Paper SAS3042-2019

Introducing the BGLIMM Procedure for Bayesian Generalized Linear Mixed Models

Amy Shi and Fang Chen, SAS Institute Inc., Cary, NC

ABSTRACT

SAS/STAT® 15.1 includes PROC BGLIMM, a new, high-performance, sampling-based procedure that provides full Bayesian inference for generalized linear mixed models. PROC BGLIMM models data from exponential family distributions that have correlations or nonconstant variability; uses syntax similar to that of the MIXED and GLIMMIX procedures (the CLASS, MODEL, RANDOM, REPEATED, and ESTIMATE statements); deploys optimal sampling algorithms that are parallelized for performance; handles multilevel nested and non-nested random-effects models; and fits models to multivariate or longitudinal data that contain repeated measurements. PROC BGLIMM provides convenient access, with improved performance, to Bayesian analysis of complex mixed models that you could previously perform with the MCMC procedure. This paper describes how to use the BGLIMM procedure for estimation, inference, and prediction.

INTRODUCTION

A generalized linear mixed model (GLMM) is an extension of the generalized linear model (GLM) in which the linear predictor contains random effects in addition to the usual fixed effects. GLMMs also inherit from GLMs the idea of extending linear mixed models to nonnormal data. Conditional on the random effects, data have distributions in the exponential family (binary, binomial, Poisson, normal, gamma, and so on). GLMMs are widely used in practice and are especially useful in applications where the data consist of collections of units and are hierarchically structured.

The popular MIXED and GLIMMIX procedures fit GLMM models by the classical approach of maximizing a marginal likelihood function (integrated over the random effects) to estimate model parameters. PROC BGLIMM instead takes a Bayesian approach, using simulation techniques to draw samples from the joint posterior distribution of all model parameters and then using these samples to estimate and infer on quantities of interest. The direct estimation of the parameters' posterior distribution, although computationally expensive, is an essential feature of Bayesian inference, and it bypasses the dependency on asymptotic sampling distributions that is required by likelihood-based inference.

PROC BGLIMM uses a variety of sampling algorithms to draw samples from the posterior distribution of parameters. These algorithms include the conjugate sampler, direct sampler, Gamerman algorithm (a variation of the Metropolis-Hastings algorithm that is tailored to generalized linear models; see Gamerman 1997), and No-U-Turn Sampler (NUTS, a self-tuning variation of the Hamiltonian Monte Carlo (HMC) method; see Neal 2011 and Hoffman and Gelman 2014). The algorithms are parallelized to reduce run time.

Successful convergence of the Markov chain results in precise estimation of the posterior distribution (which can be summarized using point and interval estimates) that you can use to quantify uncertainties about the model parameters. PROC BGLIMM estimates linear functions of model parameters directly (via the ESTIMATE statement), and you can use the posterior samples to carry out additional posterior inferences or further analysis.

In terms of syntax, PROC BGLIMM adheres to the tradition that PROC MIXED and PROC GLIMMIX established, with similar CLASS, MODEL, RANDOM, REPEATED, and ESTIMATE statements. This provides an easy transition for SAS users who are familiar with the established conventions.

The paper is organized as follows. "Notation" provides a brief overview of GLMMs. "The BGLIMM Procedure" introduces important features, statements, and options in PROC BGLIMM. "BGLIMM Procedure Details" covers high-level simulation and algorithm details of the procedure. "Prior Distributions" discusses prior specification. "Examples" presents three examples, from simple to complex, to demonstrate how to use the procedure.

NOTATION

First consider the normal linear mixed model. The quantity of primary interest, y_i, is called the response or outcome variable for the ith individual. The distribution of y_i is normal,

$$\begin{aligned} y_i &= \mathbf{x}_i \boldsymbol{\beta} + \mathbf{z}_i \boldsymbol{\gamma}_i + \epsilon_i, \quad i = 1, \ldots, I \\ \boldsymbol{\gamma}_i &\sim N(\mathbf{0}, \mathbf{G}_i) \\ \epsilon_i &\sim N(0, \mathbf{R}_i) \end{aligned}$$

where $\boldsymbol{\beta}$ is a $p \times 1$ vector of fixed effects, $\boldsymbol{\gamma}_i$ is a $q \times 1$ vector of random effects, ϵ_i is the normal noise with a variance $\mathbf{R}_i = \sigma^2$, and $\mathbf{G}_i$ is the covariance matrix of the random effects $\boldsymbol{\gamma}_i$ ($\mathbf{G}$ is a block diagonal matrix where each block is $\mathbf{G}_i$).

When an individual i has n_i repeated measurements, the random-effects model for outcome vector $\mathbf{y}_i$ is given by

$$\mathbf{y}_i = \mathbf{X}_i \boldsymbol{\beta} + \mathbf{Z}_i \boldsymbol{\gamma}_i + \boldsymbol{\epsilon}_i$$

where $\mathbf{y}_i$ is $n_i \times 1$, $\mathbf{X}_i$ is an $n_i \times p$ design matrix of fixed covariates, $\boldsymbol{\beta}$ is a $p \times 1$ vector of fixed effects, $\boldsymbol{\gamma}_i$ is a $q \times 1$ vector of random effects, $\mathbf{Z}_i$ is an $n_i \times q$ design matrix of covariates for the $\boldsymbol{\gamma}_i$, and $\boldsymbol{\epsilon}_i$ is an $n_i \times 1$ vector of random errors. $\mathbf{R}_i$ is the covariance matrix of the residual errors for the ith subject ($\mathbf{R}$ is a block diagonal matrix where each block is $\mathbf{R}_i$).

There are cases where the relationship between the design matrices and the expectation of the response is not linear, or where the distribution for the response is far from normal, even after the data are transformed. The class of GLMMs unifies the approaches that you need in order to analyze data in those cases. Let $\mathbf{Y}$ be the collection of all $\mathbf{y}_i$, and let $\mathbf{X}$ and $\mathbf{Z}$ be the collection of all $\mathbf{X}_i$ and $\mathbf{Z}_i$, respectively. A GLMM model consists of the following:

- the linear predictor $\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\boldsymbol{\gamma}$
- the link function $g(\cdot)$ that relates the linear predictor to the mean of the outcome via a monotone link function,

  $$\mathrm{E}[Y|\boldsymbol{\beta}, \boldsymbol{\gamma}] = g^{-1}(\boldsymbol{\eta}) = g^{-1}(\mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\boldsymbol{\gamma})$$

  where $g(\cdot)$ is a differentiable monotone link function and $g^{-1}(\cdot)$ is its inverse
- a response distribution in the exponential family of distributions. The distribution can also depend on a scale parameter, ϕ.

The conditional distribution of the response variable, given $\boldsymbol{\gamma}$, is a member of the exponential family of distributions (binary, binomial, Poisson, normal, gamma, and so on).

There are two types of covariance structures: the "G-side" and the "R-side." The G-side matrix, $\mathbf{G}$, is the covariance matrix of the random effects; the R-side matrix, $\mathbf{R}$, is the covariance matrix of the residuals. By default, the $\mathbf{R}$ matrix is the scaled identity matrix, $\mathbf{R} = \phi\mathbf{I}$, where the scale parameter ϕ is set to 1 if the distribution does not have a scale parameter, such as in the case of the binary, binomial, Poisson, and exponential distributions. Models without G-side effects are also known as marginal (or population-averaged) models.

THE BGLIMM PROCEDURE

PROC BGLIMM provides the following features:

- nested or non-nested hierarchical models
- repeated-measures models (balanced or unbalanced data) with normal data
- suite of covariance structures for random effects and residuals, including variance components, compound symmetry, unstructured, AR(1), Toeplitz, autoregressive, and many more
- built-in prior distributions for regression coefficients and covariance parameters
- the ability to model heterogeneity in covariance structures

- the ability to produce estimates and credible intervals for estimable linear combination of effects
- support for missing completely at random (MCAR) and missing at random (MAR) approaches in modeling missing data
- multithreading of optimal sampling algorithms for fast performance
- the ability to save posterior samples to an output data set for use in further inferences

PROC BGLIMM uses syntax similar to that of PROC MIXED and PROC GLIMMIX in specifying a GLMM. The following three statements are the most essential:

- MODEL statement: specifies the response variable, fixed effects, likelihood function (DIST= option), and link function (LINK= option)
- RANDOM statement: specifies the random effects and the G-side variance or covariance structure (TYPE= option)
- REPEATED statement: specifies the R-side residual variance or covariance structure (TYPE= option)

More detailed descriptions of the three statements follow.

MODEL *response = fixed-effects* < / model-options>;

The MODEL statement specifies the dependent variable and fixed-effects parameters. You can also use this statement to specify the response distribution via the DIST= option and to specify the link function $g(\cdot)$ via the LINK= option. Some other useful options follow:

- NOINT excludes the fixed-effects intercept from the model.
- OFFSET= specifies the offset variable.
- COEFFPRIOR= specifies the prior of the fixed-effects coefficients.
- SCALEPRIOR= specifies the prior of the scale parameter.

You can use PROC BGLIMM to fit the likelihood functions that are listed in Table 1.

Table 1 Built-In Distribution Functions

DIST= Option Value	Distribution Function
BINARY	Binary
BINOMIAL	Binary or binomial
EXPONENTIAL | EXPO	Exponential
GAMMA | GAM	Gamma
GEOMETRIC | GEOM	Geometric
INVGAUSS | IG	Inverse Gaussian
NEGBINOMIAL | NEGBIN | NB	Negative binomial
NORMAL | GAUSSIAN | GAUSS	Normal
POISSON | POI	Poisson

The default distribution is normal for continuous variable and binomial for categorical variables. Supported link functions are shown in Table 2, and the default and other commonly used link functions for the available distributions are listed in Table 3.

Table 2 Built-In Link Functions

LINK=	**Link Function**	$g(\mu) = \eta =$
CLOGLOG | CLL	Complementary log-log	$\log(-\log(1-\mu))$
IDENTITY | ID	Identity	μ
INVERSE | RECIPROCAL	Reciprocal	$1/\mu$
LOG	Logarithm	$\log(\mu)$
LOGIT	Logit	$\log(\mu/(1-\mu))$
LOGLOG	Log-log	$-\log(-\log(\mu))$
POWERMINUS2	Power with exponent –2	$1/\mu^2$
PROBIT	Probit	$\Phi^{-1}(\mu)$

Table 3 Default and Commonly Used Link Functions

DIST=	**Default**	**Other Commonly Used**
Option Value	**Link Function**	**Link Functions**
BINARY	Logit	Probit, complementary log-log, log-log
BINOMIAL	Logit	Probit, complementary log-log, log-log
EXPONENTIAL | EXPO	Log	Reciprocal
GAMMA | GAM	Log	Reciprocal
GEOMETRIC | GEOM	Log	
INVGAUSS | IG	Reciprocal square	
NEGBINOMIAL | NEGBIN | NB	Log	
NORMAL | GAUSSIAN | GAUSS	Identity	Log
POISSON | POI	Log	

RANDOM ***random-effects*** **< / options>;**

The RANDOM statement defines the **Z** matrix of the mixed model, the random effects in the $\boldsymbol{\gamma}$ vector, and the covariance structure of the **G** matrix. You specify the SUBJECT= option to identify the subjects for the random effects and thus to set up the blocks of **G**. A set of random effects is estimated for each subject level. You define the covariance structure of **G** by using the TYPE= option. The random effects can be classification or continuous effects, and you can specify multiple RANDOM statements. You can also specify the GROUP= option to identify groups by which to vary the covariance parameters; each new level of the grouping effect produces a new set of covariance parameters.

You can specify INTERCEPT (or INT) as a random effect to indicate the intercept. PROC BGLIMM does not include the intercept in the RANDOM statement by default as it does in the MODEL statement.

Table 4 lists the supported **G**-matrix covariance types. The default is TYPE=VC.

Table 4 Covariance Structures

Structure	**Description**	**Parms**	(i,j) **Element**
ANTE(1)	Antedependence	$2t-1$	$\sigma_i\sigma_j \prod_{k=i}^{j-1} \rho_k$
AR(1)	Autoregressive(1)	2	$\sigma^2\rho^{\lvert i-j \rvert}$
ARH(1)	Heterogeneous AR(1)	$t+1$	$\sigma_i\sigma_j\rho^{\lvert i-j \rvert}$
ARMA(1,1)	ARMA(1,1)	3	$\sigma^2[\gamma\rho^{\lvert i-j \rvert -1}1(i \neq j) + 1(i=j)]$
CS	Compound symmetry	2	$\sigma_1 + \sigma^2 1(i=j)$
CSH	Heterogeneous compound symmetry	$t+1$	$\sigma_i\sigma_j[\rho 1(i \neq j) + 1(i=j)]$
FA(1)	Factor analytic	$2t$	$\Sigma_{k=1}^{\min(i,j,1)}\lambda_{ik}\lambda_{jk} + d_i 1(i=j)$
HF	Huynh-Feldt	$t+1$	$(\sigma_i^2 + \sigma_j^2)/2 - \lambda 1(i \neq j)$

Table 4 *continued*

Structure	Description	Parms	(i, j) Element
TOEP	Toeplitz	t	$\sigma_{\lvert i-j \rvert+1}$
TOEP(q)	Banded Toeplitz	q	$\sigma_{\lvert i-j \rvert+1} 1(\lvert i-j \rvert < q)$
TOEPH	Heterogeneous Toeplitz	$2t-1$	$\sigma_i \sigma_j \rho_{\lvert i-j \rvert}$
TOEPH(q)	Banded heterogeneous Toeplitz	$t+q-1$	$\sigma_i \sigma_j \rho_{\lvert i-j \rvert} 1(\lvert i-j \rvert < q)$
UN	Unstructured	$t(t+1)/2$	σ_{ij}
UN(q)	Banded	$\frac{q}{2}(2t-q+1)$	$\sigma_{ij} 1(\lvert i-j \rvert < q)$
VC	Variance components	q	$\sigma_k^2 1(i=j)$ and i corresponds to the kth effect

In Table 4, Parms refers to the number of covariance parameters in the structure, t is the overall dimension of the covariance matrix, q is the order parameter, and $1(A)$ equals 1 when A is true and 0 otherwise.

REPEATED *repeated-effect* < / options>;

The REPEATED statement specifies the **R** matrix in the model. Its syntax is similar to that of the REPEATED statement in PROC MIXED. If you omit this statement, **R** is assumed to be equal to $\sigma^2\mathbf{I}$. The REPEATED statement is available only for the normal distribution with the identity link in this release.

Specifying a *repeated-effect* is required in order to inform PROC BGLIMM of the proper location of the observed repeated responses. The *repeated-effect* must contain only classification variables. You specify the SUBJECT= option to set up the blocks of **R**. You can use the TYPE= option to define the covariance structure. The levels of the *repeated-effect* must be different for each observation within a subject; otherwise, PROC BGLIMM produces an error message.

The same collection of covariance types (Table 4) is supported in the **R** matrix. Again, the default is TYPE=VC.

Descriptions of several more useful options and statements follow.

PROC BGLIMM options;

The PROC BGLIMM statement invokes the procedure. It includes these commonly used options:

- DATA= names the input data set.
- DIC computes the deviance information criterion.
- NBI= specifies the number of burn-in iterations.
- NMC= specifies the number of iterations, excluding the burn-in iterations.
- OUTPOST= names the output data set to contain posterior samples.
- SEED= specifies the random seed for simulation.
- STATS= controls posterior statistics.

BY variable(s);

You can specify a BY statement in PROC BGLIMM to obtain separate analysis of observations in groups that are defined by the BY variables.

CLASS variable(s);

The CLASS statement names the classification variables to be used in the model. You do not need to specify the response variable in the CLASS statement if it is categorical. The CLASS statement must precede the MODEL statement. You can specify the parameterization method for the classification variables—for example, the effect or reference coding scheme.

ESTIMATE 'label' *estimate-specification* < / options>;

The ESTIMATE statement enables you to compute a custom linear combination of the parameters. PROC BGLIMM produces for $\mathbf{L}'\boldsymbol{\phi}$, where $\boldsymbol{\phi}' = (\boldsymbol{\beta}'\ \boldsymbol{\gamma}')$, an estimate (by using the posterior mean), the standard deviation (by using the posterior standard deviation), and the highest posterior density (HPD) intervals.

BGLIMM PROCEDURE DETAILS

PROC BGLIMM updates parameters conditionally, through Gibbs sampling. The fixed-effects parameters $\boldsymbol{\beta}$ are drawn jointly at each iteration, the G-side and R-side covariance parameters are updated separately, and the random-effect parameters are updated by clusters. If you omit the SUBJECT= option from a RANDOM statement, all random-effects parameters from that statement are updated jointly (see the procedure documentation for more information about how the random-effects parameters can be parameterized with or without the presence of the SUBJECT= variable). Missing response values are treated as parameters and are thus sampled along with the other parameters mentioned earlier. Each missing response value is updated by using the likelihood function as the sampling distribution, conditional on the other parameters.

Conditional Distributions

Let $\boldsymbol{\theta} = \{\boldsymbol{\beta}, \mathbf{G}, \mathbf{R}\}$, the collection of all fixed-effects parameters and the covariance matrices; let $\boldsymbol{\gamma}$ denote random-effects parameters, and let $\boldsymbol{\gamma}_j$ denote the random-effects parameters from cluster j. The treatment of random effects is identical for effects in multiple RANDOM statements.

The conditional distribution of $\boldsymbol{\beta}$ is

$$\log(p(\boldsymbol{\beta}|\boldsymbol{\gamma}, \mathrm{y}, \mathbf{R})) = \log(\pi(\boldsymbol{\beta})) + \sum_{i=1}^{n} \log(f(\mathrm{y}_i|\boldsymbol{\beta}, \boldsymbol{\gamma}, \mathbf{R}))$$

where the log-likelihood function now includes the random effects $\boldsymbol{\gamma}$. This construction reflects two PROC BGLIMM modeling settings: one in which all random-effects parameters enter the likelihood function (linearly at the mean level), and one in which the fixed-effects parameters cannot be hyperparameters of $\boldsymbol{\gamma}$ (hence no $\log(\pi(\gamma_j|\boldsymbol{\beta}))$ terms).

The conditional distribution of $\mathbf{R}$ mirrors that of $\boldsymbol{\beta}$:

$$\log(p(\mathbf{R}|\boldsymbol{\gamma}, \mathrm{y}, \boldsymbol{\beta})) = \log(\pi(\mathbf{R})) + \sum_{i=1}^{n} \log(f(\mathrm{y}_i|\boldsymbol{\beta}, \boldsymbol{\gamma}, \mathbf{R}))$$

For $\boldsymbol{\gamma}_j$, the following conditional is used:

$$\log(p(\boldsymbol{\gamma}_j|\boldsymbol{\theta}, \mathrm{y})) = \log(\pi(\boldsymbol{\gamma}_j|\mathbf{G})) + \sum_{i \in \{j\text{th cluster}\}} \log(f(\mathrm{y}_i|\boldsymbol{\beta}, \boldsymbol{\gamma}_j, \mathbf{R}))$$

This computation uses only subjects from the jth cluster. This reflects the conditional independence assumption that the RANDOM statement makes. This simplification in the calculation makes updating the random-effects parameters computationally efficient and enables PROC BGLIMM to handle random effects that contain large numbers of clusters just as easily.

The G-side covariance matrix $\mathbf{G}$ depends only on the random effects γ and not on the data or other parameters, $\boldsymbol{\beta}$ or $\mathbf{R}$,

$$\log(p(\mathbf{G}|\boldsymbol{\gamma})) = \log(\pi(\mathbf{G}))) + \sum_{j} \log(\pi(\boldsymbol{\gamma}_j|\mathbf{G}))$$

where $\pi(\mathbf{G})$ is the prior distribution of $\mathbf{G}$.

Missing Values

PROC BGLIMM treats missing response values as parameters by default and samples them in the simulation. This mechanism of modeling missing values is referred to as the missing at random (MAR) approach. You can delete all observations that contain missing values by using the MISSING=CC option in the PROC BGLIMM statement.

Suppose that

$$\mathbf{y} = \{\mathbf{y}_{\mathbf{obs}}, \mathbf{y}_{\mathbf{mis}}\}$$

The response variable $\mathbf{y}$ consists of n_1 observed values, $\mathbf{y}_{\mathbf{obs}}$, and n_2 missing values, $\mathbf{y}_{\mathbf{mis}}$. At each iteration, PROC BGLIMM samples every missing response value (by using the likelihood function as the sampling distribution). After these samples are drawn, the GLMM is reduced to a full data scenario with no missing data. PROC BGLIMM then proceeds to update $\boldsymbol{\beta}$, $\boldsymbol{\gamma}$, **G**, and **R** sequentially.

Sampling Methods

Sampling methods that PROC BGLIMM uses include the conjugate sampler, direct sampler, Gamerman algorithm (a variation of the Metropolis-Hastings algorithm that is tailored to generalized linear models), and No-U-Turn Sampler (NUTS, a self-tuning variation of the Hamiltonian Monte Carlo (HMC) method).

The conjugate sampler is used in normal models and in sampling variance and covariance parameters when conjugate priors are specified. The Gamerman algorithm is used for both the fixed-effects and random-effects parameters in nonnormal models. Missing values are sampled using direct sampling methods. The NUTS algorithm is used for covariance parameters when conjugacy is not available.

PRIOR DISTRIBUTIONS

PROC BGLIMM sets default prior distributions for all parameters in the model. You can use options in the MODEL, RANDOM, and REPEATED statements to modify prior distributions for the fixed effects, the scale parameters (in applicable likelihood functions), the G-side matrix, and the R-side matrix. The prior distribution for random effects is either univariate normal or multivariate normal, and that cannot be changed.

For fixed-effects parameters, the default prior is a flat prior, and you can use the CPRIOR= option in the MODEL statement to specify a normal prior (for example, **`cprior=normal(var=1e4)`**) or use a data set (which should contain hyperparameter mean and covariance values) to specify a multivariate normal prior (for example, **`cprior=normal(input=MyPrior)`**, where **MyPrior** is the name of the SAS data set).

The TYPE= option in the RANDOM statement controls the G-side covariance type, and the COVPRIOR= option specifies the prior distributions for the parameters in the **G** matrix. This option is applicable to the UN, UN(1), VC, and TOEP covariance types. Parameters in other G-side covariance matrix are given a flat prior distribution, and that cannot be changed in this release.

For the UN, UN(1), VC, and TOEP G-side covariance types, you can specify an inverse Wishart prior (with diagonal scale matrix), a scaled inverse Wishart prior (with diagonal scale matrix), an inverse gamma prior, a uniform prior, a half-Cauchy prior, and a half-normal prior. Among the scalar prior distributions, the uniform prior is applicable to the standard deviation, and the other priors are applicable to the variance parameters.

You use the SCALEPRIOR= option in the MODEL statement to specify a prior on the scale parameters in four distributions (likelihood functions): normal, negative binomial, gamma, and inverse gamma. You can choose an inverse gamma prior, a gamma prior, or an improper prior ($\pi(\phi) \propto 1/\phi$)), although only the inverse gamma is applicable to the normal likelihood function with the identity link.

You use the COVPRIOR= option in the REPEATED statement to specify a prior distribution on the R-side variance-covariance matrix. You can choose an inverse Wishart prior (with diagonal scale matrix) or an inverse gamma prior. When you specify the COVPRIOR= option, this prior overrides the prior that you specify in the SCALEPRIOR= option in the MODEL statement for normal data.

EXAMPLES

Example 1: Logistic Regression with Random Intercepts

This example demonstrates how you can use PROC BGLIMM to fit a mixed model to binomial data.

Researchers investigated the performance of two medical procedures in a multicenter study. They randomly selected 15 centers for inclusion. One of the study goals was to compare the occurrence of side effects from the procedures. In each center, n_{A} patients were randomly selected and assigned to treatment group A, and n_{B} patients were randomly assigned to treatment group B. The following DATA step creates the data set, **MultiCenter**, for the analysis:

```
data MultiCenter;
   input Center Group$ N SideEffect @@;
   datalines;
 1  A  32  14   1  B  33  18
 2  A  30   4   2  B  28   8
 3  A  23  14   3  B  24   9
 4  A  22   7   4  B  22  10
 5  A  20   6   5  B  21  12
 6  A  19   1   6  B  20   3
 7  A  17   2   7  B  17   6
 8  A  16   7   8  B  15   9
 9  A  13   1   9  B  14   5
10  A  13   3  10  B  13   1
11  A  11   1  11  B  12   2
12  A  10   1  12  B   9   0
13  A   9   2  13  B   9   6
14  A   8   1  14  B   8   1
15  A   7   1  15  B   8   0
;
```

The variable **Group** identifies the two medical procedures, **N** is the number of patients who received a given procedure at a particular center, and **SideEffect** is the number of patients who reported side effects.

If $y_{i\mathrm{A}}$ and $y_{i\mathrm{B}}$ denote the number of patients at center i who reported side effects for procedures A and B, respectively, then for a given center these are independent binomial random variables. To model the probability of having side effects from the two procedures, $p_{i\mathrm{A}}$ and $p_{i\mathrm{B}}$, you need to account for the fixed group effect and the random selection of centers. One possibility is to assume a model that relates group and center effects linearly to the logit of the probabilities:

$$\log\left\{\frac{p_{i\mathrm{A}}}{1-p_{i\mathrm{A}}}\right\} = \beta_{\mathrm{A}} + \gamma_i$$

$$\log\left\{\frac{p_{i\mathrm{B}}}{1-p_{i\mathrm{B}}}\right\} = \beta_{\mathrm{B}} + \gamma_i$$

In this model, β_{A} and β_{B} are fixed effects, and $\beta_{\mathrm{A}} - \beta_{\mathrm{B}}$ measures the difference in the logits of experiencing side effects; the γ_i are independent random variables that result from the random selection of centers. Observations from the same center receive the same adjustment, and these adjustments vary randomly from center to center, with variance $\mathrm{Var}[\gamma_i] = \sigma_c^2$.

Because $p_{i\mathrm{A}}$ is the conditional mean of the sample proportion, $\mathrm{E}[y_{i\mathrm{A}}/n_{i\mathrm{A}}|\gamma_i] = p_{i\mathrm{A}}$, you can model the sample proportions as binomial ratios in a generalized linear mixed model. The following statements perform this analysis under the assumption of normally distributed center effects with equal variance and a logit link function:

```
ods graphics on;
proc bglimm data=MultiCenter nmc=10000 thin=2 seed=976352
   plots=all;
   class Center Group;
   model SideEffect/N = Group / noint;
   random int / subject = Center;
run;
```

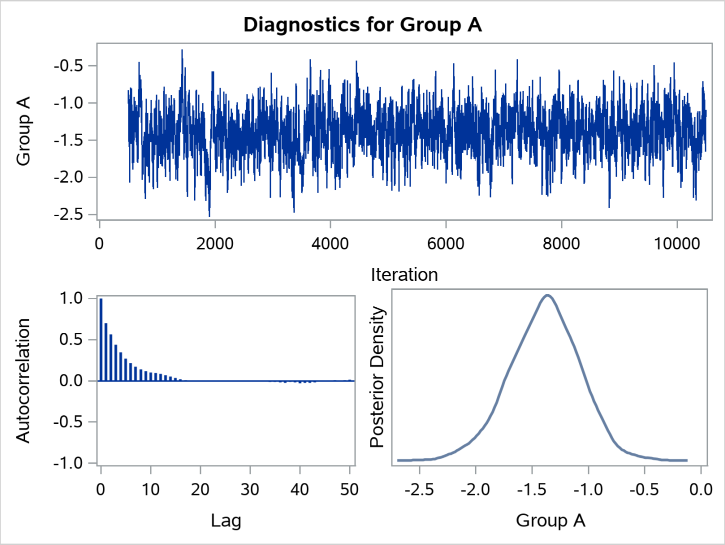
PROC BGLIMM produces posterior estimates (in the “Posterior Summaries and Intervals” table in Figure 1) for the fixed coefficients ($\boldsymbol{\beta}$) and the variance of the random center intercepts (σ_c^2). Because of the fixed-effects parameterization that is used here, the “Group A” effect is an estimate of β_{A} (–1.39), and the “Group B” effect is an estimate of β_{B} (–0.88). The two estimates show that there is a difference between the two groups. By default, posterior summary statistics of random-effects parameters are not displayed. You can display them by using the MONITOR option in the RANDOM statement.

Figure 1 Posterior Summaries and Intervals

Posterior Summaries and Intervals					
			Standard	95%	
Parameter	N	Mean	Deviation	HPD Interval	
Group A	5000	-1.3895	0.3102	-2.0071	-0.7956
Group B	5000	-0.8839	0.2968	-1.4819	-0.3186
Random Var	5000	0.9184	0.4198	0.3024	1.7515

PROC BGLIMM also produces trace plots, autocorrelation plots, and density plots of model parameters, as shown in Figure 2.

Figure 2 PROC BGLIMM Diagnostic Plots



You can use the autocall macro %TADPLOT to regenerate the same diagnostic plots for any selected parameters.

Use the ESTIMATE statement as follows to compute the log of odds ratios between the two treatment groups, A and B:

```
proc bglimm data=MultiCenter nmc=10000 thin=2 seed=976352
   outpost=CenterOut;
   class Center Group;
   model SideEffect/N = Group / noint;
   random int / subject=Center monitor;
   estimate "log OR" group 1 -1;
run;
```

The ESTIMATE statement computes $\beta_{\mathrm{A}} - \beta_{\mathrm{B}}$ for every posterior sample, and the transformed variable values are saved to the OUTPOST= data set under the variable name **Log_or**.

The table in Figure 3 shows that the posterior mean of the log of odds ratio is around –0.5, with the 95% HPD interval all negative. This indicates that patients who undergo procedure A have a lower chance of developing side effects than patients who undergo procedure B.

Figure 3 Estimated Differences in the Logits

The BGLIMM Procedure

Results from ESTIMATE Statements				
		Standard	95%	
Label	Mean	Deviation	HPD Interval	
log OR	-0.5056	0.2087	-0.9292	-0.1102

The ESTIMATE statement does not compute the difference in probabilities of side effects directly. You compute this difference by using a DATA step, where **CenterOut** is the saved OUTPOST= data set from a previous PROC BGLIMM run:

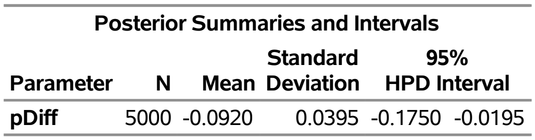
```
data prob;
   set CenterOut;
   pDiff = logistic(group_a) - logistic(group_b);
run;
```

You can use the “%SUMINT” autocall macro to compute the posterior summary statistics of **pDiff**:

```
%sumint(data=prob, var=pDiff)
```

The results are shown in Figure 4. As you can see, there is a significant difference in probabilities of side effects between the two groups.

Figure 4 Posterior Summary Statistics



Posterior Summaries and Intervals

Parameter	N	Mean	Standard Deviation	95% HPD Interval	
pDiff	5000	-0.0920	0.0395	-0.1750	-0.0195

.

Example 2: Multilevel Clinical Trial

This example illustrates how to use PROC BGLIMM to analyze hierarchical data that have nested clusters at multiple levels. It also demonstrates how you can use the deviance information criterion (DIC) to evaluate the fit of a model.

Brown and Prescott (1999) discussed a clinical trial for hypertension in which 288 patients at 29 centers were randomized to receive one of three hypertension treatments: a new drug, Carvedilol; and two standard drugs, Nifedipine and Atenolol. Patients were followed up four times, once every other week for four visits. A baseline diastolic blood pressure (DBP) was recorded before the treatment, and the DBP was recorded again at each of the four follow-up visits. One goal of this study was to assess the effect of the three treatments on DBP over the follow-up period.

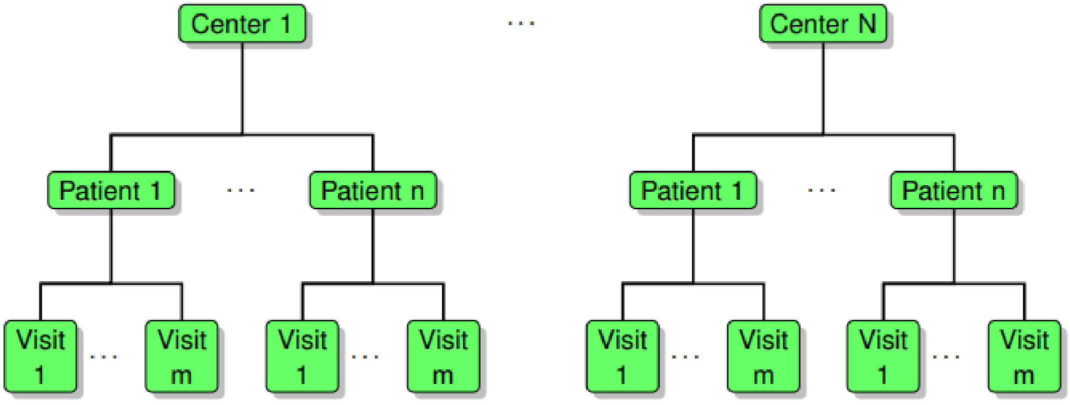
The following statements show part of the data set:

```
data DBP;
   input Patient Visit Center Treat$ DBP DBP1;
   datalines;
79 3 1 Carvedil 96 100
79 4 1 Carvedil 108 100
80 3 1 Nifedipi 82 100
80 4 1 Nifedipi 92 100
80 5 1 Nifedipi 90 100
80 6 1 Nifedipi 100 100
81 3 1 Atenolol 86 100

   ... more lines ...

237 5 41 Atenolol 80 104
237 6 41 Atenolol 90 104
238 3 41 Nifedipi 88 112
238 4 41 Nifedipi 100 112
;
```

In the INPUT statement, the variable **Patient** identifies patients; the variable **Visit** records the visit time 3, 4, 5, or 6 after randomization; the variable **Center** represents the center that each patient visited; the variable **Treat** indicates which treatment group (Carvedilol, Nifedipine, or Atenolol) each patient was assigned to; **DBP** is the diastolic blood pressure (in mmHg) measured at each follow-up visit; and **DBP1** is the baseline diastolic blood pressure measured before randomization.

As shown in Figure 5, this study has a three-level structure, where the **Visit** is the level-1 unit at the bottom of the hierarchy, the **Patient** is the level-2 unit, and the **Center** is the level-3 unit at the top. Visits are nested within patients, which are nested within centers. The units at levels higher than level 1 are sometimes called clusters. Visit time is a level-1 covariate. Baseline DBP and treatment vary only from patient to patient and are thus level-2 covariates. No level-3 covariates are measured on the centers.

Figure 5 Three-Level Structure

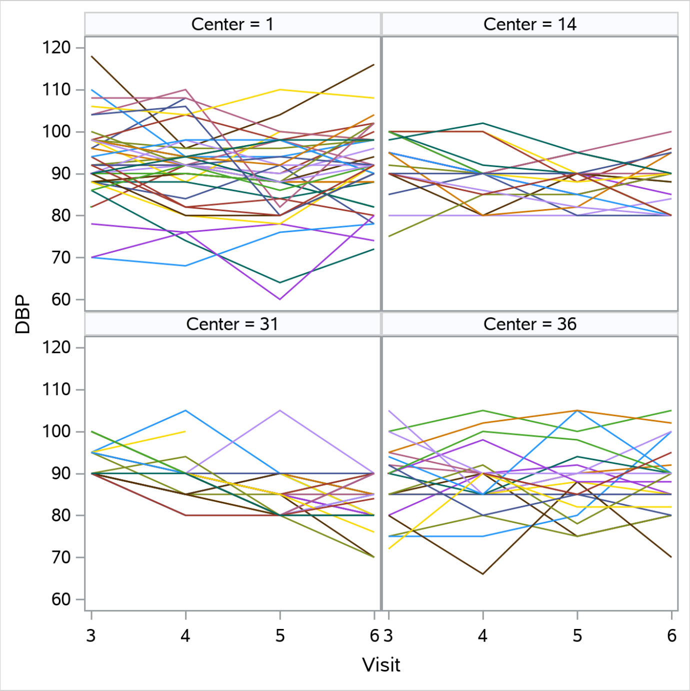


Patients at the same center tend to be more similar to each other than they are to patients from another center. The reason for within-center similarity could be the closeness of patients' residences or the shared medical practice at the center. Furthermore, repeated DBP measurements of the same patient are closer to each other than they are to measurements of a different patient.

The within-cluster dependence makes ordinary regression modeling inappropriate, but you can use multilevel models to accommodate such dependence. However, the cluster correlation is more than just a nuisance. The hierarchical design provides rich information about how the processes operate at different levels. Multilevel models enable you to disentangle such information by including covariates at different levels and assigning unexplained variability to different levels. For example, a three-level model enables you to estimate effects of covariates at the visit, patient, and center levels in the multicenter study. Moreover, you can include random effects to address the variability that is not explained by those covariates. These random effects are specified at levels that are defined by nested clusters.

Figure 6 shows a spaghetti plot of DBP against the visit time for the patients at four centers. The plot shows that the DBP trends vary significantly from patient to patient. If you picture the trend of DBP as a linear function of visit time, you can see considerable variability in the intercepts within each center.

Figure 6 Spaghetti Plot of Four Centers



Consider constructing a three-level model in the following stages:

1. The level-1 model for visit i of patient j at center k is a linear regression on visit time,

$$\text{DBP}_{ijk} = \alpha_0 + \alpha_1 \text{Visit}_{ijk} + e_{ijk}$$

2. Assume that the intercept α_0 varies among patients according to the level-2 model,

$$\alpha_0 = \delta_0 + \delta_1 \text{Baseline}_{jk} + \delta_2 \text{Treat}_{jk} + \gamma_{0,jk}^{\text{Patient}} \text{Patient}_{jk}$$

where $\gamma_{0,jk}^{\text{Patient}}$ is the patient-level random intercept.

3. Express the variability among the centers in the level-3 model,

$$\delta_0 = \lambda_0 + \gamma_{0,k}^{\text{Center}} \text{Center}_k$$

where $\gamma_{0,k}^{\text{Center}}$ is the center-level random intercept.

Substituting the level-3 model into the level-2 model and then substituting the level-2 model into the level-1 model yields

$$\begin{aligned} \text{DBP}_{ijk} &= \lambda_0 + \alpha_1 \text{Visit}_{ijk} + \delta_1 \text{Baseline}_{jk} + \delta_2 \text{Treat}_{jk} \\ &\quad + \gamma_{0,k}^{\text{Center}} \text{Center}_k \\ &\quad + \gamma_{0,jk}^{\text{Patient}} \text{Patient}_{jk} \\ &\quad + e_{ijk} \end{aligned}$$

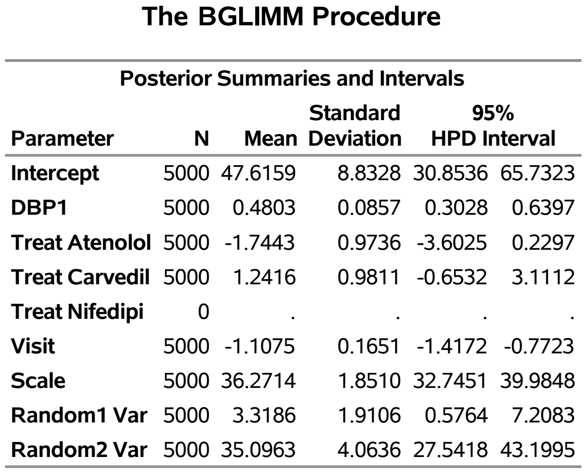
You can fit this three-level model by using the following PROC BGLIMM code:

```
proc bglimm data=DBP seed=98876 nmc=10000 thin=2 dic;
   class Patient Center Treat;
   model DBP = DBP1 Treat Visit ;
   random intercept / subject = Center;
   random intercept / subject = Patient(center);
   estimate 'Carvedil vs. Atenolol' Treat -1 1  0;
   estimate 'Carvedil vs. Nifedipi' Treat 0  1 -1;
run;
```

The two RANDOM statements specify two random intercepts with different clustering, and the second RANDOM statement has a nested subject. The two ESTIMATE statements compare the effect of the new drug, Carvedilol, with the effects of the two standard treatments, Nifedipine and Atenolol.

The “Posterior Summaries and Intervals” table in Figure 7 lists the summary statistics (posterior means, standard deviations, and HPD intervals) for each parameter, the fixed coefficients ($\boldsymbol{\beta}$), the scale parameter (σ^2), the variance at the center level (labeled “Random1 Var” because it is the first RANDOM statement), and the variance at the patient level (labeled “Random2 Var”). If you control for other covariates, the DBP decreases 1.11 mmHg (see the posterior mean for **Visit**) at each successive visit on average, and every increase of 1 mmHg in baseline DBP leads to an increase of 0.48 mmHg (see the posterior mean for **DBP1**) in posttreatment DBP. You can see that the “Treat Atenolol” effect (versus the effect of the reference group, “Treat Nifedipi,” which is fixed at 0) is –1.74 and the “Treat Carvedil” effect is 1.24.

Figure 7 Posterior Summaries and Intervals



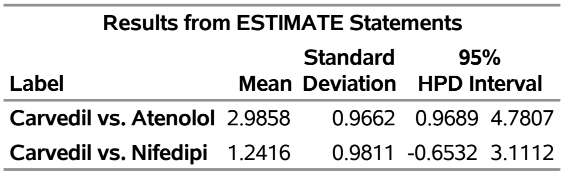
The BGLIMM Procedure

Posterior Summaries and Intervals

Parameter	N	Mean	Standard Deviation	95% HPD Interval	
Intercept	5000	47.6159	8.8328	30.8536	65.7323
DBP1	5000	0.4803	0.0857	0.3028	0.6397
Treat Atenolol	5000	-1.7443	0.9736	-3.6025	0.2297
Treat Carvedil	5000	1.2416	0.9811	-0.6532	3.1112
Treat Nifedipi	0	.	.	.	.
Visit	5000	-1.1075	0.1651	-1.4172	-0.7723
Scale	5000	36.2714	1.8510	32.7451	39.9848
Random1 Var	5000	3.3186	1.9106	0.5764	7.2083
Random2 Var	5000	35.0963	4.0636	27.5418	43.1995

Figure 8 shows the results of comparing the effects of the new drug (Carvedilol) with the effects of the two standard treatments (Nifedipine and Atenolol). You can see that the DBP of a patient who receives Carvedilol is 3 mmHg higher, on average, than the DBP of a patient who receives Atenolol and that the 95% HPD interval does not include 0. The DBP of a patient who receives Carvedilol is 1.2 mmHg higher, on average, than the DBP of a patient who receives Nifedipine, but the 95% HPD interval includes 0.

Figure 8 Estimated Differences in Treatments



Results from ESTIMATE Statements

Label	Mean	Standard Deviation	95% HPD Interval	
Carvedil vs. Atenolol	2.9858	0.9662	0.9689	4.7807
Carvedil vs. Nifedipi	1.2416	0.9811	-0.6532	3.1112

You can compute the conditional correlation between DBP measurements of two different patients at the same center and the conditional correlation between DBP measurements of the same patient at two visits. You can do this by using the posterior means of the scale parameter (σ^2), the variance at the center level, and the variance at the patient level, as follows:

$$\text{Corr}(\text{DBP}_{ijk}, \text{DBP}_{i'j'k}) = \frac{3}{3+35+36} = 0.04$$

$$\text{Corr}(\text{DBP}_{ijk}, \text{DBP}_{i'jk}) = \frac{3+35}{3+35+36} = 0.51$$

That is, of the variability in DBP that is not explained by the covariates, 4$\%$ is caused by unobserved center-specific attributes and 51$\%$ is caused by unobserved patient-specific attributes. Another way to interpret this is that DBP measurements for the same patient are much more similar to each other than are DBP measurements for different patients at the same center, as the spaghetti plot in Figure 6 indicates.

The analysis so far has revealed that visit time has a very strong negative effect on DBP. That is, the average patient's blood pressure decreases over the course of the study, regardless of treatment or center. Is this reduction rate the same for all centers? This is a question about the interaction between **Center** and **Visit**. The three-level model that was previously posited assumes that only the intercept varies among centers, but now you want to know whether the slope for time also varies among centers:

$$\begin{aligned}\text{DBP}_{ijk} &= \lambda_0 + \alpha_1 \text{Visit}_{ijk} + \delta_1 \text{Baseline}_{jk} + \delta_2 \text{Treat}_{jk} \\ &\quad + \gamma_{0,k}^{\text{Center}} \text{Center}_k + \gamma_{1,k}^{\text{Center}} \text{Visit}_{ijk} \text{Center}_k \\ &\quad + \gamma_{0,jk}^{\text{Patient}} \text{Patient}_{jk} \\ &\quad + e_{ijk}\end{aligned}$$

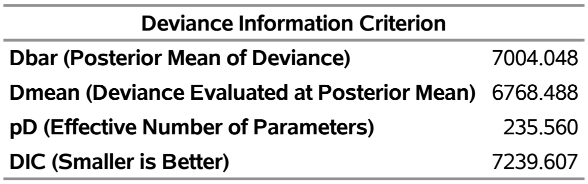
where $\gamma_{1,k}^{\text{Center}}$ is the center-level random slope for visit time.

You can fit this modified three-level model with both random intercept and slope at the center level by using the following code:

```
proc bglimm data=DBP seed=98876 nmc=10000 thin=2 dic;
   class Patient Center Treat;
   model DBP = DBP1 Treat Visit ;
   random intercept Visit / subject = Center type=un;
   random intercept / subject = Patient(center);
run;
```

The TYPE=UN option in the first RANDOM statement requests an unstructured covariance structure for center-level random effects. The DIC option in the PROC BGLIMM statement calculates the deviation information criterion, which results in a DIC value of 7239.6 (Figure 9). The random-intercept-only model, in contrast, has a larger DIC value of 7253.264 (Figure 10). The added parameters in the random-intercept-and-slope model provide a better fit for the model.

Figure 9 DIC Values from Random-Intercept-and-Slope Model



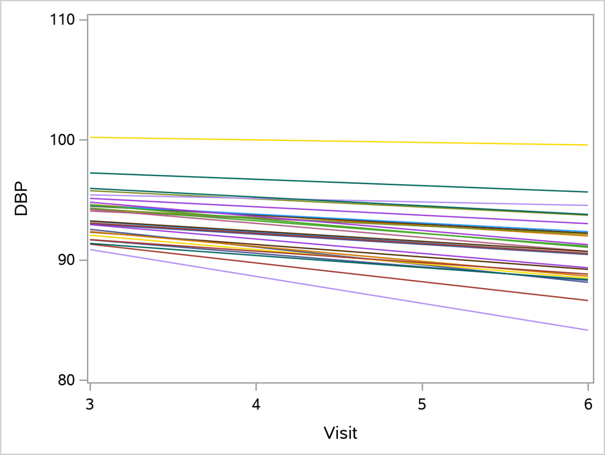
Deviance Information Criterion	
Dbar (Posterior Mean of Deviance)	7004.048
Dmean (Deviance Evaluated at Posterior Mean)	6768.488
pD (Effective Number of Parameters)	235.560
DIC (Smaller is Better)	7239.607

Figure 10 DIC Values from Random-Intercept-Only Model

Deviance Information Criterion	
Dbar (Posterior Mean of Deviance)	7024.194
Dmean (Deviance Evaluated at Posterior Mean)	6795.124
pD (Effective Number of Parameters)	229.070
DIC (Smaller is Better)	7253.264

Figure 11 plots the predicted DBP over time for each center. You can see that both intercept and slope vary significantly across centers.

Figure 11 Predicted DBP over Time across Centers



Example 3: Repeated Measurements with Heterogeneous Variance

Heterogeneity of variances occurs in many situations. A main motivation for modeling heterogeneous variances is to appropriately down-weight portions of the data that are highly variable and extract more information from portions of the data that are less variable.

As discussed in Littell et al. (2006), heterogeneous models fall into two categories: within-subject heterogeneity of the covariance parameters and between-subject heterogeneity of the covariance parameters. Within-subject heterogeneity occurs across data from the same individual. An example is the variances that change with time in a longitudinal or repeated-measures setting. Between-subject heterogeneity occurs when different groups of subjects display different variance patterns but are homogeneous within groups or when the variance components that correspond to random effects are unequal. Heterogeneous variances can be incorporated into the analysis if you specify various variance or covariance structures.

This example illustrates the two types of heterogeneity in the context of repeated-measures data. The data come from a two-treatment, randomized double-blind clinical trial for patients with rheumatoid arthritis; see Patel (1991). Sixty-seven patients enrolled in the trial. A baseline grip strength (in mmHg) was measured at the start of the trial. All patients were followed up three times, and a grip strength measurement was taken at each follow-up visit. The distribution of grip strength among males was expected to have a higher mean value than that among females.

The following DATA step reads the data set, **GripData**:

```
data GripData;
   input Subject Baseline Treat Gender $ Time T Grip;
   datalines;
26  175  1  M  1  1  161
26  175  1  M  2  2  210
26  175  1  M  3  3  230
27  165  1  M  1  1  215
27  165  1  M  2  2  245
27  165  1  M  3  3  265
```

```
   ... more lines ...

71  104  2  F  1  1  107
71  104  2  F  2  2  .
71  104  2  F  3  3  .
72   60  2  F  1  1   60
72   60  2  F  2  2   55
72   60  2  F  3  3   58
;
```

Some response data are missing; that is, some patients were measured on fewer than three occasions. By default, PROC BGLIMM treats missing response values as parameters and includes the sampling of the missing data as part of the simulation. The procedure discards all observations that have missing covariates. This is equivalent to assuming that the missing values are missing at random (MAR).

Initial Model

A reasonable initial model for most data should involve fairly general specifications for both the mean and the variance-covariance structure (as recommended by Littell et al. 2006). This initial model includes a **Baseline** covariate and three-way interactions of the class variables **Gender**, **Treat**, and **Time**. To allow general within-subject heterogeneity, the unstructured covariance is used in the R-side matrix. An advantage of considering this most general model is that you can inspect the estimates of the covariance matrix for heterogeneous patterns in both the variances and correlations.

You can fit the initial model by using the following code:

```
proc bglimm data=GripData seed=475193 dic;
   class Subject Treat Gender Time;
   model Grip = Gender*Treat*Time Baseline / noint;
   repeated Time / sub=Subject type=un r rcorr;
   run;
```

The MODEL statement specifies that the response variable is **Grip** and that the fixed effects contain 12 cell means involving **Gender**, **Treat**, and **Time** (2 treatments by 2 genders by 3 visits). The crossed effects (interactions) are specified by joining the three classification variables with asterisks as a simple way to obtain the 12 main cell means; you could also use the vertical bar operator (|) as shorthand for all main effects and interactions, which should produce an equivalent model with different interpretations for some parameters. In addition, the mean model includes a baseline covariate.

The REPEATED statement specifies that the repeated measurements be taken over the **Time** variable. The repeated effect is required in a REPEATED statement, and it must be specified as a CLASS variable. The repeated measurements are grouped according to **Subject** (the SUB= variable), and the covariance type is specified as unstructured via the TYPE=UN option. The R and RCORR options produce printouts of the estimated covariance matrix of $\mathbf{R}$ and its corresponding correlation form.

Figure 12 shows the estimated **R** covariance in the 3×3 matrix format and its correlation form. The variances appear to increase over time. And there is no obvious pattern in the correlation structures—an indication that the fully unstructured type might be necessary.

Figure 12 Estimated Covariance and Correlation Matrices of $\mathbf{R}$

Estimated R Matrix

Row	Col 1	Col 2	Col 3
1	604.96	308.00	288.96
2	308.00	950.48	885.65
3	288.96	885.65	1304.71

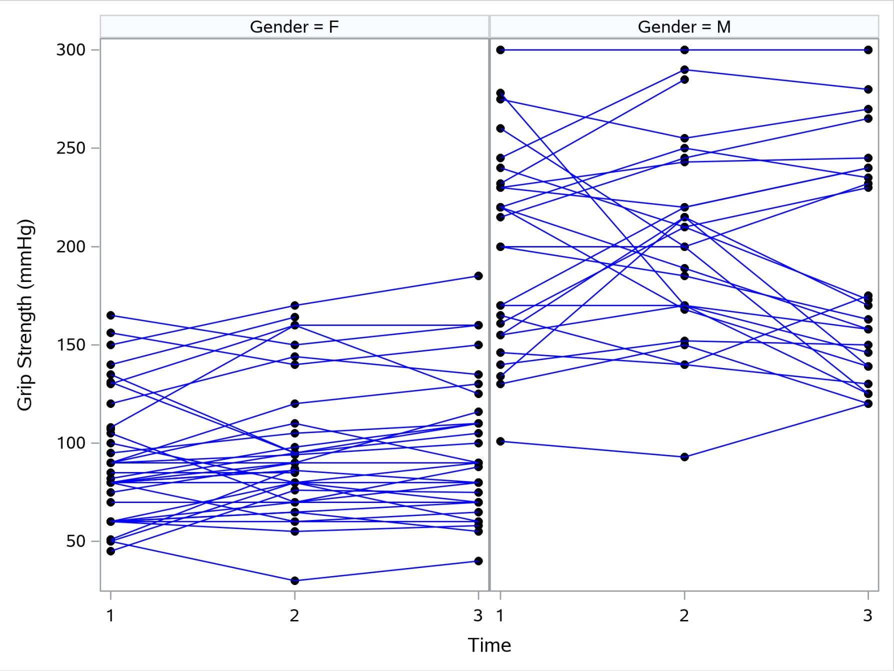
Figure 12 *continued*

Estimated R Correlation Matrix			
Row	**Col 1**	**Col 2**	**Col 3**
1	1.0000	0.4062	0.3252
2	0.4062	1.0000	0.7953
3	0.3252	0.7953	1.0000

Between-Subject Heterogeneity

There is, however, a considerate amount of between-subject heterogeneity in the data. To show this, Figure 13 plots side-by-side profiles of grip strength by time for female and male patients. Males tend to have a stronger grip and higher levels of variability across visits.

Figure 13 Grip Strength Plot by Gender



To account for distinct covariance structures of the two gender groups, you can fit the model by adding the option GROUP=GENDER to the REPEATED statement:

```
proc bglimm data=GripData seed=475193;
   class Subject Treat Gender Time;
   model Grip = Gender*Treat*Time Baseline / noint;
   repeated Time / subject=Subject type=un group=Gender r;
   run;
```

Figure 14 displays the estimated covariance matrices (the first three rows are for female patients and the last three rows are for male patients). They indicate three systematic between-gender differences: (1) male patients have higher variances across the board; (2) variances for female patients decrease over time, but the trend is reversed for males; (3) the correlation patterns are not the same if you compare the off-diagonal terms between the two gender groups.

Figure 14 Estimated Covariance of $\mathbf{R}$ for Both Genders

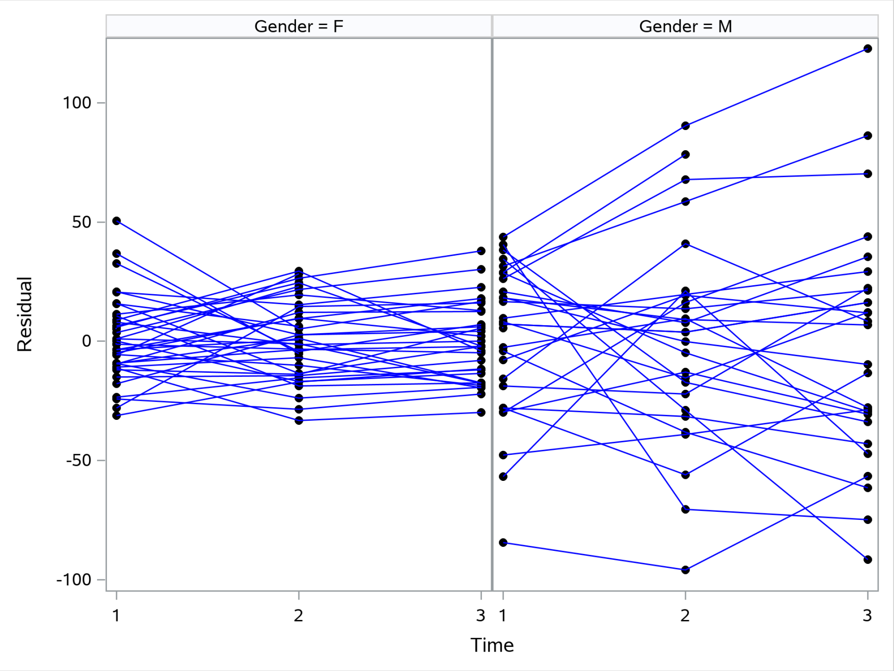
The BGLIMM Procedure

Estimated R Matrix

Group	Row	Col 1	Col 2	Col 3
Gender F	**1**	300.08	77.2769	95.2165
Gender F	**2**	77.2769	267.23	195.20
Gender F	**3**	95.2165	195.20	257.48
Gender M	**1**	960.37	591.43	528.93
Gender M	**2**	591.43	1773.63	1710.94
Gender M	**3**	528.93	1710.94	2504.11

You might notice that the increase and decrease of variances over time are not apparent in the data plot in Figure 13. This is because most of the overall variabilities in the data are explained by the baseline covariates, and the remaining variabilities are modeled by the **R** matrix. Figure 15 shows that the residuals and the variance trends are in closer agreement with the estimates.

Figure 15 Residuals Plot by Gender



Alternative Approach

As an alternative to the previous model, you can account for both between- and within-subject heterogeneity by using a random-effects model. Consider a random-intercept model with heterogeneous residual variance for the two genders, which you specify using the following statements:

```
proc bglimm data=GripData seed=475193 nmc=20000 thin=4;
   class Subject Treat Gender Time;
   model Grip = Gender*Treat*Time Baseline / noint;
   random int / sub=Subject group=Gender covprior=uniform;
   repeated Time / sub=Subject type=un group=Gender r rcorr covprior=iw(scale=500);
   run;
```

The RANDOM statement is added to account for the between-subject heterogeneity; the GROUP=GENDER option indicates that patients have different variances between genders but are homogeneous within each gender. The COVPRIOR=UNIFORM option in the RANDOM statement specifies a noninformative prior for the variance parameter to downplay the role of a relatively informative prior on the posterior distribution. The COVPRIOR=IW(SCALE=500) option in the REPEATED statement changes to use a much larger scale hyperparameter (from the default 4 to 500) for the same purpose.

The random-effects model shifts some of the variability from the **R** matrix to the G side, resulting in smaller residual variance estimates.

COMPARISON WITH PROC MIXED, PROC GLIMMIX, AND PROC MCMC

PROC MIXED and PROC GLIMMIX provide classical frequentist statistics solutions to the linear and generalized linear mixed-effects models, respectively. Frequentist estimation methods rely on maximizing the marginal likelihood function, and inferences are often based on asymptotic theorems. In linear model scenarios, PROC MIXED and PROC BGLIMM produce estimates that are nearly identical when noninformative prior distributions are used in PROC BGLIMM. PROC GLIMMIX can produce estimates that are quite different from those of PROC BGLIMM, in situations where linearization methods (such as pseudo-likelihood estimation) are used in PROC GLIMMIX.

PROC MCMC is a general-purpose, simulation-based Bayesian procedure that provides flexibility in model specification but requires more programming by the user. You can use PROC MCMC to fit GLMMs, although mixing can sometimes be less efficient because of the general sampling (and not model-specific) algorithms that PROC MCMC uses (Chen, Brown, and Stokes 2016). PROC MCMC also lacks some conveniences; for example, it does not support a CLASS statement to handle categorical variables automatically.

Certain features have yet to be implemented in PROC BGLIMM, and you need to use PROC MCMC instead. For example, if you want to work with more general prior distributions (on fixed effects, G-sided variance terms, and so on), or specify random effects with nonnormal prior distributions or nested prior distributions, or if you want to work with missing not at random data, and so on, then you need to use PROC MCMC.

SUMMARY

PROC BGLIMM is a Bayesian procedure that is designed specifically for fitting generalized linear mixed models by using Markov chain Monte Carlo methods. The procedure adopts familiar SAS syntax in specifying GLMMs, and a key enhancement over the existing MCMC procedure is its simplicity in specifying a large class of GLMMs. PROC BGLIMM uses efficient sampling algorithms that are parallelized for performance, resulting in good mixing and faster computation. PROC BGLIMM also provides functionality for handling missing data, nested multilevel models, and repeated-measures data. Additional features will be incorporated in future releases.

REFERENCES

Brown, H., and Prescott, R. (1999). *Applied Mixed Models in Medicine*. New York: John Wiley & Sons.

Chen, F., Brown, G., and Stokes, M. (2016). “Fitting Your Favorite Mixed Models with PROC MCMC.” In *Proceedings of the SAS Global Forum 2016 Conference*. Cary, NC: SAS Institute Inc. `https://support.sas.com/resources/papers/proceedings16/SAS5601-2016.pdf`.

Gamerman, D. (1997). “Sampling from the Posterior Distribution in Generalized Linear Models.” *Statistics and Computing* 7:57–68.

Hoffman, M. D., and Gelman, A. (2014). “The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo.” *Journal of Machine Learning Research* 15:1351–1381.

Littell, R. C., Milliken, G. A., Stroup, W. W., Wolfinger, R. D., and Schabenberger, O. (2006). *SAS for Mixed Models*. 2nd ed. Cary, NC: SAS Institute Inc.

Neal, R. M. (2011). “MCMC Using Hamiltonian Dynamics.” In *Handbook of Markov Chain Monte Carlo*, edited by S. Brooks, A. Gelman, G. L. Jones, and X.-L. Meng, 113–161. Boca Raton, FL: CRC Press.

RECOMMENDED READING

PROC BGLIMM requires SAS® 9.4M6. Complete documentation of the BGLIMM procedure can be found on the web at http://support.sas.com/documentation/onlinedoc/stat/151/bglimm.pdf.

You can find additional coding examples at http://support.sas.com/rnd/app/examples/STATexamples.html.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author:

Amy Shi
SAS Institute Inc.
SAS Campus Drive, Cary, NC 27513
amy.shi@sas.com