.

To illustrate the Bayesian missing value imputation, suppose that there are missing values for mother's weight in the low birth weight data set. For the missing values, PROC MCMC would use Metropolis sampling, which involves its own tuning and sampling for every missing value. In the slide above, let us say for iteration 2001 for observation number 6 (an observation with a missing value for mother's weight), the current sampled value for mother's weight is 100.



Bayesian Missing Value Imputation

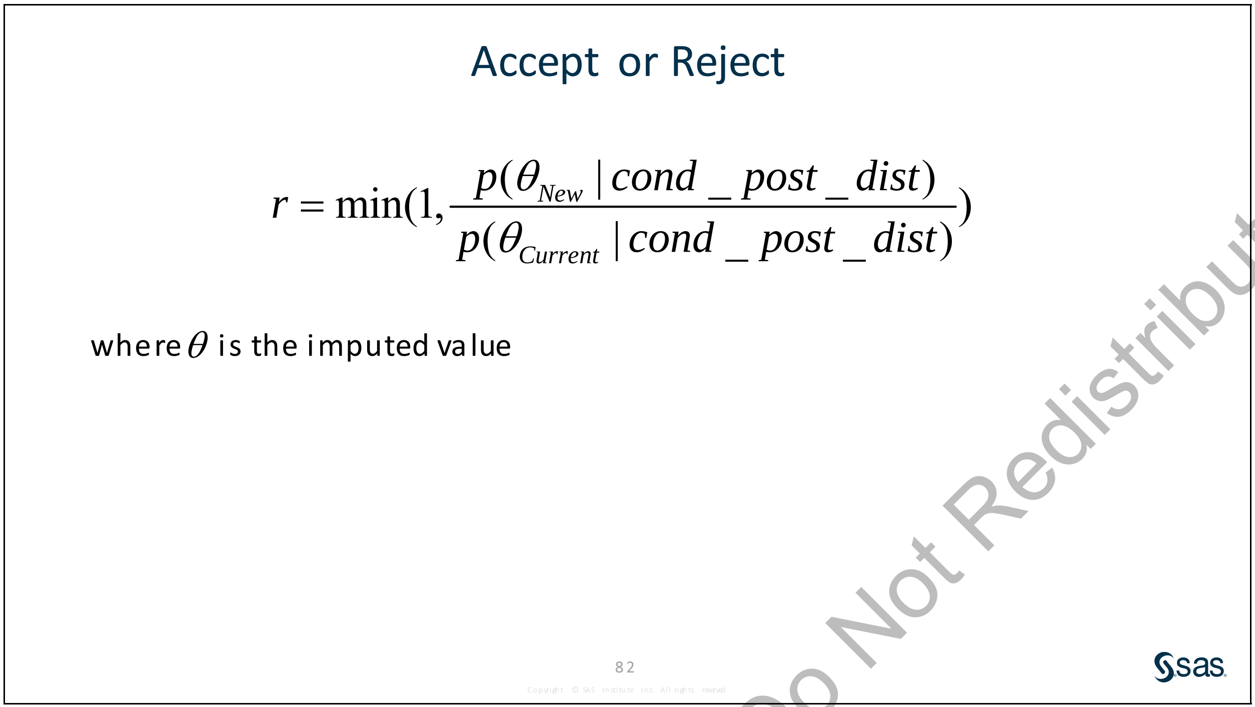
Conditional Posterior Distribution

New Value

81

For iteration 2001 for observation number 6, a candidate for mother's weight was drawn from the conditional posterior distribution. Let us say the new value is 175. The question arises: should we accept or reject the new candidate value?

It should be noted that for iteration 2002, the conditional posterior distribution for mother's weight will be different because the sampled parameters will have different values. Therefore, the shape in the slide above will be different.



Accept or Reject

$$r = \min(1, \frac{p(\theta_{New} \mid cond\_post\_dist)}{p(\theta_{Current} \mid cond\_post\_dist)})$$

where θ is the imputed value

Just like the sampling algorithm for the parameters, an r statistic is computed and if the ratio of the posterior kernels is greater than 1, the new imputed value is accepted with probability 1. If the ratio is less than 1, then a value from the uniform distribution is selected and compared with the r statistic. If the r statistic is greater than the value selected from the uniform distribution, then the new imputed value replaces the current imputed value. If the r statistic is less than the value selected from the uniform distribution, then the current imputed value is retained for usage in the next iteration.

Missing Value Imputation

Example: Examine the data set **sasuser.miss_birth** for missing values and the distribution of the values of the variables with missing values. Then fit a Bayesian model for **sasuser.miss_birth** and specify the number of iterations to 300,000, the number of iterations to use in each proposal tuning phase to 5,000, the number of burn-in iterations to 10000, the thinning rate to 10, and specify the quasi-Newton optimization in constructing the initial covariance matrix for the Metropolis-Hastings algorithm. Furthermore, create an output data set of the posterior samples of all model parameters, display all the posterior statistics, and display the Markov chain sampling history. Specify a model for the missing values for alcohol and a model for the missing values of mother's weight. Finally, compare the observed values for alcohol and mother's weight with the simulated values.

```
   /*  stbay02d05.sas  */
proc means data=sasuser.miss_birth nmiss;
   var low mother_wt alcohol prev_pretrm hist_hyp;
   title 'Variables with Missing Values';
run;
```

```
                           Variables with Missing Values

                                The MEANS Procedure

                                                                    N
            Variable       Label                                 Miss
            ---------------------------------------------------------
            low            Indicator for Birth Weight               0
            mother_wt      Weight at Last Menstrual Period         24
            alcohol        Did the mother drink during pregnancy?  26
            prev_pretrm    Previous Preterm Labors                  0
            hist_hyp       History of Hypertension                  0
            ---------------------------------------------------------
```

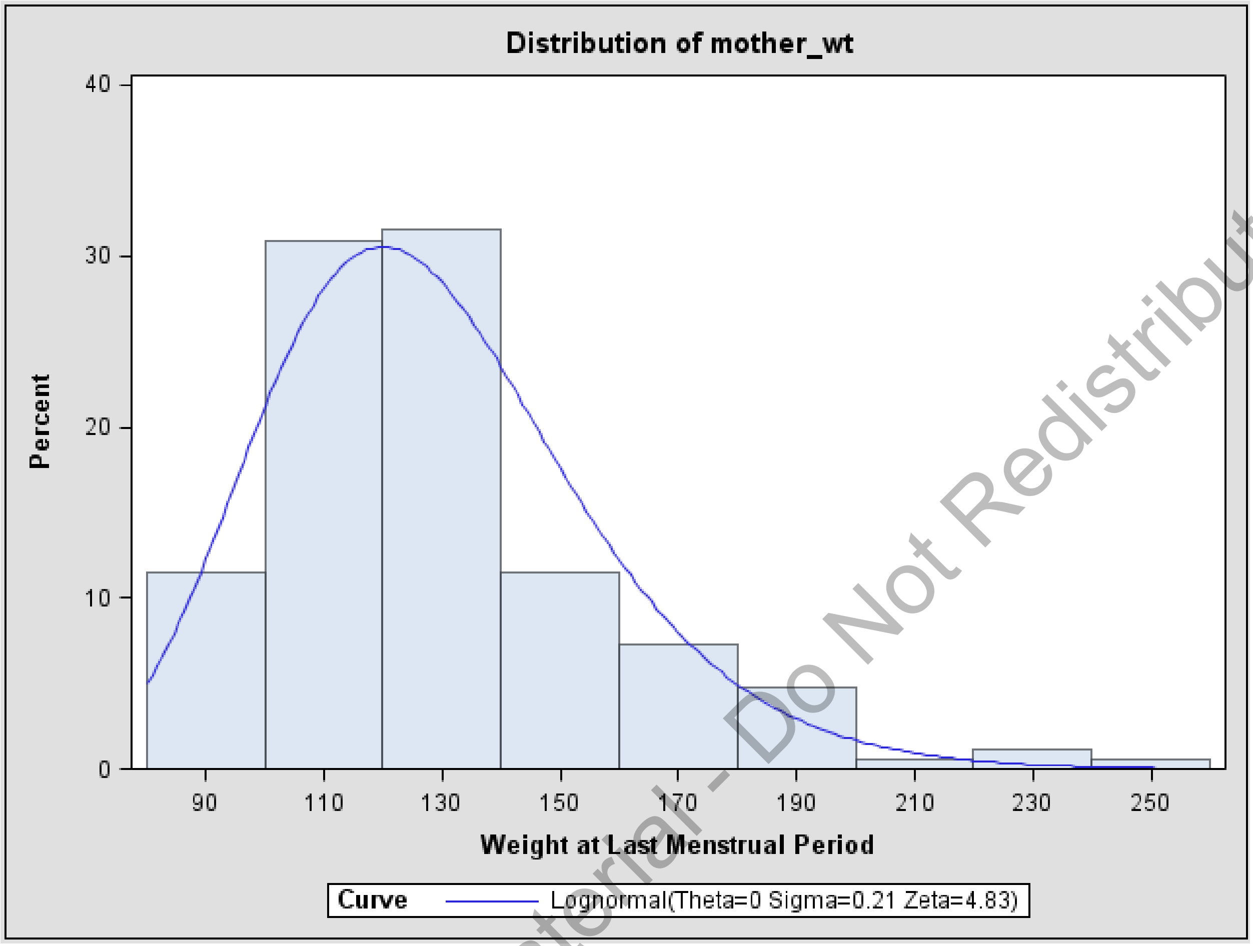
The variables mother's weight and alcohol usage has missing values. There were no missing values for the response variable low.

```
data miss_birth;
   set sasuser.miss_birth;
   if mother_wt = . then m_weight=1;
      else m_weight=0;
   if alcohol = . then m_alcohol = 1;
      else m_alcohol = 0;
run;

proc freq data=miss_birth;
   tables m_weight*m_alcohol;
   title 'Variables with Missing Values';
run;
```

```
                       Variables with Missing Values

                             The FREQ Procedure

                       Table of m_weight by m_alcohol

                    m_weight     m_alcohol

                    Frequency|
                    Percent  |
                    Row Pct  |
                    Col Pct  |       0|       1|  Total
                    ---------+--------+--------+
                           0 |    145 |     20 |    165
                             |  76.72 |  10.58 |  87.30
                             |  87.88 |  12.12 |
                             |  88.96 |  76.92 |
                    ---------+--------+--------+
                           1 |     18 |      6 |     24
                             |   9.52 |   3.17 |  12.70
                             |  75.00 |  25.00 |
                             |  11.04 |  23.08 |
                    ---------+--------+--------+
                    Total         163       26      189
                                86.24    13.76   100.00
```

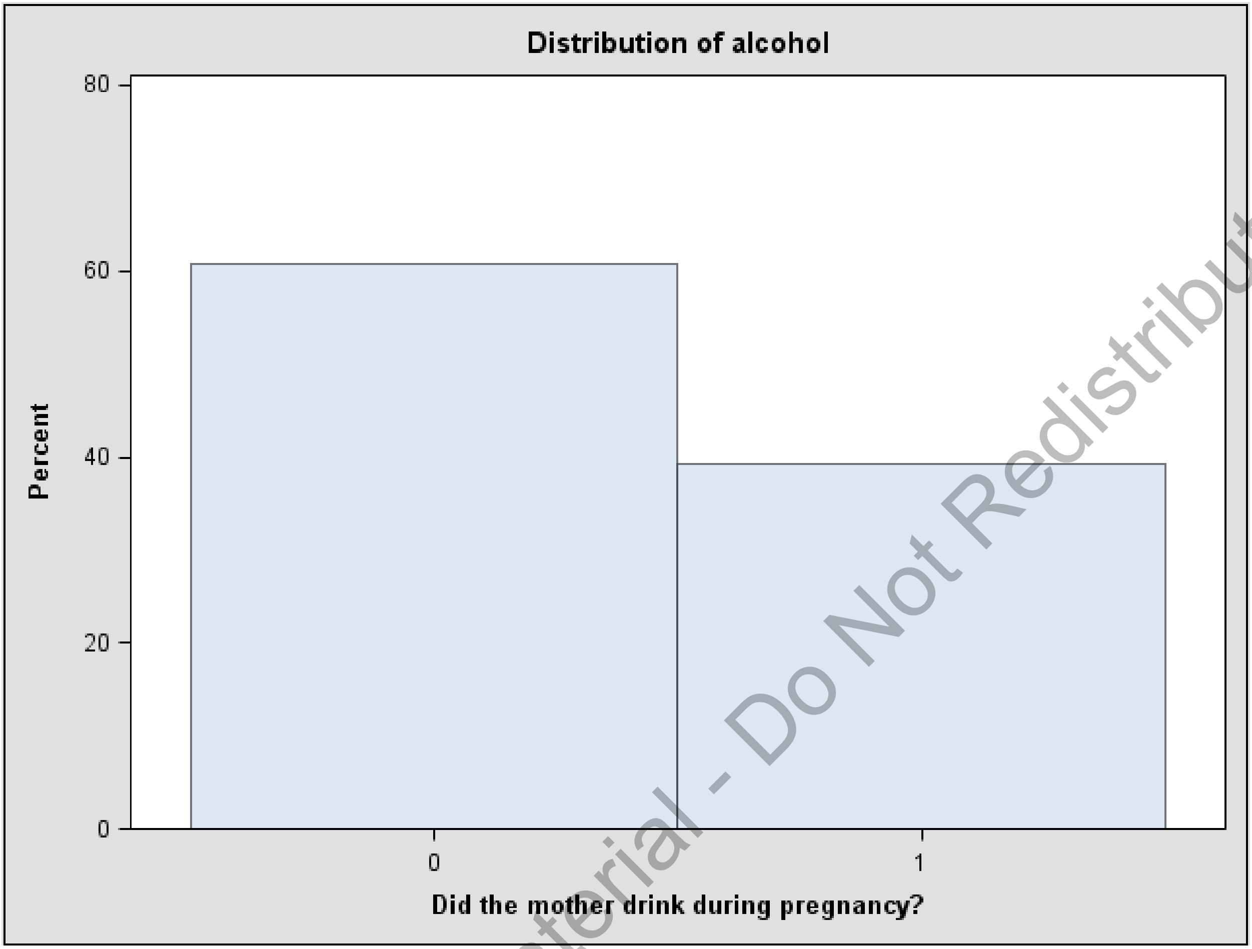
A total of six observations had missing values for both mother's weight and alcohol usage.

```
ods select histogram;
proc univariate data=sasuser.miss_birth;
   var mother_wt;
   histogram mother_wt / lognormal;
run;
```



The values of mother's weight appear to follow a lognormal distribution.

```
ods select histogram;
proc univariate data=sasuser.miss_birth;
   var alcohol;
   histogram alcohol / midpoints=0 1;
run;
```



Approximately 39% of the observed values of alcohol usage had a value of 1.

```
proc mcmc data=sasuser.miss_birth outpost=missbirthout diag=all
     propcov=quanew nbi=10000 ntu=5000 nmc=300000 thin=10
     mchistory=brief plots(smooth)=all seed=27513 statistics=all;
   parms (gamma0 gamma1 gamma2) 0;
   parms (alpha0 alpha1 alpha2 alpha3) 0;
   parms (beta0 beta1 beta2 beta3 beta4) 0;
   parms sigma2 1;
   prior gamma: alpha: beta: ~ normal(0, var=100);
   prior sigma2 ~igamma(shape=2.001, scale=1.001);
   p1=logistic(gamma0 + gamma1*hist_hyp + gamma2*prev_pretrm);
   model alcohol ~ binary(p1) monitor=(1 2 10);
   mu=alpha0 + alpha1*alcohol + alpha2*hist_hyp +
        alpha3*prev_pretrm;
   model mother_wt ~ lognormal(mu,var=sigma2) monitor=(random (3));
   p=logistic(beta0 + beta1*alcohol + beta2*hist_hyp +
                beta3*mother_wt + beta4*prev_pretrm);
   model low ~ binary(p);
   title "Bayesian Analysis of Low Birth Weight Data";
run;
```

To model missing values, you must specify the variable with missing values in a MODEL statement. The variable alcohol usage has missing values, and it will be modeled as a function of history of hypertension and previous preterm deliveries, both of which had complete data. The MODEL statement specifies that alcohol usage has a binary distribution with a parameter p1. Because alcohol usage has 26 missing values, 26 missing data variables will be created for the missing values of alcohol usage. The MONITOR= option exhibits analysis for selected missing data variables. In this example, posterior summary information for missing data variables 1, 2, and 10 will be exhibited.

The second MODEL statement models the missing values for mother's weight as a function of alcohol, history of hypertension, and previous preterm delivery. Because the observed values of mother's weight followed a lognormal distribution, the lognormal distribution will be used as the likelihood function. The MONITOR= option randomly selects three missing data variables for analysis. It should be noted that mother's weight could not be part of the model for alcohol because you need to model the missing values first before you can use the variable as a covariate in a subsequent model.

The third MODEL statement specifies the response variable low birth weight has a binary distribution with a parameter p. The results will differ from the earlier model because the posterior distributions of the parameters will be conditional on the observed and partially observed data. If we used complete case analysis, only 145 of the 189 observations would have been used in the analysis.

Partial Output:

```
                              Bayesian Analysis of Low Birth Weight Data

                                          The MCMC Procedure

                               Number of Observations Read         189
                               Number of Observations Used         189

                                    Missing Data Information Table

                                 Number of  Observation                Sampling
                  Variable     Missing Obs  Indices                    Method

                  alcohol               26  6 10 17 22 28 37 46 52      Inverse CDF
                                            58 63 69 76 82 88 92 103
                                            110 120 129 137 ...
                  mother_wt             24  3 6 9 14 17 23 28 33 42     N-Metropolis
                                            50 58 65 75 82 93 102
                                            111 118 126 138 ...
```

Because the posterior distribution for alcohol usage is a binary distribution, the inverse cumulative distribution function (CDF) sampling method is used. This sampling method is a direct and efficient sampling method for binary variables. The Metropolis sampling method is used for mother's weight because the posterior distribution is the product of the lognormal and binary densities, and no direct sampling methods are available.

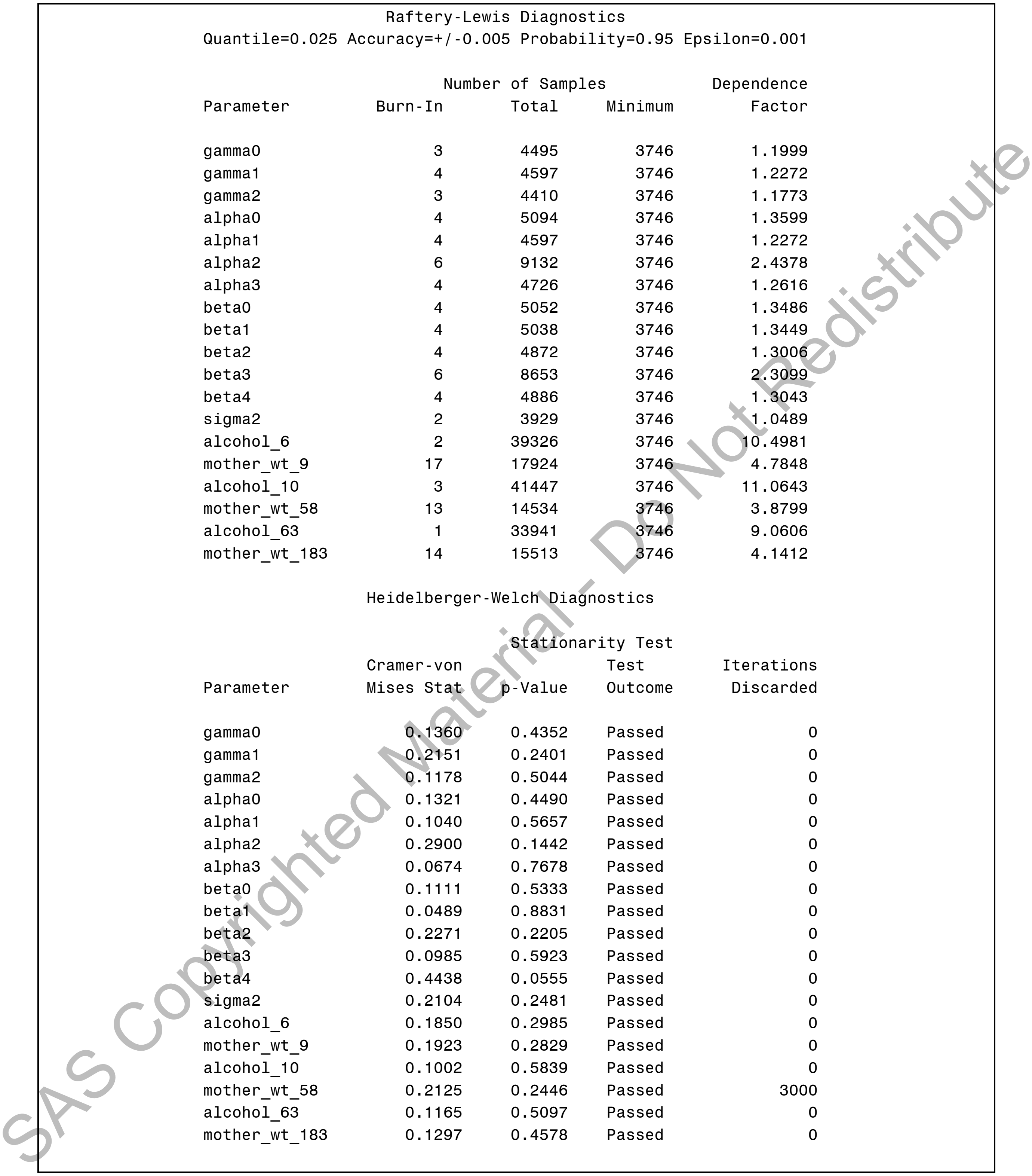
```
                                             Parameters

                                 Sampling         Initial
         Block     Parameter     Method             Value    Prior Distribution

             1     gamma0        N-Metropolis           0    normal(0, var=100)
                   gamma1                               0    normal(0, var=100)
                   gamma2                               0    normal(0, var=100)
```

```
       2    alpha0        N-Metropolis          0    normal(0, var=100)
            alpha1                              0    normal(0, var=100)
            alpha2                              0    normal(0, var=100)
            alpha3                              0    normal(0, var=100)
       3    beta0         N-Metropolis          0    normal(0, var=100)
            beta1                               0    normal(0, var=100)
            beta2                               0    normal(0, var=100)
            beta3                               0    normal(0, var=100)
            beta4                               0    normal(0, var=100)
       4    sigma2        N-Metropolis     1.0000    igamma(shape=2.001, scale=1.001)

                                      Tuning History

                                    RWM Scale      RWM Acceptance Rate
                           Phase       Low      High       Low      High

                               1     2.380     2.380     0.492     0.834
                               2     4.122     13.52     0.234     0.300

                                      Burn-In History

                                RWM Scale      RWM Acceptance Rate
                                 Low      High       Low      High

                               4.122     13.52     0.236     0.299

                                     Sampling History

                                RWM Scale      RWM Acceptance Rate
                                 Low      High       Low      High

                               4.122     13.52     0.230     0.299

                                     Posterior Summaries

                                               Standard                Percentiles
      Parameter             N        Mean     Deviation          25           50           75

      gamma0            30000     -0.5812        0.1798     -0.7015      -0.5795      -0.4608
      gamma1            30000     -0.0856        0.8531     -0.6355      -0.0647       0.4734
      gamma2            30000      0.9209        0.4603      0.6108       0.9151       1.2225
      alpha0            30000      4.8471        0.0250      4.8305       4.8468       4.8638
      alpha1            30000     -0.0346        0.0395     -0.0613      -0.0347     -0.00840
      alpha2            30000      0.1238        0.0840      0.0670       0.1239       0.1808
      alpha3            30000     -0.0557        0.0490     -0.0887      -0.0558      -0.0225
      beta0             30000      1.3058        1.0331      0.6087       1.2908       1.9803
      beta1             30000      0.7036        0.3866      0.4424       0.7012       0.9652
      beta2             30000      1.8633        0.7503      1.3510       1.8396       2.3520
      beta3             30000     -0.0224       0.00822     -0.0277      -0.0221      -0.0167
      beta4             30000      1.3371        0.4550      1.0286       1.3326       1.6404
      sigma2            30000      0.0560       0.00628      0.0516       0.0555       0.0599
      alcohol_6         30000      0.4908        0.4999           0            0       1.0000
      mother_wt_9       30000       136.5       31.4484       113.9        132.7        155.9
      alcohol_10        30000      0.4205        0.4936           0            0       1.0000
      mother_wt_58      30000       135.9       31.8738       113.5        132.8        154.2
      alcohol_63        30000      0.3310        0.4706           0            0       1.0000
      mother_wt_183     30000       134.6       32.1761       111.6        130.6        153.6
```

Compared to an earlier model, the standard deviations for this model all increased. The parameters for the intercept (beta0), alcohol (beta1), and mother's weight (beta3) showed the biggest percent differences from the earlier model.

```
                               Posterior Intervals

          Parameter        Alpha     Equal-Tail Interval        HPD Interval

          gamma0           0.050     -0.9419     -0.2335     -0.9288     -0.2216
          gamma1           0.050     -1.8178      1.5549     -1.8008      1.5676
          gamma2           0.050      0.0266      1.8468      0.0270      1.8470
          alpha0           0.050      4.7982      4.8964      4.7971      4.8952
          alpha1           0.050     -0.1118      0.0438     -0.1150      0.0399
          alpha2           0.050     -0.0418      0.2870     -0.0387      0.2897
          alpha3           0.050     -0.1520      0.0396     -0.1521      0.0394
          beta0            0.050     -0.6622      3.3809     -0.7254      3.2906
          beta1            0.050     -0.0554      1.4634     -0.0590      1.4584
          beta2            0.050      0.4530      3.3923      0.3887      3.3186
          beta3            0.050     -0.0393    -0.00704     -0.0384    -0.00641
          beta4            0.050      0.4539      2.2440      0.4426      2.2297
          sigma2           0.050      0.0451      0.0696      0.0439      0.0681
          alcohol_6        0.050           0      1.0000           0      1.0000
          mother_wt_9      0.050     84.6759       208.2     79.6164       197.4
          alcohol_10       0.050           0      1.0000           0      1.0000
          mother_wt_58     0.050     82.5675       209.5     75.9238       198.2
          alcohol_63       0.050           0      1.0000           0      1.0000
          mother_wt_183    0.050     82.7477       207.6     78.1487       200.9

                           Monte Carlo Standard Errors

                                                 Standard
                    Parameter            MCSE    Deviation     MCSE/SD

                    gamma0            0.00132       0.1798     0.00734
                    gamma1            0.00695       0.8531     0.00814
                    gamma2            0.00340       0.4603     0.00738
                    alpha0           0.000228       0.0250     0.00911
                    alpha1           0.000346       0.0395     0.00874
                    alpha2           0.000956       0.0840      0.0114
                    alpha3           0.000399       0.0490     0.00815
                    beta0              0.0103       1.0331     0.00996
                    beta1             0.00323       0.3866     0.00836
                    beta2             0.00659       0.7503     0.00878
                    beta3            0.000084      0.00822      0.0102
                    beta4             0.00373       0.4550     0.00819
                    sigma2           0.000048      0.00628     0.00764
                    alcohol_6         0.00292       0.4999     0.00584
                    mother_wt_9        0.4614      31.4484      0.0147
                    alcohol_10        0.00304       0.4936     0.00615
                    mother_wt_58       0.3951      31.8738      0.0124
                    alcohol_63        0.00272       0.4706     0.00577
                    mother_wt_183      0.4275      32.1761      0.0133
```

```
                         Posterior Autocorrelations

            Parameter          Lag 1      Lag 5     Lag 10     Lag 50

            gamma0            0.2389     0.0046     0.0072    -0.0076
            gamma1            0.3197     0.0071     0.0041    -0.0050
            gamma2            0.2429     0.0086     0.0062     0.0105
            alpha0            0.3581     0.0371     0.0058     0.0018
            alpha1            0.3206     0.0341     0.0111    -0.0093
            alpha2            0.4147     0.1033     0.0305     0.0007
            alpha3            0.2972     0.0189     0.0160    -0.0086
            beta0             0.4126     0.0558     0.0109     0.0048
            beta1             0.3466     0.0070    -0.0094     0.0034
            beta2             0.3573     0.0296     0.0025    -0.0050
            beta3             0.4158     0.0631     0.0126     0.0009
            beta4             0.3215     0.0114     0.0017    -0.0000
            sigma2            0.1415     0.0289     0.0022    -0.0008
            alcohol_6         0.0118    -0.0024    -0.0071    -0.0007
            mother_wt_9       0.7147     0.2160     0.0453     0.0020
            alcohol_10        0.0507    -0.0022    -0.0032    -0.0023
            mother_wt_58      0.6317     0.1218     0.0119    -0.0106
            alcohol_63       -0.0013    -0.0035     0.0025     0.0066
            mother_wt_183     0.6746     0.1605     0.0167    -0.0000

                             Geweke Diagnostics

                      Parameter              z    Pr > |z|

                      gamma0           -1.9836      0.0473
                      gamma1            0.5241      0.6002
                      gamma2            0.4276      0.6690
                      alpha0            0.2240      0.8227
                      alpha1            0.7931      0.4277
                      alpha2           -1.6250      0.1042
                      alpha3            0.5590      0.5762
                      beta0            -0.6189      0.5360
                      beta1            -0.1025      0.9183
                      beta2            -1.4483      0.1475
                      beta3             0.7991      0.4242
                      beta4             1.0745      0.2826
                      sigma2           -1.2344      0.2171
                      alcohol_6        -1.3377      0.1810
                      mother_wt_9      -0.8772      0.3804
                      alcohol_10       -0.6120      0.5406
                      mother_wt_58     -2.9611      0.0031
                      alcohol_63       -1.2843      0.1990
                      mother_wt_183    -0.2630      0.7926
```

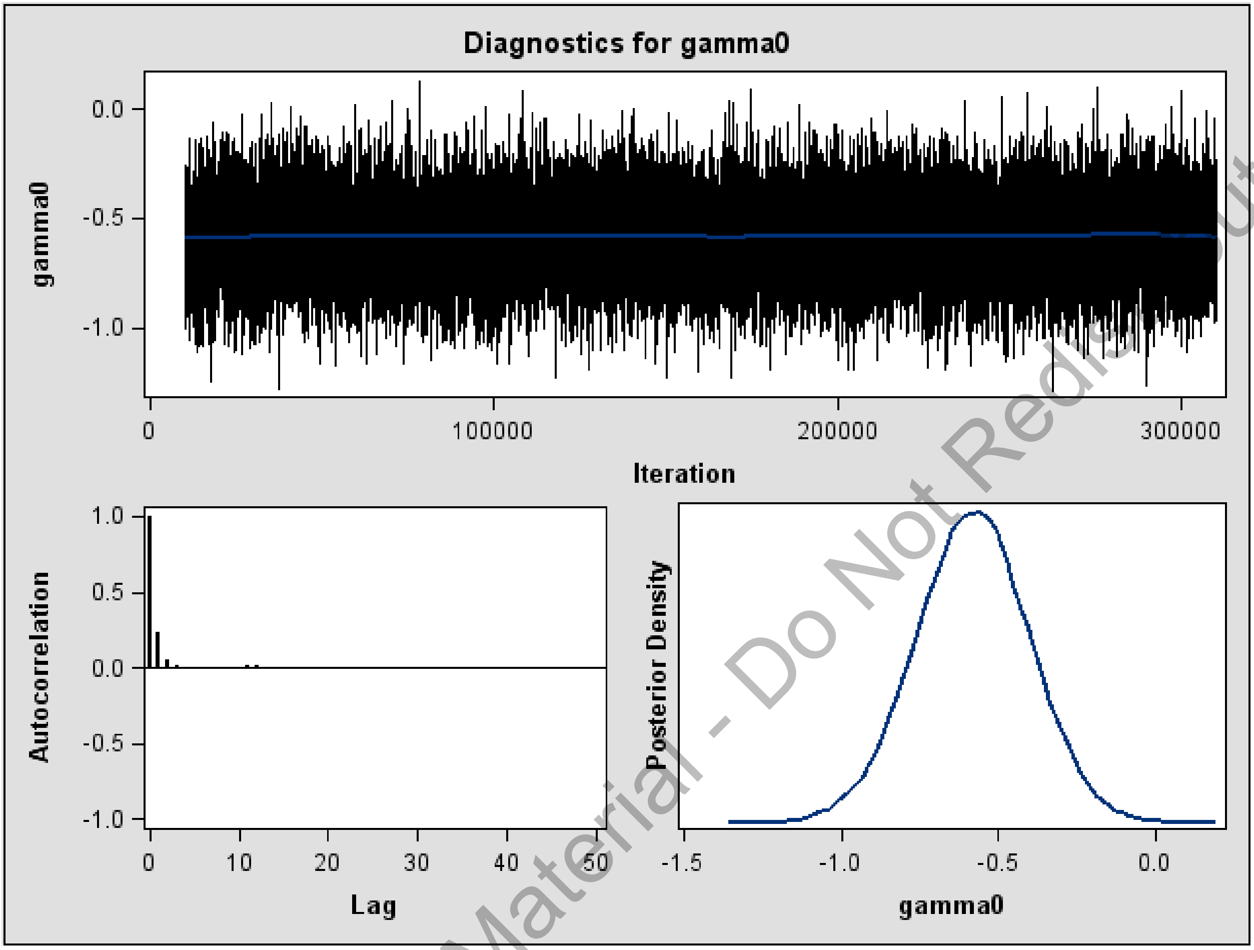


```
                       Raftery-Lewis Diagnostics
       Quantile=0.025 Accuracy=+/-0.005 Probability=0.95 Epsilon=0.001

                             Number of Samples          Dependence
       Parameter         Burn-In        Total     Minimum         Factor

       gamma0                  3         4495        3746         1.1999
       gamma1                  4         4597        3746         1.2272
       gamma2                  3         4410        3746         1.1773
       alpha0                  4         5094        3746         1.3599
       alpha1                  4         4597        3746         1.2272
       alpha2                  6         9132        3746         2.4378
       alpha3                  4         4726        3746         1.2616
       beta0                   4         5052        3746         1.3486
       beta1                   4         5038        3746         1.3449
       beta2                   4         4872        3746         1.3006
       beta3                   6         8653        3746         2.3099
       beta4                   4         4886        3746         1.3043
       sigma2                  2         3929        3746         1.0489
       alcohol_6               2        39326        3746        10.4981
       mother_wt_9            17        17924        3746         4.7848
       alcohol_10              3        41447        3746        11.0643
       mother_wt_58           13        14534        3746         3.8799
       alcohol_63              1        33941        3746         9.0606
       mother_wt_183          14        15513        3746         4.1412

                      Heidelberger-Welch Diagnostics

                                     Stationarity Test
                        Cramer-von                Test        Iterations
       Parameter        Mises Stat     p-Value    Outcome      Discarded

       gamma0               0.1360      0.4352    Passed               0
       gamma1               0.2151      0.2401    Passed               0
       gamma2               0.1178      0.5044    Passed               0
       alpha0               0.1321      0.4490    Passed               0
       alpha1               0.1040      0.5657    Passed               0
       alpha2               0.2900      0.1442    Passed               0
       alpha3               0.0674      0.7678    Passed               0
       beta0                0.1111      0.5333    Passed               0
       beta1                0.0489      0.8831    Passed               0
       beta2                0.2271      0.2205    Passed               0
       beta3                0.0985      0.5923    Passed               0
       beta4                0.4438      0.0555    Passed               0
       sigma2               0.2104      0.2481    Passed               0
       alcohol_6            0.1850      0.2985    Passed               0
       mother_wt_9          0.1923      0.2829    Passed               0
       alcohol_10           0.1002      0.5839    Passed               0
       mother_wt_58         0.2125      0.2446    Passed            3000
       alcohol_63           0.1165      0.5097    Passed               0
       mother_wt_183        0.1297      0.4578    Passed               0
```

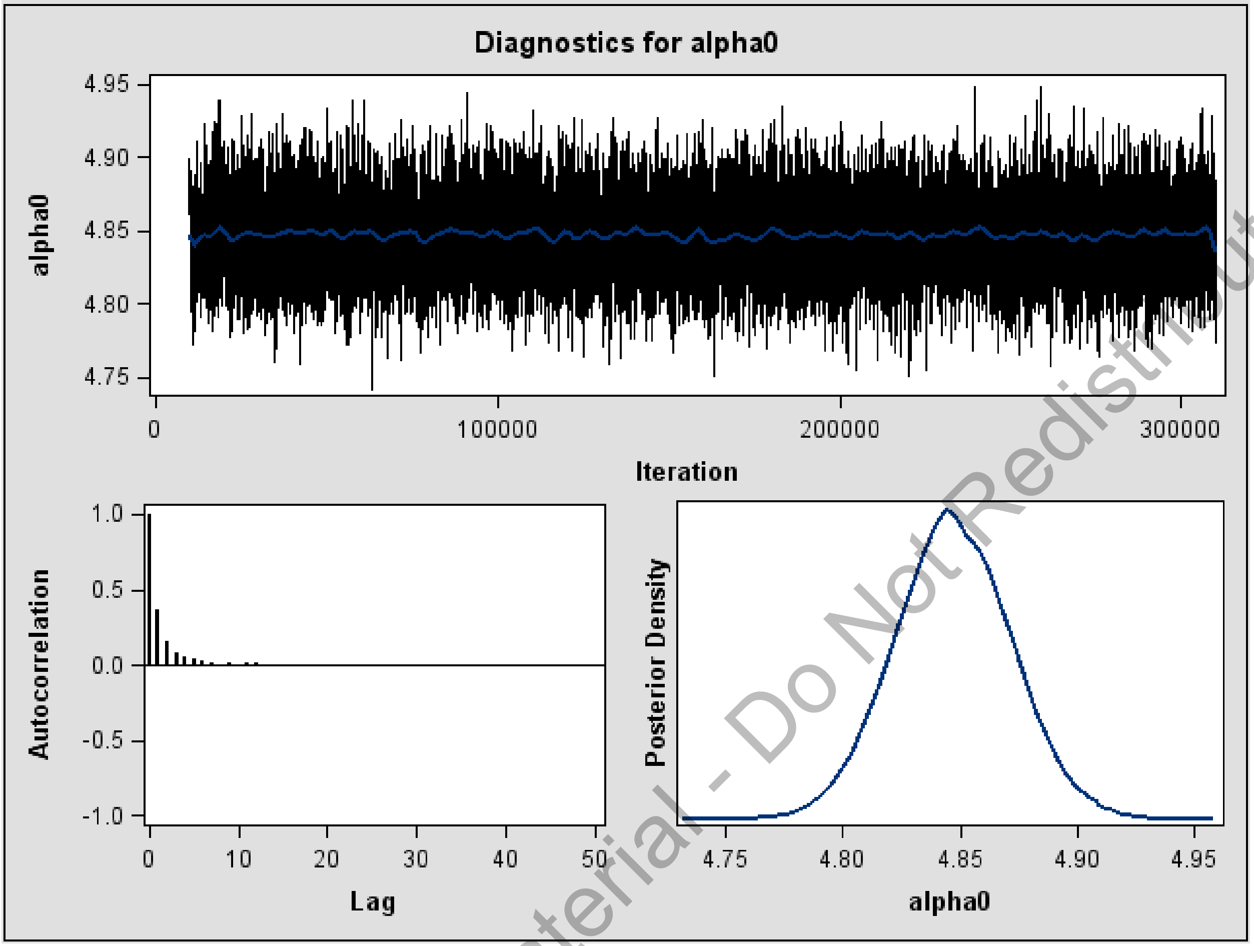
```
                         Heidelberger-Welch Diagnostics

                                      Half-Width Test
                           Half-                 Relative    Test
      Parameter            Width         Mean    Half-Width    Outcome

      gamma0             0.00250      -0.5812      -0.00430    Passed
      gamma1              0.0124      -0.0856       -0.1450    Failed
      gamma2             0.00776       0.9209       0.00843    Passed
      alpha0            0.000488       4.8471      0.000101    Passed
      alpha1            0.000708      -0.0346       -0.0205    Passed
      alpha2             0.00206       0.1238        0.0166    Passed
      alpha3            0.000970      -0.0557       -0.0174    Passed
      beta0               0.0197       1.3058        0.0151    Passed
      beta1              0.00598       0.7036       0.00850    Passed
      beta2               0.0128       1.8633       0.00687    Passed
      beta3             0.000163      -0.0224      -0.00731    Passed
      beta4              0.00868       1.3371       0.00649    Passed
      sigma2            0.000090       0.0560       0.00161    Passed
      alcohol_6          0.00572       0.4908        0.0117    Passed
      mother_wt_9         0.9677        136.5       0.00709    Passed
      alcohol_10         0.00586       0.4205        0.0139    Passed
      mother_wt_58        1.0361        136.2       0.00761    Passed
      alcohol_63         0.00571       0.3310        0.0173    Passed
      mother_wt_183       0.7984        134.6       0.00593    Passed

                              Effective Sample Sizes

                                             Autocorrelation
         Parameter              ESS                     Time    Efficiency

         gamma0             18575.1                   1.6151        0.6192
         gamma1             15078.6                   1.9896        0.5026
         gamma2             18381.4                   1.6321        0.6127
         alpha0             12043.3                   2.4910        0.4014
         alpha1             13082.7                   2.2931        0.4361
         alpha2              7720.4                   3.8858        0.2573
         alpha3             15041.1                   1.9945        0.5014
         beta0              10084.5                   2.9749        0.3361
         beta1              14309.9                   2.0965        0.4770
         beta2              12968.0                   2.3134        0.4323
         beta3               9618.4                   3.1190        0.3206
         beta4              14910.7                   2.0120        0.4970
         sigma2             17130.2                   1.7513        0.5710
         alcohol_6          29306.6                   1.0237        0.9769
         mother_wt_9         4646.0                   6.4571        0.1549
         alcohol_10         26404.1                   1.1362        0.8801
         mother_wt_58        6506.6                   4.6107        0.2169
         alcohol_63         30000.0                   1.0000        1.0000
         mother_wt_183       5666.2                   5.2946        0.1889
```

The diagnostic statistics show no indication that the Markov chain has not reached convergence.

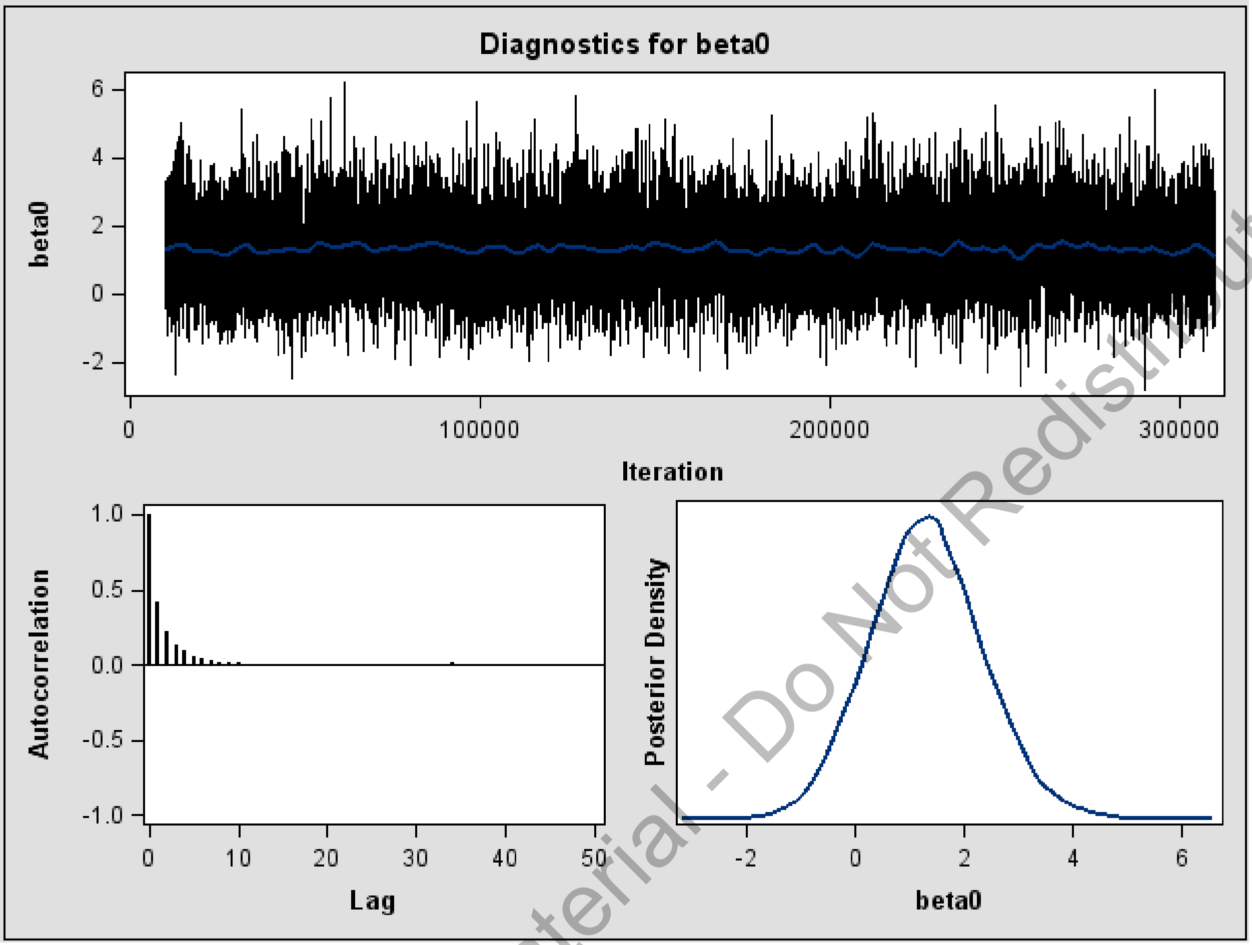
Partial Graphics Output:



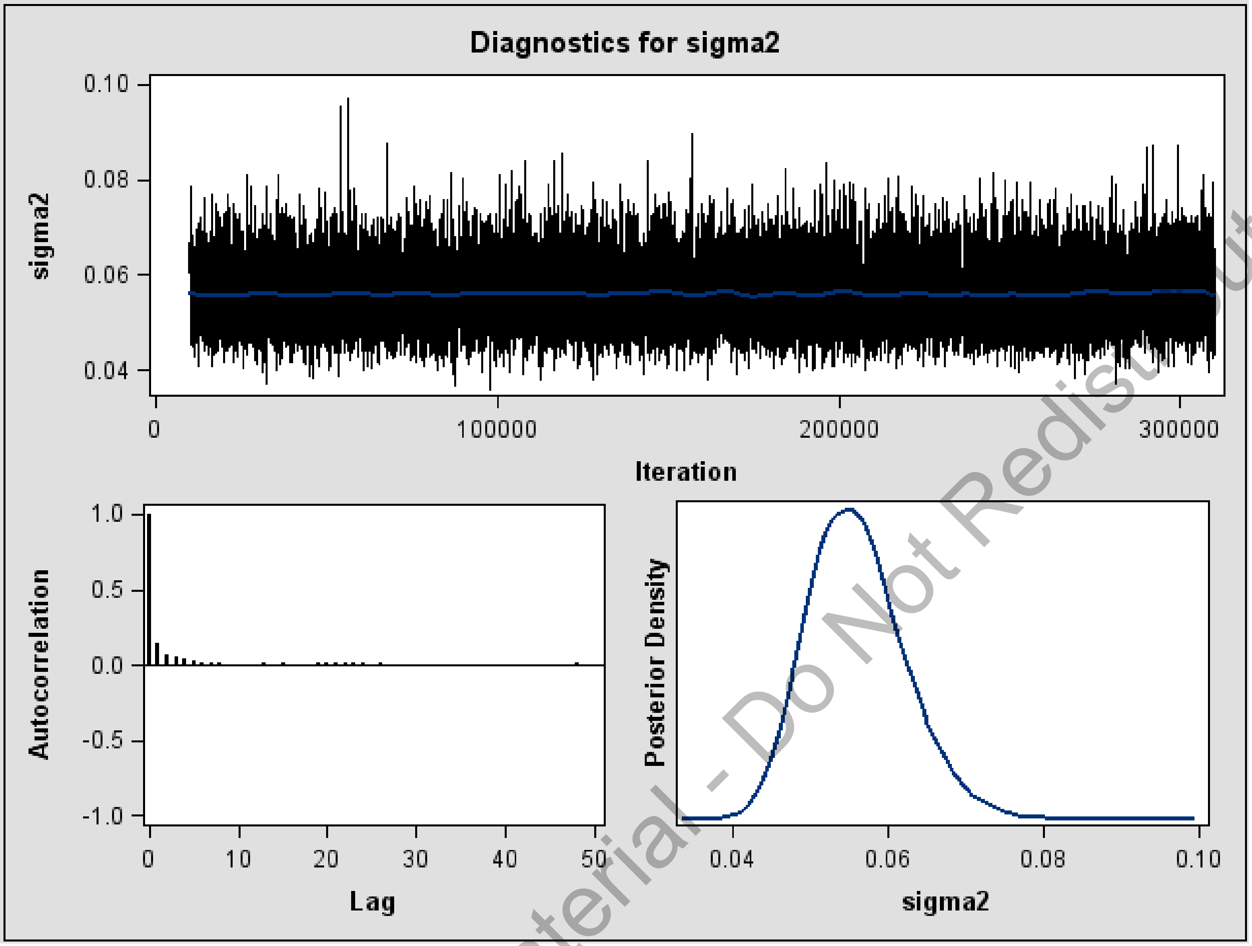
The diagnostic plots for **gamma0**, the parameter for intercept for the alcohol missing value imputation model, show no problem with convergence of the Markov chain.



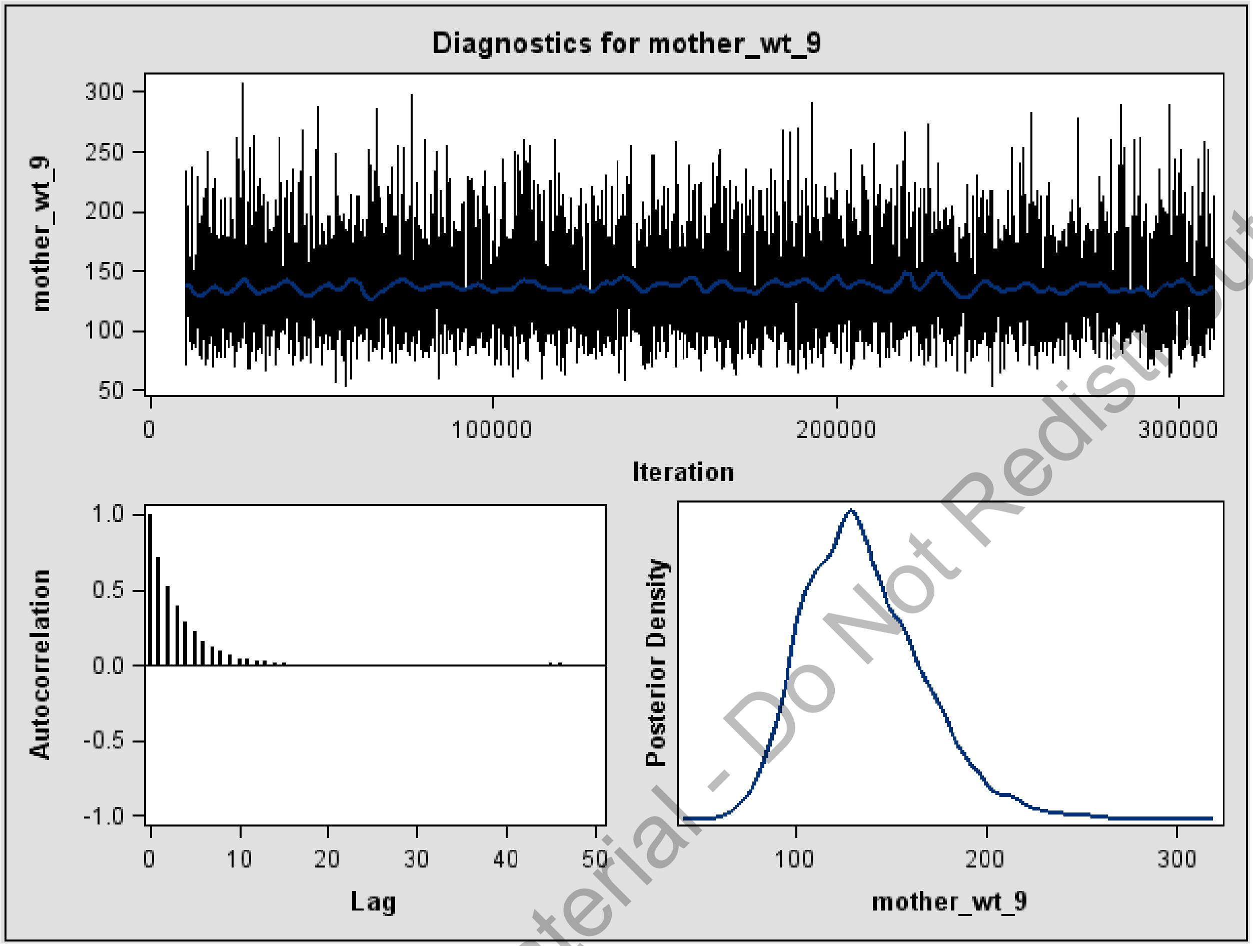
The diagnostic plots for **alpha0**, the parameter for the intercept in the mother's weight missing value imputation model, show no problem with convergence of the Markov chain.



The diagnostic plots for **beta0**, the parameter for the intercept in the low birth weight logistic model, show no problem with convergence of the Markov chain.



The diagnostic plots for **sigma2**, the parameter for the variance in the mother's weight missing value imputation model, show no problem with convergence of the Markov chain.

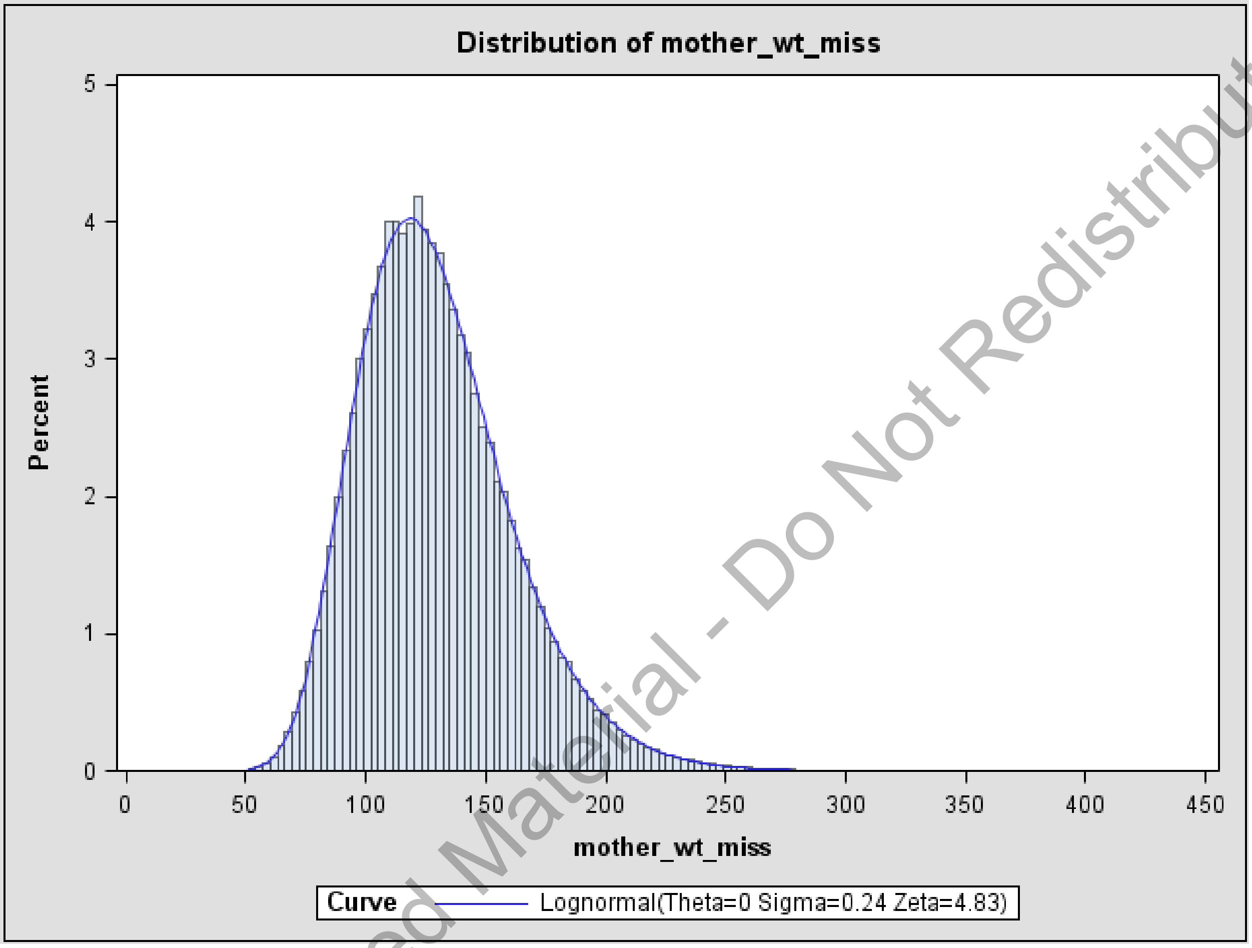


The diagnostic plots for **mother_wt_9**, the missing value variable for mother's weight for observation 9, show no problem with convergence of the Markov chain.

Note: The other diagnostic plots (not shown) show similar patterns of convergence.

```
data mweight (keep= mother_wt_miss);
   set missbirthout;
   array weights{*} mother_wt_:;
   do i=1 to dim(weights);
      mother_wt_miss=weights(i);
      output;
   end;
run;

ods select histogram;
proc univariate data=mweight;
   var mother_wt_miss;
   histogram mother_wt_miss / lognormal;
run;
```

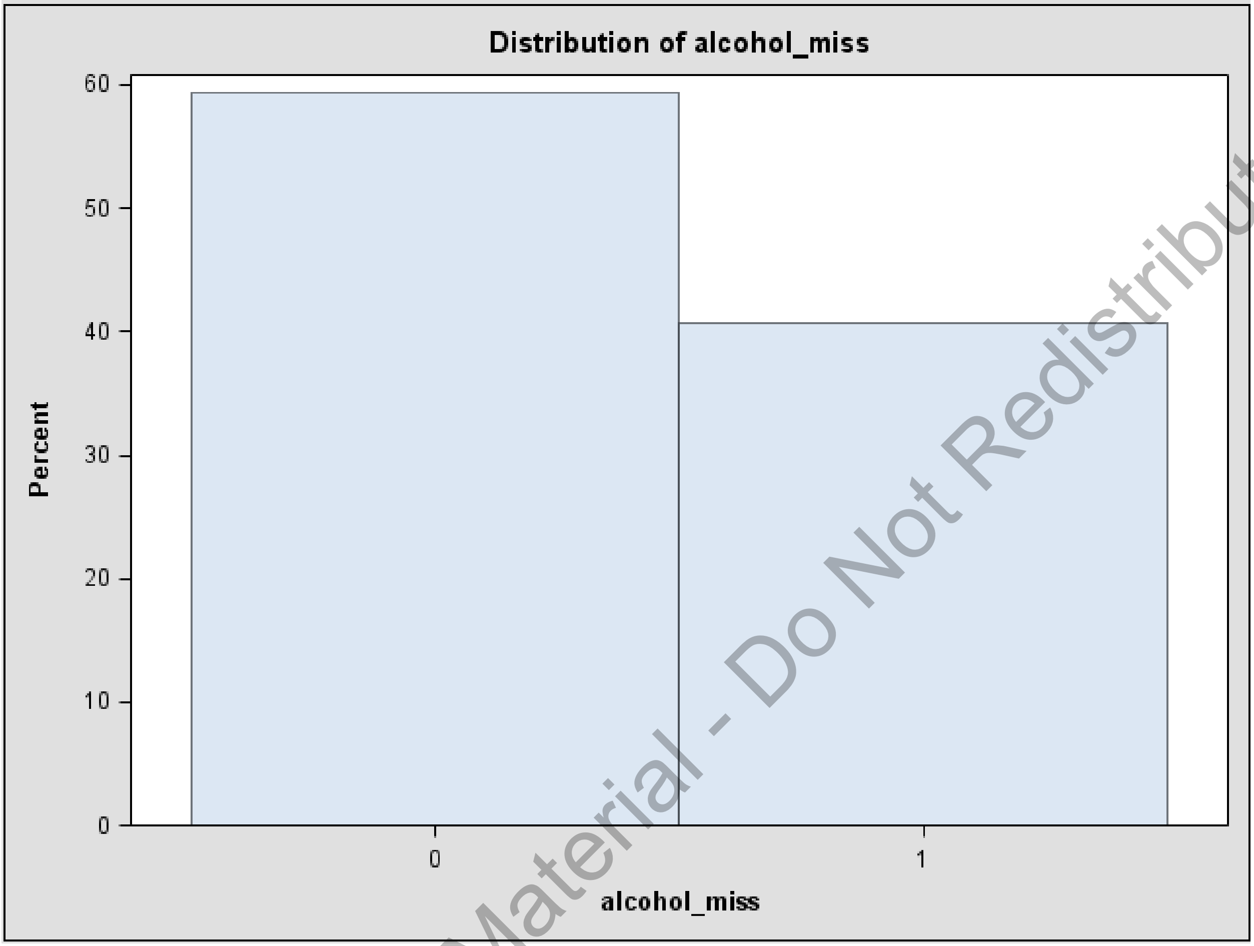
To generate the distribution of the simulated values of mother's weight, a DATA step is used to create a variable called **mother_wt_miss** that contains all the imputed values of mother's weight. The array **weights** contains all the missing value variables for mother's weight (24 variables). The DIM function returns the dimension of the array.



Compared to the distribution of the observed values of mother's weight, the parameters for the lognormal distribution are about the same (the zeta was 4.83 and sigma was 0.21).

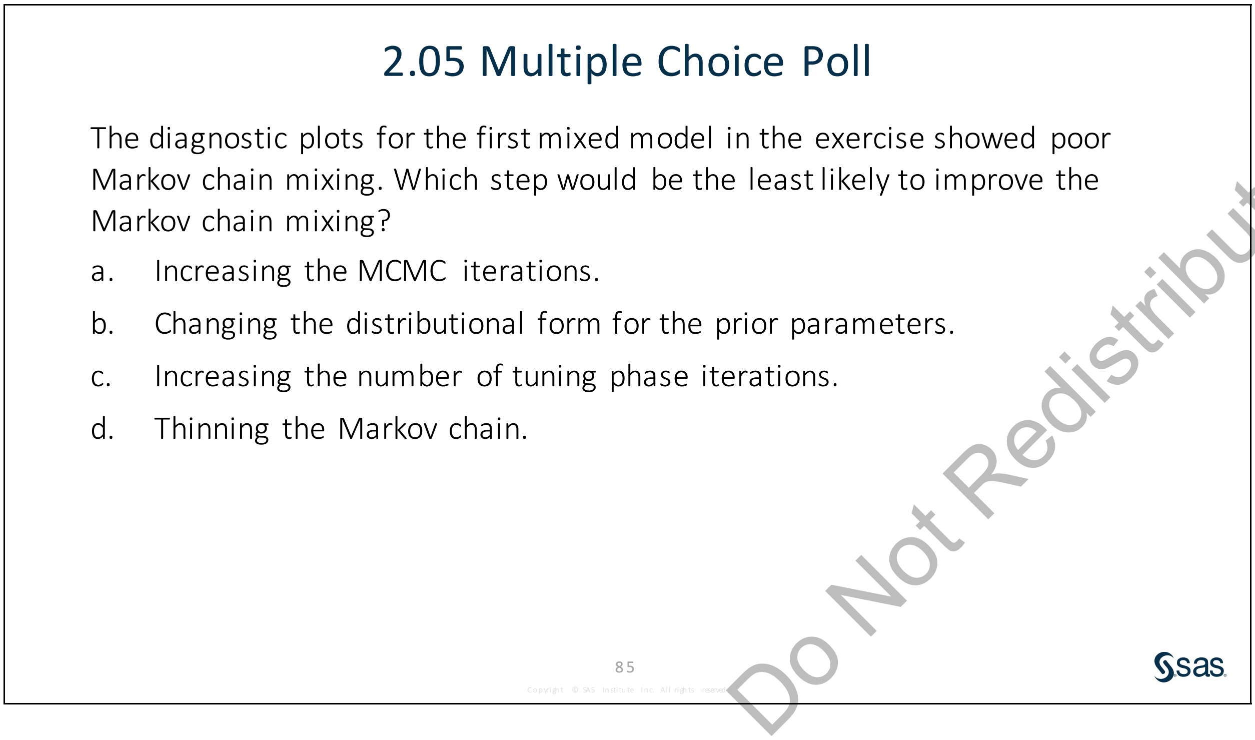
```
data malcohol (keep= alcohol_miss);
   set missbirthout;
   array alcohols{*} alcohol_:;
   do i=1 to dim(alcohols);
      alcohol_miss=alcohols(i);
      output;
   end;
run;

ods select histogram;
proc univariate data=malcohol;
   var alcohol_miss;
   histogram alcohol_miss / midpoints= 0 1;
run;
```

The distribution of the simulated values for alcohol was also generated.



The distribution of the simulated values for alcohol is about the same as the distribution of the observed values of alcohol.

End of Demonstration



2.05 Multiple Choice Poll

The diagnostic plots for the first mixed model in the exercise showed poor Markov chain mixing. Which step would be the least likely to improve the Markov chain mixing?

a. Increasing the MCMC iterations.

b. Changing the distributional form for the prior parameters.

c. Increasing the number of tuning phase iterations.

d. Thinning the Markov chain.

85

2.3 Chapter Summary

PROC MCMC is a flexible simulation-based procedure that derives inferences from simulation rather than from analytical or numerical integration methods. To use the procedure, you need to specify a likelihood function for the data and a prior distribution for the parameters. PROC MCMC then obtains samples from the corresponding posterior distributions, produces summary and diagnostic statistics, and saves the posterior samples in an output data set that can be used for further analysis.

The statements in PROC MCMC are in many ways like DATA step statements; PROC MCMC evaluates every statement in order for each observation. The syntax of programming statements used in PROC MCMC is identical to that used in the NLMIXED procedure and the MODEL procedure.

General form of the MCMC procedure:

```
PROC MCMC options;
      PARMS parameters and starting values;
      BEGINCNST;
         Programming Statements;
      ENDCNST;
      BEGINNODATA;
         Programming Statements;
      ENDNODATA;
      PRIOR parameter ~ distribution;
      RANDOM random effects specification;
      MODEL variable ~ distribution;
      PREDDIST <'label'> OUTPRED=SAS-data-
set
                   <options>;
RUN;
```

The PARMS, PRIOR, and MODEL statements form the basis of every Bayesian model. The BEGINCNST/ENDCNST and BEGINNODATA/ENDNODATA statements are used to save redundant evaluation and reduce simulation time. The RANDOM statement defines a single random effect and its prior distribution or an array of random effects and their prior distribution. The PREDDIST statement is used to generate samples from the posterior predictive distribution. In addition, you can use the ARRAY statement to define constant or parameter arrays, programming statements to specify more complicated models, and the UDS statement, which enables you to use a separate algorithm, other than the default random walk Metropolis, to update parameters in the model.

PROC MCMC uses a random walk Metropolis algorithm to obtain posterior samples. By default, PROC MCMC assumes that all observations in the data set are independent, and the logarithm of the posterior density is calculated. At each Markov chain iteration, the procedure computes the log of the posterior density by stepping through the input data set, performing the computations for each response variable value, and cumulatively adding the log-likelihood values. At the last observation, the log of the prior density is added to the sum of the log likelihood to obtain the log of the posterior density up to an additive constant.

In SAS/STAT 14.2, PROC MCMC can sample all the missing values and incorporates them in the Markov chain for the parameters. You can now obtain the posterior distributions of the incomplete data given the observed data. Furthermore, you can take into account the uncertainty about the missing values and estimate the posterior marginal distributions of the parameters of interest conditional on observed (and partially observed) data. PROC MCMC can also handle various types of missing data, including data that are missing at random (MAR) and data that are missing not at random (NMAR).

To model missing values in PROC MCMC, you must declare the variable in a MODEL statement. During the setup stage, PROC MCMC identifies the missing values for the variable specified in the MODEL statement and creates a separate missing data variable for each missing value. At each iteration, PROC MCMC samples each missing data variable from its conditional posterior distribution.