

SAS/STAT® 13.2 User's Guide
The GLIMMIX Procedure



This document is an individual chapter from *SAS/STAT® 13.2 User's Guide*.

The correct bibliographic citation for the complete manual is as follows: SAS Institute Inc. 2014. *SAS/STAT® 13.2 User's Guide*. Cary, NC: SAS Institute Inc.

Copyright © 2014, SAS Institute Inc., Cary, NC, USA

All rights reserved. Produced in the United States of America.

For a hard-copy book: No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

For a Web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

The scanning, uploading, and distribution of this book via the Internet or any other means without the permission of the publisher is illegal and punishable by law. Please purchase only authorized electronic editions and do not participate in or encourage electronic piracy of copyrighted materials. Your support of others' rights is appreciated.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government. Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007). If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation. The Government's rights in Software and documentation shall be only those set forth in this Agreement.

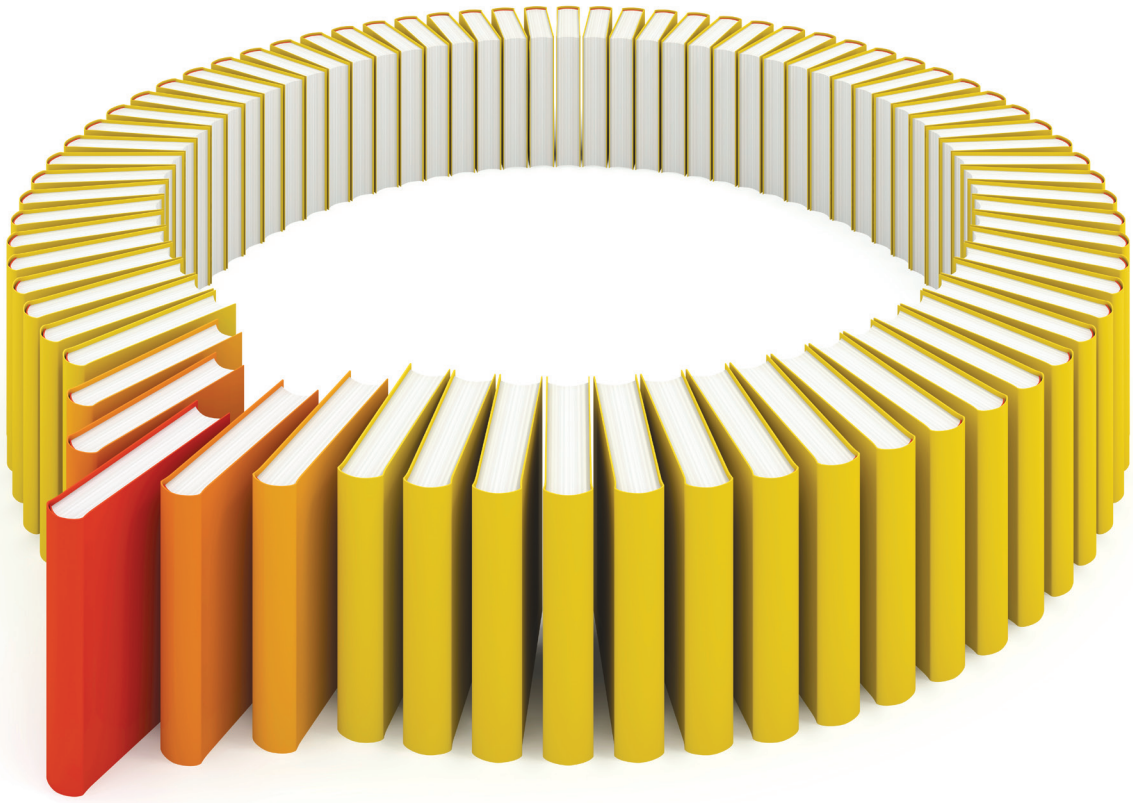
SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

August 2014

SAS provides a complete selection of books and electronic products to help customers use SAS® software to its fullest potential. For more information about our offerings, visit **support.sas.com/bookstore** or call 1-800-727-3228.

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.



Gain Greater Insight into Your SAS® Software with SAS Books.

Discover all that you need on your journey to knowledge and empowerment.

support.sas.com/bookstore
for additional books and resources.

sas
THE POWER TO KNOW®

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies. © 2013 SAS Institute Inc. All rights reserved. S107969US.0613

Chapter 44
The GLIMMIX Procedure

Contents

Overview: GLIMMIX Procedure **3052**
- Basic Features 3052
- Assumptions 3053
- Notation for the Generalized Linear Mixed Model 3054
  - The Basic Model 3054
  - G-Side and R-Side Random Effects and Covariance Structures 3055
  - Relationship with Generalized Linear Models 3056
- PROC GLIMMIX Contrasted with Other SAS Procedures 3056

Getting Started: GLIMMIX Procedure **3057**
- Logistic Regressions with Random Intercepts 3057

Syntax: GLIMMIX Procedure **3063**
- PROC GLIMMIX Statement 3064
- BY Statement 3090
- CLASS Statement 3091
- CODE Statement 3092
- CONTRAST Statement 3092
- COVTEST Statement 3096
- EFFECT Statement 3103
- ESTIMATE Statement 3104
- FREQ Statement 3110
- ID Statement 3110
- LSMEANS Statement 3111
- LSMESTIMATE Statement 3124
- MODEL Statement 3131
  - Response Variable Options 3133
  - Model Options 3134
- NLOPTIONS Statement 3145
- OUTPUT Statement 3146
- PARMS Statement 3150
- RANDOM Statement 3155
- SLICE Statement 3175
- STORE Statement 3175
- WEIGHT Statement 3175
- Programming Statements 3175
- User-Defined Link or Variance Function 3177
  - Implied Variance Functions 3177

Automatic Variables . . . 3178
Details: GLIMMIX Procedure . . . **3180**
Generalized Linear Models Theory . . . 3180
Maximum Likelihood . . . 3181
Scale and Dispersion Parameters . . . 3184
Quasi-likelihood for Independent Data . . . 3184
Effects of Adding Overdispersion . . . 3185
Generalized Linear Mixed Models Theory . . . 3186
Model or Integral Approximation . . . 3186
Pseudo-likelihood Estimation Based on Linearization . . . 3188
Maximum Likelihood Estimation Based on Laplace Approximation . . . 3192
Maximum Likelihood Estimation Based on Adaptive Quadrature . . . 3195
Aspects Common to Adaptive Quadrature and Laplace Approximation . . . 3197
Notes on Bias of Estimators . . . 3199
Pseudo-likelihood Estimation for Weighted Multilevel Models . . . 3200
GLM Mode or GLMM Mode . . . 3203
Statistical Inference for Covariance Parameters . . . 3204
The Likelihood Ratio Test . . . 3204
One- and Two-Sided Testing, Mixture Distributions . . . 3205
Handling the Degenerate Distribution . . . 3206
Wald Versus Likelihood Ratio Tests . . . 3207
Confidence Bounds Based on Likelihoods . . . 3207
Degrees of Freedom Methods . . . 3210
Between-Within Degrees of Freedom Approximation . . . 3210
Containment Degrees of Freedom Approximation . . . 3211
Satterthwaite Degrees of Freedom Approximation . . . 3211
Kenward-Roger Degrees of Freedom Approximation . . . 3212
Empirical Covariance (“Sandwich”) Estimators . . . 3213
Residual-Based Estimators . . . 3213
Design-Adjusted MBN Estimator . . . 3215
Exploring and Comparing Covariance Matrices . . . 3216
Processing by Subjects . . . 3218
Radial Smoothing Based on Mixed Models . . . 3220
From Penalized Splines to Mixed Models . . . 3220
Knot Selection . . . 3221
Odds and Odds Ratio Estimation . . . 3225
The Odds Ratio Estimates Table . . . 3226
Odds or Odds Ratio . . . 3228
Odds Ratios in Multinomial Models . . . 3229
Parameterization of Generalized Linear Mixed Models . . . 3229
Intercept . . . 3230
Interaction Effects . . . 3230
Nested Effects . . . 3230
Implications of the Non-Full-Rank Parameterization . . . 3230

Missing Level Combinations . . . 3231
Notes on the EFFECT Statement . . . 3231
Positional and Nonpositional Syntax for Contrast Coefficients . . . 3232
Response-Level Ordering and Referencing . . . 3235
Comparing the GLIMMIX and MIXED Procedures . . . 3236
Singly or Doubly Iterative Fitting . . . 3239
Default Estimation Techniques . . . 3241
Default Output . . . 3241
Model Information . . . 3242
Class Level Information . . . 3242
Number of Observations . . . 3242
Response Profile . . . 3242
Dimensions . . . 3242
Optimization Information . . . 3243
Iteration History . . . 3243
Convergence Status . . . 3244
Fit Statistics . . . 3244
Covariance Parameter Estimates . . . 3245
Type III Tests of Fixed Effects . . . 3245
Notes on Output Statistics . . . 3246
ODS Table Names . . . 3247
ODS Graphics . . . 3249
ODS Graph Names . . . 3250
Diagnostic Plots . . . 3251
Graphics for LS-Mean Comparisons . . . 3256
Examples: GLIMMIX Procedure . . . **3266**
Example 44.1: Binomial Counts in Randomized Blocks . . . 3266
Example 44.2: Mating Experiment with Crossed Random Effects . . . 3277
Example 44.3: Smoothing Disease Rates; Standardized Mortality Ratios . . . 3285
Example 44.4: Quasi-likelihood Estimation for Proportions with Unknown Distribution 3294
Example 44.5: Joint Modeling of Binary and Count Data . . . 3302
Example 44.6: Radial Smoothing of Repeated Measures Data . . . 3308
Example 44.7: Isotonic Contrasts for Ordered Alternatives . . . 3320
Example 44.8: Adjusted Covariance Matrices of Fixed Effects . . . 3321
Example 44.9: Testing Equality of Covariance and Correlation Matrices . . . 3326
Example 44.10: Multiple Trends Correspond to Multiple Extrema in Profile Likelihoods 3334
Example 44.11: Maximum Likelihood in Proportional Odds Model with Random Effects 3339
Example 44.12: Fitting a Marginal (GEE-Type) Model . . . 3345
Example 44.13: Response Surface Comparisons with Multiplicity Adjustments . . . 3349
Example 44.14: Generalized Poisson Mixed Model for Overdispersed Count Data . . . 3357
Example 44.15: Comparing Multiple B-Splines . . . 3365
Example 44.16: Diallel Experiment with Multimember Random Effects . . . 3371
Example 44.17: Linear Inference Based on Summary Data . . . 3373
Example 44.18: Weighted Multilevel Model for Survey Data . . . 3379
References . . . **3382**

Overview: GLIMMIX Procedure

The GLIMMIX procedure fits statistical models to data with correlations or nonconstant variability and where the response is not necessarily normally distributed. These models are known as generalized linear mixed models (GLMM).

GLMMs, like linear mixed models, assume normal (Gaussian) random effects. Conditional on these random effects, data can have any distribution in the exponential family. The exponential family comprises many of the elementary discrete and continuous distributions. The binary, binomial, Poisson, and negative binomial distributions, for example, are discrete members of this family. The normal, beta, gamma, and chi-square distributions are representatives of the continuous distributions in this family. In the absence of random effects, the GLIMMIX procedure fits generalized linear models (fit by the GENMOD procedure).

GLMMs are useful for the following applications:

- estimating trends in disease rates
- modeling CD4 counts in a clinical trial over time
- modeling the proportion of infected plants on experimental units in a design with randomly selected treatments or randomly selected blocks
- predicting the probability of high ozone levels in counties
- modeling skewed data over time
- analyzing customer preference
- joint modeling of multivariate outcomes

Such data often display correlations among some or all observations as well as nonnormality. The correlations can arise from repeated observation of the same sampling units, shared random effects in an experimental design, spatial (temporal) proximity, multivariate observations, and so on.

The GLIMMIX procedure does not fit hierarchical models with nonnormal random effects. With the GLIMMIX procedure you select the distribution of the response variable conditional on normally distributed random effects.

For more information about the differences between the GLIMMIX procedure and SAS procedures that specialize in certain subsets of the GLMM models, see the section "PROC GLIMMIX Contrasted with Other SAS Procedures" on page 3056.

Basic Features

The GLIMMIX procedure enables you to specify a generalized linear mixed model and to perform confirmatory inference in such models. The syntax is similar to that of the MIXED procedure and includes CLASS, MODEL, and RANDOM statements. For instructions on how to specify PROC MIXED REPEATED effects with PROC GLIMMIX, see the section "Comparing the GLIMMIX and MIXED Procedures" on page 3236. The following are some of the basic features of PROC GLIMMIX.

- SUBJECT= and GROUP= options, which enable blocking of variance matrices and parameter heterogeneity
- choice of linearization approach or integral approximation by quadrature or Laplace method for mixed models with nonlinear random effects or nonnormal distribution
- choice of linearization about expected values or expansion about current solutions of best linear unbiased predictors
- flexible covariance structures for random and residual random effects, including variance components, unstructured, autoregressive, and spatial structures
- CONTRAST, ESTIMATE, LSMEANS, and LSMESTIMATE statements, which produce hypothesis tests and estimable linear combinations of effects
- NLOPTIONS statement, which enables you to exercise control over the numerical optimization. You can choose techniques, update methods, line search algorithms, convergence criteria, and more. Or, you can choose the default optimization strategies selected for the particular class of model you are fitting.
- computed variables with SAS programming statements inside of PROC GLIMMIX (except for variables listed in the CLASS statement). These computed variables can appear in the MODEL, RANDOM, WEIGHT, or FREQ statement.
- grouped data analysis
- user-specified link and variance functions
- choice of model-based variance-covariance estimators for the fixed effects or empirical (sandwich) estimators to make analysis robust against misspecification of the covariance structure and to adjust for small-sample bias
- joint modeling for multivariate data. For example, you can model binary and normal responses from a subject jointly and use random effects to relate (fuse) the two outcomes.
- multinomial models for ordinal and nominal outcomes
- univariate and multivariate low-rank mixed model smoothing

Assumptions

The primary assumptions underlying the analyses performed by PROC GLIMMIX are as follows:

- If the model contains random effects, the distribution of the data conditional on the random effects is known. This distribution is either a member of the exponential family of distributions or one of the supplementary distributions provided by the GLIMMIX procedure. In models without random effects, the unconditional (marginal) distribution is assumed to be known for maximum likelihood estimation, or the first two moments are known in the case of quasi-likelihood estimation.

- The conditional expected value of the data takes the form of a linear mixed model after a monotonic transformation is applied.
- The problem of fitting the GLMM can be cast as a singly or doubly iterative optimization problem. The objective function for the optimization is a function of either the actual log likelihood, an approximation to the log likelihood, or the log likelihood of an approximated model.

For a model containing random effects, the GLIMMIX procedure, by default, estimates the parameters by applying pseudo-likelihood techniques as in Wolfinger and O'Connell (1993) and Breslow and Clayton (1993). In a model without random effects (GLM models), PROC GLIMMIX estimates the parameters by maximum likelihood, restricted maximum likelihood, or quasi-likelihood. See the section "Singly or Doubly Iterative Fitting" on page 3239 about when the GLIMMIX procedure applies noniterative, singly and doubly iterative algorithms, and the section "Default Estimation Techniques" on page 3241 about the default estimation methods. You can also fit generalized linear mixed models by maximum likelihood where the marginal distribution is numerically approximated by the Laplace method (METHOD=LAPLACE) or by adaptive Gaussian quadrature (METHOD=QUAD).

Once the parameters have been estimated, you can perform statistical inferences for the fixed effects and covariance parameters of the model. Tests of hypotheses for the fixed effects are based on Wald-type tests and the estimated variance-covariance matrix. The COVTEST statement enables you to perform inferences about covariance parameters based on likelihood ratio tests.

PROC GLIMMIX uses the Output Delivery System (ODS) for displaying and controlling the output from SAS procedures. ODS enables you to convert any of the output from PROC GLIMMIX into a SAS data set. See the section "ODS Table Names" on page 3247 for more information.

The GLIMMIX procedure uses ODS Graphics to create graphs as part of its output. For general information about ODS Graphics, see Chapter 21, "Statistical Graphics Using ODS." For specific information about the statistical graphics available with the GLIMMIX procedure, see the PLOTS options in the PROC GLIMMIX and LSMEANS statements.

Notation for the Generalized Linear Mixed Model

This section introduces the mathematical notation used throughout the chapter to describe the generalized linear mixed model (GLMM). See the section "Details: GLIMMIX Procedure" on page 3180 for a description of the fitting algorithms and the mathematical-statistical details.

The Basic Model

Suppose $\mathbf{Y}$ represents the $(n \times 1)$ vector of observed data and $\boldsymbol{\gamma}$ is a $(r \times 1)$ vector of random effects. Models fit by the GLIMMIX procedure assume that

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

where $g(\cdot)$ is a differentiable monotonic link function and $g^{-1}(\cdot)$ is its inverse. The matrix $\mathbf{X}$ is an $(n \times p)$ matrix of rank k, and $\mathbf{Z}$ is an $(n \times r)$ design matrix for the random effects. The random effects are assumed to be normally distributed with mean $\mathbf{0}$ and variance matrix $\mathbf{G}$.

The GLMM contains a linear mixed model inside the inverse link function. This model component is referred to as the linear predictor,

$$\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\boldsymbol{\gamma}$$

The variance of the observations, conditional on the random effects, is

$$\mathrm{Var}[\mathbf{Y}|\boldsymbol{\gamma}] = \mathbf{A}^{1/2}\mathbf{R}\mathbf{A}^{1/2}$$

The matrix $\mathbf{A}$ is a diagonal matrix and contains the variance functions of the model. The variance function expresses the variance of a response as a function of the mean. The GLIMMIX procedure determines the variance function from the DIST= option in the MODEL statement or from the user-supplied variance function (see the section "Implied Variance Functions" on page 3177). The matrix $\mathbf{R}$ is a variance matrix specified by the user through the RANDOM statement. If the conditional distribution of the data contains an additional scale parameter, it is either part of the variance functions or part of the $\mathbf{R}$ matrix. For example, the gamma distribution with mean μ has the variance function $a(\mu) = \mu^2$ and $\mathrm{Var}[Y|\boldsymbol{\gamma}] = \mu^2\phi$. If your model calls for G-side random effects only (see the next section), the procedure models $\mathbf{R} = \phi\mathbf{I}$, where $\mathbf{I}$ is the identity matrix. Table 44.19 identifies the distributions for which $\phi \equiv 1$.

G-Side and R-Side Random Effects and Covariance Structures

The GLIMMIX procedure distinguishes two types of random effects. Depending on whether the parameters of the covariance structure for random components in your model are contained in $\mathbf{G}$ or in $\mathbf{R}$, the procedure distinguishes between "G-side" and "R-side" random effects. The associated covariance structures of $\mathbf{G}$ and $\mathbf{R}$ are similarly termed the G-side and R-side covariance structure, respectively. R-side effects are also called "residual" effects. Simply put, if a random effect is an element of $\boldsymbol{\gamma}$, it is a G-side effect and you are modeling the G-side covariance structure; otherwise, you are modeling the R-side covariance structure of the model. Models without G-side effects are also known as marginal (or population-averaged) models. Models fit with the GLIMMIX procedure can have none, one, or more of each type of effect.

Note that an R-side effect in the GLIMMIX procedure is equivalent to a REPEATED effect in the MIXED procedure. The R-side covariance structure in the GLIMMIX procedure is the covariance structure that you would formulate with the REPEATED statement in the MIXED procedure. In the GLIMMIX procedure all random effects and their covariance structures are specified through the RANDOM statement. See the section "Comparing the GLIMMIX and MIXED Procedures" on page 3236 for a comparison of the GLIMMIX and MIXED procedures.

The columns of $\mathbf{X}$ are constructed from effects listed on the right side in the MODEL statement. Columns of $\mathbf{Z}$ and the variance matrices $\mathbf{G}$ and $\mathbf{R}$ are constructed from the RANDOM statement.

The $\mathbf{R}$ matrix is by default the scaled identity matrix, $\mathbf{R} = \phi\mathbf{I}$. The scale parameter ϕ is set to one if the distribution does not have a scale parameter, such as in the case of the binary, binomial, Poisson, and exponential distribution (see Table 44.19). To specify a different $\mathbf{R}$ matrix, use the RANDOM statement with the _RESIDUAL_ keyword or the RESIDUAL option. For example, to specify that the Time effect for each patient is an R-side effect with a first-order autoregressive covariance structure, use the RESIDUAL option:

```
random time / type=ar(1) subject=patient residual;
```

To add a multiplicative overdispersion parameter, use the _RESIDUAL_ keyword:

```
random _residual_;
```

You specify the link function $g(\cdot)$ with the LINK= option in the MODEL statement or with programming statements. You specify the variance function that controls the matrix $\mathbf{A}$ with the DIST= option in the MODEL statement or with programming statements.

Unknown quantities subject to estimation are the fixed-effects parameter vector $\boldsymbol{\beta}$ and the covariance parameter vector $\boldsymbol{\theta}$ that comprises all unknowns in $\mathbf{G}$ and $\mathbf{R}$. The random effects $\boldsymbol{\gamma}$ are not parameters of the model in the sense that they are not estimated. The vector $\boldsymbol{\gamma}$ is a vector of random variables. The solutions for $\boldsymbol{\gamma}$ are predictors of these random variables.

Relationship with Generalized Linear Models

Generalized linear models (Nelder and Wedderburn 1972; McCullagh and Nelder 1989) are a special case of GLMMs. If $\boldsymbol{\gamma} = \mathbf{0}$ and $\mathbf{R} = \phi\mathbf{I}$, the GLMM reduces to either a generalized linear model (GLM) or a GLM with overdispersion. For example, if $\mathbf{Y}$ is a vector of Poisson variables so that $\mathbf{A}$ is a diagonal matrix containing $\mathrm{E}[\mathbf{Y}] = \boldsymbol{\mu}$ on the diagonal, then the model is a Poisson regression model for $\phi = 1$ and overdispersed relative to a Poisson distribution for $\phi > 1$. Because the Poisson distribution does not have an extra scale parameter, you can model overdispersion by adding the following statement to your GLIMMIX program:

```
random _residual_;
```

If the only random effect is an overdispersion effect, PROC GLIMMIX fits the model by (restricted) maximum likelihood and not by one of the methods specific to GLMMs.

PROC GLIMMIX Contrasted with Other SAS Procedures

The GLIMMIX procedure generalizes the MIXED and GENMOD procedures in two important ways. First, the response can have a nonnormal distribution. The MIXED procedure assumes that the response is normally (Gaussian) distributed. Second, the GLIMMIX procedure incorporates random effects in the model and so allows for subject-specific (conditional) and population-averaged (marginal) inference. The GENMOD procedure allows only for marginal inference.

The GLIMMIX and MIXED procedure are closely related; see the syntax and feature comparison in the section "Comparing the GLIMMIX and MIXED Procedures" on page 3236. The remainder of this section compares the GLIMMIX procedure with the GENMOD, NLMIXED, LOGISTIC, and CATMOD procedures.

The GENMOD procedure fits generalized linear models for independent data by maximum likelihood. It can also handle correlated data through the marginal GEE approach of Liang and Zeger (1986) and Zeger and Liang (1986). The GEE implementation in the GENMOD procedure is a marginal method that does not incorporate random effects. The GEE estimation in the GENMOD procedure relies on R-side covariances only, and the unknown parameters in $\mathbf{R}$ are estimated by the method of moments. The GLIMMIX procedure allows G-side random effects and R-side covariances. PROC GLIMMIX can fit marginal (GEE-type) models, but the covariance parameters are not estimated by the method of moments. The parameters are estimated by likelihood-based techniques. When the GLIMMIX and GENMOD procedures fit a generalized linear model where the distribution contains a scale parameter, such as the normal, gamma, inverse Gaussian, or negative binomial distribution, the scale parameter is reported in the "Parameter Estimates" table. For

some distributions, the parameterization of this parameter differs. See the section "Scale and Dispersion Parameters" on page 3184 for details about how the GLIMMIX procedure parameterizes the log-likelihood functions and information about how the reported quantities differ between the two procedures.

Many of the fit statistics and tests in the GENMOD procedure are based on the likelihood. In a GLMM it is not always possible to derive the log likelihood of the data. Even if the log likelihood is tractable, it might be computationally infeasible. In some cases, the objective function must be constructed based on a substitute model. In other cases, only the first two moments of the marginal distribution can be approximated. Consequently, obtaining likelihood-based tests and statistics is difficult for many generalized linear mixed models. The GLIMMIX procedure relies heavily on linearization and Taylor-series techniques to construct Wald-type test statistics and confidence intervals. Likelihood ratio tests and confidence intervals for covariance parameters are available in the GLIMMIX procedure through the COVTEST statement.

The NLMIXED procedure fits nonlinear mixed models where the conditional mean function is a general nonlinear function. The class of generalized linear mixed models is a special case of the nonlinear mixed models; hence some of the models you can fit with PROC NLMIXED can also be fit with the GLIMMIX procedure. The NLMIXED procedure relies by default on approximating the marginal log likelihood through adaptive Gaussian quadrature. In the GLIMMIX procedure, maximum likelihood estimation by adaptive Gaussian quadrature is available with the METHOD=QUAD option in the PROC GLIMMIX statement. The default estimation methods thus differ between the NLMIXED and GLIMMIX procedures, because adaptive quadrature is possible for only a subset of the models available with the GLIMMIX procedure. If you choose METHOD=LAPLACE or METHOD=QUAD(QPOINTS=1) in the PROC GLIMMIX statement for a generalized linear mixed model, the GLIMMIX procedure performs maximum likelihood estimation based on a Laplace approximation of the marginal log likelihood. This is equivalent to the QPOINTS=1 option in the NLMIXED procedure.

The LOGISTIC and CATMOD procedures also fit generalized linear models; PROC LOGISTIC accommodates the independence case only. Binary, binomial, multinomial models for ordered data, and generalized logit models that can be fit with PROC LOGISTIC can also be fit with the GLIMMIX procedure. The diagnostic tools and capabilities specific to such data implemented in the LOGISTIC procedure go beyond the capabilities of the GLIMMIX procedure.

Getting Started: GLIMMIX Procedure

Logistic Regressions with Random Intercepts

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 for the procedures. In each center n_A patients were randomly selected and assigned to procedure "A," and n_B patients were randomly assigned to procedure "B." The following DATA step creates the data set 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 procedures, n is the number of patients who received a given procedure in a particular center, and sideeffect is the number of patients who reported side effects.

If Y_{iA} and Y_{iB} denote the number of patients in center i who report side effects for procedures A and B, respectively, then—for a given center—these are independent binomial random variables. To model the probability of side effects for the two drugs, π_{iA} and π_{iB}, 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:

$$\log\left\{\frac{\pi_{iA}}{1-\pi_{iA}}\right\} = \beta_0 + \beta_A + \gamma_i$$
$$\log\left\{\frac{\pi_{iB}}{1-\pi_{iB}}\right\} = \beta_0 + \beta_B + \gamma_i$$

In this model, $\beta_A - \beta_B$ measures the difference in the logits of experiencing side effects, and the γ_i are independent random variables due to the random selection of centers. If you think of β_0 as the overall intercept in the model, then the γ_i are random intercept adjustments. Observations from the same center receive the same adjustment, and these vary randomly from center to center with variance $\mathrm{Var}[\gamma_i] = \sigma_c^2$.

Because π_{iA} is the conditional mean of the sample proportion, $\mathrm{E}[Y_{iA}/n_{iA}|\gamma_i] = \pi_{iA}$, you can model the sample proportions as binomial ratios in a generalized linear mixed model. The following statements request

this analysis under the assumption of normally distributed center effects with equal variance and a logit link function:

```
proc glimmix data=multicenter;
   class center group;
   model sideeffect/n = group / solution;
   random intercept / subject=center;
run;
```

The PROC GLIMMIX statement invokes the procedure. The CLASS statement instructs the procedure to treat the variables center and group as classification variables. The MODEL statement specifies the response variable as a sample proportion by using the *events/trials* syntax. In terms of the previous formulas, sideeffect/n corresponds to Y_{iA}/n_{iA} for observations from group A and to Y_{iB}/n_{iB} for observations from group B. The SOLUTION option in the MODEL statement requests a listing of the solutions for the fixed-effects parameter estimates. Note that because of the *events/trials* syntax, the GLIMMIX procedure defaults to the binomial distribution, and that distribution's default link is the logit link. The RANDOM statement specifies that the linear predictor contains an intercept term that randomly varies at the level of the center effect. In other words, a random intercept is drawn separately and independently for each center in the study.

The results of this analysis are shown in Figure 44.1–Figure 44.9.

The "Model Information Table" in Figure 44.1 summarizes important information about the model you fit and about aspects of the estimation technique.

Figure 44.1 Model Information

The GLIMMIX Procedure

Model Information	
Data Set	WORK.MULTICENTER
Response Variable (Events)	sideeffect
Response Variable (Trials)	n
Response Distribution	Binomial
Link Function	Logit
Variance Function	Default
Variance Matrix Blocked By	center
Estimation Technique	Residual PL
Degrees of Freedom Method	Containment

PROC GLIMMIX recognizes the variables sideeffect and n as the numerator and denominator in the *events/trials* syntax, respectively. The distribution—conditional on the random center effects—is binomial. The marginal variance matrix is block-diagonal, and observations from the same center form the blocks. The default estimation technique in generalized linear mixed models is residual pseudo-likelihood with a subject-specific expansion (METHOD=RSPL).

The "Class Level Information" table lists the levels of the variables specified in the CLASS statement and the ordering of the levels. The "Number of Observations" table displays the number of observations read and used in the analysis (Figure 44.2).

Figure 44.2 Class Level Information and Number of Observations

Class Level Information		
Class	**Levels**	**Values**
center	15	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
group	2	A B

Number of Observations Read	30
Number of Observations Used	30
Number of Events	155
Number of Trials	503

There are two variables listed in the CLASS statement. The center variable has fifteen levels, and the group variable has two levels. Because the response is specified through the *events/trial* syntax, the "Number of Observations" table also contains the total number of events and trials used in the analysis.

The "Dimensions" table lists the size of relevant matrices (Figure 44.3).

Figure 44.3 Dimensions

Dimensions	
G-side Cov. Parameters	1
Columns in X	3
Columns in Z per Subject	1
Subjects (Blocks in V)	15
Max Obs per Subject	2

There are three columns in the $\mathbf{X}$ matrix, corresponding to an intercept and the two levels of the group variable. For each subject (center), the $\mathbf{Z}$ matrix contains only an intercept column.

The "Optimization Information" table provides information about the methods and size of the optimization problem (Figure 44.4).

Figure 44.4 Optimization Information

Optimization Information	
Optimization Technique	Dual Quasi-Newton
Parameters in Optimization	1
Lower Boundaries	1
Upper Boundaries	0
Fixed Effects	Profiled
Starting From	Data

The default optimization technique for generalized linear mixed models with binomial data is the quasi-Newton method. Because a residual likelihood technique is used to compute the objective function, only the covariance parameters participate in the optimization. A lower boundary constraint is placed on the variance component for the random center effect. The solution for this variance cannot be less than zero.

The "Iteration History" table displays information about the progress of the optimization process. After the initial optimization, the GLIMMIX procedure performed 15 updates before the convergence criterion was

met (Figure 44.5). At convergence, the largest absolute value of the gradient was near zero. This indicates that the process stopped at an extremum of the objective function.

Figure 44.5 Iteration History and Convergence Status

Iteration History

Iteration	Restarts	Subiterations	Objective Function	Change	Max Gradient
0	0	5	79.688580269	0.11807224	7.851E-7
1	0	3	81.294622554	0.02558021	8.209E-7
2	0	2	81.438701534	0.00166079	4.061E-8
3	0	1	81.444083567	0.00006263	2.311E-8
4	0	1	81.444265216	0.00000421	0.000025
5	0	1	81.444277364	0.00000383	0.000023
6	0	1	81.444266322	0.00000348	0.000021
7	0	1	81.44427636	0.00000316	0.000019
8	0	1	81.444267235	0.00000287	0.000017
9	0	1	81.444275529	0.00000261	0.000016
10	0	1	81.44426799	0.00000237	0.000014
11	0	1	81.444274843	0.00000216	0.000013
12	0	1	81.444268614	0.00000196	0.000012
13	0	1	81.444274277	0.00000178	0.000011
14	0	1	81.444269129	0.00000162	9.772E-6
15	0	0	81.444273808	0.00000000	9.102E-6

Convergence criterion (PCONV=1.11022E-8) satisfied.

The "Fit Statistics" table lists information about the fitted model (Figure 44.6).

Figure 44.6 Fit Statistics

Fit Statistics	
-2 Res Log Pseudo-Likelihood	81.44
Generalized Chi-Square	30.69
Gener. Chi-Square / DF	1.10

Twice the negative of the residual log likelihood in the final pseudo-model equaled 81.44. The ratio of the generalized chi-square statistic and its degrees of freedom is close to 1. This is a measure of the residual variability in the marginal distribution of the data.

The "Covariance Parameter Estimates" table displays estimates and asymptotic estimated standard errors for all covariance parameters (Figure 44.7).

Figure 44.7 Covariance Parameter Estimates

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error
Intercept	center	0.6176	0.3181

The variance of the random center intercepts on the logit scale is estimated as $\widehat{\sigma}_c^2 = 0.6176$.

The "Parameter Estimates" table displays the solutions for the fixed effects in the model (Figure 44.8).

Figure 44.8 Parameter Estimates

Solutions for Fixed Effects

Effect	group	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept		-0.8071	0.2514	14	-3.21	0.0063
group	A	-0.4896	0.2034	14	-2.41	0.0305
group	B	0	.	.	.	.

Because of the fixed-effects parameterization used in the GLIMMIX procedure, the "Intercept" effect is an estimate of $\beta_0 + \beta_B$, and the "A" group effect is an estimate of $\beta_A - \beta_B$, the log odds ratio. The associated estimated probabilities of side effects in the two groups are

$$\widehat{\pi}_A = \frac{1}{1 + \exp\{0.8071 + 0.4896\}} = 0.2147$$
$$\widehat{\pi}_B = \frac{1}{1 + \exp\{0.8071\}} = 0.3085$$

There is a significant difference between the two groups ($p = 0.0305$).

The "Type III Tests of Fixed Effect" table displays significance tests for the fixed effects in the model (Figure 44.9).

Figure 44.9 Type III Tests of Fixed Effects

Type III Tests of Fixed Effects

Effect	Num DF	Den DF	F Value	Pr > F
group	1	14	5.79	0.0305

Because the group effect has only two levels, the p-value for the effect is the same as in the "Parameter Estimates" table, and the "F Value" is the square of the "t Value" shown there.

You can produce the estimates of the average logits in the two groups and their predictions on the scale of the data with the LSMEANS statement in PROC GLIMMIX:

```
ods select lsmeans;
proc glimmix data=multicenter;
   class center group;
   model sideeffect/n = group / solution;
   random intercept / subject=center;
   lsmeans group / cl ilink;
run;
```

The LSMEANS statement requests the least squares means of the group effect on the logit scale. The CL option requests their confidence limits. The ILINK option adds estimates, standard errors, and confidence limits on the mean (probability) scale (Figure 44.10).

Figure 44.10 Least Squares Means

The GLIMMIX Procedure

group Least Squares Means

group	Estimate	Standard Error	DF	t Value	Pr > |t|	Alpha	Lower	Upper	Mean	Standard Error Mean	Lower Mean	Upper Mean
A	-1.2966	0.2601	14	-4.99	0.0002	0.05	-1.8544	-0.7388	0.2147	0.04385	0.1354	0.3233
B	-0.8071	0.2514	14	-3.21	0.0063	0.05	-1.3462	-0.2679	0.3085	0.05363	0.2065	0.4334

The "Estimate" column displays the least squares mean estimate on the logit scale, and the "Mean" column represents its mapping onto the probability scale. The "Lower" and "Upper" columns are 95% confidence limits for the logits in the two groups. The "Lower Mean" and "Upper Mean" columns are the corresponding confidence limits for the probabilities of side effects. These limits are obtained by inversely linking the confidence bounds on the linear scale, and thus are not symmetric about the estimate of the probabilities.

Syntax: GLIMMIX Procedure

The following statements are available in the GLIMMIX procedure:

PROC GLIMMIX < *options* > **;**
BY *variables* **;**
CLASS *variable* < **(REF=** *option***)** > . . . < *variable* < **(REF=** *option***)** > > < **/** *global-options* > **;**
CODE < *options* > **;**
CONTRAST '*label*' *contrast-specification* < **,** *contrast-specification* > < **,** . . . > < **/** *options* > **;**
COVTEST < '*label*' > < *test-specification* > < **/** *options* > **;**
EFFECT *effect-specification* **;**
ESTIMATE '*label*' *contrast-specification* < **(***divisor***=***n***)** >
< **,** '*label*' *contrast-specification* < **(***divisor***=***n***)** > > < **,** . . . > < **/** *options* > **;**
FREQ *variable* **;**
ID *variables* **;**
LSMEANS *fixed-effects* < **/** *options* > **;**
LSMESTIMATE *fixed-effect* < '*label*' > *values* < *divisor***=***n* >
< **,** < '*label*' > *values* < *divisor***=***n* > > < **,** . . . > < **/** *options* > **;**
MODEL *response*< **(***response-options***)** > **=** < *fixed-effects* > < **/** *model-options* > **;**
MODEL *events/trials* **=** < *fixed-effects* > < **/** *model-options* > **;**
NLOPTIONS < *options* > **;**
OUTPUT < **OUT=***SAS-data-set* >
< *keyword*< **(***keyword-options***)** > < **=***name* > > . . .
< *keyword*< **(***keyword-options***)** > < **=***name* > > < **/** *options* > **;**
PARMS **(***value-list***)** . . . < **/** *options* > **;**
RANDOM *random-effects* < **/** *options* > **;**
SLICE *model-effect* < **/** *options* > **;**
STORE < **OUT=** > *item-store-name* < **/** **LABEL=**'*label*' > **;**
WEIGHT *variable* **;**
Programming statements **;**

The CLASS, CONTRAST, COVTEST, EFFECT, ESTIMATE, LSMEANS, LSMESTIMATE, RANDOM and SLICE statements and the programming statements can appear multiple times. The PROC GLIMMIX and MODEL statements are required, and the MODEL statement must appear after the CLASS statement if a CLASS statement is included. The EFFECT statements must appear before the MODEL statement.

The SLICE statement is also available in many other procedures. A summary description of functionality and syntax for this statement is given in this chapter. You can find full documentation in the section "SLICE Statement" on page 505 in Chapter 19, "Shared Concepts and Topics."

PROC GLIMMIX Statement

PROC GLIMMIX < *options* > **;**

The PROC GLIMMIX statement invokes the GLIMMIX procedure. Table 44.1 summarizes the *options* available in the PROC GLIMMIX statement. These and other *options* in the PROC GLIMMIX statement are then described fully in alphabetical order.

Table 44.1 PROC GLIMMIX Statement Options

Option	Description
Basic Options	
DATA=	Specifies the input data set
METHOD=	Determines estimation method
NOFIT	Does not fit the model
NOPROFILE	Includes scale parameter in optimization
NOREML	Determines computation of scale parameters in GLM models
ORDER=	Determines the sort order of CLASS variables
OUTDESIGN	Writes **X** and/or **Z** matrices to a SAS data set
PROFILE	Profile scale parameters from the optimization
Displayed Output	
ASYCORR	Displays the asymptotic correlation matrix of the covariance parameter estimates
ASYCOV	Displays the asymptotic covariance matrix of the covariance parameter estimates
GRADIENT	Displays the gradient of the objective function with respect to the parameter estimates
HESSIAN	Displays the Hessian matrix
ITDETAILS	Adds estimates and gradients to the "Iteration History"
NAMELEN=	Specifies the length of long effect names
NOBSDETAIL	Shows data exclusions
NOCLPRINT	Suppresses "Class Level Information" completely or in part
ODDSRATIO	Requests odds ratios
PLOTS	Produces ODS statistical graphics
SUBGRADIENT	Writes subject-specific gradients to a SAS data set

Table 44.1 *continued*

Option	Description
Optimization Options	
MAXOPT=	Specifies the number of optimizations
Computational Options	
CHOLESKY	Constructs and solves mixed model equations using the Cholesky root of the **G** matrix
EMPIRICAL	Computes empirical ("sandwich") estimators
EXPHESSIAN	Uses the expected Hessian matrix to compute the covariance matrix of nonprofiled parameters
INFOCRIT	Affects the computation of information criteria
INITGLM	Uses fixed-effects starting values via generalized linear model
INITITER=	Sets the number of initial GLM steps
NOBOUND	Unbounds the covariance parameter estimates
NOINITGLM	Does not use fixed-effects starting values via generalized linear model
SCORING=	Applies Fisher scoring where applicable
SCOREMOD	Bases the Hessian matrix in GLMMs on a modified scoring algorithm
Singularity Tolerances	
ABSPCONV=	Determines the absolute parameter estimate convergence criterion for PL
FDIGITS=	Specifies significant digits in computing objective function
PCONV=	Specifies the relative parameter estimate convergence criterion for PL
SINGCHOL=	Tunes singularity for Cholesky decompositions
SINGRES=	Tunes singularity for the residual variance
SINGULAR=	Tunes general singularity criterion
Debugging Output	
LIST	Lists model program and variables

You can specify the following *options* in the PROC GLIMMIX statement.

ABSPCONV=*r*

specifies an absolute parameter estimate convergence criterion for doubly iterative estimation methods. For such methods, the GLIMMIX procedure by default examines the *relative* change in parameter estimates between optimizations (see PCONV=). The purpose of the ABSPCONV= criterion is to stop the process when the *absolute* change in parameter estimates is less than the tolerance criterion *r*. The criterion is based on fixed effects and covariance parameters.

Note that this convergence criterion does not affect the convergence criteria applied within any individual optimization. In order to change the convergence behavior within an optimization, you can change the ABSCONV=, ABSFCONV=, ABSGCONV=, ABSXCONV=, FCONV=, or GCONV= option in the NLOPTIONS statement.

ASYCORR

produces the asymptotic correlation matrix of the covariance parameter estimates. It is computed from the corresponding asymptotic covariance matrix (see the description of the ASYCOV option, which follows).

ASYCOV

requests that the asymptotic covariance matrix of the covariance parameter estimates be displayed. By default, this matrix is the observed inverse Fisher information matrix, which equals $m\mathbf{H}^{-1}$, where $\mathbf{H}$ is the Hessian (second derivative) matrix of the objective function. The factor m equals 1 in a GLM and equals 2 in a GLMM.

When you use the SCORING= option and PROC GLIMMIX converges without stopping the scoring algorithm, the procedure uses the expected Hessian matrix to compute the covariance matrix instead of the observed Hessian. Regardless of whether a scoring algorithm is used or the number of scoring iterations has been exceeded, you can request that the asymptotic covariance matrix be based on the expected Hessian with the EXPHESSIAN option in the PROC GLIMMIX statement. If a residual scale parameter is profiled from the likelihood equation, the asymptotic covariance matrix is adjusted for the presence of this parameter; details of this adjustment process are found in Wolfinger, Tobias, and Sall (1994) and in the section "Estimated Precision of Estimates" on page 3189.

CHOLESKY

CHOL

requests that the mixed model equations be constructed and solved by using the Cholesky root of the $\mathbf{G}$ matrix. This option applies only to estimation methods that involve mixed model equations. The Cholesky root algorithm has greater numerical stability but also requires more computing resources. When the estimated $\mathbf{G}$ matrix is not positive definite during a particular function evaluation, PROC GLIMMIX switches to the Cholesky algorithm for that evaluation and returns to the regular algorithm if $\widehat{\mathbf{G}}$ becomes positive definite again. When the CHOLESKY option is in effect, the procedure applies the algorithm all the time.

DATA=*SAS-data-set*

names the SAS data set to be used by PROC GLIMMIX. The default is the most recently created data set.

EMPIRICAL< =CLASSICAL | HC0 >

EMPIRICAL< =DF | HC1 >

EMPIRICAL< =MBN< (*mbn-options***) > >**

EMPIRICAL< =ROOT | HC2 >

EMPIRICAL< =FIRORES | HC3 >

EMPIRICAL< =FIROEEQ< (*r***) > >**

requests that the covariance matrix of the parameter estimates be computed as one of the asymptotically consistent estimators, known as *sandwich* or *empirical* estimators. The name stems from the layering of the estimator. An empirically based estimate of the inverse variance of the parameter estimates (the "meat") is wrapped by the model-based variance estimate (the "bread").

Empirical estimators are useful for obtaining inferences that are not sensitive to the choice of the covariance model. In nonmixed models, they can help, for example, to allay the effects of variance heterogeneity on the tests of fixed effects. Empirical estimators can coarsely be grouped into likelihood-based and residual-based estimators. The distinction arises from the components used to construct the

"meat" and "bread" of the estimator. If you specify the EMPIRICAL option without further qualifiers, the GLIMMIX procedure computes the classical sandwich estimator in the appropriate category.

Likelihood-Based Estimator

Let $\mathbf{H}(\boldsymbol{\alpha})$ denote the second derivative matrix of the log likelihood for some parameter vector $\boldsymbol{\alpha}$, and let $\mathbf{g}_i(\boldsymbol{\alpha})$ denote the gradient of the log likelihood with respect to $\boldsymbol{\alpha}$ for the ith of m independent sampling units. The gradient for the entire data is $\sum_{i=1}^{m} \mathbf{g}_i(\boldsymbol{\alpha})$. A sandwich estimator for the covariance matrix of $\widehat{\boldsymbol{\alpha}}$ can then be constructed as (White 1982)

$$\mathbf{H}(\widehat{\boldsymbol{\alpha}})^{-1} \left(\sum_{i=1}^{m} \mathbf{g}_i(\widehat{\boldsymbol{\alpha}}) \mathbf{g}_i(\widehat{\boldsymbol{\alpha}})' \right) \mathbf{H}(\widehat{\boldsymbol{\alpha}})^{-1}$$

If you fit a mixed model by maximum likelihood with Laplace or quadrature approximation (METHOD=LAPLACE, METHOD=QUAD), the GLIMMIX procedure constructs this likelihood-based estimator when you choose EMPIRICAL=CLASSICAL. If you choose EMPIRICAL=MBN, the likelihood-based sandwich estimator is further adjusted (see the section "Design-Adjusted MBN Estimator" on page 3215 for details). Because Laplace and quadrature estimation in GLIMMIX includes the fixed-effects parameters and the covariance parameters in the optimization, this empirical estimator adjusts the covariance matrix of both types of parameters. The following empirical estimators are not available with METHOD=LAPLACE or with METHOD=QUAD: EMPIRICAL=DF, EMPIRICAL=ROOT, EMPIRICAL=FIRORES, and EMPIRICAL=FIROEEQ.

Residual-Based Estimators

For a general model, let $\mathbf{Y}$ denote the response with mean $\boldsymbol{\mu}$ and variance $\boldsymbol{\Sigma}$, and let $\mathbf{D}$ be the matrix of first derivatives of $\boldsymbol{\mu}$ with respect to the fixed effects $\boldsymbol{\beta}$. The classical sandwich estimator (Huber 1967; White 1980) is

$$\widehat{\boldsymbol{\Omega}} \left(\sum_{i=1}^{m} \widehat{\mathbf{D}}_i' \widehat{\boldsymbol{\Sigma}}_i^{-1} \mathbf{e}_i \mathbf{e}_i' \widehat{\boldsymbol{\Sigma}}_i^{-1} \widehat{\mathbf{D}}_i \right) \widehat{\boldsymbol{\Omega}}$$

where $\boldsymbol{\Omega} = (\mathbf{D}'\boldsymbol{\Sigma}^{-1}\mathbf{D})^{-}$, $\mathbf{e}_i = \mathbf{y}_i - \widehat{\boldsymbol{\mu}}_i$, and m denotes the number of independent sampling units.

Since the expected value of $\mathbf{e}_i \mathbf{e}_i'$ does not equal $\boldsymbol{\Sigma}_i$, the classical sandwich estimator is biased, particularly if m is small. The estimator tends to underestimate the variance of $\widehat{\boldsymbol{\beta}}$. The EMPIRICAL=DF, ROOT, FIRORES, FIROEEQ, and MBN estimators are bias-corrected sandwich estimators. The DF estimator applies a simple sample size adjustment. The ROOT, FIRORES, and FIROEEQ estimators are based on Taylor series approximations applied to residuals and estimating equations. For uncorrelated data, the EMPIRICAL=FIRORES estimator can be motivated as a jackknife estimator.

In the case of a linear regression model, the various estimators reduce to the *heteroscedasticity-consistent covariance matrix* estimators (HCMM) of White (1980) and MacKinnon and White (1985). The classical estimator, HC0, was found to perform poorly in small samples. Based on simulations in regression models, MacKinnon and White (1985) and Long and Ervin (2000) strongly recommend the HC3 estimator. The sandwich estimators computed by the GLIMMIX procedure can be viewed as an

extension of the HC0—HC3 estimators of MacKinnon and White (1985) to accommodate nonnormal data and correlated observations.

The MBN estimator, introduced as a residual-based estimator (Morel 1989; Morel, Bokossa, and Neerchal 2003), applies an additive adjustment to the residual crossproduct. It is controlled by three suboptions. The valid *mbn-options* are as follows: a sample size adjustment is applied when the DF suboption is in effect. The NODF suboption suppresses this component of the adjustment. The lower bound of the design effect parameter $0 \leq r \leq 1$ can be specified with the R= option. The magnitude of Morel's δ parameter is partly determined with the D= option ($d \geq 1$).

For details about the general expression for the residual-based estimators and their relationship, see the section "Empirical Covariance ("Sandwich") Estimators" on page 3213. The MBN estimator and its parameters are explained for residual- and likelihood-based estimators in the section "Design-Adjusted MBN Estimator" on page 3215.

The EMPIRICAL=DF estimator applies a simple, multiplicative correction factor to the classical estimator (Hinkley 1977). This correction factor is

$$c = \begin{cases} m/(m-k) & m > k \\ 1 & \text{otherwise} \end{cases}$$

where k is the rank of $\mathbf{X}$, and m equals the sum of all frequencies when PROC GLIMMIX is in GLM mode and equals the number of subjects in GLMM mode. For example, the following statements fit an overdispersed GLM:

```
proc glimmix empirical;
   model y = x;
   random _residual_;
run;
```

PROC GLIMMIX is in GLM mode, and the individual observations are the independent sampling units from which the sandwich estimator is constructed. If you use a SUBJECT= effect in the RANDOM statement, however, the procedure fits the model in GLMM mode and the subjects represent the sampling units in the construction of the sandwich estimator. In other words, the following statements fit a GEE-type model with independence working covariance structure and subjects (clusters) defined by the levels of ID:

```
proc glimmix empirical;
   class id;
   model y = x;
   random _residual_ / subject=id type=vc;
run;
```

See the section "GLM Mode or GLMM Mode" on page 3203 for information about how the GLIMMIX procedure determines the estimation mode.

The EMPIRICAL=ROOT estimator is based on the residual approximation in Kauermann and Carroll (2001), and the EMPIRICAL=FIRORES estimator is based on the approximation in Mancl and DeRouen (2001). The Kauermann and Carroll estimator requires the inverse square root of a nonsymmetric matrix. This square root matrix is obtained from the singular value decomposition in PROC GLIMMIX, and thus this sandwich estimator is computationally more demanding than others. In the linear regression case, the Mancl-DeRouen estimator can be motivated as a jackknife estimator, based on the "leave-one-out" estimates of $\widehat{\boldsymbol{\beta}}$; see MacKinnon and White (1985) for details.

The EMPIRICAL=FIROEEQ estimator is based on approximating an unbiased estimating equation (Fay and Graubard 2001). It is computationally less demanding than the estimator of Kauermann and Carroll (2001) and, in certain balanced cases, gives identical results. The optional number $0 \leq r < 1$ is chosen to provide an upper bound on the correction factor. The default value for r is 0.75.

When you specify the EMPIRICAL option with a residual-based estimator, PROC GLIMMIX adjusts all standard errors and test statistics involving the fixed-effects parameters.

Sampling Units

Computation of an empirical variance estimator requires that the data can be processed by independent sampling units. This is always the case in GLMs. In this case, m, the number of independent units, equals the sum of the frequencies used in the analysis (see "Number of Observations" table). In GLMMs, empirical estimators can be computed only if the data comprise more than one subject as per the "Dimensions" table. See the section "Processing by Subjects" on page 3218 for information about how the GLIMMIX procedure determines whether the data can be processed by subjects. If a GLMM comprises only a single subject for a particular BY group, the model-based variance estimator is used instead of the empirical estimator, and a message is written to the log.

EXPHESSIAN
requests that the expected Hessian matrix be used in computing the covariance matrix of the nonprofiled parameters. By default, the GLIMMIX procedure uses the observed Hessian matrix in computing the asymptotic covariance matrix of covariance parameters in mixed models and the covariance matrix of fixed effects in models without random effects. The EXPHESSIAN option is ignored if the (conditional) distribution is not a member of the exponential family or is unknown. It is also ignored in models for nominal data.

FDIGITS=*r*
specifies the number of accurate digits in evaluations of the objective function. Fractional values are allowed. The default value is $r = -\log_{10} \epsilon$, where ϵ is the machine precision. The value of r is used to compute the interval size for the computation of finite-difference approximations of the derivatives of the objective function. It is also used in computing the default value of the FCONV= option in the NLOPTIONS statement.

GRADIENT
displays the gradient of the objective function with respect to the parameter estimates in the "Covariance Parameter Estimates" table and/or the "Parameter Estimates" table.

HESSIAN
HESS
H
displays the Hessian matrix of the optimization.

INFOCRIT=NONE | PQ | Q
IC=NONE | PQ | Q
determines the computation of information criteria in the "Fit Statistics" table. The GLIMMIX procedure computes various information criteria that typically apply a penalty to the (possibly restricted) log likelihood, log pseudo-likelihood, or log quasi-likelihood that depends on the number of parameters

and/or the sample size. If IC=NONE, these criteria are suppressed in the "Fit Statistics" table. This is the default for models based on pseudo-likelihoods.

The AIC, AICC, BIC, CAIC, and HQIC fit statistics are various information criteria. AIC and AICC represent Akaike's information criteria (Akaike 1974) and a small sample bias corrected version thereof (for AICC, see Hurvich and Tsai 1989; Burnham and Anderson 1998). BIC represents Schwarz's Bayesian criterion (Schwarz 1978). Table 44.2 gives formulas for the criteria.

Table 44.2 Information Criteria

Criterion	Formula	Reference
AIC	$-2\ell + 2d$	Akaike (1974)
AICC	$-2\ell + 2dn^*/(n^* - d - 1)$	Hurvich and Tsai (1989) Burnham and Anderson (1998)
HQIC	$-2\ell + 2d \log\log n$	Hannan and Quinn (1979)
BIC	$-2\ell + d \log n$	Schwarz (1978)
CAIC	$-2\ell + d(\log n + 1)$	Bozdogan (1987)

Here, ℓ denotes the maximum value of the (possibly restricted) log likelihood, log pseudo-likelihood, or log quasi-likelihood, d is the dimension of the model, and n, n^* reflect the size of the data.

The IC=PQ option requests that the penalties include the number of fixed-effects parameters, when estimation in models with random effects is based on a residual (restricted) likelihood. For METHOD=MSPL, METHOD=MMPL, METHOD=LAPLACE, and METHOD=QUAD, IC=Q and IC=PQ produce the same results. IC=Q is the default for linear mixed models with normal errors, and the resulting information criteria are identical to the IC option in the MIXED procedure.

The quantities d, n, and n^* depend on the model and IC= option in the following way:

GLM: IC=Q and IC=PQ options have no effect on the computation.

- d equals the number of parameters in the optimization whose solutions do not fall on the boundary or are otherwise constrained. The scale parameter is included, if it is part of the optimization. If you use the PARMS statement to place a hold on a scale parameter, that parameter does not count toward d.
- n equals the sum of the frequencies (f) for maximum likelihood and quasi-likelihood estimation and $f - \mathrm{rank}(\mathbf{X})$ for restricted maximum likelihood estimation.
- n^* equals n, unless $n < d + 2$, in which case $n^* = d + 2$.

GLMM, IC=Q:

- d equals the number of effective covariance parameters—that is, covariance parameters whose solution does not fall on the boundary. For estimation of an unrestricted objective function (METHOD=MMPL, METHOD=MSPL, METHOD=LAPLACE, METHOD=QUAD), this value is incremented by $\mathrm{rank}(\mathbf{X})$.
- n equals the effective number of subjects as displayed in the "Dimensions" table, unless this value equals 1, in which case n equals the number of levels of

the first G-side RANDOM effect specified. If the number of effective subjects equals 1 and there are no G-side random effects, n is determined as

$$n = \begin{cases} f - \text{rank}(\mathbf{X}) & \text{METHOD} = \text{RMPL}, \text{METHOD} = \text{RSPL} \\ f & \text{otherwise} \end{cases}$$

where f is the sum of frequencies used.

- n^* equals f or $f - \text{rank}(\mathbf{X})$ (for METHOD=RMPL and METHOD=RSPL), unless this value is less than $d + 2$, in which case $n^* = d + 2$.

GLMM, IC=PQ: For METHOD=MSPL, METHOD=MMPL, METHOD=LAPLACE, and METHOD=QUAD, the results are the same as for IC=Q. For METHOD=RSPL and METHOD=RMPL, d equals the number of effective covariance parameters plus $\text{rank}(\mathbf{X})$, and $n = n^*$ equals $f - \text{rank}(\mathbf{X})$. The formulas for the information criteria thus agree with Verbeke and Molenberghs (2000, Table 6.7, p. 74) and Vonesh and Chinchilli (1997, p. 263).

INITGLM

requests that the estimates from a generalized linear model fit (a model without random effects) be used as the starting values for the generalized linear mixed model. This option is the default for METHOD=LAPLACE and METHOD=QUAD.

INITITER=*number*

specifies the maximum number of iterations used when a generalized linear model is fit initially to derive starting values for the fixed effects; see the INITGLM option. By default, the initial fit involves at most four iteratively reweighted least squares updates. You can change the upper limit of initial iterations with *number*. If the model does not contain random effects, this option has no effect.

ITDETAILS

adds parameter estimates and gradients to the "Iteration History" table.

LIST

requests that the model program and variable lists be displayed. This is a debugging feature and is not normally needed. When you use programming statements to define your statistical model, this option enables you to examine the complete set of statements submitted for processing. See the section "Programming Statements" for more details about how to use SAS statements with the GLIMMIX procedure.

MAXLMMUPDATE=*number*

MAXOPT=*number*

specifies the maximum number of optimizations for doubly iterative estimation methods based on linearizations. After each optimization, a new pseudo-model is constructed through a Taylor series expansion. This step is known as the linear mixed model update. The MAXLMMUPDATE option limits the number of updates and thereby limits the number of optimizations. If this option is not specified, *number* is set equal to the value specified in the MAXITER= option in the NLOPTIONS statement. If no MAXITER= value is given, *number* defaults to 20.

METHOD=RSPL | MSPL | RMPL | MMPL | LAPLACE | QUAD< (*quad-options*) >
specifies the estimation method in a generalized linear mixed model (GLMM). The default is METHOD=RSPL.

Pseudo-Likelihood

Estimation methods ending in "PL" are pseudo-likelihood techniques. The first letter of the METHOD= identifier determines whether estimation is based on a residual likelihood ("R") or a maximum likelihood ("M"). The second letter identifies the expansion locus for the underlying approximation. Pseudo-likelihood methods for generalized linear mixed models can be cast in terms of Taylor series expansions (linearizations) of the GLMM. The expansion locus of the expansion is either the vector of random effects solutions ("S") or the mean of the random effects ("M"). The expansions are also referred to as the "S"ubject-specific and "M"arginal expansions. The abbreviation "PL" identifies the method as a pseudo-likelihood technique.

Residual methods account for the fixed effects in the construction of the objective function, which reduces the bias in covariance parameter estimates. Estimation methods involving Taylor series create pseudo-data for each optimization. Those data are transformed to have zero mean in a residual method. While the covariance parameter estimates in a residual method are the maximum likelihood estimates for the transformed problem, the fixed-effects estimates are (estimated) generalized least squares estimates. In a likelihood method that is not residual based, both the covariance parameters and the fixed-effects estimates are maximum likelihood estimates, but the former are known to have greater bias. In some problems, residual likelihood estimates of covariance parameters are unbiased.

For more information about linearization methods for generalized linear mixed models, see the section "Pseudo-likelihood Estimation Based on Linearization" on page 3188.

Maximum Likelihood with Laplace Approximation

If you choose METHOD=LAPLACE with a generalized linear mixed model, PROC GLIMMIX approximates the marginal likelihood by using Laplace's method. Twice the negative of the resulting log-likelihood approximation is the objective function that the procedure minimizes to determine parameter estimates. Laplace estimates typically exhibit better asymptotic behavior and less small-sample bias than pseudo-likelihood estimators. On the other hand, the class of models for which a Laplace approximation of the marginal log likelihood is available is much smaller compared to the class of models to which PL estimation can be applied.

To determine whether Laplace estimation can be applied in your model, consider the marginal distribution of the data in a mixed model

$$\begin{aligned} p(\mathbf{y}) &= \int p(\mathbf{y}|\boldsymbol{\gamma})\, p(\boldsymbol{\gamma})\; d\boldsymbol{\gamma} \\ &= \int \exp\left\{\log\{p(\mathbf{y}|\boldsymbol{\gamma})\} + \log\{p(\boldsymbol{\gamma})\}\right\}\; d\boldsymbol{\gamma} \\ &= \int \exp\left\{n f(\mathbf{y}, \boldsymbol{\gamma})\right\}\; d\boldsymbol{\gamma} \end{aligned}$$

The function $f(\mathbf{y}, \boldsymbol{\gamma})$ plays an important role in the Laplace approximation: it is a function of the joint distribution of the data and the random effects (see the section "Maximum Likelihood Estimation

Based on Laplace Approximation" on page 3192). In order to construct a Laplace approximation, PROC GLIMMIX requires a conditional log-likelihood $\log\{p(\mathbf{y}|\boldsymbol{\gamma})\}$ as well as the distribution of the G-side random effects. The random effects are always assumed to be normal with zero mean and covariance structure determined by the RANDOM statement. The conditional distribution is determined by the DIST= option of the MODEL statement or the default associated with a particular response type. Because a valid conditional distribution is required, R-side random effects are not permitted for METHOD=LAPLACE in the GLIMMIX procedure. In other words, the GLIMMIX procedure requires for METHOD=LAPLACE conditional independence without R-side overdispersion or covariance structure.

Because the marginal likelihood of the data is approximated numerically, certain features of the marginal distribution are not available—for example, you cannot display a marginal variance-covariance matrix. Also, the procedure includes both the fixed-effects parameters and the covariance parameters in the optimization for Laplace estimation. Consequently, this setting imposes some restrictions with respect to available options for Laplace estimation. Table 44.3 lists the options that are assumed for METHOD=LAPLACE, and Table 44.4 lists the options that are not compatible with this estimation method.

The section "Maximum Likelihood Estimation Based on Laplace Approximation" contains details about Laplace estimation in PROC GLIMMIX.

Maximum Likelihood with Adaptive Quadrature

If you choose METHOD=QUAD in a generalized linear mixed model, the GLIMMIX procedure approximates the marginal log likelihood with an adaptive Gauss-Hermite quadrature. Compared to METHOD=LAPLACE, the models for which parameters can be estimated by quadrature are further restricted. In addition to the conditional independence assumption and the absence of R-side covariance parameters, it is required that models suitable for METHOD=QUAD can be processed by subjects. (See the section "Processing by Subjects" on page 3218 about how the GLIMMIX procedure determines whether the data can be processed by subjects.) This in turn requires that all RANDOM statements have SUBJECT= effects and in the case of multiple SUBJECT= effects that these form a containment hierarchy.

In a containment hierarchy each effect is contained by another effect, and the effect contained by all is considered "the" effect for subject processing. For example, the SUBJECT= effects in the following statements form a containment hierarchy:

```
proc glimmix;
   class A B block;
   model y = A B A*B;
   random intercept / subject=block;
   random intercept / subject=A*block;
run;
```

The block effect is contained in the A*block interaction and the data are processed by block. The SUBJECT= effects in the following statements do not form a containment hierarchy:

```
proc glimmix;
   class A B block;
   model y = A B A*B;
   random intercept / subject=block;
   random block     / subject=A;
run;
```

The section "Maximum Likelihood Estimation Based on Adaptive Quadrature" on page 3195 contains important details about the computations involved with quadrature approximations. The section "Aspects Common to Adaptive Quadrature and Laplace Approximation" on page 3197 contains information about issues that apply to Laplace and adaptive quadrature, such as the computation of the prediction variance matrix and the determination of starting values.

You can specify the following *quad-options* for METHOD=QUAD in parentheses:

EBDETAILS
reports details about the empirical Bayes suboptimization process should this suboptimization fail.

EBSSFRAC=*r*
specifies the step-shortening fraction to be used while computing empirical Bayes estimates of the random effects. The default value is $r = 0.8$, and it is required that $r > 0$.

EBSSTOL=*r*
specifies the objective function tolerance for determining the cessation of step shortening while computing empirical Bayes estimates of the random effects, $r \geq 0$. The default value is *r*=1E–8.

EBSTEPS=*n*
specifies the maximum number of Newton steps for computing empirical Bayes estimates of random effects, $n \geq 0$. The default value is *n*=50.

EBSUBSTEPS=*n*
specifies the maximum number of step shortenings for computing empirical Bayes estimates of random effects. The default value is *n*=20, and it is required that $n \geq 0$.

EBTOL=*r*
specifies the convergence tolerance for empirical Bayes estimation, $r \geq 0$. The default value is $r = \epsilon \times 1\text{E}4$, where ϵ is the machine precision. This default value equals approximately 1E–12 on most machines.

INITPL=*number*
requests that adaptive quadrature commence after performing up to *number* pseudo-likelihood updates. The initial pseudo-likelihood (PL) steps (METHOD=MSPL) can be useful to provide good starting values for the quadrature algorithm. If you choose *number* large enough so that the initial PL estimation converges, the process is equivalent to starting a quadrature from the PL estimates of the fixed-effects and covariance parameters. Because this also makes available the PL random-effects solutions, the adaptive step of the quadrature that determines the number of quadrature points can take this information into account.

Note that you can combine the INITPL option with the NOINITGLM option in the PROC GLIMMIX statement to define a precise path for starting value construction to the GLIMMIX procedure. For example, the following statement generates starting values in these steps:

```
proc glimmix method=quad(initpl=5);
```

1. A GLM without random effects is fit initially to obtain as starting values for the fixed effects. The INITITER= option in the PROC GLIMMIX statement controls the number of iterations in this step.
2. Starting values for the covariance parameters are then obtained by MIVQUE0 estimation (Goodnight 1978a), using the fixed-effects parameter estimates from step 1.
3. With these values up to five pseudo-likelihood updates are computed.
4. The PL estimates for fixed-effects, covariance parameters, and the solutions for the random effects are then used to determine the number of quadrature points and used as the starting values for the quadrature.

The first step (GLM fixed-effects estimates) is omitted, if you modify the previous statement as follows:

```
proc glimmix method=quad(initpl=5) noinitglm;
```

The NOINITGLM option is the default of the pseudo-likelihood methods you select with the METHOD= option.

QCHECK

performs an adaptive recalculation of the objective function (–2 log likelihood) at the solution. The increment of the quadrature points, starting from the number of points used in the optimization, follows the same rules as the determination of the quadrature point sequence at the starting values (see the QFAC= and QMAX= suboptions). For example, the following statement estimates the parameters based on a quadrature with seven nodes in each dimension:

```
proc glimmix method=quad(qpoints=7 qcheck);
```

Because the default search sequence is $1, 3, 5, 7, 9, 11, 21, 31$, the QCHECK option computes the –2 log likelihood at the converged solution for $9, 11, 21$, and 31 quadrature points and reports relative differences to the converged value and among successive values. The ODS table produced by this option is named QuadCheck.

CAUTION: This option is useful to diagnose the sensitivity of the likelihood approximation at the solution. It does **not** diagnose the stability of the solution under changes in the number of quadrature points. For example, if increasing the number of points from 7 to 9 does not alter the objective function, this does not imply that a quadrature with 9 points would arrive at the same parameter estimates as a quadrature with 7 points.

QFAC=*r*

determines the step size for the quadrature point sequence. If the GLIMMIX procedure determines the quadrature nodes adaptively, the log likelihoods are computed for nodes in a predetermined sequence. If N_{min} and N_{max} denote the values from the QMIN= and QMAX= suboptions, respectively, the sequence for values less than 11 is constructed in increments of 2 starting at N_{min}. Values greater than 11 are incremented in steps of r. The default value is r=10. The default sequence, without specifying the QMIN=, QMAX=, or QFAC= option, is thus $1, 3, 5, 7, 9, 11, 21, 31$. By contrast, the following statement evaluates the sequence $8, 10, 30, 50$:

```
proc glimmix method=quad(qmin=8,qmax=51,qfac=20);
```

QMAX=*n*
specifies an upper bound for the number of quadrature points. The default is *n*=31.

QMIN=*n*
specifies a lower bound for the number of quadrature points. The default is *n*=1 and the value must be less than the QMAX= value.

QPOINTS=*n*
determines the number of quadrature points in each dimension of the integral. Note that if there are r random effects for each subject, the GLIMMIX procedure evaluates n^r conditional log likelihoods for each observation to compute one value of the objective function. Increasing the number of quadrature nodes can substantially increase the computational burden. If you choose QPOINTS=1, the quadrature approximation reduces to the Laplace approximation. If you do not specify the number of quadrature points, it is determined adaptively by increasing the number of nodes at the starting values. See the section "Aspects Common to Adaptive Quadrature and Laplace Approximation" on page 3197 for details.

QTOL=*r*
specifies a relative tolerance criterion for the successive evaluation of log likelihoods for different numbers of quadrature points. When the GLIMMIX procedure determines the number of quadrature points adaptively, the number of nodes are increased until the QMAX=*n* limit is reached or until two successive evaluations of the log likelihood have a relative change of less than r. In the latter case, the lesser number of quadrature nodes is used for the optimization.

The EBSSFRAC, EBSSTOL, EBSTEPS, EBSUBSTEPS, and EBTOL suboptions affect the suboptimization that leads to the empirical Bayes estimates of the random effects. Under normal circumstances, there is no reason to change from the default values. When the sub-optimizations fail, the optimization process can come to a halt. If the EBDETAILS option is in effect, you might be able to determine why the suboptimization fails and then adjust these values accordingly.

The QMIN, QMAX, QTOL, and QFAC suboptions determine the quadrature point search sequence for the adaptive component of estimation.

As for METHOD=LAPLACE, certain features of the marginal distribution are not available because the marginal likelihood of the data is approximated numerically. For example, you cannot display a marginal variance-covariance matrix. Also, the procedure includes both the fixed-effects and covariance parameters in the optimization for quadrature estimation. Consequently, this setting imposes some restrictions with respect to available options. Table 44.3 lists the options that are assumed for METHOD=QUAD and METHOD=LAPLACE, and Table 44.4 lists the options that are not compatible with these estimation methods.

Table 44.3 Defaults for METHOD=LAPLACE and METHOD=QUAD

Statement	Option
PROC GLIMMIX	NOPROFILE
PROC GLIMMIX	INITGLM
MODEL	NOCENTER

Table 44.4 Options Incompatible with METHOD=LAPLACE and METHOD=QUAD

Statement	Option
PROC GLIMMIX	EXPHESSIAN
PROC GLIMMIX	SCOREMOD
PROC GLIMMIX	SCORING
PROC GLIMMIX	PROFILE
MODEL	DDFM=KENWARDROGER
MODEL	DDFM=SATTERTHWAITE
MODEL	STDCOEF
RANDOM	RESIDUAL
RANDOM _RESIDUAL_	All R-side random effects
RANDOM	V
RANDOM	VC
RANDOM	VCI
RANDOM	VCORR
RANDOM	VI

In addition to the options displayed in Table 44.4, the NOBOUND option in the PROC GLIMMIX and the NOBOUND option in the PARMS statements are not available with METHOD=QUAD. Unbounding the covariance parameter estimates is possible with METHOD=LAPLACE, however.

No Random Effects Present

If the model does not contain G-side random effects or contains only a single overdispersion component, then the model belongs to the family of (overdispersed) generalized linear models if the distribution is known or the quasi-likelihood models for independent data if the distribution is not known. The GLIMMIX procedure then estimates model parameters by the following techniques:

- normally distributed data: residual maximum likelihood
- nonnormal data: maximum likelihood
- data with unknown distribution: quasi-likelihood

The METHOD= specification then has only an effect with respect to the divisor used in estimating the overdispersion component. With a residual method, the divisor is $f - k$, where f denotes the sum of the frequencies and k is the rank of $\mathbf{X}$. Otherwise, the divisor is f.

NAMELEN=*number*
specifies the length to which long effect names are shortened. The default and minimum value is 20.

NOBOUND
requests the removal of boundary constraints on covariance and scale parameters in mixed models. For example, variance components have a default lower boundary constraint of 0, and the NOBOUND option allows their estimates to be negative.

The NOBOUND option cannot be used for adaptive quadrature estimation with METHOD=QUAD. The scaling of the quadrature abscissas requires an inverse Cholesky root that is possibly not well defined when the **G** matrix of the mixed model is negative definite or indefinite. The Laplace approximation (METHOD=LAPLACE) is not subject to this limitation.

NOBSDETAIL
adds detailed information to the "Number of Observations" table to reflect how many observations were excluded from the analysis and for which reason.

NOCLPRINT< =*number* >
suppresses the display of the "Class Level Information" table, if you do not specify *number*. If you specify *number*, only levels with totals that are less than *number* are listed in the table.

NOFIT
suppresses fitting of the model. When the NOFIT option is in effect, PROC GLIMMIX produces the "Model Information," "Class Level Information," "Number of Observations," and "Dimensions" tables. These can be helpful to gauge the computational effort required to fit the model. For example, the "Dimensions" table informs you as to whether the GLIMMIX procedure processes the data by subjects, which is typically more computationally efficient than processing the data as a single subject. See the section "Processing by Subjects" for more information.

If you request a radial smooth with knot selection by k-d tree methods, PROC GLIMMIX also computes the knot locations of the smoother. You can then examine the knots without fitting the model. This enables you to try out different knot construction methods and bucket sizes. See the KNOTMETHOD=KDTREE option (and its suboptions) of the RANDOM statement.

If you combine the NOFIT option with the OUTDESIGN option, you can write the **X** and/or **Z** matrix of your model to a SAS data set without fitting the model.

NOINITGLM
requests that the starting values for the fixed effects not be obtained by first fitting a generalized linear model. This option is the default for the pseudo-likelihood estimation methods and for the linear mixed model. For the pseudo-likelihood methods, starting values can be implicitly defined based on an initial pseudo-data set derived from the data and the link function. For linear mixed models, starting values for the fixed effects are not necessary. The NOINITGLM option is useful in conjunction with the INITPL= suboption of METHOD=QUAD in order to perform initial pseudo-likelihood steps prior to an adaptive quadrature.

NOITPRINT
suppresses the display of the "Iteration History" table.

NOPROFILE
includes the scale parameter ϕ into the optimization for models that have such a parameter (see Table 44.19). By default, the GLIMMIX procedure profiles scale parameters from the optimization in mixed models. In generalized linear models, scale parameters are not profiled.

NOREML
determines the denominator for the computation of the scale parameter in a GLM for normal data and for overdispersion parameters. By default, the GLIMMIX procedure computes the scale parameter for the normal distribution as

$$\widehat{\phi} = \sum_{i=1}^{n} \frac{f_i (y_i - \widehat{y}_i)^2}{f - k}$$

where k is the rank of $\mathbf{X}$, f_i is the frequency associated with the ith observation, and $f = \sum f_i$. Similarly, the overdispersion parameter in an overdispersed GLM is estimated by the ratio of the Pearson statistic and $(f - k)$. If the NOREML option is in effect, the denominators are replaced by f, the sum of the frequencies. In a GLM for normal data, this yields the maximum likelihood estimate of the error variance. For this case, the NOREML option is a convenient way to change from REML to ML estimation.

In GLMM models fit by pseudo-likelihood methods, the NOREML option changes the estimation method to the nonresidual form. See the METHOD= option for the distinction between residual and nonresidual estimation methods.

ODDSRATIO

OR

requests that odds ratios be added to the output when applicable. Odds ratios and their confidence limits are reported only for models with logit, cumulative logit, or generalized logit link. Specifying the ODDSRATIO option in the PROC GLIMMIX statement has the same effect as specifying the ODDSRATIO option in the MODEL statement and in all LSMEANS statements. Note that the ODDSRATIO option in the MODEL statement has several suboptions that enable you to construct customized odds ratios. These suboptions are available only through the MODEL statement. For details about the interpretation and computation of odds and odds ratios with the GLIMMIX procedure, see the section "Odds and Odds Ratio Estimation" on page 3225.

ORDER=DATA | FORMATTED | FREQ | INTERNAL

specifies the sort order for the levels of the classification variables (which are specified in the CLASS statement).

This ordering determines which parameters in the model correspond to each level in the data, so the ORDER= option can be useful when you use CONTRAST or ESTIMATE statements. This option applies to the levels for all classification variables, except when you use the (default) ORDER=FORMATTED option with numeric classification variables that have no explicit format. In that case, the levels of such variables are ordered by their internal value.

The ORDER= option can take the following values:

Value of ORDER=	Levels Sorted By
DATA	Order of appearance in the input data set
FORMATTED	External formatted value, except for numeric variables with no explicit format, which are sorted by their unformatted (internal) value
FREQ	Descending frequency count; levels with the most observations come first in the order
INTERNAL	Unformatted value

By default, ORDER=FORMATTED. For ORDER=FORMATTED and ORDER=INTERNAL, the sort order is machine-dependent.

When the response variable appears in a CLASS statement, the ORDER= option in the PROC GLIMMIX statement applies to its sort order. Specification of a *response-option* in the MODEL

statement overrides the ORDER= option in the PROC GLIMMIX statement. For example, in the following statements the sort order of the wheeze variable is determined by the formatted value (default):

```
proc glimmix order=data;
   class city;
   model wheeze = city age / dist=binary s;
run;
```

The ORDER= option in the PROC GLIMMIX statement has no effect on the sort order of the wheeze variable because it does not appear in the CLASS statement. However, in the following statements the sort order of the wheeze variable is determined by the order of appearance in the input data set because the response variable appears in the CLASS statement:

```
proc glimmix order=data;
   class city wheeze;
   model wheeze = city age / dist=binary s;
run;
```

For more information about sort order, see the chapter on the SORT procedure in the *Base SAS Procedures Guide* and the discussion of BY-group processing in *SAS Language Reference: Concepts*.

OUTDESIGN<(*options*)><=*SAS-data-set*>

creates a data set that contains the contents of the **X** and **Z** matrix. If the data are processed by subjects as shown in the "Dimensions" table, then the **Z** matrix saved to the data set corresponds to a single subject. By default, the GLIMMIX procedure includes in the OUTDESIGN data set the **X** and **Z** matrix (if present) and the variables in the input data set. You can specify the following *options* in parentheses to control the contents of the OUTDESIGN data set:

NAMES

produces tables associating columns in the OUTDESIGN data set with fixed-effects parameter estimates and random-effects solutions.

NOMISS

excludes from the OUTDESIGN data set observations that were not used in the analysis.

NOVAR

excludes from the OUTDESIGN data set variables from the input data set. Variables listed in the BY and ID statements and variables needed for identification of SUBJECT= effects are always included in the OUTDESIGN data set.

X<=*prefix*>

saves the contents of the **X** matrix. The optional *prefix* is used to name the columns. The default naming prefix is "_X".

Z<=*prefix*>

saves the contents of the **Z** matrix. The optional *prefix* is used to name the columns. The default naming prefix is "_Z".

The order of the observations in the OUTDESIGN data set is the same as the order of the input data set. If you do not specify a data set with the OUTDESIGN option, the procedure uses the DATA*n* convention to name the data set.

PCONV=*r*

specifies the parameter estimate convergence criterion for doubly iterative estimation methods. The GLIMMIX procedure applies this criterion to fixed-effects estimates and covariance parameter estimates. Suppose $\widehat{\psi}_i^{(u)}$ denotes the estimate of the ith parameter at the uth optimization. The procedure terminates the doubly iterative process if the largest value

$$2 \times \frac{|\widehat{\psi}_i^{(u)} - \widehat{\psi}_i^{(u-1)}|}{|\widehat{\psi}_i^{(u)}| + |\widehat{\psi}_i^{(u-1)}|}$$

is less than r. To check an absolute convergence criteria as well, you can set the ABSPCONV= option in the PROC GLIMMIX statement. The default value for r is 1E8 times the machine epsilon, a product that equals about 1E–8 on most machines.

Note that this convergence criterion does not affect the convergence criteria applied within any individual optimization. In order to change the convergence behavior within an optimization, you can use the ABSCONV=, ABSFCONV=, ABSGCONV=, ABSXCONV=, FCONV=, or GCONV= option in the NLOPTIONS statement.

PLOTS <(*global-plot-options***)> <=***plot-request* **<(***options***)> >**

PLOTS <(*global-plot-options***)> <=(***plot-request* **<(***options***)> <... *plot-request* <(***options***)> >)>**

requests that the GLIMMIX procedure produce statistical graphics via ODS Graphics.

ODS Graphics must be enabled before plots can be requested. For example:

```
ods graphics on;

proc glimmix data=plants;
   class Block Type;
   model StemLength = Block Type;
   lsmeans type / diff=control plots=controlplot;
run;

ods graphics off;
```

For more information about enabling and disabling ODS Graphics, see the section "Enabling and Disabling ODS Graphics" on page 606 in Chapter 21, "Statistical Graphics Using ODS."

For examples of the basic statistical graphics produced by the GLIMMIX procedure and aspects of their computation and interpretation, see the section "ODS Graphics" on page 3249 in this chapter. You can also request statistical graphics for least squares means through the PLOTS option in the LSMEANS statement, which gives you more control over the display compared to the PLOTS option in the PROC GLIMMIX statement.

Global Plot Options

The *global-plot-options* apply to all relevant plots generated by the GLIMMIX procedure. The *global-plot-options* supported by the GLIMMIX procedure are as follows:

OBSNO
uses the data set observation number to identify observations in tooltips, provided that the observation number can be determined. Otherwise, the number displayed in tooltips is the index of the observation as it is used in the analysis within the BY group.

UNPACKPANEL
UNPACK
displays each graph separately. (By default, some graphs can appear together in a single panel.)

Specific Plot Options

The following listing describes the specific plots and their *options*.

ALL
requests that all plots appropriate for the analysis be produced. In models with G-side random effects, residual plots are based on conditional residuals (by using the BLUPs of random effects) on the linear (linked) scale. Plots of least squares means differences are produced for LSMEANS statements without options that would contradict such a display.

ANOMPLOT
ANOM
requests an analysis of means display in which least squares means are compared against an average least squares mean (Ott 1967; Nelson 1982, 1991, 1993). See the DIFF= option in the LSMEANS statement for the computation of this average. Least squares mean ANOM plots are produced only for those fixed effects that are listed in LSMEANS statements that have options that do not contradict the display. For example, if you request ANOM plots with the PLOTS= option in the PROC GLIMMIX statement, the following LSMEANS statements produce analysis of mean plots for effects A and C:

```
lsmeans A / diff=anom;
lsmeans B / diff;
lsmeans C;
```

The DIFF option in the second LSMEANS statement implies all pairwise differences.

When differences against the average LS-mean are adjusted for multiplicity with the ADJUST=NELSON option in the LSMEANS statement, the ANOMPLOT display is adjusted accordingly.

BOXPLOT <(*boxplot-options*)>
requests box plots for the effects in your model that consist of classification effects only. Note that these effects can involve more than one classification variable (interaction and nested effects), but cannot contain any continuous variables. By default, the BOXPLOT request produces box plots of (conditional) residuals for the qualifying effects in the MODEL and RANDOM statements. See the discussion of the *boxplot-options* in a later section for information about how to tune your box plot request.

CONTROLPLOT

CONTROL

requests a display in which least squares means are visually compared against a reference level. LS-mean control plots are produced only for those fixed effects that are listed in LSMEANS statements that have options that do not contradict with the display. For example, the following statements produce control plots for effects A and C if you specify PLOTS=CONTROL in the PROC GLIMMIX statement:

```
lsmeans A / diff=control('1');
lsmeans B / diff;
lsmeans C;
```

The DIFF option in the second LSMEANS statement implies all pairwise differences.

When differences against a control level are adjusted for multiplicity with the ADJUST= option in the LSMEANS statement, the control plot display is adjusted accordingly.

DIFFPLOT< (*diffplot-options*) >

DIFFOGRAM < (*diffplot-options*) >

DIFF< (*diffplot-options*) >

requests a display of all pairwise least squares mean differences and their significance. When constructed from arithmetic means, the display is also known as a "mean-mean scatter plot" (Hsu 1996; Hsu and Peruggia 1994). For each comparison a line segment, centered at the LS-means in the pair, is drawn. The length of the segment corresponds to the projected width of a confidence interval for the least squares mean difference. Segments that fail to cross the 45-degree reference line correspond to significant least squares mean differences.

If you specify the ADJUST= option in the LSMEANS statement, the lengths of the line segments are adjusted for multiplicity.

LS-mean difference plots are produced only for those fixed effects listed in LSMEANS statements that have options that do not conflict with the display. For example, the following statements request differences against a control level for the A effect, all pairwise differences for the B effect, and the least squares means for the C effect:

```
lsmeans A / diff=control('1');
lsmeans B / diff;
lsmeans C;
```

The DIFF= type in the first statement contradicts a display of all pairwise differences. Difference plots are produced only for the B and C effects if you specify PLOTS=DIFF in the PROC GLIMMIX statement.

You can specify the following *diffplot-options*. The ABS and NOABS options determine the positioning of the line segments in the plot. When the ABS option is in effect (this is the default) all line segments are shown on the same side of the reference line. The NOABS option separates comparisons according to the sign of the difference. The CENTER option marks the center point for each comparison. This point corresponds to the intersection of two least squares means. The NOLINES option suppresses the display of the line segments that represent the confidence bounds for the differences of the least squares means. The NOLINES option implies the CENTER option.

The default is to draw line segments in the upper portion of the plot area without marking the center point.

MEANPLOT< (*meanplot-options*) >
requests a display of the least squares means of effects specified in LSMEANS statements. The following *meanplot-options* affect the display. Upper and lower confidence limits are plotted when the CL option is used. When the CLBAND option is in effect, confidence limits are shown as bands and the means are connected. By default, least squares means are not joined by lines. You can achieve that effect with the JOIN or CONNECT option. Least squares means are displayed in the same order in which they appear in the "Least Squares Means" table. You can change that order for plotting purposes with the ASCENDING and DESCENDING options. The ILINK option requests that results be displayed on the inverse linked (the data) scale.

Note that there is also a MEANPLOT suboption of the PLOTS= option in the LSMEANS statement. In addition to the *meanplot-options* just described, you can also specify classification effects that give you more control over the display of interaction means through the PLOTBY= and SLICEBY= options. To display interaction means, you typically want to use the MEANPLOT option in the LSMEANS statement. For example, the next statement requests a plot in which the levels of A are placed on the horizontal axis and the means that belong to the same level of B are joined by lines:

```
lsmeans A*B / plot=meanplot(sliceby=b join);
```

NONE
requests that no plots be produced.

ODDSRATIO < (*oddsratioplot-options*) >
requests a display of odds ratios and their confidence limits when the link function permits the computation of odds ratios (see the ODDSRATIO option in the MODEL statement). Possible suboptions of the ODDSRATIO plot request are described below under the heading "Odds Ratio Plot Options."

RESIDUALPANEL< (*residualplot-options*) >
requests a paneled display constructed from raw residuals. The panel consists of a plot of the residuals against the linear predictor or predicted mean, a histogram with normal density overlaid, a *Q*-*Q* plot, and a box plot of the residuals. The *residualplot-options* enable you to specify which type of residual is being graphed. These are further discussed below under the heading "Residual Plot Options."

STUDENTPANEL< (*residualplot-options*) >
requests a paneled display constructed from studentized residuals. The same panel organization is applied as for the RESIDUALPANEL plot type.

PEARSONPANEL< (*residualplot-options*) >
requests a paneled display constructed from Pearson residuals. The same panel organization is applied as for the RESIDUALPANEL plot type.

Residual Plot Options

The *residualplot-options* apply to the RESIDUALPANEL, STUDENTPANEL, and PEARSONPANEL displays. The primary function of these options is to control which type of a residual to display. The four types correspond to *keyword-options* as for output statistics in the OUTPUT statement. The *residualplot-options* take on the following values:

BLUP

CONDITIONAL
uses the predictors of the random effects in computing the residual.

ILINK

NONLINEAR
computes the residual on the inverse linked scale (the data scale).

NOBLUP

MARGINAL
does not use the predictors of the random effects in computing the residual.

NOILINK

LINEAR
computes the residual on the linked scale.

UNPACK
produces separate plots from the elements of the panel.

You can list a plot request one or more times with different options. For example, the following statements request a panel of marginal raw residuals, individual plots generated from a panel of the conditional raw residuals, and a panel of marginal studentized residuals:

```
ods graphics on;
proc glimmix plots=(ResidualPanel(marginal)
                    ResidualPanel(unpack conditional)
                    StudentPanel(marginal));
```

The default is to compute conditional residuals on the linear scale if the model contains G-side random effects (BLUP NOILINK). Not all combinations of the BLUP/NOBLUP and ILINK/NOILINK suboptions are possible for all residual types and models. For details, see the description of output statistics for the OUTPUT statement. Pearson residuals are always displayed against the linear predictor; all other residuals are graphed versus the linear predictor if the NOILINK suboption is in effect (default), and against the corresponding prediction on the mean scale if the ILINK option is in effect. See Table 44.14 for a definition of the residual quantities and exclusions.

Box Plot Options

The *boxplot-options* determine whether box plots are produced for residuals or for residuals and observed values, and for which model effects the box plots are constructed. The available *boxplot-options* are as follows:

BLOCK
BLOCKLEGEND
displays levels of up to four classification variables of the box plot effect by using block legends instead of axis tick values.

BLUP
CONDITIONAL
constructs box plots from conditional residuals—that is, residuals that use the estimated BLUPs of random effects.

FIXED
produces box plots for all fixed effects (MODEL statement) consisting entirely of classification variables.

GROUP
produces box plots for all GROUP= effects in RANDOM statements consisting entirely of classification variables.

ILINK
NONLINEAR
computes the residual on the scale of the data (the inverse linked scale).

NOBLUP
MARGINAL
constructs box plots from marginal residuals.

NOILINK
LINEAR
computes the residual on the linked scale.

NPANELPOS=*number*
specifies the number of box positions on the graphic and provides the capability to break a box plot into multiple graphics. If *number* is negative, no balancing of the number of boxes takes place and *number* is the maximum number of boxes per graphic. If *number* is positive, the number of boxes per graphic is balanced. For example, suppose that variable A has 125 levels. The following statements request that the number of boxes per plot results be balanced and result in six plots with 18 boxes each and one plot with 17 boxes:

```
ods graphics on;
proc glimmix plots=boxplot(npanelpos=20);
   class A;
   model y = A;
run;
```

If *number* is zero (this is the default), all levels of the effect are displayed in a single plot.

OBSERVED
adds box plots of the observed data for the selected effects.

PEARSON
constructs box plots from Pearson residuals rather than from the default residuals.

PSEUDO
adds box plots of the pseudo-data for the selected effects. This option is available only for the pseudo-likelihood estimation methods that construct pseudo-data.

RANDOM
produces box plots for all effects in RANDOM statements that consist entirely of classification variables. This does not include effects specified in the GROUP= or SUBJECT= option of the RANDOM statements.

RAW
constructs box plots from raw residuals (observed minus predicted).

STUDENT
constructs box plots from studentized residuals rather than from the default residuals.

SUBJECT
produces box plots for all SUBJECT= effects in RANDOM statements consisting entirely of classification variables.

USEINDEX
uses as the horizontal axis label the index of the effect level, rather than the formatted value(s). For classification variables with many levels or model effects that involve multiple classification variables, the formatted values identifying the effect levels might take up too much space as axis tick values, leading to extensive thinning. The USEINDEX option replaces tick values constructed from formatted values with the internal level number.

By default, box plots of residuals are constructed from the raw conditional residuals (on the linked scale) in linear mixed models and from Pearson residuals in all other models. Note that not all combinations of the BLUP/NOBLUP and ILINK/NOILINK suboptions are possible for all residual types and models. For details, see the description of output statistics for the OUTPUT statement.

Odds Ratio Plot Options

The *oddsratioplot-options* determine the display of odds ratios and their confidence limits. The computation of the odds ratios follows the ODDSRATIO option in the MODEL statement. The available *oddsratioplot-options* are as follows:

LOGBASE= 2 | E | 10
log-scales the odds ratio axis.

NPANELPOS=*n*
provides the capability to break an odds ratio plot into multiple graphics having at most $|n|$ odds ratios per graphic. If n is positive, then the number of odds ratios per graphic is balanced. If n is negative, then no balancing of the number of odds ratios takes place. For

example, suppose you want to display 21 odds ratios. Then NPANELPOS=20 displays two plots, the first with 11 and the second with 10 odds ratios, and NPANELPOS=–20 displays 20 odds ratios in the first plot and a single odds ratio in the second. If *n*=0 (this is the default), then all odds ratios are displayed in a single plot.

ORDER=ASCENDING | DESCENDING
displays the odds ratios in sorted order. By default the odds ratios are displayed in the order in which they appear in the "Odds Ratio Estimates" table.

RANGE=(< *min* > < ,*max* >) | CLIP
specifies the range of odds ratios to display. If you specify RANGE=CLIP, then the confidence intervals are clipped and the range contains the minimum and maximum odds ratios. By default the range of view captures the extent of the odds ratio confidence intervals.

STATS
adds the numeric values of the odds ratio and its confidence limits to the graphic.

PROFILE
requests that scale parameters be profiled from the optimization, if possible. This is the default for generalized linear mixed models. In generalized linear models with normally distributed data, you can use the PROFILE option to request profiling of the residual variance.

SCOREMOD
requests that the Hessian matrix in GLMMs be based on a modified scoring algorithm, provided that PROC GLIMMIX is in scoring mode when the Hessian is evaluated. The procedure is in scoring mode during iteration, if the optimization technique requires second derivatives, the SCORING=*n* option is specified, and the iteration count has not exceeded *n*. The procedure also computes the expected (scoring) Hessian matrix when you use the EXPHESSIAN option in the PROC GLIMMIX statement.

The SCOREMOD option has no effect if the SCORING= or EXPHESSIAN option is not specified. The nature of the SCOREMOD modification to the expected Hessian computation is shown in Table 44.21, in the section "Pseudo-likelihood Estimation Based on Linearization" on page 3188. The modification can improve the convergence behavior of the GLMM compared to standard Fisher scoring and can provide a better approximation of the variability of the covariance parameters. For more details, see the section "Estimated Precision of Estimates" on page 3189.

SCORING=*number*
requests that Fisher scoring be used in association with the estimation method up to iteration *number*. By default, no scoring is applied. When you use the SCORING= option and PROC GLIMMIX converges without stopping the scoring algorithm, the procedure uses the expected Hessian matrix to compute approximate standard errors for the covariance parameters instead of the observed Hessian. If necessary, the standard errors of the covariance parameters as well as the output from the ASYCOV and ASYCORR options are adjusted.

If scoring stopped prior to convergence and you want to use the expected Hessian matrix in the computation of standard errors, use the EXPHESSIAN option in the PROC GLIMMIX statement.

Scoring is not possible in models for nominal data. It is also not possible for GLMs with unknown distribution or for those outside the exponential family. If you perform quasi-likelihood estimation, the GLIMMIX procedure is always in scoring mode and the SCORING= option has no effect. See

the section "Quasi-likelihood for Independent Data" for a description of the types of models where GLIMMIX applies quasi-likelihood estimation.

The SCORING= option has no effect for optimization methods that do not involve second derivatives. See the TECHNIQUE= option in the NLOPTIONS statement and the section "Choosing an Optimization Algorithm" on page 501 in Chapter 19, "Shared Concepts and Topics," for details about first- and second-order algorithms.

SINGCHOL=*number*
tunes the singularity criterion in Cholesky decompositions. The default is 1E4 times the machine epsilon; this product is approximately 1E–12 on most computers.

SINGRES=*number*
sets the tolerance for which the residual variance is considered to be zero. The default is 1E4 times the machine epsilon; this product is approximately 1E–12 on most computers.

SINGULAR=*number*
tunes the general singularity criterion applied by the GLIMMIX procedure in divisions and inversions. The default is 1E4 times the machine epsilon; this product is approximately 1E–12 on most computers.

STARTGLM
is an alias of the INITGLM option.

SUBGRADIENT< =*SAS-data-set* **>**
SUBGRAD< =*SAS-data-set* **>**
creates a data set with information about the gradient of the objective function. The contents and organization of the SUBGRADIENT= data set depend on the type of model. The following paragraphs describe the SUBGRADIENT= data set for the two major estimation modes. See the section "GLM Mode or GLMM Mode" on page 3203 for details about the estimation modes of the GLIMMIX procedure.

GLMM Mode If the GLIMMIX procedure operates in GLMM mode, the SUBGRADIENT= data set contains as many observations as there are usable subjects in the analysis. The maximum number of usable subjects is displayed in the "Dimensions" table. Gradient information is not written to the data set for subjects who do not contribute valid observations to the analysis. Note that the objective function in the "Iteration History" table is in terms of the –2 log (residual, pseudo-) likelihood. The gradients in the SUBGRADIENT= data set are gradients of that objective function.

The gradients are evaluated at the final solution of the estimation problem. If the GLIMMIX procedure fails to converge, then the information in the SUBGRADIENT= data set corresponds to the gradient evaluated at the last iteration or optimization.

The number of gradients saved to the SUBGRADIENT= data set equals the number of parameters in the optimization. For example, with METHOD=LAPLACE or METHOD=QUAD the fixed-effects parameters and the covariance parameters take part in the optimization. The order in which the gradients appear in the data set equals the order in which the gradients are displayed when the ITDETAILS option is in effect: gradients for fixed-effects parameters precede those for covariance parameters, and gradients are not reported for singular columns in the $\mathbf{X}'\mathbf{X}$ matrix. In models where the residual variance is profiled from the optimization, a

subject-specific gradient is not reported for the residual variance. To decompose this gradient by subjects, add the NOPROFILE option in the PROC GLIMMIX statement. When the subject-specific gradients in the SUBGRADIENT= data set are summed, the totals equal the values reported by the GRADIENT option.

GLM Mode When you fit a generalized linear model (GLM) or a GLM with overdispersion, the SUBGRADIENT= data set contains the observation-wise gradients of the negative log-likelihood function with respect to the parameter estimates. Note that this corresponds to the objective function in GLMs as displayed in the "Iteration History" table. However, the gradients displayed in the "Iteration History" for GLMs—when the ITDETAILS option is in effect—are possibly those of the centered and scaled coefficients. The gradients reported in the "Parameter Estimates" table and in the SUBGRADIENT= data set are gradients with respect to the uncentered and unscaled coefficients.

The gradients are evaluated at the final estimates. If the model does not converge, the gradients contain missing values. The gradients appear in the SUBGRADIENT= data set in the same order as in the "Parameter Estimates" table, with singular columns removed.

The variables from the input data set are added to the SUBGRADIENT= data set in GLM mode. The data set is organized in the same way as the input data set; observations that do not contribute to the analysis are transferred to the SUBGRADIENT= data set, but gradients are calculated only for observations that take part in the analysis. If you use an ID statement, then only the variables in the ID statement are transferred to the SUBGRADIENT= data set.

BY Statement

BY *variables* **;**

You can specify a BY statement with PROC GLIMMIX to obtain separate analyses of observations in groups that are defined by the BY variables. When a BY statement appears, the procedure expects the input data set to be sorted in order of the BY variables. If you specify more than one BY statement, only the last one specified is used.

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

- Sort the data by using the SORT procedure with a similar BY statement.
- Specify the NOTSORTED or DESCENDING option in the BY statement for the GLIMMIX procedure. The NOTSORTED option does not mean that the data are unsorted but rather that the data are arranged in groups (according to values of the BY variables) and that these groups are not necessarily in alphabetical or increasing numeric order.
- Create an index on the BY variables by using the DATASETS procedure (in Base SAS software).

Since sorting the data changes the order in which PROC GLMMIX reads observations, the sort order for the levels of the CLASS variables might be affected if you have also specified ORDER=DATA in the

PROC GLIMMIX statement. This, in turn, affects specifications in the CONTRAST, ESTIMATE, or LSMESTIMATE statement.

For more information about BY-group processing, see the discussion in *SAS Language Reference: Concepts*. For more information about the DATASETS procedure, see the discussion in the *Base SAS Procedures Guide*.

CLASS Statement

CLASS *variable* <**(REF=** *option*<**)**> ... <*variable* <**(REF=** *option*<**)**>> <*/ global-options*> **;**

The CLASS statement names the classification variables to be used in the model. Typical classification variables are Treatment, Sex, Race, Group, and Replication. If you use the CLASS statement, it must appear before the MODEL statement.

Classification variables can be either character or numeric. By default, class levels are determined from the entire set of formatted values of the CLASS variables.

NOTE: Prior to SAS 9, class levels were determined by using no more than the first 16 characters of the formatted values. To revert to this previous behavior, you can use the TRUNCATE option in the CLASS statement.

In any case, you can use formats to group values into levels. See the discussion of the FORMAT procedure in the *Base SAS Procedures Guide* and the discussions of the FORMAT statement and SAS formats in *SAS Formats and Informats: Reference*. You can adjust the order of CLASS variable levels with the ORDER= option in the PROC GLIMMIX statement.

You can specify the following REF= option to indicate how the levels of an individual classification variable are to be ordered by enclosing it in parentheses after the variable name:

REF=*'level'* | **FIRST** | **LAST**
specifies a level of the classification variable to be put at the end of the list of levels. This level thus corresponds to the reference level in the usual interpretation of the estimates with PROC GLIMMIX's singular parameterization. You can specify the *level* of the variable to use as the reference level; specify a value that corresponds to the formatted value of the variable if a format is assigned. Alternatively, you can specify REF=FIRST to designate that the first ordered level serve as the reference, or REF=LAST to designate that the last ordered level serve as the reference. To specify that REF=FIRST or REF=LAST be used for all classification variables, use the REF= *global-option* after the slash (/) in the CLASS statement.

You can specify the following *global-options* in the CLASS statement after a slash (/):

REF=FIRST | **LAST**
specifies a level of all classification variables to be put at the end of the list of levels. This level thus corresponds to the reference level in the usual interpretation of the estimates with PROC GLIMMIX's singular parameterization. Specify REF=FIRST to designate that the first ordered level for each classification variable serve as the reference. Specify REF=LAST to designate that the last ordered level serve as the reference. This option applies to all the variables specified in the CLASS statement. To specify different reference levels for different classification variables, use REF= options for individual variables.

TRUNCATE
specifies that class levels be determined by using only up to the first 16 characters of the formatted values of CLASS variables. When formatted values are longer than 16 characters, you can use this option to revert to the levels as determined in releases prior to SAS 9.

CODE Statement

CODE < *options* > **;**

The CODE statement writes SAS DATA step code for computing predicted values of the fitted model either to a file or to a catalog entry. This code can then be included in a DATA step to score new data.

Table 44.5 summarizes the *options* available in the CODE statement.

Table 44.5 CODE Statement Options

Option	Description
CATALOG=	Names the catalog entry where the generated code is saved
DUMMIES	Retains the dummy variables in the data set
ERROR	Computes the error function
FILE=	Names the file where the generated code is saved
FORMAT=	Specifies the numeric format for the regression coefficients
GROUP=	Specifies the group identifier for array names and statement labels
IMPUTE	Imputes predicted values for observations with missing or invalid covariates
LINESIZE=	Specifies the line size of the generated code
LOOKUP=	Specifies the algorithm for looking up CLASS levels
RESIDUAL	Computes residuals

For details about the syntax of the CODE statement, see the section "CODE Statement" on page 395 in Chapter 19, "Shared Concepts and Topics."

CONTRAST Statement

CONTRAST '*label*' *contrast-specification*
< **,** *contrast-specification* > < **,** . . . >
< **/** *options* > **;**

The CONTRAST statement provides a mechanism for obtaining custom hypothesis tests. It is patterned after the CONTRAST statement in PROC MIXED and enables you to select an appropriate inference space (McLean, Sanders, and Stroup 1991). The GLIMMIX procedure gives you greater flexibility in entering contrast coefficients for random effects, however, because it permits the usual *value*-oriented positional syntax for entering contrast coefficients, as well as a level-oriented syntax that simplifies entering coefficients for interaction terms and is designed to work with constructed effects that are defined through the experimental EFFECT statement. The differences between the traditional and new-style coefficient syntax are explained in detail in the section "Positional and Nonpositional Syntax for Contrast Coefficients" on page 3232.

You can test the hypothesis $\mathbf{L}'\boldsymbol{\phi} = \mathbf{0}$, where $\mathbf{L}' = [\mathbf{K}'\ \mathbf{M}']$ and $\boldsymbol{\phi}' = [\boldsymbol{\beta}'\ \boldsymbol{\gamma}']$, in several inference spaces. The inference space corresponds to the choice of $\mathbf{M}$. When $\mathbf{M} = \mathbf{0}$, your inferences apply to the entire population from which the random effects are sampled; this is known as the *broad* inference space. When all elements of $\mathbf{M}$ are nonzero, your inferences apply only to the observed levels of the random effects. This is known as the *narrow* inference space, and you can also choose it by specifying all of the random effects as fixed. The GLM procedure uses the narrow inference space. Finally, by zeroing portions of $\mathbf{M}$ corresponding to selected main effects and interactions, you can choose *intermediate* inference spaces. The broad inference space is usually the most appropriate; it is used when you do not specify random effects in the CONTRAST statement.

In the CONTRAST statement,

label identifies the contrast in the table. A label is required for every contrast specified. Labels can be up to 200 characters and must be enclosed in quotes.

contrast-specification identifies the fixed effects and random effects and their coefficients from which the $\mathbf{L}$ matrix is formed. The syntax representation of a *contrast-specification* is
< *fixed-effect values* . . . > < | *random-effect values* . . . >

fixed-effect identifies an effect that appears in the MODEL statement. The keyword INTERCEPT can be used as an effect when an intercept is fitted in the model. You do not need to include all effects that are in the MODEL statement.

random-effect identifies an effect that appears in the RANDOM statement. The first random effect must follow a vertical bar (|); however, random effects do not have to be specified.

values are constants that are elements of the $\mathbf{L}$ matrix associated with the fixed and random effects. There are two basic methods of specifying the entries of the $\mathbf{L}$ matrix. The traditional representation—also known as the positional syntax—relies on entering coefficients in the position they assume in the $\mathbf{L}$ matrix. For example, in the following statements the elements of $\mathbf{L}$ associated with the b main effect receive a 1 in the first position and a –1 in the second position:

```
class a b;
model y = a b a*b;
contrast 'B at A2' b 1 -1  a*b 0  0  1 -1;
```

The elements associated with the interaction receive a 1 in the third position and a –1 in the fourth position. In order to specify coefficients correctly for the interaction term, you need to know how the levels of a and b vary in the interaction, which is governed by the order of the variables in the CLASS statement. The nonpositional syntax is designed to make it easier to enter coefficients for interactions and is necessary to enter coefficients for effects constructed with the experimental EFFECT statement. In square brackets you enter the coefficient followed by the associated levels of the CLASS variables. If B has two and A has three levels, the previous CONTRAST statement, by using nonpositional syntax for the interaction term, becomes

```
contrast 'B at A2' b 1 -1 a*b [1, 2 1] [-1, 2 2];
```

It assigns value 1 to the interaction where A is at level 2 and B is at level 1, and it assigns –1 to the interaction where both classification variables are at level 2. The comma

separating the entry for the **L** matrix from the level indicators is optional. Further details about the nonpositional contrast syntax and its use with constructed effects can be found in the section "Positional and Nonpositional Syntax for Contrast Coefficients" on page 3232. Nonpositional syntax is available only for fixed-effects coefficients.

The rows of $\mathbf{L}'$ are specified in order and are separated by commas. The rows of the $\mathbf{K}'$ component of $\mathbf{L}'$ are specified on the left side of the vertical bars (|). These rows test the fixed effects and are, therefore, checked for estimability. The rows of the $\mathbf{M}'$ component of $\mathbf{L}'$ are specified on the right side of the vertical bars. They test the random effects, and no estimability checking is necessary.

If PROC GLIMMIX finds the fixed-effects portion of the specified contrast to be nonestimable (see the SINGULAR= option), then it displays missing values for the test statistics.

If the elements of **L** are not specified for an effect that contains a specified effect, then the elements of the unspecified effect are automatically "filled in" over the levels of the higher-order effect. This feature is designed to preserve estimability for cases where there are complex higher-order effects. The coefficients for the higher-order effect are determined by equitably distributing the coefficients of the lower-level effect as in the construction of least squares means. In addition, if the intercept is specified, it is distributed over all classification effects that are not contained by any other specified effect. If an effect is not specified and does not contain any specified effects, then all of its coefficients in **L** are set to 0. You can override this behavior by specifying coefficients for the higher-order effect.

If too many values are specified for an effect, the extra ones are ignored; if too few are specified, the remaining ones are set to 0. If no random effects are specified, the vertical bar can be omitted; otherwise, it must be present. If a SUBJECT effect is used in the RANDOM statement, then the coefficients specified for the effects in the RANDOM statement are equitably distributed across the levels of the SUBJECT effect. You can use the E option to see exactly what **L** matrix is used.

PROC GLIMMIX handles missing level combinations of classification variables similarly to PROC GLM and PROC MIXED. These procedures delete fixed-effects parameters corresponding to missing levels in order to preserve estimability. However, PROC MIXED and PROC GLIMMIX do not delete missing level combinations for random-effects parameters, because linear combinations of the random-effects parameters are always estimable. These conventions can affect the way you specify your CONTRAST coefficients.

The CONTRAST statement computes the statistic

$$F = \frac{\begin{bmatrix} \widehat{\boldsymbol{\beta}} \\ \widehat{\boldsymbol{\gamma}} \end{bmatrix}' \mathbf{L}(\mathbf{L}'\mathbf{C}\mathbf{L})^{-1}\mathbf{L}' \begin{bmatrix} \widehat{\boldsymbol{\beta}} \\ \widehat{\boldsymbol{\gamma}} \end{bmatrix}}{r}$$

where $r = \mathrm{rank}(\mathbf{L}'\mathbf{C}\mathbf{L})$, and approximates its distribution with an F distribution unless DDFM=NONE. If you select DDFM=NONE as the degrees-of-freedom method in the MODEL statement, and if you do not assign degrees of freedom to the contrast with the DF= option, then PROC GLIMMIX computes the test statistic $r \times F$ and approximates its distribution with a chi-square distribution. In the expression for F, $\mathbf{C}$ is an estimate of $\mathrm{Var}[\widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\gamma}} - \boldsymbol{\gamma}]$; see the section "Estimated Precision of Estimates" on page 3189 and the section "Aspects Common to Adaptive Quadrature and Laplace Approximation" on page 3197 for details about the computation of $\mathbf{C}$ in a generalized linear mixed model.

The numerator degrees of freedom in the F approximation and the degrees of freedom in the chi-square approximation are equal to r. The denominator degrees of freedom are taken from the "Tests of Fixed Effects" table and correspond to the final effect you list in the CONTRAST statement. You can change the denominator degrees of freedom by using the DF= option.

You can specify the following *options* in the CONTRAST statement after a slash (/).

BYCATEGORY
BYCAT

requests that in models for nominal data (generalized logit models) the contrasts not be combined across response categories but reported separately for each category. For example, assume that the response variable Style is multinomial with three (unordered) categories. The following GLIMMIX statements fit a generalized logit model relating the preferred style of instruction to school and educational program effects:

```
proc glimmix data=school;
   class School Program;
   model Style(order=data) = School Program / s ddfm=none
                             dist=multinomial link=glogit;
   freq Count;
   contrast 'School 1 vs. 2' school 1 -1;
   contrast 'School 1 vs. 2' school 1 -1 / bycat;
run;
```

The first contrast compares school effects in all categories. This is a two-degrees-of-freedom contrast because there are two nonredundant categories. The second CONTRAST statement produces two single-degree-of-freedom contrasts, one for each nonreference Style category.

The BYCATEGORY option has no effect unless your model is a generalized (mixed) logit model.

CHISQ

requests that chi-square tests be performed for all contrasts in addition to any F tests. A chi-square statistic equals its corresponding F statistic times the numerator degrees of freedom, and these same degrees of freedom are used to compute the p-value for the chi-square test. This p-value will always be less than that for the F test, because it effectively corresponds to an F test with infinite denominator degrees of freedom.

DF=*number*

specifies the denominator degrees of freedom for the F test. For the degrees of freedom methods DDFM=BETWITHIN, DDFM=CONTAIN, and DDFM=RESIDUAL, the default is the denominator degrees of freedom taken from the "Tests of Fixed Effects" table and corresponds to the final effect you list in the CONTRAST statement. For DDFM=NONE, infinite denominator degrees of freedom are assumed by default, and for DDFM=SATTERTHWAITE and DDFM=KENWARDROGER, the denominator degrees of freedom are computed separately for each contrast.

E

requests that the **L** matrix coefficients for the contrast be displayed.

GROUP *coeffs*

sets up random-effect contrasts between different groups when a GROUP= variable appears in the RANDOM statement. By default, CONTRAST statement coefficients on random effects are distributed equally across groups. If you enter a multiple row contrast, you can also enter multiple rows for the GROUP coefficients. If the number of GROUP coefficients is less than the number of contrasts in the CONTRAST statement, the GLIMMIX procedure cycles through the GROUP coefficients. For example, the following two statements are equivalent:

```
contrast 'Trt 1 vs 2 @ x=0.4' trt 1 -1  0 | x 0.4,
                              trt 1  0 -1 | x 0.4,
                              trt 1 -1  0 | x 0.5,
                              trt 1  0 -1 | x 0.5 /
                 group 1 -1, 1 0 -1, 1 -1, 1 0 -1;

contrast 'Trt 1 vs 2 @ x=0.4' trt 1 -1  0 | x 0.4,
                              trt 1  0 -1 | x 0.4,
                              trt 1 -1  0 | x 0.5,
                              trt 1  0 -1 | x 0.5 /
                 group 1 -1, 1 0 -1;
```

SINGULAR=*number*

tunes the estimability checking. If $\mathbf{v}$ is a vector, define ABS($\mathbf{v}$) to be the largest absolute value of the elements of $\mathbf{v}$. If ABS($\mathbf{K}' - \mathbf{K}'\mathbf{T}$) is greater than c**number* for any row of $\mathbf{K}'$ in the contrast, then $\mathbf{K}'\boldsymbol{\beta}$ is declared nonestimable. Here, $\mathbf{T}$ is the Hermite form matrix $(\mathbf{X}'\mathbf{X})^{-}\mathbf{X}'\mathbf{X}$, and c is ABS($\mathbf{K}'$), except when it equals 0, and then c is 1. The value for *number* must be between 0 and 1; the default is 1E–4.

SUBJECT *coeffs*

sets up random-effect contrasts between different subjects when a SUBJECT= variable appears in the RANDOM statement. By default, CONTRAST statement coefficients on random effects are distributed equally across subjects. Listing subject coefficients for multiple row CONTRAST statements follows the same rules as for GROUP coefficients.

COVTEST Statement

COVTEST <'*label*'> <*test-specification*> </ *options*> **;**

The COVTEST statement provides a mechanism to obtain statistical inferences for the covariance parameters. Significance tests are based on the ratio of (residual) likelihoods or pseudo-likelihoods. Confidence limits and bounds are computed as Wald or likelihood ratio limits. You can specify multiple COVTEST statements.

The likelihood ratio test is obtained by fitting the model subject to the constraints imposed by the *test-specification*. The test statistic is formed as twice the difference of the (possibly restricted) log (pseudo-) likelihoods of the full and the reduced models. Note that fitting the null model does not necessarily require fewer computer resources compared to fitting the full model. The optimization settings for refitting the model are the same as for the full model and can be controlled with the NLOPTIONS statement.

Common questions in mixed modeling are whether variance components are zero, whether random effects are independent, and whether rows (columns) can be added or removed from an unstructured covariance matrix. When the parameters under the null hypothesis fall on the boundary of the parameter space, the distribution of the likelihood ratio statistic can be a complicated mixture of distributions. In certain situations it is known to be a relatively straightforward mixture of central chi-square distributions. When the GLIMMIX procedure recognizes the model and hypothesis as a case for which the mixture is readily available, the p-value of the likelihood ratio test is determined accordingly as a linear combination of central chi-square probabilities. The Note column in the "Likelihood Ratio Tests for Covariance Parameters" table along with the table's footnotes informs you about when mixture distributions are used in the calculation of p-values. You can find important statistical and computational details about likelihood ratio testing of covariance parameters with the GLIMMIX procedure in the section "Statistical Inference for Covariance Parameters" on page 3204.

In generalized linear mixed models that depend on pseudo-data, the GLIMMIX procedure fits the null model for a test of covariance parameters to the final pseudo-data of the converged optimization.

Table 44.6 summarizes the *options* available in the COVTEST statement.

Table 44.6 COVTEST Statement Options

Option	Description
Test specification	
TESTDATA=	Reads in covariance parameter values from a SAS data set
GENERAL	Provides a general facility to test linear combinations of covariance parameters
Covariance Test Options	
CL	Requests confidence limits for the covariance parameter estimates
CLASSICAL	Computes the likelihood ratio test *p*-value using the classical method
DF=	Specifies the degrees of freedom
ESTIMATES	Displays the estimates of the covariance parameters under the null hypothesis
MAXITER=	Limits the number of iterations
PARMS	Displays the values of the covariance parameters under the null hypothesis
RESTART	Specifies that starting values for the covariance parameters
TOLERANCE=	Sets the tolerance level of the parameter space boundary
WALD	Produces Wald *Z* tests
WGHT=	Supplies weights for the computation of *p*-values

Test Specification

The *test-specification* in the COVTEST statement draws on *keywords* that represent a particular null hypothesis, lists or data sets of parameter values, or general contrast specifications. Valid *keywords* are as follows:

GLM | INDEP tests the model against a null model of complete independence. All G-side covariance parameters are eliminated and the R-side covariance structure is reduced to a diagonal structure.

DIAGG tests for a diagonal **G** matrix by constraining off-diagonal elements in **G** to zero. The R-side structure is not modified.

DIAGR | CINDEP tests for conditional independence by reducing the R-side covariance structure to diagonal form. The G-side structure is not modified.

HOMOGENEITY tests homogeneity of covariance parameters across groups by imposing equality constraints. For example, the following statements fit a one-way model with heterogeneous variances and test whether the model could be reduced to a one-way analysis with the same variance across groups:

```
proc glimmix;
   class A;
   model y = a;
   random _residual_ / group=A;
   covtest 'common variance' homogeneity;
run;
```

See Example 44.9 for an application with groups and unstructured covariance matrices.

START | INITIAL compares the final estimates to the starting values of the covariance parameter estimates. This option is useful, for example, if you supply starting values in the PARMS statement and want to test whether the optimization produced significantly better values. In GLMMs based on pseudo-data, the likelihoods that use the starting and the final values are based on the final pseudo-data.

ZEROG tests whether the **G** matrix can be reduced to a zero matrix. This eliminates all G-side random effects from the model.

Only a single *keyword* is permitted in the COVTEST statement. To test more complicated hypotheses, you can formulate tests with the following specifications.

TESTDATA=*data-set*

TDATA=*data-set*

reads in covariance parameter values from a SAS data set. The data set should contain the numerical variable Estimate or numerical variables named Covp*i*. The GLIMMIX procedure associates the values for Covp*i* with the *i*th covariance parameter.

For data sets containing the numerical variable Estimate, the GLIMMIX procedure fixes the *i*th covariance parameter value at the value of the *i*th observation in the data set. A missing value indicates not to fix the particular parameter. PROC GLIMMIX performs one likelihood ratio test for the TESTDATA= data set.

For data sets containing numerical variables named Covp*i*, the procedure performs one likelihood ratio test for each observation in the TESTDATA= data set. You do not have to specify a Covp*i* variable for every covariance parameter. If the value for the variable is not missing, PROC GLIMMIX fixes the associated covariance parameter in the null model. Consider the following statements:

```
data TestDataSet;
   input covp1 covp2 covp3;
   datalines;
. 0 .
0 0 .
. 0 0
0 0 0
;

proc glimmix method=mspl;
   class subject x;
   model y = x age x*age;
   random intercept age / sub=subject type=un;
   covtest testdata=TestDataSet;
run;
```

Because the **G** matrix is a (2×2) unstructured matrix, the first observation of the TestDataSet corresponds to zeroing the covariance between the random intercept and the random slope. When the reduced model is fit, the variances of the intercept and slope are reestimated. The second observation reduces the model to one with only a random slope in age. The third reduces the model to a random intercept model. The last observation eliminates the **G** matrix altogether.

Note that the tests associated with the first and last set of covariance parameters in TestDataSet can also be obtained by using *keywords*:

```
proc glimmix;
   class subject x;
   model y = x age x*age;
   random intercept age / sub=subject type=un;
   covtest DiagG;
   covtest GLM;
run;
```

value-list

supplies a list of values at which to fix the covariance parameters. A missing value in the list indicates that the covariance parameter is not fixed. If the list is shorter than the number of covariance parameters, missing values are assumed for all parameters not specified. The COVTEST statements that test the random intercept and random slope in the previous example are as follows:

```
proc glimmix;
   class subject x;
   model y = x age x*age;
   random intercept age / sub=subject type=un;
   covtest 0 0;
   covtest . 0 0;
run;
```

GENERAL *coefficients* **<,***coefficients* **> <,... >**

CONTRAST *coefficients* **<,***coefficients* **> <,... >**

provides a general facility to test linear combinations of covariance parameters. You can specify one or more sets of coefficients. The position of a coefficient in the list corresponds to the position of the parameter in the "Covariance Parameter Estimates" table. The linear combination of covariance parameters that is implied by each set of coefficients is tested against zero. If the list of coefficients is shorter than the number of covariance parameters, a zero coefficient is assumed for the remaining parameters.

For example, in a heterogeneous variance model with four groups, the following statements test the simultaneous hypothesis $H: \sigma_1^2 = \sigma_2^2, \sigma_3^2 = \sigma_4^2$:

```
proc glimmix;
   class A;
   model y = a;
   random _residual_ / group=A;
   covtest 'pair-wise homogeneity'
           general  1 -1  0  0,
                    0  0  1 -1;
run;
```

In a repeated measures study with four observations per subject, the COVTEST statement in the following example tests whether the four correlation parameters are identical:

```
proc glimmix;
   class subject drug time;
   model y = drug time drug*time;
   random _residual_ / sub=subject type=unr;
   covtest 'Homogeneous correlation'
        general  0 0 0 0 1 -1            ,
                 0 0 0 0 1  0 -1         ,
                 0 0 0 0 1  0  0 -1      ,
                 0 0 0 0 1  0  0  0 -1   ,
                 0 0 0 0 1  0  0  0  0 -1;
run;
```

Notice that the variances (the first four covariance parameters) are allowed to vary. The null model for this test is thus a heterogeneous compound symmetry model.

The degrees of freedom associated with these general linear hypotheses are determined as the rank of the matrix $\mathbf{L}\mathbf{L}'$, where $\mathbf{L}$ is the $k \times q$ matrix of coefficients and q is the number of covariance parameters. Notice that the coefficients in a row do not have to sum to zero. The following statement tests $H: \theta_1 = 3\theta_2,\ \theta_3 = 0$:

```
covtest general 1 -3, 0 0 1;
```

Covariance Test Options

You can specify the following *options* in the COVTEST statement after a slash (/).

CL< (*suboptions*) >
requests confidence limits or bounds for the covariance parameter estimates. These limits are displayed as extra columns in the "Covariance Parameter Estimates" table.

The following *suboptions* determine the computation of confidence bounds and intervals. See the section "Statistical Inference for Covariance Parameters" on page 3204 for details about constructing likelihood ratio confidence limits for covariance parameters with PROC GLIMMIX.

ALPHA=*number*
determines the confidence level for constructing confidence limits for the covariance parameters. The value of *number* must be between 0 and 1, the default is 0.05, and the confidence level is 1 – *number*.

LOWERBOUND
LOWER
requests lower confidence bounds.

TYPE=*method*
determines how the GLIMMIX procedure constructs confidence limits for covariance parameters. The valid methods are PLR (or PROFILE), ELR (or ESTIMATED), and WALD. TYPE=PLR (TYPE=PROFILE) requests confidence bounds by inversion of the profile (restricted) likelihood

ratio (PLR). If θ is the parameter of interest, L denotes the likelihood (possibly restricted and possibly a pseudo-likelihood), and $\boldsymbol{\theta}_2$ is the vector of the remaining (nuisance) parameters, then the profile likelihood is defined as

$$L(\boldsymbol{\theta}_2|\widetilde{\theta}) = \sup_{\boldsymbol{\theta}_2} L(\widetilde{\theta}, \boldsymbol{\theta}_2)$$

for a given value $\widetilde{\theta}$ of θ. If $L(\widehat{\boldsymbol{\theta}})$ is the overall likelihood evaluated at the estimates $\widehat{\boldsymbol{\theta}}$, the $(1-\alpha)\times 100\%$ confidence region for θ satisfies the inequality

$$2\left\{L(\widehat{\boldsymbol{\theta}}) - L(\boldsymbol{\theta}_2|\widetilde{\theta})\right\} \leq \chi^2_{1,(1-\alpha)}$$

where $\chi^2_{1,(1-\alpha)}$ is the cutoff from a chi-square distribution with one degree of freedom and α probability to its right. If a residual scale parameter ϕ is profiled from the estimation, and θ is expressed in terms of a ratio with ϕ during estimation, then profile likelihood confidence limits are constructed for the ratio of the parameter with the residual variance. A column showing the ratio estimates is added to the "Covariance Parameter Estimates" table in this case. To obtain profile likelihood ratio limits for the parameters, rather than their ratios, and for the residual variance, use the NOPROFILE option in the PROC GLIMMIX statement. Also note that METHOD=LAPLACE or METHOD=QUAD implies the NOPROFILE option.

The TYPE=ELR (TYPE=ESTIMATED) option constructs bounds from the estimated likelihood (Pawitan 2001), where nuisance parameters are held fixed at the (restricted) maximum (pseudo-) likelihood estimates of the model. Estimated likelihood intervals are computationally less demanding than profile likelihood intervals, but they do not take into account the variability of the nuisance parameters or the dependence among the covariance parameters. See the section "Statistical Inference for Covariance Parameters" on page 3204 for a geometric interpretation and comparison of ELR versus PLR confidence bounds. A $(1-\alpha)\times 100\%$ confidence region based on the estimated likelihood is defined by the inequality

$$2\left\{L(\widehat{\boldsymbol{\theta}}) - L(\widetilde{\theta}, \widehat{\boldsymbol{\theta}}_2)\right\} \leq \chi^2_{1,(1-\alpha)}$$

where $L(\widetilde{\theta}, \widehat{\boldsymbol{\theta}}_2)$ is the likelihood evaluated at $\widetilde{\theta}$ and the component of $\widehat{\boldsymbol{\theta}}$ that corresponds to $\boldsymbol{\theta}_2$. Estimated likelihood ratio intervals tend to perform well when the correlations between the parameter of interest and the nuisance parameters is small. Their coverage probabilities can fall short of the nominal coverage otherwise. You can display the correlation matrix of the covariance parameter estimates with the ASYCORR option in the PROC GLIMMIX statement.

If you choose TYPE=PLR or TYPE=ELR, the GLIMMIX procedure reports the right-tail probability of the associated single-degree-of-freedom likelihood ratio test along with the confidence bounds. This helps you diagnose whether solutions to the inequality could be found. If the reported probability exceeds α, the associated bound does not meet the inequality. This might occur, for example, when the parameter space is bounded and the likelihood at the boundary values has not dropped by a sufficient amount to satisfy the test inequality.

The TYPE=WALD method requests confidence limits based on the Wald-type statistic $Z_\theta = \widehat{\theta}/\mathrm{ease}(\widehat{\theta})$, where ease is the estimated asymptotic standard error of the covariance parameter. For parameters that have a lower boundary constraint of zero, a Satterthwaite approximation is used to construct limits of the form

$$\frac{\nu\widehat{\theta}}{\chi^2_{\nu,1-\alpha/2}} \leq \theta \leq \frac{\nu\widehat{\theta}}{\chi^2_{\nu,\alpha/2}}$$

where $\nu = 2Z^2$, and the denominators are quantiles of the χ^2 distribution with ν degrees of freedom. See Milliken and Johnson (1992) and Burdick and Graybill (1992) for similar techniques. For all other parameters, Wald Z-scores and normal quantiles are used to construct the limits. Such limits are also provided for variance components if you specify the NOBOUND option in the PROC GLIMMIX statement or the PARMS statement.

UPPERBOUND
UPPER
requests upper confidence bounds.

If you do not specify any *suboptions*, the default is to compute two-sided Wald confidence intervals with confidence level $1 - \alpha = 0.95$.

CLASSICAL
requests that the p-value of the likelihood ratio test be computed by the classical method. If $\widehat{\lambda}$ is the realized value of the test statistic in the likelihood ratio test,

$$p = \Pr\left(\chi^2_\nu \geq \widehat{\lambda}\right)$$

where ν is the degrees of freedom of the hypothesis.

DF=*value-list*
enables you to supply degrees of freedom $\nu_1, \cdots, \nu_k$ for the computation of p-values from chi-square mixtures. The mixture weights $w_1, \cdots, w_k$ are supplied with the WGHT= option. If no weights are specified, an equal weight distribution is assumed. If $\widehat{\lambda}$ is the realized value of the test statistic in the likelihood ratio test, PROC GLIMMIX computes the p-value as (Shapiro 1988)

$$p = \sum_{i=1}^{k} w_i \Pr\left(\chi^2_{\nu_i} \geq \widehat{\lambda}\right)$$

Note that $\chi^2_0 \equiv 0$ and that mixture weights are scaled to sum to one. If you specify more weights than degrees of freedom in *value-list*, the rank of the hypothesis (DF column) is substituted for the missing degrees of freedom.

Specifying a single value ν for *value-list* without giving mixture weights is equivalent to computing the p-value as

$$p = \Pr\left(\chi^2_\nu \geq \widehat{\lambda}\right)$$

For example, the following statements compute the p-value based on a chi-square distribution with one degree of freedom:

```
proc glimmix noprofile;
   class A sub;
   model score = A;
   random _residual_ / type=ar(1) subject=sub;
   covtest 'ELR low' 30.62555 0.7133361 / df=1;
run;
```

The DF column of the COVTEST output will continue to read 2 regardless of the DF= specification, however, because the DF column reflects the rank of the hypothesis and equals the number of constraints imposed on the full model.

ESTIMATES

EST

displays the estimates of the covariance parameters under the null hypothesis. Specifying the ESTIMATES option in one COVTEST statement has the same effect as specifying the option in every COVTEST statement.

MAXITER=*number*

limits the number of iterations when you are refitting the model under the null hypothesis to *number* iterations. If the null model does not converge before the limit is reached, no *p*-values are produced.

PARMS

displays the values of the covariance parameters under the null hypothesis. This option is useful if you supply multiple sets of parameter values with the TESTDATA= option. Specifying the PARMS option in one COVTEST statement has the same effect as specifying the option in every COVTEST statement.

RESTART

specifies that starting values for the covariance parameters for the null model are obtained by the same mechanism as starting values for the full models. For example, if you do not specify a PARMS statement, the RESTART option computes MIVQUE(0) estimates under the null model (Goodnight 1978a). If you provide starting values with the PARMS statement, the starting values for the null model are obtained by applying restrictions to the starting values for the full model.

By default, PROC GLIMMIX obtains starting values by applying null model restrictions to the converged estimates of the full model. Although this is computationally expedient, the method does not always lead to good starting values for the null model, depending on the nature of the model and hypothesis. In particular, when you receive a warning about parameters not specified under H_0 falling on the boundary, the RESTART option can be useful.

TOLERANCE=*r*

Values within tolerance $r \geq 0$ of the boundary of the parameter space are considered on the boundary when PROC GLIMMIX examines estimates of nuisance parameters under H_0 and determines whether mixture weights and degrees of freedom can be obtained. In certain cases, when parameters not specified under the null hypothesis are on boundaries, the asymptotic distribution of the likelihood ratio statistic is not a mixture of chi-squares (see, for example, case 8 in Self and Liang 1987). The default for *r* is 1E4 times the machine epsilon; this product is approximately 1E–12 on most computers.

WALD

produces Wald *Z* tests for the covariance parameters based on the estimates and asymptotic standard errors in the "Covariance Parameter Estimates" table.

WGHT=*value-list*

enables you to supply weights for the computation of *p*-values from chi-square mixtures. See the DF= option for details. Mixture weights are scaled to sum to one.

EFFECT Statement

EFFECT *effect-specification* **;**

The experimental EFFECT statement enables you to construct special collections of columns for **X** or **Z** matrices in your model. These collections are referred to as *constructed effects* to distinguish them from the usual model effects formed from continuous or classification variables.

For details about the syntax of the EFFECT statement and how columns of constructed effects are computed, see the section "EFFECT Statement" on page 397 in Chapter 19, "Shared Concepts and Topics." For specific details concerning the use of the EFFECT statement with the GLIMMIX procedure, see the section "Notes on the EFFECT Statement" on page 3231.

ESTIMATE Statement

ESTIMATE '*label*' *contrast-specification* < **(***divisor***=***n***)** >
< **,** '*label*' *contrast-specification* < **(***divisor***=***n***)** > > < **,** . . . >
< / *options* > **;**

The ESTIMATE statement provides a mechanism for obtaining custom hypothesis tests. As in the CONTRAST statement, the basic element of the ESTIMATE statement is the *contrast-specification*, which consists of MODEL and G-side random effects and their coefficients. Specifically, a *contrast-specification* takes the form

< *fixed-effect values* . . . > < | *random-effect values* . . . >

Based on the *contrast-specifications* in your ESTIMATE statement, PROC GLIMMIX constructs the matrix $\mathbf{L}' = [\mathbf{K}' \; \mathbf{M}']$, as in the CONTRAST statement, where $\mathbf{K}$ is associated with the fixed effects and $\mathbf{M}$ is associated with the G-side random effects. The GLIMMIX procedure supports nonpositional syntax for the coefficients of fixed effects in the ESTIMATE statement. For details see the section "Positional and Nonpositional Syntax for Contrast Coefficients" on page 3232.

PROC GLIMMIX then produces for each row $\mathbf{l}$ of $\mathbf{L}'$ an approximate t test of the hypothesis $H\colon \mathbf{l}\boldsymbol{\phi} = 0$, where $\boldsymbol{\phi} = [\boldsymbol{\beta}' \; \boldsymbol{\gamma}']'$. You can also obtain multiplicity-adjusted p-values and confidence limits for multirow estimates with the ADJUST= option. The output from multiple ESTIMATE statements is organized as follows. Results from unadjusted estimates are reported first in a single table, followed by separate tables for each of the adjusted estimates. Results from all ESTIMATE statements are combined in the "Estimates" ODS table.

Note that multirow estimates are permitted. Unlike the CONTRAST statement, you need to specify a '*label*' for every row of the multirow estimate, because PROC GLIMMIX produces one test per row. PROC GLIMMIX selects the degrees of freedom to match those displayed in the "Type III Tests of Fixed Effects" table for the final effect you list in the ESTIMATE statement. You can modify the degrees of freedom by using the DF= option. If you select DDFM=NONE and do not modify the degrees of freedom by using the DF= option, PROC GLIMMIX uses infinite degrees of freedom, essentially computing approximate z tests. If PROC GLIMMIX finds the fixed-effects portion of the specified estimate to be nonestimable, then it displays "Non-est" for the estimate entry.

Table 44.7 summarizes the *options* available in the ESTIMATE statement.

Table 44.7 ESTIMATE Statement Options

Option	Description
Construction and Computation of Estimable Functions	
DIVISOR=	Specifies a list of values to divide the coefficients
GROUP	Sets up random-effect contrasts between different groups
SINGULAR=	Tunes the estimability checking difference
SUBJECT	Sets up random-effect contrasts between different subjects
Degrees of Freedom and *p*-values	
ADJDFE=	Determines denominator degrees of freedom when *p*-values and confidence limits are adjusted for multiple comparisons
ADJUST=	Determines the method for multiple comparison adjustment of estimates
ALPHA=α	Determines the confidence level ($1-\alpha$)
DF=	Assigns a specific value to degrees of freedom for tests and confidence limits
LOWER	Performs one-sided, lower-tailed inference
STEPDOWN	Adjusts multiplicity-corrected *p*-values further in a step-down fashion
UPPER	Performs one-sided, upper-tailed inference
Statistical Output	
CL	Constructs *t*-type confidence limits
E	Prints the **L** matrix
Generalized Linear Modeling	
BYCATEGORY=	Reports estimates separately for each category for models with nominal data
EXP	Displays exponentiated estimates
ILINK	Computes and displays estimates and standard errors on the inverse linked scale

ADJDFE=SOURCE | ROW

specifies how denominator degrees of freedom are determined when *p*-values and confidence limits are adjusted for multiple comparisons with the ADJUST= option. When you do not specify the ADJDFE= option, or when you specify ADJDFE=SOURCE, the denominator degrees of freedom for multiplicity-adjusted results are the denominator degrees of freedom for the final effect listed in the ESTIMATE statement from the "Type III Tests of Fixed Effects" table.

The ADJDFE=ROW setting is useful if you want multiplicity adjustments to take into account that denominator degrees of freedom are not constant across estimates. This can be the case, for example, when the DDFM=SATTERTHWAITE or DDFM=KENWARDROGER degrees-of-freedom method is in effect.

ADJUST=BON | SCHEFFE | SIDAK | SIMULATE< (*simoptions*) > | T

requests a multiple comparison adjustment for the *p*-values and confidence limits for the estimates. The adjusted quantities are produced in addition to the unadjusted quantities. Adjusted confidence limits are produced if the CL or ALPHA= option is in effect. For a description of the adjustments, see Chapter 45,

"The GLM Procedure," and Chapter 67, "The MULTTEST Procedure," and the documentation for the ADJUST= option in the LSMEANS statement. The ADJUST= option is ignored for generalized logit models.

If the STEPDOWN option is in effect, the *p*-values are further adjusted in a step-down fashion.

ALPHA=*number*

requests that a *t*-type confidence interval be constructed with confidence level 1 – *number*. The value of *number* must be between 0 and 1; the default is 0.05. If DDFM=NONE and you do not specify degrees of freedom with the DF= option, PROC GLIMMIX uses infinite degrees of freedom, essentially computing a *z* interval.

BYCATEGORY

BYCAT

requests that in models for nominal data (generalized logit models) estimates be reported separately for each category. In contrast to the BYCATEGORY option in the CONTRAST statement, an ESTIMATE statement in a generalized logit model does not distribute coefficients by response category, because ESTIMATE statements always correspond to single rows of the **L** matrix.

For example, assume that the response variable Style is multinomial with three (unordered) categories. The following GLIMMIX statements fit a generalized logit model relating the preferred style of instruction to school and educational program effects:

```
proc glimmix data=school;
   class School Program;
   model Style(order=data) = School Program / s ddfm=none
                             dist=multinomial link=glogit;
   freq Count;
   estimate 'School 1 vs. 2' school 1 -1 / bycat;
   estimate 'School 1 vs. 2' school 1 -1;
run;
```

The first ESTIMATE statement compares school effects separately for each nonredundant category. The second ESTIMATE statement compares the school effects for the first non-reference category.

The BYCATEGORY option has no effect unless your model is a generalized (mixed) logit model.

CL

requests that *t*-type confidence limits be constructed. If DDFM=NONE and you do not specify degrees of freedom with the DF= option, PROC GLIMMIX uses infinite degrees of freedom, essentially computing a *z* interval. The confidence level is 0.95 by default. These intervals are adjusted for multiplicity when you specify the ADJUST= option.

DF=*number*

specifies the degrees of freedom for the *t* test and confidence limits. The default is the denominator degrees of freedom taken from the "Type III Tests of Fixed Effects" table and corresponds to the final effect you list in the ESTIMATE statement.

DIVISOR=*value-list*

specifies a list of values by which to divide the coefficients so that fractional coefficients can be entered as integer numerators. If you do not specify *value-list*, a default value of 1.0 is assumed. Missing values in the *value-list* are converted to 1.0.

If the number of elements in *value-list* exceeds the number of rows of the estimate, the extra values are ignored. If the number of elements in *value-list* is less than the number of rows of the estimate, the last value in *value-list* is copied forward.

If you specify a row-specific divisor as part of the specification of the estimate row, this value multiplies the corresponding divisor implied by the *value-list*. For example, the following statement divides the coefficients in the first row by 8, and the coefficients in the third and fourth row by 3:

```
estimate 'One vs. two'   A 2 -2  (divisor=2),
         'One vs. three' A 1  0 -1         ,
         'One vs. four'  A 3  0  0 -3      ,
         'One vs. five'  A 1  0  0  0  -1  / divisor=4,.,3;
```

Coefficients in the second row are not altered.

E

requests that the **L** matrix coefficients be displayed.

EXP

requests exponentiation of the estimate. When you model data with the logit, cumulative logit, or generalized logit link functions, and the estimate represents a log odds ratio or log cumulative odds ratio, the EXP option produces an odds ratio. See "Odds and Odds Ratio Estimation" on page 3225 for important details about the computation and interpretation of odds and odds ratio results with the GLIMMIX procedure. If you specify the CL or ALPHA= option, the (adjusted) confidence bounds are also exponentiated.

GROUP *coeffs*

sets up random-effect contrasts between different groups when a GROUP= variable appears in the RANDOM statement. By default, ESTIMATE statement coefficients on random effects are distributed equally across groups. If you enter a multirow estimate, you can also enter multiple rows for the GROUP coefficients. If the number of GROUP coefficients is less than the number of contrasts in the ESTIMATE statement, the GLIMMIX procedure cycles through the GROUP coefficients. For example, the following two statements are equivalent:

```
estimate 'Trt 1 vs 2 @ x=0.4' trt 1 -1  0 | x 0.4,
         'Trt 1 vs 3 @ x=0.4' trt 1  0 -1 | x 0.4,
         'Trt 1 vs 2 @ x=0.5' trt 1 -1  0 | x 0.5,
         'Trt 1 vs 3 @ x=0.5' trt 1  0 -1 | x 0.5 /
                group 1 -1, 1 0 -1, 1 -1, 1 0 -1;

estimate 'Trt 1 vs 2 @ x=0.4' trt 1 -1  0 | x 0.4,
         'Trt 1 vs 3 @ x=0.4' trt 1  0 -1 | x 0.4,
         'Trt 1 vs 2 @ x=0.5' trt 1 -1  0 | x 0.5,
         'Trt 1 vs 3 @ x=0.5' trt 1  0 -1 | x 0.5 /
                group 1 -1, 1 0 -1;
```

ILINK

requests that the estimate and its standard error are also reported on the scale of the mean (the inverse linked scale). PROC GLIMMIX computes the value on the mean scale by applying the inverse link to the estimate. The interpretation of this quantity depends on the *fixed-effect values* and *random-effect values* specified in your ESTIMATE statement and on the link function. In a model for binary data with logit link, for example, the following statements compute

$$\frac{1}{1+\exp\{-(\alpha_1-\alpha_2)\}}$$

where α_1 and α_2 are the fixed-effects solutions associated with the first two levels of the classification effect A:

```
proc glimmix;
   class A;
   model y = A / dist=binary link=logit;
   estimate 'A one vs. two' A 1 -1 / ilink;
run;
```

This quantity is not the difference of the probabilities associated with the two levels,

$$\pi_1-\pi_2 = \frac{1}{1+\exp\{-\beta_0-\alpha_1\}} - \frac{1}{1+\exp\{-\beta_0-\alpha_2\}}$$

The standard error of the inversely linked estimate is based on the delta method. If you also specify the CL option, the GLIMMIX procedure computes confidence limits for the estimate on the mean scale. In multinomial models for nominal data, the limits are obtained by the delta method. In other models they are obtained from the inverse link transformation of the confidence limits for the estimate. The ILINK option is specific to an ESTIMATE statement.

LOWER

LOWERTAILED

requests that the *p*-value for the *t* test be based only on values less than the test statistic. A two-tailed test is the default. A lower-tailed confidence limit is also produced if you specify the CL or ALPHA= option.

Note that for ADJUST=SCHEFFE the one-sided adjusted confidence intervals and one-sided adjusted *p*-values are the same as the corresponding two-sided statistics, because this adjustment is based on only the right tail of the *F* distribution.

SINGULAR=*number*

tunes the estimability checking as documented for the CONTRAST statement.

STEPDOWN<(*step-down-options***)>**

requests that multiplicity adjustments for the *p*-values of estimates be further adjusted in a step-down fashion. Step-down methods increase the power of multiple testing procedures by taking advantage of the fact that a *p*-value will never be declared significant unless all smaller *p*-values are also declared significant. Note that the STEPDOWN adjustment combined with ADJUST=BON corresponds to the methods of Holm (1979) and "Method 2" of Shaffer (1986); this is the default. Using step-down-adjusted *p*-values combined with ADJUST=SIMULATE corresponds to the method of Westfall (1997).

If the degrees-of-freedom method is DDFM=KENWARDROGER or DDFM=SATTERTHWAITE, then step-down-adjusted *p*-values are produced only if the ADJDFE=ROW option is in effect.

Also, the STEPDOWN option affects only *p*-values, not confidence limits. For ADJUST=SIMULATE, the generalized least squares hybrid approach of Westfall (1997) is employed to increase Monte Carlo accuracy.

You can specify the following *step-down-options* in parentheses after the STEPDOWN option.

MAXTIME=*n*
: specifies the time (in seconds) to spend computing the maximal logically consistent sequential subsets of equality hypotheses for TYPE=LOGICAL. The default is MAXTIME=60. If the MAXTIME value is exceeded, the adjusted tests are not computed. When this occurs, you can try increasing the MAXTIME value. However, note that there are common multiple comparisons problems for which this computation requires a huge amount of time—for example, all pairwise comparisons between more than 10 groups. In such cases, try to use TYPE=FREE (the default) or TYPE=LOGICAL(*n*) for small *n*.

ORDER=PVALUE | ROWS
: specifies the order in which the step-down tests are performed. ORDER=PVALUE is the default, with estimates being declared significant only if all estimates with smaller (unadjusted) *p*-values are significant. If you specify ORDER=ROWS, then significances are evaluated in the order in which they are specified in the syntax.

REPORT
: specifies that a report on the step-down adjustment be displayed, including a listing of the sequential subsets (Westfall 1997) and, for ADJUST=SIMULATE, the step-down simulation results.

TYPE=LOGICAL< (*n***) > | FREE**
: If you specify TYPE=LOGICAL, the step-down adjustments are computed by using maximal logically consistent sequential subsets of equality hypotheses (Shaffer 1986; Westfall 1997). Alternatively, for TYPE=FREE, sequential subsets are computed ignoring logical constraints. The TYPE=FREE results are more conservative than those for TYPE=LOGICAL, but they can be much more efficient to produce for many estimates. For example, it is not feasible to take logical constraints between all pairwise comparisons of more than about 10 groups. For this reason, TYPE=FREE is the default.

  However, you can reduce the computational complexity of taking logical constraints into account by limiting the depth of the search tree used to compute them, specifying the optional depth parameter as a number *n* in parentheses after TYPE=LOGICAL. As with TYPE=FREE, results for TYPE=LOGICAL(*n*) are conservative relative to the true TYPE=LOGICAL results, but even for TYPE=LOGICAL(0) they can be appreciably less conservative than TYPE=FREE and they are computationally feasible for much larger numbers of estimates. If you do not specify *n* or if *n* = –1, the full search tree is used.

SUBJECT *coeffs*
: sets up random-effect contrasts between different subjects when a SUBJECT= variable appears in the RANDOM statement. By default, ESTIMATE statement coefficients on random effects are distributed equally across subjects. Listing subject coefficients for an ESTIMATE statement with multiple rows follows the same rules as for GROUP coefficients.

UPPER

UPPERTAILED

requests that the p-value for the t test be based only on values greater than the test statistic. A two-tailed test is the default. An upper-tailed confidence limit is also produced if you specify the CL or ALPHA= option.

Note that for ADJUST=SCHEFFE the one-sided adjusted confidence intervals and one-sided adjusted p-values are the same as the corresponding two-sided statistics, because this adjustment is based on only the right tail of the F distribution.

FREQ Statement

FREQ *variable* **;**

The *variable* in the FREQ statement identifies a numeric variable in the data set or one computed through PROC GLIMMIX programming statements that contains the frequency of occurrence for each observation. PROC GLIMMIX treats each observation as if it appears f times, where f is the value of the FREQ variable for the observation. If it is not an integer, the frequency value is truncated to an integer. If the frequency value is less than 1 or missing, the observation is not used in the analysis. When the FREQ statement is not specified, each observation is assigned a frequency of 1.

The analysis produced by using a FREQ statement reflects the expanded number of observations. For an example of a FREQ statement in a model with random effects, see Example 44.11 in this chapter.

ID Statement

ID *variables* **;**

The ID statement specifies which quantities to include in the OUT= data set from the OUTPUT statement in addition to any statistics requested in the OUTPUT statement. If no ID statement is given, the GLIMMIX procedure includes all variables from the input data set in the OUT= data set. Otherwise, only the variables listed in the ID statement are included. Automatic variables such as _LINP_, _MU_, _VARIANCE_, etc. are not transferred to the OUT= data set unless they are listed in the ID statement.

The ID statement can be used to transfer computed quantities that depend on the model to an output data set. In the following example, two sets of Hessian weights are computed in a gamma regression with a noncanonical link. The covariance matrix for the fixed effects can be constructed as the inverse of $\mathbf{X}'\mathbf{W}\mathbf{X}$. $\mathbf{W}$ is a diagonal matrix of the w_{ei} or w_{oi}, depending on whether the expected or observed Hessian matrix is desired, respectively.

```
proc glimmix;
   class group age;
   model cost = group age / s error=gamma link=pow(0.5);
   output out=gmxout pred=pred;
   id _variance_ wei woi;
   vpmu  = 2*_mu_;
   if (_mu_ > 1.0e-8) then do;
      gpmu  =  0.5  * (_mu_**(-0.5));
```

```
        gppmu = -0.25 * (_mu_**(-1.5));
        wei   = 1/(_phi_*_variance_*gpmu*gpmu);
        woi   = wei + (cost-_mu_) *
                (_variance_*gppmu + vpmu*gpmu) /
                (_variance_*_variance_*gpmu*gpmu*gpmu*_phi_);
    end;
run;
```

The variables _VARIANCE_ and _MU_ and other symbols are predefined by PROC GLIMMIX and can be used in programming statements. For rules and restrictions, see the section "Programming Statements" on page 3175.

LSMEANS Statement

LSMEANS *fixed-effects* < / *options* > **;**

The LSMEANS statement computes least squares means (LS-means) of fixed effects. As in the GLM and the MIXED procedures, LS-means are *predicted population margins*—that is, they estimate the marginal means over a balanced population. In a sense, LS-means are to unbalanced designs as class and subclass arithmetic means are to balanced designs. The **L** matrix constructed to compute them is the same as the **L** matrix formed in PROC GLM; however, the standard errors are adjusted for the covariance parameters in the model. Least squares means computations are not supported for multinomial models.

Each LS-mean is computed as $\mathbf{L}\widehat{\boldsymbol{\beta}}$, where $\mathbf{L}$ is the coefficient matrix associated with the least squares mean and $\widehat{\boldsymbol{\beta}}$ is the estimate of the fixed-effects parameter vector. The approximate standard error for the LS-mean is computed as the square root of $\mathbf{L}\widehat{\mathrm{Var}}[\widehat{\boldsymbol{\beta}}]\mathbf{L}'$. The approximate variance matrix of the fixed-effects estimates depends on the estimation method.

LS-means are constructed on the linked scale—that is, the scale on which the model effects are additive. For example, in a binomial model with logit link, the least squares means are predicted population margins of the logits.

LS-means can be computed for any effect in the MODEL statement that involves only CLASS variables. You can specify multiple effects in one LSMEANS statement or in multiple LSMEANS statements, and all LSMEANS statements must appear after the MODEL statement. As in the ESTIMATE statement, the **L** matrix is tested for estimability, and if this test fails, PROC GLIMMIX displays "Non-est" for the LS-means entries.

Assuming the LS-mean is estimable, PROC GLIMMIX constructs an approximate t test to test the null hypothesis that the associated population quantity equals zero. By default, the denominator degrees of freedom for this test are the same as those displayed for the effect in the "Type III Tests of Fixed Effects" table. If the DDFM=SATTERTHWAITE or DDFM=KENWARDROGER option is specified in the MODEL statement, PROC GLIMMIX determines degrees of freedom separately for each test, unless the DDF= option overrides it for a particular effect. See the DDFM= option for more information. Table 44.8 summarizes options available in the LSMEANS statement. All LSMEANS *options* are subsequently discussed in alphabetical order.

Table 44.8 LSMEANS Statement Options

Option	Description
Construction and Computation of LS-Means	
AT	Modifies covariate value in computing LS-means
BYLEVEL	Computes separate margins
DIFF	Requests differences of LS-means
OM	Specifies weighting scheme for LS-mean computation as determined by the input data set
SINGULAR=	Tunes estimability checking
SLICE=	Partitions F tests (simple effects)
SLICEDIFF=	Requests simple effects differences
SLICEDIFFTYPE	Determines the type of simple difference
Degrees of Freedom and *P*-values	
ADJDFE=	Determines whether to compute row-wise denominator degrees of freedom with DDFM=SATTERTHWAITE or DDFM=KENWARDROGER
ADJUST=	Determines the method for multiple comparison adjustment of LS-mean differences
ALPHA=α	Determines the confidence level $(1-\alpha)$
DF=	Assigns specific value to degrees of freedom for tests and confidence limits
STEPDOWN	Adjusts multiple comparison p-values further in a step-down fashion
Statistical Output	
CL	Constructs confidence limits for means and or mean differences
CORR	Displays correlation matrix of LS-means
COV	Displays covariance matrix of LS-means
E	Prints the **L** matrix
ILINK	Applies the inverse link transform to the LS-Means (not differences) and produces the standard errors on the inverse linked scale
LINES	Produces "Lines" display for pairwise LS-mean differences
ODDS	Reports odds of levels of fixed effects if permissible by the link function
ODDSRATIO	Reports (simple) differences of least squares means in terms of odds ratios if permissible by the link function
PLOTS=	Requests ODS statistical graphics of means and mean comparisons

You can specify the following *options* in the LSMEANS statement after a slash (/).

ADJDFE=ROW | SOURCE

specifies how denominator degrees of freedom are determined when p-values and confidence limits are adjusted for multiple comparisons with the ADJUST= option. When you do not specify the ADJDFE= option, or when you specify ADJDFE=SOURCE, the denominator degrees of freedom for multiplicity-adjusted results are the denominator degrees of freedom for the LS-mean effect in the "Type III Tests of Fixed Effects" table. When you specify ADJDFE=ROW, the denominator degrees of freedom for multiplicity-adjusted results correspond to the degrees of freedom displayed in the DF column of the "Differences of Least Squares Means" table.

The ADJDFE=ROW setting is particularly useful if you want multiplicity adjustments to take into account that denominator degrees of freedom are not constant across LS-mean differences. This can be the case, for example, when the DDFM=SATTERTHWAITE or DDFM=KENWARDROGER degrees-of-freedom method is in effect.

In one-way models with heterogeneous variance, combining certain ADJUST= options with the ADJDFE=ROW option corresponds to particular methods of performing multiplicity adjustments in the presence of heteroscedasticity. For example, the following statements fit a heteroscedastic one-way model and perform Dunnett's T3 method (Dunnett 1980), which is based on the studentized maximum modulus (ADJUST=SMM):

```
proc glimmix;
   class A;
   model y = A / ddfm=satterth;
   random _residual_ / group=A;
   lsmeans A / adjust=smm adjdfe=row;
run;
```

If you combine the ADJDFE=ROW option with ADJUST=SIDAK, the multiplicity adjustment corresponds to the T2 method of Tamhane (1979), while ADJUST=TUKEY corresponds to the method of Games-Howell (Games and Howell 1976). Note that ADJUST=TUKEY gives the exact results for the case of fractional degrees of freedom in the one-way model, but it does not take into account that the degrees of freedom are subject to variability. A more conservative method, such as ADJUST=SMM, might protect the overall error rate better.

Unless the ADJUST= option is specified in the LSMEANS statement, the ADJDFE= option has no effect.

ADJUST=BON

ADJUST=DUNNETT

ADJUST=NELSON

ADJUST=SCHEFFE

ADJUST=SIDAK

ADJUST=SIMULATE<(*simoptions***)>**

ADJUST=SMM | GT2

ADJUST=TUKEY

requests a multiple comparison adjustment for the *p*-values and confidence limits for the differences of LS-means. The adjusted quantities are produced in addition to the unadjusted quantities. By default, PROC GLIMMIX performs all pairwise differences. If you specify ADJUST=DUNNETT, the procedure analyzes all differences with a control level. If you specify ADJUST=NELSON, ANOM differences are taken. The ADJUST= option implies the DIFF option, unless the SLICEDIFF= option is specified.

The BON (Bonferroni) and SIDAK adjustments involve correction factors described in Chapter 45, "The GLM Procedure," and Chapter 67, "The MULTTEST Procedure;" also see Westfall and Young (1993) and Westfall et al. (1999). When you specify ADJUST=TUKEY and your data are unbalanced, PROC GLIMMIX uses the approximation described in Kramer (1956) and identifies the adjustment as "Tukey-Kramer" in the results. Similarly, when you specify ADJUST=DUNNETT

or ADJUST=NELSON and the LS-means are correlated, the GLIMMIX procedure uses the factor-analytic covariance approximation described in Hsu (1992) and identifies the adjustment in the results as "Dunnett-Hsu" or "Nelson-Hsu," respectively. The approximation derives an approximate "effective sample sizes" for which exact critical values are computed. Note that computing the exact adjusted p-values and critical values for unbalanced designs can be computationally intensive, in particular for ADJUST=NELSON. A simulation-based approach, as specified by the ADJUST=SIM option, while nondeterministic, can provide inferences that are sufficiently accurate in much less time. The preceding references also describe the SCHEFFE and SMM adjustments.

Nelson's adjustment applies only to the analysis of means (Ott 1967; Nelson 1982, 1991, 1993), where LS-means are compared against an average LS-mean. It does not apply to all pairwise differences of least squares means, or to slice differences that you specify with the SLICEDIFF= option. See the DIFF=ANOM option for more details regarding the analysis of means with the GLIMMIX procedure.

The SIMULATE adjustment computes adjusted p-values and confidence limits from the simulated distribution of the maximum or maximum absolute value of a multivariate t random vector. All covariance parameters, except the residual scale parameter, are fixed at their estimated values throughout the simulation, potentially resulting in some underdispersion. The simulation estimates q, the true $(1-\alpha)$ quantile, where $1-\alpha$ is the confidence coefficient. The default α is 0.05, and you can change this value with the ALPHA= option in the LSMEANS statement.

The number of samples is set so that the tail area for the simulated q is within γ of $1-\alpha$ with $100(1-\epsilon)\%$ confidence. In equation form,

$$\Pr(|F(\widehat{q}) - (1-\alpha)| \leq \gamma) = 1 - \epsilon$$

where $\hat{q}$ is the simulated q and F is the true distribution function of the maximum; see Edwards and Berry (1987) for details. By default, $\gamma = 0.005$ and $\epsilon = 0.01$, placing the tail area of $\hat{q}$ within 0.005 of 0.95 with 99% confidence. The ACC= and EPS= *simoptions* reset γ and ϵ, respectively, the NSAMP= *simoption* sets the sample size directly, and the SEED= *simoption* specifies an integer used to start the pseudo-random number generator for the simulation. If you do not specify a seed, or if you specify a value less than or equal to zero, the seed is generated from reading the time of day from the computer clock. For additional descriptions of these and other simulation options, see the section "LSMEANS Statement" on page 3419 in Chapter 45, "The GLM Procedure."

If the STEPDOWN option is in effect, the p-values are further adjusted in a step-down fashion. For certain options and data, this adjustment is exact under an iid $N(0, \sigma^2)$ model for the dependent variable, in particular for the following:

- for ADJUST=DUNNETT when the means are uncorrelated
- for ADJUST=TUKEY with STEPDOWN(TYPE=LOGICAL) when the means are balanced and uncorrelated.

The first case is a consequence of the nature of the successive step-down hypotheses for comparisons with a control; the second employs an extension of the maximum studentized range distribution appropriate for partition hypotheses (Royen 1989). Finally, for STEPDOWN(TYPE=FREE), ADJUST=TUKEY employs the Royen (1989) extension in such a way that the resulting p-values are conservative.

ALPHA=*number*

requests that a *t*-type confidence interval be constructed for each of the LS-means with confidence level 1 – *number*. The value of *number* must be between 0 and 1; the default is 0.05.

AT *variable*=*value*

AT (*variable-list***)=(***value-list***)**

AT MEANS

enables you to modify the values of the covariates used in computing LS-means. By default, all covariate effects are set equal to their mean values for computation of standard LS-means. The AT option enables you to assign arbitrary values to the covariates. Additional columns in the output table indicate the values of the covariates.

If there is an effect containing two or more covariates, the AT option sets the effect equal to the product of the individual means rather than the mean of the product (as with standard LS-means calculations). The AT MEANS option sets covariates equal to their mean values (as with standard LS-means) and incorporates this adjustment to crossproducts of covariates.

As an example, consider the following invocation of PROC GLIMMIX:

```
proc glimmix;
   class A;
   model Y = A x1 x2 x1*x2;
   lsmeans A;
   lsmeans A / at means;
   lsmeans A / at x1=1.2;
   lsmeans A / at (x1 x2)=(1.2 0.3);
run;
```

For the first two LSMEANS statements, the LS-means coefficient for x1 is $\overline{x}_1$ (the mean of x1) and for x2 is $\overline{x}_2$ (the mean of x2). However, for the first LSMEANS statement, the coefficient for x1*x2 is $\overline{x_1 x_2}$, but for the second LSMEANS statement, the coefficient is $\overline{x}_1 \times \overline{x}_2$. The third LSMEANS statement sets the coefficient for x1 equal to 1.2 and leaves it at $\overline{x}_2$ for x2, and the final LSMEANS statement sets these values to 1.2 and 0.3, respectively.

Even if you specify a WEIGHT variable, the unweighted covariate means are used for the covariate coefficients if there is no AT specification. If you specify the AT option, WEIGHT or FREQ variables are taken into account as follows. The weighted covariate means are then used for the covariate coefficients for which no explicit AT values are given, or if you specify AT MEANS. Observations that do not contribute to the analysis because of a missing dependent variable are included in computing the covariate means. You should use the E option in conjunction with the AT option to check that the modified LS-means coefficients are the ones you want.

The AT option is disabled if you specify the BYLEVEL option.

BYLEVEL

requests that separate margins be computed for each level of the LSMEANS effect.

The standard LS-means have equal coefficients across classification effects. The BYLEVEL option changes these coefficients to be proportional to the observed margins. This adjustment is reasonable when you want your inferences to apply to a population that is not necessarily balanced but has the margins observed in the input data set. In this case, the resulting LS-means are actually equal to

raw means for fixed-effects models and certain balanced random-effects models, but their estimated standard errors account for the covariance structure that you have specified. If a WEIGHT statement is specified, PROC GLIMMIX uses weighted margins to construct the LS-means coefficients.

If the AT option is specified, the BYLEVEL option disables it.

CL

requests that *t*-type confidence limits be constructed for each of the LS-means. If DDFM=NONE, then PROC GLIMMIX uses infinite degrees of freedom for this test, essentially computing a *z* interval. The confidence level is 0.95 by default; this can be changed with the ALPHA= option. If you specify an ADJUST= option, then the confidence limits are adjusted for multiplicity, but if you also specify STEPDOWN, then only *p*-values are step-down adjusted, not the confidence limits.

CORR

displays the estimated correlation matrix of the least squares means as part of the "Least Squares Means" table.

COV

displays the estimated covariance matrix of the least squares means as part of the "Least Squares Means" table.

DF=*number*

specifies the degrees of freedom for the *t* test and confidence limits. The default is the denominator degrees of freedom taken from the "Type III Tests of Fixed Effects" table corresponding to the LS-means effect.

DIFF<=*difftype***>**
PDIFF<=*difftype***>**

requests that differences of the LS-means be displayed. The optional *difftype* specifies which differences to produce, with possible values ALL, ANOM, CONTROL, CONTROLL, and CONTROLU. The ALL value requests all pairwise differences, and it is the default. The CONTROL *difftype* requests differences with a control, which, by default, is the first level of each of the specified LSMEANS effects.

The ANOM value requests differences between each LS-mean and the average LS-mean, as in the *analysis of means* (Ott 1967). The average is computed as a weighted mean of the LS-means, the weights being inversely proportional to the diagonal entries of the

$$\mathbf{L}\left(\mathbf{X}'\mathbf{X}\right)^{-}\mathbf{L}'$$

matrix. If LS-means are nonestimable, this design-based weighted mean is replaced with an equally weighted mean. Note that the ANOM procedure in SAS/QC software implements both tables and graphics for the analysis of means with a variety of response types. For one-way designs and normal data with identity link, the DIFF=ANOM computations are equivalent to the results of PROC ANOM. If the LS-means being compared are uncorrelated, exact adjusted *p*-values and critical values for confidence limits can be computed in the analysis of means; see Nelson (1982, 1991, 1993); Guirguis and Tobias (2004) as well as the documentation for the ADJUST=NELSON option.

To specify which levels of the effects are the controls, list the quoted formatted values in parentheses after the CONTROL keyword. For example, if the effects A, B, and C are classification variables, each having two levels, 1 and 2, the following LSMEANS statement specifies the (1,2) level of A*B and the (2,1) level of B*C as controls:

```
lsmeans A*B B*C / diff=control('1' '2' '2' '1');
```

For multiple effects, the results depend upon the order of the list, and so you should check the output to make sure that the controls are correct.

Two-tailed tests and confidence limits are associated with the CONTROL *difftype*. For one-tailed results, use either the CONTROLL or CONTROLU *difftype*. The CONTROLL *difftype* tests whether the noncontrol levels are significantly smaller than the control; the upper confidence limits for the control minus the noncontrol levels are considered to be infinity and are displayed as missing. Conversely, the CONTROLU *difftype* tests whether the noncontrol levels are significantly larger than the control; the upper confidence limits for the noncontrol levels minus the control are considered to be infinity and are displayed as missing.

If you want to perform multiple comparison adjustments on the differences of LS-means, you must specify the ADJUST= option.

The differences of the LS-means are displayed in a table titled "Differences of Least Squares Means."

E

requests that the **L** matrix coefficients for the LSMEANS effects be displayed.

ILINK

requests that estimates and their standard errors in the "Least Squares Means" table also be reported on the scale of the mean (the inverse linked scale). The ILINK option is specific to an LSMEANS statement. If you also specify the CL option, the GLIMMIX procedure computes confidence intervals for the predicted means by applying the inverse link transform to the confidence limits on the linked (linear) scale. Standard errors on the inverse linked scale are computed by the delta method.

The GLIMMIX procedure applies the inverse link transform to the LS-mean reported in the Estimate column. In a logistic model, for example, this implies that the value reported as the inversely linked estimate corresponds to a predicted probability that is based on an average estimable function (the estimable function that produces the LS-mean on the linear scale). To compute average predicted probabilities, you can average the results from applying the ILINK option in the ESTIMATE statement for suitably chosen estimable functions.

LINES

presents results of comparisons between all pairs of least squares means by listing the means in descending order and indicating nonsignificant subsets by line segments beside the corresponding LS-means. When all differences have the same variance, these comparison lines are guaranteed to accurately reflect the inferences based on the corresponding tests, made by comparing the respective *p*-values to the value of the ALPHA= option (0.05 by default). However, equal variances might not be the case for differences between LS-means. If the variances are not all the same, then the comparison lines might be conservative, in the sense that if you base your inferences on the lines alone, you will detect fewer significant differences than the tests indicate. If there are any such differences, PROC GLIMMIX lists the pairs of means that are inferred to be significantly different by the tests but not by the comparison lines. Note, however, that in many cases, even though the variances are unequal, they are similar enough that the comparison lines accurately reflect the test inferences.

ODDS

requests that in models with logit, cumulative logit, and generalized logit link function the odds of the levels of the fixed effects are reported. If you specify the CL or ALPHA= option, confidence intervals for the odds are also computed. See the section "Odds and Odds Ratio Estimation" on page 3225 for further details about computation and interpretation of odds and odds ratios with the GLIMMIX procedure.

ODDSRATIO

OR

requests that LS-mean differences (DIFF, ADJUST= options) and simple effect comparisons (SLICEDIFF option) are also reported in terms of odds ratios. The ODDSRATIO option is ignored unless you use either the logit, cumulative logit, or generalized logit link function. If you specify the CL or ALPHA= option, confidence intervals for the odds ratios are also computed. These intervals are adjusted for multiplicity when you specify the ADJUST= option. See the section "Odds and Odds Ratio Estimation" on page 3225 for further details about computation and interpretation of odds and odds ratios with the GLIMMIX procedure.

OBSMARGINS

OM

specifies a potentially different weighting scheme for the computation of LS-means coefficients. The standard LS-means have equal coefficients across classification effects; however, the OM option changes these coefficients to be proportional to those found in the input data set. This adjustment is reasonable when you want your inferences to apply to a population that is not necessarily balanced but has the margins observed in your data.

In computing the observed margins, PROC GLIMMIX uses all observations for which there are no missing or invalid independent variables, including those for which there are missing dependent variables. Also, if you use a WEIGHT statement, PROC GLIMMIX computes weighted margins to construct the LS-means coefficients. If your data are balanced, the LS-means are unchanged by the OM option.

The BYLEVEL option modifies the observed-margins LS-means. Instead of computing the margins across all of the input data set, PROC GLIMMIX computes separate margins for each level of the LSMEANS effect in question. In this case the resulting LS-means are actually equal to raw means for fixed-effects models and certain balanced random-effects models, but their estimated standard errors account for the covariance structure that you have specified.

You can use the E option in conjunction with either the OM or BYLEVEL option to check that the modified LS-means coefficients are the ones you want. It is possible that the modified LS-means are not estimable when the standard ones are estimable, or vice versa.

PDIFF

is the same as the DIFF option. See the description of the DIFF option on page 3116.

PLOT | PLOTS< =*plot-request*< (*options*) > >

PLOT | PLOTS< =(*plot-request*< (*options*) > < . . . *plot-request*< (*options*) > >) >

creates least squares means related graphs when ODS Graphics has been enabled and the plot request does not conflict with other *options* in the LSMEANS statement. For general information about ODS Graphics, see Chapter 21, "Statistical Graphics Using ODS." For examples of the basic statistical

graphics for least squares means and aspects of their computation and interpretation, see the section "Graphics for LS-Mean Comparisons" on page 3256 in this chapter.

The *options* for a specific plot request (and their *suboptions*) of the LSMEANS statement include those for the PLOTS= option in the PROC GLIMMIX statement. You can specify classification effects in the MEANPLOT request of the LSMEANS statement to control the display of interaction means with the PLOTBY= and SLICEBY= suboptions; these are not available in the PLOTS= option in the PROC GLIMMIX statement. Options specified in the LSMEANS statement override those in the PLOTS= option in the PROC GLIMMIX statement.

The available *options* and *suboptions* are as follows.

ALL
: requests that the default plots corresponding to this LSMEANS statement be produced. The default plot depends on the options in the statement.

ANOMPLOT

ANOM
: requests an analysis of means display in which least squares means are compared to an average least squares mean. Least squares mean ANOM plots are produced only for those model effects listed in LSMEANS statements that have options that do not contradict with the display. For example, the following statements produce analysis of mean plots for effects A and C:

```
lsmeans A / diff=anom plot=anom;
lsmeans B / diff      plot=anom;
lsmeans C /           plot=anom;
```

The DIFF option in the second LSMEANS statement implies all pairwise differences.

CONTROLPLOT

CONTROL
: requests a display in which least squares means are visually compared against a reference level. These plots are produced only for statements with options that are compatible with control differences. For example, the following statements produce control plots for effects A and C:

```
lsmeans A / diff=control('1') plot=control;
lsmeans B / diff              plot=control;
lsmeans C                     plot=control;
```

The DIFF option in the second LSMEANS statement implies all pairwise differences.

DIFFPLOT<(*diffplot-options***)>**

DIFFOGRAM<(*diffplot-options***)>**

DIFF<(*diffplot-options***)>**
: requests a display of all pairwise least squares mean differences and their significance. The display is also known as a "mean-mean scatter plot" when it is based on arithmetic means (Hsu 1996; Hsu and Peruggia 1994). For each comparison a line segment, centered at the LS-means in the pair, is drawn. The length of the segment corresponds to the projected width of a confidence interval for the least squares mean difference. Segments that fail to cross the 45-degree reference line correspond to significant least squares mean differences.

LS-mean difference plots are produced only for statements with options that are compatible with the display. For example, the following statements request differences against a control level for the A effect, all pairwise differences for the B effect, and the least squares means for the C effect:

```
lsmeans A / diff=control('1') plot=diff;
lsmeans B / diff              plot=diff;
lsmeans C                     plot=diff;
```

The DIFF= type in the first statement is incompatible with a display of all pairwise differences.

You can specify the following *diffplot-options*. The ABS and NOABS options determine the positioning of the line segments in the plot. When the ABS option is in effect, and this is the default, all line segments are shown on the same side of the reference line. The NOABS option separates comparisons according to the sign of the difference. The CENTER option marks the center point for each comparison. This point corresponds to the intersection of two least squares means. The NOLINES option suppresses the display of the line segments that represent the confidence bounds for the differences of the least squares means. The NOLINES option implies the CENTER option. The default is to draw line segments in the upper portion of the plot area without marking the center point.

MEANPLOT< (*meanplot-options*) >
requests displays of the least squares means.

The following *meanplot-options* control the display of the least squares means.

ASCENDING
displays the least squares means in ascending order. This option has no effect if means are sliced or displayed in separate plots.

CL
displays upper and lower confidence limits for the least squares means. By default, 95% limits are drawn. You can change the confidence level with the ALPHA= option. Confidence limits are drawn by default if the CL option is specified in the LSMEANS statement.

CLBAND
displays confidence limits as bands. This option implies the JOIN option.

DESCENDING
displays the least squares means in descending order. This option has no effect if means are sliced or displayed in separate plots.

ILINK
requests that means (and confidence limits) are displayed on the inverse linked scale.

JOIN
CONNECT
connects the least squares means with lines. This option is implied by the CLBAND option. If the effect contains nested variables, and a SLICEBY= effect contains classification variables that appear as crossed effects, this option is ignored.

SLICEBY=*fixed-effect*
specifies an effect by which to group the means in a single plot. For example, the following statement requests a plot in which the levels of A are placed on the horizontal axis and the means that belong to the same level of B are joined by lines:

```
lsmeans A*B / plot=meanplot(sliceby=b join);
```

Unless the LS-mean effect contains at least two classification variables, the SLICEBY= option has no effect. The *fixed-effect* does not have to be an effect in your MODEL statement, but it must consist entirely of classification variables.

PLOTBY=*fixed-effect*
specifies an effect by which to break interaction plots into separate displays. For example, the following statement requests for each level of C one plot of the A*B cell means that are associated with that level of C:

```
lsmeans A*B*C / plot=meanplot(sliceby=b plotby=c clband);
```

In each plot, levels of A are displayed on the horizontal axis, and confidence bands are drawn around the means that share the same level of B.

The PLOTBY= option has no effect unless the LS-mean effect contains at least three classification variables. The *fixed-effect* does not have to be an effect in the MODEL statement, but it must consist entirely of classification variables.

NONE
requests that no plots be produced.

When LS-mean calculations are adjusted for multiplicity by using the ADJUST= option, the plots are adjusted accordingly.

SINGULAR=*number*
tunes the estimability checking as documented for the CONTRAST statement.

SLICE=*fixed-effect* | **(***fixed-effects***)**
specifies effects by which to partition interaction LSMEANS effects. This can produce what are known as tests of simple effects (Winer 1971). For example, suppose that A*B is significant, and you want to test the effect of A for each level of B. The appropriate LSMEANS statement is

```
lsmeans A*B / slice=B;
```

This statement tests for the simple main effects of A for B, which are calculated by extracting the appropriate rows from the coefficient matrix for the A*B LS-means and by using them to form an *F* test.

The SLICE option produces *F* tests that test the simultaneous equality of cell means at a fixed level of the slice effect (Schabenberger, Gregoire, and Kong 2000). You can request differences of the least squares means while holding one or more factors at a fixed level with the SLICEDIFF= option.

The SLICE option produces a table titled "Tests of Effect Slices."

SLICEDIFF=*fixed-effect* | **(***fixed-effects***)**

SIMPLEDIFF=*fixed-effect* | **(***fixed-effects***)**

requests that differences of simple effects be constructed and tested against zero. Whereas the SLICE option extracts multiple rows of the coefficient matrix and forms an F test, the SLICEDIFF option tests pairwise differences of these rows. This enables you to perform multiple comparisons among the levels of one factor at a fixed level of the other factor. For example, assume that, in a balanced design, factors A and B have $a = 4$ and $b = 3$ levels, respectively. Consider the following statements:

```
proc glimmix;
   class a b;
   model y = a b a*b;
   lsmeans a*b / slice=a;
   lsmeans a*b / slicediff=a;
run;
```

The first LSMEANS statement produces four F tests, one per level of A. The first of these tests is constructed by extracting the three rows corresponding to the first level of A from the coefficient matrix for the A*B interaction. Call this matrix $\mathbf{L}_{a1}$ and its rows $\mathbf{l}_{a1}^{(1)}$, $\mathbf{l}_{a1}^{(2)}$, and $\mathbf{l}_{a1}^{(3)}$. The SLICE tests the two-degrees-of-freedom hypothesis

$$H: \begin{cases} \left(\mathbf{l}_{a1}^{(1)} - \mathbf{l}_{a1}^{(2)}\right)\boldsymbol{\beta} = 0 \\ \left(\mathbf{l}_{a1}^{(1)} - \mathbf{l}_{a1}^{(3)}\right)\boldsymbol{\beta} = 0 \end{cases}$$

In a balanced design, where μ_{ij} denotes the mean response if A is at level i and B is at level j, this hypothesis is equivalent to $H: \mu_{11} = \mu_{12} = \mu_{13}$. The SLICEDIFF option considers the three rows of $\mathbf{L}_{a1}$ in turn and performs tests of the difference between pairs of rows. How these differences are constructed depends on the SLICEDIFFTYPE= option. By default, all pairwise differences within the subset of $\mathbf{L}$ are considered; in the example this corresponds to tests of the form

$$H: \left(\mathbf{l}_{a1}^{(1)} - \mathbf{l}_{a1}^{(2)}\right)\boldsymbol{\beta} = 0$$
$$H: \left(\mathbf{l}_{a1}^{(1)} - \mathbf{l}_{a1}^{(3)}\right)\boldsymbol{\beta} = 0$$
$$H: \left(\mathbf{l}_{a1}^{(2)} - \mathbf{l}_{a1}^{(3)}\right)\boldsymbol{\beta} = 0$$

In the example, with $a = 4$ and $b = 3$, the second LSMEANS statement produces four sets of least squares means differences. Within each set, factor A is held fixed at a particular level and each set consists of three comparisons.

When the ADJUST= option is specified, the GLIMMIX procedure also adjusts the tests for multiplicity. The adjustment is based on the number of comparisons within each level of the SLICEDIFF= effect; see the SLICEDIFFTYPE= option. The Nelson adjustment is not available for slice differences.

SLICEDIFFTYPE< =*difftype* **>**

SIMPLEDIFFTYPE< =*difftype* **>**

determines the type of simple effect differences produced with the SLICEDIFF= option.

The possible values for the *difftype* are ALL, CONTROL, CONTROLL, and CONTROLU. The *difftype* ALL requests all simple effects differences, and it is the default. The *difftype* CONTROL requests

the differences with a control, which, by default, is the first level of each of the specified LSMEANS effects.

To specify which levels of the effects are the controls, list the quoted formatted values in parentheses after the keyword CONTROL. For example, if the effects A, B, and C are classification variables, each having three levels (1, 2, and 3), the following LSMEANS statement specifies the (1,3) level of A*B as the control:

```
lsmeans A*B / slicediff=(A B)
              slicedifftype=control('1' '3');
```

This LSMEANS statement first produces simple effects differences holding the levels of A fixed, and then it produces simple effects differences holding the levels of B fixed. In the former case, level '3' of B serves as the control level. In the latter case, level '1' of A serves as the control.

For multiple effects, the results depend upon the order of the list, and so you should check the output to make sure that the controls are correct.

Two-tailed tests and confidence limits are associated with the CONTROL *difftype*. For one-tailed results, use either the CONTROLL or CONTROLU *difftype*. The CONTROLL *difftype* tests whether the noncontrol levels are significantly smaller than the control; the upper confidence limits for the control minus the noncontrol levels are considered to be infinity and are displayed as missing. Conversely, the CONTROLU *difftype* tests whether the noncontrol levels are significantly larger than the control; the upper confidence limits for the noncontrol levels minus the control are considered to be infinity and are displayed as missing.

STEPDOWN< (*step-down options*) >
requests that multiple comparison adjustments for the *p*-values of LS-mean differences be further adjusted in a step-down fashion. Step-down methods increase the power of multiple comparisons by taking advantage of the fact that a *p*-value will never be declared significant unless all smaller *p*-values are also declared significant. Note that the STEPDOWN adjustment combined with ADJUST=BON corresponds to the methods of Holm (1979) and "Method 2" of Shaffer (1986); this is the default. Using step-down-adjusted *p*-values combined with ADJUST=SIMULATE corresponds to the method of Westfall (1997).

If the degrees-of-freedom method is DDFM=KENWARDROGER or DDFM=SATTERTHWAITE, then step-down-adjusted *p*-values are produced only if the ADJDFE=ROW option is in effect.

Also, STEPDOWN affects only *p*-values, not confidence limits. For ADJUST=SIMULATE, the generalized least squares hybrid approach of Westfall (1997) is employed to increase Monte Carlo accuracy.

You can specify the following *step-down options* in parentheses:

MAXTIME=*n*
specifies the time (in seconds) to spend computing the maximal logically consistent sequential subsets of equality hypotheses for TYPE=LOGICAL. The default is MAXTIME=60. If the MAXTIME value is exceeded, the adjusted tests are not computed. When this occurs, you can try increasing the MAXTIME value. However, note that there are common multiple comparisons problems for which this computation requires a huge amount of time—for example, all pairwise comparisons between more than 10 groups. In such cases, try to use TYPE=FREE (the default) or TYPE=LOGICAL(*n*) for small *n*.

REPORT
specifies that a report on the step-down adjustment should be displayed, including a listing of the sequential subsets (Westfall 1997) and, for ADJUST=SIMULATE, the step-down simulation results.

TYPE=LOGICAL< (*n*) > | FREE
If you specify TYPE=LOGICAL, the step-down adjustments are computed by using maximal logically consistent sequential subsets of equality hypotheses (Shaffer 1986; Westfall 1997). Alternatively, for TYPE=FREE, sequential subsets are computed ignoring logical constraints. The TYPE=FREE results are more conservative than those for TYPE=LOGICAL, but they can be much more efficient to produce for many comparisons. For example, it is not feasible to take logical constraints between all pairwise comparisons of more than 10 groups. For this reason, TYPE=FREE is the default.

However, you can reduce the computational complexity of taking logical constraints into account by limiting the depth of the search tree used to compute them, specifying the optional depth parameter as a number n in parentheses after TYPE=LOGICAL. As with TYPE=FREE, results for TYPE=LOGICAL(n) are conservative relative to the true TYPE=LOGICAL results, but even for TYPE=LOGICAL(0) they can be appreciably less conservative than TYPE=FREE and they are computationally feasible for much larger numbers of comparisons. If you do not specify n or if $n = -1$, the full search tree is used.

LSMESTIMATE Statement

LSMESTIMATE *fixed-effect* < '*label*' > *values* < *divisor*=*n* >
< , < '*label*' > *values* < *divisor*=*n* > > < , . . . >
< / *options* > ;

The LSMESTIMATE statement provides a mechanism for obtaining custom hypothesis tests among the least squares means. In contrast to the hypotheses tested with the ESTIMATE or CONTRAST statements, the LSMESTIMATE statement enables you to form linear combinations of the least squares means, rather than linear combination of fixed-effects parameter estimates and/or random-effects solutions. Multiple-row sets of coefficients are permitted.

The computation of an LSMESTIMATE involves two coefficient matrices. Suppose that the *fixed-effect* has n_l levels. Then the LS-means are formed as $\mathbf{L}_1\widehat{\boldsymbol{\beta}}$, where $\mathbf{L}_1$ is a $(n_l \times p)$ coefficient matrix. The $(k \times n_l)$ coefficient matrix $\mathbf{K}$ is formed from the *values* that you supply in the k rows of the LSMESTIMATE statement. The least squares means estimates then represent the $(k \times 1)$ vector

$$\mathbf{K}\mathbf{L}_1\boldsymbol{\beta} = \mathbf{L}\boldsymbol{\beta}$$

The GLIMMIX procedure supports nonpositional syntax for the coefficients (*values*) in the LSMESTIMATE statement. For details see the section "Positional and Nonpositional Syntax for Contrast Coefficients" on page 3232.

PROC GLIMMIX produces a t test for each row of coefficients specified in the LSMESTIMATE statement. You can adjust p-values and confidence intervals for multiplicity with the ADJUST= option. You can obtain an F test of single-row or multirow LSMESTIMATEs with the FTEST option.

Note that in contrast to a multirow estimate in the ESTIMATE statement, you specify only a single fixed effect in the LSMESTIMATE statement. The row labels are optional and follow the effects specification. For example, the following statements fit a split-split-plot design and compare the average of the third and fourth LS-mean of the whole-plot factor A to the first LS-mean of the factor:

```
proc glimmix;
   class a b block;
   model y = a b a*b /  s;
   random int a / sub=block;
   lsmestimate A 'a1 vs avg(a3,a4)' 2 0 -1 -1 divisor=2;
run;
```

The order in which coefficients are assigned to the least squares means corresponds to the order in which they are displayed in the "Least Squares Means" table. You can use the ELSM option to see how coefficients are matched to levels of the *fixed-effect*.

The optional *divisor=n* specification enables you to assign a separate divisor to each row of the LSMESTIMATE. You can also assign divisor values through the DIVISOR= option. See the documentation that follows for the interaction between the two ways of specifying divisors.

Many options of the LSMESTIMATE statement affect the computation of least squares means—for example, the AT=, BYLEVEL, and OM options. See the documentation for the LSMEANS statement for details.

Table 44.9 summarizes the *options* available in the LSMESTIMATE statement.

Table 44.9 LSMESTIMATE Statement Options

Option	Description
Construction and Computation of LS-Means	
AT	Modifies covariate values in computing LS-means
BYLEVEL	Computes separate margins
DIVISOR=	Specifies a list of values to divide the coefficients
OM=	Specifies the weighting scheme for LS-means computation as determined by a data set
SINGULAR=	Tunes estimability checking
Degrees of Freedom and *p*-values	
ADJDFE=	Determines denominator degrees of freedom when *p*-values and confidence limits are adjusted for multiple comparisons
ADJUST=	Determines the method for multiple comparison adjustment of LS-means differences
ALPHA=α	Determines the confidence level $(1-\alpha)$
CHISQ	Requests a chi-square test in addition to the F test
DF=	Assigns a specific value to degrees of freedom for tests and confidence limits
FTEST	Produces an F test
LOWER	Performs one-sided, lower-tailed inference
STEPDOWN	Adjusts multiple comparison *p*-values further in a step-down fashion
UPPER	Performs one-sided, upper-tailed inference

Table 44.9 *continued*

Option	Description
Statistical Output	
CL	Constructs confidence limits for means and mean differences
CORR	Displays the correlation matrix of LS-means
COV	Displays the covariance matrix of LS-means
E	Prints the **L** matrix
ELSM	Prints the **K** matrix
JOINT	Produces a joint F or chi-square test for the LS-means and LS-means differences
Generalized Linear Modeling	
EXP	Exponentiates and displays LS-means estimates
ILINK	Computes and displays estimates and standard errors of LS-means (but not differences) on the inverse linked scale

You can specify the following *options* in the LSMESTIMATE statement after a slash (/).

ADJDFE=SOURCE | ROW
specifies how denominator degrees of freedom are determined when p-values and confidence limits are adjusted for multiple comparisons with the ADJUST= option. When you do not specify the ADJDFE= option, or when you specify ADJDFE=SOURCE, the denominator degrees of freedom for multiplicity-adjusted results are the denominator degrees of freedom for the LS-mean effect in the "Type III Tests of Fixed Effects" table.

The ADJDFE=ROW setting is useful if you want multiplicity adjustments to take into account that denominator degrees of freedom are not constant across estimates. This can be the case, for example, when DDFM=SATTERTHWAITE or DDFM=KENWARDROGER is specified in the MODEL statement.

ADJUST=BON | SCHEFFE | SIDAK | SIMULATE<(*simoptions*)> | T
requests a multiple comparison adjustment for the p-values and confidence limits for the LS-mean estimates. The adjusted quantities are produced in addition to the unadjusted p-values and confidence limits. Adjusted confidence limits are produced if the CL or ALPHA= option is in effect. For a description of the adjustments, see Chapter 45, "The GLM Procedure," and Chapter 67, "The MULTTEST Procedure," as well as the documentation for the ADJUST= option in the LSMEANS statement.

Note that not all adjustment methods of the LSMEANS statement are available for the LSMESTIMATE statement. Multiplicity adjustments in the LSMEANS statement are designed specifically for differences of least squares means.

If you specify the STEPDOWN option, the p-values are further adjusted in a step-down fashion.

ALPHA=*number*
requests that a t-type confidence interval be constructed for each of the LS-means with confidence level 1 – *number*. The value of *number* must be between 0 and 1; the default is 0.05.

AT *variable*=*value*

AT (*variable-list*)=(*value-list*)

AT MEANS

enables you to modify the values of the covariates used in computing LS-means. See the AT option in the LSMEANS statement for details.

BYLEVEL

requests that PROC GLIMMIX compute separate margins for each level of the LSMEANS effect.

The standard LS-means have equal coefficients across classification effects. The BYLEVEL option changes these coefficients to be proportional to the observed margins. This adjustment is reasonable when you want your inferences to apply to a population that is not necessarily balanced but has the margins observed in the input data set. In this case, the resulting LS-means are actually equal to raw means for fixed-effects models and certain balanced random-effects models, but their estimated standard errors account for the covariance structure that you have specified. If a WEIGHT statement is specified, PROC GLIMMIX uses weighted margins to construct the LS-means coefficients.

If the AT option is specified, the BYLEVEL option disables it.

CHISQ

requests that chi-square tests be performed in addition to F tests, when you request an F test with the FTEST option.

CL

requests that t-type confidence limits be constructed for each of the LS-means. If DDFM=NONE, then PROC GLIMMIX uses infinite degrees of freedom for this test, essentially computing a z interval. The confidence level is 0.95 by default; this can be changed with the ALPHA= option.

CORR

displays the estimated correlation matrix of the linear combination of the least squares means.

COV

displays the estimated covariance matrix of the linear combination of the least squares means.

DF=*number*

specifies the degrees of freedom for the t test and confidence limits. The default is the denominator degrees of freedom taken from the "Type III Tests of Fixed Effects" table corresponding to the LS-means effect.

DIVISOR=*value-list*

specifies a list of values by which to divide the coefficients so that fractional coefficients can be entered as integer numerators. If you do not specify *value-list*, a default value of 1.0 is assumed. Missing values in the *value-list* are converted to 1.0.

If the number of elements in *value-list* exceeds the number of rows of the estimate, the extra values are ignored. If the number of elements in *value-list* is less than the number of rows of the estimate, the last value in *value-list* is carried forward.

If you specify a row-specific divisor as part of the specification of the estimate row, this value multiplies the corresponding value in the *value-list*. For example, the following statement divides the coefficients in the first row by 8, and the coefficients in the third and fourth row by 3:

```
lsmestimate A 'One vs. two'   8 -8   divisor=2,
              'One vs. three' 1  0 -1         ,
              'One vs. four'  3  0  0 -3      ,
              'One vs. five'  3  0  0  0  -3  / divisor=4,.,3;
```

Coefficients in the second row are not altered.

E

requests that the **L** coefficients of the estimable function be displayed. These are the coefficients that apply to the fixed-effect parameter estimates. The E option displays the coefficients that you would need to enter in an equivalent ESTIMATE statement.

ELSM

requests that the **K** matrix coefficients be displayed. These are the coefficients that apply to the LS-means. This option is useful to ensure that you assigned the coefficients correctly to the LS-means.

EXP

requests exponentiation of the least squares means estimate. When you model data with the logit link function and the estimate represents a log odds ratio, the EXP option produces an odds ratio. See the section "Odds and Odds Ratio Estimation" on page 3225 for important details concerning the computation and interpretation of odds and odds ratio results with the GLIMMIX procedure. If you specify the CL or ALPHA= option, the (adjusted) confidence limits for the estimate are also exponentiated.

FTEST< (*joint-test-options*) >

JOINT< (*joint-test-options*) >

produces an F test that jointly tests the rows of the LSMESTIMATE against zero. If the LOWER or UPPER options are in effect or if you specify boundary values with the BOUNDS= suboption, the GLIMMIX procedure computes a simulation-based p-value for the constrained joint test. For more information about these simulation-based p-values, see the section "Joint Hypothesis Tests with Complex Alternatives, the Chi-Bar-Square Statistic" on page 457 in Chapter 19, "Shared Concepts and Topics." You can specify the following *joint-test-options* in parentheses:

ACC=γ

specifies the accuracy radius for determining the necessary sample size in the simulation-based approach of Silvapulle and Sen (2004) for tests with order restrictions. The value of γ must be strictly between 0 and 1; the default value is 0.005.

BOUNDS=*value-list*

specifies boundary values for the estimable linear function. The null value of the hypothesis is always zero. If you specify a positive boundary value z, the hypotheses are $H\colon \theta = 0$ vs. $H_a\colon \theta > 0$ with the added constraint that $\theta < z$. The same is true for negative boundary values. The alternative hypothesis is then $H_a\colon \theta < 0$ subject to the constraint $\theta > -|z|$. If you specify a missing value, the hypothesis is assumed to be two-sided. The BOUNDS option enables you to specify sets of one- and two-sided joint hypotheses. If all values in *value-list* are set to missing, the procedure performs a simulation-based p-value calculation for a two-sided test.

EPS=ϵ

specifies the accuracy confidence level for determining the necessary sample size in the simulation-based approach of Silvapulle and Sen (2004) for F tests with order restrictions. The value of ϵ must be strictly between 0 and 1; the default value is 0.01.

LABEL='*label*'

enables you to assign a label to the joint test that identifies the results in the "LSMFtest" table. If you do not specify a label, the first non-default label for the LSMESTIMATE rows is used to label the joint test.

NSAMP=*n*

specifies the number of samples for the simulation-based method of Silvapulle and Sen (2004). If *n* is not specified, it is constructed from the values of the ALPHA=α, the ACC=γ, and the EPS=ϵ options. With the default values for γ, ϵ, and α (0.005, 0.01, and 0.05, respectively), NSAMP=12,604 by default.

ILINK

requests that the estimate and its standard error also be reported on the scale of the mean (the inverse linked scale). PROC GLIMMIX computes the value on the mean scale by applying the inverse link to the estimate. The interpretation of this quantity depends on the coefficients that are specified in your LSMESTIMATE statement and the link function. For example, in a model for binary data with a logit link, the following LSMESTIMATE statement computes

$$q = \frac{1}{1 + \exp\{-(\tau_1 - \tau_2)\}}$$

where τ_1 and τ_2 are the least squares means associated with the first two levels of the classification effect A:

```
proc glimmix;
   class A;
   model y = A / dist=binary link=logit;
   lsmestimate A 1 -1 / ilink;
run;
```

The quantity q is not the difference of the probabilities associated with the two levels,

$$\pi_1 - \pi_2 = \frac{1}{1 + \exp\{-\tau_1\}} - \frac{1}{1 + \exp\{-\tau_2\}}$$

The standard error of the inversely linked estimate is based on the delta method. If you also specify the CL or ALPHA= option, the GLIMMIX procedure computes confidence intervals for the inversely linked estimate. These intervals are obtained by applying the inverse link to the confidence intervals on the linked scale.

JOINT<(*joint-test-options***)>**

is an alias for the FTEST option.

LOWER

LOWERTAILED

requests that the *p*-value for the *t* test be based only on values that are less than the test statistic. A two-tailed test is the default. A lower-tailed confidence limit is also produced if you specify the CL or ALPHA= option.

Note that for ADJUST=SCHEFFE the one-sided adjusted confidence intervals and one-sided adjusted *p*-values are the same as the corresponding two-sided statistics, because this adjustment is based on only the right tail of the *F* distribution.

If you request an *F* test with the FTEST option, then a one-sided left-tailed order restriction is applied to all estimable functions, and the corresponding chi-bar-square statistic of Silvapulle and Sen (2004) is computed in addition to the two-sided, standard *F* or chi-square statistic. See the description of the FTEST option for information about how to control the computation of the simulation-based chi-bar-square statistic.

OBSMARGINS

OM

specifies a potentially different weighting scheme for the computation of LS-means coefficients. The standard LS-means have equal coefficients across classification effects; however, the OM option changes these coefficients to be proportional to those found in the input data set. See the OBSMARGINS option in the LSMEANS statement for further details.

SINGULAR=*number*

tunes the estimability checking as documented for the CONTRAST statement.

STEPDOWN< (*step-down-options***) >**

requests that multiplicity adjustments for the *p*-values of LS-mean estimates be further adjusted in a step-down fashion. Step-down methods increase the power of multiple testing procedures by taking advantage of the fact that a *p*-value will never be declared significant unless all smaller *p*-values are also declared significant. Note that the STEPDOWN adjustment combined with ADJUST=BON corresponds to the Holm (1979) and "Method 2" of Shaffer (1986); this is the default. Using step-down-adjusted *p*-values combined with ADJUST=SIMULATE corresponds to the method of Westfall (1997).

If the degrees-of-freedom method is DDFM=KENWARDROGER or DDFM=SATTERTHWAITE, then step-down-adjusted *p*-values are produced only if the ADJDFE=ROW option is in effect.

Also, the STEPDOWN option affects only *p*-values, not confidence limits. For ADJUST=SIMULATE, the generalized least squares hybrid approach of Westfall (1997) is employed to increase Monte Carlo accuracy.

You can specify the following *step-down-options* in parentheses:

MAXTIME=*n*

specifies the time (in seconds) to spend computing the maximal logically consistent sequential subsets of equality hypotheses for TYPE=LOGICAL. The default is MAXTIME=60. If the MAXTIME value is exceeded, the adjusted tests are not computed. When this occurs, you can try increasing the MAXTIME value. However, note that there are common multiple comparisons problems for which this computation requires a huge amount of time—for example, all pairwise comparisons between more than 10 groups. In such cases, try to use TYPE=FREE (the default) or TYPE=LOGICAL(*n*) for small *n*.

ORDER=PVALUE | ROWS

specifies the order in which the step-down tests are performed. ORDER=PVALUE is the default, with LS-mean estimates being declared significant only if all LS-mean estimates with smaller (unadjusted) *p*-values are significant. If you specify ORDER=ROWS, then significances are evaluated in the order in which they are specified.

REPORT

specifies that a report on the step-down adjustment be displayed, including a listing of the sequential subsets (Westfall 1997) and, for ADJUST=SIMULATE, the step-down simulation results.

TYPE=LOGICAL< (*n*) > | FREE

If you specify TYPE=LOGICAL, the step-down adjustments are computed by using maximal logically consistent sequential subsets of equality hypotheses (Shaffer 1986; Westfall 1997). Alternatively, for TYPE=FREE, logical constraints are ignored when sequential subsets are computed. The TYPE=FREE results are more conservative than those for TYPE=LOGICAL, but they can be much more efficient to produce for many estimates. For example, it is not feasible to take logical constraints between all pairwise comparisons of more than about 10 groups. For this reason, TYPE=FREE is the default.

However, you can reduce the computational complexity of taking logical constraints into account by limiting the depth of the search tree used to compute them, specifying the optional depth parameter as a number *n* in parentheses after TYPE=LOGICAL. As with TYPE=FREE, results for TYPE=LOGICAL(*n*) are conservative relative to the true TYPE=LOGICAL results, but even for TYPE=LOGICAL(0), they can be appreciably less conservative than TYPE=FREE, and they are computationally feasible for much larger numbers of estimates. If you do not specify *n* or if *n* = –1, the full search tree is used.

UPPER

UPPERTAILED

requests that the *p*-value for the *t* test be based only on values that are greater than the test statistic. A two-tailed test is the default. An upper-tailed confidence limit is also produced if you specify the CL or ALPHA= option.

Note that for ADJUST=SCHEFFE the one-sided adjusted confidence intervals and one-sided adjusted *p*-values are the same as the corresponding two-sided statistics, because this adjustment is based on only the right tail of the *F* distribution.

If you request a joint test with the FTEST option, then a one-sided right-tailed order restriction is applied to all estimable functions, and the corresponding chi-bar-square statistic of Silvapulle and Sen (2004) is computed in addition to the two-sided, standard *F* or chi-square statistic. See the FTEST option for information about how to control the computation of the simulation-based chi-bar-square statistic.

MODEL Statement

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

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

The MODEL statement is required and names the dependent variable and the fixed effects. The *fixed-effects* determine the **X** matrix of the model (see the section "Notation for the Generalized Linear Mixed Model" for details). The specification of effects is the same as in the GLM or MIXED procedure. In contrast to PROC GLM, you do not specify random effects in the MODEL statement. However, in contrast to PROC GLM and PROC MIXED, continuous variables on the left and right side of the MODEL statement can be computed through PROC GLIMMIX programming statements.

An intercept is included in the fixed-effects model by default. It can be removed with the NOINT option.

The dependent variable can be specified by using either the *response* syntax or the *events/trials* syntax. The *events/trials* syntax is specific to models for binomial data. A binomial(n,π) variable is the sum of n independent Bernoulli trials with event probability π. Each Bernoulli trial results in either an event or a nonevent (with probability $1-\pi$). You use the *events/trials* syntax to indicate to the GLIMMIX procedure that the Bernoulli outcomes are grouped. The value of the second variable, *trials*, gives the number n of Bernoulli trials. The value of the first variable, *events*, is the number of events out of n. The values of both *events* and (*trials*–*events*) must be nonnegative and the value of trials must be positive. Observations for which these conditions are not met are excluded from the analysis. If the *events/trials* syntax is used, the GLIMMIX procedure defaults to the binomial distribution. The response is then the *events* variable. The *trials* variable is accounted in model fitting as an additional weight. If you use the *response* syntax, the procedure defaults to the normal distribution.

There are two sets of options in the MODEL statement. The *response-options* determine how the GLIMMIX procedure models probabilities for binary and multinomial data. The *model-options* control other aspects of model formation and inference. Table 44.10 summarizes the options available in the MODEL statement. These are subsequently discussed in detail in alphabetical order by option category.

Table 44.10 MODEL Statement Options

Option	Description
Response Variable Options	
DESCENDING	Reverses the order of response categories
EVENT=	Specifies the event category in binary models
ORDER=	Specifies the sort order for the response variable
REFERENCE=	Specifies the reference category in generalized logit models
Model Building	
DIST=	Specifies the response distribution
LINK=	Specifies the link function
NOINT	Excludes fixed-effect intercept from model
OFFSET=	Specifies the offset variable for linear predictor
OBSWEIGHT=	Specifies the weight variable for the observation level unit
Statistical Computations	
ALPHA=α	Determines the confidence level ($1-\alpha$)
CHISQ	Requests chi-square tests
DDF=	Specifies the denominator degrees of freedom (list)
DDFM=	Specifies the method for computing denominator degrees of freedom
HTYPE=	Selects the type of hypothesis test
LWEIGHT	Determines how weights are used

Table 44.10 *continued*

Option	Description
NOCENTER	Suppresses centering and scaling of **X** columns during the estimation phase
REFLINP	Specifies a value for the linear predictor
ZETA=	Tunes sensitivity in computing Type III functions
Statistical Output	
CL	Displays confidence limits for fixed-effects parameter estimates
CORRB	Displays the correlation matrix of fixed-effects parameter estimates
COVB	Displays the covariance matrix of fixed-effects parameter estimates
COVBI	Displays the inverse covariance matrix of fixed-effects parameter estimates
E, E1, E2, E3	Displays the **L** matrix coefficients
INTERCEPT	Adds a row for the intercept to test tables
ODDSRATIO	Displays odds ratios and confidence limits
SOLUTION	Displays fixed-effects parameter estimates (and scale parameter in GLM models)
STDCOEF	Displays standardized coefficients

Response Variable Options

Response variable options determine how the GLIMMIX procedure models probabilities for binary and multinomial data.

You can specify the following *options* by enclosing them in parentheses after the response variable. See the section "Response-Level Ordering and Referencing" on page 3235 for more detail and examples.

DESCENDING
DESC
reverses the order of the response categories. If both the DESCENDING and ORDER= options are specified, PROC GLIMMIX orders the response categories according to the ORDER= option and then reverses that order.

EVENT='*category***' | *keyword***
specifies the event category for the binary response model. PROC GLIMMIX models the probability of the event category. The EVENT= option has no effect when there are more than two response categories. You can specify the value (formatted, if a format is applied) of the event category in quotes, or you can specify one of the following *keywords*:

FIRST
designates the first ordered category as the event. This is the default.

LAST
designates the last ordered category as the event.

ORDER=DATA | FORMATTED | FREQ | INTERNAL
specifies the sort order for the levels of the response variable. When ORDER=FORMATTED (the default) for numeric variables for which you have supplied no explicit format (that is, for which there

is no corresponding FORMAT statement in the current PROC GLIMMIX run or in the DATA step that created the data set), the levels are ordered by their internal (numeric) value. If you specify the ORDER= option in the MODEL statement and the ORDER= option in the PROC GLIMMIX statement, the former takes precedence. The following table shows the interpretation of the ORDER= values:

Value of ORDER=	Levels Sorted By
DATA	order of appearance in the input data set
FORMATTED	external formatted value, except for numeric variables with no explicit format, which are sorted by their unformatted (internal) value
FREQ	descending frequency count; levels with the most observations come first in the order
INTERNAL	unformatted value

By default, ORDER=FORMATTED. For the FORMATTED and INTERNAL values, the sort order is machine dependent.

For more information about sort order, see the chapter on the SORT procedure in the *Base SAS Procedures Guide* and the discussion of BY-group processing in *SAS Language Reference: Concepts*.

REFERENCE='*category*' | *keyword*

REF='*category*' | *keyword*

specifies the reference category for the generalized logit model and the binary response model. For the generalized logit model, each nonreference category is contrasted with the reference category. For the binary response model, specifying one response category as the reference is the same as specifying the other response category as the event category. You can specify the value (formatted if a format is applied) of the reference category in quotes, or you can specify one of the following *keywords*:

FIRST

designates the first ordered category as the reference category.

LAST

designates the last ordered category as the reference category. This is the default.

Model Options

ALPHA=*number*

requests that a *t*-type confidence interval be constructed for each of the fixed-effects parameters with confidence level 1 – *number*. The value of *number* must be between 0 and 1; the default is 0.05.

CHISQ

requests that chi-square tests be performed for all specified effects in addition to the *F* tests. Type III tests are the default; you can produce the Type I and Type II tests by using the HTYPE= option.

CL

requests that *t*-type confidence limits be constructed for each of the fixed-effects parameter estimates. The confidence level is 0.95 by default; this can be changed with the ALPHA= option.

CORRB

produces the correlation matrix from the approximate covariance matrix of the fixed-effects parameter estimates.

COVB< (DETAILS) >

produces the approximate variance-covariance matrix of the fixed-effects parameter estimates $\widehat{\boldsymbol{\beta}}$. In a generalized linear mixed model this matrix typically takes the form $(\mathbf{X}'\widehat{\mathbf{V}}^{-1}\mathbf{X})^{-}$ and can be obtained by sweeping the mixed model equations; see the section "Estimated Precision of Estimates" on page 3189. In a model without random effects, it is obtained from the inverse of the observed or expected Hessian matrix. Which Hessian is used in the computation depends on whether the procedure is in scoring mode (see the SCORING= option in the PROC GLIMMIX statement) and whether the EXPHESSIAN option is in effect. Note that if you use EMPIRICAL= or DDFM=KENWARDROGER, the matrix displayed by the COVB option is the empirical (sandwich) estimator or the adjusted estimator, respectively.

The DETAILS suboption of the COVB option enables you to obtain a table of statistics about the covariance matrix of the fixed effects. If an adjusted estimator is used because of the EMPIRICAL= or DDFM=KENWARDROGER option, the GLIMMIX procedure displays statistics for the adjusted and unadjusted estimators as well as statistics comparing them. This enables you to diagnose, for example, changes in rank (because of an insufficient number of subjects for the empirical estimator) and to assess the extent of the covariance adjustment. In addition, the GLIMMIX procedure then displays the unadjusted (=model-based) covariance matrix of the fixed-effects parameter estimates. For more details, see the section "Exploring and Comparing Covariance Matrices" on page 3216.

COVBI

produces the inverse of the approximate covariance matrix of the fixed-effects parameter estimates.

DDF=*value-list*

DF=*value-list*

enables you to specify your own denominator degrees of freedom for the fixed effects. The *value-list* specification is a list of numbers or missing values (.) separated by commas. The degrees of freedom should be listed in the order in which the effects appear in the "Type III Tests of Fixed Effects" table. If you want to retain the default degrees of freedom for a particular effect, use a missing value for its location in the list. For example, the statement assigns 3 denominator degrees of freedom to A and 4.7 to A*B, while those for B remain the same:

```
model Y = A B A*B / ddf=3,.,4.7;
```

If you select a degrees-of-freedom method with the DDFM= option, then nonmissing, positive values in *value-list* override the degrees of freedom for the particular effect. For example, the statement assigns 3 and 6 denominator degrees of freedom in the test of the A main effect and the A*B interaction, respectively:

```
model Y = A B A*B / ddf=3,.,6  ddfm=Satterth;
```

The denominator degrees of freedom for the test for the B effect are determined from a Satterthwaite approximation.

Note that the DDF= and DDFM= options determine the degrees of freedom in the "Type I Tests of Fixed Effects," "Type II Tests of Fixed Effects," and "Type III Tests of Fixed Effects" tables. These

degrees of freedom are also used in determining the degrees of freedom in tests and confidence intervals from the CONTRAST, ESTIMATE, LSMEANS, and LSMESTIMATE statements. Exceptions from this rule are noted in the documentation for the respective statements.

DDFM=*method*

specifies the method for computing the denominator degrees of freedom for the tests of fixed effects that result from the MODEL, CONTRAST, ESTIMATE, LSMEANS, and LSMESTIMATE statements.

You can specify the following *methods*:

BETWITHIN

BW

assigns within-subject degrees of freedom to a fixed effect if the fixed effect changes within a subject, and between-subject degrees of freedom otherwise. This method is the default for models with only R-side random effects and a SUBJECT= option. Computational details can be found in the section "Degrees of Freedom Methods" on page 3210.

CONTAIN

CON

invokes the containment method to compute denominator degrees of freedom. This method is the default when the model contains G-side random effects. Computational details can be found in the section "Degrees of Freedom Methods" on page 3210.

KENWARDROGER< (FIRSTORDER) >

KENROGER< (FIRSTORDER) >

KR< (FIRSTORDER) >

applies the (prediction) standard error and degrees-of-freedom correction detailed by Kenward and Roger (1997). This approximation involves adjusting the estimated variance-covariance matrix of the fixed and random effects in a manner similar to that of Prasad and Rao (1990); Harville and Jeske (1992); Kackar and Harville (1984). Satterthwaite-type degrees of freedom are then computed based on this adjustment. Computational details can be found in the section "Degrees of Freedom Methods" on page 3210.

KENWARDROGER2

KENROGER2

KR2

applies the (prediction) standard error and degrees-of-freedom correction that are detailed by Kenward and Roger (2009). This correction further reduces the precision estimator bias for the fixed and random effects under nonlinear covariance structures. Computational details can be found in the section "Degrees of Freedom Methods" on page 3210.

NONE

specifies that no denominator degrees of freedom be applied. PROC GLIMMIX then essentially assumes that infinite degrees of freedom are available in the calculation of p-values. The p-values for t tests are then identical to p-values that are derived from the standard normal distribution. In the case of F tests, the p-values are equal to those of chi-square tests that are determined as follows: if F_{obs} is the observed value of the F test with l numerator degrees of freedom, then

$$p = \Pr\{F_{l,\infty} > F_{obs}\} = \Pr\{\chi^2_l > lF_{obs}\}$$

Regardless of the DDFM= method, you can obtain these chi-square p-values with the CHISQ option in the MODEL statement.

RESIDUAL

RES

performs all tests by using the residual degrees of freedom, $n - \mathrm{rank}(\mathbf{X})$, where n is the sum of the frequencies of observations or the sum of frequencies of *event/trials* pairs. This method is the default degrees of freedom method for GLMs and overdispersed GLMs.

SATTERTHWAITE

SAT

performs a general Satterthwaite approximation for the denominator degrees of freedom in a generalized linear mixed model. This method is a generalization of the techniques that are described in Giesbrecht and Burns (1985); McLean and Sanders (1988); Fai and Cornelius (1996). The method can also include estimated random effects. Computational details can be found in the section "Degrees of Freedom Methods" on page 3210.

When the asymptotic variance matrix of the covariance parameters is found to be singular, a generalized inverse is used. Covariance parameters with zero variance then do not contribute to the degrees of freedom adjustment for DDFM=SATTERTH and DDFM=KENWARDROGER, and a message is written to the log.

DISTRIBUTION=*keyword*

DIST=*keyword*

D=*keyword*

ERROR=*keyword*

E=*keyword*

specifies the built-in (conditional) probability distribution of the data. If you specify the DIST= option and you do not specify a user-defined link function, a default link function is chosen according to the following table. If you do not specify a distribution, the GLIMMIX procedure defaults to the normal distribution for continuous response variables and to the multinomial distribution for classification or character variables, unless the *events/trial* syntax is used in the MODEL statement. If you choose the *events/trial* syntax, the GLIMMIX procedure defaults to the binomial distribution.

Table 44.11 lists the values of the DIST= option and the corresponding default link functions. For the case of generalized linear models with these distributions, you can find expressions for the log-likelihood functions in the section "Maximum Likelihood" on page 3181.

Table 44.11 Keyword Values of the DIST= Option

DIST=	Distribution	Default Link Function	Numeric Value
BETA	beta	logit	12
BINARY	binary	logit	4
BINOMIAL | BIN | B	binomial	logit	3
EXPONENTIAL | EXPO	exponential	log	9
GAMMA | GAM	gamma	log	5
GAUSSIAN | *G* **| NORMAL | N**	normal	identity	1
GEOMETRIC | GEOM	geometric	log	8
INVGAUSS | IGAUSSIAN | IG	inverse Gaussian	inverse squared (power(–2))	6
LOGNORMAL | LOGN	lognormal	identity	11
MULTINOMIAL | MULTI | MULT	multinomial	cumulative logit	NA
NEGBINOMIAL | NEGBIN | NB	negative binomial	log	7
POISSON | POI | P	Poisson	log	2
TCENTRAL | TDIST | T	*t*	identity	10
BYOBS(*variable*)	multivariate	varied	NA

Note that the PROC GLIMMIX default link for the gamma or exponential distribution is not the canonical link (the reciprocal link).

The numeric value in the last column of Table 44.11 can be used in combination with DIST=BYOBS. The BYOBS(*variable*) syntax designates a variable whose value identifies the distribution to which an observation belongs. If the variable is numeric, its values must match values in the last column of Table 44.11. If the variable is not numeric, an observation's distribution is identified by the first four characters of the distribution's name in the leftmost column of the table. Distributions whose numeric value is "NA" cannot be used with DIST=BYOBS.

If the variable in BYOBS(*variable*) is a data set variable, it can also be used in the CLASS statement of the GLIMMIX procedure. For example, this provides a convenient method to model multivariate data jointly while varying fixed-effects components across outcomes. Assume that, for example, for each patient, a count and a continuous outcome were observed; the count data are modeled as Poisson data and the continuous data are modeled as gamma variates. The following statements fit a Poisson and a gamma regression model simultaneously:

```
proc sort data=yourdata;
   by patient;
run;
data yourdata;
   set yourdata;
   by patient;
   if first.patient then dist='POIS' else dist='GAMM';
run;
proc glimmix data=yourdata;
   class treatment dist;
   model y = dist treatment*dist / dist=byobs(dist);
run;
```

The two models have separate intercepts and treatment effects. To correlate the outcomes, you can share a random effect between the observations from the same patient:

```
proc glimmix data=yourdata;
   class treatment dist patient;
   model y = dist treatment*dist / dist=byobs(dist);
   random intercept / subject=patient;
run;
```

Or, you could use an R-side correlation structure:

```
proc glimmix data=yourdata;
   class treatment dist patient;
   model y = dist treatment*dist / dist=byobs(dist);
   random _residual_ / subject=patient type=un;
run;
```

Although DIST=BYOBS(*variable*) is used to model multivariate data, you only need a single response variable in PROC GLIMMIX. The responses are in "univariate" form. This allows, for example, different missing value patterns across the responses. It does, however, require that all response variables be numeric.

The default links that are assigned when DIST=BYOBS is in effect correspond to the respective default links in Table 44.11.

When you choose DIST=LOGNORMAL, the GLIMMIX procedure models the logarithm of the response variable as a normal random variable. That is, the mean and variance are estimated on the logarithmic scale, assuming a normal distribution, $\log\{Y\} \sim N(\mu, \sigma^2)$. This enables you to draw on options that require a distribution in the exponential family—for example, by using a scoring algorithm in a GLM. To convert means and variances for $\log\{Y\}$ into those of Y, use the relationships

$$\begin{aligned} \mathrm{E}[Y] &= \exp\{\mu\}\sqrt{\omega} \\ \mathrm{Var}[Y] &= \exp\{2\mu\}\omega(\omega - 1) \\ \omega &= \exp\{\sigma^2\} \end{aligned}$$

The DIST=T option models the data as a shifted and scaled central t variable. This enables you to model data with heavy-tailed distributions. If Y denotes the response and X has a t_ν distribution with ν degrees of freedom, then PROC GLIMMIX models

$$Y = \mu + \sqrt{\phi}X$$

In this parameterization, Y has mean μ and variance $\phi\nu/(\nu - 2)$.

By default, $\nu = 3$. You can supply different degrees of freedom for the t variate as in the following statements:

```
proc glimmix;
   class a b;
   model y = b x b*x / dist=tcentral(9.6);
   random a;
run;
```

The GLIMMIX procedure does not accept values for the degrees of freedom parameter less than 3.0. If the *t* distribution is used with the DIST=BYOBS(*variable*) specification, the degrees of freedom are fixed at $\nu = 3$. For mixed models where parameters are estimated based on linearization, choosing DIST=T instead of DIST=NORMAL affects only the residual variance, which decreases by the factor $\nu/(\nu - 2)$.

Note that in SAS 9.1, the GLIMMIX procedure modeled $Y = \mu + \phi^* \sqrt{\frac{\nu-2}{\nu}} X$. The scale parameter of the parameterizations are related as $\phi = \phi^* \times \phi^* \times (\nu - 2)/\nu$.

The DIST=BETA option implements the parameterization of the beta distribution in Ferrari and Cribari-Neto (2004). If Y has a $\text{beta}(\alpha, \beta)$ density, so that $\text{E}[Y] = \mu = \alpha/(\alpha + \beta)$, this parameterization uses the variance function $a(\mu) = \mu(1 - \mu)$ and $\text{Var}[Y] = a(\mu)/(1 + \phi)$.

See the section "Maximum Likelihood" on page 3181 for the log likelihoods of the distributions fitted by the GLIMMIX procedure.

E
requests that Type I, Type II, and Type III **L** matrix coefficients be displayed for all specified effects.

E1 | EI
requests that Type I **L** matrix coefficients be displayed for all specified effects.

E2 | EII
requests that Type II **L** matrix coefficients be displayed for all specified effects.

E3 | EIII
requests that Type III **L** matrix coefficients be displayed for all specified effects.

HTYPE=*value-list*
indicates the type of hypothesis test to perform on the fixed effects. Valid entries for values in the *value-list* are 1, 2, and 3, corresponding to Type I, Type II, and Type III tests. The default value is 3. You can specify several types by separating the values with a comma or a space. The ODS table names are "Tests1," "Tests2," and "Tests3" for the Type I, Type II, and Type III tests, respectively.

INTERCEPT
adds a row to the tables for Type I, II, and III tests corresponding to the overall intercept.

LINK=*keyword*
specifies the link function in the generalized linear mixed model. The *keywords* and their associated built-in link functions are shown in Table 44.12.

Table 44.12 Built-in Link Functions of the GLIMMIX Procedure

LINK=	Link Function	$g(\mu) = \eta =$	Numeric Value
CUMCLL | CCLL	cumulative complementary log-log	$\log(-\log(1 - \pi))$	NA
CUMLOGIT | CLOGIT	cumulative logit	$\log(\gamma/(1 - \pi))$	NA
CUMLOGLOG	cumulative log-log	$-\log(-\log(\pi))$	NA
CUMPROBIT | CPROBIT	cumulative probit	$\Phi^{-1}(\pi)$	NA
CLOGLOG | CLL	complementary log-log	$\log(-\log(1 - \mu))$	5

Table 44.12 *continued*

LINK=	Link Function	$g(\mu) = \eta =$	Numeric Value
GLOGIT | GENLOGIT	generalized logit		NA
IDENTITY | ID	identity	μ	1
LOG	log	$\log(\mu)$	4
LOGIT	logit	$\log(\mu/(1-\mu))$	2
LOGLOG	log-log	$-\log(-\log(\mu))$	6
PROBIT	probit	$\Phi^{-1}(\mu)$	3
POWER(λ) | **POW**(λ)	power with exponent λ= *number*	$\begin{cases} \mu^{\lambda} & \text{if } \lambda \neq 0 \\ \log(\mu) & \text{if } \lambda = 0 \end{cases}$	NA
POWERMINUS2	power with exponent -2	$1/\mu^2$	8
RECIPROCAL | INVERSE	reciprocal	$1/\mu$	7
BYOBS(*variable*)	varied	varied	NA

For the probit and cumulative probit links, $\Phi^{-1}(\cdot)$ denotes the quantile function of the standard normal distribution. For the other cumulative links, π denotes a cumulative category probability. The cumulative and generalized logit link functions are appropriate only for the multinomial distribution. When you choose a cumulative link function, PROC GLIMMIX assumes that the data are ordinal. When you specify LINK=GLOGIT, the GLIMMIX procedure assumes that the data are nominal (not ordered).

The numeric value in the rightmost column of Table 44.12 can be used in conjunction with LINK=BYOBS(*variable*). This syntax designates a *variable* whose values identify the link function associated with an observation. If the variable is numeric, its values must match those in the last column of Table 44.12. If the variable is not numeric, an observation's link function is determined by the first four characters of the link's name in the first column. Those link functions whose numeric value is "NA" cannot be used with LINK=BYOBS(*variable*).

You can define your own link function through programming statements. See the section "User-Defined Link or Variance Function" on page 3177 for more information about how to specify a link function. If a user-defined link function is in effect, the specification in the LINK= option is ignored. If you specify neither the LINK= option nor a user-defined link function, then the default link function is chosen according to Table 44.11.

LWEIGHT=FIRSTORDER | FIRO

LWEIGHT=NONE

LWEIGHT=VAR

determines how weights are used in constructing the coefficients for Type I through Type III **L** matrices. The default is LWEIGHT=VAR, and the values of the WEIGHT variable are used in forming crossproduct matrices. If you specify LWEIGHT=FIRO, the weights incorporate the WEIGHT variable as well as the first-order weights of the linearized model. For LWEIGHT=NONE, the **L** matrix coefficients are based on the raw crossproduct matrix, whether a WEIGHT variable is specified or not.

NOCENTER

requests that the columns of the **X** matrix are not centered and scaled. By default, the columns of **X** are centered and scaled. Unless the NOCENTER option is in effect, **X** is replaced by $\mathbf{X}^*$ during estimation. The columns of $\mathbf{X}^*$ are computed as follows:

- In models with an intercept, the intercept column remains the same and the *j*th entry in row *i* of $\mathbf{X}^*$ is

$$x^*_{ij} = \frac{x_{ij} - \overline{x}_j}{\sqrt{\sum_{i=1}^{n}(x_{ij} - \overline{x}_j)^2}}$$

- In models without intercept, no centering takes place and the *j*th entry in row *i* of $\mathbf{X}^*$ is

$$x^*_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{n}(x_{ij} - \overline{x}_j)^2}}$$

The effects of centering and scaling are removed when results are reported. For example, if the covariance matrix of the fixed effects is printed with the COVB option of the MODEL statement, the covariances are reported in terms of the original parameters, not the centered and scaled versions. If you specify the STDCOEF option, fixed-effects parameter estimates and their standard errors are reported in terms of the standardized (scaled and/or centered) coefficients in addition to the usual results in noncentered form.

NOINT

requests that no intercept be included in the fixed-effects model. An intercept is included by default.

ODDSRATIO< (*odds-ratio-options*) >

OR< (*odds-ratio-options*) >

requests estimates of odds ratios and their confidence limits, provided the link function is the logit, cumulative logit, or generalized logit. Odds ratios are produced for the following:

- classification main effects, if they appear in the MODEL statement
- continuous variables in the MODEL statement, unless they appear in an interaction with a classification effect
- continuous variables in the MODEL statement at fixed levels of a classification effect, if the MODEL statement contains an interaction of the two
- continuous variables in the MODEL statement, if they interact with other continuous variables

You can specify the following *odds-ratio-options* to create customized odds ratio results.

AT *var-list*=*value-list*

specifies the reference values for continuous variables in the model. By default, the average value serves as the reference. Consider, for example, the following statements:

```
proc glimmix;
   class A;
   model y = A x A*x / dist=binary oddsratio;
run;
```

Odds ratios for A are based on differences of least squares means for which x is set to its mean. Odds ratios for x are computed by differencing two sets of least squares mean for the A factor. One set is computed at $x = \overline{x} + 1$, and the second set is computed at $x = \overline{x}$. The following MODEL statement changes the reference value for x to 3:

```
model y = A x A*x / dist=binary
                    oddsratio(at x=3);
```

DIFF<=*difftype***>**
controls the type of differences for classification main effects. By default, odds ratios compare the odds of a response for level *j* of a factor to the odds of the response for the last level of that factor (DIFF=LAST). The DIFF=FIRST option compares the levels against the first level, DIFF=ALL produces odds ratios based on all pairwise differences, and DIFF=NONE suppresses odds ratios for classification main effects.

LABEL
displays a label in the "Odds Ratio Estimates" table. The table describes the comparison associated with the table row.

UNIT *var-list*=*value-list*
specifies the units in which the effects of continuous variable in the model are assessed. By default, odds ratios are computed for a change of one unit from the average. Consider a model with a classification factor A with 4 levels. The following statements produce an "Odds Ratio Estimates" table with 10 rows:

```
proc glimmix;
   class A;
   model y = A x A*x / dist=binary
                       oddsratio(diff=all unit x=2);
run;
```

The first $4 \times 3/2 = 6$ rows correspond to pairwise differences of levels of A. The underlying log odds ratios are computed as differences of A least squares means. In the least squares mean computation the covariate x is set to $\overline{x}$. The next four rows compare least squares means for A at $x = \overline{x} + 2$ and at $x = \overline{x}$. You can combine the AT and UNIT options to produce custom odds ratios. For example, the following statements produce an "Odds Ratio Estimates" table with 8 rows:

```
proc glimmix;
   class A;
   model y = A x x*z / dist=binary
                       oddsratio(diff=all
                                 at   x   = 3
                                 unit x z = 2 4);
run;
```

The first $4 \times 3/2 = 6$ rows correspond to pairwise differences of levels of A. The underlying log odds ratios are computed as differences of A least squares means. In the least squares mean computation, the covariate x is set to 3, and the covariate x*z is set to $3\overline{z}$. The next odds ratio measures the effect of a change in x. It is based on differencing the linear predictor for $x = 3 + 2$ and $x*z = (3 + 2)\overline{z}$ with the linear predictor for $x = 3$ and $x*z = 3\overline{z}$. The last odds ratio expresses

a change in z by contrasting the linear predictors based on x = 3 and x*z = $3(\bar{z} + 4)$ with the predictor based on x = 3 and x*z = $3\bar{z}$.

To compute odds and odds ratios for general estimable functions and least squares means, see the ODDSRATIO option in the LSMEANS statement and the EXP options in the ESTIMATE and LSMESTIMATE statements.

For important details concerning interpretation and computation of odds ratios with the GLIMMIX procedure, see the section "Odds and Odds Ratio Estimation" on page 3225.

OFFSET=*variable*

specifies a variable to be used as an offset for the linear predictor. An offset plays the role of a fixed effect whose coefficient is known to be 1. You can use an offset in a Poisson model, for example, when counts have been obtained in time intervals of different lengths. With a log link function, you can model the counts as Poisson variables with the logarithm of the time interval as the offset variable. The offset variable cannot appear in the CLASS statement or elsewhere in the MODEL or RANDOM statement.

REFLINP=*r*

specifies a value for the linear predictor of the reference level in the generalized logit model for nominal data. By default *r*=0.

SOLUTION

S

requests that a solution for the fixed-effects parameters be produced. Using notation from the section "Notation for the Generalized Linear Mixed Model" on page 3054, the fixed-effects parameter estimates are $\widehat{\boldsymbol{\beta}}$, and their (approximate) estimated standard errors are the square roots of the diagonal elements of $\widehat{\mathrm{Var}}[\widehat{\boldsymbol{\beta}}]$. This matrix commonly is of the form $(\mathbf{X}'\widehat{\mathbf{V}}^{-1}\mathbf{X})^{-}$ in GLMMs. You can output this approximate variance matrix with the COVB option. See the section "Details: GLIMMIX Procedure" on page 3180 on the construction of $\widehat{\mathbf{V}}$ in the various models.

Along with the estimates and their approximate standard errors, a *t* statistic is computed as the estimate divided by its standard error. The degrees of freedom for this *t* statistic matches the one appearing in the "Type III Tests of Fixed Effects" table under the effect containing the parameter. If DDFM=KENWARDROGER or DDFM=SATTERTHWAITE, the degrees of freedom are computed separately for each fixed-effect estimate, unless you override the value for any specific effect with the DDF=*value-list* option. The "Pr > |t|" column contains the two-tailed *p*-value corresponding to the *t* statistic and associated degrees of freedom. You can use the CL option to request confidence intervals for the fixed-effects parameters; they are constructed around the estimate by using a radius of the standard error times a percentage point from the *t* distribution.

STDCOEF

reports solutions for fixed effects in terms of the standardized (scaled and/or centered) coefficients. This option has no effect when the NOCENTER option is specified or in models for multinomial data.

OBSWEIGHT< =*variable* **>**

OBSWT< =*variable* **>**

specifies a variable to be used as the weight for the observation-level unit in a weighted multilevel model. If a weight variable is not specified in the OBSWEIGHT option, a weight of 1 is used. For details on the use of weights in multilevel models, see the section "Pseudo-likelihood Estimation for Weighted Multilevel Models" on page 3200.

ZETA=*number*
tunes the sensitivity in forming Type III functions. Any element in the estimable function basis with an absolute value less than *number* is set to 0. The default is 1E–8.

NLOPTIONS Statement

NLOPTIONS < *options* > **;**

Most models fit with the GLIMMIX procedure typically have one or more nonlinear parameters. Estimation requires nonlinear optimization methods. You can control the optimization through options in the NLOPTIONS statement.

Several estimation methods of the GLIMMIX procedure (METHOD=RSPL, MSPL, RMPL, MMPL) are doubly iterative in the following sense. The generalized linear mixed model is approximated by a linear mixed model based on current values of the covariance parameter estimates. The resulting linear mixed model is then fit, which is itself an iterative process (with some exceptions). On convergence, new covariance parameters and fixed-effects estimates are obtained and the approximated linear mixed model is updated. Its parameters are again estimated iteratively. It is thus reasonable to refer to *outer* and *inner* iterations. The outer iterations involve the repeated updates of the linear mixed models, and the inner iterations are the iterative steps that lead to parameter estimates in any given linear mixed model. The NLOPTIONS statement controls the inner iterations. The outer iteration behavior can be controlled with options in the PROC GLIMMIX statement, such as the MAXLMMUPDATE=, PCONV=, and ABSPCONV= options. If the estimation method involves a singly iterative approach, then there is no need for the outer cycling and the model is fit in a single optimization controlled by the NLOPTIONS statement (see the section "Singly or Doubly Iterative Fitting" on page 3239).

The syntax and options of the NLOPTIONS statement are described in the section "NLOPTIONS Statement" on page 488 in Chapter 19, "Shared Concepts and Topics."

Note that in a GLMM with pseudo-likelihood estimation, specifying TECHNIQUE=NONE has the same effect as specifying the NOITER option in the PARMS statement. If you estimate the parameters by METHOD=LAPLACE or METHOD=QUAD, TECHNIQUE=NONE applies to the optimization after starting values have been determined.

The GLIMMIX procedure applies the default optimization technique shown in Table 44.13, depending on your model.

Table 44.13 Default Techniques

Model Family	Setting	TECHNIQUE=
GLM	DIST=NORMAL LINK=IDENTITY	NONE
GLM	otherwise	NEWRAP
GLMM	PARMS NOITER, PL	NONE
GLMM	binary data, PL	NRRIDG
GLMM	otherwise	QUANEW

OUTPUT Statement

OUTPUT < **OUT=***SAS-data-set* >
< *keyword*< **(***keyword-options***)** > < **=***name* > > . . .
< *keyword*< **(***keyword-options***)** > < **=***name* > > < / *options* > **;**

The OUTPUT statement creates a data set that contains predicted values and residual diagnostics, computed after fitting the model. By default, all variables in the original data set are included in the output data set.

You can use the ID statement to select a subset of the variables from the input data set as well as computed variables for adding to the output data set. If you reassign a data set variable through programming statements, the value of the variable from the input data set supersedes the recomputed value when observations are written to the output data set. If you list the variable in the ID statement, however, PROC GLIMMIX saves the current value of the variable after the programming statements have been executed.

For example, suppose that data set Scores contains the variables score, machine, and person. The following statements fit a model with fixed machine and random person effects. The variable score divided by 100 is assumed to follow an inverse Gaussian distribution. The (conditional) mean and residuals are saved to the data set igausout. Because no ID statement is given, the variable score in the output data set contains the values from the input data set.

```
proc glimmix;
   class machine person;
   score = score/100;
   p = 4*_linp_;
   model score = machine  / dist=invgauss;
   random int / sub=person;
   output out=igausout pred=p resid=r;
run;
```

On the contrary, the following statements list explicitly which variables to save to the OUTPUT data set. Because the variable score is listed in the ID statement, and is (re-)assigned through programming statements, the values of score saved to the OUTPUT data set are the input values divided by 100.

```
proc glimmix;
   class machine person;
   score = score / 100;
   model score = machine  / dist=invgauss;
   random int / sub=person;
   output out=igausout pred=p resid=r;
   id machine score _xbeta_ _zgamma_;
run;
```

You can specify the following syntax elements in the OUTPUT statement before the slash (/).

OUT=*SAS-data-set*
specifies the name of the output data set. If the OUT= option is omitted, the procedure uses the DATA*n* convention to name the output data set.

*keyword***<** **(***keyword-options***)** **>** **<** **=***name* **>**
specifies a statistic to include in the output data set and optionally assigns the variable the name *name*. You can use the *keyword-options* to control which type of a particular statistic to compute. The *keyword-options* can take on the following values:

BLUP	uses the predictors of the random effects in computing the statistic.
ILINK	computes the statistic on the scale of the data.
NOBLUP	does not use the predictors of the random effects in computing the statistic.
NOILINK	computes the statistic on the scale of the link function.

The default is to compute statistics by using BLUPs on the scale of the link function (the linearized scale). For example, the following OUTPUT statements are equivalent:

```
output out=out1  pred=predicted lcl=lower;


output out=out1  pred(blup noilink)=predicted
                 lcl (blup noilink)=lower;
```

If a particular combination of keyword and keyword options is not supported, the statistic is not computed and a message is produced in the SAS log.

A *keyword* can appear multiple times in the OUTPUT statement. Table 44.14 lists the *keywords* and the default names assigned by the GLIMMIX procedure if you do not specify a *name*. In this table, y denotes the observed response, and p denotes the linearized pseudo-data. See the section "Pseudo-likelihood Estimation Based on Linearization" on page 3188 for details on notation and the section "Notes on Output Statistics" on page 3246 for further details regarding the output statistics.

Table 44.14 Keywords for Output Statistics

Keyword	Options	Description	Expression	Name
PREDICTED	Default	Linear predictor	$\widehat{\eta} = \mathbf{x}'\widehat{\boldsymbol{\beta}} + \mathbf{z}'\widehat{\boldsymbol{\gamma}}$	Pred
	NOBLUP	Marginal linear predictor	$\widehat{\eta}_m = \mathbf{x}'\widehat{\boldsymbol{\beta}}$	PredPA
	ILINK	Predicted mean	$g^{-1}(\widehat{\eta})$	PredMu
	NOBLUP ILINK	Marginal mean	$g^{-1}(\widehat{\eta}_m)$	PredMuPA
STDERR	Default	Standard deviation of linear predictor	$\sqrt{\mathrm{Var}[\widehat{\eta} - \mathbf{z}'\boldsymbol{\gamma}]}$	StdErr
	NOBLUP	Standard deviation of marginal linear predictor	$\sqrt{\mathrm{Var}[\widehat{\eta}_m]}$	StdErrPA
	ILINK	Standard deviation of mean	$\sqrt{\mathrm{Var}[g^{-1}(\widehat{\eta} - \mathbf{z}'\boldsymbol{\gamma})]}$	StdErr
	NOBLUP ILINK	Standard deviation of marginal mean	$\sqrt{\mathrm{Var}[g^{-1}(\widehat{\eta}_m)]}$	StdErrMuPA
RESIDUAL	Default	Residual	$r = p - \widehat{\eta}$	Resid
	NOBLUP	Marginal residual	$r_m = p_m - \widehat{\eta}_m$	ResidPA
	ILINK	Residual on mean scale	$r_y = y - g^{-1}(\widehat{\eta})$	ResidMu
	NOBLUP ILINK	Marginal residual on mean scale	$r_{ym} = y - g^{-1}(\widehat{\eta}_m)$	ResidMuPA
PEARSON	Default	Pearson-type residual	$r/\sqrt{\widehat{\mathrm{Var}}[p\|\boldsymbol{\gamma}]}$	Pearson

Table 44.14 *continued*

Keyword	Options	Description	Expression	Name
	NOBLUP	Marginal Pearson-type residual	$r_m / \sqrt{\widehat{\mathrm{Var}}[p_m]}$	PearsonPA
	ILINK	Conditional Pearson-type mean residual	$r_y / \sqrt{\widehat{\mathrm{Var}}[Y\|\boldsymbol{\gamma}]}$	PearsonMu
STUDENT	Default	Studentized residual	$r / \sqrt{\widehat{\mathrm{Var}}[r]}$	Student
	NOBLUP	Studentized marginal residual	$r_m / \sqrt{\widehat{\mathrm{Var}}[r_m]}$	StudentPA
LCL	Default	Lower prediction limit for linear predictor		LCL
	NOBLUP	Lower confidence limit for marginal linear predictor		LCLPA
	ILINK	Lower prediction limit for mean		LCLMu
	NOBLUP ILINK	Lower confidence limit for marginal mean		LCLMuPA
UCL	Default	Upper prediction limit for linear predictor		UCL
	NOBLUP	Upper confidence limit for marginal linear predictor		UCLPA
	ILINK	Upper prediction limit for mean		UCLMu
	NOBLUP ILINK	Upper confidence limit for marginal mean		UCLMuPA
VARIANCE	Default	Conditional variance of pseudo-data	$\widehat{\mathrm{Var}}[p\|\boldsymbol{\gamma}]$	Variance
	NOBLUP	Marginal variance of pseudo-data	$\widehat{\mathrm{Var}}[p_m]$	VariancePA
	ILINK	Conditional variance of response	$\widehat{\mathrm{Var}}[Y\|\boldsymbol{\gamma}]$	Variance_Dep
	NOBLUP ILINK	Marginal variance of response	$\widehat{\mathrm{Var}}[Y]$	Variance_DepPA

Studentized residuals are computed only on the linear scale (scale of the link), unless the link is the identity, in which case the two scales are equal. The keywords RESIDUAL, PEARSON, STUDENT, and VARIANCE are not available with the multinomial distribution. You can use the following shortcuts to request statistics: PRED for PREDICTED, STD for STDERR, RESID for RESIDUAL, and VAR for VARIANCE. Output statistics that depend on the marginal variance $\mathrm{Var}[Y_i]$ are not available with METHOD=LAPLACE or METHOD=QUAD.

Table 44.15 summarizes the *options* available in the OUTPUT statement.

Table 44.15 OUTPUT Statement Options

Option	Description
ALLSTATS	Computes all statistics
ALPHA=α	Determines the confidence level $(1-\alpha)$
CPSEUDO	Changes the way in which marginal residuals are computed
DERIVATIVES	Adds derivatives of model quantities to the output data set
NOMISS	Outputs only observations used in the analysis
NOUNIQUE	Requests that names not be made unique
NOVAR	Requests that variables from the input data set not be added to the output data set
OBSCAT	Writes statistics to output data set only for the response level corresponding to the observed level of the observation
SYMBOLS	Adds computed variables to the output data set

You can specify the following *options* in the OUTPUT statement after a slash (/).

ALLSTATS
requests that all statistics are computed. If you do not use a keyword to assign a name, the GLIMMIX procedure uses the default name.

ALPHA=*number*
determines the coverage probability for two-sided confidence and prediction intervals. The coverage probability is computed as 1 – *number*. The value of *number* must be between 0 and 1; the default is 0.05.

CPSEUDO
changes the way in which marginal residuals are computed when model parameters are estimated by pseudo-likelihood methods. See the section "Notes on Output Statistics" on page 3246 for details.

DERIVATIVES
DER
adds derivatives of model quantities to the output data set. If, for example, the model fit requires the (conditional) log likelihood of the data, then the DERIVATIVES option writes for each observation the evaluations of the first and second derivatives of the log likelihood with respect to _LINP_ and _PHI_ to the output data set. The particular derivatives produced by the GLIMMIX procedure depend on the type of model and the estimation method.

NOMISS
requests that records be written to the output data only for those observations that were used in the analysis. By default, the GLIMMIX procedure produces output statistics for all observations in the input data set.

NOUNIQUE
requests that names not be made unique in the case of naming conflicts. By default, the GLIMMIX procedure avoids naming conflicts by assigning a unique name to each output variable. If you specify the NOUNIQUE option, variables with conflicting names are not renamed. In that case, the first variable added to the output data set takes precedence.

NOVAR
requests that variables from the input data set not be added to the output data set. This option does not apply to variables listed in the BY statement or to computed variables listed in the ID statement.

OBSCAT
requests that in models for multinomial data statistics be written to the output data set only for the response level that corresponds to the observed level of the observation.

SYMBOLS
SYM
adds to the output data set computed variables that are defined or referenced in the program.

PARMS Statement

PARMS <(*value-list*)> ... </ *options*> **;**

The PARMS statement specifies initial values for the covariance or scale parameters, or it requests a grid search over several values of these parameters in generalized linear mixed models.

The *value-list* specification can take any of several forms:

m	a single value
$m_1, m_2, \ldots, m_n$	several values
m to n	a sequence where m equals the starting value, n equals the ending value, and the increment equals 1
m to n by i	a sequence where m equals the starting value, n equals the ending value, and the increment equals i
m_1, m_2 to m_3	mixed values and sequences

Using the PARMS Statement with a GLM

If you are fitting a GLM or a GLM with overdispersion, the scale parameters are listed at the end of the "Parameter Estimates" table in the same order as *value-list*. If you specify more than one set of initial values, PROC GLIMMIX uses only the first value listed for each parameter. Grid searches by using scale parameters are not possible for these models, because the fixed effects are part of the optimization.

Using the PARMS Statement with a GLMM

If you are fitting a GLMM, the *value-list* corresponds to the parameters as listed in the "Covariance Parameter Estimates" table. Note that this order can change depending on whether a residual variance is profiled or not; see the NOPROFILE option in the PROC GLIMMIX statement.

If you specify more than one set of initial values, PROC GLIMMIX performs a grid search of the objective function surface and uses the best point on the grid for subsequent analysis. Specifying a large number of grid points can result in long computing times.

Options in the PARMS Statement

You can specify the following *options* in the PARMS statement after a slash (/).

HOLD=*value-list*

specifies which parameter values PROC GLIMMIX should hold equal to the specified values. For example, the following statement constrains the first and third covariance parameters to equal 5 and 2, respectively:

```
parms (5) (3) (2) (3) / hold=1,3;
```

Covariance or scale parameters that are held fixed with the HOLD= option are treated as constrained parameters in the optimization. This is different from evaluating the objective function, gradient, and Hessian matrix at known values of the covariance parameters. A constrained parameter introduces a singularity in the optimization process. The covariance matrix of the covariance parameters (see the ASYCOV option of the PROC GLIMMIX statement) is then based on the projected Hessian matrix. As a consequence, the variance of parameters subjected to a HOLD= is zero. Such parameters do not contribute to the computation of denominator degrees of freedom with the DDFM=KENWARDROGER and DDFM=SATTERTHWAITE methods, for example. If you want to treat the covariance parameters as known, without imposing constraints on the optimization, you should use the NOITER option.

When you place a hold on all parameters (or when you specify the NOITER) option in a GLMM, you might notice that PROC GLIMMIX continues to produce an iteration history. Unless your model is a linear mixed model, several recomputations of the pseudo-response might be required in linearization-based methods to achieve agreement between the pseudo-data and the covariance matrix. In other words, the GLIMMIX procedure continues to update the fixed-effects estimates (and random-effects solutions) until convergence is achieved.

In certain models, placing a hold on covariance parameters implies that the procedure processes the parameters in the same order as if the NOPROFILE were in effect. This can change the order of the covariance parameters when you place a hold on one or more parameters. Models that are subject to this reordering are those with R-side covariance structures whose scale parameter could be profiled. This includes the TYPE=CS, TYPE=SP, TYPE=AR(1), TYPE=TOEP, and TYPE=ARMA(1,1) covariance structures.

LOWERB=*value-list*

enables you to specify lower boundary constraints for the covariance or scale parameters. The *value-list* specification is a list of numbers or missing values (.) separated by commas. You must list the numbers in the same order that PROC GLIMMIX uses for the *value-list* in the PARMS statement, and each number corresponds to the lower boundary constraint. A missing value instructs PROC GLIMMIX to use its default constraint, and if you do not specify numbers for all of the covariance parameters, PROC GLIMMIX assumes that the remaining ones are missing.

This option is useful, for example, when you want to constrain the **G** matrix to be positive definite in order to avoid the more computationally intensive algorithms required when **G** becomes singular. The corresponding statements for a random coefficients model are as follows:

```
proc glimmix;
   class person;
   model y = time;
```

```
   random int time / type=chol sub=person;
   parms / lowerb=1e-4,.,1e-4;
run;
```

Here, the TYPE=CHOL structure is used in order to specify a Cholesky root parameterization for the 2×2 unstructured blocks in $\mathbf{G}$. This parameterization ensures that the $\mathbf{G}$ matrix is nonnegative definite, and the PARMS statement then ensures that it is positive definite by constraining the two diagonal terms to be greater than or equal to 1E–4.

NOBOUND

requests the removal of boundary constraints on covariance and scale parameters in mixed models. For example, variance components have a default lower boundary constraint of 0, and the NOBOUND option allows their estimates to be negative. See the NOBOUND option in the PROC GLIMMIX statement for further details.

NOITER

requests that no optimization of the covariance parameters be performed. This option has no effect in generalized linear models.

If you specify the NOITER option, PROC GLIMMIX uses the values for the covariance parameters given in the PARMS statement to perform statistical inferences. Note that the NOITER option is not equivalent to specifying a HOLD= value for all covariance parameters. If you use the NOITER option, covariance parameters are not constrained in the optimization. This prevents singularities that might otherwise occur in the optimization process.

If a residual variance is profiled, the parameter estimates can change from the initial values you provide as the residual variance is recomputed. To prevent an update of the residual variance, combine the NOITER option with the NOPROFILE option in the PROC GLIMMIX statements, as in the following code:

```
proc glimmix noprofile;
   class A B C rep mp sp;
   model y = A | B | C;
   random rep mp sp;
   parms (180) (200) (170) (1000) / noiter;
run;
```

When you specify the NOITER option in a model where parameters are estimated by pseudo-likelihood techniques, you might notice that the GLIMMIX procedure continues to produce an iteration history. Unless your model is a linear mixed model, several recomputations of the pseudo-response might be required in linearization-based methods to achieve agreement between the pseudo-data and the covariance matrix. In other words, the GLIMMIX procedure continues to update the profiled fixed-effects estimates (and random-effects solutions) until convergence is achieved. To prevent these updates, use the MAXLMMUPDATE= option in the PROC GLIMMIX statement. Specifying the NOITER option in the PARMS statement of a GLMM with pseudo-likelihood estimation has the same effect as choosing TECHNIQUE=NONE in the NLOPTIONS statement.

If you want to base initial fixed-effects estimates on the results of fitting a generalized linear model, then you can combine the NOITER option with the TECHNIQUE= option. For example, the following statements determine the starting values for the fixed effects by fitting a logistic model (without random effects) with the Newton-Raphson algorithm:

```
proc glimmix startglm inititer=10;
   class clinic A;
   model y/n = A / link=logit dist=binomial;
   random clinic;
   parms (0.4) / noiter;
   nloptions technique=newrap;
run;
```

The initial GLM fit stops at convergence or after at most 10 iterations, whichever comes first. The pseudo-data for the linearized GLMM is computed from the GLM estimates. The variance of the Clinic random effect is held constant at 0.4 during subsequent iterations that update the fixed effects only.

If you also want to combine the GLM fixed-effects estimates with known and fixed covariance parameter values without updating the fixed effects, you can add the MAXLMMUPDATE=0 option:

```
proc glimmix startglm inititer=10 maxlmmupdate=0;
   class clinic A;
   model y/n = A / link=logit dist=binomial;
   random clinic;
   parms (0.4) / noiter;
   nloptions technique=newrap;
run;
```

In a GLMM with parameter estimation by METHOD=LAPLACE or METHOD=QUAD the NOITER option also leads to an iteration history, since the fixed-effects estimates are part of the optimization and the PARMS statement places restrictions on only the covariance parameters.

Finally, the NOITER option can be useful if you want to obtain minimum variance quadratic unbiased estimates (with 0 priors), also known as MIVQUE0 estimates (Goodnight 1978a). Because MIVQUE0 estimates are starting values for covariance parameters—unless you provide (*value-list*) in the PARMS statement—the following statements produce MIVQUE0 mixed model estimates:

```
proc glimmix noprofile;
   class A B;
   model y = A;
   random int / subject=B;
   parms / noiter;
run;
```

PARMSDATA=*SAS-data-set*

PDATA=*SAS-data-set*

reads in covariance parameter values from a SAS data set. The data set should contain the numerical variable ESTIMATE or the numerical variables Covp1–Covp$q$, where q denotes the number of covariance parameters.

If the PARMSDATA= data set contains multiple sets of covariance parameters, the GLIMMIX procedure evaluates the initial objective function for each set and commences the optimization step by using the set with the lowest function value as the starting values. For example, the following SAS statements request that the objective function be evaluated for three sets of initial values:

```
data data_covp;
   input covp1-covp4;
   datalines;
  180 200 170 1000
  170 190 160  900
  160 180 150  800
;
proc glimmix;
   class A B C rep mainEU smallEU;
   model yield = A|B|C;
   random rep mainEU smallEU;
   parms / pdata=data_covp;
run;
```

Each set comprises four covariance parameters.

The order of the observations in a data set with the numerical variable Estimate corresponds to the order of the covariance parameters in the "Covariance Parameter Estimates" table. In a GLM, the PARMSDATA= option can be used to set the starting value for the exponential family scale parameter. A grid search is not conducted for GLMs if you specify multiple values.

The PARMSDATA= data set must not contain missing values.

If the GLIMMIX procedure is processing the input data set in BY groups, you can add the BY variables to the PARMSDATA= data set. If this data set is sorted by the BY variables, the GLIMMIX procedure matches the covariance parameter values to the current BY group. If the PARMSDATA= data set does not contain all BY variables, the data set is processed in its entirety for every BY group and a message is written to the log. This enables you to provide a single set of starting values across BY groups, as in the following statements:

```
data data_covp;
   input covp1-covp4;
   datalines;
  180 200 170 1000
;


proc glimmix;
   class A B C rep mainEU smallEU;
   model yield = A|B|C;
   random rep mainEU smallEU;
   parms / pdata=data_covp;
   by year;
run;
```

The same set of starting values is used for each value of the year variable.

UPPERB=*value-list*

enables you to specify upper boundary constraints on the covariance parameters. The *value-list* specification is a list of numbers or missing values (.) separated by commas. You must list the numbers in the same order that PROC GLIMMIX uses for the *value-list* in the PARMS statement, and each

number corresponds to the upper boundary constraint. A missing value instructs PROC GLIMMIX to use its default constraint. If you do not specify numbers for all of the covariance parameters, PROC GLIMMIX assumes that the remaining ones are missing.

RANDOM Statement

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

Using notation from "Notation for the Generalized Linear Mixed Model" on page 3054, the RANDOM statement defines the **Z** matrix of the mixed model, the random effects in the $\boldsymbol{\gamma}$ vector, the structure of **G**, and the structure of **R**.

The **Z** matrix is constructed exactly like the **X** matrix for the fixed effects, and the **G** matrix is constructed to correspond to the effects constituting **Z**. The structures of **G** and **R** are defined by using the TYPE= option described on page 3162. The random effects can be classification or continuous effects, and multiple RANDOM statements are possible.

Some reserved *keywords* have special significance in the *random-effects* list. You can specify INTERCEPT (or INT) as a random effect to indicate the intercept. PROC GLIMMIX does not include the intercept in the RANDOM statement by default as it does in the MODEL statement. You can specify the _RESIDUAL_ keyword (or RESID, RESIDUAL, _RESID_) before the option slash (/) to indicate a residual-type (R-side) random component that defines the **R** matrix. Basically, the _RESIDUAL_ keyword takes the place of the *random-effect* if you want to specify R-side variances and covariance structures. These *keywords* take precedence over variables in the data set with the same name. If your data or the covariance structure requires that an effect is specified, you can use the RESIDUAL option to instruct the GLIMMIX procedure to model the R-side variances and covariances.

In order to add an overdispersion component to the variance function, simply specify a single residual random component. For example, the following statements fit a polynomial Poisson regression model with overdispersion. The variance function $a(\mu) = \mu$ is replaced by $\phi a(\mu)$:

```
proc glimmix;
   model count = x x*x / dist=poisson;
   random _residual_;
run;
```

Table 44.16 summarizes the *options* available in the RANDOM statement. All *options* are subsequently discussed in alphabetical order.

Table 44.16 RANDOM Statement Options

Option	Description
Construction of Covariance Structure	
GCOORD=	Determines coordinate association for G-side spatial structures with repeat levels
GROUP=	Varies covariance parameters by groups
LDATA=	Specifies a data set with coefficient matrices for TYPE= LIN
NOFULLZ	Eliminates columns in **Z** corresponding to missing values
RESIDUAL	Designates a covariance structure as R-side

Table 44.16 *continued*

Option	Description
SUBJECT=	Identifies the subjects in the model
TYPE=	Specifies the covariance structure
WEIGHT=	Specifies the weights for the subjects
Mixed Model Smoothing	
KNOTINFO	Displays spline knots
KNOTMAX=	Specifies the upper limit for knot construction
KNOTMETHOD	Specifies the method for constructing knots for radial smoother and penalized B-splines
KNOTMIN=	Specifies the lower limit for knot construction
Statistical Output	
ALPHA=α	Determines the confidence level $(1-\alpha)$
CL	Requests confidence limits for predictors of random effects
G	Displays the estimated **G** matrix
GC	Displays the Cholesky root (lower) of the estimated **G** matrix
GCI	Displays the inverse Cholesky root (lower) of the estimated **G** matrix
GCORR	Displays the correlation matrix that corresponds to the estimated **G** matrix
GI	Displays the inverse of the estimated **G** matrix
SOLUTION	Displays solutions $\widehat{\boldsymbol{\gamma}}$ of the G-side random effects
V	Displays blocks of the estimated **V** matrix
VC	Displays the lower-triangular Cholesky root of blocks of the estimated **V** matrix
VCI	Displays the inverse Cholesky root of blocks of the estimated **V** matrix
VCORR	Displays the correlation matrix corresponding to blocks of the estimated **V** matrix
VI	Displays the inverse of the blocks of the estimated **V** matrix

You can specify the following *options* in the RANDOM statement after a slash (/).

ALPHA=*number*

requests that a *t*-type confidence interval with confidence level 1 – *number* be constructed for the predictors of G-side random effects in this statement. The value of *number* must be between 0 and 1; the default is 0.05. Specifying the ALPHA= option implies the CL option.

CL

requests that *t*-type confidence limits be constructed for each of the predictors of G-side random effects in this statement. The confidence level is 0.95 by default; this can be changed with the ALPHA= option. The CL option implies the SOLUTION option.

G

requests that the estimated **G** matrix be displayed for G-side random effects associated with this RANDOM statement. PROC GLIMMIX displays blanks for values that are 0.

GC

displays the lower-triangular Cholesky root of the estimated **G** matrix for G-side random effects.

GCI

displays the inverse Cholesky root of the estimated **G** matrix for G-side random effects.

GCOORD=LAST | FIRST | MEAN

determines how the GLIMMIX procedure associates coordinates for TYPE=SP() covariance structures with effect levels for G-side random effects. In these covariance structures, you specify one or more variables that identify the coordinates of a data point. The levels of classification variables, on the other hand, can occur multiple times for a particular subject. For example, in the following statements the same level of A can occur multiple times, and the associated values of x might be different:

```
proc glimmix;
   class A B;
   model y = B;
   random A / type=sp(pow)(x);
run;
```

The GCOORD=LAST option determines the coordinates for a level of the random effect from the last observation associated with the level. Similarly, the GCOORD=FIRST and GCOORD=MEAN options determine the coordinate from the first observation and from the average of the observations. Observations not used in the analysis are not considered in determining the first, last, or average coordinate. The default is GCOORD=LAST.

GCORR

displays the correlation matrix that corresponds to the estimated **G** matrix for G-side random effects.

GI

displays the inverse of the estimated **G** matrix for G-side random effects.

GROUP=*effect*

GRP=*effect*

identifies groups by which to vary the covariance parameters. Each new level of the grouping effect produces a new set of covariance parameters. Continuous variables and computed variables are permitted as group effects. PROC GLIMMIX does not sort by the values of the continuous variable; rather, it considers the data to be from a new group whenever the value of the continuous variable changes from the previous observation. Using a continuous variable decreases execution time for models with a large number of groups and also prevents the production of a large "Class Levels Information" table.

Specifying a GROUP effect can greatly increase the number of estimated covariance parameters, which can adversely affect the optimization process.

KNOTINFO

displays the number and coordinates of the knots as determined by the KNOTMETHOD= option.

KNOTMAX=*number-list*

provides upper limits for the values of random effects used in the construction of knots for TYPE=RSMOOTH. The items in *number-list* correspond to the random effects of the radial smooth. If

the KNOTMAX= option is not specified, or if the value associated with a particular random effect is set to missing, the maximum is based on the values in the data set for KNOTMETHOD=EQUAL or KNOTMETHOD=KDTREE, and is based on the values in the knot data set for KNOTMETHOD=DATA.

KNOTMETHOD=KDTREE<(*tree-options*)>

KNOTMETHOD=EQUAL<(*number-list*)>

KNOTMETHOD=DATA(*SAS-data-set*)

determines the method of constructing knots for the radial smoother fit with the TYPE=RSMOOTH covariance structure and the TYPE=PSPLINE covariance structure.

Unless you select the TYPE=RSMOOTH or TYPE=PSPLINE covariance structure, the KNOTMETHOD= option has no effect. The default for TYPE=RSMOOTH is KNOTMETHOD=KDTREE. For TYPE=PSPLINE, only equally spaced knots are used and you can use the optional *numberlist* argument of KNOTMETHOD=EQUAL to determine the number of interior knots for TYPE=PSPLINE.

Knot Construction for TYPE=RSMOOTH

PROC GLIMMIX fits a low-rank smoother, meaning that the number of knots is considerably less than the number of observations. By default, PROC GLIMMIX determines the knot locations based on the vertices of a k-d tree (Friedman, Bentley, and Finkel 1977; Cleveland and Grosse 1991). The k-d tree is a tree data structure that is useful for efficiently determining the m nearest neighbors of a point. The k-d tree also can be used to obtain a grid of points that adapts to the configuration of the data. The process starts with a hypercube that encloses the values of the random effects. The space is then partitioned recursively by splitting cells at the median of the data in the cell for the random effect. The procedure is repeated for all cells that contain more than a specified number of points, b. The value b is called the bucket size.

The k-d tree is thus a division of the data into cells such that cells representing leaf nodes contain at most b values. You control the building of the k-d tree through the BUCKET= *tree-option*. You control the construction of knots from the cell coordinates of the tree with the other *options* as follows.

BUCKET=*number*

determines the bucket size b. A larger bucket size will result in fewer knots. For k-d trees in more than one dimension, the correspondence between bucket size and number of knots is difficult to determine. It depends on the data configuration and on other suboptions. In the multivariate case, you might need to try out different bucket sizes to obtain the desired number of knots. The default value of *number* is 4 for univariate trees (a single random effect) and $\lfloor 0.1n \rfloor$ in the multidimensional case.

KNOTTYPE=*type*

specifies whether the knots are based on vertices of the tree cells or the centroid. The two possible values of *type* are VERTEX and CENTER. The default is KNOTTYPE=VERTEX. For multidimensional smoothing, such as smoothing across irregularly shaped spatial domains, the KNOTTYPE=CENTER option is useful to move knot locations away from the bounding hypercube toward the convex hull.

NEAREST
specifies that knot coordinates are the coordinates of the nearest neighbor of either the centroid or vertex of the cell, as determined by the KNOTTYPE= suboption.

TREEINFO
displays details about the construction of the *k*-d tree, such as the cell splits and the split values.

See the section "Knot Selection" on page 3221 for a detailed example of how the specification of the bucket size translates into the construction of a *k*-d tree and the spline knots.

The KNOTMETHOD=EQUAL option enables you to define a regular grid of knots. By default, PROC GLIMMIX constructs 10 knots for one-dimensional smooths and 5 knots in each dimension for smoothing in higher dimensions. You can specify a different number of knots with the optional *number-list*. Missing values in the *number-list* are replaced with the default values. A minimum of two knots in each dimension is required. For example, the following statements use a rectangular grid of 35 knots, five knots for x1 combined with seven knots for x2:

```
proc glimmix;
   model y=;
   random x1 x2 / type=rsmooth knotmethod=equal(5 7);
run;
```

When you use the NOFIT option in the PROC GLIMMIX statement, the GLIMMIX procedure computes the knots but does not fit the model. This can be useful if you want to compare knot selections with different suboptions of KNOTMETHOD=KDTREE. Suppose you want to determine the number of knots based on a particular bucket size. The following statements compute and display the knots in a bivariate smooth, constructed from nearest neighbors of the vertices of a *k*-d tree with bucket size 10:

```
proc glimmix nofit;
   model y = Latitude Longitude;
   random Latitude Longitude / type=rsmooth
                  knotmethod=kdtree(knottype=vertex
                  nearest bucket=10) knotinfo;
run;
```

You can specify a data set that contains variables whose values give the knot coordinates with the KNOTMETHOD=DATA option. The data set must contain numeric variables with the same name as the radial smoothing *random-effects*. PROC GLIMMIX uses only the unique knot coordinates in the knot data set. This option is useful to provide knot coordinates different from those that can be produced from a *k*-d tree. For example, in spatial problems where the domain is irregularly shaped, you might want to determine knots by a space-filling algorithm. The following SAS statements invoke the OPTEX procedure to compute 45 knots that uniformly cover the convex hull of the data locations (see *SAS/QC User's Guide* for details about the OPTEX procedure).

```
proc optex coding=none;
   model latitude longitude / noint;
   generate n=45 criterion=u method=m_fedorov;
   output out=knotdata;
run;
```

```
proc glimmix;
   model y = Latitude Longitude;
   random Latitude Longitude / type=rsmooth
                   knotmethod=data(knotdata);
run;
```

Knot Construction for TYPE=PSPLINE

Only evenly spaced knots are supported when you fit penalized B-splines with the GLIMMIX procedure. For the TYPE=PSPLINE covariance structure, the *number-list* argument specifies the number m of interior knots, the default is $m = 10$. Suppose that $x_{(1)}$ and $x_{(n)}$ denote the smallest and largest values, respectively. For a B-spline of degree d (De Boor 2001), the interior knots are supplemented with d exterior knots below $x_{(1)}$ and $\max\{1, d\}$ exterior knots above $x_{(n)}$. PROC GLIMMIX computes the location of these $m + d + \max\{1, d\}$ knots as follows. Let $\delta_x = (x_{(n)} - x_{(1)})/(m + 1)$, then interior knots are placed at

$$x_{(1)} + j\delta_x, \quad j = 1, \cdots, m$$

The exterior knots are also evenly spaced with step size δ_x and start at $x_{(1)} \pm 100$ times the machine epsilon. At least one interior knot is required.

KNOTMIN=*number-list*

provides lower limits for the values of random effects used in the construction of knots for TYPE=RSMOOTH. The items in *number-list* correspond to the random effects of the radial smooth. If the KNOTMIN= option is not specified, or if the value associated with a particular random effect is set to missing, the minimum is based on the values in the data set for KNOTMETHOD=EQUAL or KNOTMETHOD=KDTREE, and is based on the values in the knot data set for KNOTMETHOD=DATA.

LDATA=*SAS-data-set*

reads the coefficient matrices $\mathbf{A}_1, \cdots, \mathbf{A}_q$ for the TYPE=LIN(q) option. You can specify the LDATA= data set in a sparse or dense form. In the sparse form the data set must contain the numeric variables Parm, Row, Col, and Value. The Parm variable contains the indices $i = 1, \cdots, q$ of the $\mathbf{A}_i$ matrices. The Row and Col variables identify the position within a matrix and the Value variable contains the matrix element. Values not specified for a particular row and column are set to zero. Missing values are allowed in the Value column of the LDATA= data set; these values are also replaced by zeros. The sparse form is particularly useful if the $\mathbf{A}$ matrices have only a few nonzero elements.

In the dense form the LDATA= data set contains the numeric variables Parm and Row (with the same function as above), in addition to the numeric variables Col1–Col$q$. If you omit one or more of the Col1–Col$q$ variables from the data set, zeros are assumed for the respective rows and columns of the $\mathbf{A}$ matrix. Missing values for Col1–Col$q$ are ignored in the dense form.

The GLIMMIX procedure assumes that the matrices $\mathbf{A}_1, \cdots, \mathbf{A}_q$ are symmetric. In the sparse LDATA= form you do not need to specify off-diagonal elements in position (i, j) and (j, i). One of them is sufficient. Row-column indices are converted in both storage forms into positions in lower triangular storage. If you specify multiple values in row ($\max\{i, j\}$ and column $\min\{i, j\}$) of a particular matrix, only the last value is used. For example, assume you are specifying elements of a 4×4 matrix. The lower triangular storage of matrix $\mathbf{A}_3$ defined by

```
data ldata;
   input parm row col value;
   datalines;
3  2  1  2
3  1  2  5
;
```

is

$$\begin{bmatrix} 0 & & & \\ 5 & 0 & & \\ 0 & 0 & 0 & \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

NOFULLZ

eliminates the columns in $\mathbf{Z}$ corresponding to missing levels of random effects involving CLASS variables. By default, these columns are included in $\mathbf{Z}$. It is sufficient to specify the NOFULLZ option on any G-side RANDOM statement.

RESIDUAL

RSIDE

specifies that the random effects listed in this statement be R-side effects. You use the RESIDUAL option in the RANDOM statement if the nature of the covariance structure requires you to specify an effect. For example, if it is necessary to order the columns of the R-side AR(1) covariance structure by the time variable, you can use the RESIDUAL option as in the following statements:

```
class time id;
random time / subject=id type=ar(1) residual;
```

SOLUTION

S

requests that the solution $\widehat{\boldsymbol{\gamma}}$ for the random-effects parameters be produced, if the statement defines G-side random effects.

The numbers displayed in the Std Err Pred column of the "Solution for Random Effects" table are not the standard errors of the $\widehat{\boldsymbol{\gamma}}$ displayed in the Estimate column; rather, they are the square roots of the prediction errors $\widehat{\boldsymbol{\gamma}}_i - \boldsymbol{\gamma}_i$, where $\widehat{\boldsymbol{\gamma}}_i$ is the predictor of the ith random effect and $\boldsymbol{\gamma}_i$ is the ith random effect. In pseudo-likelihood methods that are based on linearization, these EBLUPs are the estimated best linear unbiased predictors in the linear mixed pseudo-model. In models fit by maximum likelihood by using the Laplace approximation or by using adaptive quadrature, the SOLUTION option displays the empirical Bayes estimates (EBE) of $\boldsymbol{\gamma}_i$.

SUBJECT=*effect*

SUB=*effect*

identifies the subjects in your generalized linear mixed model. Complete independence is assumed across subjects. Specifying a subject effect is equivalent to nesting all other effects in the RANDOM statement within the subject effect.

Continuous variables and computed variables are permitted with the SUBJECT= option. PROC GLIMMIX does not sort by the values of the continuous variable but considers the data to be from a new subject whenever the value of the continuous variable changes from the previous observation. Using a continuous variable can decrease execution time for models with a large number of subjects and also prevents the production of a large "Class Levels Information" table.

TYPE=*covariance-structure*

specifies the covariance structure of $\mathbf{G}$ for G-side effects and the covariance structure of $\mathbf{R}$ for R-side effects.

Although a variety of structures are available, many applications call for either simple diagonal (TYPE=VC) or unstructured covariance matrices. The TYPE=VC (variance components) option is the default structure, and it models a different variance component for each random effect. It is recommended to model unstructured covariance matrices in terms of their Cholesky parameterization (TYPE=CHOL) rather than TYPE=UN.

If you want different covariance structures in different parts of $\mathbf{G}$, you must use multiple RANDOM statements with different TYPE= options.

Valid values for *covariance-structure* are as follows. Examples are shown in Table 44.18.

The variances and covariances in the formulas that follow in the TYPE= descriptions are expressed in terms of generic random variables ξ_i and ξ_j. They represent the G-side random effects or the residual random variables for which the $\mathbf{G}$ or $\mathbf{R}$ matrices are constructed.

ANTE(1)

specifies a first-order ante-dependence structure (Kenward 1987; Patel 1991) parameterized in terms of variances and correlation parameters. If t ordered random variables $\xi_1, \cdots, \xi_t$ have a first-order ante-dependence structure, then each ξ_j, $j > 1$, is independent of all other $\xi_k, k < j$, given ξ_{j-1}. This Markovian structure is characterized by its inverse variance matrix, which is tridiagonal. Parameterizing an ANTE(1) structure for a random vector of size t requires $2t - 1$ parameters: variances $\sigma_1^2, \cdots, \sigma_t^2$ and $t - 1$ correlation parameters $\rho_1, \cdots, \rho_{t-1}$. The covariances among random variables ξ_i and ξ_j are then constructed as

$$\text{Cov}\left[\xi_i, \xi_j\right] = \sqrt{\sigma_i^2 \sigma_j^2} \prod_{k=i}^{j-1} \rho_k$$

PROC GLIMMIX constrains the correlation parameters to satisfy $|\rho_k| < 1, \forall k$. For variable-order ante-dependence models see Macchiavelli and Arnold (1994).

AR(1)

specifies a first-order autoregressive structure,

$$\text{Cov}\left[\xi_i, \xi_j\right] = \sigma^2 \rho^{|i^* - j^*|}$$

The values i^* and j^* are derived for the ith and jth observations, respectively, and are not necessarily the observation numbers. For example, in the following statements the values correspond to the class levels for the time effect of the ith and jth observation within a particular subject:

```
proc glimmix;
   class time patient;
   model y = x x*x;
   random time / sub=patient type=ar(1);
run;
```

PROC GLIMMIX imposes the constraint $|\rho| < 1$ for stationarity.

ARH(1)
specifies a heterogeneous first-order autoregressive structure,

$$\mathrm{Cov}\left[\xi_i, \xi_j\right] = \sqrt{\sigma_i^2 \sigma_j^2}\, \rho^{|i^* - j^*|}$$

with $|\rho| < 1$. This covariance structure has the same correlation pattern as the TYPE=AR(1) structure, but the variances are allowed to differ.

ARMA(1,1)
specifies the first-order autoregressive moving-average structure,

$$\mathrm{Cov}\left[\xi_i, \xi_j\right] = \begin{cases} \sigma^2 & i = j \\ \sigma^2 \gamma \rho^{|i^* - j^*| - 1} & i \neq j \end{cases}$$

Here, ρ is the autoregressive parameter, γ models a moving-average component, and σ^2 is a scale parameter. In the notation of Fuller (1976, p. 68), $\rho = \theta_1$ and

$$\gamma = \frac{(1 + b_1\theta_1)(\theta_1 + b_1)}{1 + b_1^2 + 2b_1\theta_1}$$

The example in Table 44.18 and $|b_1| < 1$ imply that

$$b_1 = \frac{\beta - \sqrt{\beta^2 - 4\alpha^2}}{2\alpha}$$

where $\alpha = \gamma - \rho$ and $\beta = 1 + \rho^2 - 2\gamma\rho$. PROC GLIMMIX imposes the constraints $|\rho| < 1$ and $|\gamma| < 1$ for stationarity, although for some values of ρ and γ in this region the resulting covariance matrix is not positive definite. When the estimated value of ρ becomes negative, the computed covariance is multiplied by $\cos(\pi d_{ij})$ to account for the negativity.

CHOL<(*q*)>
specifies an unstructured variance-covariance matrix parameterized through its Cholesky root. This parameterization ensures that the resulting variance-covariance matrix is at least positive semidefinite. If all diagonal values are nonzero, it is positive definite. For example, a 2×2 unstructured covariance matrix can be written as

$$\mathrm{Var}[\boldsymbol{\xi}] = \begin{bmatrix} \theta_1 & \theta_{12} \\ \theta_{12} & \theta_2 \end{bmatrix}$$

Without imposing constraints on the three parameters, there is no guarantee that the estimated variance matrix is positive definite. Even if θ_1 and θ_2 are nonzero, a large value for θ_{12} can lead to a negative eigenvalue of $\mathrm{Var}[\boldsymbol{\xi}]$. The Cholesky root of a positive definite matrix $\mathbf{A}$ is a lower

triangular matrix $\mathbf{C}$ such that $\mathbf{CC}' = \mathbf{A}$. The Cholesky root of the above 2×2 matrix can be written as

$$\mathbf{C} = \begin{bmatrix} \alpha_1 & 0 \\ \alpha_{12} & \alpha_2 \end{bmatrix}$$

The elements of the unstructured variance matrix are then simply $\theta_1 = \alpha_1^2$, $\theta_{12} = \alpha_1\alpha_{12}$, and $\theta_2 = \alpha_{12}^2 + \alpha_2^2$. Similar operations yield the generalization to covariance matrices of higher orders.

For example, the following statements model the covariance matrix of each subject as an unstructured matrix:

```
proc glimmix;
   class sub;
   model y = x;
   random _residual_ / subject=sub type=un;
run;
```

The next set of statements accomplishes the same, but the estimated $\mathbf{R}$ matrix is guaranteed to be nonnegative definite:

```
proc glimmix;
   class sub;
   model y = x;
   random _residual_ / subject=sub type=chol;
run;
```

The GLIMMIX procedure constrains the diagonal elements of the Cholesky root to be positive. This guarantees a unique solution when the matrix is positive definite.

The optional order parameter $q > 0$ determines how many bands below the diagonal are modeled. Elements in the lower triangular portion of $\mathbf{C}$ in bands higher than q are set to zero. If you consider the resulting covariance matrix $\mathbf{A} = \mathbf{CC}'$, then the order parameter has the effect of zeroing all off-diagonal elements that are at least q positions away from the diagonal.

Because of its good computational and statistical properties, the Cholesky root parameterization is generally recommended over a completely unstructured covariance matrix (TYPE=UN). However, it is computationally slightly more involved.

CS

specifies the compound-symmetry structure, which has constant variance and constant covariance

$$\mathrm{Cov}\left[\xi_i, \xi_j\right] = \begin{cases} \phi + \sigma & i = j \\ \sigma & i \neq j \end{cases}$$

The compound symmetry structure arises naturally with nested random effects, such as when subsampling error is nested within experimental error. The models constructed with the following two sets of GLIMMIX statements have the same marginal variance matrix, provided σ is positive:

```
proc glimmix;
   class block A;
   model y = block A;
   random block*A / type=vc;
run;


proc glimmix;
   class block A;
   model y = block A;
   random _residual_ / subject=block*A
                       type=cs;
run;
```

In the first case, the block*A random effect models the G-side experimental error. Because the distribution defaults to the normal, the $\mathbf{R}$ matrix is of form $\phi\mathbf{I}$ (see Table 44.19), and ϕ is the subsampling error variance. The marginal variance for the data from a particular experimental unit is thus $\sigma^2_{b*a}\mathbf{J} + \phi\mathbf{I}$. This matrix is of compound symmetric form.

Hierarchical random assignments or selections, such as subsampling or split-plot designs, give rise to compound symmetric covariance structures. This implies exchangeability of the observations on the subunit, leading to constant correlations between the observations. Compound symmetric structures are thus usually not appropriate for processes where correlations decline according to some metric, such as spatial and temporal processes.

Note that R-side compound-symmetry structures do not impose any constraint on σ. You can thus use an R-side TYPE=CS structure to emulate a variance-component model with unbounded estimate of the variance component.

CSH

specifies the heterogeneous compound-symmetry structure, which is an equi-correlation structure but allows for different variances

$$\mathrm{Cov}\left[\xi_i, \xi_j\right] = \begin{cases} \sqrt{\sigma_i^2\sigma_j^2} & i = j \\ \rho\sqrt{\sigma_i^2\sigma_j^2} & i \neq j \end{cases}$$

FA(*q*)

specifies the factor-analytic structure with q factors (Jennrich and Schluchter 1986). This structure is of the form $\mathbf{\Lambda}\mathbf{\Lambda}' + \mathbf{D}$, where $\mathbf{\Lambda}$ is a $t \times q$ rectangular matrix and $\mathbf{D}$ is a $t \times t$ diagonal matrix with t different parameters. When $q > 1$, the elements of $\mathbf{\Lambda}$ in its upper-right corner (that is, the elements in the ith row and jth column for $j > i$) are set to zero to fix the rotation of the structure.

FA0(*q*)

specifies a factor-analytic structure with q factors of the form $\mathrm{Var}[\boldsymbol{\xi}] = \mathbf{\Lambda}\mathbf{\Lambda}'$, where $\mathbf{\Lambda}$ is a $t \times q$ rectangular matrix and t is the dimension of $\mathbf{Y}$. When $q > 1$, $\mathbf{\Lambda}$ is a lower triangular matrix. When $q < t$—that is, when the number of factors is less than the dimension of the matrix—this structure is nonnegative definite but not of full rank. In this situation, you can use it to approximate an unstructured covariance matrix.

HF

specifies a covariance structure that satisfies the general Huynh-Feldt condition (Huynh and Feldt 1970). For a random vector with t elements, this structure has $t + 1$ positive parameters and covariances

$$\text{Cov}\left[\xi_i, \xi_j\right] = \begin{cases} \sigma_i^2 & i = j \\ 0.5(\sigma_i^2 + \sigma_j^2) - \lambda & i \neq j \end{cases}$$

A covariance matrix $\mathbf{\Sigma}$ generally satisfies the Huynh-Feldt condition if it can be written as $\mathbf{\Sigma} = \boldsymbol{\tau}\mathbf{1}' + \mathbf{1}\boldsymbol{\tau}' + \lambda\mathbf{I}$. The preceding parameterization chooses $\tau_i = 0.5(\sigma_i^2 - \lambda)$. Several simpler covariance structures give rise to covariance matrices that also satisfy the Huynh-Feldt condition. For example, TYPE=CS, TYPE=VC, and TYPE=UN(1) are nested within TYPE=HF. You can use the COVTEST statement to test the HF structure against one of these simpler structures. Note also that the HF structure is nested within an unstructured covariance matrix.

The TYPE=HF covariance structure can be sensitive to the choice of starting values and the default MIVQUE(0) starting values can be poor for this structure; you can supply your own starting values with the PARMS statement.

LIN(*q*)

specifies a general linear covariance structure with q parameters. This structure consists of a linear combination of known matrices that you input with the LDATA= option. Suppose that you want to model the covariance of a random vector of length t, and further suppose that $\mathbf{A}_1, \cdots, \mathbf{A}_q$ are symmetric $(t \times t)$ matrices constructed from the information in the LDATA= data set. Then,

$$\text{Cov}\left[\xi_i, \xi_j\right] = \sum_{k=1}^{q} \theta_k [\mathbf{A}_k]_{ij}$$

where $[\mathbf{A}_k]_{ij}$ denotes the element in row i, column j of matrix $\mathbf{A}_k$.

Linear structures are very flexible and general. You need to exercise caution to ensure that the variance matrix is positive definite. Note that PROC GLIMMIX does not impose boundary constraints on the parameters $\theta_1, \cdots, \theta_k$ of a general linear covariance structure. For example, if classification variable A has 6 levels, the following statements fit a variance component structure for the random effect without boundary constraints:

```
data ldata;
   retain parm 1 value 1;
   do row=1 to 6; col=row; output; end;
run;

proc glimmix data=MyData;
   class A B;
   model Y = B;
   random A / type=lin(1) ldata=ldata;
run;
```

PSPLINE< (*options*) >

requests that PROC GLIMMIX form a B-spline basis and fits a penalized B-spline (P-spline, Eilers and Marx 1996) with random spline coefficients. This covariance structure is available only for G-side random effects and only a single continuous random effect can be specified with TYPE=PSPLINE. As for TYPE=RSMOOTH, PROC GLIMMIX forms a modified $\mathbf{Z}$ matrix and fits a mixed model in which the random variables associated with the columns of $\mathbf{Z}$ are independent with a common variance. The $\mathbf{Z}$ matrix is constructed as follows.

Denote as $\widetilde{\mathbf{Z}}$ the $(n \times K)$ matrix of B-splines of degree d and denote as $\mathbf{D}_r$ the $(K - r \times K)$ matrix of rth-order differences. For example, for $K = 5$,

$$\mathbf{D}_1 = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

$$\mathbf{D}_2 = \begin{bmatrix} 1 & -2 & 1 & 0 & 0 \\ 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 1 & -2 & 1 \end{bmatrix}$$

$$\mathbf{D}_3 = \begin{bmatrix} 1 & -3 & 3 & -1 & 0 \\ 0 & 1 & -3 & 3 & -1 \end{bmatrix}$$

Then, the $\mathbf{Z}$ matrix used in fitting the mixed model is the $(n \times K - r)$ matrix

$$\mathbf{Z} = \widetilde{\mathbf{Z}}(\mathbf{D}_r'\mathbf{D}_r)^{-}\mathbf{D}_r'$$

The construction of the B-spline knots is controlled with the KNOTMETHOD= EQUAL(m) option and the DEGREE=d suboption of TYPE=PSPLINE. The total number of knots equals the number m of equally spaced interior knots plus d knots at the low end and $\max\{1, d\}$ knots at the high end. The number of columns in the B-spline basis equals $K = m + d + 1$. By default, the interior knots exclude the minimum and maximum of the random-effect values and are based on $m - 1$ equally spaced intervals. Suppose $x_{(1)}$ and $x_{(n)}$ are the smallest and largest random-effect values; then interior knots are placed at

$$x_{(1)} + j(x_{(n)} - x_{(1)})/(m + 1), \quad j = 1, \cdots, m$$

In addition, d evenly spaced exterior knots are placed below $x_{(1)}$ and $\max\{d, 1\}$ exterior knots are placed above $x_{(m)}$. The exterior knots are evenly spaced and start at $x_{(1)} \pm 100$ times the machine epsilon. For example, based on the defaults $d = 3$, $r = 3$, the following statements lead to 26 total knots and 21 columns in $\mathbf{Z}$, $m = 20$, $K = m + d + 1 = 24$, $K - r = 21$:

```
proc glimmix;
   model y = x;
   random x / type=pspline knotmethod=equal(20);
run;
```

Details about the computation and properties of B-splines can be found in De Boor (2001). You can extend or limit the range of the knots with the KNOTMIN= and KNOTMAX= options. Table 44.17 lists some of the parameters that control this covariance type and their relationships.

Table 44.17 P-Spline Parameters

Parameter	Description
d	Degree of B-spline, default $d = 3$
r	Order of differencing in construction of $\mathbf{D}_r$, default $r = 3$
m	Number of interior knots, default $m = 10$
$m + d + \max\{1, d\}$	Total number of knots
$K = m + d + 1$	Number of columns in B-spline basis
$K - r$	Number of columns in $\mathbf{Z}$

You can specify the following *options* for TYPE=PSPLINE:

DEGREE=*d* specifies the degree of the B-spline. The default is $d = 3$.

DIFFORDER=*r* specifies the order of the differencing matrix $\mathbf{D}_r$. The default and maximum is $r = 3$.

RSMOOTH< (*m* | **NOLOG) >**

specifies a radial smoother covariance structure for G-side random effects. This results in an approximate low-rank thin-plate spline where the smoothing parameter is obtained by the estimation method selected with the METHOD= option of the PROC GLIMMIX statement. The smoother is based on the automatic smoother in Ruppert, Wand, and Carroll (2003, Chapter 13.4–13.5), but with a different method of selecting the spline knots. See the section "Radial Smoothing Based on Mixed Models" on page 3220 for further details about the construction of the smoother and the knot selection.

Radial smoothing is possible in one or more dimensions. A univariate smoother is obtained with a single random effect, while multiple random effects in a RANDOM statement yield a multivariate smoother. Only continuous random effects are permitted with this covariance structure. If n_r denotes the number of continuous random effects in the RANDOM statement, then the covariance structure of the random effects $\boldsymbol{\gamma}$ is determined as follows. Suppose that $\mathbf{z}_i$ denotes the vector of random effects for the ith observation. Let $\boldsymbol{\tau}_k$ denote the $(n_r \times 1)$ vector of knot coordinates, $k = 1, \cdots, K$, and K is the total number of knots. The Euclidean distance between the knots is computed as

$$d_{kp} = ||\boldsymbol{\tau}_k - \boldsymbol{\tau}_p|| = \sqrt{\sum_{j=1}^{n_r} (\tau_{jk} - \tau_{jp})^2}$$

and the distance between knots and effects is computed as

$$h_{ik} = ||\mathbf{z}_i - \boldsymbol{\tau}_k|| = \sqrt{\sum_{j=1}^{n_r} (z_{ij} - \tau_{jk})^2}$$

The $\mathbf{Z}$ matrix for the GLMM is constructed as

$$\mathbf{Z} = \widetilde{\mathbf{Z}}\boldsymbol{\Omega}^{-1/2}$$

where the $(n \times K)$ matrix $\widetilde{\mathbf{Z}}$ has typical element

$$[\widetilde{\mathbf{Z}}]_{ik} = \begin{cases} h_{ik}^{p} & n_r \text{ odd} \\ h_{ik}^{p} \log\{h_{ik}\} & n_r \text{ even} \end{cases}$$

and the $(K \times K)$ matrix $\mathbf{\Omega}$ has typical element

$$[\mathbf{\Omega}]_{kp} = \begin{cases} d_{kp}^{p} & n_r \text{ odd} \\ d_{kp}^{p} \log\{d_{kp}\} & n_r \text{ even} \end{cases}$$

The exponent in these expressions equals $p = 2m - n_r$, where the optional value m corresponds to the derivative penalized in the thin-plate spline. A larger value of m will yield a smoother fit. The GLIMMIX procedure requires $p > 0$ and chooses by default $m = 2$ if $n_r < 3$ and $m = n_r/2 + 1$ otherwise. The NOLOG option removes the $\log\{h_{ik}\}$ and $\log\{d_{kp}\}$ terms from the computation of the $\widetilde{\mathbf{Z}}$ and $\mathbf{\Omega}$ matrices when n_r is even; this yields invariance under rescaling of the coordinates.

Finally, the components of $\boldsymbol{\gamma}$ are assumed to have equal variance σ_r^2. The "smoothing parameter" λ of the low-rank spline is related to the variance components in the model, $\lambda^2 = f(\phi, \sigma_r^2)$. See Ruppert, Wand, and Carroll (2003) for details. If the conditional distribution does not provide a scale parameter ϕ, you can add a single R-side residual parameter.

The knot selection is controlled with the KNOTMETHOD= option. The GLIMMIX procedure selects knots automatically based on the vertices of a k-d tree or reads knots from a data set that you supply. See the section "Radial Smoothing Based on Mixed Models" on page 3220 for further details on radial smoothing in the GLIMMIX procedure and its connection to a mixed model formulation.

SIMPLE

is an alias for TYPE=VC.

SP(EXP)(*c-list*)

models an exponential spatial or temporal covariance structure, where the covariance between two observations depends on a distance metric d_{ij}. The *c-list* contains the names of the numeric variables used as coordinates to determine distance. For a stochastic process in R^k, there are k elements in *c-list*. If the $(k \times 1)$ vectors of coordinates for observations i and j are $\mathbf{c}_i$ and $\mathbf{c}_j$, then PROC GLIMMIX computes the Euclidean distance

$$d_{ij} = ||\mathbf{c}_i - \mathbf{c}_j|| = \sqrt{\sum_{m=1}^{k} (c_{mi} - c_{mj})^2}$$

The covariance between two observations is then

$$\text{Cov}\left[\xi_i, \xi_j\right] = \sigma^2 \exp\{-d_{ij}/\alpha\}$$

The parameter α is *not* what is commonly referred to as the range parameter in geostatistical applications. The practical range of a (second-order stationary) spatial process is the distance $d^{(p)}$ at which the correlations fall below 0.05. For the SP(EXP) structure, this distance is $d^{(p)} = 3\alpha$. PROC GLIMMIX constrains α to be positive.

SP(GAU)(*c-list*)
models a Gaussian covariance structure,

$$\text{Cov}\left[\xi_i, \xi_j\right] = \sigma^2 \exp\{-d_{ij}^2/\alpha^2\}$$

See TYPE=SP(EXP) for the computation of the distance d_{ij}. The parameter α is related to the range of the process as follows. If the practical range $d^{(p)}$ is defined as the distance at which the correlations fall below 0.05, then $d^{(p)} = \sqrt{3}\alpha$. PROC GLIMMIX constrains α to be positive. See TYPE=SP(EXP) for the computation of the distance d_{ij} from the variables specified in *c-list*.

SP(MAT)(*c-list*)
models a covariance structure in the Matérn class of covariance functions (Matérn 1986). The covariance is expressed in the parameterization of Handcock and Stein (1993); Handcock and Wallis (1994); it can be written as

$$\text{Cov}\left[\xi_i, \xi_j\right] = \sigma^2 \frac{1}{\Gamma(\nu)} \left(\frac{d_{ij}\sqrt{\nu}}{\rho}\right)^{\nu} 2K_\nu \left(\frac{2d_{ij}\sqrt{\nu}}{\rho}\right)$$

The function K_ν is the modified Bessel function of the second kind of (real) order $\nu > 0$. The smoothness (continuity) of a stochastic process with covariance function in the Matérn class increases with ν. This class thus enables data-driven estimation of the smoothness properties of the process. The covariance is identical to the exponential model for $\nu = 0.5$ (TYPE=SP(EXP)(*c-list*)), while for $\nu = 1$ the model advocated by Whittle (1954) results. As $\nu \to \infty$, the model approaches the Gaussian covariance structure (TYPE=SP(GAU)(*c-list*)).

Note that the MIXED procedure offers covariance structures in the Matérn class in two parameterizations, TYPE=SP(MATERN) and TYPE=SP(MATHSW). The TYPE=SP(MAT) in the GLIMMIX procedure is equivalent to TYPE=SP(MATHSW) in the MIXED procedure.

Computation of the function K_ν and its derivatives is numerically demanding; fitting models with Matérn covariance structures can be time-consuming. Good starting values are essential.

SP(POW)(*c-list*)
models a power covariance structure,

$$\text{Cov}\left[\xi_i, \xi_j\right] = \sigma^2 \rho^{d_{ij}}$$

where $\rho \geq 0$. This is a reparameterization of the exponential structure, TYPE=SP(EXP). Specifically, $\log\{\rho\} = -1/\alpha$. See TYPE=SP(EXP) for the computation of the distance d_{ij} from the variables specified in *c-list*. When the estimated value of ρ becomes negative, the computed covariance is multiplied by $\cos(\pi d_{ij})$ to account for the negativity.

SP(POWA)(*c-list*)
models an anisotropic power covariance structure in k dimensions, provided that the coordinate list *c-list* has k elements. If c_{im} denotes the coordinate for the ith observation of the mth variable in *c-list*, the covariance between two observations is given by

$$\text{Cov}\left[\xi_i, \xi_j\right] = \sigma^2 \rho_1^{|c_{i1}-c_{j1}|} \rho_2^{|c_{i2}-c_{j2}|} \ldots \rho_k^{|c_{ik}-c_{jk}|}$$

Note that for $k = 1$, TYPE=SP(POWA) is equivalent to TYPE=SP(POW), which is itself a reparameterization of TYPE=SP(EXP). When the estimated value of ρ_m becomes negative, the computed covariance is multiplied by $\cos(\pi|c_{im} - c_{jm}|)$ to account for the negativity.

SP(SPH)(*c-list*)
models a spherical covariance structure,

$$\mathrm{Cov}\left[\xi_i, \xi_j\right] = \begin{cases} \sigma^2 \left\{1 - \frac{3d_{ij}}{2\alpha} + \frac{1}{2}\left(\frac{d_{ij}}{\alpha}\right)^3\right\} & d_{ij} \leq \alpha \\ 0 & d_{ij} > \alpha \end{cases}$$

The spherical covariance structure has a true range parameter. The covariances between observations are exactly zero when their distance exceeds α. See TYPE=SP(EXP) for the computation of the distance d_{ij} from the variables specified in *c-list*.

TOEP
models a Toeplitz covariance structure. This structure can be viewed as an autoregressive structure with order equal to the dimension of the matrix,

$$\mathrm{Cov}\left[\xi_i, \xi_j\right] = \begin{cases} \sigma^2 & i = j \\ \sigma_{|i-j|} & i \neq j \end{cases}$$

TOEP(*q*)
specifies a banded Toeplitz structure,

$$\mathrm{Cov}\left[\xi_i, \xi_j\right] = \begin{cases} \sigma^2 & i = j \\ \sigma_{|i-j|} & |i-j| < q \end{cases}$$

This can be viewed as a moving-average structure with order equal to $q - 1$. The specification TYPE=TOEP(1) is the same as $\sigma^2\mathbf{I}$, and it can be useful for specifying the same variance component for several effects.

TOEPH< (*q*) >
models a Toeplitz covariance structure. The correlations of this structure are banded as the TOEP or TOEP(q) structures, but the variances are allowed to vary:

$$\mathrm{Cov}\left[\xi_i, \xi_j\right] = \begin{cases} \sigma_i^2 & i = j \\ \rho_{|i-j|}\sqrt{\sigma_i^2\sigma_j^2} & i \neq j \end{cases}$$

The correlation parameters satisfy $|\rho_{|i-j|}| < 1$. If you specify the optional value q, the correlation parameters with $|i - j| \geq q$ are set to zero, creating a banded correlation structure. The specification TYPE=TOEPH(1) results in a diagonal covariance matrix with heterogeneous variances.

UN< (*q*) >
specifies a completely general (unstructured) covariance matrix parameterized directly in terms of variances and covariances,

$$\mathrm{Cov}\left[\xi_i, \xi_j\right] = \sigma_{ij}$$

The variances are constrained to be nonnegative, and the covariances are unconstrained. This structure is not constrained to be nonnegative definite in order to avoid nonlinear constraints; however, you can use the TYPE=CHOL structure if you want this constraint to be imposed by a Cholesky factorization. If you specify the order parameter q, then PROC GLIMMIX estimates only the first q bands of the matrix, setting elements in all higher bands equal to 0.

UNR<(*q*)>

specifies a completely general (unstructured) covariance matrix parameterized in terms of variances and correlations,

$$\mathrm{Cov}\left[\xi_i, \xi_j\right] = \sigma_i \sigma_j \rho_{ij}$$

where σ_i denotes the standard deviation and the correlation ρ_{ij} is zero when $i = j$ and when $|i - j| \geq q$, provided the order parameter q is given. This structure fits the same model as the TYPE=UN(q) option, but with a different parameterization. The ith variance parameter is σ_i^2. The parameter ρ_{ij} is the correlation between the ith and jth measurements; it satisfies $|\rho_{ij}| < 1$. If you specify the order parameter q, then PROC GLIMMIX estimates only the first q bands of the matrix, setting all higher bands equal to zero.

VC

specifies standard variance components and is the default structure for both G-side and R-side covariance structures. In a G-side covariance structure, a distinct variance component is assigned to each effect. In an R-side structure TYPE=VC is usually used only to add overdispersion effects or with the GROUP= option to specify a heterogeneous variance model.

Table 44.18 Covariance Structure Examples

Description	Structure	Example
Variance Components	VC (default)	$\begin{bmatrix} \sigma_B^2 & 0 & 0 & 0 \\ 0 & \sigma_B^2 & 0 & 0 \\ 0 & 0 & \sigma_{AB}^2 & 0 \\ 0 & 0 & 0 & \sigma_{AB}^2 \end{bmatrix}$
Compound Symmetry	CS	$\begin{bmatrix} \sigma+\phi & \sigma & \sigma & \sigma \\ \sigma & \sigma+\phi & \sigma & \sigma \\ \sigma & \sigma & \sigma+\phi & \sigma \\ \sigma & \sigma & \sigma & \sigma+\phi \end{bmatrix}$
Heterogeneous CS	CSH	$\begin{bmatrix} \sigma_1^2 & \sigma_1\sigma_2\rho & \sigma_1\sigma_3\rho & \sigma_1\sigma_4\rho \\ \sigma_2\sigma_1\rho & \sigma_2^2 & \sigma_2\sigma_3\rho & \sigma_2\sigma_4\rho \\ \sigma_3\sigma_1\rho & \sigma_3\sigma_2\rho & \sigma_3^2 & \sigma_3\sigma_4\rho \\ \sigma_4\sigma_1\rho & \sigma_4\sigma_2\rho & \sigma_4\sigma_3\rho & \sigma_4^2 \end{bmatrix}$
First-Order Autoregressive	AR(1)	$\sigma^2 \begin{bmatrix} 1 & \rho & \rho^2 & \rho^3 \\ \rho & 1 & \rho & \rho^2 \\ \rho^2 & \rho & 1 & \rho \\ \rho^3 & \rho^2 & \rho & 1 \end{bmatrix}$
Heterogeneous AR(1)	ARH(1)	$\begin{bmatrix} \sigma_1^2 & \sigma_1\sigma_2\rho & \sigma_1\sigma_3\rho^2 & \sigma_1\sigma_4\rho^3 \\ \sigma_2\sigma_1\rho & \sigma_2^2 & \sigma_2\sigma_3\rho & \sigma_2\sigma_4\rho^2 \\ \sigma_3\sigma_1\rho^2 & \sigma_3\sigma_2\rho & \sigma_3^2 & \sigma_3\sigma_4\rho \\ \sigma_4\sigma_1\rho^3 & \sigma_4\sigma_2\rho & \sigma_4\sigma_3\rho & \sigma_4^2 \end{bmatrix}$
Unstructured	UN	$\begin{bmatrix} \sigma_1^2 & \sigma_{21} & \sigma_{31} & \sigma_{41} \\ \sigma_{21} & \sigma_2^2 & \sigma_{32} & \sigma_{42} \\ \sigma_{31} & \sigma_{32} & \sigma_3^2 & \sigma_{43} \\ \sigma_{41} & \sigma_{42} & \sigma_{43} & \sigma_4^2 \end{bmatrix}$

Table 44.18 *continued*

Description	Structure	Example
Banded Main Diagonal	UN(1)	$\begin{bmatrix} \sigma_1^2 & 0 & 0 & 0 \\ 0 & \sigma_2^2 & 0 & 0 \\ 0 & 0 & \sigma_3^2 & 0 \\ 0 & 0 & 0 & \sigma_4^2 \end{bmatrix}$
Unstructured Correlations	UNR	$\begin{bmatrix} \sigma_1^2 & \sigma_1\sigma_2\rho_{21} & \sigma_1\sigma_3\rho_{31} & \sigma_1\sigma_4\rho_{41} \\ \sigma_2\sigma_1\rho_{21} & \sigma_2^2 & \sigma_2\sigma_3\rho_{32} & \sigma_2\sigma_4\rho_{42} \\ \sigma_3\sigma_1\rho_{31} & \sigma_3\sigma_2\rho_{32} & \sigma_3^2 & \sigma_3\sigma_4\rho_{43} \\ \sigma_4\sigma_1\rho_{41} & \sigma_4\sigma_2\rho_{42} & \sigma_4\sigma_3\rho_{43} & \sigma_4^2 \end{bmatrix}$
Toeplitz	TOEP	$\begin{bmatrix} \sigma^2 & \sigma_1 & \sigma_2 & \sigma_3 \\ \sigma_1 & \sigma^2 & \sigma_1 & \sigma_2 \\ \sigma_2 & \sigma_1 & \sigma^2 & \sigma_1 \\ \sigma_3 & \sigma_2 & \sigma_1 & \sigma^2 \end{bmatrix}$
Toeplitz with Two Bands	TOEP(2)	$\begin{bmatrix} \sigma^2 & \sigma_1 & 0 & 0 \\ \sigma_1 & \sigma^2 & \sigma_1 & 0 \\ 0 & \sigma_1 & \sigma^2 & \sigma_1 \\ 0 & 0 & \sigma_1 & \sigma^2 \end{bmatrix}$
Heterogeneous Toeplitz	TOEPH	$\begin{bmatrix} \sigma_1^2 & \sigma_1\sigma_2\rho_1 & \sigma_1\sigma_3\rho_2 & \sigma_1\sigma_4\rho_3 \\ \sigma_2\sigma_1\rho_1 & \sigma_2^2 & \sigma_2\sigma_3\rho_1 & \sigma_2\sigma_4\rho_2 \\ \sigma_3\sigma_1\rho_2 & \sigma_3\sigma_2\rho_1 & \sigma_3^2 & \sigma_3\sigma_4\rho_1 \\ \sigma_4\sigma_1\rho_3 & \sigma_4\sigma_2\rho_2 & \sigma_4\sigma_3\rho_1 & \sigma_4^2 \end{bmatrix}$
Spatial Power	SP(POW)(*c-list*)	$\sigma^2 \begin{bmatrix} 1 & \rho^{d_{12}} & \rho^{d_{13}} & \rho^{d_{14}} \\ \rho^{d_{21}} & 1 & \rho^{d_{23}} & \rho^{d_{24}} \\ \rho^{d_{31}} & \rho^{d_{32}} & 1 & \rho^{d_{34}} \\ \rho^{d_{41}} & \rho^{d_{42}} & \rho^{d_{43}} & 1 \end{bmatrix}$
First-Order Autoregressive Moving-Average	ARMA(1,1)	$\sigma^2 \begin{bmatrix} 1 & \gamma & \gamma\rho & \gamma\rho^2 \\ \gamma & 1 & \gamma & \gamma\rho \\ \gamma\rho & \gamma & 1 & \gamma \\ \gamma\rho^2 & \gamma\rho & \gamma & 1 \end{bmatrix}$
First-Order Factor Analytic	FA(1)	$\begin{bmatrix} \lambda_1^2 + d_1 & \lambda_1\lambda_2 & \lambda_1\lambda_3 & \lambda_1\lambda_4 \\ \lambda_2\lambda_1 & \lambda_2^2 + d_2 & \lambda_2\lambda_3 & \lambda_2\lambda_4 \\ \lambda_3\lambda_1 & \lambda_3\lambda_2 & \lambda_3^2 + d_3 & \lambda_3\lambda_4 \\ \lambda_4\lambda_1 & \lambda_4\lambda_2 & \lambda_4\lambda_3 & \lambda_4^2 + d_4 \end{bmatrix}$
Huynh-Feldt	HF	$\begin{bmatrix} \sigma_1^2 & \frac{\sigma_1^2+\sigma_2^2}{2} - \lambda & \frac{\sigma_1^2+\sigma_3^2}{2} - \lambda \\ \frac{\sigma_2^2+\sigma_1^2}{2} - \lambda & \sigma_2^2 & \frac{\sigma_2^2+\sigma_3^2}{2} - \lambda \\ \frac{\sigma_3^2+\sigma_1^2}{2} - \lambda & \frac{\sigma_3^2+\sigma_2^2}{2} - \lambda & \sigma_3^2 \end{bmatrix}$
First-Order Ante-dependence	ANTE(1)	$\begin{bmatrix} \sigma_1^2 & \sigma_1\sigma_2\rho_1 & \sigma_1\sigma_3\rho_1\rho_2 \\ \sigma_2\sigma_1\rho_1 & \sigma_2^2 & \sigma_2\sigma_3\rho_2 \\ \sigma_3\sigma_1\rho_2\rho_1 & \sigma_3\sigma_2\rho_2 & \sigma_3^2 \end{bmatrix}$

V< =*value-list* >
requests that blocks of the estimated marginal variance-covariance matrix $\mathbf{V}(\widehat{\boldsymbol{\theta}})$ be displayed in generalized linear mixed models. This matrix is based on the last linearization as described in the section "The Pseudo-model" on page 3188. You can use the *value-list* to select the subjects for which the matrix is displayed. If *value-list* is not specified, the $\mathbf{V}$ matrix for the first subject is chosen.

Note that the *value-list* refers to subjects as the processing units in the "Dimensions" table. For example, the following statements request that the estimated marginal variance matrix for the second subject be displayed:

```
proc glimmix;
   class A B;
   model y = B;
   random int / subject=A;
   random int / subject=A*B v=2;
run;
```

The subject effect for processing in this case is the A effect, because it is contained in the A*B interaction. If there is only a single subject as per the "Dimensions" table, then the V option displays an $(n \times n)$ matrix.

See the section "Processing by Subjects" on page 3218 for how the GLIMMIX procedure determines the number of subjects in the "Dimensions" table.

The GLIMMIX procedure displays blanks for values that are 0.

VC< =*value-list* >
displays the lower-triangular Cholesky root of the blocks of the estimated $\mathbf{V}(\widehat{\boldsymbol{\theta}})$ matrix. See the V option for the specification of *value-list*.

VCI< =*value-list* >
displays the inverse Cholesky root of the blocks of the estimated $\mathbf{V}(\widehat{\boldsymbol{\theta}})$ matrix. See the V option for the specification of *value-list*.

VCORR< =*value-list* >
displays the correlation matrix corresponding to the blocks of the estimated $\mathbf{V}(\widehat{\boldsymbol{\theta}})$ matrix. See the V option for the specification of *value-list*.

VI< =*value-list* >
displays the inverse of the blocks of the estimated $\mathbf{V}(\widehat{\boldsymbol{\theta}})$ matrix. See the V option for the specification of *value-list*.

WEIGHT< =*variable* >
WT< =*variable* >
specifies a variable to be used as the weight for the units at the current level in a weighted multilevel model. If a weight variable is not specified in the WEIGHT option, a weight of 1 is used. For details on the use of weights in multilevel models, see the section "Pseudo-likelihood Estimation for Weighted Multilevel Models" on page 3200.

SLICE Statement

SLICE *model-effect* < / *options* > **;**

The SLICE statement provides a general mechanism for performing a partitioned analysis of the LS-means for an interaction. This analysis is also known as an analysis of simple effects (Winer 1971).

The SLICE statement uses most of the options of the LSMEANS statement that are summarized in Table 44.8. The options SLICEDIFF=, SLICEDIFFTYPE=, and ODDS do not apply to the SLICE statement; in the SLICE statement, the relevant options for SLICEDIFF= and SLICEDIFFTYPE= are the SLICEBY= and the DIFF= options, respectively.

For details about the syntax of the SLICE statement, see the section "SLICE Statement" on page 505 in Chapter 19, "Shared Concepts and Topics."

STORE Statement

STORE < **OUT=** >*item-store-name* < / **LABEL=**'*label*' > **;**

The STORE statement requests that the procedure save the context and results of the statistical analysis. The resulting item store has a binary file format that cannot be modified. The contents of the item store can be processed with the PLM procedure.

For details about the syntax of the STORE statement, see the section "STORE Statement" on page 508 in Chapter 19, "Shared Concepts and Topics."

WEIGHT Statement

WEIGHT *variable* **;**

The WEIGHT statement replaces $\mathbf{R}$ with $\mathbf{W}^{-1/2}\mathbf{R}\mathbf{W}^{-1/2}$, where $\mathbf{W}$ is a diagonal matrix containing the weights. Observations with nonpositive or missing weights are not included in the resulting PROC GLIMMIX analysis. If a WEIGHT statement is not included, all observations used in the analysis are assigned a weight of 1.

Programming Statements

This section lists the programming statements available in PROC GLIMMIX to compute various aspects of the generalized linear mixed model or output quantities. For example, you can compute model effects, weights, frequency, subject, group, and other variables. You can use programming statements to define the mean and variance functions. This section also documents the differences between programming statements in PROC GLIMMIX and programming statements in the SAS DATA step. The syntax of programming statements used in PROC GLIMMIX is identical to that used in the NLMIXED procedure (see Chapter 70, "The NLMIXED Procedure," and the MODEL procedure (see the *SAS/ETS User's Guide*). Most of the programming statements that can be used in the DATA step can also be used in the GLIMMIX procedure. See *SAS Statements: Reference* for a description of SAS programming statements. The following are valid statements:

ABORT;
ARRAY *arrayname* <**[** *dimensions* **]**> <**$**> <*variables-and-constants*>**;**
CALL *name* <**(***expression* <**,** *expression . . .* >**)**>**;**
DELETE;
DO <*variable* **=** *expression* <**TO** *expression*> <**BY** *expression*>>
    <**,** *expression* <**TO** *expression*> <**BY** *expression*>> . . .
    <**WHILE** *expression*> <**UNTIL** *expression*>**;**
END;
GOTO *statement-label*;
IF *expression*;
IF *expression* **THEN** *program-statement*;
            ELSE *program-statement*;
variable **=** *expression*;
variable **+** *expression*;
LINK *statement-label*;
PUT <*variable*> <**=**> . . .**;**
RETURN;
SELECT <**(***expression***)**>**;**
STOP;
SUBSTR(*variable***,** *index***,** *length***)=** *expression*;
WHEN (*expression***)***program-statement*;
    OTHERWISE *program-statement*;

For the most part, the SAS programming statements work the same as they do in the SAS DATA step, as documented in *SAS Language Reference: Concepts*. However, there are several differences:

- The ABORT statement does not allow any arguments.
- The DO statement does not allow a character index variable. Thus

```
do i = 1,2,3;
```

  is supported; however, the following statement is not supported:

```
do i = 'A','B','C';
```

- The LAG function is not supported with PROC GLIMMIX.
- The PUT statement, used mostly for program debugging in PROC GLIMMIX, supports only some of the features of the DATA step PUT statement, and it has some features not available with the DATA step PUT statement:
  - The PROC GLIMMIX PUT statement does not support line pointers, factored lists, iteration factors, overprinting, _INFILE_, the colon (:) format modifier, or "$".
  - The PROC GLIMMIX PUT statement does support expressions, but the expression must be enclosed in parentheses. For example, the following statement displays the square root of x:

```
put (sqrt(x));
```

  - The PROC GLIMMIX PUT statement supports the item _PDV_ to display a formatted listing of all variables in the program. For example:

```
put _pdv_;
```

- The WHEN and OTHERWISE statements enable you to specify more than one target statement. That is, DO/END groups are not necessary for multiple statement WHENs. For example, the following syntax is valid:

```
select;
   when (exp1) stmt1;
               stmt2;
   when (exp2) stmt3;
               stmt4;
end;
```

The LINK statement is used in a program to jump immediately to the label *statement_label* and to continue program execution at that point. It is not used to specify a user-defined link function.

When coding your programming statements, you should avoid defining variables that begin with an underscore (_), because they might conflict with internal variables created by PROC GLIMMIX.

User-Defined Link or Variance Function

Implied Variance Functions

While link functions are not unique for each distribution (see Table 44.12 for the default link functions), the distribution does determine the variance function $a(\mu)$. This function expresses the variance of an observation as a function of the mean, apart from weights, frequencies, and additional scale parameters. The implied variance functions $a(\mu)$ of the GLIMMIX procedure are shown in Table 44.19 for the supported distributions. For the binomial distribution, n denotes the number of trials in the *events/trials* syntax. For the negative binomial distribution, k denotes the scale parameter. The multiplicative scale parameter ϕ is not included for the other distributions. The last column of the table indicates whether ϕ has a value equal to 1.0 for the particular distribution.

Table 44.19 Variance Functions in PROC GLIMMIX

		Variance function	
DIST=	**Distribution**	$a(\mu)$	$\phi \equiv 1$
BETA	beta	$\mu(1-\mu)/(1+\phi)$	No
BINARY	binary	$\mu(1-\mu)$	Yes
BINOMIAL | BIN | B	binomial	$\mu(1-\mu)/n$	Yes
EXPONENTIAL | EXPO	exponential	μ^2	Yes
GAMMA | GAM	gamma	μ^2	No
GAUSSIAN | G | NORMAL | N	normal	1	No
GEOMETRIC | GEOM	geometric	$\mu+\mu^2$	Yes
INVGAUSS | IGAUSSIAN | IG	inverse Gaussian	μ^3	No
LOGNORMAL | LOGN	lognormal	1	No
NEGBINOMIAL | NEGBIN | NB	negative binomial	$\mu+k\mu^2$	Yes

Table 44.19 *continued*

DIST=	Distribution	Variance function $a(\mu)$	$\phi \equiv 1$
POISSON | POI | P	Poisson	μ	Yes
TCENTRAL | TDIST | T	t	$\nu/(\nu-2)$	No

To change the variance function, you can use SAS programming statements and the predefined automatic variables, as outlined in the following section. Your definition of a variance function will override the DIST= option and its implied variance function. This has the following implication for parameter estimation with the GLIMMIX procedure. When a user-defined link is available, the distribution of the data is determined from the DIST= option, or the respective default for the type of response. In a GLM, for example, this enables maximum likelihood estimation. If a user-defined variance function is provided, the DIST= option is not honored and the distribution of the data is assumed unknown. In a GLM framework, only quasi-likelihood estimation is then available to estimate the model parameters.

Automatic Variables

To specify your own link or variance function you can use SAS programming statements and draw on the following automatic variables:

LINP is the current value of the linear predictor. It equals either $\widehat{\eta} = \mathbf{x}'\widehat{\boldsymbol{\beta}} + \mathbf{z}'\widehat{\boldsymbol{\gamma}} + o$ or $\widehat{\eta} = \mathbf{x}'\widehat{\boldsymbol{\beta}} + o$, where o is the value of the offset variable, or 0 if no offset is specified. The estimated random effects solutions $\widehat{\boldsymbol{\gamma}}$ are used in the calculation of the linear predictor during the model fitting phase, if a linearization expands about the current values of $\boldsymbol{\gamma}$. During the computation of output statistics, the EBLUPs are used if statistics depend on them. For example, the following statements add the variable p to the output data set glimmixout:

```
proc glimmix;
   model y = x / dist=binary;
   random int / subject=b;
   p = 1/(1+exp(-_linp_);
   output out=glimmixout;
   id p;
run;
```

Because no output statistics are requested in the OUTPUT statement that depend on the random-effects solutions (BLUPs, EBEs), the value of _LINP_ in this example equals $\mathbf{x}'\widehat{\boldsymbol{\beta}}$. On the contrary, the following statements also request conditional residuals on the logistic scale:

```
proc glimmix;
   model y = x / dist=binary;
   random int / subject=b;
   p = 1/(1+exp(-_linp_);
   output out=glimmixout resid(blup)=r;
   id p;
run;
```

	The value of _LINP_ when computing the variable p is $\mathbf{x}'\widehat{\boldsymbol{\beta}}+\mathbf{z}'\widehat{\boldsymbol{\gamma}}$. To ensure that computed statistics are formed from $\mathbf{x}'\widehat{\boldsymbol{\beta}}$ and $\mathbf{z}'\widehat{\boldsymbol{\gamma}}$ terms as needed, it is recommended that you use the automatic variables _XBETA_ and _ZGAMMA_ instead of _LINP_.
_MU_	expresses the mean of an observation as a function of the linear predictor, $\widehat{\mu} = g^{-1}(\widehat{\eta})$.
_N_	is the observation number in the sequence of the data read.
VARIANCE	is the estimate of the variance function, $a(\widehat{\mu})$.
XBETA	equals $\mathbf{x}'\widehat{\boldsymbol{\beta}}$.
ZGAMMA	equals $\mathbf{z}'\widehat{\boldsymbol{\gamma}}$.

The automatic variable _N_ is incremented whenever the procedure reads an observation from the data set. Observations that are not used in the analysis—for example, because of missing values or invalid weights—are counted. The counter is reset to 1 at the start of every new BY group. Only in some circumstances will _N_ equal the actual observation number. The symbol should thus be used sparingly to avoid unexpected results.

You must observe the following syntax rules when you use the automatic variables. The _LINP_ symbol cannot appear on the left side of programming statements; you cannot make an assignment to the _LINP_ variable. The value of the linear predictor is controlled by the CLASS, MODEL, and RANDOM statements as well as the current parameter estimates and solutions. You can, however, use the _LINP_ variable on the right side of other operations. Suppose, for example, that you want to transform the linear predictor prior to applying the inverse log link. The following statements are not valid because the linear predictor appears in an assignment:

```
proc glimmix;
   _linp_ = sqrt(abs(_linp_));
   _mu_   = exp(_linp_);
   model count = logtstd / dist=poisson;
run;
```

The next statements achieve the desired result:

```
proc glimmix;
   _mu_ = exp(sqrt(abs(_linp_)));
   model count = logtstd / dist=poisson;
run;
```

If the value of the linear predictor is altered in any way through programming statements, you need to ensure that an assignment to _MU_ follows. The assignment to variable P in the next set of GLIMMIX statements is without effect:

```
proc glimmix;
   p = _linp_ + rannor(454);
   model count = logtstd / dist=poisson;
run;
```

A user-defined link function is implied by expressing _MU_ as a function of _LINP_. That is, if $\mu = g^{-1}(\eta)$, you are providing an expression for the inverse link function with programming statements. It is neither necessary nor possible to give an expression for the inverse operation, $\eta = g(\mu)$. The variance function is determined by expressing _VARIANCE_ as a function of _MU_. If the _MU_ variable appears in an assignment statement inside PROC GLIMMIX, the LINK= option of the MODEL statement is ignored. If the _VARIANCE_ function appears in an assignment statement, the DIST= option is ignored. Furthermore, the associated variance function per Table 44.19 is not honored. In short, user-defined expressions take precedence over built-in defaults.

If you specify your own link and variance function, the assignment to _MU_ must precede an assignment to the variable _VARIANCE_.

The following two sets of GLIMMIX statements yield the same parameter estimates, but the models differ statistically:

```
proc glimmix;
   class block entry;
   model y/n = block entry / dist=binomial link=logit;
run;

proc glimmix;
   class block entry;
   prob = 1 / (1+exp(- _linp_));
   _mu_ = n * prob ;
   _variance_ = n * prob *(1-prob);
   model y = block entry;
run;
```

The first GLIMMIX invocation models the proportion y/n as a binomial proportion with a logit link. The DIST= and LINK= options are superfluous in this case, because the GLIMMIX procedure defaults to the binomial distribution in light of the *events/trials* syntax. The logit link is that distribution's default link. The second set of GLIMMIX statements models the count variable y and takes the binomial sample size into account through assignments to the mean and variance function. In contrast to the first set of GLIMMIX statements, the distribution of y is unknown. Only its mean and variance are known. The model parameters are estimated by maximum likelihood in the first case and by quasi-likelihood in the second case.

Details: GLIMMIX Procedure

Generalized Linear Models Theory

A generalized linear model consists of the following:

- a linear predictor $\eta = \mathbf{x}'\boldsymbol{\beta}$
- a monotonic mapping between the mean of the data and the linear predictor
- a response distribution in the exponential family of distributions

A density or mass function in this family can be written as

$$f(y) = \exp\left\{\frac{y\theta - b(\theta)}{\phi} + c(y, f(\phi))\right\}$$

for some functions $b(\cdot)$ and $c(\cdot)$. The parameter θ is called the natural (canonical) parameter. The parameter ϕ is a scale parameter, and it is not present in all exponential family distributions. See Table 44.19 for a list of distributions for which $\phi \equiv 1$. In the case where observations are weighted, the scale parameter is replaced with ϕ/w in the preceding density (or mass function), where w is the weight associated with the observation y.

The mean and variance of the data are related to the components of the density, $\mathrm{E}[Y] = \mu = b'(\theta)$, $\mathrm{Var}[Y] = \phi b''(\theta)$, where primes denote first and second derivatives. If you express θ as a function of μ, the relationship is known as the natural link or the canonical link function. In other words, modeling data with a canonical link assumes that $\theta = \mathbf{x}'\boldsymbol{\beta}$; the effect contributions are additive on the canonical scale. The second derivative of $b(\cdot)$, expressed as a function of μ, is the variance function of the generalized linear model, $a(\mu) = b''(\theta(\mu))$. Note that because of this relationship, the distribution determines the variance function and the canonical link function. You cannot, however, proceed in the opposite direction. If you provide a user-specified variance function, the GLIMMIX procedure assumes that only the first two moments of the response distribution are known. The full distribution of the data is then unknown and maximum likelihood estimation is not possible. Instead, the GLIMMIX procedure then estimates parameters by quasi-likelihood.

Maximum Likelihood

The GLIMMIX procedure forms the log likelihoods of generalized linear models as

$$L(\boldsymbol{\mu}, \phi; \mathbf{y}) = \sum_{i=1}^{n} f_i \, l(\mu_i, \phi; y_i, w_i)$$

where $l(\mu_i, \phi; y_i, w_i)$ is the log likelihood contribution of the ith observation with weight w_i and f_i is the value of the frequency variable. For the determination of w_i and f_i, see the WEIGHT and FREQ statements. The individual log likelihood contributions for the various distributions are as follows.

Beta

$$\begin{aligned} l(\mu_i, \phi; y_i, w_i) = \log &\left\{ \frac{\Gamma(\phi/w_i)}{\Gamma(\mu\phi/w_i)\Gamma((1-\mu)\phi/w_i)} \right\} \\ &+ (\mu\phi/w_i - 1)\log\{y_i\} \\ &+ ((1-\mu)\phi/w_i - 1)\log\{1-y_i\} \end{aligned}$$

$\mathrm{Var}[Y] = \mu(1-\mu)/(1+\phi)$, $\phi > 0$. See Ferrari and Cribari-Neto (2004).

Binary

$$l(\mu_i, \phi; y_i, w_i) = w_i(y_i \log\{\mu_i\} + (1-y_i)\log\{1-\mu_i\})$$

$\mathrm{Var}[Y] = \mu(1-\mu)$, $\phi \equiv 1$.

Binomial

$$\begin{aligned} l(\mu_i, \phi; y_i, w_i) = &\; w_i \left(y_i \log\{\mu_i\} + (n_i - y_i)\log\{1-\mu_i\}\right) \\ &+ w_i \left(\log\{\Gamma(n_i+1)\} - \log\{\Gamma(y_i+1)\} - \log\{\Gamma(n_i - y_i + 1)\}\right) \end{aligned}$$

where y_i and n_i are the *events* and *trials* in the *events/trials* syntax, and $0 < \mu < 1$. $\mathrm{Var}[Y/n] = \mu(1-\mu)/n$, $\phi \equiv 1$.

Exponential

$$l(\mu_i, \phi; y_i, w_i) = \begin{cases} -\log\{\mu_i\} - y_i/\mu_i & w_i = 1 \\ w_i \log\left\{\frac{w_i y_i}{\mu_i}\right\} - \frac{w_i y_i}{\mu_i} - \log\{y_i \Gamma(w_i)\} & w_i \neq 1 \end{cases}$$

$\mathrm{Var}[Y] = \mu^2$, $\phi \equiv 1$.

Gamma

$$l(\mu_i, \phi; y_i, w_i) = w_i \phi \log\left\{\frac{w_i y_i \phi}{\mu_i}\right\} - \frac{w_i y_i \phi}{\mu_i} - \log\{y_i\} - \log\{\Gamma(w_i \phi)\}$$

$\mathrm{Var}[Y] = \phi\mu^2$, $\phi > 0$.

Geometric

$$l(\mu_i, \phi; y_i, w_i) = y_i \log\left\{\frac{\mu_i}{w_i}\right\} - (y_i + w_i)\log\left\{1 + \frac{\mu_i}{w_i}\right\} + \log\left\{\frac{\Gamma(y_i + w_i)}{\Gamma(w_i)\Gamma(y_i + 1)}\right\}$$

$\mathrm{Var}[Y] = \mu + \mu^2$, $\phi \equiv 1$.

Inverse Gaussian

$$l(\mu_i, \phi; y_i, w_i) = -\frac{1}{2}\left[\frac{w_i(y_i - \mu_i)^2}{y_i \phi \mu_i^2} + \log\left\{\frac{\phi y_i^3}{w_i}\right\} + \log\{2\pi\}\right]$$

$\mathrm{Var}[Y] = \phi\mu^3$, $\phi > 0$.

"Lognormal"

$$l(\mu_i, \phi; \log\{y_i\}, w_i) = -\frac{1}{2}\left[\frac{w_i(\log\{y_i\} - \mu_i)^2}{\phi} + \log\left\{\frac{\phi}{w_i}\right\} + \log\{2\pi\}\right]$$

$\mathrm{Var}[\log\{Y\}] = \phi$, $\phi > 0$.
If you specify DIST=LOGNORMAL with response variable Y, the GLIMMIX procedure assumes that $\log\{Y\} \sim N(\mu, \sigma^2)$. Note that the preceding density is not the density of Y.

Multinomial

$$l(\boldsymbol{\mu}_i, \phi; \mathbf{y}_i, w_i) = w_i \sum_{j=1}^{J} y_{ij} \log\{\mu_{ij}\}$$

$\phi \equiv 1$.

Negative Binomial

$$l(\mu_i, \phi; y_i, w_i) = y_i \log\left\{\frac{k\mu_i}{w_i}\right\} - (y_i + w_i/k)\log\left\{1 + \frac{k\mu_i}{w_i}\right\} + \log\left\{\frac{\Gamma(y_i + w_i/k)}{\Gamma(w_i/k)\Gamma(y_i + 1)}\right\}$$

$\mathrm{Var}[Y] = \mu + k\mu^2$, $k > 0, \phi \equiv 1$.
For a given k, the negative binomial distribution is a member of the exponential family. The parameter k is related to the scale of the data, because it is part of the variance function. However, it cannot be factored from the variance, as is the case with the ϕ parameter in many other distributions. The parameter k is designated as "Scale" in the "Parameter Estimates" table of the GLIMMIX procedure.

Normal (Gaussian)

$$l(\mu_i, \phi; y_i, w_i) = -\frac{1}{2}\left[\frac{w_i(y_i - \mu_i)^2}{\phi} + \log\left\{\frac{\phi}{w_i}\right\} + \log\{2\pi\}\right]$$

$$\text{Var}[Y] = \phi,\ \phi > 0.$$

Poisson

$$l(\mu_i, \phi; y_i, w_i) = w_i(y_i \log\{\mu_i\} - \mu_i - \log\{\Gamma(y_i + 1)\})$$

$$\text{Var}[Y] = \mu,\ \phi \equiv 1.$$

Shifted T

$$\begin{aligned} z_i &= -0.5\log\{\phi/\sqrt{w_i}\} + \log\{\Gamma(0.5(\nu + 1)\} \\ &\quad - \log\{\Gamma(0.5\nu)\} - 0.5 \times \log\{\pi\nu\} \\ l(\mu_i, \phi; y_i, w_i) &= -(\nu/2 + 0.5)\log\left\{1 + \frac{w_i}{\nu}\frac{(y_i - \mu_i)^2}{\phi}\right\} + z_i \end{aligned}$$

$$\phi > 0, \nu > 0,\ \text{Var}[Y] = \phi\nu/(\nu - 2).$$

Define the parameter vector for the generalized linear model as $\boldsymbol{\theta} = \boldsymbol{\beta}$, if $\phi \equiv 1$, and as $\boldsymbol{\theta} = [\boldsymbol{\beta}', \phi]'$ otherwise. $\boldsymbol{\beta}$ denotes the fixed-effects parameters in the linear predictor. For the negative binomial distribution, the relevant parameter vector is $\boldsymbol{\theta} = [\boldsymbol{\beta}', k]'$. The gradient and Hessian of the negative log likelihood are then

$$\mathbf{g} = -\frac{\partial L(\boldsymbol{\theta}; \mathbf{y})}{\partial \boldsymbol{\theta}} \qquad \mathbf{H} = -\frac{\partial^2 L(\boldsymbol{\theta}; \mathbf{y})}{\partial \boldsymbol{\theta}\, \partial \boldsymbol{\theta}'}$$

The GLIMMIX procedure computes the gradient vector and Hessian matrix analytically, unless your programming statements involve functions whose derivatives are determined by finite differences. If the procedure is in scoring mode, **H** is replaced by its expected value. PROC GLIMMIX is in scoring mode when the number *n* of SCORING=*n* iterations has not been exceeded and the optimization technique uses second derivatives, or when the Hessian is computed at convergence and the EXPHESSIAN option is in effect. Note that the objective function is the negative log likelihood when the GLIMMIX procedure fits a GLM model. The procedure performs a minimization problem in this case.

In models for independent data with known distribution, parameter estimates are obtained by the method of maximum likelihood. No parameters are profiled from the optimization. The default optimization technique for GLMs is the Newton-Raphson algorithm, except for Gaussian models with identity link, which do not require iterative model fitting. In the case of a Gaussian model, the scale parameter is estimated by restricted maximum likelihood, because this estimate is unbiased. The results from the GLIMMIX procedure agree with those from the GLM and REG procedure for such models. You can obtain the maximum likelihood estimate of the scale parameter with the NOREML option in the PROC GLIMMIX statement. To change the optimization algorithm, use the TECHNIQUE= option in the NLOPTIONS statement.

Standard errors of the parameter estimates are obtained from the inverse of the (observed or expected) second derivative matrix **H**.

Scale and Dispersion Parameters

The parameter ϕ in the log-likelihood functions is a scale parameter. McCullagh and Nelder (1989, p. 29) refer to it as the dispersion parameter. With the exception of the normal distribution, ϕ does not correspond to the variance of an observation, the variance of an observation in a generalized linear model is a function of ϕ and μ. In a generalized linear model (GLM mode), the GLIMMIX procedure displays the estimate of ϕ is as "Scale" in the "Parameter Estimates" table. Note that for some distributions this scale is different from that reported by the GENMOD procedure in its "Parameter Estimates" table. The scale reported by PROC GENMOD is sometimes a transformation of the dispersion parameter in the log-likelihood function. Table 44.19 displays the relationship between the "Scale" entries reported by the two procedures in terms of the ϕ (or k) parameter in the GLIMMIX log-likelihood functions.

Table 44.19 Scales in Parameter Estimates Table

Distribution	GLIMMIX Reports	GENMOD Reports
Beta	$\widehat{\phi}$	N/A
Gamma	$\widehat{\phi}$	$\widehat{\phi}$
Inverse Gaussian	$\widehat{\phi}$	$\sqrt{\widehat{\phi}}$
Negative binomial	$\widehat{k}$	$\widehat{k}$
Normal	$\widehat{\phi} = \widehat{\mathrm{Var}}[Y]$	$\sqrt{\widehat{\phi}}$

Note that for normal linear models, PROC GLIMMIX by default estimates the parameters by restricted maximum likelihood, whereas PROC GENMOD estimates the parameters by maximum likelihood. As a consequence, the scale parameter in the "Parameter Estimates" table of the GLIMMIX procedure coincides for these models with the mean-squared error estimate of the GLM or REG procedures. To obtain maximum likelihood estimates in a normal linear model in the GLIMMIX procedure, specify the NOREML option in the PROC GLIMMIX statement.

Quasi-likelihood for Independent Data

Quasi-likelihood estimation uses only the first and second moment of the response. In the case of independent data, this requires only a specification of the mean and variance of your data. The GLIMMIX procedure estimates parameters by quasi-likelihood, if the following conditions are met:

- The response distribution is unknown, because of a user-specified variance function.
- There are no G-side random effects.
- There are no R-side covariance structures or at most an overdispersion parameter.

Under some mild regularity conditions, the function

$$Q(\mu_i, y_i) = \int_{y_i}^{\mu_i} \frac{y_i - t}{\phi a(t)}\, dt$$

known as the log quasi-likelihood of the ith observation, has some properties of a log-likelihood function (McCullagh and Nelder 1989, p. 325). For example, the expected value of its derivative is zero, and the

variance of its derivative equals the negative of the expected value of the second derivative. Consequently,

$$QL(\boldsymbol{\mu}, \phi, \mathbf{y}) = \sum_{i=1}^{n} f_i w_i \frac{Y_i - \mu_i}{\phi a(\mu_i)}$$

can serve as the score function for estimation. Quasi-likelihood estimation takes as the gradient and "Hessian" matrix—with respect to the fixed-effects parameters $\boldsymbol{\beta}$—the quantities

$$\mathbf{g}_{ql} = \left[g_{ql,j}\right] = \left[\frac{\partial QL(\boldsymbol{\mu}, \phi, \mathbf{y})}{\partial \beta_j}\right] = \mathbf{D}'\mathbf{V}^{-1}(\mathbf{Y} - \boldsymbol{\mu})/\phi$$

$$\mathbf{H}_{ql} = \left[h_{ql,jk}\right] = \left[\frac{\partial^2 QL(\boldsymbol{\mu}, \phi, \mathbf{y})}{\partial \beta_j \partial \beta_k}\right] = \mathbf{D}'\mathbf{V}^{-1}\mathbf{D}/\phi$$

In this expression, $\mathbf{D}$ is a matrix of derivatives of $\boldsymbol{\mu}$ with respect to the elements in $\boldsymbol{\beta}$, and $\mathbf{V}$ is a diagonal matrix containing variance functions, $\mathbf{V} = [a(\mu_1), \cdots, a(\mu_n)]$. Notice that $\mathbf{H}_{ql}$ is not the second derivative matrix of $Q(\boldsymbol{\mu}, \mathbf{y})$. Rather, it is the negative of the expected value of $\partial \mathbf{g}_{ql}/\partial \boldsymbol{\beta}$. $\mathbf{H}_{ql}$ thus has the form of a "scoring Hessian."

The GLIMMIX procedure fixes the scale parameter ϕ at 1.0 by default. To estimate the parameter, add the statement

```
random _residual_;
```

The resulting estimator (McCullagh and Nelder 1989, p. 328) is

$$\widehat{\phi} = \frac{1}{m} \sum_{i=1}^{n} f_i w_i \frac{(y_i - \widehat{\mu}_i)^2}{a(\widehat{\mu}_i)}$$

where $m = f - \text{rank}\{\mathbf{X}\}$ if the NOREML option is in effect, $m = f$ otherwise, and f is the sum of the frequencies.

See Example 44.4 for an application of quasi-likelihood estimation with PROC GLIMMIX.

Effects of Adding Overdispersion

You can add a multiplicative overdispersion parameter to a generalized linear model in the GLIMMIX procedure with the statement

```
random _residual_;
```

For models in which $\phi \equiv 1$, this effectively lifts the constraint of the parameter. In models that already contain a ϕ or k scale parameter—such as the normal, gamma, or negative binomial model—the statement adds a multiplicative scalar (the overdispersion parameter, ϕ_o) to the variance function.

The overdispersion parameter is estimated from Pearson's statistic after all other parameters have been determined by (restricted) maximum likelihood or quasi-likelihood. This estimate is

$$\widehat{\phi}_o = \frac{1}{\phi^p m} \sum_{i=1}^{n} f_i w_i \frac{(y_i - \mu_i)^2}{a(\mu_i)}$$

where $m = f - \text{rank}\{\mathbf{X}\}$ if the NOREML option is in effect, and $m = f$ otherwise, and f is the sum of the frequencies. The power p is –1 for the gamma distribution and 1 otherwise.

Adding an overdispersion parameter does not alter any of the other parameter estimates. It only changes the variance-covariance matrix of the estimates by a certain factor. If overdispersion arises from correlations among the observations, then you should investigate more complex random-effects structures.

Generalized Linear Mixed Models Theory

Model or Integral Approximation

In a generalized linear model, the log likelihood is well defined, and an objective function for estimation of the parameters is simple to construct based on the independence of the data. In a GLMM, several problems must be overcome before an objective function can be computed.

- The model might be vacuous in the sense that no valid joint distribution can be constructed either in general or for a particular set of parameter values. For example, if $\mathbf{Y}$ is an equicorrelated $(n \times 1)$ vector of binary responses with the same success probability and a symmetric distribution, then the lower bound on the correlation parameter depends on n and π (Gilliland and Schabenberger 2001). If further restrictions are placed on the joint distribution, as in Bahadur (1961), the correlation is also restricted from above.
- The dependency between mean and variance for nonnormal data places constraints on the possible correlation models that simultaneously yield valid joint distributions and a desired conditional distributions. Thus, for example, aspiring for conditional Poisson variates that are marginally correlated according to a spherical spatial process might not be possible.
- Even if the joint distribution is feasible mathematically, it still can be out of reach computationally. When data are independent, conditional on the random effects, the marginal log likelihood can in principle be constructed by integrating out the random effects from the joint distribution. However, numerical integration is practical only when the number of random effects is small and when the data have a clustered (subject) structure.

Because of these special features of generalized linear mixed models, many estimation methods have been put forth in the literature. The two basic approaches are (1) to approximate the objective function and (2) to approximate the model. Algorithms in the second category can be expressed in terms of Taylor series (linearizations) and are hence also known as linearization methods. They employ expansions to approximate the model by one based on pseudo-data with fewer nonlinear components. The process of computing the linear approximation must be repeated several times until some criterion indicates lack of further progress. Schabenberger and Gregoire (1996) list numerous algorithms based on Taylor series for the case of clustered data alone. The fitting methods based on linearizations are usually doubly iterative. The generalized linear mixed model is approximated by a linear mixed model based on current values of the covariance parameter estimates. The resulting linear mixed model is then fit, which is itself an iterative process. On convergence, the new parameter estimates are used to update the linearization, which results in a new linear mixed model. The process stops when parameter estimates between successive linear mixed model fits change only within a specified tolerance.

Integral approximation methods approximate the log likelihood of the GLMM and submit the approximated function to numerical optimization. Various techniques are used to compute the approximation: Laplace methods, quadrature methods, Monte Carlo integration, and Markov chain Monte Carlo methods. The advantage of integral approximation methods is to provide an actual objective function for optimization. This enables you to perform likelihood ratio tests among nested models and to compute likelihood-based fit statistics. The estimation process is singly iterative. The disadvantage of integral approximation methods is the difficulty of accommodating crossed random effects and multiple subject effects, and the inability to accommodate R-side covariance structures, even only R-side overdispersion. The number of random effects should be small for integral approximation methods to be practically feasible.

The advantages of linearization-based methods include a relatively simple form of the linearized model that typically can be fit based on only the mean and variance in the linearized form. Models for which the joint distribution is difficult—or impossible—to ascertain can be fit with linearization-based approaches. Models with correlated errors, a large number of random effects, crossed random effects, and multiple types of subjects are thus excellent candidates for linearization methods. The disadvantages of this approach include the absence of a true objective function for the overall optimization process and potentially biased estimates, especially for binary data when the number of observations per subject is small (see the section "Notes on Bias of Estimators" on page 3199 for further comments and considerations about the bias of estimates in generalized linear mixed models). Because the objective function to be optimized after each linearization update depends on the current pseudo-data, objective functions are not comparable across linearizations. The estimation process can fail at both levels of the double iteration scheme.

By default the GLIMMIX procedure fits generalized linear mixed models based on linearizations. The default estimation method in GLIMMIX for models containing random effects is a technique known as restricted pseudo-likelihood (RPL) (Wolfinger and O'Connell 1993) estimation with an expansion around the current estimate of the best linear unbiased predictors of the random effects (METHOD=RSPL).

Two maximum likelihood estimation methods based on integral approximation are available in the GLIMMIX procedure. If you choose METHOD=LAPLACE in a GLMM, the GLIMMIX procedure performs maximum likelihood estimation based on a Laplace approximation of the marginal log likelihood. See the section "Maximum Likelihood Estimation Based on Laplace Approximation" on page 3192 for details about the Laplace approximation with PROC GLIMMIX. If you choose METHOD=QUAD in the PROC GLIMMIX statement in a generalized linear mixed model, the GLIMMIX procedure estimates the model parameters by adaptive Gauss-Hermite quadrature. See the section "Maximum Likelihood Estimation Based on Adaptive Quadrature" on page 3195 for details about the adaptive Gauss-Hermite quadrature approximation with PROC GLIMMIX.

The following subsections discuss the three estimation methods in turn. Keep in mind that your modeling possibilities are increasingly restricted in the order of these subsections. For example, in the class of generalized linear mixed models, the pseudo-likelihood estimation methods place no restrictions on the covariance structure, and Laplace estimation adds restriction with respect to the R-side covariance structure. Adaptive quadrature estimation further requires a clustered data structure—that is, the data must be processed by subjects.

Table 44.20 Model Restrictions Depending on Estimation Method

Method	Restriction
RSPL, RMPL	None
MSPL, MMPL	None
LAPLACE	No R-side effects
QUAD	No R-side effects Requires SUBJECT= effect Requires processing by subjects

Pseudo-likelihood Estimation Based on Linearization

The Pseudo-model

Recall from the section "Notation for the Generalized Linear Mixed Model" on page 3054 that

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

where $\boldsymbol{\gamma} \sim N(\mathbf{0}, \mathbf{G})$ and $\mathrm{Var}[\mathbf{Y}|\boldsymbol{\gamma}] = \mathbf{A}^{1/2}\mathbf{R}\mathbf{A}^{1/2}$. Following Wolfinger and O'Connell (1993), a first-order Taylor series of $\boldsymbol{\mu}$ about $\widetilde{\boldsymbol{\beta}}$ and $\widetilde{\boldsymbol{\gamma}}$ yields

$$g^{-1}(\boldsymbol{\eta}) \doteq g^{-1}(\widetilde{\boldsymbol{\eta}}) + \widetilde{\boldsymbol{\Delta}}\mathbf{X}(\boldsymbol{\beta} - \widetilde{\boldsymbol{\beta}}) + \widetilde{\boldsymbol{\Delta}}\mathbf{Z}(\boldsymbol{\gamma} - \widetilde{\boldsymbol{\gamma}})$$

where

$$\widetilde{\boldsymbol{\Delta}} = \left(\frac{\partial g^{-1}(\boldsymbol{\eta})}{\partial \boldsymbol{\eta}}\right)_{\widetilde{\boldsymbol{\beta}},\widetilde{\boldsymbol{\gamma}}}$$

is a diagonal matrix of derivatives of the conditional mean evaluated at the expansion locus. Rearranging terms yields the expression

$$\widetilde{\boldsymbol{\Delta}}^{-1}(\boldsymbol{\mu} - g^{-1}(\widetilde{\boldsymbol{\eta}})) + \mathbf{X}\widetilde{\boldsymbol{\beta}} + \mathbf{Z}\widetilde{\boldsymbol{\gamma}} \doteq \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\boldsymbol{\gamma}$$

The left side is the expected value, conditional on $\boldsymbol{\gamma}$, of

$$\widetilde{\boldsymbol{\Delta}}^{-1}(\mathbf{Y} - g^{-1}(\widetilde{\boldsymbol{\eta}})) + \mathbf{X}\widetilde{\boldsymbol{\beta}} + \mathbf{Z}\widetilde{\boldsymbol{\gamma}} \equiv \mathbf{P}$$

and

$$\mathrm{Var}[\mathbf{P}|\boldsymbol{\gamma}] = \widetilde{\boldsymbol{\Delta}}^{-1}\mathbf{A}^{1/2}\mathbf{R}\mathbf{A}^{1/2}\widetilde{\boldsymbol{\Delta}}^{-1}$$

You can thus consider the model

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

which is a linear mixed model with pseudo-response $\mathbf{P}$, fixed effects $\boldsymbol{\beta}$, random effects $\boldsymbol{\gamma}$, and $\mathrm{Var}[\boldsymbol{\epsilon}] = \mathrm{Var}[\mathbf{P}|\boldsymbol{\gamma}]$.

Objective Functions

Now define

$$\mathbf{V}(\boldsymbol{\theta}) = \mathbf{Z}\mathbf{G}\mathbf{Z}' + \widetilde{\boldsymbol{\Delta}}^{-1}\mathbf{A}^{1/2}\mathbf{R}\mathbf{A}^{1/2}\widetilde{\boldsymbol{\Delta}}^{-1}$$

as the marginal variance in the linear mixed pseudo-model, where $\boldsymbol{\theta}$ is the $(q \times 1)$ parameter vector containing all unknowns in $\mathbf{G}$ and $\mathbf{R}$. Based on this linearized model, an objective function can be defined, assuming that the distribution of $\mathbf{P}$ is known. The GLIMMIX procedure assumes that $\boldsymbol{\epsilon}$ has a normal distribution. The maximum log pseudo-likelihood (M*x*PL) and restricted log pseudo-likelihood (R*x*PL) for $\mathbf{P}$ are then

$$l(\boldsymbol{\theta}, \mathbf{p}) = -\frac{1}{2}\log|\mathbf{V}(\boldsymbol{\theta})| - \frac{1}{2}\mathbf{r}'\mathbf{V}(\boldsymbol{\theta})^{-1}\mathbf{r} - \frac{f}{2}\log\{2\pi\}$$

$$l_R(\boldsymbol{\theta}, \mathbf{p}) = -\frac{1}{2}\log|\mathbf{V}(\boldsymbol{\theta})| - \frac{1}{2}\mathbf{r}'\mathbf{V}(\boldsymbol{\theta})^{-1}\mathbf{r} - \frac{1}{2}\log|\mathbf{X}'\mathbf{V}(\boldsymbol{\theta})^{-1}\mathbf{X}| - \frac{f-k}{2}\log\{2\pi\}$$

with $\mathbf{r} = \mathbf{p} - \mathbf{X}(\mathbf{X}'\mathbf{V}^{-1}\mathbf{X})^{-}\mathbf{X}'\mathbf{V}^{-1}\mathbf{p}$. f denotes the sum of the frequencies used in the analysis, and k denotes the rank of $\mathbf{X}$. The fixed-effects parameters $\boldsymbol{\beta}$ are profiled from these expressions. The parameters in $\boldsymbol{\theta}$ are estimated by the optimization techniques specified in the NLOPTIONS statement. The objective function for minimization is $-2l(\boldsymbol{\theta}, \mathbf{p})$ or $-2l_R(\boldsymbol{\theta}, \mathbf{p})$. At convergence, the profiled parameters are estimated and the random effects are predicted as

$$\widehat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{V}(\widehat{\boldsymbol{\theta}})^{-1}\mathbf{X})^{-}\mathbf{X}'\mathbf{V}(\widehat{\boldsymbol{\theta}})^{-1}\mathbf{p}$$
$$\widehat{\boldsymbol{\gamma}} = \widehat{\mathbf{G}}\mathbf{Z}'\mathbf{V}(\widehat{\boldsymbol{\theta}})^{-1}\widehat{\mathbf{r}}$$

With these statistics, the pseudo-response and error weights of the linearized model are recomputed and the objective function is minimized again. The predictors $\widehat{\boldsymbol{\gamma}}$ are the estimated BLUPs in the approximated linear model. This process continues until the relative change between parameter estimates at two successive (outer) iterations is sufficiently small. See the PCONV= option in the PROC GLIMMIX statement for the computational details about how the GLIMMIX procedure compares parameter estimates across optimizations.

If the conditional distribution contains a scale parameter $\phi \neq 1$ (Table 44.19), the GLIMMIX procedure profiles this parameter in GLMMs from the log pseudo-likelihoods as well. To this end define

$$\mathbf{V}(\boldsymbol{\theta}^*) = \widetilde{\boldsymbol{\Delta}}^{-1}\mathbf{A}^{1/2}\mathbf{R}^*\mathbf{A}^{1/2}\widetilde{\boldsymbol{\Delta}}^{-1} + \mathbf{Z}\mathbf{G}^*\mathbf{Z}'$$

where $\boldsymbol{\theta}^*$ is the covariance parameter vector with $q-1$ elements. The matrices $\mathbf{G}^*$ and $\mathbf{R}^*$ are appropriately reparameterized versions of $\mathbf{G}$ and $\mathbf{R}$. For example, if $\mathbf{G}$ has a variance component structure and $\mathbf{R} = \phi\mathbf{I}$, then $\boldsymbol{\theta}^*$ contains ratios of the variance components and ϕ, and $\mathbf{R}^* = \mathbf{I}$. The solution for $\widehat{\phi}$ is

$$\widehat{\phi} = \widehat{\mathbf{r}}'\mathbf{V}(\widehat{\boldsymbol{\theta}}^*)^{-1}\widehat{\mathbf{r}}/m$$

where $m = f$ for M*x*PL and $m = f - k$ for R*x*PL. Substitution into the previous functions yields the profiled log pseudo-likelihoods,

$$l(\boldsymbol{\theta}^*, \mathbf{p}) = -\frac{1}{2}\log|\mathbf{V}(\boldsymbol{\theta}^*)| - \frac{f}{2}\log\left\{\mathbf{r}'\mathbf{V}(\boldsymbol{\theta}^*)^{-1}\mathbf{r}\right\} - \frac{f}{2}(1 + \log\{2\pi/f\})$$
$$l_R(\boldsymbol{\theta}^*, \mathbf{p}) = -\frac{1}{2}\log|\mathbf{V}(\boldsymbol{\theta}^*)| - \frac{f-k}{2}\log\left\{\mathbf{r}'\mathbf{V}(\boldsymbol{\theta}^*)^{-1}\mathbf{r}\right\}$$
$$-\frac{1}{2}\log|\mathbf{X}'\mathbf{V}(\boldsymbol{\theta}^*)^{-1}\mathbf{X}| - \frac{f-k}{2}(1 + \log\{2\pi/(f-k)\})$$

Profiling of ϕ can be suppressed with the NOPROFILE option in the PROC GLIMMIX statement.

Where possible, the objective function, its gradient, and its Hessian employ the sweep-based W-transformation (Hemmerle and Hartley 1973; Goodnight 1979; Goodnight and Hemmerle 1979). Further details about the minimization process in the general linear mixed model can be found in Wolfinger, Tobias, and Sall (1994).

Estimated Precision of Estimates

The GLIMMIX procedure produces estimates of the variability of $\widehat{\boldsymbol{\beta}}$, $\widehat{\boldsymbol{\theta}}$, and estimates of the prediction variability for $\widehat{\boldsymbol{\gamma}}$, $\mathrm{Var}[\widehat{\boldsymbol{\gamma}} - \boldsymbol{\gamma}]$. Denote as $\mathbf{S}$ the matrix

$$\mathbf{S} \equiv \widehat{\mathrm{Var}}[\mathbf{P}|\boldsymbol{\gamma}] = \widetilde{\boldsymbol{\Delta}}^{-1}\mathbf{A}^{1/2}\mathbf{R}\mathbf{A}^{1/2}\widetilde{\boldsymbol{\Delta}}^{-1}$$

where all components on the right side are evaluated at the converged estimates. The mixed model equations (Henderson 1984) in the linear mixed (pseudo-)model are then

$$\begin{bmatrix} \mathbf{X}'\mathbf{S}^{-1}\mathbf{X} & \mathbf{X}'\mathbf{S}^{-1}\mathbf{Z} \\ \mathbf{Z}'\mathbf{S}^{-1}\mathbf{X} & \mathbf{Z}'\mathbf{S}^{-1}\mathbf{Z} + \mathbf{G}(\widehat{\boldsymbol{\theta}})^{-1} \end{bmatrix} \begin{bmatrix} \widehat{\boldsymbol{\beta}} \\ \widehat{\boldsymbol{\gamma}} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{S}^{-1}\mathbf{p} \\ \mathbf{Z}'\mathbf{S}^{-1}\mathbf{p} \end{bmatrix}$$

and

$$\mathbf{C} = \begin{bmatrix} \mathbf{X}'\mathbf{S}^{-1}\mathbf{X} & \mathbf{X}'\mathbf{S}^{-1}\mathbf{Z} \\ \mathbf{Z}'\mathbf{S}^{-1}\mathbf{X} & \mathbf{Z}'\mathbf{S}^{-1}\mathbf{Z} + \mathbf{G}(\widehat{\boldsymbol{\theta}})^{-1} \end{bmatrix}^{-}$$

$$= \begin{bmatrix} \widehat{\boldsymbol{\Omega}} & -\widehat{\boldsymbol{\Omega}}\mathbf{X}'\mathbf{V}(\widehat{\boldsymbol{\theta}})^{-1}\mathbf{Z}\mathbf{G}(\widehat{\boldsymbol{\theta}}) \\ -\mathbf{G}(\widehat{\boldsymbol{\theta}})\mathbf{Z}'\mathbf{V}(\widehat{\boldsymbol{\theta}})^{-1}\mathbf{X}\widehat{\boldsymbol{\Omega}} & \mathbf{M} + \mathbf{G}(\widehat{\boldsymbol{\theta}})\mathbf{Z}'\mathbf{V}(\widehat{\boldsymbol{\theta}})^{-1}\mathbf{X}\widehat{\boldsymbol{\Omega}}\mathbf{X}'\mathbf{V}(\widehat{\boldsymbol{\theta}})^{-1}\mathbf{Z}\mathbf{G}(\widehat{\boldsymbol{\theta}}) \end{bmatrix}$$

is the approximate estimated variance-covariance matrix of $[\widehat{\boldsymbol{\beta}}', \widehat{\boldsymbol{\gamma}}' - \boldsymbol{\gamma}']'$. Here, $\widehat{\boldsymbol{\Omega}} = (\mathbf{X}'\mathbf{V}(\widehat{\boldsymbol{\theta}})^{-1}\mathbf{X})^{-}$ and $\mathbf{M} = (\mathbf{Z}'\mathbf{S}^{-1}\mathbf{Z} + \mathbf{G}(\widehat{\boldsymbol{\theta}})^{-1})^{-1}$.

The square roots of the diagonal elements of $\widehat{\boldsymbol{\Omega}}$ are reported in the Standard Error column of the "Parameter Estimates" table. This table is produced with the SOLUTION option in the MODEL statement. The prediction standard errors of the random-effects solutions are reported in the Std Err Pred column of the "Solution for Random Effects" table. This table is produced with the SOLUTION option in the RANDOM statement.

As a cautionary note, $\mathbf{C}$ tends to underestimate the true sampling variability of $[\widehat{\boldsymbol{\beta}}', \widehat{\boldsymbol{\gamma}}']'$, because no account is made for the uncertainty in estimating $\mathbf{G}$ and $\mathbf{R}$. Although inflation factors have been proposed (Kackar and Harville 1984; Kass and Steffey 1989; Prasad and Rao 1990), they tend to be small for data sets that are fairly well balanced. PROC GLIMMIX does not compute any inflation factors by default. The DDFM=KENWARDROGER option in the MODEL statement prompts PROC GLIMMIX to compute a specific inflation factor (Kenward and Roger 1997), along with Satterthwaite-based degrees of freedom.

If $\mathbf{G}(\widehat{\boldsymbol{\theta}})$ is singular, or if you use the CHOL option of the PROC GLIMMIX statement, the mixed model equations are modified as follows. Let $\mathbf{L}$ denote the lower triangular matrix so that $\mathbf{L}\mathbf{L}' = \mathbf{G}(\widehat{\boldsymbol{\theta}})$. PROC GLIMMIX then solves the equations

$$\begin{bmatrix} \mathbf{X}'\mathbf{S}^{-1}\mathbf{X} & \mathbf{X}'\mathbf{S}^{-1}\mathbf{Z}\mathbf{L} \\ \mathbf{L}'\mathbf{Z}'\mathbf{S}^{-1}\mathbf{X} & \mathbf{L}'\mathbf{Z}'\mathbf{S}^{-1}\mathbf{Z}\mathbf{L} + \mathbf{I} \end{bmatrix} \begin{bmatrix} \widehat{\boldsymbol{\beta}} \\ \widehat{\boldsymbol{\tau}} \end{bmatrix} = \begin{bmatrix} \mathbf{X}'\mathbf{S}^{-1}\mathbf{p} \\ \mathbf{L}'\mathbf{Z}'\mathbf{S}^{-1}\mathbf{p} \end{bmatrix}$$

and transforms $\widehat{\boldsymbol{\tau}}$ and a generalized inverse of the left-side coefficient matrix by using $\mathbf{L}$.

The asymptotic covariance matrix of the covariance parameter estimator $\widehat{\boldsymbol{\theta}}$ is computed based on the observed or expected Hessian matrix of the optimization procedure. Consider first the case where the scale parameter ϕ is not present or not profiled. Because $\boldsymbol{\beta}$ is profiled from the pseudo-likelihood, the objective function for minimization is $f(\boldsymbol{\theta}) = -2l(\boldsymbol{\theta}, \mathbf{p})$ for METHOD=MSPL and METHOD=MMPL and $f(\boldsymbol{\theta}) = -2l_R(\boldsymbol{\theta}, \mathbf{p})$ for METHOD=RSPL and METHOD=RMPL. Denote the observed Hessian (second derivative) matrix as

$$\mathbf{H} = \frac{\partial^2 f(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}\, \partial \boldsymbol{\theta}'}$$

The GLIMMIX procedure computes the variance of $\widehat{\boldsymbol{\theta}}$ by default as $2\mathbf{H}^{-1}$. If the Hessian is not positive definite, a sweep-based generalized inverse is used instead. When the EXPHESSIAN option of the PROC GLIMMIX statement is used, or when the procedure is in scoring mode at convergence (see the SCORING option in the PROC GLIMMIX statement), the observed Hessian is replaced with an approximated expected Hessian matrix in these calculations.

Following Wolfinger, Tobias, and Sall (1994), define the following components of the gradient and Hessian in the optimization:

$$\mathbf{g}_1 = \frac{\partial}{\partial \boldsymbol{\theta}} \mathbf{r}'\mathbf{V}(\boldsymbol{\theta})^{-1}\mathbf{r}$$
$$\mathbf{H}_1 = \frac{\partial^2}{\partial \boldsymbol{\theta}\, \partial \boldsymbol{\theta}'} \log\{\mathbf{V}(\boldsymbol{\theta})\}$$
$$\mathbf{H}_2 = \frac{\partial^2}{\partial \boldsymbol{\theta}\, \partial \boldsymbol{\theta}'} \mathbf{r}'\mathbf{V}(\boldsymbol{\theta})^{-1}\mathbf{r}$$
$$\mathbf{H}_3 = \frac{\partial^2}{\partial \boldsymbol{\theta}\, \partial \boldsymbol{\theta}'} \log\{|\mathbf{X}'\mathbf{V}(\boldsymbol{\theta})^{-1}\mathbf{X}|\}$$

Table 44.21 gives expressions for the Hessian matrix $\mathbf{H}$ depending on estimation method, profiling, and scoring.

Table 44.21 Hessian Computation in GLIMMIX

Profiling	Scoring	M*x*PL	R*x*PL
No	No	$\mathbf{H}_1 + \mathbf{H}_2$	$\mathbf{H}_1 + \mathbf{H}_2 + \mathbf{H}_3$
No	Yes	$-\mathbf{H}_1$	$-\mathbf{H}_1 + \mathbf{H}_3$
No	Modified	$-\mathbf{H}_1$	$-\mathbf{H}_1 - \mathbf{H}_3$
Yes	No	$\begin{bmatrix} \mathbf{H}_1 + \mathbf{H}_2/\phi & -\mathbf{g}_2/\phi^2 \\ -\mathbf{g}_2'/\phi^2 & f/\phi^2 \end{bmatrix}$	$\begin{bmatrix} \mathbf{H}_1 + \mathbf{H}_2/\phi + \mathbf{H}_3 & -\mathbf{g}_2/\phi^2 \\ -\mathbf{g}_2'/\phi^2 & (f-k)/\phi^2 \end{bmatrix}$
Yes	Yes	$\begin{bmatrix} -\mathbf{H}_1 & -\mathbf{g}_2/\phi^2 \\ -\mathbf{g}_2'/\phi^2 & f/\phi^2 \end{bmatrix}$	$\begin{bmatrix} -\mathbf{H}_1 + \mathbf{H}_3 & -\mathbf{g}_2/\phi^2 \\ -\mathbf{g}_2'/\phi^2 & (f-k)/\phi^2 \end{bmatrix}$
Yes	Modified	$\begin{bmatrix} -\mathbf{H}_1 & -\mathbf{g}_2/\phi^2 \\ -\mathbf{g}_2'/\phi^2 & f/\phi^2 \end{bmatrix}$	$\begin{bmatrix} -\mathbf{H}_1 - \mathbf{H}_3 & -\mathbf{g}_2/\phi^2 \\ -\mathbf{g}_2'/\phi^2 & (f-k)/\phi^2 \end{bmatrix}$

The "Modified" expressions for the Hessian under scoring in R*x*PL estimation refer to a modified scoring method. In some cases, the modification leads to faster convergence than the standard scoring algorithm. The modification is requested with the SCOREMOD option in the PROC GLIMMIX statement.

Finally, in the case of a profiled scale parameter ϕ, the Hessian for the $(\boldsymbol{\theta}^*, \phi)$ parameterization is converted into that for the $\boldsymbol{\theta}$ parameterization as

$$\mathbf{H}(\boldsymbol{\theta}) = \mathbf{B}\mathbf{H}(\boldsymbol{\theta}^*, \phi)\mathbf{B}'$$

where

$$\mathbf{B} = \begin{bmatrix} 1/\phi & 0 & \cdots & 0 & 0 \\ 0 & 1/\phi & \cdots & 0 & 0 \\ 0 & \cdots & \cdots & 1/\phi & 0 \\ -\theta_1^*/\phi & -\theta_2^*/\phi & \cdots & -\theta_{q-1}^*/\phi & 1 \end{bmatrix}$$

Subject-Specific and Population-Averaged (Marginal) Expansions

There are two basic choices for the expansion locus of the linearization. A subject-specific (SS) expansion uses

$$\widetilde{\boldsymbol{\beta}} = \widehat{\boldsymbol{\beta}} \quad \widetilde{\boldsymbol{\gamma}} = \widehat{\boldsymbol{\gamma}}$$

which are the current estimates of the fixed effects and estimated BLUPs. The population-averaged (PA) expansion expands about the same fixed effects and the expected value of the random effects

$$\widetilde{\boldsymbol{\beta}} = \widehat{\boldsymbol{\beta}} \quad \widetilde{\boldsymbol{\gamma}} = \mathbf{0}$$

To recompute the pseudo-response and weights in the SS expansion, the BLUPs must be computed every time the objective function in the linear mixed model is maximized. The PA expansion does not require any BLUPs. The four pseudo-likelihood methods implemented in the GLIMMIX procedure are the 2×2 factorial combination between two expansion loci and residual versus maximum pseudo-likelihood estimation. The following table shows the combination and the corresponding values of the METHOD= option (PROC GLIMMIX statement); METHOD=RSPL is the default.

Type of PL	Expansion Locus $\widehat{\boldsymbol{\gamma}}$	Expansion Locus $\mathrm{E}[\boldsymbol{\gamma}]$
residual	RSPL	RMPL
maximum	MSPL	MMPL

Maximum Likelihood Estimation Based on Laplace Approximation

Objective Function

Let $\boldsymbol{\beta}$ denote the vector of fixed-effects parameters and $\boldsymbol{\theta}$ the vector of covariance parameters. For Laplace estimation in the GLIMMIX procedure, $\boldsymbol{\theta}$ includes the G-side parameters and a possible scale parameter ϕ, provided that the conditional distribution of the data contains such a scale parameter. $\boldsymbol{\theta}^*$ is the vector of the G-side parameters.

The marginal distribution of the data in a mixed model can be expressed as

$$\begin{aligned} p(\mathbf{y}) &= \int p(\mathbf{y}|\boldsymbol{\gamma},\boldsymbol{\beta},\phi)\, p(\boldsymbol{\gamma}|\boldsymbol{\theta}^*)\, d\boldsymbol{\gamma} \\ &= \int \exp\left\{\log\{p(\mathbf{y}|\boldsymbol{\gamma},\boldsymbol{\beta},\phi)\} + \log\{p(\boldsymbol{\gamma}|\boldsymbol{\theta}^*)\}\right\}\, d\boldsymbol{\gamma} \\ &= \int \exp\left\{c_l\, f(\mathbf{y},\boldsymbol{\beta},\boldsymbol{\theta};\boldsymbol{\gamma})\right\}\, d\boldsymbol{\gamma} \end{aligned}$$

If the constant c_l is large, the Laplace approximation of this integral is

$$L(\boldsymbol{\beta},\boldsymbol{\theta};\widehat{\boldsymbol{\gamma}},\mathbf{y}) = \left(\frac{2\pi}{c_l}\right)^{n_\gamma/2} |-f''(\mathbf{y},\boldsymbol{\beta},\boldsymbol{\theta};\widehat{\boldsymbol{\gamma}})|^{-1/2}\, e^{c_l f(\mathbf{y},\boldsymbol{\beta},\boldsymbol{\theta};\widehat{\boldsymbol{\gamma}})}$$

where n_γ is the number of elements in $\boldsymbol{\gamma}$, f'' is the second derivative matrix

$$f''(\mathbf{y},\boldsymbol{\beta},\boldsymbol{\theta};\widehat{\boldsymbol{\gamma}}) = \frac{\partial^2 f(\mathbf{y},\boldsymbol{\beta},\boldsymbol{\theta};\boldsymbol{\gamma})}{\partial\boldsymbol{\gamma}\,\partial\boldsymbol{\gamma}'}\Big|_{\widehat{\boldsymbol{\gamma}}}$$

and $\widehat{\boldsymbol{\gamma}}$ satisfies the first-order condition

$$\frac{\partial f(\mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\theta}; \boldsymbol{\gamma})}{\partial \boldsymbol{\gamma}} = \mathbf{0}$$

The objective function for Laplace parameter estimation in the GLIMMIX procedure is $-2\log\{L(\boldsymbol{\beta}, \boldsymbol{\theta}\ \widehat{\boldsymbol{\gamma}}, \mathbf{y})\}$. The optimization process is singly iterative, but because $\widehat{\boldsymbol{\gamma}}$ depends on $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{\theta}}$, the GLIMMIX procedure solves a suboptimization problem to determine for given values of $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{\theta}}$ the random-effects solution vector that maximizes $f(\mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\theta}; \boldsymbol{\gamma})$.

When you have longitudinal or clustered data with m independent subjects or clusters, the vector of observations can be written as $\mathbf{y} = [\mathbf{y}_1', \cdots, \mathbf{y}_m']'$, where $\mathbf{y}_i$ is an $n_i \times 1$ vector of observations for subject (cluster) i ($i = 1, \cdots, m$). In this case, assuming conditional independence such that

$$p(\mathbf{y}_i|\boldsymbol{\gamma}_i) = \prod_{j=1}^{n_i} p(y_{ij}|\boldsymbol{\gamma}_i)$$

the marginal distribution of the data can be expressed as

$$\begin{aligned} p(\mathbf{y}) = \prod_{i=1}^{m} p(\mathbf{y}_i) &= \prod_{i=1}^{m} \int p(\mathbf{y}_i|\boldsymbol{\gamma}_i) p(\boldsymbol{\gamma}_i)\, d\boldsymbol{\gamma}_i \\ &= \prod_{i=1}^{m} \int \exp\{n_i f(\mathbf{y}_i, \boldsymbol{\beta}, \boldsymbol{\theta}; \boldsymbol{\gamma}_i)\}\ d\boldsymbol{\gamma}_i \end{aligned}$$

where

$$\begin{aligned} n_i f(\mathbf{y}_i, \boldsymbol{\beta}, \boldsymbol{\theta}; \boldsymbol{\gamma}_i) &= \log\{p(\mathbf{y}_i|\boldsymbol{\gamma}_i)\ p(\boldsymbol{\gamma}_i)\} \\ &= \sum_{j=1}^{n_i} \log\left\{p(y_{ij}|\boldsymbol{\gamma}_i)\right\} + \log\{p(\boldsymbol{\gamma}_i)\} \end{aligned}$$

When the number of observations within a cluster, n_i, is large, the Laplace approximation to the ith individual's marginal probability density function is

$$\begin{aligned} p(\mathbf{y}_i|\boldsymbol{\beta}, \boldsymbol{\theta}) &= \int \exp\{n_i f(\mathbf{y}_i, \boldsymbol{\beta}, \boldsymbol{\theta}; \boldsymbol{\gamma}_i)\}\ d\boldsymbol{\gamma}_i \\ &= \frac{(2\pi)^{n_\gamma}/2}{|-n_i f''(\mathbf{y}_i, \boldsymbol{\beta}, \boldsymbol{\theta}; \widehat{\boldsymbol{\gamma}}_i)|^{1/2}} \exp\{n_i f(\mathbf{y}_i, \boldsymbol{\beta}, \boldsymbol{\theta}; \widehat{\boldsymbol{\gamma}}_i)\} \end{aligned}$$

where $n_{\gamma i}$ is the common dimension of the random effects, $\boldsymbol{\gamma}_i$. In this case, provided that the constant $c_l = \min\{n_i\}$ is large, the Laplace approximation to the marginal log likelihood is

$$\begin{aligned} \log\{L(\boldsymbol{\beta}, \boldsymbol{\theta}; \widehat{\boldsymbol{\gamma}}, \mathbf{y})\} = \sum_{i=1}^{m} \Big\{ & n_i f(\mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\theta}; \widehat{\boldsymbol{\gamma}}_i) + \frac{n_{\gamma i}}{2} \log\{2\pi\} \\ & - \frac{1}{2} \log|-n_i f''(\boldsymbol{\beta}, \boldsymbol{\theta}; \widehat{\boldsymbol{\gamma}}_i)| \Big\} \end{aligned}$$

which serves as the objective function for the METHOD=LAPLACE estimator in PROC GLIMMIX.

The Laplace approximation implemented in the GLIMMIX procedure differs from that in Wolfinger (1993) and Pinheiro and Bates (1995) in important respects. Wolfinger (1993) assumed a flat prior for $\boldsymbol{\beta}$ and expanded the integrand around $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$, leaving only the covariance parameters for the overall optimization. The "fixed" effects $\boldsymbol{\beta}$ and the random effects $\boldsymbol{\gamma}$ are determined in a suboptimization that takes the form of a linear mixed model step with pseudo-data. The GLIMMIX procedure involves only the random effects vector $\boldsymbol{\gamma}$ in the suboptimization. Pinheiro and Bates (1995) and Wolfinger (1993) consider a modified Laplace approximation that replaces the second derivative $f''(\mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\theta}; \widehat{\boldsymbol{\gamma}})$ with an (approximate) expected value, akin to scoring. The GLIMMIX procedure does not use an approximation to $f''(\mathbf{y}, \boldsymbol{\beta}, \boldsymbol{\theta}; \widehat{\boldsymbol{\gamma}})$. The METHOD=RSPL estimates in PROC GLIMMIX are equivalent to the estimates obtained with the modified Laplace approximation in Wolfinger (1993). The objective functions of METHOD=RSPL and Wolfinger (1993) differ in a constant that depends on the number of parameters.

Asymptotic Properties and the Importance of Subjects

Suppose that the GLIMMIX procedure processes your data by subjects (see the section "Processing by Subjects" on page 3218) and let n_i denote the number of observations per subject, $i = 1, \ldots, s$. Arguments in Vonesh (1996) show that the maximum likelihood estimator based on the Laplace approximation is a consistent estimator to order $O_p\{\max\{1/\sqrt{s}\}, 1/\min\{n_i\}\}$. In other words, as the number of subjects and the number of observations per subject grows, the small-sample bias of the Laplace estimator disappears. Note that the term involving the number of subjects in this maximum relates to standard asymptotic theory, and the term involving the number of observations per subject relates to the accuracy of the Laplace approximation (Vonesh 1996). In the case where random effects enter the model linearly, the Laplace approximation is exact and the requirement that $\min\{n_i\} \rightarrow \infty$ can be dropped.

If your model is not processed by subjects but is equivalent to a subject model, the asymptotics with respect to s still apply, because the Hessian matrix of the suboptimization for $\boldsymbol{\gamma}$ breaks into s separate blocks. For example, the following two models are equivalent with respect to s and n_i, although only for the first model does PROC GLIMMIX process the data explicitly by subjects:

```
proc glimmix method=laplace;
   class sub A;
   model y = A;
   random intercept / subject=sub;
run;

proc glimmix method=laplace;
   class sub A;
   model y = A;
   random sub;
run;
```

The same holds, for example, for models with independent nested random effects. The following two models are equivalent, and you can derive asymptotic properties related to s and $\min\{n_i\}$ from the model in the first run:

```
proc glimmix method=laplace;
   class A B block;
   model y = A B A*B;
   random intercept A / subject=block;
run;

proc glimmix method=laplace;
```

```
   class A B block;
   model y = A B A*B;
   random block a*block;
run;
```

The Laplace approximation requires that the dimension of the integral does not increase with the size of the sample. Otherwise the error of the likelihood approximation does not diminish with n_i. This is the case, for example, with exchangeable arrays (Shun and McCullagh 1995), crossed random effects (Shun 1997), and correlated random effects of arbitrary dimension (Raudenbush, Yang, and Yosef 2000). Results in Shun (1997), for example, show that even in this case the standard Laplace approximation has smaller bias than pseudo-likelihood estimates.

Maximum Likelihood Estimation Based on Adaptive Quadrature

Quadrature methods, like the Laplace approximation, approximate integrals. If you choose METHOD=QUAD for a generalized linear mixed model, the GLIMMIX procedure approximates the marginal log likelihood with an adaptive Gauss-Hermite quadrature rule. Gaussian quadrature is particularly well suited to numerically evaluate integrals against probability measures (Lange 1999, Ch. 16). And Gauss-Hermite quadrature is appropriate when the density has kernel $\exp\{-x^2\}$ and integration extends over the real line, as is the case for the normal distribution. Suppose that $p(x)$ is a probability density function and the function $f(x)$ is to be integrated against it. Then the quadrature rule is

$$\int_{-\infty}^{\infty} f(x)\,p(x)\,dx \approx \sum_{i=1}^{N} w_i\,f(x_i)$$

where N denotes the number of quadrature points, the w_i are the quadrature weights, and the x_i are the abscissas. The Gaussian quadrature chooses abscissas in areas of high density, and if $p(x)$ is continuous, the quadrature rule is exact if $f(x)$ is a polynomial of up to degree $2N - 1$. In the generalized linear mixed model the roles of $f(x)$ and $p(x)$ are played by the conditional distribution of the data given the random effects, and the random-effects distribution, respectively. Quadrature abscissas and weights are those of the standard Gauss-Hermite quadrature (Golub and Welsch 1969; see also Table 25.10 of Abramowitz and Stegun 1972; Evans 1993).

A numerical integration rule is called adaptive when it uses a variable step size to control the error of the approximation. For example, an adaptive trapezoidal rule uses serial splitting of intervals at midpoints until a desired tolerance is achieved. The quadrature rule in the GLIMMIX procedure is adaptive in the following sense: if you do not specify the number of quadrature points (nodes) with the QPOINTS= suboption of the METHOD=QUAD option, then the number of quadrature points is determined by evaluating the log likelihood at the starting values at a successively larger number of nodes until a tolerance is met (for more details see the text under the heading "Starting Values" in the next section). Furthermore, the GLIMMIX procedure centers and scales the quadrature points by using the empirical Bayes estimates (EBEs) of the random effects and the Hessian (second derivative) matrix from the EBE suboptimization. This centering and scaling improves the likelihood approximation by placing the abscissas according to the density function of the random effects. It is not, however, adaptiveness in the previously stated sense.

Objective Function

Let $\boldsymbol{\beta}$ denote the vector of fixed-effects parameters and $\boldsymbol{\theta}$ the vector of covariance parameters. For quadrature estimation in the GLIMMIX procedure, $\boldsymbol{\theta}$ includes the G-side parameters and a possible scale parameter ϕ, provided that the conditional distribution of the data contains such a scale parameter. $\boldsymbol{\theta}^*$ is the vector of the

G-side parameters. The marginal distribution of the data for subject i in a mixed model can be expressed as

$$p(\mathbf{y}_i) = \int \cdots \int p(\mathbf{y}_i|\boldsymbol{\gamma}_i, \boldsymbol{\beta}, \phi)\, p(\boldsymbol{\gamma}_i|\boldsymbol{\theta}^*)\, d\boldsymbol{\gamma}_i$$

Suppose N_q denotes the number of quadrature points in each dimension (for each random effect) and r denotes the number of random effects. For each subject, obtain the empirical Bayes estimates of $\boldsymbol{\gamma}_i$ as the vector $\widehat{\boldsymbol{\gamma}}_i$ that minimizes

$$-\log\left\{p(\mathbf{y}_i|\boldsymbol{\gamma}_i, \boldsymbol{\beta}, \phi)p(\boldsymbol{\gamma}_i|\boldsymbol{\theta}^*)\right\} = f(\mathbf{y}_i, \boldsymbol{\beta}, \boldsymbol{\theta}; \boldsymbol{\gamma}_i)$$

If $\mathbf{z} = [z_1, \cdots, z_{N_q}]$ are the standard abscissas for Gauss-Hermite quadrature, and $\mathbf{z}_j^* = [z_{j_1}, \cdots, z_{j_r}]$ is a point on the r-dimensional quadrature grid, then the centered and scaled abscissas are

$$\mathbf{a}_j^* = \widehat{\boldsymbol{\gamma}}_i + 2^{1/2} f''(\mathbf{y}_i, \boldsymbol{\beta}, \boldsymbol{\theta}; \widehat{\boldsymbol{\gamma}}_i)^{-1/2} \mathbf{z}_j^*$$

As for the Laplace approximation, f'' is the second derivative matrix with respect to the random effects,

$$f''(\mathbf{y}_i, \boldsymbol{\beta}, \boldsymbol{\theta}; \widehat{\boldsymbol{\gamma}}_i) = \frac{\partial^2 f(\mathbf{y}_i, \boldsymbol{\beta}, \boldsymbol{\theta}; \boldsymbol{\gamma}_i)}{\partial \boldsymbol{\gamma}_i \partial \boldsymbol{\gamma}_i'}\Big|_{\widehat{\boldsymbol{\gamma}}_i}$$

These centered and scaled abscissas, along with the Gauss-Hermite quadrature weights $\mathbf{w} = [w_1, \cdots, w_{N_q}]$, are used to construct the r-dimensional integral by a sequence of one-dimensional rules

$$\begin{aligned} p(\mathbf{y}_i) &= \int \cdots \int p(\mathbf{y}_i|\boldsymbol{\gamma}_i, \boldsymbol{\beta}, \phi)\, p(\boldsymbol{\gamma}_i|\boldsymbol{\theta}^*)\, d\boldsymbol{\gamma}_i \\ &\approx 2^{r/2} |f''(\mathbf{y}_i, \boldsymbol{\beta}, \boldsymbol{\theta}; \widehat{\boldsymbol{\gamma}}_i)|^{-1/2} \\ &\quad \sum_{j_1=1}^{N_q} \cdots \sum_{j_r=1}^{N_q} \left[p(\mathbf{y}_i|\mathbf{a}_j^*, \boldsymbol{\beta}, \phi) p(\mathbf{a}_j^*|\boldsymbol{\theta}^*) \prod_{k=1}^{r} w_{j_k} \exp z_{j_k}^2 \right] \end{aligned}$$

The right-hand side of this expression, properly accumulated across subjects, is the objective function for adaptive quadrature estimation in the GLIMMIX procedure.

Quadrature or Laplace Approximation

If you select the quadrature rule with a single quadrature point, namely

```
proc glimmix method=quad(qpoints=1);
```

the results will be identical to METHOD=LAPLACE. Computationally, the two methods are not identical, however. METHOD=LAPLACE can be applied to a considerably larger class of models. For example, crossed random effects, models without subjects, or models with non-nested subjects can be handled with the Laplace approximation but not with quadrature. Furthermore, METHOD=LAPLACE draws on a number of computational simplifications that can increase its efficiency compared to a quadrature algorithm with a single node. For example, the Laplace approximation is possible with unbounded covariance parameter estimates (NOBOUND option in the PROC GLIMMIX statement) and can permit certain types of negative definite or indefinite **G** matrices. The adaptive quadrature approximation with scaled abscissas typically breaks down when **G** is not at least positive semidefinite.

As the number of random effects grows—for example, if you have nested random effects—quadrature quickly becomes computationally infeasible, due to the high dimensionality of the integral. To this end it is worthwhile to clarify the issues of dimensionality and computational effort as related to the number of quadrature nodes. Suppose that the A effect has 4 levels and consider the following statements:

```
proc glimmix method=quad(qpoints=5);
   class A id;
   model y = / dist=negbin;
   random A / subject=id;
run;
```

For each subject, computing the marginal log likelihood requires the numerical evaluation of a four-dimensional integral. As part of this evaluation $5^4 = 625$ conditional log likelihoods need to be computed for each observation on each pass through the data. As the number of quadrature points or the number of random effects increases, this constitutes a sizable computational effort. Suppose, for example, that an additional random effect with $b = 2$ levels is added as an interaction. The following statements then require evaluation of $5^{(4+8)} = 244140625$ conditional log likelihoods for each observation one each pass through the data:

```
proc glimmix method=quad(qpoints=5);
   class A B id;
   model y = / dist=negbin;
   random A A*B / subject=id;
run;
```

As the number of random effects increases, Laplace approximation presents a computationally more expedient alternative.

If you wonder whether METHOD=LAPLACE would present a viable alternative to a model that you can fit with METHOD=QUAD, the "Optimization Information" table can provide some insights. The table contains as its last entry the number of quadrature points determined by PROC GLIMMIX to yield a sufficiently accurate approximation of the log likelihood (at the starting values). In many cases, a single quadrature node is sufficient, in which case the estimates are identical to those of METHOD=LAPLACE.

Aspects Common to Adaptive Quadrature and Laplace Approximation

Estimated Precision of Estimates

Denote as $\mathbf{H}$ the second derivative matrix

$$\mathbf{H} = -\frac{\partial^2 \log\{L(\boldsymbol{\beta}, \boldsymbol{\theta}\ \widehat{\boldsymbol{\gamma}})\}}{\partial[\boldsymbol{\beta}, \boldsymbol{\theta}]\partial[\boldsymbol{\beta}', \boldsymbol{\theta}']}$$

evaluated at the converged solution of the optimization process. Partition its inverse as

$$\mathbf{H}^{-1} = \begin{bmatrix} \mathbf{C}(\boldsymbol{\beta}, \boldsymbol{\beta}) & \mathbf{C}(\boldsymbol{\beta}, \boldsymbol{\theta}) \\ \mathbf{C}(\boldsymbol{\theta}, \boldsymbol{\beta}) & \mathbf{C}(\boldsymbol{\theta}, \boldsymbol{\theta}) \end{bmatrix}$$

For METHOD=LAPLACE and METHOD=QUAD, the GLIMMIX procedure computes $\mathbf{H}$ by finite forward differences based on the analytic gradient of $\log\{L(\boldsymbol{\beta}, \boldsymbol{\theta}\ \widehat{\boldsymbol{\gamma}})\}$. The partition $\mathbf{C}(\boldsymbol{\theta}, \boldsymbol{\theta})$ serves as the asymptotic covariance matrix of the covariance parameter estimates (ASYCOV option in the PROC GLIMMIX statement). The standard errors reported in the "Covariance Parameter Estimates" table are based on the diagonal entries of this partition.

If you request an empirical standard error matrix with the EMPIRICAL option in the PROC GLIMMIX statement, a likelihood-based sandwich estimator is computed based on the subject-specific gradients of the Laplace or quadrature approximation. The sandwich estimator then replaces $\mathbf{H}^{-1}$ in calculations following convergence.

To compute the standard errors and prediction standard errors of linear combinations of $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$, PROC GLIMMIX forms an approximate prediction variance matrix for $[\widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\gamma}}]'$ from

$$\mathbf{P} = \begin{bmatrix} \mathbf{H}^{-1} & \mathbf{H}^{-1}\left(\frac{\partial \widehat{\boldsymbol{\gamma}}}{\partial[\boldsymbol{\beta},\boldsymbol{\theta}]}\right) \\ \left(\frac{\partial \widehat{\boldsymbol{\gamma}}}{\partial[\boldsymbol{\beta}',\boldsymbol{\theta}']}\right)\mathbf{H}^{-1} & \boldsymbol{\Gamma}^{-1} + \left(\frac{\partial \widehat{\boldsymbol{\gamma}}}{\partial[\boldsymbol{\beta}',\boldsymbol{\theta}']}\right)\mathbf{H}^{-1}\left(\frac{\partial \widehat{\boldsymbol{\gamma}}}{\partial[\boldsymbol{\beta},\boldsymbol{\theta}]}\right) \end{bmatrix}$$

where $\boldsymbol{\Gamma}$ is the second derivative matrix from the $\boldsymbol{\gamma}$ suboptimization that maximizes $f(\mathrm{y}, \boldsymbol{\beta}, \boldsymbol{\theta}; \boldsymbol{\gamma})$ for given values of $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$. The prediction variance submatrix for the random effects is based on approximating the conditional mean squared error of prediction as in Booth and Hobert (1998). Note that even in the normal linear mixed model, the approximate conditional prediction standard errors are not identical to the prediction standard errors you obtain by inversion of the mixed model equations.

Conditional Fit and Output Statistics

When you estimate the parameters of a mixed model by Laplace approximation or quadrature, the GLIMMIX procedure displays fit statistics related to the marginal distribution as well as the conditional distribution $p(\mathrm{y}|\widehat{\boldsymbol{\gamma}}, \widehat{\boldsymbol{\beta}}, \widehat{\phi})$. The ODS name of the "Conditional Fit Statistics" table is CondFitStatistics. Because the marginal likelihood is approximated numerically for these methods, statistics based on the marginal distribution are not available. Instead of the generalized Pearson chi-square statistic in the "Fit Statistics" table, PROC GLIMMIX reports the Pearson statistic of the conditional distribution in the "Conditional Fit Statistics" table.

The unavailability of the marginal distribution also affects the set of output statistics that can be produced with METHOD=LAPLACE and METHOD=QUAD. Output statistics and statistical graphics that depend on the marginal variance of the data are not available with these estimation methods.

User-Defined Variance Function

If you provide your own variance function, PROC GLIMMIX generally assumes that the (conditional) distribution of the data is unknown. Laplace or quadrature estimation would then not be possible. When you specify a variance function with METHOD=LAPLACE or METHOD=QUAD, the procedure assumes that the conditional distribution is normal. For example, consider the following statements to fit a mixed model to count data:

```
proc glimmix method=laplace;
   class sub;
   _variance_ = _phi_*_mu_;
   model count = x / s link=log;
   random int / sub=sub;
run;
```

The variance function and the link suggest an overdispersed Poisson model. The Poisson distribution cannot accommodate the extra scale parameter _PHI_, however. In this situation, the GLIMMIX procedure fits a mixed model with random intercepts, log link function, and variance function $\phi\mu$, assuming that the count variable is normally distributed, given the random effects.

Starting Values

Good starting values for the fixed effects and covariance parameters are important for Laplace and quadrature methods because the process commences with a suboptimization in which the empirical Bayes estimates of the random effects must be obtained before the optimization can get under way. Furthermore, the starting values are important for the adaptive choice of the number of quadrature points.

If you choose METHOD=LAPLACE or METHOD=QUAD and you do not provide starting values for the covariance parameters through the PARMS statement, the GLIMMIX procedure determines starting values in the following steps.

1. A GLM is fit initially to obtain starting values for the fixed-effects parameters. No output is produced from this stage. The number of initial iterations of this GLM fit can be controlled with the INITITER= option in the PROC GLIMMIX statement. You can suppress this step with the NOINITGLM option in the PROC GLIMMIX statement.

2. Given the fixed-effects estimates, starting values for the covariance parameters are computed by a MIVQUE0 step (Goodnight 1978a).

3. For METHOD=QUAD you can follow these steps with several pseudo-likelihood updates to improve on the estimates and to obtain solutions for the random effects. The number of pseudo-likelihood steps is controlled by the INITPL= suboption of METHOD=QUAD.

4. For METHOD=QUAD, if you do not specify the number of quadrature points with the suboptions of the METHOD option, the GLIMMIX procedure attempts to determine a sufficient number of points adaptively as follows. Suppose that N_q denotes the number of nodes in each dimension. If N_{min} and N_{max} denote the values from the QMIN= and QMAX= suboptions, respectively, the sequence for values less than 11 is constructed in increments of 2 starting at N_{min}. Values greater than 11 are incremented in steps of r. The default value is $r = 10$. The default sequence, without specifying the QMIN=, QMAX=, or QFAC= option, is thus $1, 3, 5, 7, 9, 11, 21, 31$. If the relative difference of the log-likelihood approximation for two values in the sequence is less than the QTOL=t value (default $t = 0.0001$), the GLIMMIX procedure uses the lesser value for N_q in the subsequent optimization. If the relative difference does not fall below the tolerance t for any two subsequent values in the sequence, no estimation takes place.

Notes on Bias of Estimators

Generalized linear mixed models are nonlinear models, and the estimation techniques rely on approximations to the log likelihood or approximations of the model. It is thus not surprising that the estimates of the covariance parameters and the fixed effects are usually not unbiased. Whenever estimates are biased, questions arise about the magnitude of the bias, its dependence on other model quantities, and the order of the bias. The order is important because it determines how quickly the bias vanishes while some aspect of the data increases. Typically, studies of asymptotic properties in models for hierarchical data suppose that the number of subjects (clusters) tends to infinity while the size of the clusters is held constant or grows at a particular rate. Note that asymptotic results so established do not extend to designs with fully crossed random effects, for example.

The following paragraphs summarize some important findings from the literature regarding the bias in covariance parameter and fixed-effects estimates with pseudo-likelihood, Laplace, and adaptive quadrature methods. The remarks draw in particular on results in Breslow and Lin (1995); Lin and Breslow (1996); Pinheiro and Chao (2006). Breslow and Lin (1995); Lin and Breslow (1996) study the "worst case" scenario of binary responses in a matched-pairs design. Their models have a variance component structure, comprising either a single variance component (a subject-specific random intercept; Breslow and Lin 1995) or a diagonal $\mathbf{G}$ matrix (Lin and Breslow 1996). They study the bias in the estimates of the fixed-effects $\boldsymbol{\beta}$ and the covariance parameters $\boldsymbol{\theta}$ when the variance components are near the origin and for a canonical link function.

The matched-pairs design gives rise to a generalized linear mixed model with a cluster (subject) size of 2. Recall that the pseudo-likelihood methods rely on a linearization and a probabilistic assumption that the pseudo-data so obtained follow a normal linear mixed model. Obviously, it is difficult to imagine how the subject-specific (conditional) distribution would follow a normal linear mixed models with binary data in a cluster size of 2. The bias in the pseudo-likelihood estimator of $\boldsymbol{\beta}$ is of order $||\boldsymbol{\theta}||$. The bias for the Laplace estimator of $\boldsymbol{\beta}$ is of smaller magnitude; its asymptotic bias has order $||\boldsymbol{\theta}||^2$.

The Laplace methods and the pseudo-likelihood method produce biased estimators of the variance component θ for the model considered in Breslow and Lin (1995). The order of the asymptotic bias for both estimation methods is θ, as θ approaches zero. Breslow and Lin (1995) comment on the fact that even with matched pairs, the bias vanishes very quickly in the binomial setting. If the conditional mean in the two groups is equal to 0.5, then the asymptotic bias factor of the pseudo-likelihood estimator is $1 - 1/(2n)$, where n is the binomial denominator. This term goes to 1 quickly as n increases. This result underlines the importance of grouping binary observations into binomial responses whenever possible.

The results of Breslow and Lin (1995) and Lin and Breslow (1996) are echoed in the simulation study in Pinheiro and Chao (2006). These authors also consider adaptive quadrature in models with nested, hierarchical, random effects and show that adaptive quadrature with a sufficient number of nodes leads to nearly unbiased—or least biased—estimates. Their results also show that results for binary data cannot so easily be ported to other distributions. Even with a cluster size of 2, the pseudo-likelihood estimates of fixed effects and covariance parameters are virtually unbiased in their simulation of a Poisson GLMM. Breslow and Lin (1995) and Lin and Breslow (1996) "eschew" the residual PL version (METHOD=RSPL) over the maximum likelihood form (METHOD=MSPL). Pinheiro and Chao (2006) consider both forms in their simulation study. As expected, the residual form shows less bias than the MSPL form, for the same reasons REML estimation leads to less biased estimates compared to ML estimation in linear mixed models. However, the gain is modest; see, for example, Table 1 in Pinheiro and Chao (2006). When the variance components are small, there are a sufficient number of observations per cluster, and a reasonable number of clusters, then pseudo-likelihood methods for binary data are very useful—they provide a computationally expedient alternative to numerical integration, and they allow the incorporation of R-side covariance structure into the model. Because many group randomized trials involve many observations per group and small random-effects variances, Murray, Varnell, and Blitstein (2004) refer to the use of conditional models for trials that have a binary outcome as an "overreaction."

Pseudo-likelihood Estimation for Weighted Multilevel Models

Multilevel models provide a flexible and powerful tool for the analysis of data that are observed in nested units at multiple levels. A multilevel model is a special case of generalized linear mixed models that can be handled by the GLIMMIX procedure. In proc GLIMMIX, the SUBJECT= option in the RANDOM statement identifies the clustering structure for the random effects. When the subjects of the multiple RANDOM statements are nested, the model is a multilevel model and each RANDOM statement corresponds to one level.

Using a pseudo-maximum-likelihood approach, you can extend the multilevel model framework to accommodate weights at different levels. Such an approach is very useful in analyzing survey data that arise from multistage sampling. In these sampling designs, survey weights are often constructed to account for unequal sampling probabilities, nonresponse adjustments, and poststratification.

The following survey example from a three-stage sampling design illustrates the use of multiple levels of weighting in a multilevel model. Extending this example to models that have more than three levels is straightforward. Let $i = 1, \ldots, n^{(3)}$, $j = 1, \ldots n_i^{(2)}$, and $k = 1, \ldots n_j^{(1)}$ denote the indices of units at level

3, level 2, and level 1, respectively. Let superscript (l) denote the lth level and $n^{(l)}$ denote the number of level-l units in the sample. Assume that the first-stage cluster (level-3 unit) i is selected with probability π_i; the second-stage cluster (level-2 unit) j is selected with probability $\pi_{j|i}$, given that the first-stage cluster i is already selected in the sample; and the third-stage unit (level-1 unit) k is selected with probability $\pi_{k|ij}$, given that the second-stage cluster j within the first-stage cluster i is already selected in the sample.

If you use the inverse selection probability weights $w_{j|i} = 1/\pi_{j|i}$, $w_{k|ij} = 1/\pi_{k|ij}$, the conditional log likelihood contribution of the first-stage cluster i is

$$\log\left(p(y_i|\gamma_i^{(2)}, \gamma_i^{(3)})\right) = \sum_{j=1}^{n_i^{(2)}} w_{j|i} \sum_{k=1}^{n_j^{(1)}} w_{k|ij} \log\left(p(y_{ijk}|\gamma_{ij}^{(2)}\gamma_i^{(3)})\right)$$

where $\gamma_{ij}^{(2)}$ is the random-effects vector for the jth second-stage cluster, $\gamma_i^{(2)} = (\gamma_{i1}', \gamma_{i2}', \ldots, \gamma_{in_i^{(2)}}')'$, and $\gamma_i^{(3)}$ is the random-effects vector for the ith first-stage cluster.

As with unweighted multilevel models, the adaptive quadrature method is used to compute the pseudo-likelihood of the first-stage cluster i:

$$p(y_i) = \int p(y_i|\gamma_i^{(2)}, \gamma_i^{(3)}) p(\gamma_i^{(2)}) p(\gamma_i^{(3)}) d(\gamma_i^{(2)}) d(\gamma_i^{(3)})$$

The total log pseudo-likelihood is

$$\log\left(p(y)\right) = \sum_{i=1}^{n^{(3)}} w_i \log\left(p(y_i)\right)$$

where $w_i = 1/\pi_i$.

To illustrate weighting in a multilevel model, consider the following data set. In these simulated data, the response `y` is a Poisson-distributed count, `w3` is the weight for the first-stage clusters, `w2` is the weight for the second-stage clusters, and `w1` is the weight for the observation-level units.

```
data d;
   input A w3 AB w2 w1 y x;
   datalines;
1    6.1     1    5.3     7.1    56    -.214
1    6.1     1    5.3     3.9    41    0.732
1    6.1     2    7.3     6.3    50    0.372
1    6.1     2    7.3     3.9    36    -.892
1    6.1     3    4.6     8.4    39    0.424
1    6.1     3    4.6     6.3    35    -.200
2    8.5     1    4.8     7.4    30    0.868
2    8.5     1    4.8     6.7    25    0.110
2    8.5     2    8.4     3.5    36    0.004
2    8.5     2    8.4     4.1    28    0.755
2    8.5     3    .80     3.8    33    -.600
2    8.5     3    .80     7.4    30    -.525
3    9.5     1    8.2     6.7    32    -.454
3    9.5     1    8.2     7.1    24    0.458
3    9.5     2     11     4.8    31    0.162
3    9.5     2     11     7.9    27    1.099
```

```
3    9.5    3    3.9    3.8    15    -1.57
3    9.5    3    3.9    5.5    19    -.448
4    4.5    1    8.0    5.7    30    -.468
4    4.5    1    8.0    2.9    25    0.890
4    4.5    2    6.0    5.0    35    0.635
4    4.5    2    6.0    3.0    30    0.743
4    4.5    3    6.8    7.3    17    -.015
4    4.5    3    6.8    3.1    18    -.560
;
```

You can use the following statements to fit a weighted three-level model:

```
proc glimmix  method=quadrature empirical=classical;
   class A AB;
   model y = x / dist=poisson link=log obsweight=w1;
   random int  / subject=A weight=w3;
   random int  / subject=AB(A) weight=w2;
run;
```

The SUBJECT= option in the first and second RANDOM statements specifies the first-stage and second-stage clusters A and AB(A), respectively. The OBSWEIGHT= option in the MODEL statement specifies the variable for the weight at the observation level. The WEIGHT= option in the RANDOM statement specifies the variable for the weight at the level that is specified by the SUBJECT= option.

For inference about fixed effects and variance that are estimated by pseudo-likelihood, you can use the empirical (sandwich) variance estimators. For weighted multilevel models, the only empirical estimator available in PROC GLIMMIX is EMPIRICAL=CLASSICAL. The EMPIRICAL=CLASSICAL variance estimator can be described as follows.

Let $\alpha = (\beta', \theta')'$, where β is vector of the fixed-effects parameters and θ is the vector of covariance parameters. For an L-level model, Rabe-Hesketh and Skrondal (2006) show that the gradient can be written as a weighted sum of the gradients of the top-level units:

$$\sum_{i=1}^{n^{(L)}} w_i \frac{\partial \log(p(y_i;\alpha))}{\partial \alpha} \equiv \sum_{i=1}^{n^{(L)}} S_i(\alpha)$$

where $n^{(L)}$ is the number of level-L units and $S_i(\alpha)$ is the weighted score vector of the top-level unit i. The estimator of the "meat" of the sandwich estimator can be written as

$$J = \frac{n^{(L)}}{n^{(L)}-1} \sum_{i=1}^{n^{(L)}} S_i(\hat{\alpha}) S_i(\hat{\alpha})'$$

The empirical estimator of the covariance matrix of $\hat{\alpha}$ can be constructed as

$$H(\hat{\alpha})^{-1} J H(\hat{\alpha})^{-1}$$

where $H(\alpha)$ is the second derivative matrix of the log pseudo-likelihood with respect to α.

The covariance parameter estimators that are obtained by the pseudo-maximum-likelihood method can be biased when the sample size is small. Pfeffermann et al. (1998) and Rabe-Hesketh and Skrondal (2006) discuss two weight-scaling methods for reducing the biases of the covariance parameter estimators in a two-level model. To derive the scaling factor λ for a two-level model, let n_i denote the number of level-1

units in the level-2 unit i and let $w_{j|i}$ denote the weight of the jth level-1 unit in level-2 unit i. The first method computes an "apparent" cluster size as the "effective" sample size:

$$\sum_{j=1}^{n_i} \lambda w_{j|i} = \frac{(\sum_{j=1}^{n_i} w_{j|i})^2}{\sum_{j=1}^{n_i} w_{j|i}^2}$$

Therefore the scale factor is

$$\lambda = \frac{\sum_{j=1}^{n_i} w_{j|i}}{\sum_{j=1}^{n_i} w_{j|i}^2}$$

The second method sets the apparent cluster size equal to the actual cluster size so that the scale factor is

$$\lambda = \frac{n_i}{\sum_{j=1}^{n_i} w_{j|i}}$$

PROC GLIMMIX uses the weights provided in the data set directly. To use the scaled weights, you need to provide them in the data set.

GLM Mode or GLMM Mode

The GLIMMIX procedure uses two basic modes of parameter estimation, and it can be important for you to understand the differences between the two modes.

In GLM mode, the data are never correlated and there can be no G-side random effects. Typical examples are logistic regression and normal linear models. When you fit a model in GLM mode, the METHOD= option in the PROC GLIMMIX statement has no effect. PROC GLIMMIX estimates the parameters of the model by maximum likelihood, (restricted) maximum likelihood, or quasi-likelihood, depending on the distributional properties of the model (see the section "Default Estimation Techniques" on page 3241). The "Model Information" table tells you which estimation method was applied. In GLM mode, the individual observations are considered the sampling units. This has bearing, for example, on how sandwich estimators are computed (see the EMPIRICAL option and the section "Empirical Covariance ("Sandwich") Estimators" on page 3213).

In GLMM mode, the procedure assumes that the model contains random effects or possibly correlated errors, or that the data have a clustered structure. PROC GLIMMIX then estimates the parameters by using the techniques specified in the METHOD= option in the PROC GLIMMIX statement.

In general, adding one overdispersion parameter to a generalized linear model does not trigger GLMM mode. For example, the model that is defined by the following statements is fit in GLM mode:

```
proc glimmix;
   model y = x1 x2 / dist=poisson;
   random _residual_;
run;
```

The parameters of the fixed effects are estimated by maximum likelihood, and the covariance matrix of the fixed-effects parameters is adjusted by the overdispersion parameter.

In a model that contains uncorrelated data, you can trigger GLMM mode by specifying the SUBJECT= or GROUP= option in the RANDOM statement. For example, the following statements fit the model by using the residual pseudo-likelihood algorithm:

```
proc glimmix;
   class id;
   model y = x1 x2 / dist=poisson;
   random _residual_ / subject=id;
run;
```

If in doubt, you can determine whether a model was fit in GLM mode or GLMM mode. In GLM mode the "Covariance Parameter Estimates" table is not produced. Scale and dispersion parameters in the model appear in the "Parameter Estimates" table.

Statistical Inference for Covariance Parameters

The Likelihood Ratio Test

The likelihood ratio test (LRT) compares the likelihoods of two models where parameter estimates are obtained in two parameter spaces, the space Ω and the restricted subspace Ω_0. In the GLIMMIX procedure, the full model defines Ω and the *test-specification* in the COVTEST statement determines the null parameter space Ω_0. The likelihood ratio procedure consists of the following steps (see, for example, Bickel and Doksum 1977, p. 210):

1. Find the estimate $\widehat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta} \in \Omega$. Compute the likelihood $L(\widehat{\boldsymbol{\theta}})$.
2. Find the estimate $\widehat{\boldsymbol{\theta}}_0$ of $\boldsymbol{\theta} \in \Omega_0$. Compute the likelihood $L(\widehat{\boldsymbol{\theta}}_0)$.
3. Form the likelihood ratio

$$\overline{\lambda} = \frac{L(\widehat{\boldsymbol{\theta}})}{L(\widehat{\boldsymbol{\theta}}_0)}$$

4. Find a function $f\left(\overline{\lambda}\right)$ that has a known distribution. $f(\cdot)$ serves as the test statistic for the likelihood ratio test.

Please note the following regarding the implementation of these steps in the COVTEST statement of the GLIMMIX procedure.

- The function $f(\cdot)$ in step 4 is always taken to be

$$\lambda = 2\log\left\{\overline{\lambda}\right\}$$

  which is twice the difference between the log likelihoods for the full model and the model under the COVTEST restriction.

- For METHOD=RSPL and METHOD=RMPL, the test statistic is based on the restricted likelihood.
- For GLMMs involving pseudo-data, the test statistics are based on the pseudo-likelihood or the restricted pseudo-likelihood and are based on the final pseudo-data.

- The parameter space Ω for the full model is typically not an unrestricted space. The GLIMMIX procedure imposes boundary constraints for variance components and scale parameters, for example. The specification of the subspace Ω_0 must be consistent with these full-model constraints; otherwise the test statistic λ does not have the needed distribution. You can remove the boundary restrictions with the NOBOUND option in the PROC GLIMMIX statement or the NOBOUND option in the PARMS statement.

One- and Two-Sided Testing, Mixture Distributions

Consider testing the hypothesis $H_0: \theta_i = 0$. If Ω is the open interval $(0, \infty)$, then only a one-sided alternative hypothesis is meaningful,

$$H_0: \theta_i = 0 \quad H_a: \theta_i > 0$$

This is the appropriate set of hypotheses, for example, when θ_i is the variance of a G-side random effect. The positivity constraint on Ω is required for valid conditional and marginal distributions of the data. Verbeke and Molenberghs (2003) refer to this situation as the constrained case.

However, if one focuses on the validity of the marginal distribution alone, then negative values for θ_i might be permissible, provided that the marginal variance remains positive definite. In the vernacular or Verbeke and Molenberghs (2003), this is the unconstrained case. The appropriate alternative hypothesis is then two-sided,

$$H_0: \theta_i = 0 \quad H_a: \theta_i \neq 0$$

Several important issues are connected to the choice of hypotheses. The GLIMMIX procedure by default imposes constraints on some covariance parameters. For example, variances and scale parameters have a lower bound of 0. This implies a constrained setting with one-sided alternatives. If you specify the NOBOUND option in the PROC GLIMMIX statement, or the NOBOUND option in the PARMS statement, the boundary restrictions are lifted from the covariance parameters and the GLIMMIX procedure takes an unconstrained stance in the sense of Verbeke and Molenberghs (2003). The alternative hypotheses for variance components are then two-sided.

When $H_0: \theta_i = 0$ and $\Omega = (0, \infty)$, the value of θ_i under the null hypothesis is on the boundary of the parameter space. The distribution of the likelihood ratio test statistic λ is then nonstandard. In general, it is a mixture of distributions, and in certain special cases, it is a mixture of central chi-square distributions. Important contributions to the understanding of the asymptotic behavior of the likelihood ratio and score test statistic in this situation have been made by, for example, Self and Liang (1987); Shapiro (1988); Silvapulle and Silvapulle (1995). Stram and Lee (1994, 1995) applied the results of Self and Liang (1987) to likelihood ratio testing in the mixed model with uncorrelated errors. Verbeke and Molenberghs (2003) compared the score and likelihood ratio tests in random effects models with unstructured $\mathbf{G}$ matrix and provide further results on mixture distributions.

The GLIMMIX procedure recognizes the following special cases in the computation of p-values ($\widehat{\lambda}$ denotes the realized value of the test statistic). Notice that the probabilities of general chi-square mixture distributions do not equal linear combination of central chi-square probabilities (Davis 1977; Johnson, Kotz, and Balakrishnan 1994, Section 18.8).

1. ν parameters are tested, and neither parameters specified under H_0 nor nuisance parameters are on the boundary of the parameters space (Case 4 in Self and Liang 1987). The p-value is computed by the classical result:

$$p = \Pr\left(\chi^2_\nu \geq \widehat{\lambda}\right)$$

2. One parameter is specified under H_0 and it falls on the boundary. No other parameters are on the boundary (Case 5 in Self and Liang 1987).

$$p = \begin{cases} 1 & \widehat{\lambda} = 0 \\ 0.5 \Pr\left(\chi^2_1 \geq \widehat{\lambda}\right) & \widehat{\lambda} > 0 \end{cases}$$

Note that this implies a 50:50 mixture of a χ^2_0 and a χ^2_1 distribution. This is also Case 1 in Verbeke and Molenberghs (2000, p. 69).

3. Two parameters are specified under H_0, and one falls on the boundary. No nuisance parameters are on the boundary (Case 6 in Self and Liang 1987).

$$p = 0.5 \Pr\left(\chi^2_1 \geq \widehat{\lambda}\right) + 0.5 \Pr\left(\chi^2_2 \geq \widehat{\lambda}\right)$$

A special case of this scenario is the addition of a random effect to a model with a single random effect and unstructured covariance matrix (Case 2 in Verbeke and Molenberghs 2000, p. 70).

4. Removing j random effects from $j + k$ uncorrelated random effects (Verbeke and Molenberghs 2003).

$$p = 2^{-j} \sum_{i=0}^{j} \left(\begin{array}{c} j \\ i \end{array} \right) \Pr\left(\chi^2_i \geq \widehat{\lambda}\right)$$

Note that this case includes the case of testing a single random effects variance against zero, which leads to a 50:50 mixture of a χ^2_0 and a χ^2_1 as in 2.

5. Removing a random effect from an unstructured **G** matrix (Case 3 in Verbeke and Molenberghs 2000, p. 71).

$$p = 0.5 \Pr\left(\chi^2_k \geq \widehat{\lambda}\right) + 0.5 \Pr\left(\chi^2_{k-1} \geq \widehat{\lambda}\right)$$

where k is the number of random effects (columns of **G**) in the full model. Case 5 in Self and Liang (1987) describes a special case.

When the GLIMMIX procedure determines that estimates of nuisance parameters (parameters not specified under H_0) fall on the boundary, no mixture results are computed.

You can request that the procedure not use mixtures with the CLASSICAL option in the COVTEST statement. If mixtures are used, the Note column of the "Likelihood Ratio Tests of Covariance Parameters" table contains the "MI" entry. The "DF" entry is used when PROC GLIMMIX determines that the standard computation of p-values is appropriate. The "–" entry is used when the classical computation was used because the testing and model scenario does not match one of the special cases described previously.

Handling the Degenerate Distribution

Likelihood ratio testing in mixed models invariably involves the chi-square distribution with zero degrees of freedom. The χ^2_0 random variable is degenerate at 0, and it occurs in two important circumstances. First, it is a component of mixtures, where typically the value of the test statistic is not zero. In that case, the contribution of the χ^2_0 component of the mixture to the p-value is nil. Second, a degenerate distribution of the test statistic occurs when the null model is identical to the full model—for example, if you test a hypothesis that does not impose any (new) constraints on the parameter space. The following statements test whether the **R** matrix in a variance component model is diagonal:

```
proc glimmix;
   class a b;
   model y = a;
   random b a*b;
   covtest diagR;
run;
```

Because no R-side covariance structure is specified (all random effects are G-side effects), the $\mathbf{R}$ matrix is diagonal in the full model and the COVTEST statement does not impose any further restriction on the parameter space. The likelihood ratio test statistic is zero. The GLIMMIX procedure computes the p-value as the probability to observe a value at least as large as the test statistic under the null hypothesis. Hence,

$$p = \Pr(\chi^2_0 \geq 0) = 1$$

Wald Versus Likelihood Ratio Tests

The Wald test and the likelihood ratio tests are asymptotic tests, meaning that the distribution from which p-values are calculated for a finite number of samples draws on the distribution of the test statistic as the sample size grows to infinity. The Wald test is a simple test that is easy to compute based only on parameter estimates and their (asymptotic) standard errors. The likelihood ratio test, on the other hand, requires the likelihoods of the full model and the model reduced under H_0. It is computationally more demanding, but also provides the asymptotically more powerful and reliable test. The likelihood ratio test is almost always preferable to the Wald test, unless computational demands make it impractical to refit the model.

Confidence Bounds Based on Likelihoods

Families of statistical tests can be inverted to produce confidence limits for parameters. The confidence region for parameter θ is the set of values for which the corresponding test fails to reject $H: \theta = \theta_0$. When parameters are estimated by maximum likelihood or a likelihood-based technique, it is natural to consider the likelihood ratio test statistic for H in the test inversion. When there are multiple parameters in the model, however, you need to supply values for these nuisance parameters during the test inversion as well.

In the following, suppose that $\boldsymbol{\theta}$ is the covariance parameter vector and that one of its elements, θ, is the parameter of interest for which you want to construct a confidence interval. The other elements of $\boldsymbol{\theta}$ are collected in the nuisance parameter vector $\boldsymbol{\theta}_2$. Suppose that $\widehat{\boldsymbol{\theta}}$ is the estimate of $\boldsymbol{\theta}$ from the overall optimization and that $L(\widehat{\boldsymbol{\theta}})$ is the likelihood evaluated at that estimate. If estimation is based on pseudo-data, then $L(\widehat{\boldsymbol{\theta}})$ is the pseudo-likelihood based on the final pseudo-data. If estimation uses a residual (restricted) likelihood, then L denotes the restricted maximum likelihood and $\widehat{\boldsymbol{\theta}}$ is the REML estimate.

Profile Likelihood Bounds

The likelihood ratio test statistic for testing $H: \theta = \theta_0$ is

$$2\left\{\log\left\{L(\widehat{\boldsymbol{\theta}})\right\} - \log\left\{L(\theta_0, \widehat{\boldsymbol{\theta}}_2)\right\}\right\}$$

where $\widehat{\boldsymbol{\theta}}_2$ is the likelihood estimate of $\boldsymbol{\theta}_2$ under the restriction that $\theta = \theta_0$. To invert this test, a function is defined that returns the maximum likelihood for a fixed value of θ by seeking the maximum over the remaining parameters. This function is termed the profile likelihood (Pawitan 2001, Ch. 3.4),

$$\lambda_p = L(\boldsymbol{\theta}_2|\widetilde{\theta}) = \sup_{\boldsymbol{\theta}_2} L(\widetilde{\theta}, \boldsymbol{\theta}_2)$$

In computing λ_p, θ is fixed at $\widetilde{\theta}$ and $\boldsymbol{\theta}_2$ is estimated. In mixed models, this step typically requires a separate, iterative optimization to find the estimate of $\boldsymbol{\theta}_2$ while θ is held fixed. The $(1-\alpha)\times 100\%$ profile likelihood confidence interval for θ is then defined as the set of values for $\widetilde{\theta}$ that satisfy

$$2\left\{\log\left\{L(\widehat{\boldsymbol{\theta}})\right\} - \log\left\{L(\boldsymbol{\theta}_2|\widetilde{\theta})\right\}\right\} \leq \chi^2_{1,(1-\alpha)}$$

The GLIMMIX procedure seeks the values $\widetilde{\theta}_l$ and $\widetilde{\theta}_u$ that mark the endpoints of the set around $\widehat{\theta}$ that satisfy the inequality. The values ($\widetilde{\theta}_l$ and $\widetilde{\theta}_u$) are then called the $(1-\alpha)\times 100\%$ confidence bounds for θ. Note that the GLIMMIX procedure assumes that the confidence region is not disjoint and relies on the convexity of $L(\widehat{\boldsymbol{\theta}})$.

It is not always possible to find values $\widetilde{\theta}_l$ and $\widetilde{\theta}_u$ that satisfy the inequalities. For example, when the parameter space is $(0,\infty)$ and

$$2\left\{\log\left\{L(\widehat{\boldsymbol{\theta}})\right\} - \log\left\{L(\boldsymbol{\theta}_2|0)\right\}\right\} > \chi^2_{1,(1-\alpha)}$$

a lower bound cannot be found at the desired confidence level. The GLIMMIX procedure reports the right-tail probabilities that are achieved by the underlying likelihood ratio statistic separately for lower and upper bounds.

Effect of Scale Parameter

When a scale parameter ϕ is eliminated from the optimization by profiling from the likelihood, some parameters might be expressed as ratios with ϕ in the optimization. This is the case, for example, in variance component models. The profile likelihood confidence bounds are reported on the scale of the parameter in the overall optimization. In case parameters are expressed as ratios with ϕ or functions of ϕ, the column RatioEstimate is added to the "Covariance Parameter Estimates" table. If parameters are expressed as ratios with ϕ and you want confidence bounds for the unscaled parameter, you can prevent profiling of ϕ from the optimization with the NOPROFILE option in the PROC GLIMMIX statement, or choose estimated likelihood confidence bounds with the TYPE=ELR suboption of the CL option in the COVTEST statement. Note that the NOPROFILE option is automatically in effect with METHOD=LAPLACE and METHOD=QUAD.

Estimated Likelihood Bounds

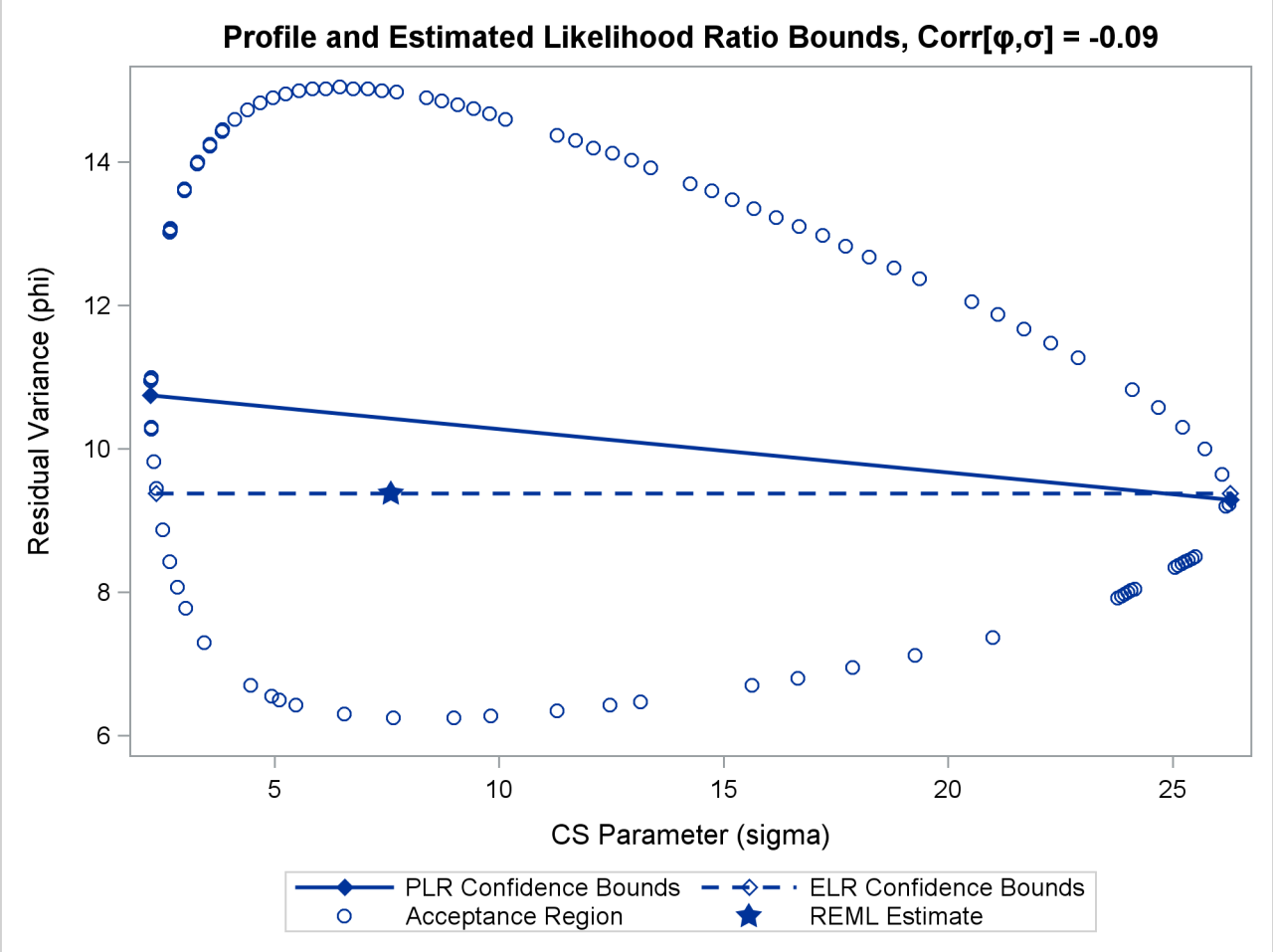
Computing profile likelihood ratio confidence bounds can be computationally expensive, because of the need to repeatedly estimate $\boldsymbol{\theta}_2$ in a constrained optimization. A computationally simpler method to construct confidence bounds from likelihood-based quantities is to use the estimated likelihood (Pawitan 2001, Ch. 10.7) instead of the profile likelihood. An estimated likelihood technique replaces the nuisance parameters in the test inversion with some other estimate. If you choose the TYPE=ELR suboption of the CL option in the COVTEST statement, the GLIMMIX procedure holds the nuisance parameters fixed at the likelihood estimates. The estimated likelihood statistic for inversion is then

$$\lambda_e = L(\widetilde{\theta}, \widehat{\boldsymbol{\theta}}_2)$$

where $\widehat{\boldsymbol{\theta}}_2$ are the elements of $\widehat{\boldsymbol{\theta}}$ that correspond to the nuisance parameters. As the values of $\widetilde{\theta}$ are varied, no reestimation of $\boldsymbol{\theta}_2$ takes place. Although computationally more economical, estimated likelihood intervals do not take into account the variability associated with the nuisance parameters. Their coverage can be satisfactory if the parameter of interest is not (or only weakly) correlated with the nuisance parameters. Estimated likelihood ratio intervals can fall short of the nominal coverage otherwise.

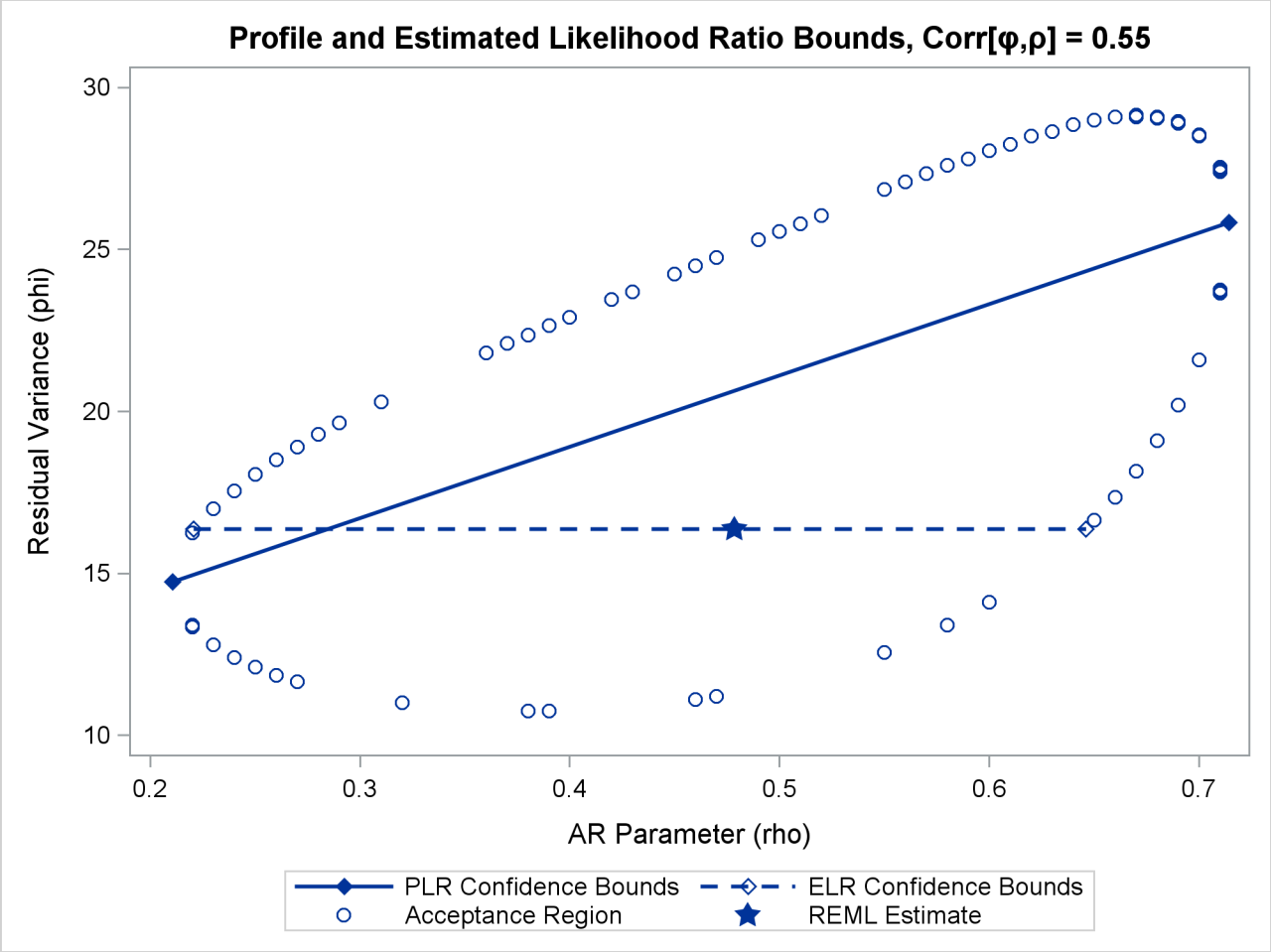
Figure 44.11 depicts profile and estimated likelihood ratio intervals for the parameter σ in a two-parameter compound-symmetric model, $\boldsymbol{\theta} = [\sigma, \phi]'$, in which the correlation between the covariance parameters is small. The elliptical shape traces the set of values for which the likelihood ratio test rejects the hypothesis of equality with the solution. The interior of the ellipse is the "acceptance" region of the test. The solid and dashed lines depict the PLR and ELR confidence limits for σ, respectively. Note that both confidence limits intersect the ellipse and that the ELR interval passes through the REML estimate of ϕ. The PLR bounds are found as those points intersecting the ellipse, where ϕ equals the constrained REML estimate.

Figure 44.11 PLR and ELR Intervals, Small Correlation between Parameters



The major axes of the ellipse in Figure 44.11 are nearly aligned with the major axes of the coordinate system. As a consequence, the line connecting the PLR bounds passes close to the REML estimate in the full model. As a result, ELR bounds will be similar to PLR bounds. Figure 44.12 displays a different scenario, a two-parameter AR(1) covariance structure with a more substantial correlation between the AR(1) parameter (ρ) and the residual variance (ϕ).

Figure 44.12 PLR and ELR Intervals, Large Correlation between Parameters



The correlation between the parameters yields an acceptance region whose major axes are not aligned with the axes of the coordinate system. The ELR bound for ρ passes through the REML estimate of ϕ from the full model and is much shorter than the PLR interval. The PLR interval aligns with the major axis of the acceptance region; it is the preferred confidence interval.

Degrees of Freedom Methods

Between-Within Degrees of Freedom Approximation

The DDFM=BETWITHIN option divides the residual degrees of freedom into between-subject and within-subject portions. PROC GLIMMIX then determines whether a fixed effect changes within any subject. If so, it assigns within-subject degrees of freedom to the effect; otherwise, it assigns the between-subject degrees of freedom to the effect (Schluchter and Elashoff 1990). If the GLIMMIX procedure does not process the data by subjects, the DDFM=BETWITHIN option has no effect. See the section "Processing by Subjects" on page 3218 for more information.

If multiple within-subject effects contain classification variables, the within-subject degrees of freedom are partitioned into components that correspond to the subject-by-effect interactions.

One exception to the preceding method is the case where you model only R-side covariation with an unstructured covariance matrix (TYPE=UN). In this case, all fixed effects are assigned the between-subject degrees of freedom to provide for better small-sample approximations to the relevant sampling distributions. The DDFM=BETWITHIN method is the default for models with only R-side random effects and a SUBJECT= option.

Containment Degrees of Freedom Approximation

The DDFM=CONTAIN method is carried out as follows: Denote the fixed effect in question as A and search the G-side random effect list for the effects that *syntactically* contain A. For example, the effect B(A) contains A, but the effect C does not, even if it has the same levels as B(A).

Among the random effects that contain A, compute their rank contributions to the $[\mathbf{X}\ \mathbf{Z}]$ matrix (in order). The denominator degrees of freedom that is assigned to A is the smallest of these rank contributions. If no effects are found, the denominator degrees of freedom for A is set equal to the residual degrees of freedom, $n - \mathrm{rank}[\mathbf{X}\ \mathbf{Z}]$. This choice of degrees of freedom is the same as for the tests performed for balanced split-plot designs and should be adequate for moderately unbalanced designs.

NOTE: If you have a $\mathbf{Z}$ matrix with a large number of columns, the overall memory requirements and the computing time after convergence can be substantial for the containment method. In this case, you might want to use a different degrees-of-freedom method, such as DDFM=RESIDUAL, DDFM=NONE, or DDFM=BETWITHIN.

Satterthwaite Degrees of Freedom Approximation

The DDFM=SATTERTHWAITE option in the MODEL statement requests that denominator degrees of freedom in t tests and F tests be computed according to a general Satterthwaite approximation.

The general Satterthwaite approximation computed in PROC GLIMMIX for the test

$$H\colon \mathbf{L}\begin{bmatrix}\widehat{\boldsymbol{\beta}}\\ \widehat{\boldsymbol{\gamma}}\end{bmatrix} = \mathbf{0}$$

is based on the F statistic

$$F = \frac{\begin{bmatrix}\widehat{\boldsymbol{\beta}}\\ \widehat{\boldsymbol{\gamma}}\end{bmatrix}' \mathbf{L}'(\mathbf{L}\mathbf{C}\mathbf{L}')^{-1}\mathbf{L}\begin{bmatrix}\widehat{\boldsymbol{\beta}}\\ \widehat{\boldsymbol{\gamma}}\end{bmatrix}}{r}$$

where $r = \mathrm{rank}(\mathbf{L}\mathbf{C}\mathbf{L}')$ and $\mathbf{C}$ is the approximate variance matrix of $[\widehat{\boldsymbol{\beta}}', \widehat{\boldsymbol{\gamma}}' - \boldsymbol{\gamma}']'$. See the section "Estimated Precision of Estimates" on page 3189 and the section "Aspects Common to Adaptive Quadrature and Laplace Approximation" on page 3197.

The approximation proceeds by first performing the spectral decomposition $\mathbf{L}\mathbf{C}\mathbf{L}' = \mathbf{U}'\mathbf{D}\mathbf{U}$, where $\mathbf{U}$ is an orthogonal matrix of eigenvectors and $\mathbf{D}$ is a diagonal matrix of eigenvalues, both of dimension $r \times r$. Define $\mathbf{b}_j$ to be the jth row of $\mathbf{U}\mathbf{L}$, and let

$$\nu_j = \frac{2(D_j)^2}{\mathbf{g}_j'\mathbf{A}\mathbf{g}_j}$$

where D_j is the jth diagonal element of $\mathbf{D}$ and $\mathbf{g}_j$ is the gradient of $\mathbf{b}_j\mathbf{C}\mathbf{b}_j'$ with respect to $\boldsymbol{\theta}$, evaluated at $\widehat{\boldsymbol{\theta}}$. The matrix $\mathbf{A}$ is the asymptotic variance-covariance matrix of $\widehat{\boldsymbol{\theta}}$, which is obtained from the second

derivative matrix of the likelihood equations. You can display this matrix with the ASYCOV option in the PROC GLIMMIX statement.

Finally, let

$$E = \sum_{j=1}^{r} \frac{\nu_j}{\nu_j - 2} I(\nu_j > 2)$$

where the indicator function eliminates terms for which $\nu_j \leq 2$. The degrees of freedom for F are then computed as

$$\nu = \frac{2E}{E - \mathrm{rank}(\mathbf{L})}$$

provided $E > r$; otherwise ν is set to 0.

In the one-dimensional case, when PROC GLIMMIX computes a t test, the Satterthwaite degrees of freedom for the t statistic

$$t = \frac{\mathbf{l}' \begin{bmatrix} \widehat{\boldsymbol{\beta}} \\ \widehat{\boldsymbol{\gamma}} \end{bmatrix}}{\sqrt{\mathbf{l}'\mathbf{C}\mathbf{l}}}$$

are computed as

$$\nu = \frac{2(\mathbf{l}'\mathbf{C}\mathbf{l})^2}{\mathbf{g}'\mathbf{A}\mathbf{g}}$$

where $\mathbf{g}$ is the gradient of $\mathbf{l}'\mathbf{C}\mathbf{l}$ with respect to $\boldsymbol{\theta}$, evaluated at $\widehat{\boldsymbol{\theta}}$.

The calculation of Satterthwaite degrees of freedom requires extra memory to hold q matrices that are the size of the mixed model equations, where q is the number of covariance parameters. Extra computing time is also required to process these matrices. The implemented Satterthwaite method is intended to produce an accurate F approximation; however, the results can differ from those produced by PROC GLM. Also, the small-sample properties of this approximation have not been extensively investigated for the various models available with PROC GLIMMIX.

Kenward-Roger Degrees of Freedom Approximation

The DDFM=KENWARDROGER option prompts PROC GLIMMIX to compute the denominator degrees of freedom in t tests and F tests by using the approximation described in Kenward and Roger (1997). For inference on the linear combination $L\beta$ in a Gaussian linear model, they propose a scaled Wald statistic

$$\begin{aligned} F^* &= \lambda F \\ &= \frac{\lambda}{l}(\hat{\beta} - \beta)^T L (L^T \hat{\Phi}_A L)^{-1} L^T (\hat{\beta} - \beta), \end{aligned}$$

where $l = \mathrm{rank}(L)$, $\hat{\Phi}_A$ is a bias-adjusted estimator of the precision of $\hat{\beta}$, and $0 < \lambda < 1$. An appropriate $F_{l,m}$ approximation to the sampling distribution of F^* is derived by matching the first two moments of F^* with those from the approximating F distribution and solving for the values of λ and m. The value of m thus derived is the Kenward-Roger degrees of freedom. The precision estimator $\hat{\Phi}_A$ is bias-adjusted, in contrast

to the conventional precision estimator $\Phi(\hat{\sigma}) = (X'V(\hat{\sigma})^{-1}X)^{-1}$, which is obtained by simply replacing σ with $\hat{\sigma}$ in $\Phi(\sigma)$, the asymptotic variance of $\hat{\beta}$. This method uses $\hat{\Phi}_A$ to address the fact that $\Phi(\hat{\sigma})$ is a biased estimator of $\Phi(\sigma)$, and $\Phi(\sigma)$ itself underestimates $\text{var}(\hat{\beta})$ when σ is unknown. This bias-adjusted precision estimator is also discussed in Prasad and Rao (1990); Harville and Jeske (1992); Kackar and Harville (1984).

By default, the observed information matrix of the covariance parameter estimates is used in the calculations. For covariance structures that have nonzero second derivatives with respect to the covariance parameters, the Kenward-Roger covariance matrix adjustment includes a second-order term. This term can result in standard error shrinkage. Also, the resulting adjusted covariance matrix can then be indefinite and is not invariant under reparameterization. The FIRSTORDER suboption of the DDFM=KENWARDROGER option eliminates the second derivatives from the calculation of the covariance matrix adjustment. For scalar estimable functions, the resulting estimator is referred to as the Prasad-Rao estimator $\widetilde{m}^{@}$ in Harville and Jeske (1992). You can use the COVB(DETAILS) option to diagnose the adjustments that PROC GLIMMIX makes to the covariance matrix of fixed-effects parameter estimates. An application with DDFM=KENWARDROGER is presented in Example 44.8. The following are examples of covariance structures that generally lead to nonzero second derivatives: TYPE=ANTE(1), TYPE=AR(1), TYPE=ARH(1), TYPE=ARMA(1,1), TYPE=CHOL, TYPE=CSH, TYPE=FA0(q), TYPE=TOEPH, TYPE=UNR, and all TYPE=SP() structures.

DDFM=KENWARDROGER2 specifies an improved F approximation of the DDFM=KENWARD-ROGER type that uses a less biased precision estimator, as proposed by Kenward and Roger (2009). An important feature of the KR2 precision estimator is that it is invariant under reparameterization within the classes of intrinsically linear and intrinsically linear inverse covariance structures. For the invariance to hold within these two classes of covariance structures, a modified expected Hessian matrix is used in the computation of the covariance matrix of σ. The two cells classified as "Modified" scoring for RxPL estimation in Table 44.21 give the modified Hessian expressions for the cases where the scale parameter is profiled and not profiled. You can enforce the use of the modified expected Hessian matrix by specifying both the EXPHESSIAN and SCOREMOD options in the PROC GLIMMIX statement. Kenward and Roger (2009) note that for an intrinsically linear covariance parameterization, DDFM=KR2 produces the same precision estimator as that obtained using DDFM=KR(FIRSTORDER).

Empirical Covariance ("Sandwich") Estimators

Residual-Based Estimators

The GLIMMIX procedure can compute the classical sandwich estimator of the covariance matrix of the fixed effects, as well as several bias-adjusted estimators. This requires that the model is either an (overdispersed) GLM or a GLMM that can be processed by subjects (see the section "Processing by Subjects" on page 3218).

Consider a statistical model of the form

$$\mathbf{Y} = \boldsymbol{\mu} + \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim (\mathbf{0}, \boldsymbol{\Sigma})$$

The general expression of a sandwich covariance estimator is then

$$c \times \widehat{\boldsymbol{\Omega}} \left(\sum_{i=1}^{m} \mathbf{A}_i \widehat{\mathbf{D}}_i' \widehat{\boldsymbol{\Sigma}}_i^{-1} \mathbf{F}_i' \mathbf{e}_i \mathbf{e}_i' \mathbf{F}_i \widehat{\boldsymbol{\Sigma}}_i^{-1} \widehat{\mathbf{D}}_i \mathbf{A}_i \right) \widehat{\boldsymbol{\Omega}}$$

where $\mathbf{e}_i = \mathbf{y}_i - \widehat{\boldsymbol{\mu}}_i$, $\boldsymbol{\Omega} = (\mathbf{D}'\boldsymbol{\Sigma}^{-1}\mathbf{D})^{-}$.

For a GLMM estimated by one of the pseudo-likelihood techniques that involve linearization, you can make the following substitutions: $\mathbf{Y} \rightarrow \mathbf{P}$, $\boldsymbol{\Sigma} \rightarrow \mathbf{V}(\boldsymbol{\theta})$, $\mathbf{D} \rightarrow \mathbf{X}$, $\widehat{\boldsymbol{\mu}} \rightarrow \mathbf{X}\widehat{\boldsymbol{\beta}}$. These matrices are defined in the section "Pseudo-likelihood Estimation Based on Linearization" on page 3188.

The various estimators computed by the GLIMMIX procedure differ in the choice of the constant c and the matrices $\mathbf{F}_i$ and $\mathbf{A}_i$. You obtain the classical estimator, for example, with $c = 1$, and $\mathbf{F}_i = \mathbf{A}_i$ equal to the identity matrix.

The EMPIRICAL=ROOT estimator of Kauermann and Carroll (2001) is based on the approximation

$$\text{Var}\left[\mathbf{e}_i\mathbf{e}_i'\right] \approx (\mathbf{I} - \mathbf{H}_i)\boldsymbol{\Sigma}_i$$

where $\mathbf{H}_i = \mathbf{D}_i\boldsymbol{\Omega}\mathbf{D}_i'\boldsymbol{\Sigma}_i^{-1}$. The EMPIRICAL=FIRORES estimator is based on the approximation

$$\text{Var}\left[\mathbf{e}_i\mathbf{e}_i'\right] \approx (\mathbf{I} - \mathbf{H}_i)\boldsymbol{\Sigma}_i(\mathbf{I} - \mathbf{H}_i')$$

of Mancl and DeRouen (2001). Finally, the EMPIRICAL=FIROEEQ estimator is based on approximating an unbiased estimating equation (Fay and Graubard 2001). For this estimator, $\mathbf{A}_i$ is a diagonal matrix with entries

$$[\mathbf{A}_i]_{jj} = \left(1 - \min\{r, [\mathbf{Q}]_{jj}\}\right)^{-1/2}$$

where $\mathbf{Q} = \mathbf{D}_i'\widehat{\boldsymbol{\Sigma}}_i^{-1}\mathbf{D}_i\widehat{\boldsymbol{\Omega}}$. The optional number $0 \leq r < 1$ is chosen to provide an upper bound on the correction factor. For $r = 0$, the classical sandwich estimator results. PROC GLIMMIX chooses as default value $r = 3/4$. The diagonal entries of $\mathbf{A}_i$ are then no greater than 2.

Table 44.22 summarizes the components of the computation for the GLMM based on linearization, where m denotes the number of subjects and k is the rank of $\mathbf{X}$.

Table 44.22 Empirical Covariance Estimators for a Linearized GLMM

EMPIRICAL=	c	$\mathbf{A}_i$	$\mathbf{F}_i$
CLASSICAL	1	$\mathbf{I}$	$\mathbf{I}$
DF	$\begin{cases} \frac{m}{m-k} & m > k \\ 1 & \text{otherwise} \end{cases}$	$\mathbf{I}$	$\mathbf{I}$
ROOT	1	$\mathbf{I}$	$(\mathbf{I} - \mathbf{H}_i')^{-1/2}$
FIRORES	1	$\mathbf{I}$	$(\mathbf{I} - \mathbf{H}_i')^{-1}$
FIROEEQ(r)	1	$\text{Diag}\{(1 - \min\{r, [\mathbf{Q}]_{jj}\})^{-1/2}\}$	$\mathbf{I}$

Computation of an empirical variance estimator requires that the data can be processed by independent sampling units. This is always the case in GLMs. In this case, m equals the sum of all frequencies. In GLMMs, the empirical estimators require that the data consist of multiple subjects. In that case, m equals the number of subjects as per the "Dimensions" table. The following section discusses how the GLIMMIX procedure determines whether the data can be processed by subjects. The section "GLM Mode or GLMM Mode" on page 3203 explains how PROC GLIMMIX determines whether a model is fit in GLM mode or in GLMM mode.

Design-Adjusted MBN Estimator

Morel (1989) and Morel, Bokossa, and Neerchal (2003) suggested a bias correction of the classical sandwich estimator that rests on an additive correction of the residual crossproducts and a sample size correction. This estimator is available with the EMPIRICAL=MBN option in the PROC GLIMMIX statement. In the notation of the previous section, the residual-based MBN estimator can be written as

$$\widehat{\boldsymbol{\Omega}}\left(\sum_{i=1}^{m}\widehat{\mathbf{D}}_i'\widehat{\boldsymbol{\Sigma}}_i^{-1}\left(c\mathbf{e}_i\mathbf{e}_i'+\mathbf{B}_i\right)\widehat{\boldsymbol{\Sigma}}_i^{-1}\widehat{\mathbf{D}}_i\right)\widehat{\boldsymbol{\Omega}}$$

where

- $c = (f-1)/(f-k)\times m/(m-1)$ or $c = 1$ when you specify the EMPIRICAL=MBN(NODF) option
- f is the sum of the frequencies
- k equals the rank of $\mathbf{X}$
- $\mathbf{B}_i = \delta_m\phi\widehat{\boldsymbol{\Sigma}}_i$
- $\phi = \max\left\{r, \text{trace}\left(\widehat{\boldsymbol{\Omega}}\mathbf{M}\right)/k^*\right\}$
- $\mathbf{M} = \sum_{i=1}^{m}\widehat{\mathbf{D}}_i'\widehat{\boldsymbol{\Sigma}}_i^{-1}\mathbf{e}_i\mathbf{e}_i'\widehat{\boldsymbol{\Sigma}}_i^{-1}\widehat{\mathbf{D}}_i$
- $k^* = k$ if $m \geq k$, otherwise k^* equals the number of nonzero singular values of $\widehat{\boldsymbol{\Omega}}\mathbf{M}$
- $\delta_m = k/(m-k)$ if $m > (d+1)k$ and $\delta_m = 1/d$ otherwise
- $d \geq 1$ and $0 \leq r \leq 1$ are parameters supplied with the *mbn-options* of the EMPIRICAL=MBN(*mbn-options*) option. The default values are $d = 2$ and $r = 1$. When the NODF option is in effect, the factor c is set to 1.

Rearranging terms, the MBN estimator can also be written as an additive adjustment to a sample-size corrected classical sandwich estimator

$$c\times\widehat{\boldsymbol{\Omega}}\left(\sum_{i=1}^{m}\widehat{\mathbf{D}}_i'\widehat{\boldsymbol{\Sigma}}_i^{-1}\mathbf{e}_i\mathbf{e}_i'\widehat{\boldsymbol{\Sigma}}_i^{-1}\widehat{\mathbf{D}}_i\right)\widehat{\boldsymbol{\Omega}}+\delta_m\phi\widehat{\boldsymbol{\Omega}}$$

Because δ_m is of order m^{-1}, the additive adjustment to the classical estimator vanishes as the number of independent sampling units (subjects) increases. The parameter ϕ is a measure of the design effect (Morel, Bokossa, and Neerchal 2003). Besides good statistical properties in terms of Type I error rates in small-m situations, the MBN estimator also has the desirable property of recovering rank when the number of sampling units is small. If $m < k$, the "meat" piece of the classical sandwich estimator is essentially a sum of rank one matrices. A small number of subjects relative to the rank of $\mathbf{X}$ can result in a loss of rank and subsequent loss of numerator degrees of freedom in tests. The additive MBN adjustment counters the rank exhaustion. You can examine the rank of an adjusted covariance matrix with the COVB(DETAILS) option in the MODEL statement.

When the principle of the MBN estimator is applied to the likelihood-based empirical estimator, you obtain

$$\mathbf{H}(\widehat{\boldsymbol{\alpha}})^{-1}\left(\sum_{i=1}^{m}c\mathbf{g}_i(\widehat{\boldsymbol{\alpha}})\mathbf{g}_i(\widehat{\boldsymbol{\alpha}})'+\mathbf{B}_i\right)\mathbf{H}(\widehat{\boldsymbol{\alpha}})^{-1}$$

where $\mathbf{B}_i = -\delta_m \phi \mathbf{H}_i(\widehat{\boldsymbol{\alpha}})$, and $\mathbf{H}_i(\widehat{\boldsymbol{\alpha}})$ is the second derivative of the log likelihood for the ith sampling unit (subject) evaluated at the vector of parameter estimates, $\widehat{\boldsymbol{\alpha}}$. Also, $\mathbf{g}_i(\widehat{\boldsymbol{\alpha}})$ is the first derivative of the log likelihood for the ith sampling unit. This estimator is computed if you request EMPIRICAL=MBN with METHOD=LAPLACE or METHOD=QUAD.

In terms of adjusting the classical likelihood-based estimator (White 1982), the likelihood MBN estimator can be written as

$$c \times \mathbf{H}(\widehat{\boldsymbol{\alpha}})^{-1} \left(\sum_{i=1}^{m} \mathbf{g}_i(\widehat{\boldsymbol{\alpha}}) \mathbf{g}_i(\widehat{\boldsymbol{\alpha}})' \right) \mathbf{H}(\widehat{\boldsymbol{\alpha}})^{-1} - \delta_m \phi \mathbf{H}(\widehat{\boldsymbol{\alpha}})^{-1}$$

The parameter ϕ is determined as

- $\phi = \max\left\{r, \text{trace}\left(-\mathbf{H}(\widehat{\boldsymbol{\alpha}})^{-1}\mathbf{M}\right)/k^*\right\}$
- $\mathbf{M} = \sum_{i=1}^{m} \mathbf{g}_i(\widehat{\boldsymbol{\alpha}}) \mathbf{g}_i(\widehat{\boldsymbol{\alpha}})'$
- $k^* = k$ if $m \geq k$, otherwise k^* equals the number of nonzero singular values of $-\mathbf{H}(\widehat{\boldsymbol{\alpha}})^{-1}\mathbf{M}$

Exploring and Comparing Covariance Matrices

If you use an empirical (sandwich) estimator with the EMPIRICAL= option in the PROC GLIMMIX statement, the procedure replaces the model-based estimator of the covariance of the fixed effects with the sandwich estimator. This affects aspects of inference, such as prediction standard errors, tests of fixed effects, estimates, contrasts, and so forth. Similarly, if you choose the DDFM=KENWARDROGER degrees-of-freedom method in the MODEL statement, PROC GLIMMIX adjusts the model-based covariance matrix of the fixed effects according to Kenward and Roger (1997) or according to Kackar and Harville (1984) and Harville and Jeske (1992).

In this situation, the COVB(DETAILS) option in the MODEL statement has two effects. The GLIMMIX procedure displays the (adjusted) covariance matrix of the fixed effects and the model-based covariance matrix (the ODS name of the table with the model-based covariance matrix is CovBModelBased). The procedure also displays a table of statistics for the unadjusted and adjusted covariance matrix and for their comparison. The ODS name of this table is CovBDetails.

If the model-based covariance matrix is not replaced with an adjusted estimator, the COVB(DETAILS) option displays the model-based covariance matrix and provides diagnostic measures for it in the CovBDetails table.

The table generated by the COVB(DETAILS) option consists of several sections. See Example 44.8 for an application.

The trace and log determinant of covariance matrices are general scalar summaries that are sometimes used in direct comparisons, or in formulating other statistics, such as the difference of log determinants. The trace simply represents the sum of the variances of all fixed-effects parameters. If a matrix is indefinite, the determinant is reported instead of the log determinant.

The model-based and adjusted covariance matrices should have the same general makeup of eigenvalues. There should not be any negative eigenvalues, and they should have the same numbers of positive and zero eigenvalues. A reduction in rank due to the adjustment is troublesome for aspects of inference. Negative eigenvalues are listed in the table only if they occur, because a covariance matrix should be at least positive

semi-definite. However, the GLIMMIX procedure examines the model-based and adjusted covariance matrix for negative eigenvalues. The condition numbers reported by PROC GLIMMIX for positive (semi-)definite matrices are computed as the ratio of the largest and smallest nonzero eigenvalue. A large condition number reflects poor conditioning of the matrix.

Matrix norms are extensions of the concept of vector norms to measure the "length" of a matrix. The Frobenius norm of an $(n \times m)$ matrix $\mathbf{A}$ is the direct equivalent of the Euclidean vector norm, the square root of the sum of the squared elements,

$$||\mathbf{A}||_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij}^2}$$

The ∞- and 1-norms of matrix $\mathbf{A}$ are the maximum absolute row and column sums, respectively:

$$||\mathbf{A}||_\infty = \max\left\{\sum_{j=1}^{m} |a_{ij}| \; : i = 1, \cdots, n\right\}$$

$$||\mathbf{A}||_1 = \max\left\{\sum_{i=1}^{n} |a_{ij}| \; : j = 1, \cdots, m\right\}$$

These two norms are identical for symmetric matrices.

The "Comparison" section of the CovBDetails table provides several statistics that set the matrices in relationship. The concordance correlation reported by the GLIMMIX procedure is a standardized measure of the closeness of the model-based and adjusted covariance matrix. It is a slight modification of the covariance concordance correlation in Vonesh, Chinchilli, and Pu (1996) and Vonesh and Chinchilli (1997, Ch. 8.3). Denote as $\mathbf{\Omega}$ the $(p \times p)$ model-based covariance matrix and as $\mathbf{\Omega}_a$ the adjusted matrix. Suppose that $\mathbf{K}$ is the matrix obtained from the identity matrix of size p by replacing diagonal elements corresponding to singular rows in $\mathbf{\Omega}$ with zeros. The lower triangular portion of $\mathbf{\Omega}^{-1/2}\mathbf{\Omega}_a\mathbf{\Omega}^{-1/2}$ is stored in vector $\boldsymbol{\omega}$ and the lower triangular portion of $\mathbf{K}$ is stored in vector $\mathbf{k}$. The matrix $\mathbf{\Omega}^{-1/2}$ is constructed from an eigenanalysis of $\mathbf{\Omega}$ and is symmetric. The covariance concordance correlation is then

$$r(\boldsymbol{\omega}) = 1 - \frac{||\boldsymbol{\omega} - \mathbf{k}||^2}{||\boldsymbol{\omega}||^2 + ||\mathbf{k}||^2}$$

This measure is 1 if $\mathbf{\Omega} = \mathbf{\Omega}_a$. If $\boldsymbol{\omega}$ is orthogonal to $\mathbf{k}$, there is total disagreement between the model-based and the adjusted covariance matrix and $r(\boldsymbol{\omega})$ is zero.

The discrepancy function reported by PROC GLIMMIX is computed as

$$d = \log\{|\mathbf{\Omega}|\} - \log\{|\mathbf{\Omega}_a|\} + \text{trace}\{\mathbf{\Omega}_a\mathbf{\Omega}^{-}\} - \text{rank}\{\mathbf{\Omega}\}$$

In diagnosing departures between an assumed covariance structure and $\text{Var}[\mathbf{Y}]$—using an empirical estimator—Vonesh, Chinchilli, and Pu (1996) find that the concordance correlation is useful in detecting gross departures and propose $\lambda = n_s d$ to test the correctness of the assumed model, where n_s denotes the number of subjects.

Processing by Subjects

Some mixed models can be expressed in different but mathematically equivalent ways with PROC GLIMMIX statements. While equivalent statements lead to equivalent statistical models, the data processing and estimation phase can be quite different, depending on how you write the GLIMMIX statements. For example, the particular use of the SUBJECT= option in the RANDOM statement affects data processing and estimation. Certain options are available only when the data are processed by subject, such as the EMPIRICAL option in the PROC GLIMMIX statement.

Consider a GLIMMIX model where variables A and Rep are classification variables with a and r levels, respectively. The following pairs of statements produce the same random-effects structure:

```
class Rep A;
random Rep*A;
```

```
class Rep A;
random intercept / subject=Rep*A;
```

```
class Rep A;
random Rep / subject=A;
```

```
class Rep A;
random A / subject=Rep;
```

In the first case, PROC GLIMMIX does not process the data by subjects because no SUBJECT= option was given. The computation of empirical covariance estimators, for example, will not be possible. The marginal variance-covariance matrix has the same block-diagonal structure as for cases 2–4, where each block consists of the observations belonging to a unique combination of Rep and A. More importantly, the dimension of the $\mathbf{Z}$ matrix of this model will be $n \times ra$, and $\mathbf{Z}$ will be sparse. In the second case, the $\mathbf{Z}_i$ matrix for each of the ra subjects is a vector of ones.

If the data can be processed by subjects, the procedure typically executes faster and requires less memory. The differences can be substantial, especially if the number of subjects is large. Recall that fitting of generalized linear mixed models might be doubly iterative. Small gains in efficiency for any one optimization can produce large overall savings.

If you interpret the intercept as "1," then a RANDOM statement with TYPE=VC (the default) and no SUBJECT= option can be converted into a statement with subject by dividing the random effect by the eventual subject effect. However, the presence of the SUBJECT= option does not imply processing by subject. If a RANDOM statement does not have a SUBJECT= effect, processing by subjects is not possible unless the random effect is a pure R-side overdispersion effect. In the following example, the data will not be processed by subjects, because the first RANDOM statement specifies a G-side component and does not use a SUBJECT= option:

```
proc glimmix;
   class A B;
   model y = B;
   random A;
   random B / subject=A;
run;
```

To allow processing by subjects, you can write the equivalent model with the following statements:

```
proc glimmix;
   class A B;
   model y = B;
   random int / subject=A;
   random B   / subject=A;
run;
```

If you denote a variance component effect X with subject effect S as X–(S), then the "calculus of random effects" applied to the first RANDOM statement reads A = Int*A = Int–(A) = A–(Int). For the second statement there are even more equivalent formulations: A*B = A*B*Int = A*B–(Int) = A–(B) = B–(A) = Int–(A*B).

If there are multiple subject effects, processing by subjects is possible if the effects are equal or contained in each other. Note that in the last example the A*B interaction is a random effect. The following statements give an equivalent specification to the previous model:

```
proc glimmix;
   class A B;
   model y = B;
   random int / subject=A;
   random A   / subject=B;
run;
```

Processing by subjects is not possible in this case, because the two subject effects are not syntactically equal or contained in each other. The following statements depict a case where subject effects are syntactically contained:

```
proc glimmix;
   class A B;
   model y = B;
   random int / subject=A;
   random int / subject=A*B;
run;
```

The A main effect is contained in the A*B interaction. The GLIMMIX procedure chooses as the subject effect for processing the effect that is contained in all other subject effects. In this case, the subjects are defined by the levels of A.

You can examine the "Model Information" and "Dimensions" tables to see whether the GLIMMIX procedure processes the data by subjects and which effect is used to define subjects. The "Model Information" table displays whether the marginal variance matrix is diagonal (GLM models), blocked, or not blocked. The "Dimensions" table tells you how many subjects (=blocks) there are.

Finally, nesting and crossing of interaction effects in subject effects are equivalent. The following two RANDOM statements are equivalent:

```
class Rep A;
random intercept / subject=Rep*A;

class Rep A;
random intercept / subject=Rep(A);
```

Radial Smoothing Based on Mixed Models

The radial smoother implemented with the TYPE=RSMOOTH option in the RANDOM statement is an approximate low-rank thin-plate spline as described in Ruppert, Wand, and Carroll (2003, Chapter 13.4–13.5). The following sections discuss in more detail the mathematical-statistical connection between mixed models and penalized splines and the determination of the number of spline knots and their location as implemented in the GLIMMIX procedure.

From Penalized Splines to Mixed Models

The connection between splines and mixed models arises from the similarity of the penalized spline fitting criterion to the minimization problem that yields the mixed model equations and solutions for $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$. This connection is made explicit in the following paragraphs. An important distinction between classical spline fitting and its mixed model smoothing variant, however, lies in the nature of the spline coefficients. Although they address similar minimization criteria, the solutions for the spline coefficients in the GLIMMIX procedure are the solutions of random effects, not fixed effects. Standard errors of predicted values, for example, account for this source of variation.

Consider the linearized mixed pseudo-model from the section "The Pseudo-model" on page 3188, $\mathbf{P} = \mathbf{X}\boldsymbol{\beta} + \mathbf{Z}\boldsymbol{\gamma} + \boldsymbol{\epsilon}$. One derivation of the mixed model equations, whose solutions are $\widehat{\boldsymbol{\beta}}$ and $\widehat{\boldsymbol{\gamma}}$, is to maximize the joint density of $f(\boldsymbol{\gamma}, \boldsymbol{\epsilon})$ with respect to $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$. This is not a true likelihood problem, because $\boldsymbol{\gamma}$ is not a parameter, but a random vector.

In the special case with $\mathrm{Var}[\boldsymbol{\epsilon}] = \phi\mathbf{I}$ and $\mathrm{Var}[\boldsymbol{\gamma}] = \sigma^2\mathbf{I}$, the maximization of $f(\boldsymbol{\gamma}, \boldsymbol{\epsilon})$ is equivalent to the minimization of

$$Q(\boldsymbol{\beta}, \boldsymbol{\gamma}) = \phi^{-1}(\mathbf{p} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\gamma})'(\mathbf{p} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\gamma}) + \sigma^{-2}\boldsymbol{\gamma}'\boldsymbol{\gamma}$$

Now consider a linear spline as in Ruppert, Wand, and Carroll (2003, p. 108),

$$p_i = \beta_0 + \beta_1 x_i + \sum_{j=1}^{K} \gamma_j (x_i - t_j)_+$$

where the γ_j denote the spline coefficients at knots $t_1, \cdots, t_K$. The truncated line function is defined as

$$(x - t)_+ = \begin{cases} x - t & x > t \\ 0 & \text{otherwise} \end{cases}$$

If you collect the intercept and regressor x into the matrix $\mathbf{X}$, and if you collect the truncated line functions into the $(n \times K)$ matrix $\mathbf{Z}$, then fitting the linear spline amounts to minimization of the penalized spline criterion

$$Q^*(\boldsymbol{\beta}, \boldsymbol{\gamma}) = (\mathbf{p} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\gamma})'(\mathbf{p} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\gamma}) + \lambda^2\boldsymbol{\gamma}'\boldsymbol{\gamma}$$

where λ is the smoothing parameter.

Because minimizing $Q^*(\boldsymbol{\beta}, \boldsymbol{\gamma})$ with respect to $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ is equivalent to minimizing $Q^*(\boldsymbol{\beta}, \boldsymbol{\gamma})/\phi$, both problems lead to the same solution, and $\lambda = \phi/\sigma$ is the smoothing parameter. The mixed model formulation of spline smoothing has the advantage that the smoothing parameter is selected "automatically." It is a function of the covariance parameter estimates, which, in turn, are estimated according to the method you specify with the METHOD= option in the PROC GLIMMIX statement.

To accommodate nonnormal responses and general link functions, the GLIMMIX procedure uses $\mathrm{Var}[\boldsymbol{\epsilon}] = \phi\widetilde{\boldsymbol{\Delta}}^{-1}\mathbf{A}\widetilde{\boldsymbol{\Delta}}^{-1}$, where $\mathbf{A}$ is the matrix of variance functions and $\boldsymbol{\Delta}$ is the diagonal matrix of mean derivatives defined earlier. The correspondence between spline smoothing and mixed modeling is then one between a weighted linear mixed model and a weighted spline. In other words, the minimization criterion that yields the estimates $\widehat{\boldsymbol{\beta}}$ and solutions $\widehat{\boldsymbol{\gamma}}$ is then

$$Q(\boldsymbol{\beta},\boldsymbol{\gamma}) = \phi^{-1}(\mathbf{p} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\gamma})'\widetilde{\boldsymbol{\Delta}}\mathbf{A}^{-1}\widetilde{\boldsymbol{\Delta}}(\mathbf{p} - \mathbf{X}\boldsymbol{\beta} - \mathbf{Z}\boldsymbol{\gamma})' + \sigma^{-2}\boldsymbol{\gamma}'\boldsymbol{\gamma}$$

If you choose the TYPE=RSMOOTH covariance structure, PROC GLIMMIX chooses radial basis functions as the spline basis and transforms them to approximate a thin-plate spline as in Chapter 13.4 of Ruppert, Wand, and Carroll (2003). For computational expediency, the number of knots is chosen to be less than the number of data points. Ruppert, Wand, and Carroll (2003) recommend one knot per every four unique regressor values for one-dimensional smoothers. In the multivariate case, general recommendations are more difficult, because the optimal number and placement of knots depend on the spatial configuration of samples. Their recommendation for a bivariate smoother is one knot per four samples, but at least 20 and no more than 150 knots (Ruppert, Wand, and Carroll 2003, p. 257).

The magnitude of the variance component σ^2 depends on the metric of the random effects. For example, if you apply radial smoothing in time, the variance changes if you measure time in days or minutes. If the solution for the variance component is near zero, then a rescaling of the random effect data can help the optimization problem by moving the solution for the variance component away from the boundary of the parameter space.

Knot Selection

The GLIMMIX procedure computes knots for low-rank smoothing based on the vertices or centroids of a k-d tree. The default is to use the vertices of the tree as the knot locations, if you use the TYPE=RSMOOTH covariance structure. The construction of this tree amounts to a partitioning of the random regressor space until all partitions contain at most b observations. The number b is called the *bucket size* of the k-d tree. You can exercise control over the construction of the tree by changing the bucket size with the BUCKET= suboption of the KNOTMETHOD=KDTREE option in the RANDOM statement. A large bucket size leads to fewer knots, but it is not correct to assume that K, the number of knots, is simply $\lfloor n/b \rfloor$. The number of vertices depends on the configuration of the values in the regressor space. Also, coordinates of the bounding hypercube are vertices of the tree. In the one-dimensional case, for example, the extreme values of the random effect are vertices.

To demonstrate how the k-d tree partitions the random-effects space based on observed data and the influence of the bucket size, consider the following example from Chapter 59, "The LOESS Procedure." The SAS data set Gas contains the results of an engine exhaust emission study (Brinkman 1981). The covariate in this analysis, E, is a measure of the air-fuel mixture richness. The response, NOx, measures the nitric oxide concentration (in micrograms per joule, and normalized).

```
data Gas;
   input NOx E;
   format NOx E f5.3;
   datalines;
 4.818  0.831
 2.849  1.045
 3.275  1.021
 4.691  0.97
```

```
 4.255  0.825
 5.064  0.891
 2.118  0.71
 4.602  0.801
 2.286  1.074
 0.97   1.148
 3.965  1
 5.344  0.928
 3.834  0.767
 1.99   0.701
 5.199  0.807
 5.283  0.902
 3.752  0.997
 0.537  1.224
 1.64   1.089
 5.055  0.973
 4.937  0.98
 1.561  0.665
;
```

There are 22 observations in the data set, and the values of the covariate are unique. If you want to smooth these data with a low-rank radial smoother, you need to choose the number of knots, as well as their placement within the support of the variable E. The k-d tree construction depends on the observed values of the variable E; it is independent of the values of nitric oxide in the data. The following statements construct a tree based on a bucket size of $b = 11$ and display information about the tree and the selected knots:

```
ods select KDtree KnotInfo;
proc glimmix data=gas nofit;
   model NOx = e;
   random e  / type=rsmooth
               knotmethod=kdtree(bucket=11 treeinfo knotinfo);
run;
```

The NOFIT option prevents the GLIMMIX procedure from fitting the model. This option is useful if you want to investigate the knot construction for various bucket sizes. The TREEINFO and KNOTINFO suboptions of the KNOTMETHOD=KDTREE option request displays of the k-d tree and the knot coordinates derived from it. Construction of the tree commences by splitting the data in half. For $b = 11$, $n = 22$, neither of the two splits contains more than b observations and the process stops. With a single split value, and the two extreme values, the tree has two terminal nodes and leads to three knots (Figure 44.13). Note that for one-dimensional problems, vertices of the k-d tree always coincide with data values.

Figure 44.13 K-d Tree and Knots for Bucket Size 11

The GLIMMIX Procedure

kd-Tree for RSmooth(E)

Node Number	Left Child	Right Child	Split Direction	Split Value
0	1	2	E	0.9280
1			TERMINAL	
2			TERMINAL	

Figure 44.13 *continued*

Radial Smoother Knots for RSmooth(E)

Knot Number	E
1	0.6650
2	0.9280
3	1.2240

If the bucket size is reduced to $b = 8$, the following statements produce the tree and knots in Figure 44.14:

```
ods select KDtree KnotInfo;
proc glimmix data=gas nofit;
   model NOx = e;
   random e / type=rsmooth
              knotmethod=kdtree(bucket=8 treeinfo knotinfo);
run;
```

The initial split value of 0.9280 leads to two sets of 11 observations. In order to achieve a partition into cells that contain at most eight observations, each initial partition is split at its median one more time. Note that one split value is greater and one split value is less than 0.9280.

Figure 44.14 K-d Tree and Knots for Bucket Size 8

The GLIMMIX Procedure

kd-Tree for RSmooth(E)

Node Number	Left Child	Right Child	Split Direction	Split Value
0	1	2	E	0.9280
1	3	4	E	0.8070
2	5	6	E	1.0210
3			TERMINAL	
4			TERMINAL	
5			TERMINAL	
6			TERMINAL	

Radial Smoother Knots for RSmooth(E)

Knot Number	E
1	0.6650
2	0.8070
3	0.9280
4	1.0210
5	1.2240

A further reduction in bucket size to $b = 4$ leads to the tree and knot information shown in Figure 44.15.

Figure 44.15 K-d Tree and Knots for Bucket Size 4

The GLIMMIX Procedure

kd-Tree for RSmooth(E)

Node Number	Left Child	Right Child	Split Direction	Split Value
0	1	2	E	0.9280
1	3	4	E	0.8070
2	9	10	E	1.0210
3	5	6	E	0.7100
4	7	8	E	0.8910
5			TERMINAL	
6			TERMINAL	
7			TERMINAL	
8			TERMINAL	
9	11	12	E	0.9800
10	13	14	E	1.0890
11			TERMINAL	
12			TERMINAL	
13			TERMINAL	
14			TERMINAL	

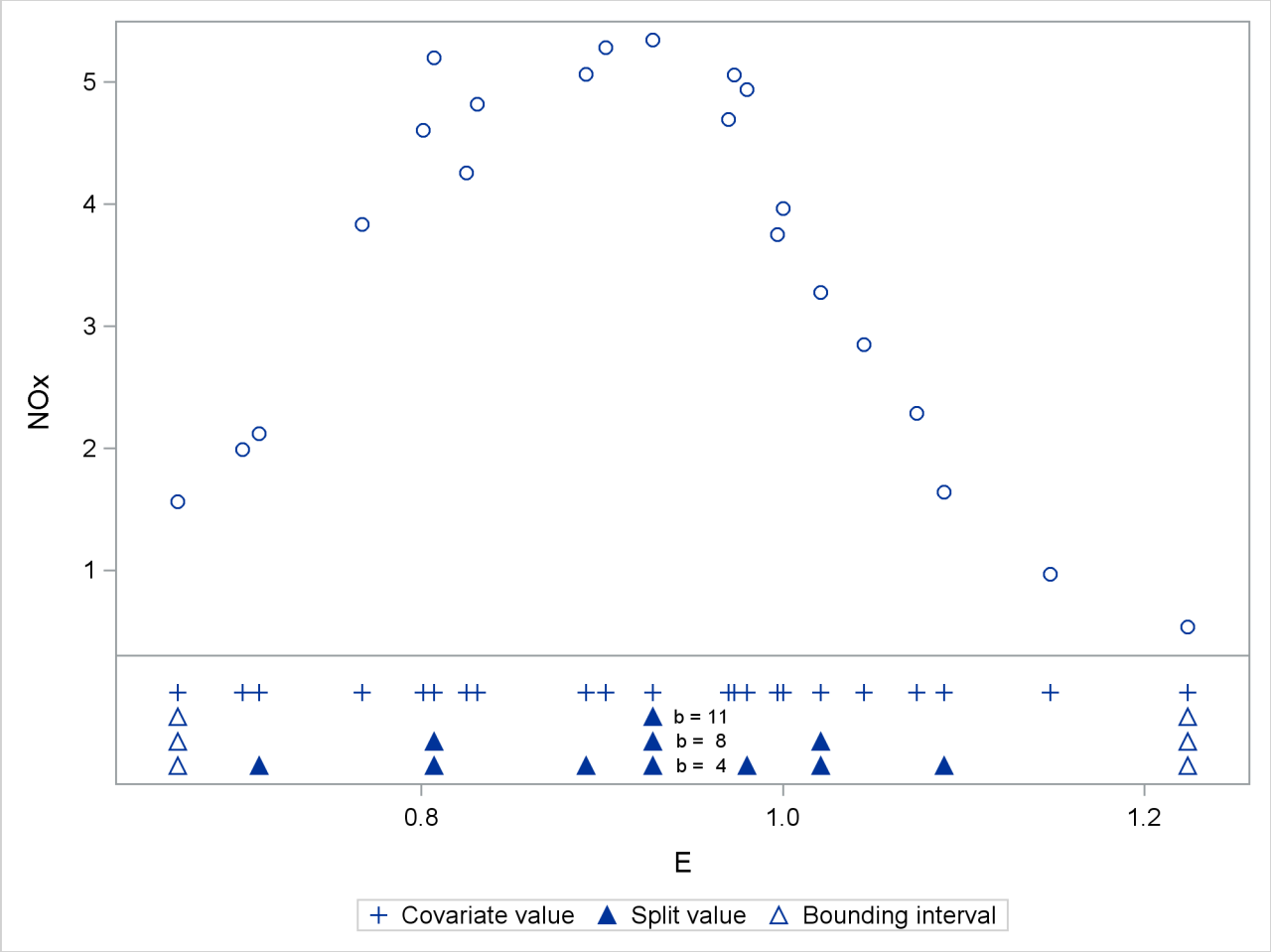
Radial Smoother Knots for RSmooth(E)

Knot Number	E
1	0.6650
2	0.7100
3	0.8070
4	0.8910
5	0.9280
6	0.9800
7	1.0210
8	1.0890
9	1.2240

The split value for $b = 11$ is also a split value for $b = 8$, the split values for $b = 8$ are a subset of those for $b = 4$, and so forth. Figure 44.16 displays the data and the location of split values for the three cases. For a one-dimensional problem (a univariate smoother), the vertices comprise the split values and the values on the bounding interval.

You might want to move away from the boundary, in particular for an irregular data configuration or for multivariate smoothing. The KNOTTYPE=CENTER suboption of the KNOTMETHOD= option chooses centroids of the leaf node cells instead of vertices. This tends to move the outer knot locations closer to the convex hull, but not necessarily to data locations. In the emission example, choosing a bucket size of $b = 11$ and centroids as knot locations yields two knots at E=0.7956 and E=1.076. If you choose the NEAREST suboption, then the nearest neighbor of a vertex or centroid will serve as the knot location. In this case, the knot locations are a subset of the data locations, regardless of the dimension of the smooth.

Figure 44.16 Vertices of k-d Trees for Various Bucket Sizes



Odds and Odds Ratio Estimation

In models with a logit, generalized logit, or cumulative logit link, you can obtain estimates of odds ratios through the ODDSRATIO options in the PROC GLIMMIX, LSMEANS, and MODEL statements. This section provides details about the computation and interpretation of the computed quantities. Note that for these link functions the EXP option in the ESTIMATE and LSMESTIMATE statements also produces odds or odds ratios.

Consider first a model with a dichotomous outcome variable, linear predictor $\eta = \mathbf{x}'\boldsymbol{\beta} + \mathbf{z}'\boldsymbol{\gamma}$, and logit link function. Suppose that η_0 represents the linear predictor for a condition of interest. For example, in a simple logistic regression model with $\eta = \alpha + \beta x$, η_0 might correspond to the linear predictor at a particular value of the covariate—say, $\eta_0 = \alpha + \beta x_0$.

The modeled probability is $\pi = 1/(1 + \exp\{-\eta\})$, and the odds for $\eta = \eta_0$ are

$$\frac{\pi_0}{1-\pi_0} = \frac{1/(1+\exp\{-\eta_0\})}{\exp\{-\eta_0\}/(1+\exp\{-\eta_0\})} = \exp\{\eta_0\}$$

Because η_0 is a logit, it represents the log odds. The odds ratio $\psi(\eta_1, \eta_0)$ is defined as the ratio of odds for η_1 and η_0,

$$\psi(\eta_1, \eta_0) = \exp\{\eta_1 - \eta_0\}$$

The odds ratio compares the odds of the outcome under the condition expressed by η_1 to the odds under the condition expressed by η_0. In the preceding simple logistic regression example, this ratio equals $\exp\{\beta(x_1 - x_0)\}$. The exponentiation of the estimate of β is thus an estimate of the odds ratio comparing conditions for which $x_1 - x_0 = 1$. If x and $x + 1$ represent standard and experimental conditions, for example, $\exp\{\beta\}$ compares the odds of the outcome under the experimental condition to the odds under the standard condition. For many other types of models, odds ratios can be expressed as simple functions of parameter estimates. For example, suppose you are fitting a logistic model with a single classification effect with three levels:

```
proc glimmix;
   class A;
   model y = A / dist=binary;
run;
```

The estimated linear predictor for level j of A is $\widehat{\eta}_j = \widehat{\beta} + \widehat{\alpha}_j$, $j = 1, 2, 3$. Because the $\mathbf{X}$ matrix is singular in this model due to the presence of an overall intercept, the solution for the intercept estimates $\beta + \alpha_3$, and the solution for the jth treatment effect estimates $\alpha_j - \alpha_3$. Exponentiating the solutions for α_1 and α_2 thus produces odds ratios comparing the odds for these levels against the third level of A.

Results designated as odds or odds ratios in the GLIMMIX procedure might reduce to simple exponentiations of solutions in the "Parameter Estimates" table, but they are computed by a different mechanism if the model contains classification variables. The computations rely on general estimable functions; for the MODEL, LSMEANS, and LSMESTIMATE statements, these functions are based on least squares means. This enables you to obtain odds ratio estimates in more complicated models that involve main effects and interactions, including interactions between continuous and classification variables.

In all cases, the results represent the exponentiation of a linear function of the fixed-effects parameters, $\eta = \mathbf{l}'\boldsymbol{\beta}$. If L_η and U_η are the confidence limits for η on the logit scale, confidence limits for the odds or the odds ratio are obtained as $\exp\{L_\eta\}$ and $\exp\{U_\eta\}$.

The Odds Ratio Estimates Table

This table is produced by the ODDSRATIO option in the MODEL statement. It consists of estimates of odds ratios and their confidence limits. Odds ratios are produced for the following:

- classification main effects, if they appear in the MODEL statement
- continuous variables in the MODEL statement, unless they appear in an interaction with a classification effect
- continuous variables in the MODEL statement at fixed levels of a classification effect, if the MODEL statement contains an interaction of the two.
- continuous variables in the MODEL statements if they interact with other continuous variables

The Default Table

Consider the following PROC GLIMMIX statements that fit a logistic model with one classification effect, one continuous variable, and their interaction (the ODDSRATIO option in the MODEL statement requests the "Odds Ratio Estimates" table).

```
proc glimmix;
   class A;
   model y = A x A*x / dist=binary oddsratio;
run;
```

By default, odds ratios are computed as follows:

- The covariate is set to its average, $\overline{x}$, and the least squares means for the A effect are obtained. Suppose $\mathbf{L}^{(1)}$ denotes the matrix of coefficients defining the estimable functions that produce the a least squares means $\mathbf{L}\widehat{\boldsymbol{\beta}}$, and $\mathrm{l}_j^{(1)}$ denotes the jth row of $\mathbf{L}^{(1)}$. Differences of the least squares means against the last level of the A factor are computed and exponentiated:

$$\begin{aligned}
\psi(A_1, A_a) &= \exp\left\{\left(\mathrm{l}_1^{(1)} - \mathrm{l}_a^{(1)}\right)\widehat{\boldsymbol{\beta}}\right\} \\
\psi(A_2, A_a) &= \exp\left\{\left(\mathrm{l}_2^{(1)} - \mathrm{l}_a^{(1)}\right)\widehat{\boldsymbol{\beta}}\right\} \\
&\vdots \\
\psi(A_{a-1}, A_a) &= \exp\left\{\left(\mathrm{l}_{a-1}^{(1)} - \mathrm{l}_a^{(1)}\right)\widehat{\boldsymbol{\beta}}\right\}
\end{aligned}$$

  The differences are checked for estimability. Notice that this set of odds ratios can also be obtained with the following LSMESTIMATE statement (assuming A has five levels):

```
lsmestimate A 1  0  0  0 -1,
              0  1  0  0 -1,
              0  0  1  0 -1,
              0  0  0  1 -1 / exp cl;
```

  You can also obtain the odds ratios with this LSMEANS statement (assuming the last level of A is coded as 5):

```
lsmeans A / diff=control('5') oddsratio cl;
```

- The odds ratios for the covariate must take into account that x occurs in an interaction with the A effect. A second set of least squares means are computed, where x is set to $\overline{x} + 1$. Denote the coefficients of the estimable functions for this set of least squares means as $\mathbf{L}^{(2)}$. Differences of the least squares means at a given level of factor A are then computed and exponentiated:

$$\begin{aligned}
\psi(A(\overline{x}+1)_1, A(\overline{x})_1) &= \exp\left\{\left(\mathrm{l}_1^{(2)} - \mathrm{l}_1^{(1)}\right)\widehat{\boldsymbol{\beta}}\right\} \\
\psi(A(\overline{x}+1)_2, A(\overline{x})_2) &= \exp\left\{\left(\mathrm{l}_2^{(2)} - \mathrm{l}_2^{(1)}\right)\widehat{\boldsymbol{\beta}}\right\} \\
&\vdots \\
\psi(A(\overline{x}+1)_a, A(\overline{x})_a) &= \exp\left\{\left(\mathrm{l}_a^{(2)} - \mathrm{l}_a^{(1)}\right)\widehat{\boldsymbol{\beta}}\right\}
\end{aligned}$$

The differences are checked for estimability. If the continuous covariate does not appear in an interaction with the A variable, only a single odds ratio estimate related to x would be produced, relating the odds of a one-unit shift in the regressor from $\overline{x}$.

Suppose you fit a model that contains interactions of continuous variables, as with the following statements:

```
proc glimmix;
   class A;
   model y = A x x*z / dist=binary oddsratio;
run;
```

In the computation of the A least squares means, the continuous effects are set to their means—that is, $\overline{x}$ and $\overline{xz}$. In the computation of odds ratios for x, linear predictors are computed at $x = \overline{x}$, $x*z = \overline{x} \times \overline{z}$ and at $x = \overline{x} + 1$, $x*z = (\overline{x} + 1)\overline{z}$.

Modifying the Default Table, Customized Odds Ratios

Several suboptions of the ODDSRATIO option in the MODEL statement are available to obtain customized odds ratio estimates. For customized odds ratios that cannot be obtained with these suboptions, use the EXP option in the ESTIMATE or LSMESTIMATE statement.

The type of differences constructed when the levels of a classification factor are varied is controlled by the DIFF= suboption. By default, differences against the last level are taken. DIFF=FIRST computes differences from the first level, and DIFF=ALL computes odds ratios based on all pairwise differences.

For continuous variables in the model, you can change both the reference value (with the AT suboption) and the units of change (with the UNIT suboption). By default, a one-unit change from the mean of the covariate is assessed. For example, the following statements produce all pairwise differences for the A factor:

```
proc glimmix;
   class A;
   model y = A x A*x / dist=binary
                       oddsratio(diff=all
                                 at   x=4
                                 unit x=3);
run;
```

The covariate x is set to the reference value $x = 4$ in the computation of the least squares means for the A odds ratio estimates. The odds ratios computed for the covariate are based on differencing this set of least squares means with a set of least squares means computed at $x = 4 + 3$.

Odds or Odds Ratio

The odds ratio is the exponentiation of a difference on the logit scale,

$$\psi(\eta_1, \eta_0) = \exp\{(\mathrm{l}_1 - \mathrm{l}_0)\boldsymbol{\beta}\}$$

and $\exp\{\mathrm{l}_1\boldsymbol{\beta}\}$ and $\exp\{\mathrm{l}_0\boldsymbol{\beta}\}$ are the corresponding odds. If the ODDSRATIO option is specified in a suitable model in the PROC GLIMMIX statement or the individual statements that support the option, odds ratios are computed in the "Odds Ratio Estimates" table (MODEL statement), the "Differences of Least Squares Means" table (LSMEANS / DIFF), and the "Simple Effect Comparisons of Least Squares Means" table (LSMEANS / SLICEDIFF=). Odds are computed in the "Least Squares Means" table.

Odds Ratios in Multinomial Models

The GLIMMIX procedure fits two kinds of models to multinomial data. Models with cumulative link functions apply to ordinal data, and generalized logit models are fit to nominal data. If you model a multinomial response with LINK=CUMLOGIT or LINK=GLOGIT, odds ratio results are available for these models.

In the generalized logit model, you model baseline category logits. By default, the GLIMMIX procedure chooses the last category as the reference category. If your nominal response has J categories, the baseline logit for category j is

$$\log\left\{\pi_j/\pi_J\right\} = \eta_j = \mathbf{x}'\boldsymbol{\beta}_j + \mathbf{z}'\mathbf{u}_j$$

and

$$\pi_j = \frac{\exp\{\eta_j\}}{\sum_{k=1}^{J}\exp\{\eta_k\}}$$
$$\eta_J = 0$$

As before, suppose that the two conditions to be compared are identified with subscripts 1 and 0. The log odds ratio of outcome j versus J for the two conditions is then

$$\begin{aligned}\log\left\{\psi\left(\eta_{j1},\eta_{j0}\right)\right\} &= \log\left\{\frac{\pi_{j1}/\pi_{J1}}{\pi_{j0}/\pi_{J0}}\right\} = \log\left\{\frac{\exp\{\eta_{j1}\}}{\exp\{\eta_{j0}\}}\right\}\\ &= \eta_{j1}-\eta_{j0}\end{aligned}$$

Note that the log odds ratios are again differences on the scale of the linear predictor, but they depend on the response category. The GLIMMIX procedure determines the estimable functions whose differences represent log odds ratios as discussed previously but produces separate estimates for each nonreference response category.

In models for ordinal data, PROC GLIMMIX models the logits of cumulative probabilities. Thus, the estimates on the linear scale represent log cumulative odds. The cumulative logits are formed as

$$\log\left\{\frac{\Pr(Y\le j)}{\Pr(Y>j)}\right\} = \eta_j = \alpha_j + \mathbf{x}'\boldsymbol{\beta} + \mathbf{z}'\boldsymbol{\gamma} = \alpha_j + \tilde{\eta}$$

so that the linear predictor depends on the response category only through the intercepts (cutoffs) $\alpha_1,\cdots,\alpha_{J-1}$. The odds ratio comparing two conditions represented by linear predictors η_{j1} and η_{j0} is then

$$\begin{aligned}\psi\left(\eta_{j1},\eta_{j0}\right) &= \exp\left\{\eta_{j1}-\eta_{j0}\right\}\\ &= \exp\left\{\tilde{\eta}_1-\tilde{\eta}_0\right\}\end{aligned}$$

and is independent of category.

Parameterization of Generalized Linear Mixed Models

PROC GLIMMIX constructs a generalized linear mixed model according to the specifications in the CLASS, MODEL, and RANDOM statements. Each effect in the MODEL statement generates one or more columns

in the matrix **X**, and each G-side effect in the RANDOM statement generates one or more columns in the matrix **Z**. R-side effects in the RANDOM statement do not generate model matrices; they serve only to index observations within subjects. This section shows how the GLIMMIX procedure builds **X** and **Z**. You can output the **X** and **Z** matrices to a SAS data set with the OUTDESIGN= option in the PROC GLIMMIX statement.

The general rules and techniques for parameterization of a linear model are given in "GLM Parameterization of Classification Variables and Effects" on page 387 in Chapter 19, "Shared Concepts and Topics." The following paragraphs discuss how these rules differ in a mixed model, in particular, how parameterization differs between the **X** and the **Z** matrix.

Intercept

By default, all models automatically include a column of 1s in **X** to estimate a fixed-effect intercept parameter. You can use the NOINT option in the MODEL statement to suppress this intercept. The NOINT option is useful when you are specifying a classification effect in the MODEL statement and you want the parameter estimates to be in terms of the (linked) mean response for each level of that effect, rather than in terms of a deviation from an overall mean.

By contrast, the intercept is not included by default in **Z**. To obtain a column of 1s in **Z**, you must specify in the RANDOM statement either the INTERCEPT effect or some effect that has only one level.

Interaction Effects

Often a model includes interaction (crossed) effects. With an interaction, PROC GLIMMIX first reorders the terms to correspond to the order of the variables in the CLASS statement. Thus, B*A becomes A*B if A precedes B in the CLASS statement. Then, PROC GLIMMIX generates columns for all combinations of levels that occur in the data. The order of the columns is such that the rightmost variables in the cross index faster than the leftmost variables. Empty columns (which would contain all 0s) are not generated for **X**, but they are for **Z**.

See Table 19.5 in the section "GLM Parameterization of Classification Variables and Effects" on page 387 in Chapter 19, "Shared Concepts and Topics," for an example of an interaction parameterization.

Nested Effects

Nested effects are generated in the same manner as crossed effects. Hence, the design columns generated by the following two statements are the same (but the ordering of the columns is different):

Note that nested effects are often distinguished from interaction effects by the implied randomization structure of the design. That is, they usually indicate random effects within a fixed-effects framework. The fact that random effects can be modeled directly in the RANDOM statement might make the specification of nested effects in the MODEL statement unnecessary.

See Table 19.6 in the section "GLM Parameterization of Classification Variables and Effects" on page 387 in Chapter 19, "Shared Concepts and Topics," for an example of the parameterization of a nested effect.

Implications of the Non-Full-Rank Parameterization

For models with fixed effects involving classification variables, there are more design columns in **X** constructed than there are degrees of freedom for the effect. Thus, there are linear dependencies among the

columns of **X**. In this event, all of the parameters are not estimable; there is an infinite number of solutions to the mixed model equations. The GLIMMIX procedure uses a generalized inverse (a g_2-inverse, Pringle and Rayner (1971), to obtain values for the estimates (Searle 1971). The solution values are not displayed unless you specify the SOLUTION option in the MODEL statement. The solution has the characteristic that estimates are 0 whenever the design column for that parameter is a linear combination of previous columns. With this parameterization, hypothesis tests are constructed to test linear functions of the parameters that are estimable.

Some procedures (such as the CATMOD and LOGISTIC procedures) reparameterize models to full rank by using restrictions on the parameters. PROC GLM, PROC MIXED, and PROC GLIMMIX do not reparameterize, making the hypotheses that are commonly tested more understandable. See Goodnight (1978b) for additional reasons for not reparameterizing.

Missing Level Combinations

PROC GLIMMIX handles missing level combinations of classification variables in the same manner as PROC GLM and PROC MIXED. These procedures delete fixed-effects parameters corresponding to missing levels in order to preserve estimability. However, PROC GLIMMIX does not delete missing level combinations for random-effects parameters because linear combinations of the random-effects parameters are always predictable. These conventions can affect the way you specify your CONTRAST and ESTIMATE coefficients.

Notes on the EFFECT Statement

Some restrictions and limitations for models that contain constructed effects are in place with the GLIMMIX procedure. Also, you should be aware of some special defaults and handling that apply only when the model contains constructed fixed and/or random effects.

- Constructed effects can be used in the MODEL and RANDOM statements but not to specify SUBJECT= or GROUP= effects.
- Computed variables are not supported in the specification of a constructed effect. All variables needed to form the collection of columns for a constructed effect must be in the data set.
- You cannot use constructed effects that comprise continuous variables or interactions with other constructed effects as the LSMEANS or LSMESTIMATE effect.
- The calculation of quantities that depend on least squares means, such as odds ratios in the "Odds Ratio Estimates" table, is not possible if the model contains fixed effects that consist of more than one constructed effects, unless all constructed effects are of spline type. For example, least squares means computations are not possible in the following model because the MM_AB*cvars effect contains two constructed effects:

```
proc glimmix;
   class A B C;
   effect MM_AB = MM(A B);
   effect cvars = COLLECTION(x1 x2 x3);
   model y = C MM_AB*cvars;
run;
```

- If the MODEL or RANDOM statement contains constructed effects, the default degrees-of-freedom method for mixed models is DDFM=KENWARDROGER. The containment degrees-of-freedom method (DDFM=CONTAIN) is not available in these models.
- If the model contains fixed spline effects, least squares means are computed at the average spline coefficients across the usable data, possibly further averaged over levels of CLASS variables that interact with the spline effects in the model. You can use the AT option in the LSMEANS and LSMESTIMATE statements to construct the splines for particular values of the covariates involved. Consider, for example, the following statements:

```
proc glimmix;
   class A;
   effect spl = spline(x);
   model y = A spl;
   lsmeans A;
   lsmeans A / at means;
   lsmeans A / at x=0.4;
run;
```

  Suppose that the spl effect contributes seven columns $[s_1, \cdots, s_7]$ to the $\mathbf{X}$ matrix. The least squares means coefficients for the spl effect in the first LSMEANS statement are $[\overline{s}_1, \cdots, \overline{s}_7]$ with the averages taken across the observations used in the analysis. The second LSMEANS statement computes the spline coefficient at the average value of x: $[s(\overline{x})_1, \cdots, s(\overline{x})_7]$. The final LSMEANS statement uses $[s(0.4)_1, \cdots, s(0.4)_7]$. Using the AT option for least squares means calculations with spline effects can resolve inestimability issues.

- Using a spline effect with B-spline basis in the RANDOM statement is **not** the same as using a penalized B-spline (P-spline) through the TYPE=PSPLINE option in the RANDOM statement. The following statement constructs a penalized B-spline by using mixed model methodology:

```
random x / type=pspline;
```

  The next set of statements defines a set of B-spline columns in the $\mathbf{Z}$ matrix with uncorrelated random effects and homogeneous variance:

```
effect bspline = spline(x);
random bspline / type=vc;
```

  This does not lead to a properly penalized fit. See the documentation on TYPE=PSPLINE about the construction of penalties for B-splines through the covariance matrix of random effects.

Positional and Nonpositional Syntax for Contrast Coefficients

When you define custom linear hypotheses with the CONTRAST or ESTIMATE statement, the GLIMMIX procedure sets up an $\mathbf{L}$ vector or matrix that conforms to the fixed-effects solutions or the fixed- and random-effects solutions. With the LSMESTIMATE statement, you specify coefficients of the matrix $\mathbf{K}$ that is then converted into a coefficient matrix that conforms to the fixed-effects solutions.

There are two methods for specifying the entries in a coefficient matrix (hereafter simply referred to as the $\mathbf{L}$ matrix), termed the positional and nonpositional methods. In the positional form, and this is the traditional

method, you provide a list of values that occupy the elements of the **L** matrix associated with the effect in question in the order in which the values are listed. For traditional model effects comprising continuous and classification variables, the positional syntax is simpler in some cases (main effects) and more cumbersome in others (interactions). When you work with effects constructed through the experimental EFFECT statement, the nonpositional syntax is essential.

Consider, for example, the following two-way model with interactions where factors A and B have three and two levels, respectively:

```
proc glimmix;
   class a b block;
   model y = a b a*b / ddfm=kr;
   random block a*block;
run;
```

To test the difference of the B levels at the second level of A with a CONTRAST statement (a slice), you need to assign coefficients 1 and –1 to the levels of B and to the levels of the interaction where A is at the second level. Two examples of equivalent CONTRAST statements by using positional and nonpositional syntax are as follows:

```
contrast 'B at A2' b 1 -1 a*b 0  0  1 -1      ;
contrast 'B at A2' b 1 -1 a*b [1 2 1] [-1 2 2];
```

Because A precedes B in the CLASS statement, the levels of the interaction are formed as $\alpha_1\beta_1, \alpha_1\beta_2, \alpha_2\beta_1, \alpha_2\beta_2, \cdots$. If B precedes A in the CLASS statement, you need to modify the coefficients accordingly:

```
proc glimmix;
   class b a block;
   model y = a b a*b / ddfm=kr;
   random block a*block;
   contrast 'B at A2' b 1 -1 a*b 0  1  0  0 -1     ;
   contrast 'B at A2' b 1 -1 a*b [1  1 2] [-1  2 2];
   contrast 'B at A2' b 1 -1 a*b [1, 1 2] [-1, 2 2];
run;
```

You can optionally separate the **L** value entry from the level indicators with a comma, as in the last CONTRAST statement.

The general syntax for defining coefficients with the nonpositional syntax is as follows:

effect-name [*multiplier* <,> *level-values*] . . . <[*multiplier* <,> *level-values*]>

The first entry in square brackets is the multiplier that is applied to the elements of **L** for the effect after the *level-values* have been resolved and any necessary action forming **L** has been taken.

The *level-values* are organized in a specific form:

- The number of entries should equal the number of terms needed to construct the effect. For effects that do not contain any constructed effects, this number is simply the number of terms in the name of the effect.
- Values of continuous variables needed for the construction of the **L** matrix precede the level indicators of CLASS variables.

- If the effect involves constructed effects, then you need to provide as many continuous and classification variables as are needed for the effect formation. For example, if a grouping effect is defined as

```
class c;
effect v = vars(x1 x2 c);
```

  then a proper nonpositional syntax would be, for example,

```
v [0.5,  0.2 0.3 3]
```

- If an effect contains both regular terms (old-style effects) and constructed effects, then the order of the coefficients is as follows: continuous values for old-style effects, class levels for CLASS variables in old-style effects, continuous values for constructed effects, and finally class levels needed for constructed effects.

  Assume that C has four levels so that effect v contributes six elements to the **L** matrix. When PROC GLIMMIX resolves this syntax, the values 0.2 and 0.3 are assigned to the positions for x1 and x2 and a 1 is associated with the third level of C. The resulting vector is then multiplied by 0.5 to produce

  [0.1 0.15 0 0 0.5 0]

Note that you enter the **levels** of the classification variables in the square brackets, not their formatted values. The ordering of the levels of CLASS variables can be gleaned from the "Class Level Information" table.

To specify values for continuous variables, simply give their value as one of the terms in the effect. The nonpositional syntax in the following ESTIMATE statement is read as "1-time the value 0.4 in the column associated with level 2 of A"

```
proc glimmix;
   class a;
   model y = a a*x / s;
   lsmeans a / e at x=0.4;
   estimate 'A2 at x=0.4' intercept 1 a 0 1 a*x [1,0.4 2] / e;
run;
```

Because the value before the comma serves as a multiplier, the same estimable function could also be constructed with the following statements:

```
estimate 'A2 at x=0.4' intercept 1 a 0 1 a*x [ 4,  0.1 2];
estimate 'A2 at x=0.4' intercept 1 a 0 1 a*x [ 2,  0.2 2];
estimate 'A2 at x=0.4' intercept 1 a 0 1 a*x [-1, -0.4 2];
```

Note that continuous variables needed to construct an effect are always listed before any CLASS variables.

When you work with constructed effects, the nonpositional syntax works in the same way. For example, the following model contains a classification effect and a B-spline. The first two ESTIMATE statements produce predicted values for level one of C when the continuous variable x takes on the values 20 and 10, respectively.

```
proc glimmix;
   class c;
   effect spl = spline(x / knotmethod=equal(5));
   model y = c spl;
   estimate 'C = 1 @ x=20' intercept 1 c 1 spl [1,20],
            'C = 1 @ x=10' intercept 1 c 1 spl [1,10];
   estimate 'Difference'   spl [1,20] [-1,10];
run;
```

The GLIMMIX procedure computes the spline coefficients for the first ESTIMATE statement based on $x = 20$, and similarly in the second statement for $x = 10$. The third ESTIMATE statement computes the difference of the predicted values. Because the spline effect does not interact with the classification variable, this difference does not depend on the level of C. If such an interaction is present, you can estimate the difference in predicted values for a given level of C by using the nonpositional syntax. Because the effect C*spl contains both old-style terms (C) and a constructed effect, you specify the values for the old-style terms before assigning values to constructed effects:

```
proc glimmix;
   class c;
   effect spl = spline(x / knotmethod=equal(5));
   model y = spl*c;
   estimate 'C2 = 1, x=20' intercept 1 c*spl [1,1 20];
   estimate 'C2 = 2, x=20' intercept 1 c*spl [1,2 20];
   estimate 'C diff at x=20' c*spl [1,1 20] [-1,2 20];
run;
```

It is recommended to add the E option to the CONTRAST, ESTIMATE, or LSMESTIMATE statement to verify that the **L** matrix is formed according to your expectations.

In any row of an ESTIMATE or CONTRAST statement you can choose positional and nonpositional syntax separately for each effect. You cannot mix the two forms of syntax for coefficients of a single effect, however. For example, the following statement is not proper because both forms of syntax are used for the interaction effect:

```
estimate 'A1B1 - A1B2' b 1 -1  a*b 0 1  [-1, 1 2];
```

Response-Level Ordering and Referencing

In models for binary and multinomial data, the response-level ordering is important because it reflects the following:

- which probability is modeled with binary data
- how categories are ordered for ordinal data
- which category serves as the reference category in nominal generalized logit models (models for nominal data)

You should view the "Response Profile" table to ensure that the categories are properly arranged and that the desired outcome is modeled. In this table, response levels are arranged by *Ordered Value*. The lowest

response level is assigned Ordered Value 1, the next lowest is assigned Ordered Value 2, and so forth. In binary models, the probability modeled is the probability of the response level with the lowest Ordered Value.

You can change which probability is modeled and the Ordered Value in the "Response Profile" table with the DESCENDING, EVENT=, ORDER=, and REF= response variable options in the MODEL statement. See the section "Response Level Ordering" on page 4564 in Chapter 60, "The LOGISTIC Procedure," for examples about how to use these options to affect the probability being modeled for binary data.

For multinomial models, the response-level ordering affects two important aspects. In cumulative link models the categories are assumed ordered according to their Ordered Value in the "Response Profile" table. If the response variable is a character variable or has a format, you should check this table carefully as to whether the Ordered Values reflect the correct ordinal scale.

In generalized logit models (for multinomial data with unordered categories), one response category is chosen as the reference category in the formulation of the generalized logits. By default, the linear predictor in the reference category is set to 0, and the reference category corresponds to the entry in the "Response Profile" table with the highest Ordered Value. You can affect the assignment of Ordered Values with the DESCENDING and ORDER= options in the MODEL statement. You can choose a different reference category with the REF= option. The choice of the reference category for generalized logit models affects the results. It is sometimes recommended that you choose the category with the highest frequency as the reference (see, for example, Brown and Prescott 1999, p. 160). You can achieve this with the GLIMMIX procedure by combining the ORDER= and REF= options, as in the following statements:

```
proc glimmix;
   class preference;
   model preference(order=freq ref=first) = feature price /
                  dist=multinomial
                  link=glogit;
   random intercept / subject=store group=preference;
run;
```

The ORDER=FREQ option arranges the categories by descending frequency. The REF=FIRST option then selects the response category with the lowest Ordered Value—the most frequent category—as the reference.

Comparing the GLIMMIX and MIXED Procedures

The MIXED procedure is subsumed by the GLIMMIX procedure in the following sense:

- Linear mixed models are a special case in the family of generalized linear mixed models; a linear mixed model is a generalized linear mixed model where the conditional distribution is normal and the link function is the identity function.
- Most models that can be fit with the MIXED procedure can also be fit with the GLIMMIX procedure.

Despite this overlap in functionality, there are also some important differences between the two procedures. Awareness of these differences enables you to select the most appropriate tool in situations where you have a choice between procedures and to identify situations where a choice does not exist. Furthermore, the %GLIMMIX macro, which fits generalized linear mixed models by linearization methods, essentially calls the MIXED procedure repeatedly. If you are aware of the syntax differences between the procedures, you can easily convert your %GLIMMIX macro statements.

Important functional differences between PROC GLIMMIX and PROC MIXED for linear models and linear mixed models include the following:

- The MIXED procedure models R-side effects through the REPEATED statement and G-side effects through the RANDOM statement. The GLIMMIX procedure models all random components of the model through the RANDOM statement. You use the _RESIDUAL_ keyword or the RESIDUAL option in the RANDOM statement to model R-side covariance structure in the GLIMMIX procedure. For example, the PROC MIXED statement

  ```
  repeated / subject=id type=ar(1);
  ```

  is equivalent to the following RANDOM statement in the GLIMMIX procedure:

  ```
  random _residual_ / subject=id type=ar(1);
  ```

  If you need to specify an effect for levelization—for example, because the construction of the **R** matrix is order-dependent or because you need to account for missing values—the RESIDUAL option in the RANDOM statement of the GLIMMIX procedure is used to indicate that you are modeling an R-side covariance nature. For example, the PROC MIXED statements

  ```
  class time id;
  repeated time / subject=id type=ar(1);
  ```

  are equivalent to the following PROC GLIMMIX statements:

  ```
  class time id;
  random time / subject=id type=ar(1) residual;
  ```

- There is generally considerable overlap in the covariance structures available through the TYPE= option in the RANDOM statement in PROC GLIMMIX and through the TYPE= options in the RANDOM and REPEATED statements in PROC MIXED. However, the Kronecker-type structures, the geometrically anisotropic spatial structures, and the GDATA= option in the RANDOM statement of the MIXED procedure are currently not supported in the GLIMMIX procedure. The MIXED procedure, on the other hand, does not support TYPE=RSMOOTH and TYPE=PSPLINE.

- For normal linear mixed models, the (default) METHOD=RSPL in PROC GLIMMIX is identical to the default METHOD=REML in PROC MIXED. Similarly, METHOD=MSPL in PROC GLIMMIX is identical for these models to METHOD=ML in PROC MIXED. The GLIMMIX procedure does not support Type I through Type III (ANOVA) estimation methods for variance component models. Also, the procedure does not have a METHOD=MIVQUE0 option, but you can produce these estimates through the NOITER option in the PARMS statement.

- The MIXED procedure solves the iterative optimization problem by means of a ridge-stabilized Newton-Raphson algorithm. With the GLIMMIX procedure, you can choose from a variety of optimization methods via the NLOPTIONS statement. The default method for most GLMMs is a quasi-Newton algorithm. A ridge-stabilized Newton-Raphson algorithm, akin to the optimization method in the MIXED procedure, is available in the GLIMMIX procedure through the TECHNIQUE=NRRIDG

option in the NLOPTIONS statement. Because of differences in the line-search methods, update methods, and the convergence criteria, you might get slightly different estimates with the two procedures in some instances. The GLIMMIX procedure, for example, monitors several convergence criteria simultaneously.

- You can produce predicted values, residuals, and confidence limits for predicted values with both procedures. The mechanics are slightly different, however. With the MIXED procedure you use the OUTPM= and OUTP= options in the MODEL statement to write statistics to data sets. With the GLIMMIX procedure you use the OUTPUT statement and indicate with *keywords* which "flavor" of a statistic to compute.

- The following GLIMMIX statements are not available in the MIXED procedure: COVTEST, EFFECT, FREQ, LSMESTIMATE, OUTPUT, and programming statements.

- A sampling-based Bayesian analysis as through the PRIOR statement in the MIXED procedure is not available in the GLIMMIX procedure.

- In the GLIMMIX procedure, several RANDOM statement options apply to the RANDOM statement in which they are specified. For example, the following statements in the GLIMMIX procedure request that the solution vector be printed for the A and A*B*C random effects and that the **G** matrix corresponding to the A*B interaction random effect be displayed:

```
random a     / s;
random a*b   / G;
random a*b*c / alpha=0.04;
```

Confidence intervals with a 0.96 coverage probability are produced for the solutions of the A*B*C effect. In the MIXED procedure, the S option, for example, when specified in one RANDOM statement, applies to all RANDOM statements.

- If you select nonmissing values in the *value-list* of the DDF= option in the MODEL statement, PROC GLIMMIX uses these values to override degrees of freedom for this effect that might be determined otherwise. For example, the following statements request that the denominator degrees of freedom for tests and confidence intervals involving the A effect be set to 4:

```
proc glimmix;
   class block a b;
   model y = a b a*b / s ddf=4,.,. ddfm=satterthwaite;
   random block a*block / s;
   lsmeans a b a*b  / diff;
run;
```

In the example, this applies to the "Type III Tests of Fixed Effects," "Least Squares Means," and "Differences of Least Squares Means" tables. In the MIXED procedure, the Satterthwaite approximation overrides the DDF= specification.

- The DDFM=BETWITHIN degrees-of-freedom method in the GLIMMIX procedure requires that the data be processed by subjects; see the section "Processing by Subjects" on page 3218.

- When you add the response variable to the CLASS statement, PROC GLIMMIX defaults to the multinomial distribution. Adding the response variable to the CLASS statement in PROC MIXED has no effect on the fitted model.
- The ODS name of the table for the solution of fixed effects is SolutionF in the MIXED procedure. In PROC GLIMMIX, the name of the table that contains fixed-effects solutions is ParameterEstimates. In generalized linear models, this table also contains scale parameters and overdispersion parameters. The MIXED procedure always produces a "Covariance Parameter Estimates" table. The GLIMMIX procedure produces this table only in mixed models or models with nontrivial R-side covariance structure.
- If you compute predicted values in the GLIMMIX procedure in a model with only R-side random components and missing values for the dependent variable, the predicted values will *not* be kriging predictions as is the case with the MIXED procedure.

Singly or Doubly Iterative Fitting

Depending on the structure of your model, the GLIMMIX procedure determines the appropriate approach for estimating the parameters of the model. The elementary algorithms fall into three categories:

1. **Noniterative algorithms**
   A closed form solution exists for all model parameters. Standard linear models with homoscedastic, uncorrelated errors can be fit with noniterative algorithms.
2. **Singly iterative algorithms**
   A single optimization, consisting of one or more iterations, is performed to obtain solutions for the parameter estimates by numerical techniques. Linear mixed models for normal data can be fit with singly iterative algorithms. Laplace and quadrature estimation for generalized linear mixed models uses a singly iterative algorithm with a separate suboptimization to compute the random-effects solutions as modes of the log-posterior distribution.
3. **Doubly iterative algorithms**
   A model of simpler structure is derived from the target model. The parameters of the simpler model are estimated by noniterative or singly iterative methods. Based on these new estimates, the model of simpler structure is rederived and another estimation step follows. The process continues until changes in the parameter estimates are sufficiently small between two recomputations of the simpler model or until some other criterion is met. The rederivation of the model can often be cast as a change of the response to some pseudo-data along with an update of implicit model weights.

Obviously, noniterative algorithms are preferable to singly iterative ones, which in turn are preferable to doubly iterative algorithms. Two drawbacks of doubly iterative algorithms based on linearization are that likelihood-based measures apply to the pseudo-data, not the original data, and that at the outer level the progress of the algorithm is tied to monitoring the parameter estimates. The advantage of doubly iterative algorithms, however, is to offer—at convergence—the statistical inference tools that apply to the simpler models.

The output and log messages contain information about which algorithm is employed. For a noniterative algorithm, PROC GLIMMIX produces a message that no optimization was performed. Noniterative algorithms are employed automatically for normal data with identity link.

You can determine whether a singly or doubly iterative algorithm was used, based on the "Iteration History" table and the "Convergence Status" table (Figure 44.17).

Figure 44.17 Iteration History and Convergence Status in Singly Iterative Fit

The GLIMMIX Procedure

Iteration History

Iteration	Restarts	Evaluations	Objective Function	Change	Max Gradient
0	0	4	83.039723731	.	13.63536
1	0	3	82.189661988	0.85006174	0.281308
2	0	3	82.189255211	0.00040678	0.000174
3	0	3	82.189255211	0.00000000	1.05E-10

Convergence criterion (GCONV=1E-8) satisfied.

The "Iteration History" table contains the Evaluations column that shows how many function evaluations were performed in a particular iteration. The convergence status message informs you which convergence criterion was met when the estimation process concluded. In a singly iterative fit, the criterion is one that applies to the optimization. In other words, it is one of the criteria that can be controlled with the NLOPTIONS statement: see the ABSCONV=, ABSFCONV=, ABSGCONV=, ABSXCONV=, FCONV=, or GCONV= option.

In a doubly iterative fit, the "Iteration History" table does not contain an Evaluations column. Instead it displays the number of iterations within an optimization (Subiterations column in Figure 44.18).

Figure 44.18 Iteration History and Convergence Status in Doubly Iterative Fit

Iteration History

Iteration	Restarts	Subiterations	Objective Function	Change	Max Gradient
0	0	5	79.688580269	0.11807224	7.851E-7
1	0	3	81.294622554	0.02558021	8.209E-7
2	0	2	81.438701534	0.00166079	4.061E-8
3	0	1	81.444083567	0.00006263	2.311E-8
4	0	1	81.444265216	0.00000421	0.000025
5	0	1	81.444277364	0.00000383	0.000023
6	0	1	81.444266322	0.00000348	0.000021
7	0	1	81.44427636	0.00000316	0.000019
8	0	1	81.444267235	0.00000287	0.000017
9	0	1	81.444275529	0.00000261	0.000016
10	0	1	81.44426799	0.00000237	0.000014
11	0	1	81.444274843	0.00000216	0.000013
12	0	1	81.444268614	0.00000196	0.000012
13	0	1	81.444274277	0.00000178	0.000011
14	0	1	81.444269129	0.00000162	9.772E-6
15	0	0	81.444273808	0.00000000	9.102E-6

Figure 44.18 *continued*

Convergence criterion (PCONV=1.11022E-8) satisfied.

The Iteration column then counts the number of optimizations. The "Convergence Status" table indicates that the estimation process concludes when a criterion is met that monitors the parameter estimates across optimization, namely the PCONV= or ABSPCONV= criterion.

You can control the optimization process with the GLIMMIX procedure through the NLOPTIONS statement. Its options affect the individual optimizations. In a doubly iterative scheme, these apply to all optimizations.

The default optimization techniques are TECHNIQUE=NONE for noniterative estimation, TECHNIQUE=NEWRAP for singly iterative methods in GLMs, TECHNIQUE=NRRIDG for pseudo-likelihood estimation with binary data, and TECHNIQUE=QUANEW for other mixed models.

Default Estimation Techniques

Based on the structure of the model, the GLIMMIX procedure selects the estimation technique for estimating the model parameters. If you fit a generalized linear mixed model, you can change the estimation technique with the METHOD= option in the PROC GLIMMIX statement. The defaults are determined as follows:

- generalized linear model
  - normal distribution: restricted maximum likelihood
  - all other distributions: maximum likelihood
- generalized linear model with overdispersion
  Parameters ($\boldsymbol{\beta}$; ϕ, if present) are estimated by (restricted) maximum likelihood as for generalized linear models. The overdispersion parameter is estimated from the Pearson statistic after all other parameters have been estimated.
- generalized linear mixed models
  The default technique is METHOD=RSPL, corresponding to maximizing the residual log pseudo-likelihood with an expansion about the current solutions of the best linear unbiased predictors of the random effects. In models for normal data with identity link, METHOD=RSPL and METHOD=RMPL are equivalent to restricted maximum likelihood estimation, and METHOD=MSPL and METHOD=MMPL are equivalent to maximum likelihood estimation. This is reflected in the labeling of statistics in the "Fit Statistics" table.

Default Output

The following sections describe the output that PROC GLIMMIX produces by default. The output is organized into various tables, which are discussed in the order of appearance. Note that the contents of a table can change with the estimation method or the model being fit.

Model Information

The "Model Information" table displays basic information about the fitted model, such as the link and variance functions, the distribution of the response, and the data set. If important model quantities—for example, the response, weights, link, or variance function—are user-defined, the "Model Information" table displays the final assignment to the respective variable, as determined from your programming statements. If the table indicates that the variance matrix is blocked by an effect, then PROC GLIMMIX processes the data by subjects. The "Dimensions" table displays the number of subjects. For more information about processing by subjects, see the section "Processing by Subjects" on page 3218. The ODS name of the "Model Information" table is ModelInfo.

Class Level Information

The "Class Level Information" table lists the levels of every variable specified in the CLASS statement. You should check this information to make sure that the data are correct. You can adjust the order of the CLASS variable levels with the ORDER= option in the PROC GLIMMIX statement. The ODS name of the "Class Level Information" table is ClassLevels.

Number of Observations

The "Number of Observations" table displays the number of observations read from the input data set and the number of observations used in the analysis. If you specify a FREQ statement, the table also displays the sum of frequencies read and used. If the *events/trials* syntax is used for the response, the table also displays the number of events and trials used in the analysis. The ODS name of the "Number of Observations" table is NObs.

Response Profile

For binary and multinomial data, the "Response Profile" table displays the Ordered Value from which the GLIMMIX procedure determines the following:

- the probability being modeled for binary data
- the ordering of categories for ordinal data
- the reference category for generalized logit models

For each response category level, the frequency used in the analysis is reported. The section "Response-Level Ordering and Referencing" on page 3235 explains how you can use the DESCENDING, EVENT=, ORDER=, and REF= options to affect the assignment of Ordered Values to the response categories. The ODS name of the "Response Profile" table is ResponseProfile.

Dimensions

The "Dimensions" table displays information from which you can determine the size of relevant matrices in the model. This table is useful in determining CPU time and memory requirements. The ODS name of the "Dimensions" table is "Dimensions."

Optimization Information

The "Optimization Information" table displays important details about the optimization process.

The optimization technique that is displayed in the table is the technique that applies to any single optimization. For singly iterative methods that is *the* optimization method.

The number of parameters that are updated in the optimization equals the number of parameters in this table minus the number of equality constraints. The number of constraints is displayed if you fix covariance parameters with the HOLD= option in the PARMS statement. The GLIMMIX procedure also lists the number of upper and lower boundary constraints. Note that the procedure might impose boundary constraints for certain parameters, such as variance components and correlation parameters. Covariance parameters for which a HOLD= was issued have an upper and lower boundary equal to the parameter value.

If a residual scale parameter is profiled from the optimization, it is also shown in the "Optimization Information" table.

In a GLMM for which the parameters are estimated by one of the linearization methods, you need to initiate the process of computing the pseudo-response. This can be done based on existing estimates of the fixed effects, or by using the data themselves—possibly after some suitable adjustment—as an estimate of the initial mean. The default in PROC GLIMMIX is to use the data themselves to derive initial estimates of the mean function and to construct the pseudo-data. The "Optimization Information" table shows how the pseudo-data are determined initially. Note that this issue is separate from the determination of starting values for the covariance parameters. These are computed as minimum variance quadratic unbiased estimates (with 0 priors, MIVQUE0; Goodnight 1978a) or obtained from the *value-list* in the PARMS statement.

The ODS name of the table is OptInfo.

Iteration History

The "Iteration History" table describes the progress of the estimation process. In singly iterative methods, the table displays the following:

- the iteration count, Iteration
- the number of restarts, Restarts
- the number of function evaluations, Evaluations
- the objective function, Objective
- the change in the objective function, Change
- the absolute value of the largest (projected) gradient, MaxGradient

Note that the change in the objective function is not the convergence criterion monitored by the GLIMMIX procedure. PROC GLIMMIX tracks several convergence criteria simultaneously; see the ABSCONV=, ABSFCONV=, ABSGCONV=, ABSXCONV=, FCONV=, or GCONV= option in the NLOPTIONS statement.

For doubly iterative estimation methods, the "Iteration History" table does not display the progress of the individual optimizations; instead, it reports on the progress of the outer iterations. Every row of the table then corresponds to an update of the linearization, the computation of a new set of pseudo-data, and a new optimization. In the listing, PROC GLIMMIX displays the following:

- the optimization count, Iteration
- the number of restarts, Restarts
- the number of iterations per optimization, Subiterations
- the change in the parameter estimates, Change
- the absolute value of the largest (projected) gradient at the end of the optimization, MaxGradient

By default, the change in the parameter estimates is expressed in terms of the relative PCONV criterion. If you request an absolute criterion with the ABSPCONV option of the PROC GLIMMIX statement, the change reflects the largest absolute difference since the last optimization.

If you specify the ITDETAILS option in the PROC GLIMMIX statement, parameter estimates and their gradients are added to the "Iteration History" table. The ODS name of the "Iteration History" table is IterHistory.

Convergence Status

The "Convergence Status" table contains a status message describing the reason for termination of the optimization. The message is also written to the log. The ODS name of the "Convergence Status" table is ConvergenceStatus, and you can query the nonprinting numeric variable Status to check for a successful optimization. This is useful in batch processing, or when processing BY groups, such as in simulations. Successful optimizations are indicated by the value 0 of the Status variable.

Fit Statistics

The "Fit Statistics" table provides statistics about the estimated model. The first entry of the table corresponds to the negative of twice the (possibly restricted) log likelihood, log pseudo-likelihood, or log quasi-likelihood. If the estimation method permits the true log likelihood or residual log likelihood, the description of the first entry reads accordingly. Otherwise, the fit statistics are preceded by the words *Pseudo-* or *Quasi-*, for Pseudo- and Quasi-Likelihood estimation, respectively.

Note that the (residual) log pseudo-likelihood in a GLMM is the (residual) log likelihood of a linearized model. You should not compare these values across different statistical models, even if the models are nested with respect to fixed and/or G-side random effects. It is possible that between two nested models the larger model has a smaller pseudo-likelihood. For this reason, IC=NONE is the default for GLMMs fit by pseudo-likelihood methods.

See the IC= option of the PROC GLIMMIX statement and Table 44.2 for the definition and computation of the information criteria reported in the "Fit Statistics" table.

For generalized linear models, the GLIMMIX procedure reports Pearson's chi-square statistic

$$X^2 = \sum_i \frac{w_i (y_i - \widehat{\mu}_i)^2}{a(\widehat{\mu}_i)}$$

where $a(\widehat{\mu}_i)$ is the variance function evaluated at the estimated mean.

For GLMMs, the procedure typically reports a generalized chi-square statistic,

$$X^2_g = \widehat{\mathbf{r}}' \mathbf{V}(\widehat{\boldsymbol{\theta}}^*)^{-1} \widehat{\mathbf{r}}$$

so that the ratio of X^2 or X^2_g and the degrees of freedom produces the usual residual dispersion estimate.

If the R-side scale parameter ϕ is not extracted from $\mathbf{V}$, the GLIMMIX procedure computes

$$X^2_g = \widehat{\mathbf{r}}'\mathbf{V}(\widehat{\boldsymbol{\theta}})^{-1}\widehat{\mathbf{r}}$$

as the generalized chi-square statistic. This is the case, for example, if R-side covariance structures are varied by a GROUP= effect or if the scale parameter is not profiled for an R-side TYPE=CS, TYPE=SP, TYPE=AR, TYPE=TOEP, or TYPE=ARMA covariance structure.

For METHOD=LAPLACE, the generalized chi-square statistic is not reported. Instead, the Pearson statistic for the conditional distribution appears in the "Conditional Fit Statistics" table.

If your model contains smooth components (such as TYPE=RSMOOTH), then the "Fit Statistics" table also displays the residual degrees of freedom of the smoother. These degrees of freedom are computed as

$$df_{smooth,res} = f - \text{trace}(\mathbf{S})$$

where $\mathbf{S}$ is the "smoother" matrix—that is, the matrix that produces the predicted values on the linked scale.

The ODS name of the "Fit Statistics" table is FitStatistics.

Covariance Parameter Estimates

In a GLMM, the "Covariance Parameter Estimates" table displays the estimates of the covariance parameters and their asymptotic standard errors. This table is produced only for generalized linear *mixed* models. In generalized linear models with scale parameter, or when an overdispersion parameter is present, the estimates of parameters related to the dispersion are displayed in the "Parameter Estimates" table.

The standard error of the covariance parameters is determined from the diagonal entries of the asymptotic variance matrix of the covariance parameter estimates. You can display this matrix with the ASYCOV option in the PROC GLIMMIX statement.

The ODS name of the "Covariance Parameter Estimates" table is CovParms.

Type III Tests of Fixed Effects

The "Type III Tests of Fixed Effects" table contains hypothesis tests for the significance of each of the fixed effects specified in the MODEL statement. By default, PROC GLIMMIX computes these tests by first constructing a Type III $\mathbf{L}$ matrix for each effect; see Chapter 15, "The Four Types of Estimable Functions." The $\mathbf{L}$ matrix is then used to construct the test statistic

$$F = \frac{\widehat{\boldsymbol{\beta}}'\mathbf{L}'(\mathbf{L}\mathbf{Q}\mathbf{L}')^{-1}\mathbf{L}\widehat{\boldsymbol{\beta}}}{\text{rank}(\mathbf{L}\mathbf{Q}\mathbf{L}')}$$

where the matrix $\mathbf{Q}$ depends on the estimation method and options. For example, in a GLMM, the default is $\mathbf{Q} = (\mathbf{X}'\mathbf{V}(\widehat{\boldsymbol{\theta}})^{-1}\mathbf{X})^{-}$, where $\mathbf{V}(\boldsymbol{\theta})$ is the marginal variance of the pseudo-response. If you specify the DDFM=KENWARDROGER option, $\mathbf{Q}$ is the estimated variance matrix of the fixed effects, adjusted by the method of Kenward and Roger (1997). If the EMPIRICAL= option is in effect, $\mathbf{Q}$ corresponds to the selected sandwich estimator.

You can use the HTYPE= option in the MODEL statement to obtain tables of Type I (sequential) tests and Type II (adjusted) tests in addition to or instead of the table of Type III (partial) tests.

The ODS names of the "Type I Tests of Fixed Effects" through the "Type III Tests of Fixed Effects" tables are Tests1 through Tests3, respectively.

Notes on Output Statistics

Table 44.14 lists the statistics computed with the OUTPUT statement of the GLIMMIX procedure and their default names. This section provides further details about these statistics.

The distinction between prediction and confidence limits in Table 44.14 stems from the involvement of the predictors of the random effects. If the random-effect solutions (BLUPs, EBES) are involved, then the associated standard error used in computing the limits are standard errors of prediction rather than standard errors of estimation. The prediction limits are *not* limits for the prediction of a new observation.

The Pearson residuals in Table 44.14 are "Pearson-type" residuals, because the residuals are standardized by the square root of the marginal or conditional variance of an observation. Traditionally, Pearson residuals in generalized linear models are divided by the square root of the variance function. The GLIMMIX procedure divides by the square root of the variance so that marginal and conditional residuals have similar expressions. In other words, scale and overdispersion parameters are included.

When residuals or predicted values involve only the fixed effects part of the linear predictor (that is, $\widehat{\eta}_m = \mathbf{x}'\widehat{\boldsymbol{\beta}}$), then all model quantities are computed based on this predictor. For example, if the variance by which to standardize a marginal residual involves the variance function, then the variance function is also evaluated at the marginal mean, $g^{-1}(\widehat{\eta}_m)$. Thus the residuals $p - \widehat{\eta}$ and $p_m - \widehat{\eta}_m$ can also be expressed as $(y - \mu)/\partial\mu$ and $(y - \mu_m)/\partial\mu_m$, respectively, where $\partial\mu$ is the derivative with respect to the linear predictor. To construct the residual $p - \widehat{\eta}_m$ in a GLMM, you can add the value of _ZGAMMA_ to the conditional residual $p - \widehat{\eta}$. (The residual $p - \widehat{\eta}_m$ is computed instead of the default marginal residual when you specify the CPSEUDO option in the OUTPUT statement.) If the predictor involves the BLUPs, then all relevant expressions and evaluations involve the conditional mean $g^{-1}(\widehat{\eta})$.

The naming convention to add "PA" to quantities not involving the BLUPs is chosen to suggest the concept of a population average. When the link function is nonlinear, these are not truly population-averaged quantities, because $g^{-1}(\mathbf{x}'\boldsymbol{\beta})$ does not equal $\mathrm{E}[Y]$ in the presence of random effects. For example, if

$$\mu_i = g^{-1}(\mathbf{x}_i'\boldsymbol{\beta} + \mathbf{z}_i'\boldsymbol{\gamma}_i)$$

is the conditional mean for subject i, then

$$g^{-1}(\mathbf{x}_i'\widehat{\boldsymbol{\beta}})$$

does not estimate the average response in the population of subjects but the response of the average subject (the subject for which $\boldsymbol{\gamma}_i = \mathbf{0}$). For models with identity link, the average response and the response of the average subject are identical.

The GLIMMIX procedure obtains standard errors on the scale of the mean by the delta method. If the link is a nonlinear function of the linear predictor, these standard errors are only approximate. For example,

$$\mathrm{Var}[g^{-1}(\widehat{\eta}_m)] \doteq \left(\frac{\partial g^{-1}(t)}{\partial t}\Big|_{\widehat{\eta}_m}\right)^2 \mathrm{Var}[\widehat{\eta}_m]$$

Confidence limits on the scale of the data are usually computed by applying the inverse link function to the confidence limits on the linked scale. The resulting limits on the data scale have the same coverage probability as the limits on the linked scale, but they are possibly asymmetric.

In generalized logit models, confidence limits on the mean scale are based on symmetric limits about the predicted mean in a category. Suppose that the multinomial response in such a model has J categories. The

probability of a response in category i is computed as

$$\widehat{\mu}_i = \frac{\exp\{\widehat{\eta}_i\}}{\sum_{j=1}^{J}\exp\{\widehat{\eta}_i\}}$$

The variance of $\widehat{\mu}_i$ is then approximated as

$$\mathrm{Var}[\widehat{\mu}_i] \doteq \zeta = \boldsymbol{v}_i'\mathrm{Var}\left[\begin{array}{cccc}\widehat{\eta}_1 & \widehat{\eta}_2 & \cdots & \widehat{\eta}_J\end{array}\right]\boldsymbol{v}_i$$

where $\boldsymbol{v}_i$ is a $J \times 1$ vector with kth element

$$\begin{array}{ll}\widehat{\mu}_i(1-\widehat{\mu}_i) & i = k \\ -\widehat{\mu}_i\widehat{\mu}_k & i \neq k\end{array}$$

The confidence limits in the generalized logit model are then obtained as

$$\widehat{\mu}_i \pm t_{\nu,\alpha/2}\sqrt{\zeta}$$

where $t_{\nu,\alpha/2}$ is the $100 \times (1-\alpha/2)$ percentile from a t distribution with ν degrees of freedom. Confidence limits are truncated if they fall outside the $[0, 1]$ interval.

ODS Table Names

Each table created by PROC GLIMMIX has a name associated with it, and you must use this name to reference the table when you use ODS statements. These names are listed in Table 44.23.

Table 44.23 ODS Tables Produced by PROC GLIMMIX

Table Name	Description	Required Statement / Option
AsyCorr	asymptotic correlation matrix of covariance parameters	PROC GLIMMIX ASYCORR
AsyCov	asymptotic covariance matrix of covariance parameters	PROC GLIMMIX ASYCOV
CholG	Cholesky root of the estimated **G** matrix	RANDOM / GC
CholV	Cholesky root of blocks of the estimated **V** matrix	RANDOM / VC
ClassLevels	level information from the CLASS statement	default output
Coef	**L** matrix coefficients	E option in MODEL, CONTRAST, ESTIMATE, LSMESTIMATE, or LSMEANS; ELSM option in LSMESTIMATE
ColumnNames	name association for OUTDESIGN data set	PROC GLIMMIX OUTDESIGN(NAMES)
CondFitStatistics	conditional fit statistics	PROC GLIMMIX METHOD=LAPLACE
Contrasts	results from the CONTRAST statements	CONTRAST

Table 44.23 *continued*

Table Name	Description	Required Statement / Option
ConvergenceStatus	status of optimization at conclusion	default output
CorrB	approximate correlation matrix of fixed-effects parameter estimates	MODEL / CORRB
CovB	approximate covariance matrix of fixed-effects parameter estimates	MODEL / COVB
CovBDetails	details about model-based and/or adjusted covariance matrix of fixed effects	MODEL / COVB(DETAILS)
CovBI	inverse of approximate covariance matrix of fixed-effects parameter estimates	MODEL / COVBI
CovBModelBased	model-based (unadjusted) covariance matrix of fixed effects if DDFM=KR or EMPIRICAL option is used	MODEL / COVB(DETAILS)
CovParms	estimated covariance parameters in GLMMs	default output (in GLMMs)
CovTests	results from COVTEST statements (except for confidence bounds)	COVTEST
Diffs	differences of LS-means	LSMEANS / DIFF (or PDIFF)
Dimensions	dimensions of the model	default output
Estimates	results from ESTIMATE statements	ESTIMATE
FitStatistics	fit statistics	default
G	estimated **G** matrix	RANDOM / G
GCorr	correlation matrix from the estimated **G** matrix	RANDOM / GCORR
Hessian	Hessian matrix (observed or expected)	PROC GLIMMIX HESSIAN
InvCholG	inverse Cholesky root of the estimated **G** matrix	RANDOM / GCI
InvCholV	inverse Cholesky root of the blocks of the estimated **V** matrix	RANDOM / VCI
InvG	inverse of the estimated **G** matrix	RANDOM / GI
InvV	inverse of blocks of the estimated **V** matrix	RANDOM / VI
IterHistory	iteration history	default output
kdTree	*k*-d tree information	RANDOM / TYPE=RSMOOTH KNOTMETHOD= KDTREE(TREEINFO)
KnotInfo	knot coordinates of low-rank spline smoother	RANDOM / TYPE=RSMOOTH KNOTINFO
LSMeans	LS-means	LSMEANS
LSMEstimates	estimates among LS-means	LSMESTIMATE
LSMFtest	*F* test for LSMESTIMATEs	LSMESTIMATE / FTEST
LSMLines	lines display for LS-means	LSMEANS / LINES

Table 44.23 *continued*

Table Name	Description	Required Statement / Option
ModelInfo	model information	default output
NObs	number of observations read and used, number of trials and events	default output
OddsRatios	odds ratios of parameter estimates	MODEL / ODDSRATIO
OptInfo	optimization information	default output
ParameterEstimates	fixed-effects solution; overdispersion and scale parameter in GLMs	MODEL / S
ParmSearch	parameter search values	PARMS
QuadCheck	adaptive recalculation of quadrature approximation at solution	METHOD=QUAD(QCHECK)
ResponseProfile	response categories and category modeled	default output in models with binary or nominal response
Slices	tests of LS-means slices	LSMEANS / SLICE=
SliceDiffs	differences of simple LS-means effects	LSMEANS / SLICEDIFF=
SolutionR	random-effects solution vector	RANDOM / S
StandardizedCoefficients	fixed-effects solutions from centered and/or scaled model	MODEL / STDCOEF
Tests1	Type I tests of fixed effects	MODEL / HTYPE=1
Tests2	Type II tests of fixed effects	MODEL / HTYPE=2
Tests3	Type III tests of fixed effects	default output
V	blocks of the estimated $\mathbf{V}$ matrix	RANDOM / V
VCorr	correlation matrix from the blocks of the estimated $\mathbf{V}$ matrix	RANDOM / VCORR

The SLICE statement also creates tables, which are not listed in Table 44.23. For information about these tables, see the section "SLICE Statement" on page 505 in Chapter 19, "Shared Concepts and Topics."

ODS Graphics

Statistical procedures use ODS Graphics to create graphs as part of their output. ODS Graphics is described in detail in Chapter 21, "Statistical Graphics Using ODS."

Before you create graphs, ODS Graphics must be enabled (for example, by specifying the ODS GRAPHICS ON statement). For more information about enabling and disabling ODS Graphics, see the section "Enabling and Disabling ODS Graphics" on page 606 in Chapter 21, "Statistical Graphics Using ODS."

The overall appearance of graphs is controlled by ODS styles. Styles and other aspects of using ODS Graphics are discussed in the section "A Primer on ODS Statistical Graphics" on page 605 in Chapter 21, "Statistical Graphics Using ODS."

The following subsections provide information about the basic ODS statistical graphics produced by the GLIMMIX procedure. The graphics fall roughly into two categories: diagnostic plots and graphics for least squares means.

ODS Graph Names

The GLIMMIX procedure does not produce graphs by default. You can reference every graph produced through ODS Graphics with a name. The names of the graphs that PROC GLIMMIX generates are listed in Table 44.24, along with the required statements and options.

Table 44.24 Graphs Produced by PROC GLIMMIX

ODS Graph Name	Plot Description	Option
AnomPlot	Plot of LS-mean differences against the average LS-mean	PLOTS=ANOMPLOT LSMEANS / PLOTS=ANOMPLOT
Boxplot	Box plots of residuals and/or observed values for model effects	PLOTS=BOXPLOT
ControlPlot	Plot of LS-mean differences against a control level	PLOTS=CONTROLPLOT LSMEANS / PLOTS=CONTROLPLOT
DiffPlot	Plot of LS-mean pairwise differences	PLOTS=DIFFPLOT LSMEANS / PLOTS=DIFFPLOT
MeanPlot	Plot of least squares means	PLOTS=MEANPLOT LSMEANS / PLOTS=MEANPLOT
ORPlot	Plot of odds ratios	PLOTS=ODDSRATIO
PearsonBoxplot	Box plot of Pearson residuals	PLOTS=PEARSONPANEL(UNPACK)
PearsonByPredicted	Pearson residuals vs. mean	PLOTS=PEARSONPANEL(UNPACK)
PearsonHistogram	Histogram of Pearson residuals	PLOTS=PEARSONPANEL(UNPACK)
PearsonPanel	Panel of Pearson residuals	PLOTS=PEARSONPANEL
PearsonQQplot	*Q-Q* plot of Pearson residuals	PLOTS=PEARSONPANEL(UNPACK)
ResidualBoxplot	Box plot of (raw) residuals	PLOTS=RESIDUALPANEL(UNPACK)
ResidualByPredicted	Residuals vs. mean or linear predictor	PLOTS=RESIDUALPANEL(UNPACK)
ResidualHistogram	Histogram of (raw) residuals	PLOTS=RESIDUALPANEL(UNPACK)
ResidualPanel	Panel of (raw) residuals	PLOTS=RESIDUALPANEL
ResidualQQplot	*Q-Q* plot of (raw) residuals	PLOTS=RESIDUALPANEL(UNPACK)
StudentBoxplot	Box plot of studentized residuals	PLOTS=STUDENTPANEL(UNPACK)
StudentByPredicted	Studentized residuals vs. mean or linear predictor	PLOTS=STUDENTPANEL(UNPACK)
StudentHistogram	Histogram of studentized residuals	PLOTS=STUDENTPANEL(UNPACK)
StudentPanel	Panel of studentized residuals	PLOTS=STUDENTPANEL
StudentQQplot	*Q-Q* plot of studentized residuals	PLOTS=STUDENTPANEL(UNPACK)

When ODS Graphics is enabled, the SLICE statement can produce plots that are associated with its analysis. For information about these plots, see the section "SLICE Statement" on page 505 in Chapter 19, "Shared Concepts and Topics."

Diagnostic Plots

Residual Panels

There are three types of residual panels in the GLIMMIX procedure. Their makeup of four component plots is the same; the difference lies in the type of residual from which the panel is computed. Raw residuals are displayed with the PLOTS=RESIDUALPANEL option. Studentized residuals are displayed with the PLOTS=STUDENTPANEL option, and Pearson residuals with the PLOTS=PEARSONPANEL option. By default, conditional residuals are used in the construction of the panels if the model contains G-side random effects. For example, consider the following statements:

```
proc glimmix plots=residualpanel;
   class A;
   model y = x1 x2 / dist=Poisson;
   random int / sub=A;
run;
```

The parameters are estimated by a pseudo-likelihood method, and at the final stage pseudo-data are related to a linear mixed model with random intercepts. The residual panel is constructed from

$$r = p - \mathbf{x}'\widehat{\boldsymbol{\beta}} + \mathbf{z}'\widehat{\boldsymbol{\gamma}}$$

where p is the pseudo-data.

The following hypothetical data set contains yields of an industrial process. Material was available from five randomly selected vendors to produce a chemical reaction whose yield depends on two factors (pressure and temperature at 3 and 2 levels, respectively).

```
data Yields;
   input Vendor Pressure Temp Yield @@;
   datalines;
 1  1  1  10.20    1  1  2   9.48    1  2  1   9.74
 1  2  2   8.92    1  3  1  11.79    1  3  2   8.85
 2  1  1  10.43    2  1  2  10.59    2  2  1  10.29
 2  2  2  10.15    2  3  1  11.12    2  3  2   9.30
 3  1  1   6.46    3  1  2   7.34    3  2  1   9.44
 3  2  2   8.11    3  3  1   9.38    3  3  2   8.37
 4  1  1   7.36    4  1  2   9.92    4  2  1  10.99
 4  2  2  10.34    4  3  1  10.24    4  3  2   9.96
 5  1  1  11.72    5  1  2  10.60    5  2  1  11.28
 5  2  2   9.03    5  3  1  14.09    5  3  2   8.92
;
```

Consider a linear mixed model with a two-way factorial fixed-effects structure for pressure and temperature effects and independent, homoscedastic random effects for the vendors. The following statements fit this model and request panels of marginal and conditional residuals:

```
ods graphics on;

proc glimmix data=Yields
      plots=residualpanel(conditional marginal);
   class Vendor Pressure Temp;
   model Yield = Pressure Temp Pressure*Temp;
   random vendor;
run;

ods graphics off;
```

The suboptions of the RESIDUALPANEL request produce two panels. The panel of conditional residuals is constructed from $y - \mathbf{x}'\widehat{\boldsymbol{\beta}} - \mathbf{z}'\widehat{\boldsymbol{\gamma}}$ (Figure 44.19). The panel of marginal residuals is constructed from $y - \mathbf{x}'\widehat{\boldsymbol{\beta}}$ (Figure 44.20). Note that these residuals are deviations from the observed data, because the model is a normal linear mixed model, and hence it does not involve pseudo-data. Whenever the random-effects solutions $\widehat{\boldsymbol{\gamma}}$ are involved in constructing residuals, the title of the residual graphics identifies them as conditional residuals (Figure 44.19).

Figure 44.19 Conditional Residuals

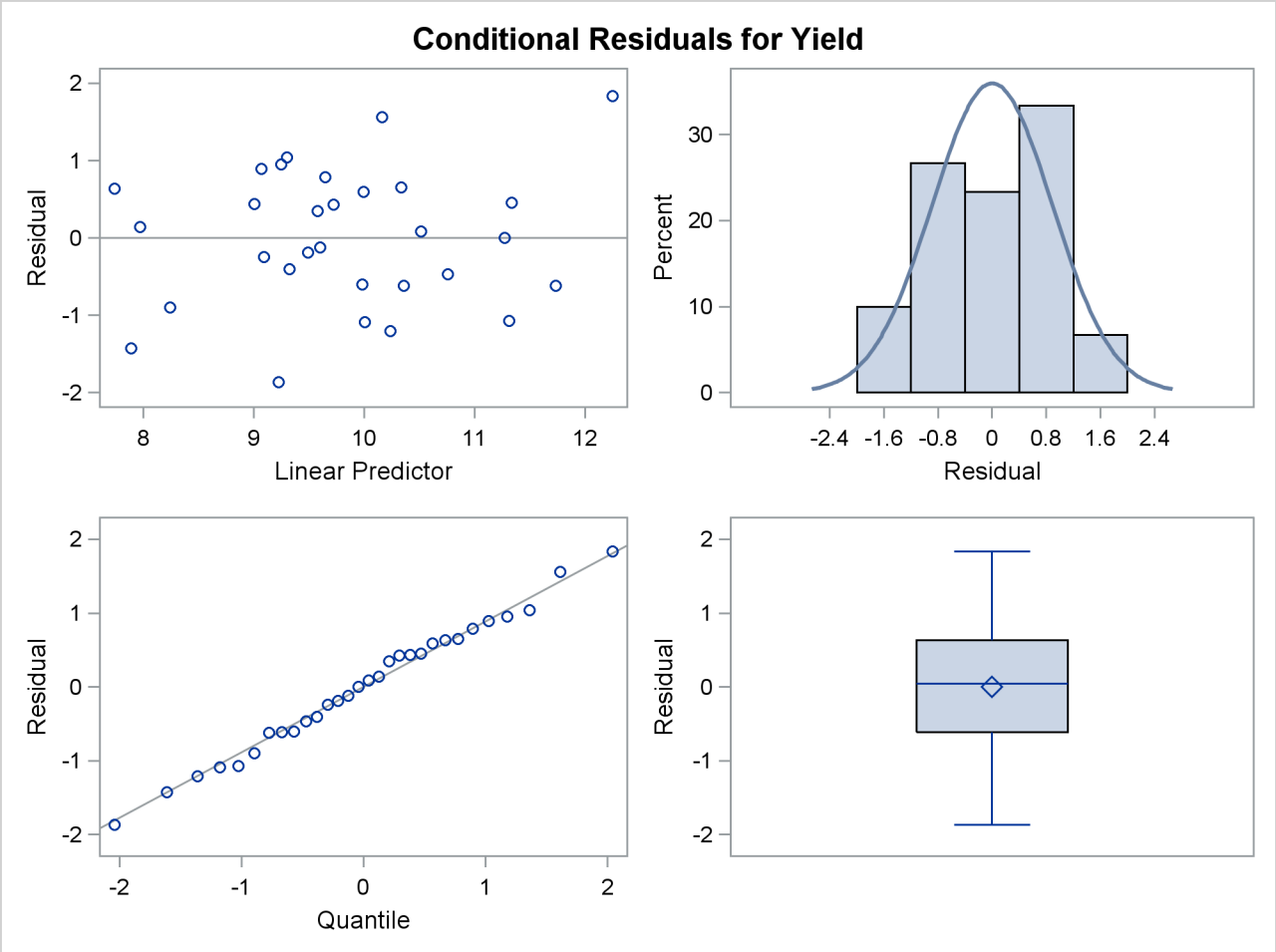
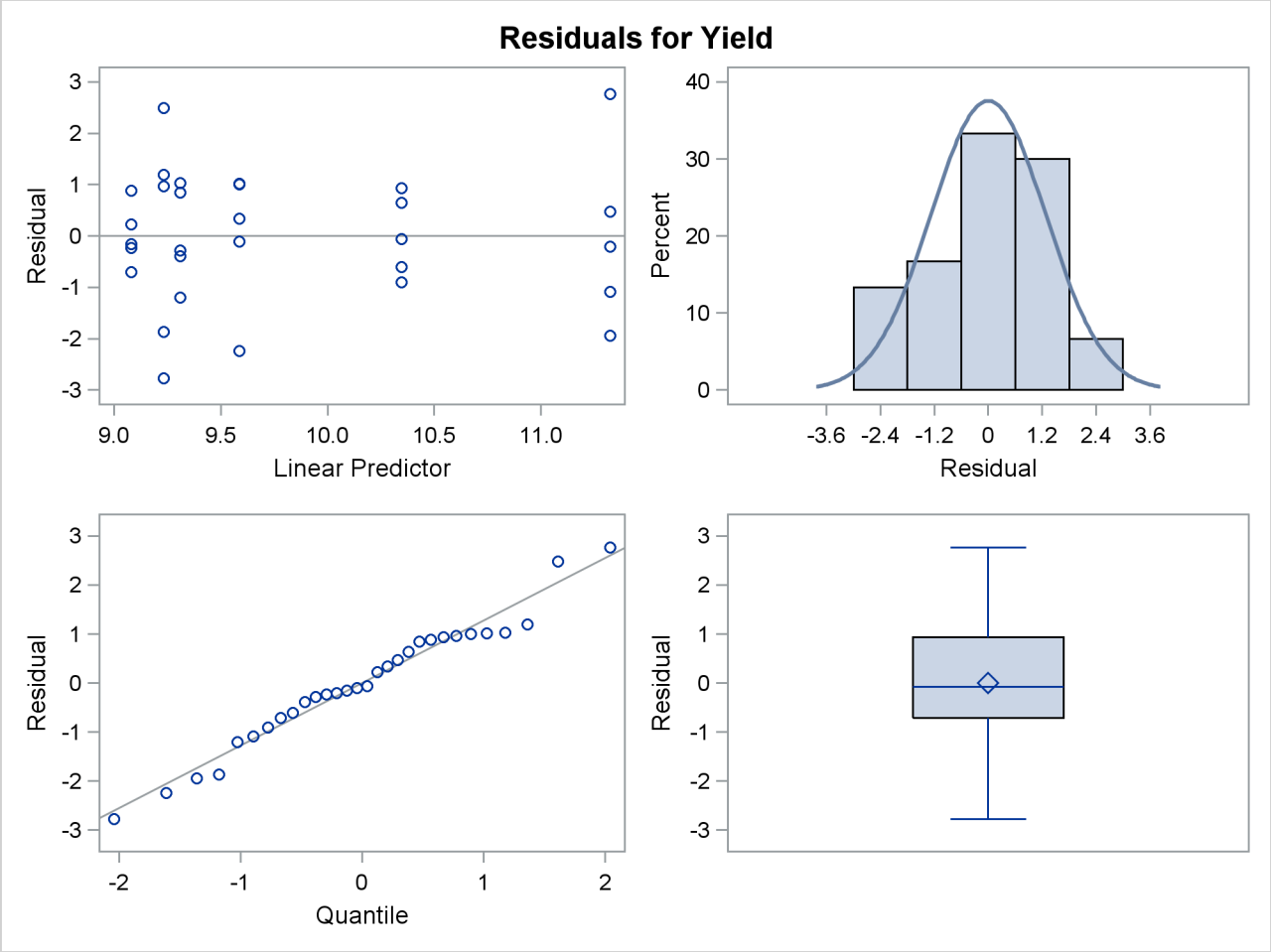


Figure 44.20 Marginal Residuals



The predictor takes on only six values for the marginal residuals, corresponding to the combinations of three temperature and two pressure levels. The assumption of a zero mean for the vendor random effect seems justified; the marginal residuals in the upper-left plot of Figure 44.20 do not exhibit any trend. The conditional residuals in Figure 44.19 are smaller and somewhat closer to normality compared to the marginal residuals.

Box Plots

You can produce box plots of observed data, pseudo-data, and various residuals for effects in your model that consist of classification variables. Because you might not want to produce box plots for all such effects, you can request subsets with the suboptions of the BOXPLOT option in the PLOTS option. The BOXPLOT request in the following PROC GLIMMIX statement produces box plots for the random effects—in this case, the vendor effect. By default, PROC GLIMMIX constructs box plots from conditional residuals. The MARGINAL, CONDITIONAL, and OBSERVED suboptions instruct the procedure to construct three box plots for each random effect: box plots of the observed data (Figure 44.21), the marginal residuals (Figure 44.22), and the conditional residuals (Figure 44.23).

```
ods graphics on;

proc glimmix data=Yields
       plots=boxplot(random marginal conditional observed);
   class Vendor Pressure Temp;
   model Yield = Pressure Temp Pressure*Temp;
   random vendor;
run;

ods graphics off;
```

The observed vendor means in Figure 44.21 are different; in particular, vendors 3 and 5 appear to differ from the other vendors and from each other. There is also heterogeneity of variance in the five groups. The marginal residuals in Figure 44.22 continue to show the differences in means by vendor, because vendor enters the model as a random effect. The marginal means are adjusted for vendor effects only in the sense that the vendor variance component affects the marginal variance that is involved in the generalized least squares solution for the pressure and temperature effects.

Figure 44.21 Box Plots of Observed Values

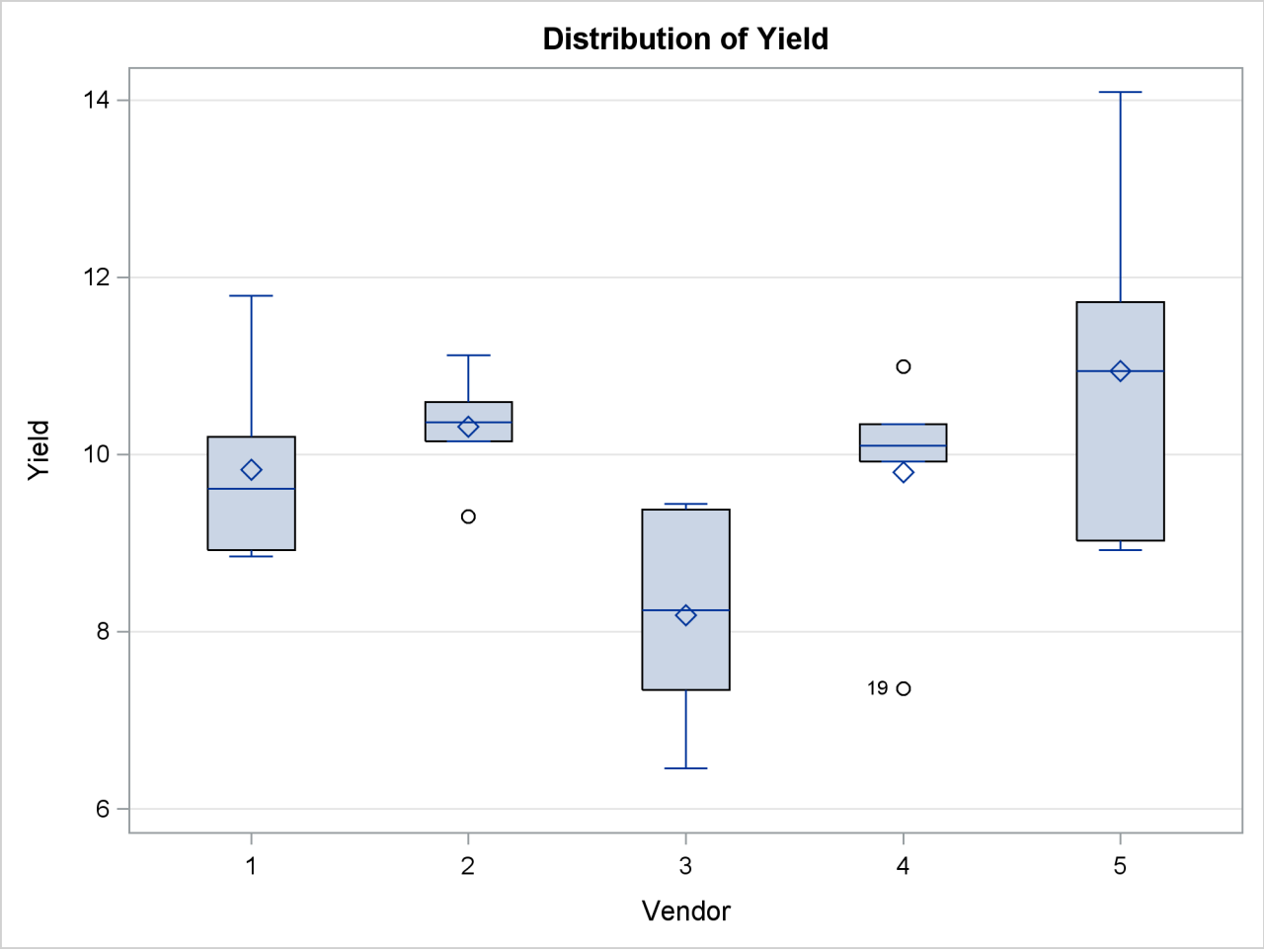
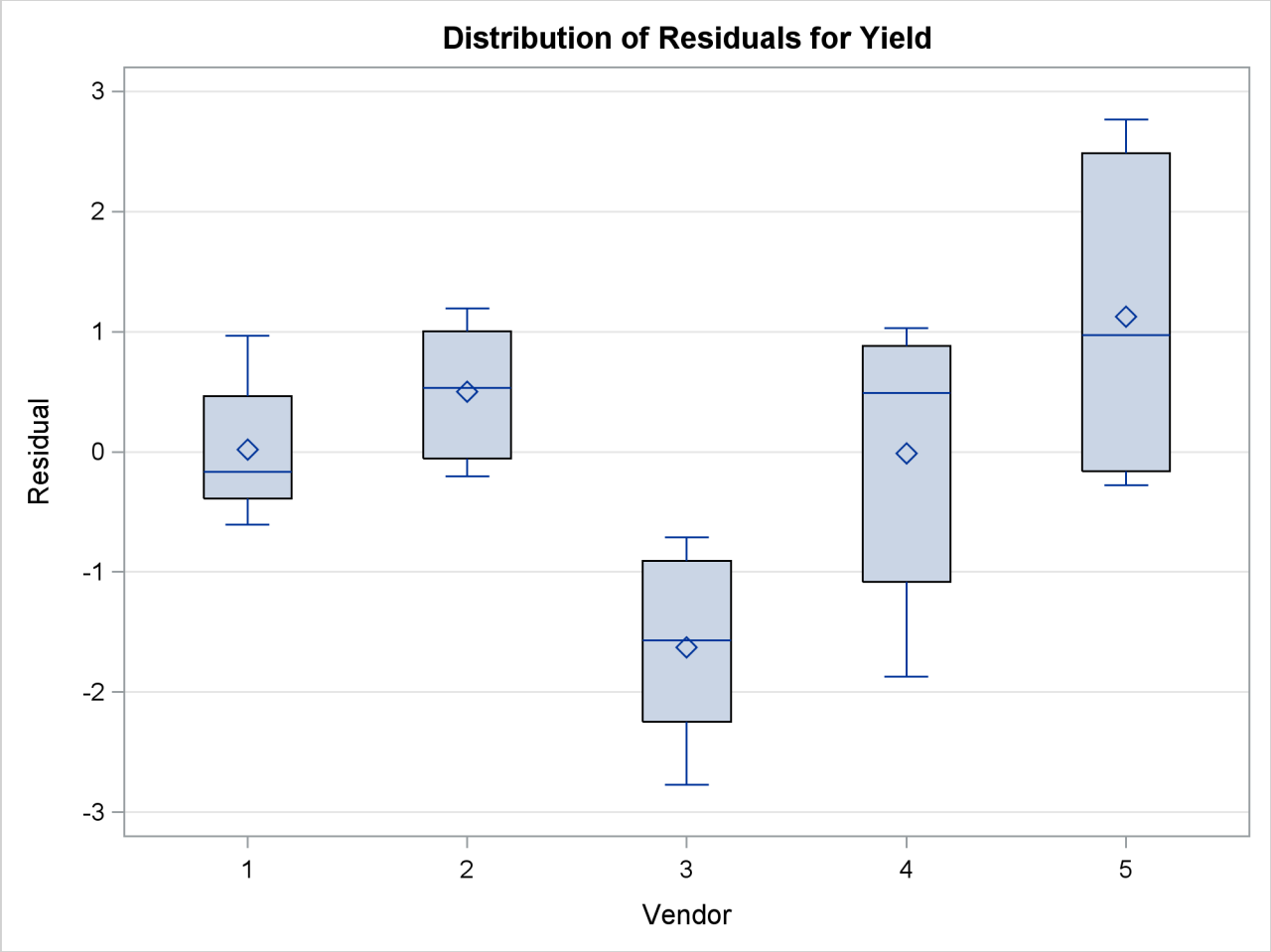
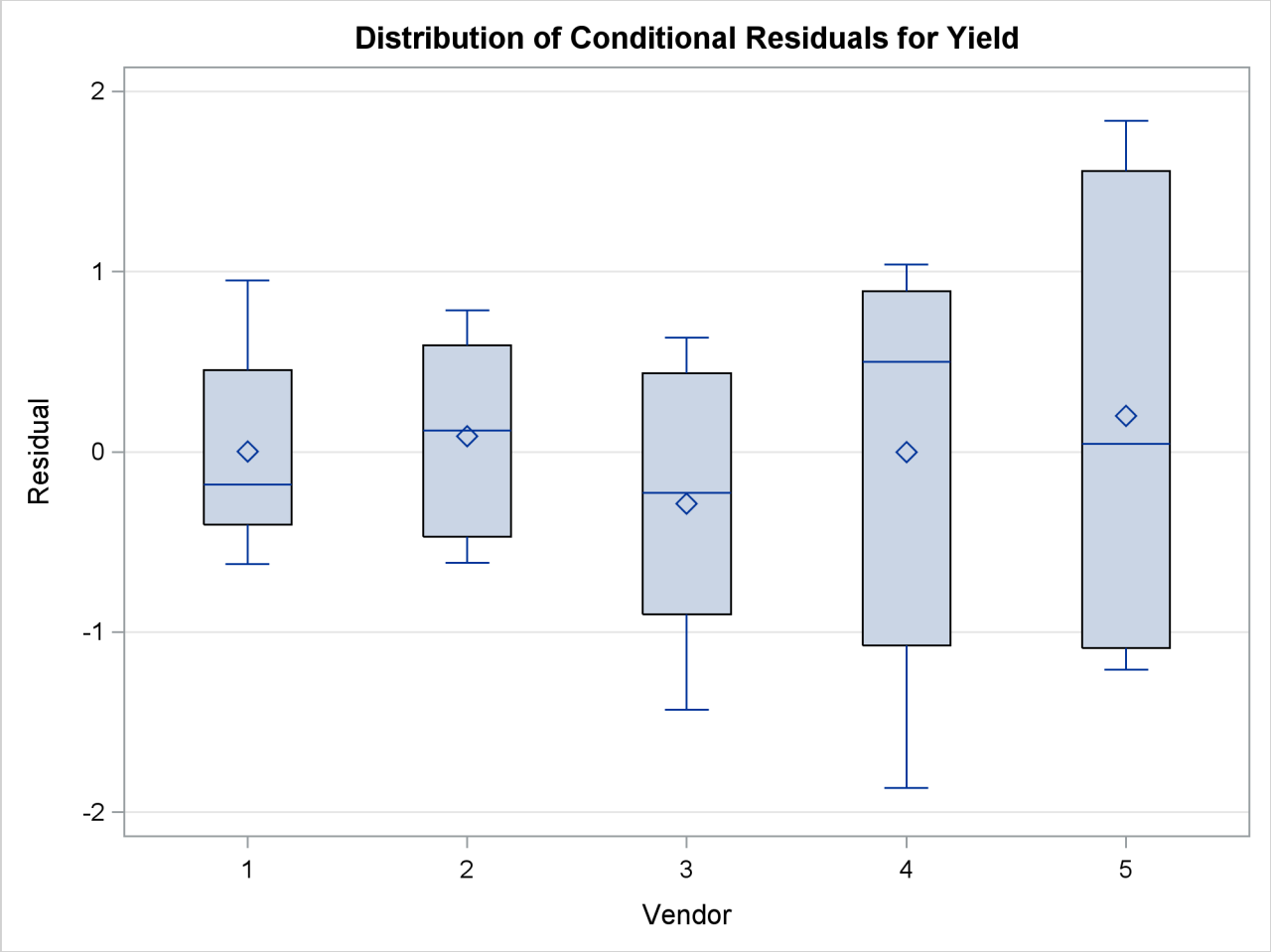


Figure 44.22 Box Plots of Marginal Residuals



The conditional residuals account for the vendor effects through the empirical BLUPs. The means and medians have stabilized near zero, but some heterogeneity in these residuals remains (Figure 44.23).

Figure 44.23 Box Plots of Conditional Residuals



Graphics for LS-Mean Comparisons

The following subsections provide information about the ODS statistical graphics for least squares means produced by the GLIMMIX procedure. Mean plots display marginal or interaction means. The diffogram, control plot, and ANOM plot display least squares mean comparisons.

Mean Plots

The following SAS statements request a plot of the Pressure×Temp means in which the pressure trends are plotted for each temperature.

```
ods graphics on;
ods select CovParms Tests3 MeanPlot;
proc glimmix data=Yields;
   class Vendor Pressure Temp;
   model Yield = Pressure Temp Pressure*Temp;
   random Vendor;
   lsmeans Pressure*Temp / plot=mean(sliceby=Temp join);
run;
ods graphics off;
```

There is a significant effect of temperature and an interaction between pressure and temperature (Figure 44.24). Notice that the pressure main effect might be masked by the interaction. Because of the interaction, temperature comparisons depend on the pressure and vice versa. The mean plot option requests a display of the Pressure × Temp least squares means with separate trends for each temperature (Figure 44.25).

Figure 44.24 Tests for Fixed Effects

The GLIMMIX Procedure

Covariance Parameter Estimates

Cov Parm	Estimate	Standard Error
Vendor	0.8602	0.7406
Residual	1.1039	0.3491

Type III Tests of Fixed Effects

Effect	Num DF	Den DF	F Value	Pr > F
Pressure	2	20	1.42	0.2646
Temp	1	20	6.48	0.0193
Pressure*Temp	2	20	3.82	0.0393

The interaction between the two effects is evident in the lack of parallelism in Figure 44.25. The masking of the pressure main effect can be explained by slopes of different sign for the two trends. Based on these results, inferences about the pressure effects are conducted for a specific temperature. For example, Figure 44.26 is produced by adding the following statement:

```
lsmeans pressure*temp / slicediff=temp slice=temp;
```

Figure 44.25 Interaction Plot for Pressure x Temperature

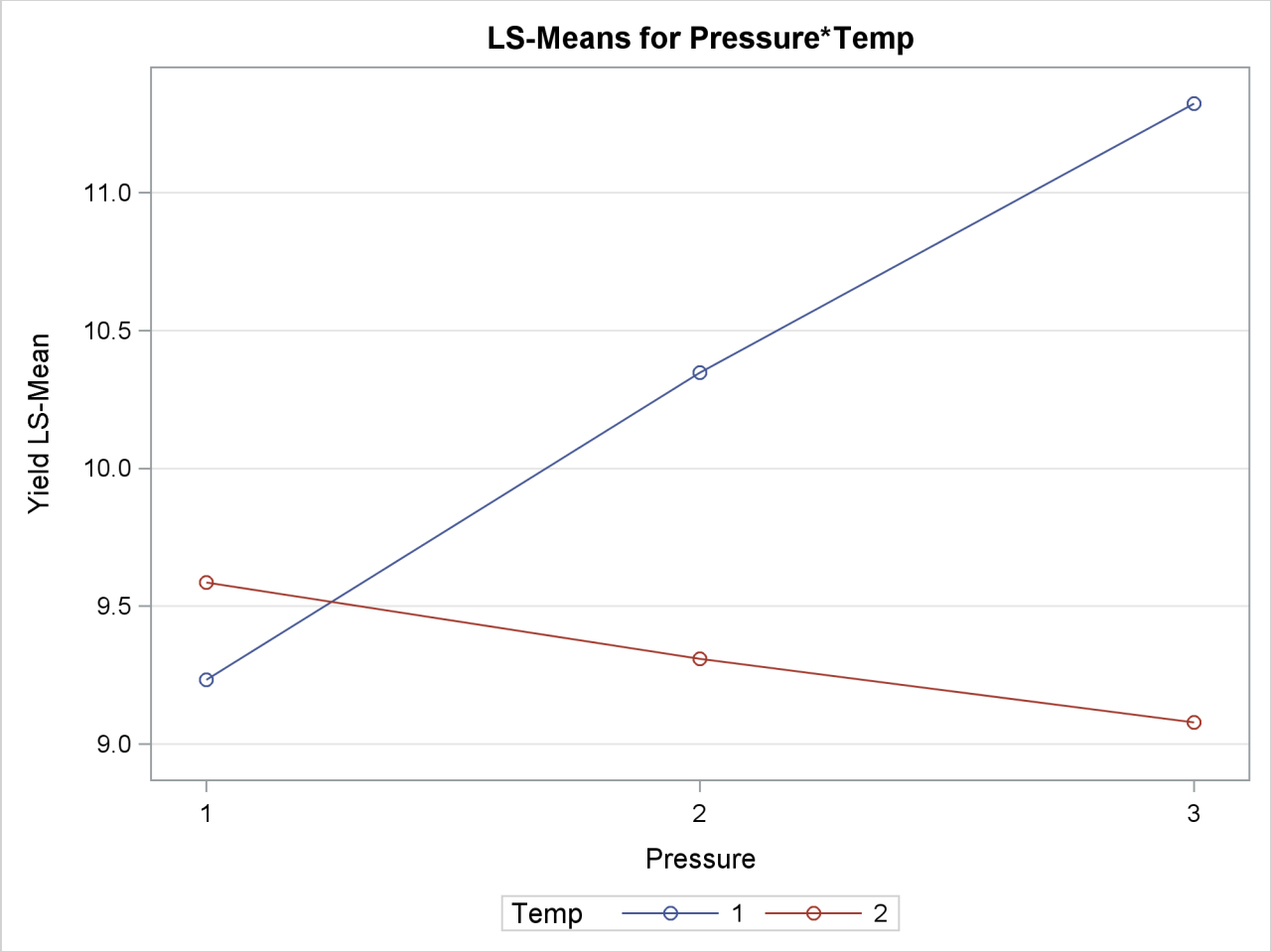


Figure 44.26 Pressure Comparisons at a Given Temperature

The GLIMMIX Procedure

Tests of Effect Slices for Pressure*Temp Sliced By Temp

Temp	Num DF	Den DF	F Value	Pr > F
1	2	20	4.95	0.0179
2	2	20	0.29	0.7508

Figure 44.26 *continued*

Simple Effect Comparisons of Pressure*Temp Least Squares Means By Temp

Simple Effect Level	Pressure	_Pressure	Estimate	Standard Error	DF	t Value	Pr > |t|
Temp 1	1	2	-1.1140	0.6645	20	-1.68	0.1092
Temp 1	1	3	-2.0900	0.6645	20	-3.15	0.0051
Temp 1	2	3	-0.9760	0.6645	20	-1.47	0.1575
Temp 2	1	2	0.2760	0.6645	20	0.42	0.6823
Temp 2	1	3	0.5060	0.6645	20	0.76	0.4553
Temp 2	2	3	0.2300	0.6645	20	0.35	0.7329

The slope differences are evident by the change in sign for comparisons within temperature 1 and within temperature 2. There is a significant effect of pressure at temperature 1 ($p = 0.0179$), but not at temperature 2 ($p = 0.7508$).

Pairwise Difference Plot (Diffogram)

Graphical displays of LS-means-related analyses consist of plots of all pairwise differences (DiffPlot), plots of differences against a control level (ControlPlot), and plots of differences against an overall average (AnomPlot). The following data set is from an experiment to investigate how snapdragons grow in various soils (Stenstrom 1940). To eliminate the effect of local fertility variations, the experiment is run in blocks, with each soil type sampled in each block. See the "Examples" section of Chapter 45, "The GLM Procedure," for an in-depth analysis of these data.

```
data plants;
   input Type $ @;
   do Block = 1 to 3;
      input StemLength @;
      output;
   end;
   datalines;
Clarion   32.7 32.3 31.5
Clinton   32.1 29.7 29.1
Knox      35.7 35.9 33.1
ONeill    36.0 34.2 31.2
Compost   31.8 28.0 29.2
Wabash    38.2 37.8 31.9
Webster   32.5 31.1 29.7
;
```

The following statements perform the analysis of the experiment with the GLIMMIX procedure:

```
ods graphics on;
ods select LSMeans DiffPlot;

proc glimmix data=plants order=data plots=Diffogram;
   class Block Type;
   model StemLength = Block Type;
   lsmeans Type;
run;

ods graphics off;
```

The PLOTS= option in the PROC GLIMMIX statement requests that plots of pairwise least squares means differences are produced for effects that are listed in corresponding LSMEANS statements. This is the Type effect.

The Type LS-means are shown in Figure 44.27. Note that the order in which the levels appear corresponds to the order in which they were read from the data set. This was accomplished with the ORDER=DATA option in the PROC GLIMMIX statement.

Figure 44.27 Least Squares Means for Type Effect

The GLIMMIX Procedure

Type Least Squares Means

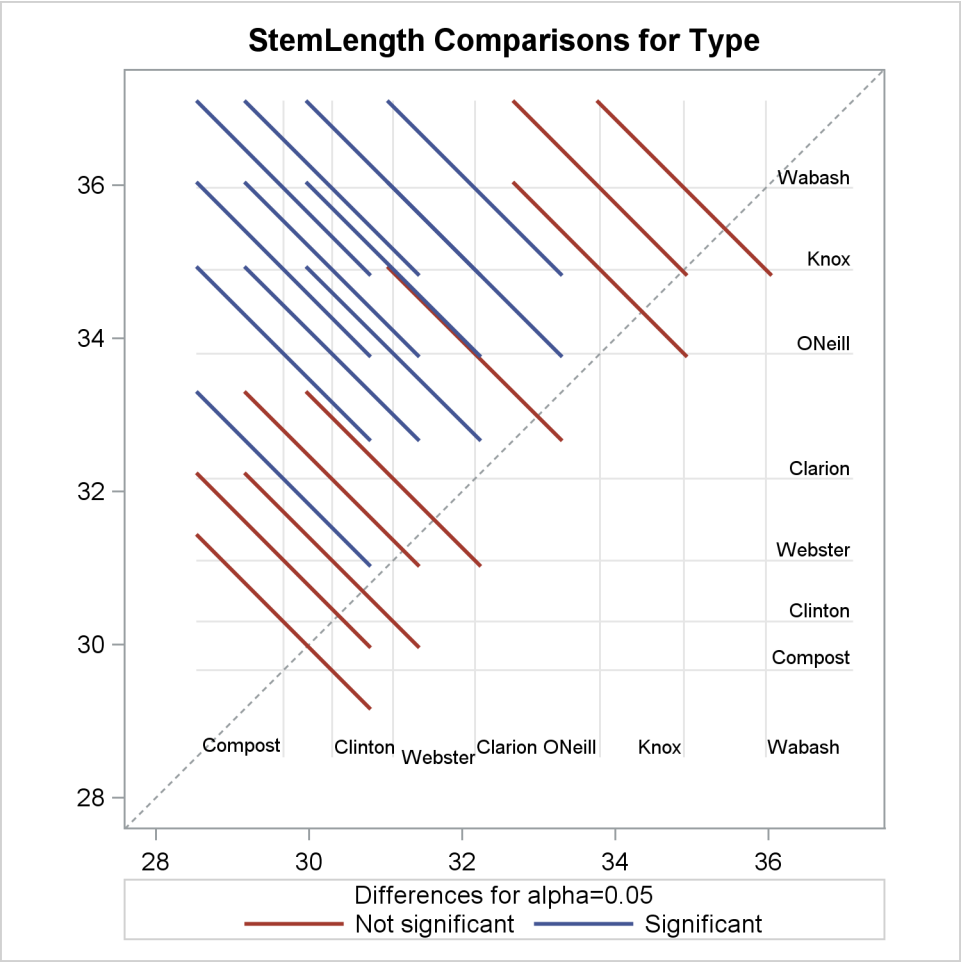
Type	Estimate	Standard Error	DF	t Value	Pr > |t|
Clarion	32.1667	0.7405	12	43.44	<.0001
Clinton	30.3000	0.7405	12	40.92	<.0001
Knox	34.9000	0.7405	12	47.13	<.0001
ONeill	33.8000	0.7405	12	45.64	<.0001
Compost	29.6667	0.7405	12	40.06	<.0001
Wabash	35.9667	0.7405	12	48.57	<.0001
Webster	31.1000	0.7405	12	42.00	<.0001

Because there are seven levels of Type in this analysis, there are $7(6-1)/2 = 21$ pairwise comparisons among the least squares means. The comparisons are performed in the following fashion: the first level of Type is compared against levels 2 through 7; the second level of Type is compared against levels 3 through 7; and so forth.

The default difference plot for these data is shown in Figure 44.28. The display is also known as a "mean-mean scatter plot" (Hsu 1996; Hsu and Peruggia 1994). It contains 21 lines rotated by 45 degrees counterclockwise, and a reference line (dashed 45-degree line). The (x, y) coordinate for the center of each line corresponds to the two least squares means being compared. Suppose that $\widehat{\eta}_{.i}$ and $\widehat{\eta}_{.j}$ denote the ith and jth least squares mean, respectively, for the effect in question, where $i < j$ according to the ordering of the effect levels. If the ABS option is in effect, which is the default, the line segment is centered at $(\min\{\widehat{\eta}_{i.}, \widehat{\eta}_{j.}\}, \max\{\widehat{\eta}_{i.}, \widehat{\eta}_{j.}\})$. Take, for example, the comparison of "Clarion" and "Compost" types. The respective estimates of their LS-means are $\widehat{\eta}_{.1} = 32.1667$ and $\widehat{\eta}_{.5} = 29.6667$. The center of the line segment for $H_0\colon \eta_{.1} = \eta_{.5}$ is placed at $(29.6667, 32.1667)$.

The length of the line segment for the comparison between means i and j corresponds to the width of the confidence interval for the difference $\eta_{.i} - \eta_{.j}$. This length is adjusted for the rotation in the plot. As a consequence, comparisons whose confidence interval covers zero cross the 45-degree reference line. These are the nonsignificant comparisons. Lines associated with significant comparisons do not touch or cross the reference line. Because these data are balanced, the estimated standard errors of all pairwise comparisons are identical, and the widths of the line segments are the same.

Figure 44.28 LS-Means Plot of Pairwise Differences



The background grid of the difference plot is drawn at the values of the least squares means for the seven type levels. These grid lines are used to find a particular comparison by intersection. Also, the labels of the grid lines indicate the ordering of the least squares means.

In the next set of statements, the NOABS and CENTER suboptions of the PLOTS=DIFFOGRAM option in the LSMEANS statement modify the appearance of the diffogram:

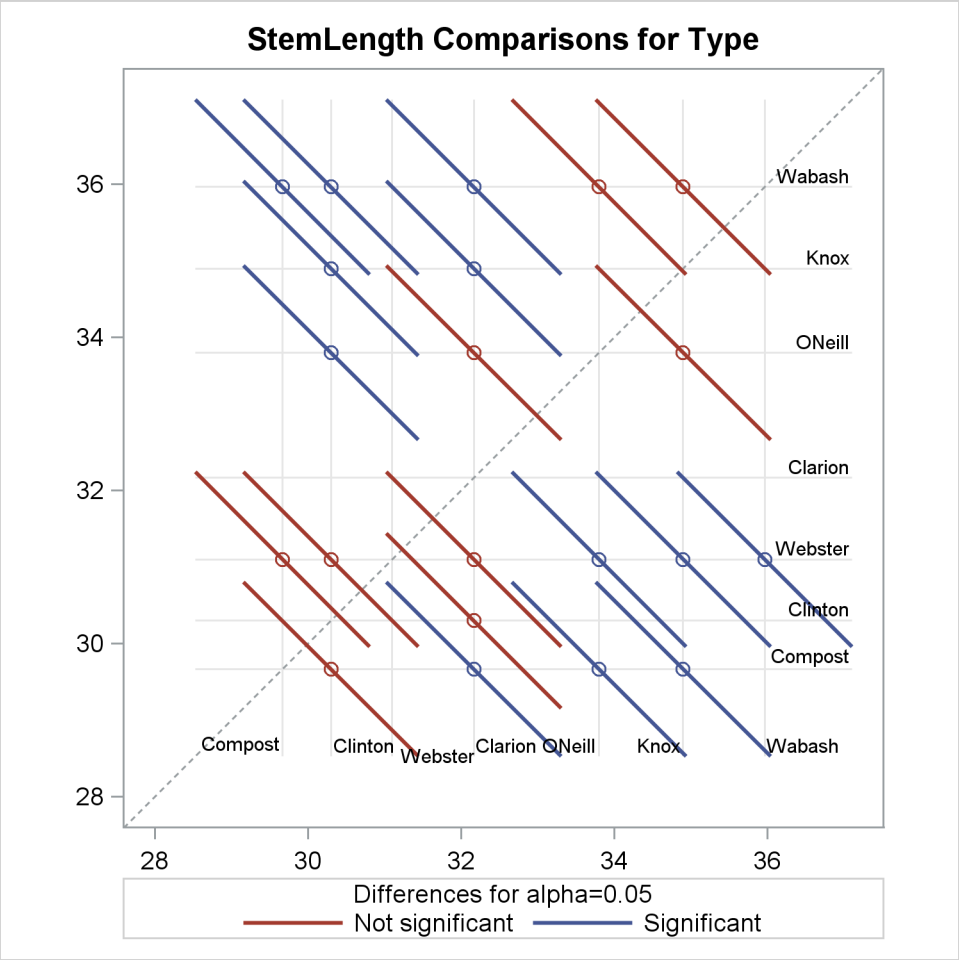
```
ods graphics on;
proc glimmix data=plants order=data;
   class Block Type;
   model StemLength = Block Type;
   lsmeans Type / plots=diffogram(noabs center);
run;
ods graphics off;
```

The NOABS suboption of the difference plot changes the way in which the GLIMMIX procedure places the line segments (Figure 44.29). If the NOABS suboption is in effect, the line segment is centered at the point $(\widehat{\eta}_{.i}, \widehat{\eta}_{.j})$, $i < j$. For example, the center of the line segment for a comparison of "Clarion" and "Compost" types is centered at $(\widehat{\eta}_{.1}, \widehat{\eta}_{.5}) = (32.1667, 29.6667)$. Whether a line segment appears above or below the reference line depends on the magnitude of the least squares means and the order of their appearance in the

"Least Squares Means" table. The CENTER suboption places a marker at the intersection of the least squares means.

Because the ABS option places lines on the same side of the 45-degree reference, it can help to visually discover groups of significant and nonsignificant differences. On the other hand, when the number of levels in the effect is large, the display can get crowded. The NOABS option can then provide a more accessible resolution.

Figure 44.29 Diffogram with NOABS and CENTER Options



Least Squares Mean Control Plot

The following SAS statements create the same data set as before, except that one observation for Type="Knox" has been removed for illustrative purposes:

```
data plants;
   input Type $ @;
   do Block = 1 to 3;
      input StemLength @;
      output;
   end;
   datalines;
```

```
Clarion   32.7 32.3 31.5
Clinton   32.1 29.7 29.1
Knox      35.7 35.9   .
ONeill    36.0 34.2 31.2
Compost   31.8 28.0 29.2
Wabash    38.2 37.8 31.9
Webster   32.5 31.1 29.7
;
```

The following statements request control plots for effects in LSMEANS statements with compatible option:

```
ods graphics on;
ods select Diffs ControlPlot;

proc glimmix data=plants order=data plots=ControlPlot;
   class Block Type;
   model StemLength = Block Type;
   lsmeans Type / diff=control('Clarion') adjust=dunnett;
run;

ods graphics off;
```

The LSMEANS statement for the Type effect is compatible; it requests comparisons of Type levels against "Clarion," adjusted for multiplicity with Dunnett's method. Because "Clarion" is the first level of the effect, the LSMEANS statement is equivalent to

```
lsmeans type / diff=control adjust=dunnett;
```

The "Differences of Type Least Squares Means" table in Figure 44.30 shows the six comparisons between Type levels and the control level.

Figure 44.30 Least Squares Means Differences

The GLIMMIX Procedure

Differences of Type Least Squares Means
Adjustment for Multiple Comparisons: Dunnett

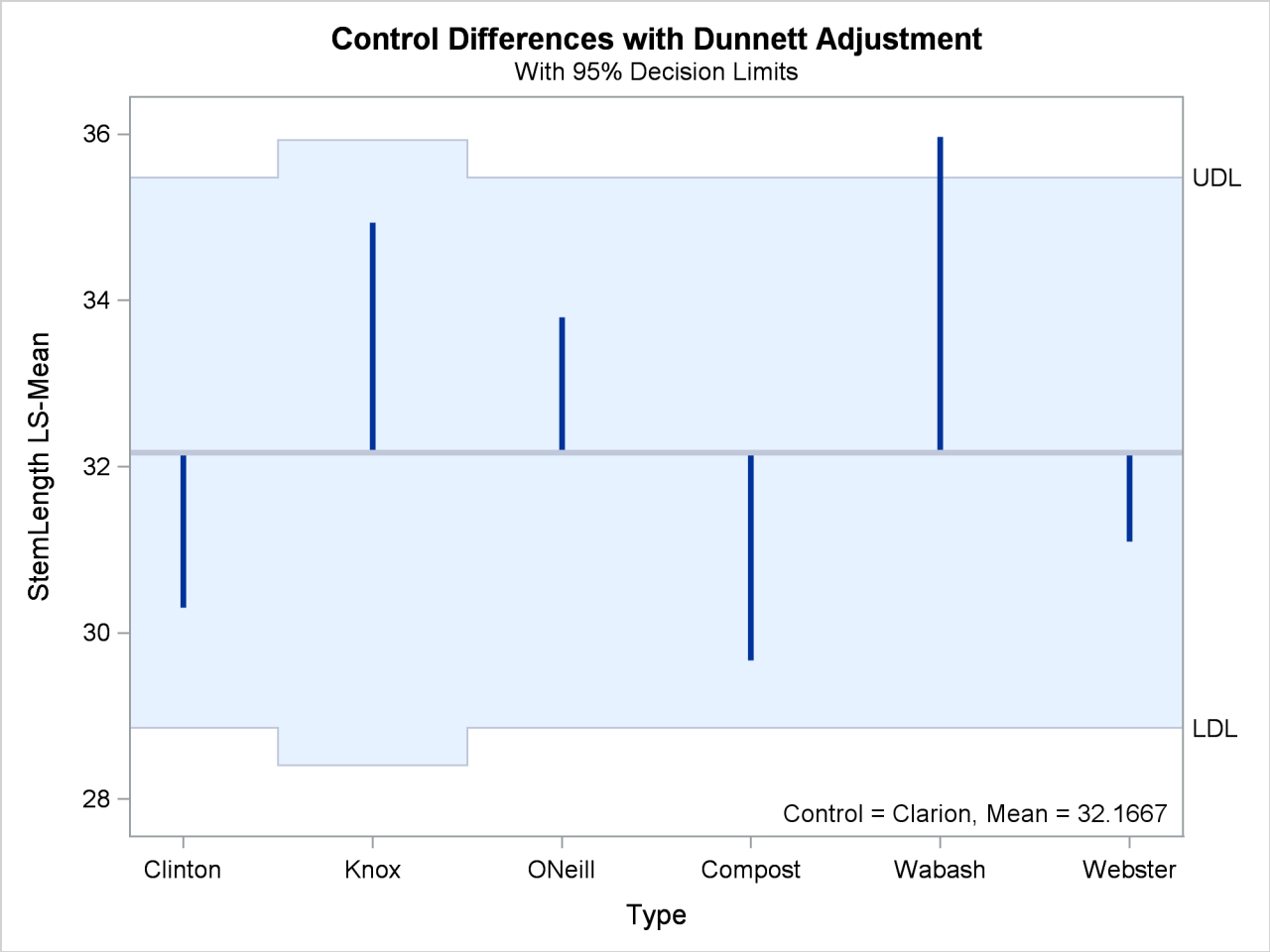
Type	_Type	Estimate	Standard Error	DF	t Value	Pr > |t|	Adj P
Clinton	Clarion	-1.8667	1.0937	11	-1.71	0.1159	0.3936
Knox	Clarion	2.7667	1.2430	11	2.23	0.0479	0.1854
ONeill	Clarion	1.6333	1.0937	11	1.49	0.1635	0.5144
Compost	Clarion	-2.5000	1.0937	11	-2.29	0.0431	0.1688
Wabash	Clarion	3.8000	1.0937	11	3.47	0.0052	0.0236
Webster	Clarion	-1.0667	1.0937	11	-0.98	0.3504	0.8359

The two rightmost columns of the table give the unadjusted and multiplicity-adjusted p-values. At the 5% significance level, both "Knox" and "Wabash" differ significantly from "Clarion" according to the unadjusted tests. After adjusting for multiplicity, only "Wabash" has a least squares mean significantly different from the control mean. Note that the standard error for the comparison involving "Knox" is larger than that for other comparisons because of the reduced sample size for that soil type.

In the plot of control differences a horizontal line is drawn at the value of the "Clarion" least squares mean. Vertical lines emanating from this reference line terminate in the least squares means for the other levels (Figure 44.31).

The dashed upper and lower horizontal reference lines are the upper and lower decision limits for tests against the control level. If a vertical line crosses the upper or lower decision limit, the corresponding least squares mean is significantly different from the LS-mean in the control group. If the data had been balanced, the UDL and LDL would be straight lines, because all estimates $\widehat{\eta}_{.i} - \widehat{\eta}_{.j}$ would have had the same standard error. The limits for the comparison between "Knox" and "Clarion" are wider than for other comparisons, because of the reduced sample size for the "Knox" soil type.

Figure 44.31 LS-Means Plot of Differences against a Control



The significance level of the decision limits is determined from the ALPHA= level in the LSMEANS statement. The default are 95% limits. If you choose one-sided comparisons with DIFF=CONTROLL or DIFF=CONTROLU in the LSMEANS statement, only one of the decision limits is drawn.

Analysis of Means (ANOM) Plot

The analysis of means in PROC GLIMMIX compares least squares means not by contrasting them against each other as with all pairwise differences or control differences. Instead, the least squares means are compared against an average value. Consequently, there are k comparisons for a factor with k levels. The following statements request ANOM differences for the Type least squares means (Figure 44.32) and plots the differences (Figure 44.33):

```
ods graphics on;
ods select Diffs AnomPlot;
proc glimmix data=plants order=data plots=AnomPlot;
   class Block Type;
   model StemLength = Block Type;
   lsmeans Type / diff=anom;
run;
ods graphics off;
```

Figure 44.32 ANOM LS-Mean Differences

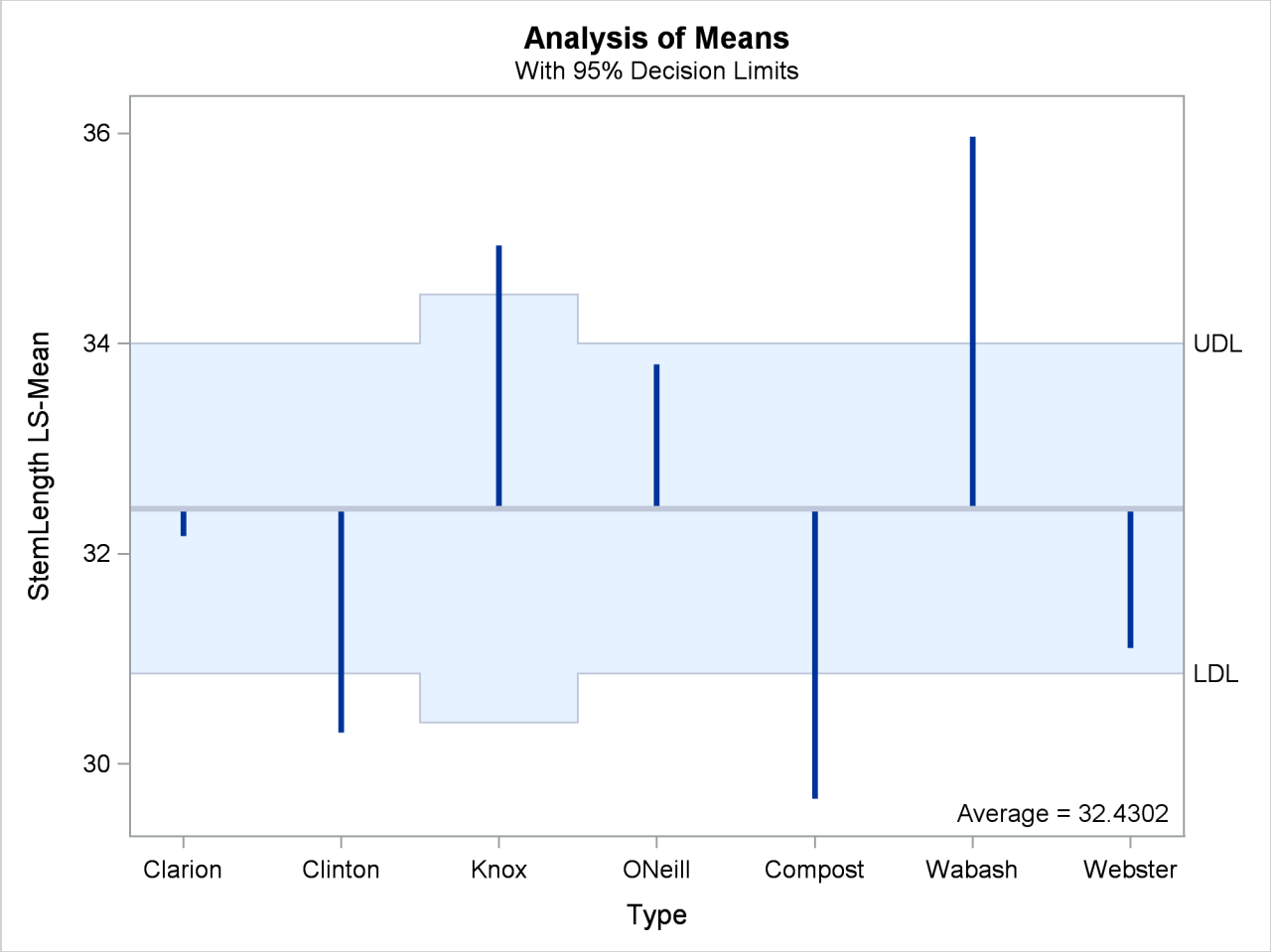
The GLIMMIX Procedure

Differences of Type Least Squares Means

Type	_Type	Estimate	Standard Error	DF	t Value	Pr > |t|
Clarion	Avg	-0.2635	0.7127	11	-0.37	0.7186
Clinton	Avg	-2.1302	0.7127	11	-2.99	0.0123
Knox	Avg	2.5032	0.9256	11	2.70	0.0205
ONeill	Avg	1.3698	0.7127	11	1.92	0.0809
Compost	Avg	-2.7635	0.7127	11	-3.88	0.0026
Wabash	Avg	3.5365	0.7127	11	4.96	0.0004
Webster	Avg	-1.3302	0.7127	11	-1.87	0.0888

At the 5% level, the "Clarion," "O'Neill," and "Webster" soil types are not significantly different from the average. Note that the artificial lack of balance introduced previously reduces the precision of the ANOM comparison for the "Knox" soil type.

Figure 44.33 LS-Means Analysis of Means (ANOM) Plot



The reference line in the ANOM plot is drawn at the average. Vertical lines extend from this reference line upward or downward, depending on the magnitude of the least squares means compared to the reference value. This enables you to quickly see which levels perform above and below the average. The horizontal reference lines are 95% upper and lower decision limits. If a vertical line crosses the limits, you conclude that the least squares mean is significantly different (at the 5% significance level) from the average. You can adjust the comparisons for multiplicity by adding the ADJUST=NELSON option in the LSMEANS statement.

Examples: GLIMMIX Procedure

Example 44.1: Binomial Counts in Randomized Blocks

In the context of spatial prediction in generalized linear models, Gotway and Stroup (1997) analyze data from an agronomic field trial. Researchers studied 16 varieties (entries) of wheat for their resistance to infestation

by the Hessian fly. They arranged the varieties in a randomized complete block design on an 8×8 grid. Each 4×4 quadrant of that arrangement constitutes a block.

The outcome of interest was the number of damaged plants (Y_{ij}) out of the total number of plants growing on the unit (n_{ij}). The two subscripts identify the block ($i = 1, \cdots, 4$) and the entry ($j = 1, \cdots, 16$). The following SAS statements create the data set. The variables lat and lng denote the coordinate of an experimental unit on the 8×8 grid.

```
data HessianFly;
   label Y = 'No. of damaged plants'
         n = 'No. of plants';
   input block entry lat lng n Y @@;
   datalines;
  1 14 1 1  8 2    1 16 1 2  9 1
  1  7 1 3 13 9    1  6 1 4  9 9
  1 13 2 1  9 2    1 15 2 2 14 7
  1  8 2 3  8 6    1  5 2 4 11 8
  1 11 3 1 12 7    1 12 3 2 11 8
  1  2 3 3 10 8    1  3 3 4 12 5
  1 10 4 1  9 7    1  9 4 2 15 8
  1  4 4 3 19 6    1  1 4 4  8 7
  2 15 5 1 15 6    2  3 5 2 11 9
  2 10 5 3 12 5    2  2 5 4  9 9
  2 11 6 1 20 10   2  7 6 2 10 8
  2 14 6 3 12 4    2  6 6 4 10 7
  2  5 7 1  8 8    2 13 7 2  6 0
  2 12 7 3  9 2    2 16 7 4  9 0
  2  9 8 1 14 9    2  1 8 2 13 12
  2  8 8 3 12 3    2  4 8 4 14 7
  3  7 1 5  7 7    3 13 1 6  7 0
  3  8 1 7 13 3    3 14 1 8  9 0
  3  4 2 5 15 11   3 10 2 6  9 7
  3  3 2 7 15 11   3  9 2 8 13 5
  3  6 3 5 16 9    3  1 3 6  8 8
  3 15 3 7  7 0    3 12 3 8 12 8
  3 11 4 5  8 1    3 16 4 6 15 1
  3  5 4 7 12 7    3  2 4 8 16 12
  4  9 5 5 15 8    4  4 5 6 10 6
  4 12 5 7 13 5    4  1 5 8 15 9
  4 15 6 5 17 6    4  6 6 6  8 2
  4 14 6 7 12 5    4  7 6 8 15 8
  4 13 7 5 13 2    4  8 7 6 13 9
  4  3 7 7  9 9    4 10 7 8  6 6
  4  2 8 5 12 8    4 11 8 6  9 7
  4  5 8 7 11 10   4 16 8 8 15 7
;
```

Analysis as a GLM

If infestations are independent among experimental units, and all plants within a unit have the same propensity for infestation, then the Y_{ij} are binomial random variables. The first model considered is a standard generalized linear model for independent binomial counts:

```
proc glimmix data=HessianFly;
   class block entry;
   model y/n = block entry / solution;
run;
```

The PROC GLIMMIX statement invokes the procedure. The CLASS statement instructs the GLIMMIX procedure to treat both block and entry as classification variables. The MODEL statement specifies the response variable and the fixed effects in the model. PROC GLIMMIX constructs the **X** matrix of the model from the terms on the right side of the MODEL statement. The GLIMMIX procedure supports two kinds of syntax for the response variable. This example uses the *events/trials* syntax. The variable y represents the number of successes (*events*) out of n Bernoulli *trials*. When the *events/trials* syntax is used, the GLIMMIX procedure automatically selects the binomial distribution as the response distribution. Once the distribution is determined, the procedure selects the link function for the model. The default link for binomial data is the logit link. The preceding statements are thus equivalent to the following statements:

```
proc glimmix data=HessianFly;
   class block entry;
   model y/n = block entry / dist=binomial link=logit solution;
run;
```

The SOLUTION option in the MODEL statement requests that solutions for the fixed effects (parameter estimates) be displayed.

The "Model Information" table describes the model and methods used in fitting the statistical model (Output 44.1.1).

The GLIMMIX procedure recognizes that this is a model for uncorrelated data (variance matrix is diagonal) and that parameters can be estimated by maximum likelihood. The default degrees-of-freedom method to denominator degrees of freedom for F tests and t tests is the RESIDUAL method. This corresponds to choosing $f - \mathrm{rank}(\mathbf{X})$ as the degrees of freedom, where f is the sum of the frequencies used in the analysis. You can change the degrees of freedom method with the DDFM= option in the MODEL statement.

Output 44.1.1 Model Information in GLM Analysis

The GLIMMIX Procedure

Model Information	
Data Set	WORK.HESSIANFLY
Response Variable (Events)	Y
Response Variable (Trials)	n
Response Distribution	Binomial
Link Function	Logit
Variance Function	Default
Variance Matrix	Diagonal
Estimation Technique	Maximum Likelihood
Degrees of Freedom Method	Residual

The "Class Level Information" table lists the levels of the variables specified in the CLASS statement and the ordering of the levels (Output 44.1.2). The "Number of Observations" table displays the number of observations read and used in the analysis.

Output 44.1.2 Class Level Information and Number of Observations

Class Level Information		
Class	**Levels**	**Values**
block	4	1 2 3 4
entry	16	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Number of Observations Read	64
Number of Observations Used	64
Number of Events	396
Number of Trials	736

The "Dimensions" table lists the size of relevant matrices (Output 44.1.3).

Output 44.1.3 Model Dimensions Information in GLM Analysis

Dimensions	
Columns in X	21
Columns in Z	0
Subjects (Blocks in V)	1
Max Obs per Subject	64

Because of the absence of G-side random effects in this model, there are no columns in the $\mathbf{Z}$ matrix. The 21 columns in the $\mathbf{X}$ matrix comprise the intercept, 4 columns for the block effect and 16 columns for the entry effect. Because no RANDOM statement with a SUBJECT= option was specified, the GLIMMIX procedure does not process the data by subjects (see the section "Processing by Subjects" on page 3218 for details about subject processing).

The "Optimization Information" table provides information about the methods and size of the optimization problem (Output 44.1.4).

Output 44.1.4 Optimization Information in GLM Analysis

Optimization Information	
Optimization Technique	Newton-Raphson
Parameters in Optimization	19
Lower Boundaries	0
Upper Boundaries	0
Fixed Effects	Not Profiled

With few exceptions, models fit with the GLIMMIX procedure require numerical methods for parameter estimation. The default optimization method for (overdispersed) GLM models is the Newton-Raphson algorithm. In this example, the optimization involves 19 parameters, corresponding to the number of linearly independent columns of the $\mathbf{X}'\mathbf{X}$ matrix.

The "Iteration History" table shows that the procedure converged after 3 iterations and 13 function evaluations (Output 44.1.5). The Change column measures the change in the objective function between iterations; however, this is not the monitored convergence criterion. The GLIMMIX procedure monitors several features simultaneously to determine whether to stop an optimization.

Output 44.1.5 Iteration History in GLM Analysis

Iteration History					
Iteration	Restarts	Evaluations	Objective Function	Change	Max Gradient
0	0	4	134.13393738	.	4.899609
1	0	3	132.85058236	1.28335502	0.206204
2	0	3	132.84724263	0.00333973	0.000698
3	0	3	132.84724254	0.00000009	3.029E-8

Convergence criterion (GCONV=1E-8) satisfied.

The "Fit Statistics" table lists information about the fitted model (Output 44.1.6). The –2 Log Likelihood values are useful for comparing nested models, and the information criteria AIC, AICC, BIC, CAIC, and HQIC are useful for comparing nonnested models. On average, the ratio between the Pearson statistic and its degrees of freedom should equal one in GLMs. Values larger than one indicate overdispersion. With a ratio of 2.37, these data appear to exhibit more dispersion than expected under a binomial model with block and varietal effects.

Output 44.1.6 Fit Statistics in GLM Analysis

Fit Statistics	
-2 Log Likelihood	265.69
AIC (smaller is better)	303.69
AICC (smaller is better)	320.97
BIC (smaller is better)	344.71
CAIC (smaller is better)	363.71
HQIC (smaller is better)	319.85
Pearson Chi-Square	106.74
Pearson Chi-Square / DF	2.37

The "Parameter Estimates" table displays the maximum likelihood estimates (Estimate), standard errors, and *t* tests for the hypothesis that the estimate is zero (Output 44.1.7).

Output 44.1.7 Parameter Estimates in GLM Analysis

Parameter Estimates

Effect	block	entry	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept			-1.2936	0.3908	45	-3.31	0.0018
block	1		-0.05776	0.2332	45	-0.25	0.8055
block	2		-0.1838	0.2303	45	-0.80	0.4289
block	3		-0.4420	0.2328	45	-1.90	0.0640
block	4		0	.	.	.	.
entry		1	2.9509	0.5397	45	5.47	<.0001
entry		2	2.8098	0.5158	45	5.45	<.0001
entry		3	2.4608	0.4956	45	4.97	<.0001
entry		4	1.5404	0.4564	45	3.38	0.0015
entry		5	2.7784	0.5293	45	5.25	<.0001
entry		6	2.0403	0.4889	45	4.17	0.0001
entry		7	2.3253	0.4966	45	4.68	<.0001
entry		8	1.3006	0.4754	45	2.74	0.0089
entry		9	1.5605	0.4569	45	3.42	0.0014
entry		10	2.3058	0.5203	45	4.43	<.0001
entry		11	1.4957	0.4710	45	3.18	0.0027
entry		12	1.5068	0.4767	45	3.16	0.0028
entry		13	-0.6296	0.6488	45	-0.97	0.3370
entry		14	0.4460	0.5126	45	0.87	0.3889
entry		15	0.8342	0.4698	45	1.78	0.0826
entry		16	0	.	.	.	.

The "Type III Tests of Fixed Effect" table displays significance tests for the two fixed effects in the model (Output 44.1.8).

Output 44.1.8 Type III Tests of Block and Entry Effects in GLM Analysis

Type III Tests of Fixed Effects

Effect	Num DF	Den DF	F Value	Pr > F
block	3	45	1.42	0.2503
entry	15	45	6.96	<.0001

These tests are Wald-type tests, not likelihood ratio tests. The entry effect is clearly significant in this model with a p-value of <0.0001, indicating that the 16 wheat varieties are not equally susceptible to infestation by the Hessian fly.

Analysis with Random Block Effects

There are several possible reasons for the overdispersion noted in Output 44.1.6 (Pearson ratio = 2.37). The data might not follow a binomial distribution, one or more important effects might not have been accounted for in the model, or the data might be positively correlated. If important fixed effects have been omitted, then you might need to consider adding them to the model. Because this is a designed experiment, it is reasonable not to expect further effects apart from the block and entry effects that represent the treatment and error control design structure. The reasons for the overdispersion must lie elsewhere.

If overdispersion stems from correlations among the observations, then the model should be appropriately adjusted. The correlation can have multiple sources. First, it might not be the case that the plants within an experimental unit responded independently. If the probability of infestation of a particular plant is altered by the infestation of a neighboring plant within the same unit, the infestation counts are not binomial and a different probability model should be used. A second possible source of correlations is the lack of independence of experimental units. Even if treatments were assigned to units at random, they might not respond independently. Shared spatial soil effects, for example, can be the underlying factor. The following analyses take these spatial effects into account.

First, assume that the environmental effects operate at the scale of the blocks. By making the block effects random, the marginal responses will be correlated due to the fact that observations within a block share the same random effects. Observations from different blocks will remain uncorrelated, in the spirit of separate randomizations among the blocks. The next set of statements fits a generalized linear mixed model (GLMM) with random block effects:

```
proc glimmix data=HessianFly;
   class block entry;
   model y/n = entry / solution;
   random block;
run;
```

Because the conditional distribution—conditional on the block effects—is binomial, the marginal distribution will be overdispersed relative to the binomial distribution. In contrast to adding a multiplicative scale parameter to the variance function, treating the block effects as random changes the estimates compared to a model with fixed block effects.

In the presence of random effects and a conditional binomial distribution, PROC GLIMMIX does not use maximum likelihood for estimation. Instead, the GLIMMIX procedure applies a restricted (residual) pseudo-likelihood algorithm (Output 44.1.9). The "restricted" attribute derives from the same rationale by which restricted (residual) maximum likelihood methods for linear mixed models attain their name; the likelihood equations are adjusted for the presence of fixed effects in the model to reduce bias in covariance parameter estimates.

Output 44.1.9 Model Information in GLMM Analysis

The GLIMMIX Procedure

Model Information	
Data Set	WORK.HESSIANFLY
Response Variable (Events)	Y
Response Variable (Trials)	n
Response Distribution	Binomial
Link Function	Logit
Variance Function	Default
Variance Matrix	Not blocked
Estimation Technique	Residual PL
Degrees of Freedom Method	Containment

The "Class Level Information" and "Number of Observations" tables are as before (Output 44.1.10).

Output 44.1.10 Class Level Information and Number of Observations

Class Level Information		
Class	**Levels**	**Values**
block	4	1 2 3 4
entry	16	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

Number of Observations Read	64
Number of Observations Used	64
Number of Events	396
Number of Trials	736

The "Dimensions" table indicates that there is a single G-side parameter, the variance of the random block effect (Output 44.1.11). The "Dimensions" table has changed from the previous model (compare Output 44.1.11 to Output 44.1.3). Note that although the block effect has four levels, only a single variance component is estimated. The $\mathbf{Z}$ matrix has four columns, however, corresponding to the four levels of the block effect. Because no SUBJECT= option is used in the RANDOM statement, the GLIMMIX procedure treats these data as having arisen from a single subject with 64 observations.

Output 44.1.11 Model Dimensions Information in GLMM Analysis

Dimensions	
G-side Cov. Parameters	1
Columns in X	17
Columns in Z	4
Subjects (Blocks in V)	1
Max Obs per Subject	64

The "Optimization Information" table indicates that a quasi-Newton method is used to solve the optimization problem. This is the default optimization method for GLMM models (Output 44.1.12).

Output 44.1.12 Optimization Information in GLMM Analysis

Optimization Information	
Optimization Technique	Dual Quasi-Newton
Parameters in Optimization	1
Lower Boundaries	1
Upper Boundaries	0
Fixed Effects	Profiled
Starting From	Data

In contrast to the Newton-Raphson method, the quasi-Newton method does not require second derivatives. Because the covariance parameters are not unbounded in this example, the procedure enforces a lower boundary constraint (zero) for the variance of the block effect, and the optimization method is changed to a dual quasi-Newton method. The fixed effects are profiled from the likelihood equations in this model. The resulting optimization problem involves only the covariance parameters.

The "Iteration History" table appears to indicate that the procedure converged after four iterations (Output 44.1.13). Notice, however, that this table has changed slightly from the previous analysis (see Output 44.1.5). The Evaluations column has been replaced by the Subiterations column, because the GLIMMIX

procedure applied a doubly iterative fitting algorithm. The entire process consisted of five optimizations, each of which was iterative. The initial optimization required four iterations, the next one required three iterations, and so on.

Output 44.1.13 Iteration History in GLMM Analysis

Iteration History

Iteration	Restarts	Subiterations	Objective Function	Change	Max Gradient
0	0	4	173.28473428	0.81019251	0.000197
1	0	3	181.66726674	0.17550228	0.000739
2	0	2	182.20789493	0.00614874	7.018E-6
3	0	1	182.21315596	0.00004386	1.213E-8
4	0	0	182.21317662	0.00000000	3.349E-6

Convergence criterion (PCONV=1.11022E-8) satisfied.

The "Fit Statistics" table shows information about the fit of the GLMM (Output 44.1.14). The log likelihood reported in the table is not the residual log likelihood of the data. It is the residual log likelihood for an approximated model. The generalized chi-square statistic measures the residual sum of squares in the final model, and the ratio with its degrees of freedom is a measure of variability of the observation about the mean model.

Output 44.1.14 Fit Statistics in GLMM Analysis

Fit Statistics	
-2 Res Log Pseudo-Likelihood	182.21
Generalized Chi-Square	107.96
Gener. Chi-Square / DF	2.25

The variance of the random block effects is rather small (Output 44.1.15).

Output 44.1.15 Estimated Covariance Parameters and Approximate Standard Errors

Covariance Parameter Estimates

Cov Parm	Estimate	Standard Error
block	0.01116	0.03116

If the environmental effects operate on a spatial scale smaller than the block size, the random block model does not provide a suitable adjustment. From the coarse layout of the experimental area, it is not surprising that random block effects alone do not account for the overdispersion in the data. Adding a random component to a generalized linear model is different from adding a multiplicative overdispersion component, for example, via the PSCALE option in PROC GENMOD or a

```
random _residual_;
```

statement in PROC GLIMMIX. Such overdispersion components do not affect the parameter estimates, only their standard errors. A genuine random effect, on the other hand, affects both the parameter estimates and their standard errors (compare Output 44.1.16 to Output 44.1.7).

Output 44.1.16 Parameter Estimates for Fixed Effects in GLMM Analysis

Solutions for Fixed Effects

Effect	entry	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept		-1.4637	0.3738	3	-3.92	0.0296
entry	1	2.9609	0.5384	45	5.50	<.0001
entry	2	2.7807	0.5138	45	5.41	<.0001
entry	3	2.4339	0.4934	45	4.93	<.0001
entry	4	1.5347	0.4542	45	3.38	0.0015
entry	5	2.7653	0.5276	45	5.24	<.0001
entry	6	2.0014	0.4865	45	4.11	0.0002
entry	7	2.3518	0.4952	45	4.75	<.0001
entry	8	1.2927	0.4739	45	2.73	0.0091
entry	9	1.5663	0.4554	45	3.44	0.0013
entry	10	2.2896	0.5179	45	4.42	<.0001
entry	11	1.5018	0.4682	45	3.21	0.0025
entry	12	1.5075	0.4752	45	3.17	0.0027
entry	13	-0.5955	0.6475	45	-0.92	0.3626
entry	14	0.4573	0.5111	45	0.89	0.3758
entry	15	0.8683	0.4682	45	1.85	0.0702
entry	16	0	.	.	.	.

Output 44.1.17 Type III Test of Entry in GLMM Analysis

Type III Tests of Fixed Effects

Effect	Num DF	Den DF	F Value	Pr > F
entry	15	45	6.90	<.0001

Because the block variance component is small, the Type III test for the variety effect in Output 44.1.17 is affected only very little compared to the GLM (Output 44.1.8).

Analysis with Smooth Spatial Trends

You can also consider these data in an observational sense, where the covariation of the observations is subject to modeling. Rather than deriving model components from the experimental design alone, environmental effects can be modeled by adjusting the mean and/or correlation structure. Gotway and Stroup (1997) and Schabenberger and Pierce (2002) supplant the coarse block effects with smooth-scale spatial components.

The model considered by Gotway and Stroup (1997) is a marginal model in that the correlation structure is modeled through residual-side (R-side) random components. This exponential covariance model is fit with the following statements:

```
proc glimmix data=HessianFly;
   class entry;
   model y/n = entry / solution ddfm=contain;
   random _residual_ / subject=intercept type=sp(exp)(lng lat);
run;
```

Note that the block effects have been removed from the statements. The keyword _RESIDUAL_ in the RANDOM statement instructs the GLIMMIX procedure to model the **R** matrix. Here, **R** is to be modeled as an exponential covariance structure matrix. The SUBJECT=INTERCEPT option means that all observations are considered correlated. Because the random effects are residual-type (R-side) effects, there are no columns in the **Z** matrix for this model (Output 44.1.18).

Output 44.1.18 Model Dimension Information in Marginal Spatial Analysis

The GLIMMIX Procedure

Dimensions	
R-side Cov. Parameters	2
Columns in X	17
Columns in Z per Subject	0
Subjects (Blocks in V)	1
Max Obs per Subject	64

In addition to the fixed effects, the GLIMMIX procedure now profiles one of the covariance parameters, the variance of the exponential covariance model (Output 44.1.19). This reduces the size of the optimization problem. Only a single parameter is part of the optimization, the "range" (SP(EXP)) of the spatial process.

Output 44.1.19 Optimization Information in Spatial Analysis

Optimization Information	
Optimization Technique	Dual Quasi-Newton
Parameters in Optimization	1
Lower Boundaries	1
Upper Boundaries	0
Fixed Effects	Profiled
Residual Variance	Profiled
Starting From	Data

The practical range of a spatial process is that distance at which the correlation between data points has decreased to at most 0.05. The parameter reported by the GLIMMIX procedure as SP(EXP) in Output 44.1.20 corresponds to one-third of the practical range. The practical range in this process is $3 \times 0.9052 = 2.7156$. Correlations extend beyond a single experimental unit, but they do not appear to exist on the scale of the block size.

Output 44.1.20 Estimates of Covariance Parameters

Covariance Parameter Estimates			
Cov Parm	Subject	Estimate	Standard Error
SP(EXP)	Intercept	0.9052	0.4404
Residual		2.5315	0.6974

The sill of the spatial process, the variance of the underlying residual effect, is estimated as 2.5315.

Output 44.1.21 Type III Test of Entry Effect in Spatial Analysis

Type III Tests of Fixed Effects

Effect	Num DF	Den DF	F Value	Pr > F
entry	15	48	3.60	0.0004

The F value for the entry effect has been sharply reduced compared to the previous analyses. The smooth spatial variation accounts for some of the variation among the varieties (Output 44.1.21).

In this example three models were considered for the analysis of a randomized block design with binomial outcomes. If data are correlated, a standard generalized linear model often will indicate overdispersion relative to the binomial distribution. Two courses of action are considered in this example to address this overdispersion. First, the inclusion of G-side random effects models the correlation indirectly; it is induced through the sharing of random effects among responses from the same block. Second, the R-side spatial covariance structure models covariation directly. In generalized linear (mixed) models these two modeling approaches can lead to different inferences, because the models have different interpretation. The random block effects are modeled on the linked (logit) scale, and the spatial effects were modeled on the mean scale. Only in a linear mixed model are the two scales identical.

Example 44.2: Mating Experiment with Crossed Random Effects

McCullagh and Nelder (1989, Ch. 14.5) describe a mating experiment—conducted by S. Arnold and P. Verell at the University of Chicago, Department of Ecology and Evolution—involving two geographically isolated populations of mountain dusky salamanders. One goal of the experiment was to determine whether barriers to interbreeding have evolved in light of the geographical isolation of the populations. In this case, matings within a population should be more successful than matings between the populations. The experiment conducted in the summer of 1986 involved 40 animals, 20 rough butt (R) and 20 whiteside (W) salamanders, with equal numbers of males and females. The animals were grouped into two sets of R males, two sets of R females, two sets of W males, and two sets of W females, so that each set comprised five salamanders. Each set was mated against one rough butt and one whiteside set, creating eight crossings. Within the pairings of sets, each female was paired to three male animals. The salamander mating data have been used by a number of authors; see, for example, McCullagh and Nelder (1989); Schall (1991); Karim and Zeger (1992); Breslow and Clayton (1993); Wolfinger and O'Connell (1993); Shun (1997).

The following DATA step creates the data set for the analysis.

```
data salamander;
   input day fpop$ fnum mpop$ mnum mating @@;
   datalines;
4  rb  1 rb  1 1  4  rb  2 rb  5 1
4  rb  3 rb  2 1  4  rb  4 rb  4 1
4  rb  5 rb  3 1  4  rb  6 ws  9 1
4  rb  7 ws  8 0  4  rb  8 ws  6 0
4  rb  9 ws 10 0  4  rb 10 ws  7 0
4  ws  1 rb  9 0  4  ws  2 rb  7 0
4  ws  3 rb  8 0  4  ws  4 rb 10 0
4  ws  5 rb  6 0  4  ws  6 ws  5 0
4  ws  7 ws  4 1  4  ws  8 ws  1 1
```

```
4  ws  9 ws  3 1  4  ws 10 ws  2 1
8  rb  1 ws  4 1  8  rb  2 ws  5 1
8  rb  3 ws  1 0  8  rb  4 ws  2 1
8  rb  5 ws  3 1  8  rb  6 rb  9 1
8  rb  7 rb  8 0  8  rb  8 rb  6 1
8  rb  9 rb  7 0  8  rb 10 rb 10 0
8  ws  1 ws  9 1  8  ws  2 ws  6 0
8  ws  3 ws  7 0  8  ws  4 ws 10 1
8  ws  5 ws  8 1  8  ws  6 rb  2 0
8  ws  7 rb  1 1  8  ws  8 rb  4 0
8  ws  9 rb  3 1  8  ws 10 rb  5 0
12 rb  1 rb  5 1  12 rb  2 rb  3 1
12 rb  3 rb  1 1  12 rb  4 rb  2 1
12 rb  5 rb  4 1  12 rb  6 ws 10 1
12 rb  7 ws  9 0  12 rb  8 ws  7 0
12 rb  9 ws  8 1  12 rb 10 ws  6 1
12 ws  1 rb  7 1  12 ws  2 rb  9 0
12 ws  3 rb  6 0  12 ws  4 rb  8 1
12 ws  5 rb 10 0  12 ws  6 ws  3 1
12 ws  7 ws  5 1  12 ws  8 ws  2 1
12 ws  9 ws  1 1  12 ws 10 ws  4 0
16 rb  1 ws  1 0  16 rb  2 ws  3 1
16 rb  3 ws  4 1  16 rb  4 ws  5 0
16 rb  5 ws  2 1  16 rb  6 rb  7 0
16 rb  7 rb  9 1  16 rb  8 rb 10 0
16 rb  9 rb  6 1  16 rb 10 rb  8 0
16 ws  1 ws 10 1  16 ws  2 ws  7 1
16 ws  3 ws  9 0  16 ws  4 ws  8 1
16 ws  5 ws  6 0  16 ws  6 rb  4 0
16 ws  7 rb  2 0  16 ws  8 rb  5 0
16 ws  9 rb  1 1  16 ws 10 rb  3 1
20 rb  1 rb  4 1  20 rb  2 rb  1 1
20 rb  3 rb  3 1  20 rb  4 rb  5 1
20 rb  5 rb  2 1  20 rb  6 ws  6 1
20 rb  7 ws  7 0  20 rb  8 ws 10 1
20 rb  9 ws  9 1  20 rb 10 ws  8 1
20 ws  1 rb 10 0  20 ws  2 rb  6 0
20 ws  3 rb  7 0  20 ws  4 rb  9 0
20 ws  5 rb  8 0  20 ws  6 ws  2 0
20 ws  7 ws  1 1  20 ws  8 ws  5 1
20 ws  9 ws  4 1  20 ws 10 ws  3 1
24 rb  1 ws  5 1  24 rb  2 ws  2 1
24 rb  3 ws  3 1  24 rb  4 ws  4 1
24 rb  5 ws  1 1  24 rb  6 rb  8 1
24 rb  7 rb  6 0  24 rb  8 rb  9 1
24 rb  9 rb 10 1  24 rb 10 rb  7 0
24 ws  1 ws  8 1  24 ws  2 ws 10 0
24 ws  3 ws  6 1  24 ws  4 ws  9 1
24 ws  5 ws  7 0  24 ws  6 rb  1 0
24 ws  7 rb  5 1  24 ws  8 rb  3 0
24 ws  9 rb  4 0  24 ws 10 rb  2 0
;
```

The first observation, for example, indicates that rough butt female 1 was paired in the laboratory on day 4 of the experiment with rough butt male 1, and the pair mated. On the same day rough butt female 7 was paired with whiteside male 8, but the pairing did not result in mating of the animals.

The model adopted by many authors for these data comprises fixed effects for gender and population, their interaction, and male and female random effects. Specifically, let π_{RR}, π_{RW}, π_{WR}, and π_{WW} denote the mating probabilities between the populations, where the first subscript identifies the female partner of the pair. Then, you model

$$\log\left\{\frac{\pi_{kl}}{1-\pi_{kl}}\right\} = \tau_{kl} + \gamma_f + \gamma_m \quad k,l \in \{R,W\}$$

where γ_f and γ_m are independent random variables representing female and male random effects (20 each), and τ_{kl} denotes the average logit of mating between females of population k and males of population l. The following statements fit this model by pseudo-likelihood:

```
proc glimmix data=salamander;
   class fpop fnum mpop mnum;
   model mating(event='1') = fpop|mpop / dist=binary;
   random fpop*fnum mpop*mnum;
   lsmeans fpop*mpop / ilink;
run;
```

The response variable is the two-level variable mating. Because it is coded as zeros and ones, and because PROC GLIMMIX models by default the probability of the first level according to the response-level ordering, the EVENT='1' option instructs PROC GLIMMIX to model the probability of a successful mating. The distribution of the mating variable, conditional on the random effects, is binary.

The fpop*fnum effect in the RANDOM statement creates a random intercept for each female animal. Because fpop and fnum are CLASS variables, the effect has 20 levels (10 rb and 10 ws females). Similarly, the mpop*mnum effect creates the random intercepts for the male animals. Because no TYPE= is specified in the RANDOM statement, the covariance structure defaults to TYPE=VC. The random effects and their levels are independent, and each effect has its own variance component. Because the conditional distribution of the data, conditioned on the random effects, is binary, no extra scale parameter (ϕ) is added.

The LSMEANS statement requests least squares means for the four levels of the fpop*mpop effect, which are estimates of the cell means in the 2×2 classification of female and male populations. The ILINK option in the LSMEANS statement requests that the estimated means and standard errors are also reported on the scale of the data. This yields estimates of the four mating probabilities, π_{RR}, π_{RW}, π_{WR}, and π_{WW}.

The "Model Information" table displays general information about the model being fit (Output 44.2.1).

Output 44.2.1 Analysis of Mating Experiment with Crossed Random Effects

The GLIMMIX Procedure

Model Information	
Data Set	WORK.SALAMANDER
Response Variable	mating
Response Distribution	Binary
Link Function	Logit
Variance Function	Default
Variance Matrix	Not blocked
Estimation Technique	Residual PL
Degrees of Freedom Method	Containment

The response variable mating follows a binary distribution (conditional on the random effects). Hence, the mean of the data is an event probability, π, and the logit of this probability is linearly related to the linear predictor of the model. The variance function is the default function that is implied by the distribution, $a(\pi) = \pi(1-\pi)$. The variance matrix is not blocked, because the GLIMMIX procedure does not process the data by subjects (see the section "Processing by Subjects" on page 3218 for details). The estimation technique is the default method for GLMMs, residual pseudo-likelihood (METHOD=RSPL), and degrees of freedom for tests and confidence intervals are determined by the containment method.

The "Class Level Information" table in Output 44.2.2 lists the levels of the variables listed in the CLASS statement, as well as the order of the levels.

Output 44.2.2 Class Level Information and Number of Observations

Class Level Information		
Class	**Levels**	**Values**
fpop	2	rb ws
fnum	10	1 2 3 4 5 6 7 8 9 10
mpop	2	rb ws
mnum	10	1 2 3 4 5 6 7 8 9 10

Number of Observations Read	120
Number of Observations Used	120

Note that there are two female populations and two male populations; also, the variables fnum and mnum have 10 levels each. As a consequence, the effects fpop*fnum and mpop*mnum identify the 20 females and males, respectively. The effect fpop*mpop identifies the four mating types.

The "Response Profile Table," which is displayed for binary or multinomial data, lists the levels of the response variable and their order (Output 44.2.3). With binary data, the table also provides information about which level of the response variable defines the event. Because of the EVENT='1' response variable option in the MODEL statement, the probability being modeled is that of the higher-ordered value.

Output 44.2.3 Response Profiles

Response Profile		
Ordered Value	mating	Total Frequency
1	0	50
2	1	70

The GLIMMIX procedure is modeling the probability that mating='1'.

There are two covariance parameters in this model, the variance of the fpop*fnum effect and the variance of the mpop*mnum effect (Output 44.2.4). Both parameters are modeled as G-side parameters. The nine columns in the **X** matrix comprise the intercept, two columns each for the levels of the fpop and mpop effects, and four columns for their interaction. The **Z** matrix has 40 columns, one for each animal. Because the data are not processed by subjects, PROC GLIMMIX assumes the data consist of a single subject (a single block in **V**).

Output 44.2.4 Model Dimensions Information

Dimensions	
G-side Cov. Parameters	2
Columns in X	9
Columns in Z	40
Subjects (Blocks in V)	1
Max Obs per Subject	120

The "Optimization Information" table displays basic information about the optimization (Output 44.2.5). The default technique for GLMMs is the quasi-Newton method. There are two parameters in the optimization, which correspond to the two variance components. The 17 fixed effects parameters are not part of the optimization. The initial optimization computes pseudo-data based on the response values in the data set rather than from estimates of a generalized linear model fit.

Output 44.2.5 Optimization Information

Optimization Information	
Optimization Technique	Newton-Raphson with Ridging
Parameters in Optimization	2
Lower Boundaries	2
Upper Boundaries	0
Fixed Effects	Profiled
Starting From	Data

The GLIMMIX procedure performs eight optimizations after the initial optimization (Output 44.2.6). That is, following the initial pseudo-data creation, the pseudo-data were updated eight more times and a total of nine linear mixed models were estimated.

Output 44.2.6 Iteration History and Convergence Status

Iteration History					
Iteration	Restarts	Subiterations	Objective Function	Change	Max Gradient
0	0	4	537.09173501	2.00000000	1.719E-8
1	0	3	544.12516903	0.66319780	1.14E-8
2	0	2	545.89139118	0.13539318	1.609E-6
3	0	2	546.10489538	0.01742065	5.89E-10
4	0	1	546.13075146	0.00212475	9.654E-7
5	0	1	546.13374731	0.00025072	1.346E-8
6	0	1	546.13409761	0.00002931	1.84E-10
7	0	0	546.13413861	0.00000000	4.285E-6

Convergence criterion (PCONV=1.11022E-8) satisfied.

The "Covariance Parameter Estimates" table lists the estimates for the two variance components and their estimated standard errors (Output 44.2.7). The heterogeneity (in the logit of the mating probabilities) among the females is considerably larger than the heterogeneity among the males.

Output 44.2.7 Estimated Covariance Parameters and Approximate Standard Errors

Covariance Parameter Estimates		
Cov Parm	Estimate	Standard Error
fpop*fnum	1.4099	0.8871
mpop*mnum	0.08963	0.4102

The "Type III Tests of Fixed Effects" table indicates a significant interaction between the male and female populations (Output 44.2.8). A comparison in the logits of mating success in pairs with R females and W females depends on whether the male partner in the pair is the same species. The "fpop*mpop Least Squares Means" table shows this effect more clearly (Output 44.2.9).

Output 44.2.8 Tests of Main Effects and Interaction

Type III Tests of Fixed Effects				
Effect	Num DF	Den DF	F Value	Pr > F
fpop	1	18	2.86	0.1081
mpop	1	17	4.71	0.0444
fpop*mpop	1	81	9.61	0.0027

Output 44.2.9 Interaction Least Squares Means

fpop*mpop Least Squares Means

fpop	mpop	Estimate	Standard Error	DF	t Value	Pr > |t|	Mean	Standard Error Mean
rb	rb	1.1629	0.5961	81	1.95	0.0545	0.7619	0.1081
rb	ws	0.7839	0.5729	81	1.37	0.1750	0.6865	0.1233
ws	rb	-1.4119	0.6143	81	-2.30	0.0241	0.1959	0.09678
ws	ws	1.0151	0.5871	81	1.73	0.0876	0.7340	0.1146

In a pairing with a male rough butt salamander, the logit drops sharply from 1.1629 to –1.4119 when the male is paired with a whiteside female instead of a female from its own population. The corresponding estimated probabilities of mating success are $\widehat{\pi}_{RR} = 0.7619$ and $\widehat{\pi}_{WR} = 0.1959$. If the same comparisons are made in pairs with whiteside males, then you also notice a drop in the logit if the female comes from a different population, 1.0151 versus 0.7839. The change is considerably less, though, corresponding to mating probabilities of $\widehat{\pi}_{WW} = 0.7340$ and $\widehat{\pi}_{RW} = 0.6865$. Whiteside females appear to be successful with their own population. Whiteside males appear to succeed equally well with female partners of the two populations.

This insight into the factor-level comparisons can be amplified by graphing the least squares mean comparisons and by subsetting the differences of least squares means. This is accomplished with the following statements:

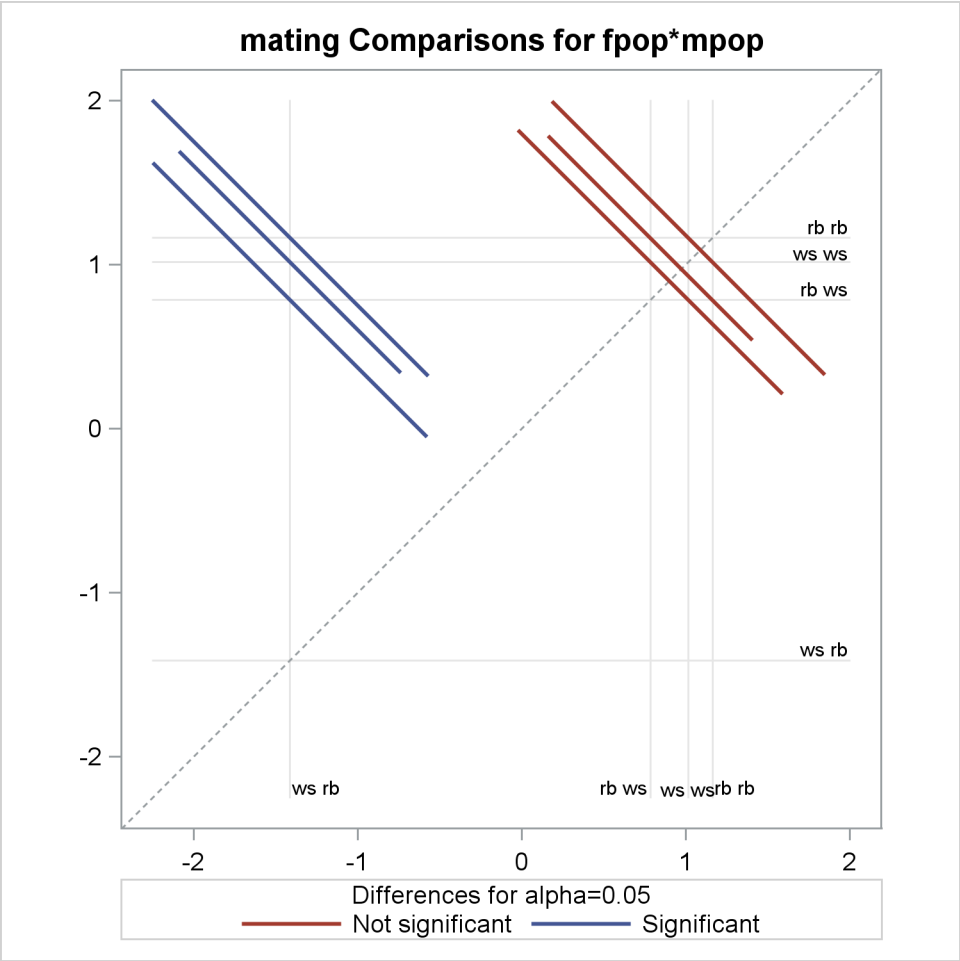
```
ods graphics on;
ods select DiffPlot SliceDiffs;
proc glimmix data=salamander;
   class fpop fnum mpop mnum;
   model mating(event='1') = fpop|mpop / dist=binary;
   random fpop*fnum mpop*mnum;
   lsmeans fpop*mpop / plots=diffplot;
   lsmeans fpop*mpop / slicediff=(mpop fpop);
run;
ods graphics off;
```

The PLOTS=DIFFPLOT option in the first LSMEANS statement requests a comparison plot that displays the result of all pairwise comparisons (Output 44.2.10). The SLICEDIFF=(mpop fpop) option requests differences of simple effects.

The comparison plot in Output 44.2.10 is also known as a mean-mean scatter plot (Hsu 1996). Each solid line in the plot corresponds to one of the possible $4 \times 3/2 = 6$ unique pairwise comparisons. The line is centered at the intersection of two least squares means, and the length of the line segments corresponds to the width of a 95% confidence interval for the difference between the two least squares means. The length of the segment is adjusted for the rotation. If a line segment crosses the dashed 45-degree line, the comparison between the two factor levels is not significant; otherwise, it is significant. The horizontal and vertical axes of the plot are drawn in least squares means units, and the grid lines are placed at the values of the least squares means.

The six pairs of least squares means comparisons separate into two sets of three pairs. Comparisons in the first set are significant; comparisons in the second set are not significant. For the significant set, the female partner in one of the pairs is a whiteside salamander. For the nonsignificant comparisons, the male partner in one of the pairs is a whiteside salamander.

Output 44.2.10 LS-Means Diffogram



The "Simple Effect Comparisons" tables show the results of the SLICEDIFF= option in the second LSMEANS statement (Output 44.2.11).

Output 44.2.11 Simple Effect Comparisons

Simple Effect Comparisons of fpop*mpop Least Squares Means By mpop

Simple Effect Level	fpop	_fpop	Estimate	Standard Error	DF	t Value	Pr > |t|
mpop rb	rb	ws	2.5748	0.8458	81	3.04	0.0031
mpop ws	rb	ws	-0.2312	0.8092	81	-0.29	0.7758

Simple Effect Comparisons of fpop*mpop Least Squares Means By fpop

Simple Effect Level	mpop	_mpop	Estimate	Standard Error	DF	t Value	Pr > |t|
fpop rb	rb	ws	0.3790	0.6268	81	0.60	0.5471
fpop ws	rb	ws	-2.4270	0.6793	81	-3.57	0.0006

The first table of simple effect comparisons holds fixed the level of the mpop factor and compares the levels of the fpop factor. Because there is only one possible comparison for each male population, there are two entries in the table. The first entry compares the logits of mating probabilities when the male partner is a rough butt, and the second entry applies when the male partner is from the whiteside population. The second table of simple effects comparisons applies the same logic, but holds fixed the level of the female partner in the pair. Note that these four comparisons are a subset of all six possible comparisons, eliminating those where both factors are varied at the same time. The simple effect comparisons show that there is no difference in mating probabilities if the male partner is a whiteside salamander, or if the female partner is a rough butt. Rough butt females also appear to mate indiscriminately.

Example 44.3: Smoothing Disease Rates; Standardized Mortality Ratios

Clayton and Kaldor (1987, Table 1) present data on observed and expected cases of lip cancer in the 56 counties of Scotland between 1975 and 1980. The expected number of cases was determined by a separate multiplicative model that accounted for the age distribution in the counties. The goal of the analysis is to estimate the county-specific log-relative risks, also known as standardized mortality ratios (SMR).

If Y_i is the number of incident cases in county i and E_i is the expected number of incident cases, then the ratio of observed to expected counts, Y_i/E_i, is the standardized mortality ratio. Clayton and Kaldor (1987) assume there exists a relative risk λ_i that is specific to each county and is a random variable. Conditional on λ_i, the observed counts are independent Poisson variables with mean $E_i\lambda_i$.

An elementary mixed model for λ_i specifies only a random intercept for each county, in addition to a fixed intercept. Breslow and Clayton (1993), in their analysis of these data, also provide a covariate that measures the percentage of employees in agriculture, fishing, and forestry. The expanded model for the region-specific relative risk in Breslow and Clayton (1993) is

$$\lambda_i = \exp\{\beta_0 + \beta_1 x_i/10 + \gamma_i\}, \quad i = 1, \cdots, 56$$

where β_0 and β_1 are fixed effects, and the γ_i are county random effects.

The following DATA step creates the data set lipcancer. The expected number of cases is based on the observed standardized mortality ratio for counties with lip cancer cases, and based on the expected counts reported by Clayton and Kaldor (1987, Table 1) for the counties without cases. The sum of the expected counts then equals the sum of the observed counts.

```
data lipcancer;
   input county observed expected employment SMR;
   if (observed > 0) then expCount = 100*observed/SMR;
   else expCount = expected;
   datalines;
 1  9  1.4 16 652.2
 2 39  8.7 16 450.3
 3 11  3.0 10 361.8
 4  9  2.5 24 355.7
 5 15  4.3 10 352.1
 6  8  2.4 24 333.3
 7 26  8.1 10 320.6
 8  7  2.3  7 304.3
 9  6  2.0  7 303.0
10 20  6.6 16 301.7
11 13  4.4  7 295.5
```

```
12  5  1.8 16 279.3
13  3  1.1 10 277.8
14  8  3.3 24 241.7
15 17  7.8  7 216.8
16  9  4.6 16 197.8
17  2  1.1 10 186.9
18  7  4.2  7 167.5
19  9  5.5  7 162.7
20  7  4.4 10 157.7
21 16 10.5  7 153.0
22 31 22.7 16 136.7
23 11  8.8 10 125.4
24  7  5.6  7 124.6
25 19 15.5  1 122.8
26 15 12.5  1 120.1
27  7  6.0  7 115.9
28 10  9.0  7 111.6
29 16 14.4 10 111.3
30 11 10.2 10 107.8
31  5  4.8  7 105.3
32  3  2.9 24 104.2
33  7  7.0 10  99.6
34  8  8.5  7  93.8
35 11 12.3  7  89.3
36  9 10.1  0  89.1
37 11 12.7 10  86.8
38  8  9.4  1  85.6
39  6  7.2 16  83.3
40  4  5.3  0  75.9
41 10 18.8  1  53.3
42  8 15.8 16  50.7
43  2  4.3 16  46.3
44  6 14.6  0  41.0
45 19 50.7  1  37.5
46  3  8.2  7  36.6
47  2  5.6  1  35.8
48  3  9.3  1  32.1
49 28 88.7  0  31.6
50  6 19.6  1  30.6
51  1  3.4  1  29.1
52  1  3.6  0  27.6
53  1  5.7  1  17.4
54  1  7.0  1  14.2
55  0  4.2 16   0.0
56  0  1.8 10   0.0
;
```

Because the mean of the Poisson variates, conditional on the random effects, is $\mu_i = E_i\lambda_i$, applying a log link yields

$$\log\{\mu_i\} = \log\{E_i\} + \beta_0 + \beta_1 x_i/10 + \gamma_i$$

The term $\log\{E_i\}$ is an offset, a regressor variable whose coefficient is known to be one. Note that it is assumed that the E_i are known; they are not treated as random variables.

The following statements fit this model by residual pseudo-likelihood:

```
proc glimmix data=lipcancer;
   class county;
   x    = employment / 10;
   logn = log(expCount);
   model observed = x / dist=poisson offset=logn
                        solution ddfm=none;
   random county;
   SMR_pred = 100*exp(_zgamma_ + _xbeta_);
   id employment SMR SMR_pred;
   output out=glimmixout;
run;
```

The offset is created with the assignment statement

```
logn = log(expCount);
```

and is associated with the linear predictor through the OFFSET= option in the MODEL statement. The statement

```
x = employment / 10;
```

transforms the covariate measuring percentage of employment in agriculture, fisheries, and forestry to agree with the analysis of Breslow and Clayton (1993). The DDFM=NONE option in the MODEL statement requests chi-square tests and z tests instead of the default F tests and t tests by setting the denominator degrees of freedom in tests of fixed effects to ∞.

The statement

```
SMR_pred = 100*exp(_zgamma_ + _xbeta_);
```

calculates the fitted standardized mortality rate. Note that the offset variable does not contribute to the exponentiated term.

The OUTPUT statement saves results of the calculations to the output data set glimmixout. The ID statement specifies that only the listed variables are written to the output data set.

Output 44.3.1 Model Information in Poisson GLMM

The GLIMMIX Procedure

Model Information	
Data Set	WORK.LIPCANCER
Response Variable	observed
Response Distribution	Poisson
Link Function	Log
Variance Function	Default
Offset Variable	logn = log(expCount);
Variance Matrix	Not blocked
Estimation Technique	Residual PL
Degrees of Freedom Method	None

Output 44.3.1 *continued*

Class Level Information		
Class	**Levels**	**Values**
county	56	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56

Number of Observations Read	56
Number of Observations Used	56

Dimensions	
G-side Cov. Parameters	1
Columns in X	2
Columns in Z	56
Subjects (Blocks in V)	1
Max Obs per Subject	56

The GLIMMIX procedure displays in the "Model Information" table that the offset variable was computed with programming statements and the final assignment statement from your GLIMMIX statements (Output 44.3.1). There are two columns in the $\mathbf{X}$ matrix, corresponding to the intercept and the regressor $x/10$. There are 56 columns in the $\mathbf{Z}$ matrix, however, one for each observation in the data set (Output 44.3.1).

The optimization involves only a single covariance parameter, the variance of the county effect (Output 44.3.2). Because this parameter is a variance, the GLIMMIX procedure imposes a lower boundary constraint; the solution for the variance is bounded by zero from below.

Output 44.3.2 Optimization Information in Poisson GLMM

Optimization Information	
Optimization Technique	Dual Quasi-Newton
Parameters in Optimization	1
Lower Boundaries	1
Upper Boundaries	0
Fixed Effects	Profiled
Starting From	Data

Following the initial creation of pseudo-data and the fit of a linear mixed model, the procedure goes through five more updates of the pseudo-data, each associated with a separate optimization (Output 44.3.3). Although the objective function in each optimization is the negative of twice the restricted maximum likelihood for that pseudo-data, there is no guarantee that across the outer iterations the objective function decreases in subsequent optimizations. In this example, minus twice the residual maximum likelihood at convergence takes on its smallest value at the initial optimization and increases in subsequent optimizations.

Output 44.3.3 Iteration History in Poisson GLMM

Iteration History

Iteration	Restarts	Subiterations	Objective Function	Change	Max Gradient
0	0	4	123.64113992	0.20997891	3.848E-8
1	0	3	127.05866018	0.03393332	0.000048
2	0	2	127.48839749	0.00223427	5.753E-6
3	0	1	127.50502469	0.00006946	1.938E-7
4	0	1	127.50528068	0.00000118	1.09E-7
5	0	0	127.50528481	0.00000000	1.299E-6

Convergence criterion (PCONV=1.11022E-8) satisfied.

The "Covariance Parameter Estimates" table in Output 44.3.4 shows the estimate of the variance of the region-specific log-relative risks. There is significant county-to-county heterogeneity in risks. If the covariate were removed from the analysis, as in Clayton and Kaldor (1987), the heterogeneity in county-specific risks would increase. (The fitted SMRs in Table 6 of Breslow and Clayton (1993) were obtained without the covariate x in the model.)

Output 44.3.4 Estimated Covariance Parameters in Poisson GLMM

Covariance Parameter Estimates

Cov Parm	Estimate	Standard Error
county	0.3567	0.09869

The "Solutions for Fixed Effects" table displays the estimates of β_0 and β_1 along with their standard errors and test statistics (Output 44.3.5). Because of the DDFM=NONE option in the MODEL statement, PROC GLIMMIX assumes that the degrees of freedom for the t tests of $H_0: \beta_j = 0$ are infinite. The p-values correspond to probabilities under a standard normal distribution. The covariate measuring employment percentages in agriculture, fisheries, and forestry is significant. This covariate might be a surrogate for the exposure to sunlight, an important risk factor for lip cancer.

Output 44.3.5 Fixed-Effects Parameter Estimates in Poisson GLMM

Solutions for Fixed Effects

Effect	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept	-0.4406	0.1572	Infty	-2.80	0.0051
x	0.6799	0.1409	Infty	4.82	<.0001

You can examine the quality of the fit of this model with various residual plots. A panel of studentized residuals is requested with the following statements:

```
ods graphics on;
ods select StudentPanel;

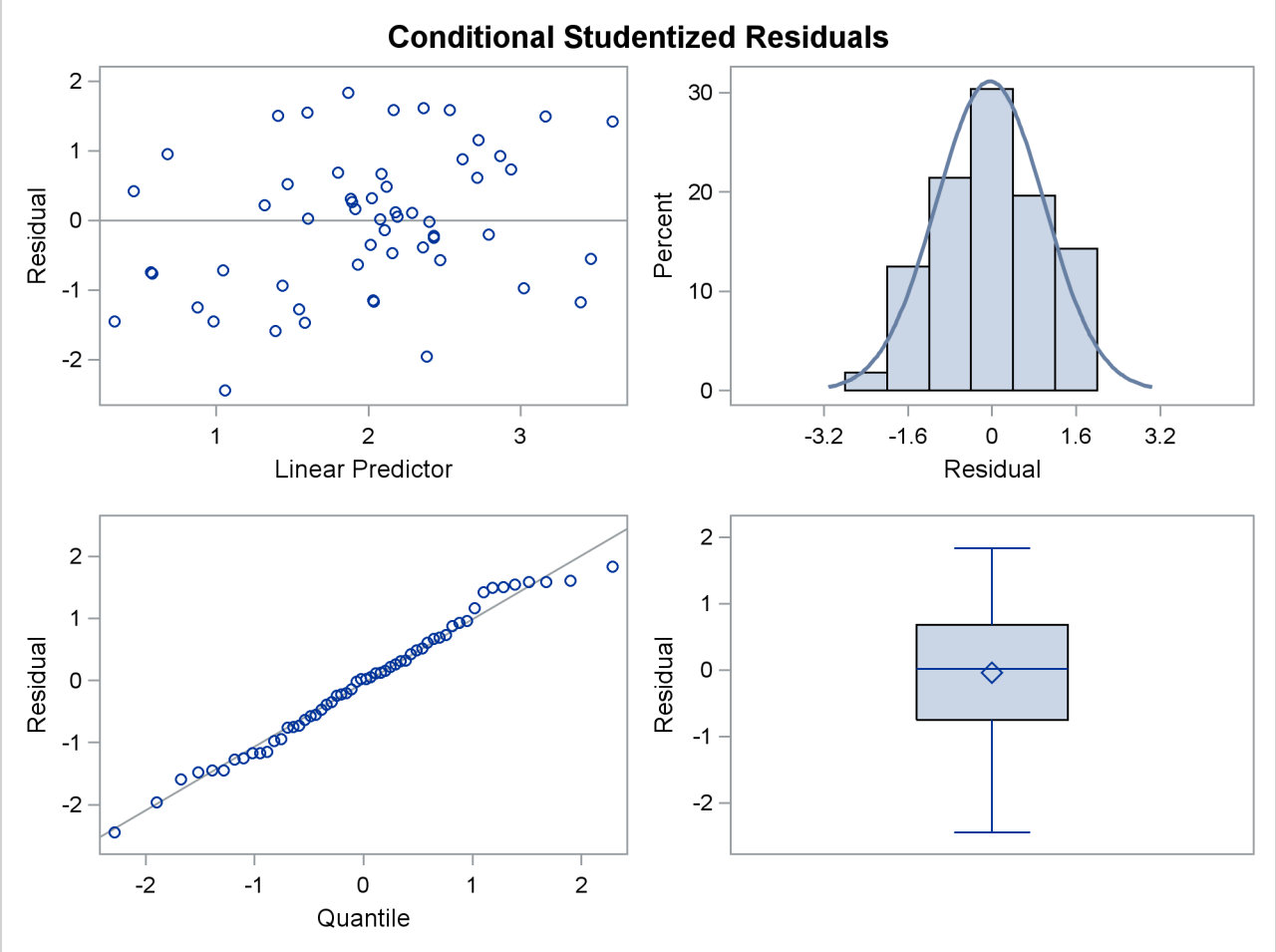
proc glimmix data=lipcancer plots=studentpanel;
   class county;
```

```
   x    = employment / 10;
   logn = log(expCount);
   model observed = x / dist=poisson offset=logn s ddfm=none;
   random county;
run;

ods graphics off;
```

The graph in the upper-left corner of the panel displays studentized residuals plotted against the linear predictor (Output 44.3.6). The default of the GLIMMIX procedure is to use the estimated BLUPs in the construction of the residuals and to present them on the linear scale, which in this case is the logarithmic scale. You can change the type of the computed residual with the TYPE= suboptions of each paneled display. For example, the option PLOTS=STUDENTPANEL(TYPE=NOBLUP) would request a paneled display of the marginal residuals on the linear scale.

Output 44.3.6 Panel of Studentized Residuals



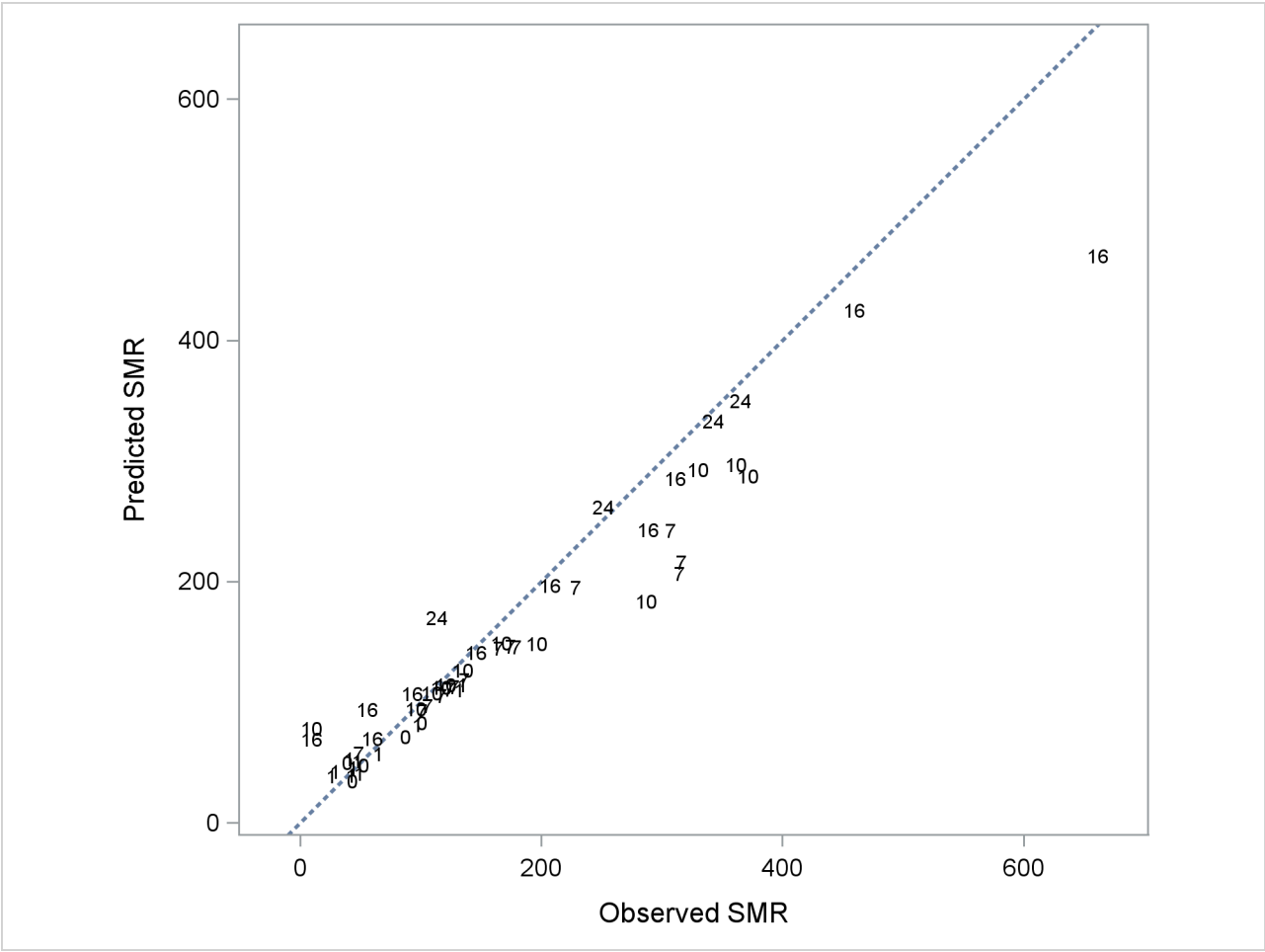
The graph in the upper-right corner of the panel shows a histogram with overlaid normal density. A Q-Q plot and a box plot are shown in the lower cells of the panel.

The following statements produce a graph of the observed and predicted standardized mortality ratios (Output 44.3.7):

```
proc template;
   define statgraph scatter;
      BeginGraph;
         layout overlayequated / yaxisopts=(label='Predicted SMR')
                                 xaxisopts=(label='Observed SMR')
                                 equatetype=square;
            lineparm y=0 slope=1 x=0 /
                   lineattrs = GraphFit(pattern=dash)
                   extend    = true;
            scatterplot y=SMR_pred x=SMR /
                    markercharacter = employment;
         endlayout;
      EndGraph;
   end;
run;
proc sgrender data=glimmixout template=scatter;
run;
```

In Output 44.3.7, fitted SMRs tend to be larger than the observed SMRs for counties with small observed SMR and smaller than the observed SMRs for counties with high observed SMR.

Output 44.3.7 Observed and Predicted SMRs; Data Labels Indicate Covariate Values



To demonstrate the impact of the random effects adjustment to the log-relative risks, the following statements fit a Poisson regression model (a GLM) by maximum likelihood:

```
proc glimmix data=lipcancer;
   x    = employment / 10;
   logn = log(expCount);
   model observed = x / dist=poisson offset=logn
                        solution ddfm=none;
   SMR_pred = 100*exp(_zgamma_ + _xbeta_);
   id employment SMR SMR_pred;
   output out=glimmixout;
run;
```

The GLIMMIX procedure defaults to maximum likelihood estimation because these statements fit a generalized linear model with nonnormal distribution. As a consequence, the SMRs are county specific only to the extent that the risks vary with the value of the covariate. But risks are no longer adjusted based on county-to-county heterogeneity in the observed incidence count.

Because of the absence of random effects, the GLIMMIX procedure recognizes the model as a generalized linear model and fits it by maximum likelihood (Output 44.3.8). The variance matrix is diagonal because the observations are uncorrelated.

Output 44.3.8 Model Information in Poisson GLM

The GLIMMIX Procedure

Model Information	
Data Set	WORK.LIPCANCER
Response Variable	observed
Response Distribution	Poisson
Link Function	Log
Variance Function	Default
Offset Variable	logn = log(expCount);
Variance Matrix	Diagonal
Estimation Technique	Maximum Likelihood
Degrees of Freedom Method	None

The "Dimensions" table shows that there are no G-side random effects in this model and no R-side scale parameter either (Output 44.3.9).

Output 44.3.9 Model Dimensions Information in Poisson GLM

Dimensions	
Columns in X	2
Columns in Z	0
Subjects (Blocks in V)	1
Max Obs per Subject	56

Because this is a GLM, the GLIMMIX procedure defaults to the Newton-Raphson algorithm, and the fixed effects (intercept and slope) comprise the parameters in the optimization (Output 44.3.10). (The default optimization technique for a GLM is the Newton-Raphson method.)

Output 44.3.10 Optimization Information in Poisson GLM

Optimization Information	
Optimization Technique	Newton-Raphson
Parameters in Optimization	2
Lower Boundaries	0
Upper Boundaries	0
Fixed Effects	Not Profiled

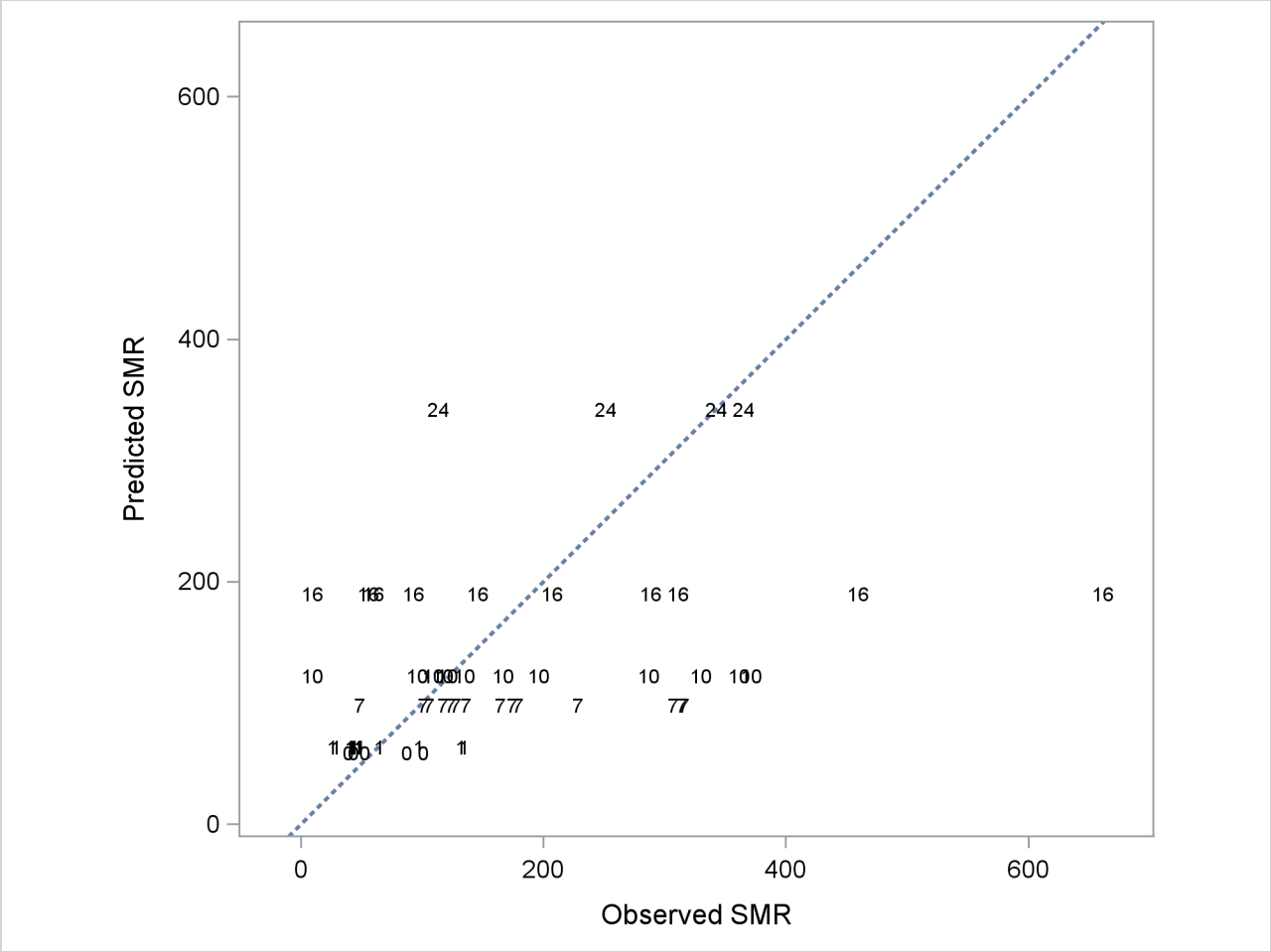
The estimates of β_0 and β_1 have changed from the previous analysis. In the GLMM, the estimates were $\widehat{\beta}_0 = -0.4406$ and $\widehat{\beta}_1 = 0.6799$ (Output 44.3.11).

Output 44.3.11 Parameter Estimates in Poisson GLM

Parameter Estimates					
Effect	**Estimate**	**Standard Error**	**DF**	**t Value**	**Pr > |t|**
Intercept	-0.5419	0.06951	Infty	-7.80	<.0001
x	0.7374	0.05954	Infty	12.38	<.0001

More importantly, without the county-specific adjustments through the best linear unbiased predictors of the random effects, the predicted SMRs are the same for all counties with the same percentage of employees in agriculture, fisheries, and forestry (Output 44.3.12).

Output 44.3.12 Observed and Predicted SMRs in Poisson GLM



Example 44.4: Quasi-likelihood Estimation for Proportions with Unknown Distribution

Wedderburn (1974) analyzes data on the incidence of leaf blotch (*Rhynchosporium secalis*) on barley.

The data represent the percentage of leaf area affected in a two-way layout with 10 barley varieties at nine sites. The following DATA step converts these data to proportions, as analyzed in McCullagh and Nelder (1989, Ch. 9.2.4). The purpose of the analysis is to make comparisons among the varieties, adjusted for site effects.

```
data blotch;
   array p{9} pct1-pct9;
   input variety pct1-pct9;
   do site = 1 to 9;
      prop = p{site}/100;
      output;
   end;
   drop pct1-pct9;
   datalines;
1  0.05  0.00  1.25  2.50  5.50  1.00  5.00  5.00 17.50
2  0.00  0.05  1.25  0.50  1.00  5.00  0.10 10.00 25.00
3  0.00  0.05  2.50  0.01  6.00  5.00  5.00  5.00 42.50
4  0.10  0.30 16.60  3.00  1.10  5.00  5.00  5.00 50.00
5  0.25  0.75  2.50  2.50  2.50  5.00 50.00 25.00 37.50
6  0.05  0.30  2.50  0.01  8.00  5.00 10.00 75.00 95.00
7  0.50  3.00  0.00 25.00 16.50 10.00 50.00 50.00 62.50
8  1.30  7.50 20.00 55.00 29.50  5.00 25.00 75.00 95.00
9  1.50  1.00 37.50  5.00 20.00 50.00 50.00 75.00 95.00
10 1.50 12.70 26.25 40.00 43.50 75.00 75.00 75.00 95.00
;
```

Little is known about the distribution of the leaf area proportions. The outcomes are not binomial proportions, because they do not represent the ratio of a count over a total number of Bernoulli trials. However, because the mean proportion μ_{ij} for variety j on site i must lie in the interval $[0, 1]$, you can commence the analysis with a model that treats Prop as a "pseudo-binomial" variable:

$$
\begin{aligned}
\mathrm{E}[\mathsf{Prop}_{ij}] &= \mu_{ij} \\
\mu_{ij} &= 1/(1+\exp\{-\eta_{ij}\}) \\
\eta_{ij} &= \beta_0 + \alpha_i + \tau_j \\
\mathrm{Var}[\mathsf{Prop}_{ij}] &= \phi\mu_{ij}(1-\mu_{ij})
\end{aligned}
$$

Here, η_{ij} is the linear predictor for variety j on site i, α_i denotes the ith site effect, and τ_j denotes the jth barley variety effect. The logit of the expected leaf area proportions is linearly related to these effects. The variance function of the model is that of a binomial(n,μ_{ij}) variable, and ϕ is an overdispersion parameter. The moniker "pseudo-binomial" derives not from the pseudo-likelihood methods used to estimate the parameters in the model, but from treating the response variable as if it had first and second moment properties akin to a binomial random variable.

The model is fit in the GLIMMIX procedure with the following statements:

```
proc glimmix data=blotch;
   class site variety;
   model prop = site variety / link=logit dist=binomial;
   random _residual_;
   lsmeans variety / diff=control('1');
run;
```

The MODEL statement specifies the distribution as binomial and the logit link. Because the variance function of the binomial distribution is $a(\mu) = \mu(1-\mu)$, you use the statement

```
random _residual_;
```

to specify the scale parameter ϕ. The LSMEANS statement requests estimates of the least squares means for the barley variety. The DIFF=CONTROL('1') option requests tests of least squares means differences against the first variety.

The "Model Information" table in Output 44.4.1 describes the model and methods used in fitting the statistical model. It is assumed here that the data are binomial proportions.

Output 44.4.1 Model Information in Pseudo-binomial Analysis

The GLIMMIX Procedure

Model Information	
Data Set	WORK.BLOTCH
Response Variable	prop
Response Distribution	Binomial
Link Function	Logit
Variance Function	Default
Variance Matrix	Diagonal
Estimation Technique	Maximum Likelihood
Degrees of Freedom Method	Residual

The "Class Level Information" table in Output 44.4.2 lists the number of levels of the Site and Variety effects and their values. All 90 observations read from the data are used in the analysis.

Output 44.4.2 Class Levels and Number of Observations

Class Level Information		
Class	**Levels**	**Values**
site	9	1 2 3 4 5 6 7 8 9
variety	10	1 2 3 4 5 6 7 8 9 10

Number of Observations Read	90
Number of Observations Used	90

In Output 44.4.3, the "Dimensions" table shows that the model does not contain G-side random effects. There is a single covariance parameter, which corresponds to ϕ. The "Optimization Information" table shows that the optimization comprises 18 parameters (Output 44.4.3). These correspond to the 18 nonsingular columns of the $\mathbf{X}'\mathbf{X}$ matrix.

Output 44.4.3 Model Fit in Pseudo-binomial Analysis

Dimensions	
Covariance Parameters	1
Columns in X	20
Columns in Z	0
Subjects (Blocks in V)	1
Max Obs per Subject	90

Optimization Information	
Optimization Technique	Newton-Raphson
Parameters in Optimization	18
Lower Boundaries	0
Upper Boundaries	0
Fixed Effects	Not Profiled

Fit Statistics	
-2 Log Likelihood	57.15
AIC (smaller is better)	93.15
AICC (smaller is better)	102.79
BIC (smaller is better)	138.15
CAIC (smaller is better)	156.15
HQIC (smaller is better)	111.30
Pearson Chi-Square	6.39
Pearson Chi-Square / DF	0.09

There are significant site and variety effects in this model based on the approximate Type III F tests (Output 44.4.4).

Output 44.4.4 Tests of Site and Variety Effects in Pseudo-binomial Analysis

Type III Tests of Fixed Effects				
Effect	**Num DF**	**Den DF**	**F Value**	**Pr > F**
site	8	72	18.25	<.0001
variety	9	72	13.85	<.0001

Output 44.4.5 displays the Variety least squares means for this analysis. These are obtained by averaging

$$\text{logit}(\widehat{\mu}_{ij}) = \widehat{\eta}_{ij}$$

across the sites. In other words, LS-means are computed on the linked scale where the model effects are additive. Note that the least squares means are ordered by variety. The estimate of the expected proportion of infected leaf area for the first variety is

$$\widehat{\mu}_{.,1} = \frac{1}{1 + \exp\{4.38\}} = 0.0124$$

and that for the last variety is

$$\widehat{\mu}_{.,10} = \frac{1}{1 + \exp\{0.127\}} = 0.468$$

Output 44.4.5 Variety Least Squares Means in Pseudo-binomial Analysis

variety Least Squares Means

variety	Estimate	Standard Error	DF	t Value	Pr > |t|
1	-4.3800	0.5643	72	-7.76	<.0001
2	-4.2300	0.5383	72	-7.86	<.0001
3	-3.6906	0.4623	72	-7.98	<.0001
4	-3.3319	0.4239	72	-7.86	<.0001
5	-2.7653	0.3768	72	-7.34	<.0001
6	-2.0089	0.3320	72	-6.05	<.0001
7	-1.8095	0.3228	72	-5.61	<.0001
8	-1.0380	0.2960	72	-3.51	0.0008
9	-0.8800	0.2921	72	-3.01	0.0036
10	-0.1270	0.2808	72	-0.45	0.6523

Because of the ordering of the least squares means, the differences against the first variety are also ordered from smallest to largest (Output 44.4.6).

Output 44.4.6 Variety Differences against the First Variety

Differences of variety Least Squares Means

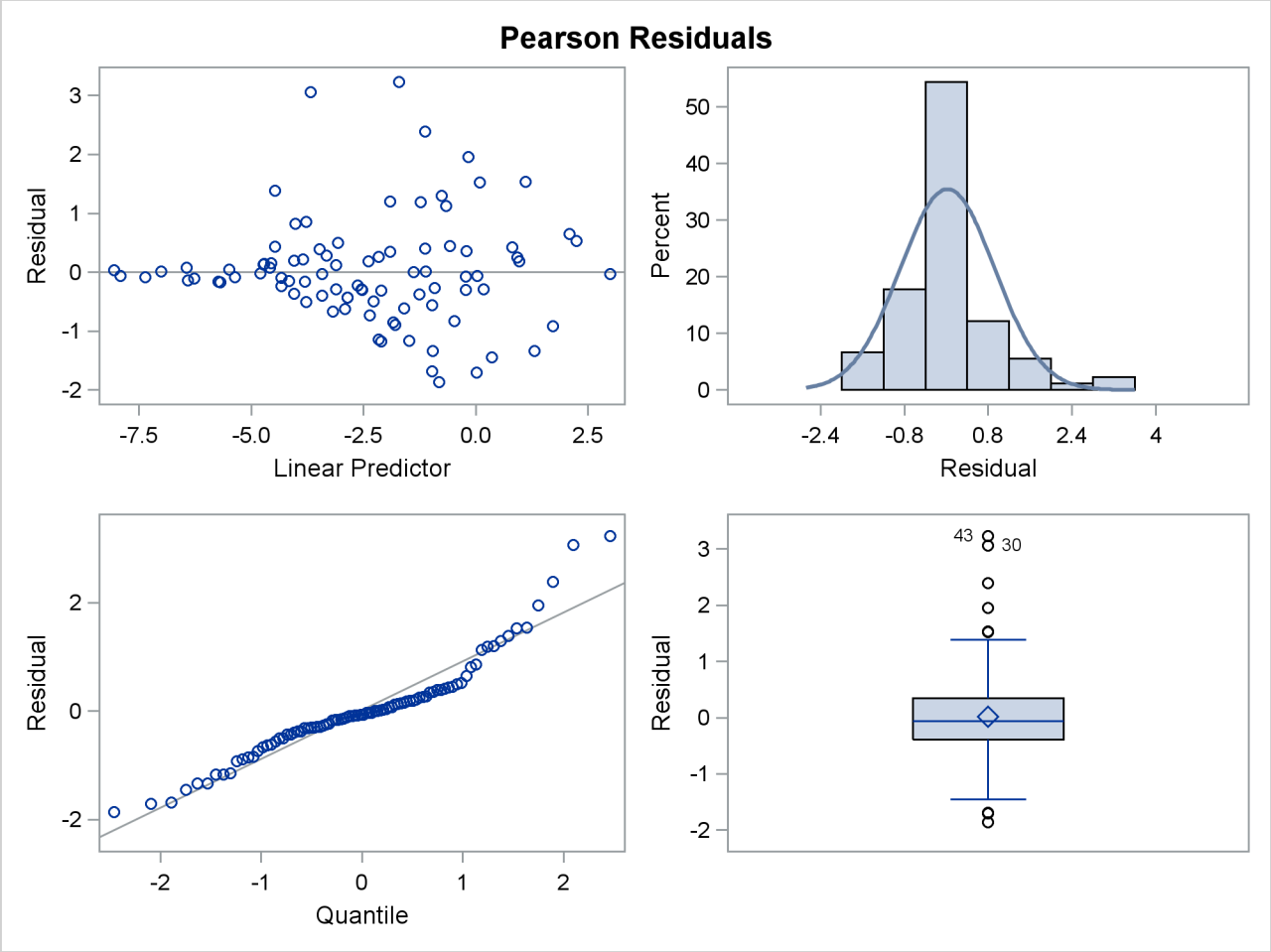
variety	_variety	Estimate	Standard Error	DF	t Value	Pr > |t|
2	1	0.1501	0.7237	72	0.21	0.8363
3	1	0.6895	0.6724	72	1.03	0.3086
4	1	1.0482	0.6494	72	1.61	0.1109
5	1	1.6147	0.6257	72	2.58	0.0119
6	1	2.3712	0.6090	72	3.89	0.0002
7	1	2.5705	0.6065	72	4.24	<.0001
8	1	3.3420	0.6015	72	5.56	<.0001
9	1	3.5000	0.6013	72	5.82	<.0001
10	1	4.2530	0.6042	72	7.04	<.0001

This analysis depends on your choice for the variance function that was implied by the binomial distribution. You can diagnose the distributional assumption by examining various graphical diagnostics measures. The following statements request a panel display of the Pearson-type residuals:

```
ods graphics on;
ods select PearsonPanel;
proc glimmix data=blotch plots=pearsonpanel;
   class site variety;
   model prop = site variety / link=logit dist=binomial;
   random _residual_;
run;
ods graphics off;
```

Output 44.4.7 clearly indicates that the chosen variance function is not appropriate for these data. As μ approaches zero or one, the variability in the residuals is less than that implied by the binomial variance function.

Output 44.4.7 Panel of Pearson-Type Residuals in Pseudo-binomial Analysis



To remedy this situation, McCullagh and Nelder (1989) consider instead the variance function

$$\text{Var}[\text{Prop}_{ij}] = \mu_{ij}^2(1-\mu_{ij})^2$$

Imagine two varieties with $\mu_{.i} = 0.1$ and $\mu_{.k} = 0.5$. Under the binomial variance function, the variance of the proportion for variety k is 2.77 times larger than that for variety i. Under the revised model this ratio increases to $2.77^2 = 7.67$.

The analysis of the revised model is obtained with the next set of GLIMMIX statements. Because you need to model a variance function that does not correspond to any of the built-in distributions, you need to supply a function with an assignment to the automatic variable _VARIANCE_. The GLIMMIX procedure then considers the distribution of the data as unknown. The corresponding estimation technique is quasi-likelihood. Because this model does not include an extra scale parameter, you can drop the RANDOM _RESIDUAL_ statement from the analysis.

```
ods graphics on;
ods select ModelInfo FitStatistics LSMeans Diffs PearsonPanel;
proc glimmix data=blotch plots=pearsonpanel;
   class site variety;
   _variance_ = _mu_**2 * (1-_mu_)**2;
   model prop = site variety / link=logit;
   lsmeans variety / diff=control('1');
run;
ods graphics off;
```

The "Model Information" table in Output 44.4.8 now displays the distribution as "Unknown," because of the assignment made in the GLIMMIX statements to _VARIANCE_. The table also shows the expression evaluated as the variance function.

Output 44.4.8 Model Information in Quasi-likelihood Analysis

The GLIMMIX Procedure

Model Information	
Data Set	WORK.BLOTCH
Response Variable	prop
Response Distribution	Unknown
Link Function	Logit
Variance Function	_mu_**2 * (1-_mu_)**2
Variance Matrix	Diagonal
Estimation Technique	Quasi-Likelihood
Degrees of Freedom Method	Residual

The fit statistics of the model are now expressed in terms of the log quasi-likelihood. It is computed as

$$\sum_{i=1}^{9}\sum_{j=1}^{10}\int_{y_{ij}}^{\mu_{ij}} \frac{y_{ij}-t}{t^2(1-t)^2}\,dt$$

Twice the negative of this sum equals –85.74, which is displayed in the "Fit Statistics" table (Output 44.4.9).

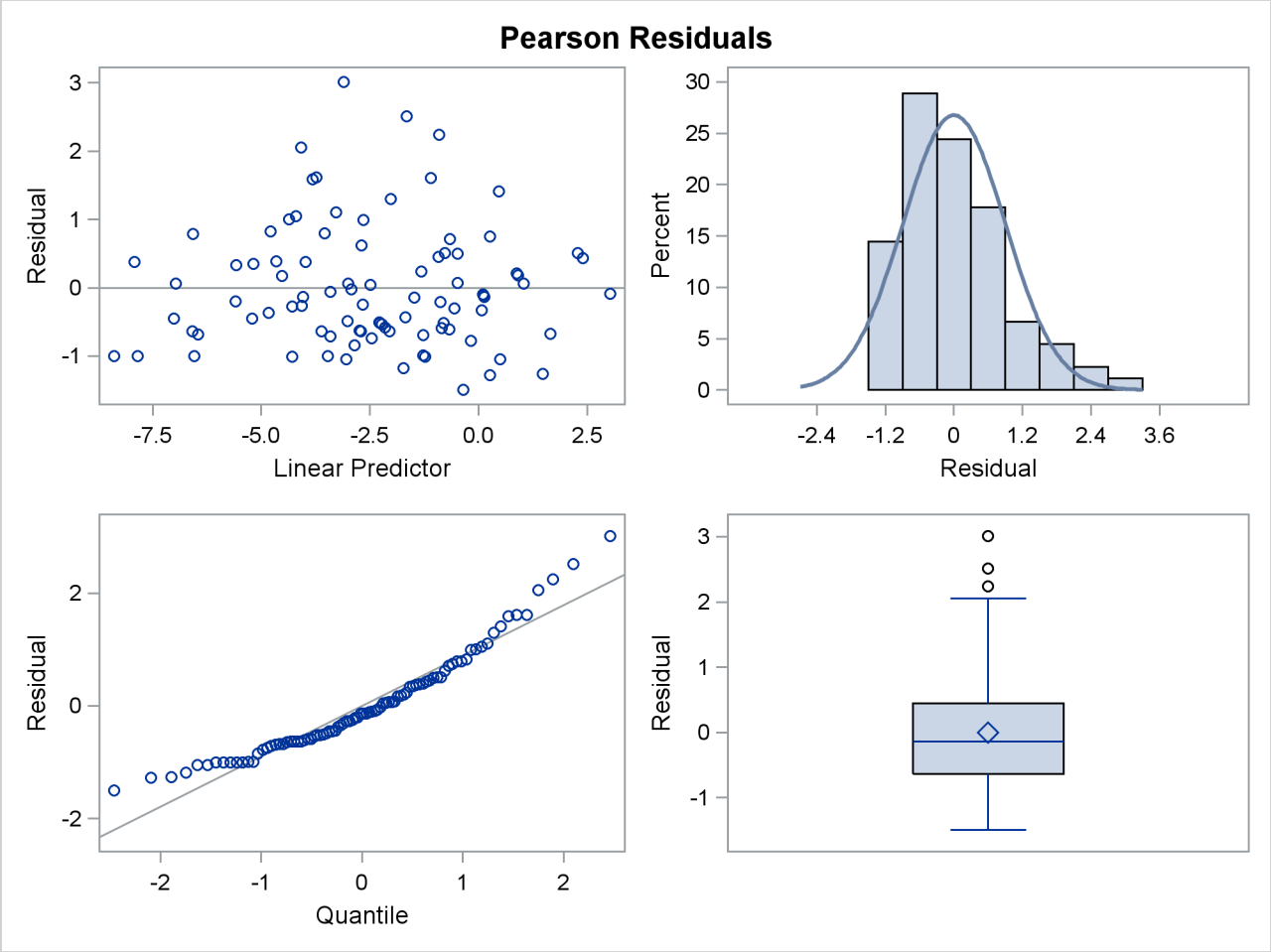
The scaled Pearson statistic is now 0.99. Inclusion of an extra scale parameter ϕ would have little or no effect on the results.

Output 44.4.9 Fit Statistics in Quasi-likelihood Analysis

Fit Statistics	
-2 Log Quasi-Likelihood	-85.74
Quasi-AIC (smaller is better)	-49.74
Quasi-AICC (smaller is better)	-40.11
Quasi-BIC (smaller is better)	-4.75
Quasi-CAIC (smaller is better)	13.25
Quasi-HQIC (smaller is better)	-31.60
Pearson Chi-Square	71.17
Pearson Chi-Square / DF	0.99

The panel of Pearson-type residuals now shows a much more adequate distribution for the residuals and a reduction in the number of outlying residuals (Output 44.4.10).

Output 44.4.10 Panel of Pearson-Type Residuals (Quasi-likelihood)



The least squares means are no longer ordered in size by variety (Output 44.4.11). For example, $\text{logit}(\widehat{\mu}_{.1}) > \text{logit}(\widehat{\mu}_{.2})$. Under the revised model, the second variety has a greater percentage of its leaf area covered by blotch, compared to the first variety. Varieties 5 and 6 and varieties 8 and 9 show similar reversal in ranking.

Output 44.4.11 Variety Least Squares Means in Quasi-likelihood Analysis

variety Least Squares Means

variety	Estimate	Standard Error	DF	t Value	Pr > |t|
1	-4.0453	0.3333	72	-12.14	<.0001
2	-4.5126	0.3333	72	-13.54	<.0001
3	-3.9664	0.3333	72	-11.90	<.0001
4	-3.0912	0.3333	72	-9.27	<.0001
5	-2.6927	0.3333	72	-8.08	<.0001
6	-2.7167	0.3333	72	-8.15	<.0001
7	-1.7052	0.3333	72	-5.12	<.0001
8	-0.7827	0.3333	72	-2.35	0.0216
9	-0.9098	0.3333	72	-2.73	0.0080
10	-0.1580	0.3333	72	-0.47	0.6369

Interestingly, the standard errors are constant among the LS-means (Output 44.4.11) and among the LS-means differences (Output 44.4.12). This is due to the fact that for the logit link

$$\frac{\partial \mu}{\partial \eta} = \mu(1-\mu)$$

which cancels with the square root of the variance function in the estimating equations. The analysis is thus orthogonal.

Output 44.4.12 Variety Differences in Quasi-likelihood Analysis

Differences of variety Least Squares Means

variety	_variety	Estimate	Standard Error	DF	t Value	Pr > |t|
2	1	-0.4673	0.4714	72	-0.99	0.3249
3	1	0.07885	0.4714	72	0.17	0.8676
4	1	0.9541	0.4714	72	2.02	0.0467
5	1	1.3526	0.4714	72	2.87	0.0054
6	1	1.3286	0.4714	72	2.82	0.0062
7	1	2.3401	0.4714	72	4.96	<.0001
8	1	3.2626	0.4714	72	6.92	<.0001
9	1	3.1355	0.4714	72	6.65	<.0001
10	1	3.8873	0.4714	72	8.25	<.0001

Example 44.5: Joint Modeling of Binary and Count Data

Clustered data arise when multiple observations are collected on the same sampling or experimental unit. Often, these multiple observations refer to the same attribute measured at different points in time or space. This leads to repeated measures, longitudinal, and spatial data, which are special forms of multivariate data. A different class of multivariate data arises when the multiple observations refer to different attributes.

The data set hernio, created in the following DATA step, provides an example of a bivariate outcome variable. It reflects the condition and length of hospital stay for 32 herniorrhaphy patients. These data are based on data given by Mosteller and Tukey (1977) and reproduced in Hand et al. (1994, pp. 390, 391). The data set

that follows does not contain all the covariates given in these sources. The response variables are leave and los; these denote the condition of the patient upon leaving the operating room and the length of hospital stay after the operation (in days). The variable leave takes on the value one if a patient experiences a routine recovery, and the value zero if postoperative intensive care was required. The binary variable OKstatus distinguishes patients based on their postoperative physical status ("1" implies better status).

```
data hernio;
   input patient age gender$ OKstatus leave los;
   datalines;
 1   78  m   1   0   9
 2   60  m   1   0   4
 3   68  m   1   1   7
 4   62  m   0   1  35
 5   76  m   0   0   9
 6   76  m   1   1   7
 7   64  m   1   1   5
 8   74  f   1   1  16
 9   68  m   0   1   7
10   79  f   1   0  11
11   80  f   0   1   4
12   48  m   1   1   9
13   35  f   1   1   2
14   58  m   1   1   4
15   40  m   1   1   3
16   19  m   1   1   4
17   79  m   0   0   3
18   51  m   1   1   5
19   57  m   1   1   8
20   51  m   0   1   8
21   48  m   1   1   3
22   48  m   1   1   5
23   66  m   1   1   8
24   71  m   1   0   2
25   75  f   0   0   7
26    2  f   1   1   0
27   65  f   1   0  16
28   42  f   1   0   3
29   54  m   1   0   2
30   43  m   1   1   3
31    4  m   1   1   3
32   52  m   1   1   8
;
```

While the response variable los is a Poisson count variable, the response variable leave is a binary variable. You can perform separate analysis for the two outcomes, for example, by fitting a logistic model for the operating room exit condition and a Poisson regression model for the length of hospital stay. This, however, would ignore the correlation between the two outcomes. Intuitively, you would expect that the length of postoperative hospital stay is longer for those patients who had more tenuous exit conditions.

The following DATA step converts the data set hernio from the multivariate form to the univariate form. In the multivariate form the responses are stored in separate variables. The GLIMMIX procedure requires the univariate data structure.

```
data hernio_uv;
   length dist $7;
   set hernio;
   response = (leave=1);
   dist     = "Binary";
   output;
   response = los;
   dist     = "Poisson";
   output;
   keep patient age OKstatus response dist;
run;
```

This DATA step expands the 32 observations in the data set hernio into 64 observations, stacking two observations per patient. The character variable dist identifies the distribution that is assumed for the respective observations within a patient. The first observation for each patient corresponds to the binary response.

The following GLIMMIX statements fit a logistic regression model with two regressors (age and OKStatus) to the binary observations:

```
proc glimmix data=hernio_uv(where=(dist="Binary"));
   model response(event='1') = age OKStatus / s dist=binary;
run;
```

The EVENT=('1') response option requests that PROC GLIMMIX model the probability $\Pr(\text{leave} = 1)$—that is, the probability of routine recovery. The fit statistics and parameter estimates for this univariate analysis are shown in Output 44.5.1. The coefficient for the age effect is negative (–0.07725) and marginally significant at the 5% level ($p = 0.0491$). The negative sign indicates that the probability of routine recovery decreases with age. The coefficient for the OKStatus variable is also negative. Its large standard error and the p-value of 0.7341 indicate, however, that this regressor is not significant.

Output 44.5.1 Univariate Logistic Regression

The GLIMMIX Procedure

Fit Statistics	
-2 Log Likelihood	32.77
AIC (smaller is better)	38.77
AICC (smaller is better)	39.63
BIC (smaller is better)	43.17
CAIC (smaller is better)	46.17
HQIC (smaller is better)	40.23
Pearson Chi-Square	30.37
Pearson Chi-Square / DF	1.05

Parameter Estimates

Effect	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept	5.7694	2.8245	29	2.04	0.0503
age	-0.07725	0.03761	29	-2.05	0.0491
OKstatus	-0.3516	1.0253	29	-0.34	0.7341

Based on the univariate logistic regression analysis, you would probably want to revisit the model, examine other regressor variables, test for gender effects and interactions, and so forth. The two-regressor model is sufficient for this example. It is illustrative to trace the relative importance of the two regressors through various types of models.

The next statements fit the same regressors to the count data:

```
proc glimmix data=hernio_uv(where=(dist="Poisson"));
   model response = age OKStatus / s dist=Poisson;
run;
```

For this response, both regressors appear to make significant contributions at the 5% significance level (Output 44.5.2). The sign of the coefficient seems appropriate; the length of hospital stay should increase with patient age and be shorter for patients with better preoperative health. The magnitude of the scaled Pearson statistic (4.48) indicates, however, that there is considerable overdispersion in this model. This could be due to omitted variables or an improper distributional assumption. The importance of preoperative health status, for example, can change with a patient's age, which could call for an interaction term.

Output 44.5.2 Univariate Poisson Regression

The GLIMMIX Procedure

Fit Statistics	
-2 Log Likelihood	215.52
AIC (smaller is better)	221.52
AICC (smaller is better)	222.38
BIC (smaller is better)	225.92
CAIC (smaller is better)	228.92
HQIC (smaller is better)	222.98
Pearson Chi-Square	129.98
Pearson Chi-Square / DF	4.48

Parameter Estimates

Effect	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept	1.2640	0.3393	29	3.72	0.0008
age	0.01525	0.004454	29	3.42	0.0019
OKstatus	-0.3301	0.1562	29	-2.11	0.0433

You can also model both responses jointly. The following statements request a multivariate analysis:

```
proc glimmix data=hernio_uv;
   class dist;
   model response(event='1') = dist dist*age dist*OKstatus /
                    noint s dist=byobs(dist);
run;
```

The DIST=BYOBS option in the MODEL statement instructs the GLIMMIX procedure to examine the variable dist in order to identify the distribution of an observation. The variable can be character or numeric. See the DIST= option of the MODEL statement for a list of the numeric codes for the various distributions that are compatible with the DIST=BYOBS formulation. Because no LINK= option is specified, the link functions are chosen as the default links that correspond to the respective distributions. In this case, the logit link is applied to the binary observations and the log link is applied to the Poisson outcomes. The dist variable

is also listed in the CLASS statement, which enables you to use interaction terms in the MODEL statement to vary the regression coefficients by response distribution. The NOINT option is used here so that the parameter estimates of the joint model are directly comparable to those in Output 44.5.1 and Output 44.5.2.

The "Fit Statistics" and "Parameter Estimates" tables of this bivariate estimation process are shown in Output 44.5.3.

Output 44.5.3 Bivariate Analysis – Independence

The GLIMMIX Procedure

Fit Statistics

Description	Binary	Poisson	Total
-2 Log Likelihood	32.77	215.52	248.29
AIC (smaller is better)	44.77	227.52	260.29
AICC (smaller is better)	48.13	230.88	261.77
BIC (smaller is better)	53.56	236.32	273.25
CAIC (smaller is better)	59.56	242.32	279.25
HQIC (smaller is better)	47.68	230.44	265.40
Pearson Chi-Square	30.37	129.98	160.35
Pearson Chi-Square / DF	1.05	4.48	2.76

Parameter Estimates

Effect	dist	Estimate	Standard Error	DF	t Value	Pr > |t|
dist	Binary	5.7694	2.8245	58	2.04	0.0456
dist	Poisson	1.2640	0.3393	58	3.72	0.0004
age*dist	Binary	-0.07725	0.03761	58	-2.05	0.0445
age*dist	Poisson	0.01525	0.004454	58	3.42	0.0011
OKstatus*dist	Binary	-0.3516	1.0253	58	-0.34	0.7329
OKstatus*dist	Poisson	-0.3301	0.1562	58	-2.11	0.0389

The "Fit Statistics" table now contains a separate column for each response distribution, as well as an overall contribution. Because the model does not specify any random effects or R-side correlations, the log likelihoods are additive. The parameter estimates and their standard errors in this joint model are identical to those in Output 44.5.1 and Output 44.5.2. The *p*-values reflect the larger "sample size" in the joint analysis. Note that the coefficients would be different from the separate analyses if the dist variable had not been used to form interactions with the model effects.

There are two ways in which the correlations between the two responses for the same patient can be incorporated. You can induce them through shared random effects or model the dependency directly. The following statements fit a model that induces correlation:

```
proc glimmix data=hernio_uv;
   class patient dist;
   model response(event='1') = dist dist*age dist*OKstatus /
                    noint s dist=byobs(dist);
   random int / subject=patient;
run;
```

Notice that the patient variable has been added to the CLASS statement and as the SUBJECT= effect in the RANDOM statement.

The "Fit Statistics" table in Output 44.5.4 no longer has separate columns for each response distribution, because the data are not independent. The log (pseudo-)likelihood does not factor into additive component that correspond to distributions. Instead, it factors into components associated with subjects.

Output 44.5.4 Bivariate Analysis – Mixed Model

The GLIMMIX Procedure

Fit Statistics	
-2 Res Log Pseudo-Likelihood	226.71
Generalized Chi-Square	52.25
Gener. Chi-Square / DF	0.90

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error
Intercept	patient	0.2990	0.1116

Solutions for Fixed Effects

Effect	dist	Estimate	Standard Error	DF	t Value	Pr > |t|
dist	Binary	5.7783	2.9048	29	1.99	0.0562
dist	Poisson	0.8410	0.5696	29	1.48	0.1506
age*dist	Binary	-0.07572	0.03791	29	-2.00	0.0552
age*dist	Poisson	0.01875	0.007383	29	2.54	0.0167
OKstatus*dist	Binary	-0.4697	1.1251	29	-0.42	0.6794
OKstatus*dist	Poisson	-0.1856	0.3020	29	-0.61	0.5435

The estimate of the variance of the random patient intercept is 0.2990, and the estimated standard error of this variance component estimate is 0.1116. There appears to be significant patient-to-patient variation in the intercepts. The estimates of the fixed effects as well as their estimated standard errors have changed from the bivariate-independence analysis (see Output 44.5.3). When the length of hospital stay and the postoperative condition are modeled jointly, the preoperative health status (variable OKStatus) no longer appears significant. Compare this result to Output 44.5.3; in the separate analyses the initial health status was a significant predictor of the length of hospital stay. A further joint analysis of these data would probably remove this predictor from the model entirely.

A joint model of the second kind, where correlations are modeled directly, is fit with the following GLIMMIX statements:

```
proc glimmix data=hernio_uv;
   class patient dist;
   model response(event='1') = dist dist*age dist*OKstatus /
                    noint s dist=byobs(dist);
   random _residual_ / subject=patient type=chol;
run;
```

Instead of a shared G-side random effect, an R-side covariance structure is used to model the correlations. It is important to note that this is a marginal model that models covariation on the scale of the data. The previous model involves the $\mathbf{Z}\boldsymbol{\gamma}$ random components inside the linear predictor.

The _RESIDUAL_ keyword instructs PROC GLIMMIX to model the R-side correlations. Because of the SUBJECT=PATIENT option, data from different patients are independent, and data from a single patient

follow the covariance model specified with the TYPE= option. In this case, a generally unstructured 2×2 covariance matrix is modeled, but in its Cholesky parameterization. This ensures that the resulting covariance matrix is at least positive semidefinite and stabilizes the numerical optimizations.

Output 44.5.5 Bivariate Analysis – Marginal Correlated Error Model

The GLIMMIX Procedure

Fit Statistics	
-2 Res Log Pseudo-Likelihood	240.98
Generalized Chi-Square	58.00
Gener. Chi-Square / DF	1.00

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error
CHOL(1,1)	patient	1.0162	0.1334
CHOL(2,1)	patient	0.3942	0.3893
CHOL(2,2)	patient	2.0819	0.2734

Solutions for Fixed Effects

Effect	dist	Estimate	Standard Error	DF	t Value	Pr > |t|
dist	Binary	5.6514	2.8283	26	2.00	0.0563
dist	Poisson	1.2463	0.7189	26	1.73	0.0948
age*dist	Binary	-0.07568	0.03765	26	-2.01	0.0549
age*dist	Poisson	0.01548	0.009432	26	1.64	0.1128
OKstatus*dist	Binary	-0.3421	1.0384	26	-0.33	0.7445
OKstatus*dist	Poisson	-0.3253	0.3310	26	-0.98	0.3349

The "Covariance Parameter Estimates" table in Output 44.5.5 contains three entries for this model, corresponding to a (2×2) covariance matrix for each patient. The Cholesky root of the $\mathbf{R}$ matrix is

$$\mathbf{L} = \begin{bmatrix} 1.0162 & 0 \\ 0.3942 & 2.0819 \end{bmatrix}$$

so that the covariance matrix can be obtained as

$$\mathbf{LL}' = \begin{bmatrix} 1.0162 & 0 \\ 0.3942 & 2.0819 \end{bmatrix} \begin{bmatrix} 1.0162 & 0.3942 \\ 0 & 2.0819 \end{bmatrix} = \begin{bmatrix} 1.0326 & 0.4005 \\ 0.4005 & 4.4897 \end{bmatrix}$$

This is not the covariance matrix of the data, however, because the variance functions need to be accounted for.

The p-values in the "Solutions for Fixed Effects" table indicate the same pattern of significance and nonsignificance as in the conditional model with random patient intercepts.

Example 44.6: Radial Smoothing of Repeated Measures Data

This example of a repeated measures study is taken from Diggle, Liang, and Zeger (1994, p. 100). The data consist of body weights of 27 cows, measured at 23 unequally spaced time points over a period of

approximately 22 months. Following Diggle, Liang, and Zeger (1994), one animal is removed from the analysis, one observation is removed according to their Figure 5.7, and the time is shifted to start at 0 and is measured in 10-day increments. The design is a 2×2 factorial, and the factors are the infection of an animal with *M. paratuberculosis* and whether the animal is receiving iron dosing.

The following DATA steps create the data and arrange them in univariate format.

```
data times;
   input time1-time23;
   datalines;
 122  150  166  179  219  247  276  296  324  354  380  445
 478  508  536  569  599  627  655  668  723  751  781
;

data cows;
   if _n_ = 1 then merge times;
   array t{23} time1   - time23;
   array w{23} weight1 - weight23;
   input cow iron infection weight1-weight23 @@;
   do i=1 to 23;
      weight = w{i};
      tpoint = (t{i}-t{1})/10;
      output;
   end;
   keep cow iron infection tpoint weight;
   datalines;
  1 0 0  4.7    4.905  5.011  5.075  5.136  5.165  5.298  5.323
         5.416  5.438  5.541  5.652  5.687  5.737  5.814  5.799
         5.784  5.844  5.886  5.914  5.979  5.927  5.94
  2 0 0  4.868  5.075  5.193  5.22   5.298  5.416  5.481  5.521
         5.617  5.635  5.687  5.768  5.799  5.872  5.886  5.872
         5.914  5.966  5.991  6.016  6.087  6.098  6.153
  3 0 0  4.868  5.011  5.136  5.193  5.273  5.323  5.416  5.46
         5.521  5.58   5.617  5.687  5.72   5.753  5.784  5.784
         5.784  5.814  5.829  5.872  5.927  5.9    5.991
  4 0 0  4.828  5.011  5.136  5.193  5.273  5.347  5.438  5.561
         5.541  5.598  5.67    .     5.737  5.844  5.858  5.872
         5.886  5.927  5.94   5.979  6.052  6.028  6.12
  5 1 0  4.787  4.977  5.043  5.136  5.106  5.298  5.298  5.371
         5.438  5.501  5.561  5.652  5.67   5.737  5.784  5.768
         5.784  5.784  5.829  5.858  5.914  5.9    5.94
  6 1 0  4.745  4.868  5.043  5.106  5.22   5.298  5.347  5.347
         5.416  5.501  5.561  5.58   5.687  5.72   5.737  5.72
         5.737  5.753  5.768  5.784  5.844  5.844  5.9
  7 1 0  4.745  4.905  5.011  5.106  5.165  5.273  5.371  5.416
         5.416  5.521  5.541  5.635  5.687  5.704  5.784  5.768
         5.768  5.814  5.829  5.858  5.94   5.94   6.004
  8 0 1  4.942  5.106  5.136  5.193  5.298  5.347  5.46   5.521
         5.561  5.58   5.635  5.704  5.784  5.823  5.858  5.9
         5.94   5.991  6.016  6.064  6.052  6.016  5.979
  9 0 1  4.605  4.745  4.868  4.905  4.977  5.22   5.165  5.22
         5.22   5.247  5.298  5.416  5.501  5.521  5.58   5.58
         5.635  5.67   5.72   5.753  5.799  5.829  5.858
```

```
10 0 1  4.7   4.868  4.905  4.977  5.011  5.106  5.165  5.22
        5.22  5.22   5.273  5.384  5.438  5.438  5.501  5.501
        5.541 5.598  5.58   5.635  5.687  5.72   5.704
11 0 1  4.828 5.011  5.075  5.165  5.247  5.323  5.394  5.46
        5.46  5.501  5.541  5.609  5.687  5.704  5.72   5.704
        5.704 5.72   5.737  5.768  5.858  5.9    5.94
12 0 1  4.7   4.828  4.905  5.011  5.075  5.165  5.247  5.298
        5.298 5.323  5.416  5.505  5.561  5.58   5.561  5.635
        5.687 5.72   5.72   5.737  5.784  5.814  5.799
13 0 1  4.828 5.011  5.075  5.136  5.22   5.273  5.347  5.416
        5.438 5.416  5.521  5.628  5.67   5.687  5.72   5.72
        5.799 5.858  5.872  5.914  5.94   5.991  6.016
14 0 1  4.828 4.942  5.011  5.075  5.075  5.22   5.273  5.298
        5.323 5.298  5.394  5.489  5.541  5.58   5.617  5.67
        5.704 5.753  5.768  5.814  5.872  5.927  5.927
15 0 1  4.745 4.905  4.977  5.075  5.193  5.22   5.298  5.323
        5.394 5.394  5.438  5.583  5.617  5.652  5.687  5.72
        5.753 5.768  5.814  5.844  5.886  5.886  5.886
16 0 1  4.7   4.868  5.011  5.043  5.106  5.165  5.247  5.298
        5.347 5.371  5.438  5.455  5.617  5.635  5.704  5.737
        5.784 5.768  5.814  5.844  5.886  5.94   5.927
17 1 1  4.605 4.787  4.828  4.942  5.011  5.136  5.22   5.247
        5.273 5.247  5.347  5.366  5.416  5.46   5.541  5.481
        5.501 5.635  5.652  5.598  5.635  5.635  5.598
18 1 1  4.828 4.977  5.011  5.136  5.273  5.298  5.371  5.46
        5.416 5.416  5.438  5.557  5.617  5.67   5.72   5.72
        5.799 5.858  5.886  5.914  5.979  6.004  6.028
19 1 1  4.7   4.905  4.942  5.011  5.043  5.136  5.193  5.193
        5.247 5.22   5.323  5.338  5.371  5.394  5.438  5.416
        5.501 5.561  5.541  5.58   5.652  5.67   5.704
20 1 1  4.745 4.905  4.977  5.043  5.136  5.273  5.347  5.394
        5.416 5.394  5.521  5.617  5.617  5.617  5.67   5.635
        5.652 5.687  5.652  5.617  5.687  5.768  5.814
21 1 1  4.787 4.942  4.977  5.106  5.165  5.247  5.323  5.416
        5.394 5.371  5.438  5.521  5.521  5.561  5.635  5.617
        5.687 5.72   5.737  5.737  5.768  5.768  5.704
22 1 1  4.605 4.828  4.828  4.977  5.043  5.165  5.22   5.273
        5.247 5.22   5.298  5.375  5.371  5.416  5.501  5.501
        5.521 5.561  5.617  5.635  5.72   5.737  5.768
23 1 1  4.7   4.905  5.011  5.075  5.106  5.22   5.22   5.298
        5.323 5.347  5.416  5.472  5.501  5.541  5.598  5.598
        5.598 5.652  5.67   5.704  5.737  5.768  5.784
24 1 1  4.745 4.942  5.011  5.075  5.106  5.247  5.273  5.323
        5.347 5.371  5.416  5.481  5.501  5.541  5.598  5.598
        5.635 5.687  5.704  5.72   5.829  5.844  5.9
25 1 1  4.654 4.828  4.828  4.977  4.977  5.043  5.136  5.165
        5.165 5.165  5.193  5.204  5.22   5.273  5.371  5.347
        5.46  5.58   5.635  5.67   5.753  5.799  5.844
26 1 1  4.828 4.977  5.011  5.106  5.165  5.22   5.273  5.323
        5.371 5.394  5.46   5.576  5.652  5.617  5.687  5.67
        5.72  5.784  5.784  5.784  5.829  5.814  5.844
;
```

The mean response profiles of the cows are not of particular interest; what matters are inferences about the Iron effect, the Infection effect, and their interaction. Nevertheless, the body weight of the cows changes over the 22-month period, and you need to account for these changes in the analysis. A reasonable approach is to apply the approximate low-rank smoother to capture the trends over time. This approach frees you from having to stipulate a parametric model for the response trajectories over time. In addition, you can test hypotheses about the smoothing parameter; for example, whether it should be varied by treatment.

The following statements fit a model with a 2×2 factorial treatment structure and smooth trends over time, choosing the Newton-Raphson algorithm with ridging for the optimization:

```
proc glimmix data=cows;
   t2 = tpoint / 100;
   class cow iron infection;
   model weight = iron infection iron*infection tpoint;
   random t2 / type=rsmooth subject=cow
                   knotmethod=kdtree(bucket=100 knotinfo);
   output out=gmxout pred(blup)=pred;
   nloptions tech=newrap;
run;
```

The continuous time effect appears in both the MODEL statement (tpoint) and the RANDOM statement (t2). Because the variance of the radial smoothing component depends on the temporal metric, the time scale was rescaled for the RANDOM effect to move the parameter estimate away from the boundary. The knots of the radial smoother are selected as the vertices of a k-d tree. Specifying BUCKET=100 sets the bucket size of the tree to $b = 100$. Because measurements at each time point are available for 26 (or 25) cows, this groups approximately four time points in a single bucket. The KNOTINFO keyword of the KNOTMETHOD= option requests a printout of the knot locations for the radial smoother. The OUTPUT statement saves the predictions of the mean of each observations to the data set gmxout. Finally, the TECH=NEWRAP option in the NLOPTIONS statement specifies the Newton-Raphson algorithm for the optimization technique.

The "Class Level Information" table lists the number of levels of the Cow, Iron, and Infection effects (Output 44.6.1).

Output 44.6.1 Model Information and Class Levels in Repeated Measures Analysis

The GLIMMIX Procedure

Model Information	
Data Set	WORK.COWS
Response Variable	weight
Response Distribution	Gaussian
Link Function	Identity
Variance Function	Default
Variance Matrix Blocked By	cow
Estimation Technique	Restricted Maximum Likelihood
Degrees of Freedom Method	Containment

Class Level Information		
Class	**Levels**	**Values**
cow	26	1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
iron	2	0 1
infection	2	0 1

The "Radial Smoother Knots for RSmooth(t2)" table displays the knots computed from the vertices of the t2 k-d tree (Output 44.6.2). Notice that knots are spaced unequally and that the extreme time points are among the knot locations. The "Number of Observations" table shows that one observation was not used in the analysis. The 12th observation for cow 4 has a missing value.

Output 44.6.2 Knot Information and Number of Observations

Radial Smoother Knots for RSmooth(t2)	
Knot Number	t2
1	0
2	0.04400
3	0.1250
4	0.2020
5	0.3230
6	0.4140
7	0.5050
8	0.6010
9	0.6590

Number of Observations Read	598
Number of Observations Used	597

The "Dimensions" table shows that the model contains only two covariance parameters, the G-side variance of the spline coefficients (σ^2) and the R-side scale parameter (ϕ, Output 44.6.3). For each subject (cow), there are nine columns in the $\mathbf{Z}$ matrix, one per knot location. The GLIMMIX procedure processes these data by subjects (cows).

Output 44.6.3 Dimensions Information in Repeated Measures Analysis

Dimensions	
G-side Cov. Parameters	1
R-side Cov. Parameters	1
Columns in X	10
Columns in Z per Subject	9
Subjects (Blocks in V)	26
Max Obs per Subject	23

The "Optimization Information" table displays information about the optimization process. Because fixed effects and the residual scale parameter can be profiled from the optimization, the iterative algorithm involves only a single covariance parameter, the variance of the spline coefficients (Output 44.6.4).

Output 44.6.4 Optimization Information in Repeated Measures Analysis

Optimization Information	
Optimization Technique	Newton-Raphson
Parameters in Optimization	1
Lower Boundaries	1
Upper Boundaries	0
Fixed Effects	Profiled
Residual Variance	Profiled
Starting From	Data

After 11 iterations, the optimization process terminates (Output 44.6.5). In this case, the absolute gradient convergence criterion was met.

Output 44.6.5 Iteration History and Convergence Status

Iteration History					
Iteration	**Restarts**	**Evaluations**	**Objective Function**	**Change**	**Max Gradient**
0	**0**	4	-1302.549272	.	20.33682
1	**0**	3	-1451.587367	149.03809501	9.940495
2	**0**	3	-1585.640946	134.05357887	4.71531
3	**0**	3	-1694.516203	108.87525722	2.176741
4	**0**	3	-1775.290458	80.77425512	0.978577
5	**0**	3	-1829.966584	54.67612585	0.425724
6	**0**	3	-1862.878184	32.91160012	0.175992
7	**0**	3	-1879.329133	16.45094875	0.066061
8	**0**	3	-1885.175082	5.84594887	0.020137
9	**0**	3	-1886.238032	1.06295071	0.00372
10	**0**	3	-1886.288519	0.05048659	0.000198
11	**0**	3	-1886.288673	0.00015425	6.364E-7

Convergence criterion (ABSGCONV=0.00001) satisfied.

The generalized chi-square statistic in the "Fit Statistics" table is small for this model (Output 44.6.6). There is very little residual variation. The radial smoother is associated with 433.55 residual degrees of freedom, computed as 597 minus the trace of the smoother matrix.

Output 44.6.6 Fit Statistics in Repeated Measures Analysis

Fit Statistics	
-2 Res Log Likelihood	-1886.29
AIC (smaller is better)	-1882.29
AICC (smaller is better)	-1882.27
BIC (smaller is better)	-1879.77
CAIC (smaller is better)	-1877.77
HQIC (smaller is better)	-1881.56
Generalized Chi-Square	0.47
Gener. Chi-Square / DF	0.00
Radial Smoother df(res)	433.55

The "Covariance Parameter Estimates" table in Output 44.6.7 displays the estimates of the covariance parameters. The variance of the random spline coefficients is estimated as $\widehat{\sigma}^2 = 0.5961$, and the scale parameter (=residual variance) estimate is $\widehat{\phi} = 0.0008$.

Output 44.6.7 Estimated Covariance Parameters

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error
Var[RSmooth(t2)]	cow	0.5961	0.08144
Residual		0.000800	0.000059

The "Type III Tests of Fixed Effects" table displays F tests for the fixed effects in the MODEL statement (Output 44.6.8). There is a strong infection effect as well as the absence of an interaction between infection with *M. paratuberculosis* and iron dosing. It is important to note, however, that the interpretation of these tests rests on the assumption that the random effects in the mixed model have zero mean; in this case, the radial smoother coefficients.

Output 44.6.8 Tests of Fixed Effects

Type III Tests of Fixed Effects

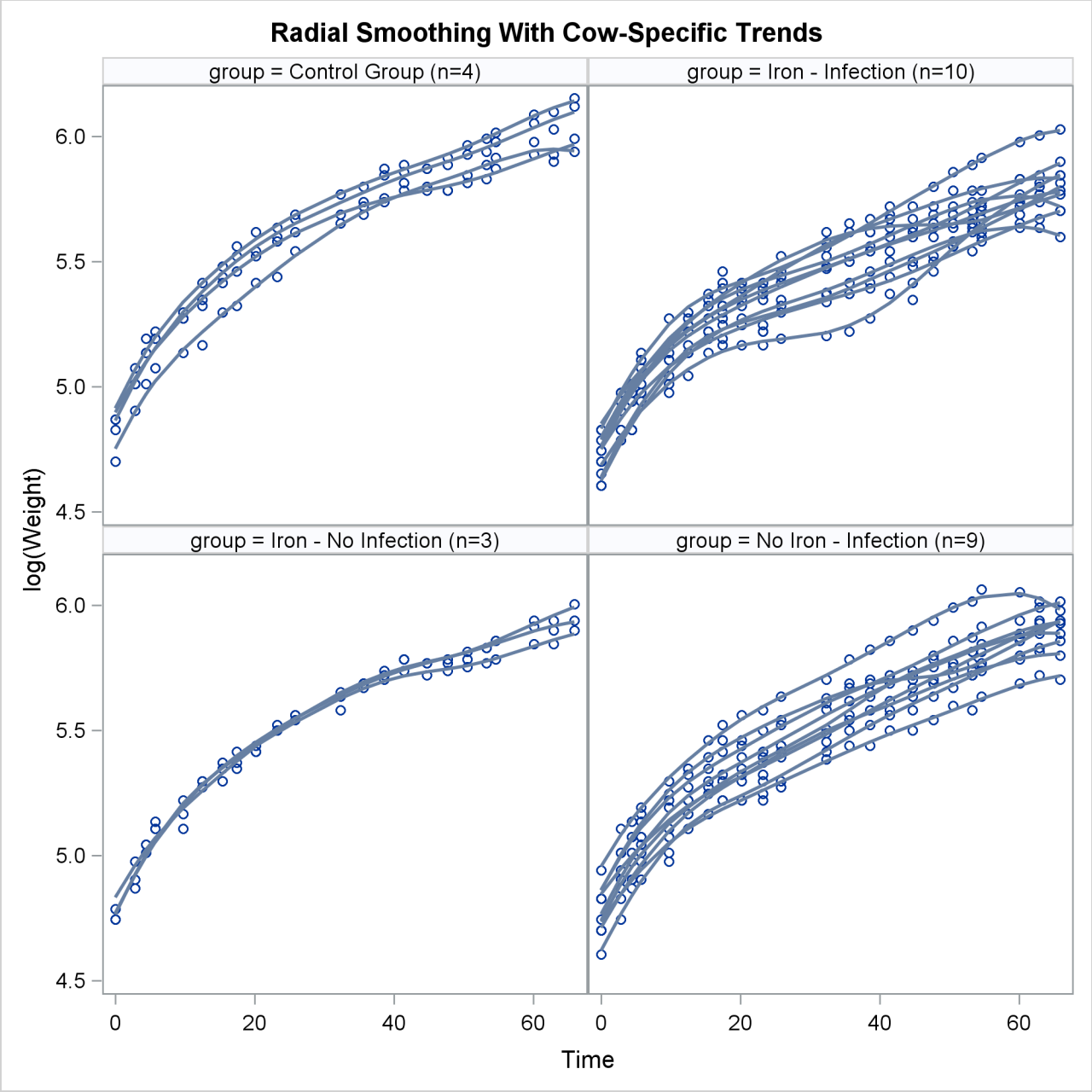
Effect	Num DF	Den DF	F Value	Pr > F
iron	1	358	3.59	0.0588
infection	1	358	21.16	<.0001
iron*infection	1	358	0.09	0.7637
tpoint	1	358	53.88	<.0001

A graph of the observed data and fitted profiles in the four groups is produced with the following statements (Output 44.6.9):

```
data plot;
   set gmxout;
   length group $26;
   if      (iron=0) and (infection=0) then group='Control Group (n=4)';
   else if (iron=1) and (infection=0) then group='Iron - No Infection (n=3)';
   else if (iron=0) and (infection=1) then group='No Iron - Infection (n=9)';
   else group = 'Iron - Infection (n=10)';
run;
proc sort data=plot; by group cow;
run;

proc sgpanel data=plot noautolegend;
   title 'Radial Smoothing With Cow-Specific Trends';
   label tpoint='Time' weight='log(Weight)';
   panelby group / columns=2 rows=2;
   scatter x=tpoint y=weight;
   series  x=tpoint y=pred / group=cow lineattrs=GraphFit;
run;
```

Output 44.6.9 Observed and Predicted Profiles



The trends are quite smooth, and you can see how the radial smoother adapts to the cow-specific profile. This is the reason for the small scale parameter estimate, $\widehat{\phi} = 0.008$. Comparing the panels at the top to the panels at the bottom of Output 44.6.9 reveals the effect of Infection. A comparison of the panels on the left to those on the right indicates the weak Iron effect.

The smoothing parameter in this analysis is related to the covariance parameter estimates. Because there is only one radial smoothing variance component, the amount of smoothing is the same in all four treatment groups. To test whether the smoothing parameter should be varied by group, you can refine the analysis of the previous model. The following statements fit the same general model, but they vary the covariance

parameters by the levels of the Iron*Infection interaction. This is accomplished with the GROUP= option in the RANDOM statement.

```
ods select OptInfo FitStatistics CovParms;
proc glimmix data=cows;
   t2 = tpoint / 100;
   class cow iron infection;
   model weight = iron infection iron*infection tpoint;
   random t2 / type=rsmooth
               subject=cow
               group=iron*infection
               knotmethod=kdtree(bucket=100);
   nloptions tech=newrap;
run;
```

All observations that have the same value combination of the Iron and Infection effects share the same covariance parameter. As a consequence, you obtain different smoothing parameters result in the four groups.

In Output 44.6.10, the "Optimization Information" table shows that there are now four covariance parameters in the optimization, one spline coefficient variance for each group.

Output 44.6.10 Analysis with Group-Specific Smoothing Parameter

The GLIMMIX Procedure

Optimization Information	
Optimization Technique	Newton-Raphson
Parameters in Optimization	4
Lower Boundaries	4
Upper Boundaries	0
Fixed Effects	Profiled
Residual Variance	Profiled
Starting From	Data

Fit Statistics	
-2 Res Log Likelihood	-1887.95
AIC (smaller is better)	-1877.95
AICC (smaller is better)	-1877.85
BIC (smaller is better)	-1871.66
CAIC (smaller is better)	-1866.66
HQIC (smaller is better)	-1876.14
Generalized Chi-Square	0.48
Gener. Chi-Square / DF	0.00
Radial Smoother df(res)	434.72

Covariance Parameter Estimates

Cov Parm	Subject	Group	Estimate	Standard Error
Var[RSmooth(t2)]	cow	iron*infection 0 0	0.4788	0.1922
Var[RSmooth(t2)]	cow	iron*infection 0 1	0.5152	0.1182
Var[RSmooth(t2)]	cow	iron*infection 1 0	0.4904	0.2195
Var[RSmooth(t2)]	cow	iron*infection 1 1	0.7105	0.1409
Residual			0.000807	0.000060

Varying this variance component by groups has changed the –2 Res Log Likelihood from –1886.29 to –1887.95 (Output 44.6.10). The difference, 1.66, can be viewed (asymptotically) as the realization of a chi-square random variable with three degrees of freedom. The difference is not significant ($p = 0.64586$). The "Covariance Parameter Estimates" table confirms that the estimates of the spline coefficient variance are quite similar in the four groups, ranging from 0.4788 to 0.7105.

Finally, you can apply a different technique for varying the temporal trends among the cows. From Output 44.6.9 it appears that an assumption of parallel trends within groups might be reasonable. In other words, you can fit a model in which the "overall" trend over time in each group is modeled nonparametrically, and this trend is shifted up or down to capture the behavior of the individual cow. You can accomplish this with the following statements:

```
ods select FitStatistics CovParms;
proc glimmix data=cows;
   t2 = tpoint / 100;
   class cow iron infection;
   model weight = iron infection iron*infection tpoint;
   random t2 / type=rsmooth
               subject=iron*infection
               knotmethod=kdtree(bucket=100);
   random intercept / subject=cow;
   output out=gmxout pred(blup)=pred;
   nloptions tech=newrap;
run;
```

There are now two subject effects in this analysis. The first RANDOM statement applies the radial smoothing and identifies the experimental conditions as the subject. For each condition, a separate realization of the random spline coefficients is obtained. The second RANDOM statement adds a random intercept to the trend for each cow. This random intercept results in the parallel shift of the trends over time.

Results from this analysis are shown in Output 44.6.11.

Output 44.6.11 Analysis with Parallel Shifts

The GLIMMIX Procedure

Fit Statistics	
-2 Res Log Likelihood	-1788.52
AIC (smaller is better)	-1782.52
AICC (smaller is better)	-1782.48
BIC (smaller is better)	-1788.52
CAIC (smaller is better)	-1785.52
HQIC (smaller is better)	-1788.52
Generalized Chi-Square	1.17
Gener. Chi-Square / DF	0.00
Radial Smoother df(res)	547.21

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error
Var[RSmooth(t2)]	iron*infection	0.5398	0.1940
Intercept	cow	0.007122	0.002173
Residual		0.001976	0.000121

Because the parallel shift model is not nested within either one of the previous models, the models cannot be compared with a likelihood ratio test. However, you can draw on the other fit statistics.

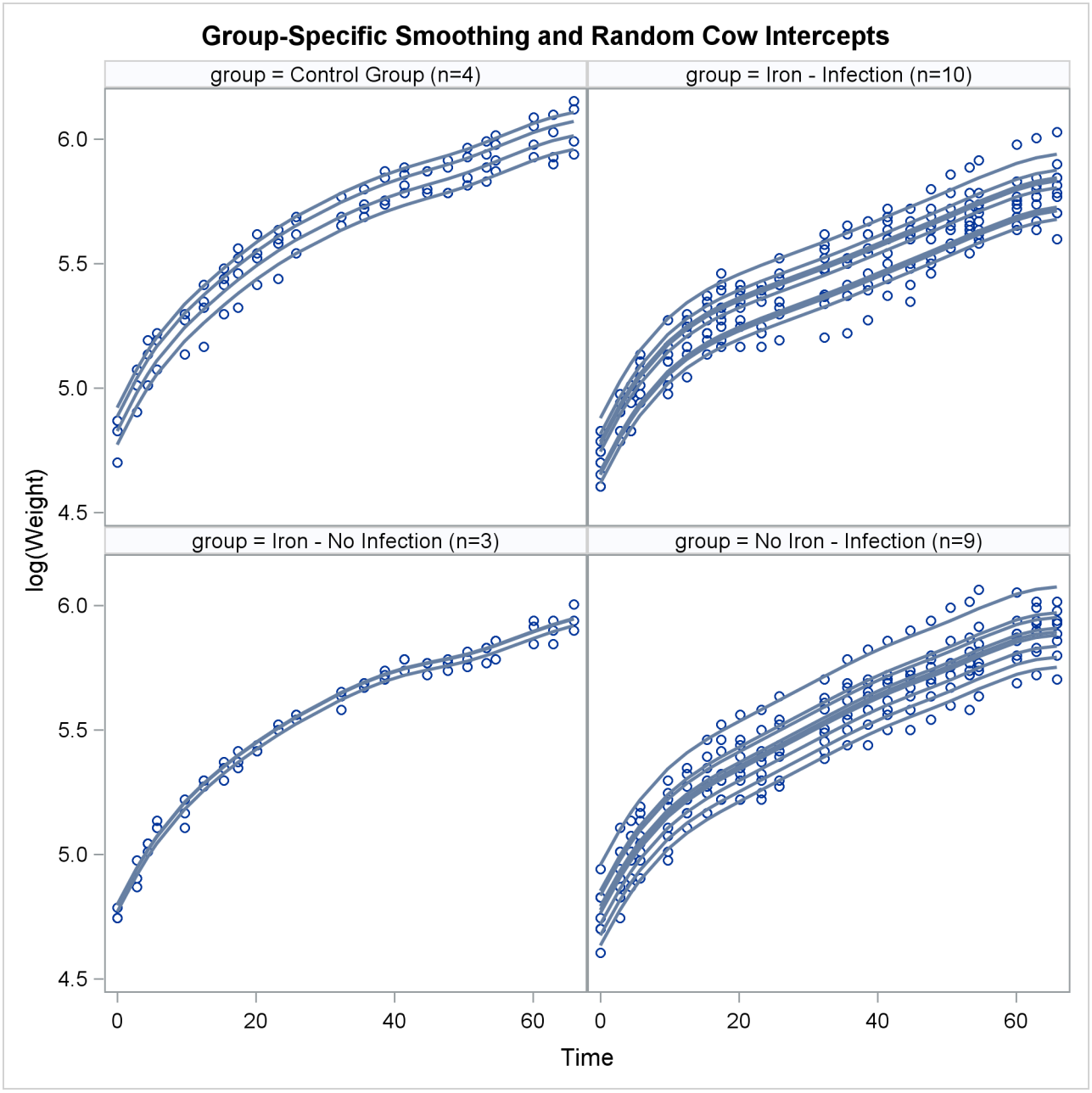
All statistics indicate that this model does not fit the data as well as the initial model that varies the spline coefficients by cow. The Pearson chi-square statistic is more than twice as large as in the previous model, indicating much more residual variation in the fit. On the other hand, this model generates only four sets of spline coefficients, one for each treatment group, and thus retains more residual degrees of freedom.

The "Covariance Parameter Estimates" table in Output 44.6.11 displays the solutions for the covariance parameters. The estimate of the variance of the spline coefficients is not that different from the estimate obtained in the first model (0.5961). The residual variance, however, has more than doubled.

Using similar SAS statements as previously, you can produce a plot of the observed and predicted profiles (Output 44.6.12).

The parallel shifts of the nonparametric smooths are clearly visible in Output 44.6.12. In the groups receiving only iron or only an infection, the parallel lines assumption holds quite well. In the control group and the group receiving iron and the infection, the parallel shift assumption does not hold as well. Two of the profiles in the iron-only group are nearly indistinguishable.

Output 44.6.12 Observed and Predicted Profiles



This example demonstrates that mixed model smoothing techniques can be applied not only to achieve scatter plot smoothing, but also to longitudinal or repeated measures data. You can then use the SUBJECT= option in the RANDOM statement to obtain independent sets of spline coefficients for different subjects, and the GROUP= option in the RANDOM statement to vary the degree of smoothing across groups. Also, radial smoothers can be combined with other random effects. For the data considered here, the appropriate model is one with a single smoothing parameter for all treatment group and cow-specific spline coefficients.

Example 44.7: Isotonic Contrasts for Ordered Alternatives

Dose response studies often focus on testing for monotone increasing or decreasing behavior in the mean values of the dependent variable. Hirotsu and Srivastava (2000) demonstrate one approach by using data that originally appeared in Moriguchi (1976). The data, which follow, consist of ferrite cores subjected to four increasing temperatures. The response variable is the magnetic force of each core.

```
data FerriteCores;
   do Temp = 1 to 4;
      do rep = 1 to 5; drop rep;
         input MagneticForce @@;
         output;
      end;
   end;
   datalines;
10.8  9.9 10.7 10.4  9.7
10.7 10.6 11.0 10.8 10.9
11.9 11.2 11.0 11.1 11.3
11.4 10.7 10.9 11.3 11.7
;
```

It is of interest to test whether the magnetic force of the cores rises monotonically with temperature. The approach of Hirotsu and Srivastava (2000) depends on the lower confidence limits of the *isotonic contrasts* of the force means at each temperature, adjusted for multiplicity. The corresponding isotonic contrast compares the average of a particular group and the preceding groups with the average of the succeeding groups. You can compute adjusted confidence intervals for isotonic contrasts by using the LSMESTIMATE statement.

The following statements request an analysis of the FerriteCores data as a one-way design and multiplicity-adjusted lower confidence limits for the isotonic contrasts. For the multiplicity adjustment, the LSMESTIMATE statement employs simulation, which provides adjusted *p*-values and lower confidence limits that are exact up to Monte Carlo error.

```
proc glimmix data=FerriteCores;
   class Temp;
   model MagneticForce = Temp;
   lsmestimate Temp
        'avg(1:1)<avg(2:4)' -3  1  1  1 divisor=3,
        'avg(1:2)<avg(3:4)' -1 -1  1  1 divisor=2,
        'avg(1:3)<avg(4:4)' -1 -1 -1  3 divisor=3
        / adjust=simulate(seed=1) cl upper;
   ods select LSMestimates;
run;
```

The results are shown in Output 44.7.1.

Output 44.7.1 Analysis of LS-Means with Isotonic Contrasts

The GLIMMIX Procedure

Least Squares Means Estimates
Adjustment for Multiplicity: Simulated

Effect	Label	Estimate	Standard Error	DF	t Value	Tails	Pr > t	Adj P	Alpha	Lower	Upper	Adj Lower	Adj Upper
Temp	avg(1:1)<avg(2:4)	0.8000	0.1906	16	4.20	Upper	0.0003	0.0010	0.05	0.4672	Infty	0.3771	Infty
Temp	avg(1:2)<avg(3:4)	0.7000	0.1651	16	4.24	Upper	0.0003	0.0009	0.05	0.4118	Infty	0.3337	Infty
Temp	avg(1:3)<avg(4:4)	0.4000	0.1906	16	2.10	Upper	0.0260	0.0625	0.05	0.06721	Infty	-0.02291	Infty

With an adjusted p-value of 0.001, the magnetic force at the first temperature is significantly less than the average of the other temperatures. Likewise, the average of the first two temperatures is significantly less than the average of the last two ($p = 0.0009$). However, the magnetic force at the last temperature is not significantly greater than the average magnetic force of the others ($p = 0.0625$). These results indicate a significant monotone increase over the first three temperatures, but not across all four temperatures.

Example 44.8: Adjusted Covariance Matrices of Fixed Effects

The following data are from Pothoff and Roy (1964) and consist of growth measurements for 11 girls and 16 boys at ages 8, 10, 12, and 14. Some of the observations are suspect (for example, the third observation for person 20); however, all of the data are used here for comparison purposes.

```
data pr;
   input child gender$ y1 y2 y3 y4;
   array yy y1-y4;
   do time=1 to 4;
      age = time*2 + 6;
      y   = yy{time};
      output;
   end;
   drop y1-y4;
   datalines;
 1   F   21.0   20.0   21.5   23.0
 2   F   21.0   21.5   24.0   25.5
 3   F   20.5   24.0   24.5   26.0
 4   F   23.5   24.5   25.0   26.5
 5   F   21.5   23.0   22.5   23.5
 6   F   20.0   21.0   21.0   22.5
 7   F   21.5   22.5   23.0   25.0
 8   F   23.0   23.0   23.5   24.0
 9   F   20.0   21.0   22.0   21.5
10   F   16.5   19.0   19.0   19.5
11   F   24.5   25.0   28.0   28.0
12   M   26.0   25.0   29.0   31.0
13   M   21.5   22.5   23.0   26.5
14   M   23.0   22.5   24.0   27.5
15   M   25.5   27.5   26.5   27.0
16   M   20.0   23.5   22.5   26.0
17   M   24.5   25.5   27.0   28.5
18   M   22.0   22.0   24.5   26.5
19   M   24.0   21.5   24.5   25.5
```

```
20   M   23.0    20.5    31.0    26.0
21   M   27.5    28.0    31.0    31.5
22   M   23.0    23.0    23.5    25.0
23   M   21.5    23.5    24.0    28.0
24   M   17.0    24.5    26.0    29.5
25   M   22.5    25.5    25.5    26.0
26   M   23.0    24.5    26.0    30.0
27   M   22.0    21.5    23.5    25.0
;
```

Jennrich and Schluchter (1986) analyze these data with various models for the fixed effects and the covariance structure. The strategy here is to fit a growth curve model for the boys and girls and to account for subject-to-subject variation through G-side random effects. In addition, serial correlation among the observations within each child is accounted for by a time series process. The data are assumed to be Gaussian, and their –2 restricted log likelihood is minimized to estimate the model parameters.

The following statements fit a mixed model in which a separate growth curve is assumed for each gender:

```
proc glimmix data=pr;
   class child gender time;
   model y = gender age gender*age / covb(details) ddfm=kr;
   random intercept age / type=chol sub=child;
   random time / subject=child type=ar(1) residual;
   ods select ModelInfo CovB CovBModelBased CovBDetails;
run;
```

The growth curve for an individual child differs from the gender-specific trend because of a random intercept and a random slope. The two G-side random effects are assumed to be correlated. Their unstructured covariance matrix is parameterized in terms of the Cholesky root to guarantee a positive (semi-)definite estimate. An AR(1) covariance structure is modeled for the observations over time for each child. Notice the RESIDUAL option in the second RANDOM statement. It identifies this as an R-side random effect.

The DDFM=KR option requests that the covariance matrix of the fixed-effect parameter estimates and denominator degrees of freedom for t and F tests are determined according to Kenward and Roger (1997). This is reflected in the "Model Information" table (Output 44.8.1).

Output 44.8.1 Model Information with DDFM=KR

The GLIMMIX Procedure

Model Information	
Data Set	WORK.PR
Response Variable	y
Response Distribution	Gaussian
Link Function	Identity
Variance Function	Default
Variance Matrix Blocked By	child
Estimation Technique	Restricted Maximum Likelihood
Degrees of Freedom Method	Kenward-Roger
Fixed Effects SE Adjustment	Kenward-Roger

The COVB option in the MODEL statement requests that the covariance matrix used for inference about fixed effects in this model is displayed; this is the Kenward-Roger-adjusted covariance matrix. The DETAILS suboption requests that the unadjusted covariance matrix is also displayed (Output 44.8.2). In addition, a table of diagnostic measures for the covariance matrices is produced.

Output 44.8.2 Model-Based and Adjusted Covariance Matrix

Model Based Covariance Matrix for Fixed Effects (Unadjusted)

Effect	gender	Row	Col1	Col2	Col3	Col4	Col5	Col6
Intercept		1	0.9969	-0.9969		-0.07620	0.07620	
gender	F	2	-0.9969	2.4470		0.07620	-0.1870	
gender	M	3						
age		4	-0.07620	0.07620		0.007581	-0.00758	
age*gender	F	5	0.07620	-0.1870		-0.00758	0.01861	
age*gender	M	6						

Covariance Matrix for Fixed Effects

Effect	gender	Row	Col1	Col2	Col3	Col4	Col5	Col6
Intercept		1	0.9724	-0.9724		-0.07412	0.07412	
gender	F	2	-0.9724	2.3868		0.07412	-0.1819	
gender	M	3						
age		4	-0.07412	0.07412		0.007256	-0.00726	
age*gender	F	5	0.07412	-0.1819		-0.00726	0.01781	
age*gender	M	6						

Diagnostics for Covariance Matrices of Fixed Effects

		Model-Based	Adjusted
Dimensions	Rows	6	6
	Non-zero entries	16	16
Summaries	Trace	3.4701	3.3843
	Log determinant	-11.95	-12.17
Eigenvalues	> 0	4	4
	= 0	2	2
	max abs	2.972	2.8988
	min abs non-zero	0.0009	0.0008
	Condition number	3467.8	3698.2
Norms	Frobenius	3.0124	2.9382
	Infinity	3.7072	3.6153
Comparisons	Concordance correlation		0.9979
	Discrepancy function		0.0084
	Frobenius norm of difference		0.0742
	Trace(Adjusted Inv(MBased))		3.7801

Determinant and inversion results apply to the nonsingular partitions of the covariance matrices.

The "Diagnostics for Covariance Matrices" table in Output 44.8.2 consists of several sections. The trace and log determinant of covariance matrices are general scalar summaries that are sometimes used in direct comparisons, or in formulating further statistics, such as the difference of log determinants. The trace simply represents the sum of the variances of all fixed-effects parameters.

The two matrices have the same number of positive and zero eigenvalues; hence they are of the same rank. There are no negative eigenvalues; hence the matrices are positive semi-definite.

The "Comparisons" section of the table provides several statistics that set the matrices in relationship. The statistics enable you to assess the extent to which the adjustment affected the model-based matrix. If the two matrices are identical, the concordance correlation equals 1, the discrepancy function and the Frobenius norm of the differences equal 0, and the trace of the adjusted and the (generalized) inverse of the model-based matrix equals the rank. See the section "Exploring and Comparing Covariance Matrices" on page 3216 for computational details regarding these statistics. With increasing discrepancy between the matrices, the difference norm and discrepancy function increase, the concordance correlation falls below 1, and the trace deviates from the rank. In this particular example, there is strong agreement between the two matrices; the adjustment to the covariance matrix associated with DDFM=KR is only slight. It is noteworthy, however, that the trace of the adjusted covariance matrix falls short of the trace of the unadjusted one. Indeed, from Output 44.8.2 you can see that the diagonal elements of the adjusted covariance matrices are uniformly smaller than those of the model-based covariance matrix.

Standard error "shrinkage" for the Kenward-Roger covariance adjustment is due to the term $-0.25\mathbf{R}_{ij}$ in equation (3) of Kenward and Roger (1997), which is nonzero for covariance structures with second derivatives, such as the TYPE=ANTE(1), TYPE=AR(1), TYPE=ARH(1), TYPE=ARMA(1,1), TYPE=CHOL, TYPE=CSH, TYPE=FA0(q), TYPE=TOEPH, and TYPE=UNR structures and all TYPE=SP() structures.

For covariance structures that are linear in the parameters, $\mathbf{R}_{ij} = \mathbf{0}$. You can add the FIRSTORDER suboption to the DDFM=KR option to request that second derivative matrices $\mathbf{R}_{ij}$ are excluded from computing the covariance matrix adjustment. The resulting covariance adjustment is that of Kackar and Harville (1984) and Harville and Jeske (1992). This estimator is denoted as $\widetilde{m}^{@}$ in Harville and Jeske (1992) and is referred to there as the Prasad-Rao estimator after related work by Prasad and Rao (1990). This standard error adjustment is guaranteed to be positive (semi-)definite. The following statements fit the model with the Kackar-Harville-Jeske estimator and compare model-based and adjusted covariance matrices:

```
proc glimmix data=pr;
   class child gender time;
   model y = gender age gender*age / covb(details)
                                     ddfm=kr(firstorder);
   random intercept age / type=chol sub=child;
   random time / subject=child type=ar(1) residual;
   ods select ModelInfo CovB CovBDetails;
run;
```

The standard error adjustment is reflected in the "Model Information" table (Output 44.8.3).

Output 44.8.3 Model Information with DDFM=KR(FIRSTORDER)

The GLIMMIX Procedure

Model Information	
Data Set	WORK.PR
Response Variable	y
Response Distribution	Gaussian
Link Function	Identity
Variance Function	Default
Variance Matrix Blocked By	child
Estimation Technique	Restricted Maximum Likelihood
Degrees of Freedom Method	Kenward-Roger
Fixed Effects SE Adjustment	Prasad-Rao-Kackar-Harville-Jeske

Output 44.8.4 displays the adjusted covariance matrix. Notice that the elements of this matrix, in particular the diagonal elements, are larger in absolute value than those of the model-based estimator (Output 44.8.2).

Output 44.8.4 Adjusted Covariance Matrix and Comparison to Model-Based Estimator

Covariance Matrix for Fixed Effects							
Effect	**gender**	**Row**	**Col1**	**Col2**	**Col3**	**Col4**	**Col5**	**Col6**
Intercept		**1**	1.0122	-1.0122		-0.07758	0.07758	
gender	F	**2**	-1.0122	2.4845		0.07758	-0.1904	
gender	M	**3**						
age		**4**	-0.07758	0.07758		0.007706	-0.00771	
age*gender	F	**5**	0.07758	-0.1904		-0.00771	0.01891	
age*gender	M	**6**						

Output 44.8.4 *continued*

Diagnostics for Covariance Matrices of Fixed Effects			
		Model-Based	Adjusted
Dimensions	Rows	6	6
	Non-zero entries	16	16
Summaries	Trace	3.4701	3.5234
	Log determinant	-11.95	-11.91
Eigenvalues	> 0	4	4
	= 0	2	2
	max abs	2.972	3.0176
	min abs non-zero	0.0009	0.0009
	Condition number	3467.8	3513.4
Norms	Frobenius	3.0124	3.0587
	Infinity	3.7072	3.7647
Comparisons	Concordance correlation		0.9999
	Discrepancy function		0.0003
	Frobenius norm of difference		0.0463
	Trace(Adjusted Inv(MBased))		4.0352
Determinant and inversion results apply to the nonsingular partitions of the covariance matrices.			

The "Comparisons" statistics show that the model-based and adjusted covariance matrix of the fixed-effects parameter estimates are very similar. The concordance correlation is near 1, the discrepancy is near zero, and the trace is very close to the number of positive eigenvalues. This is due to the balanced nature of these repeated measures data. Shrinkage of standard errors, however, can not occur with the Kackar-Harville-Jeske estimator.

Example 44.9: Testing Equality of Covariance and Correlation Matrices

Fisher's iris data are widely used in multivariate statistics. They comprise measurements in millimeters of four flower attributes, the length and width of sepals and petals for 50 specimens from each of three species, *Iris setosa*, *I. versicolor*, and *I. virginica* (Fisher 1936).

When modeling multiple attributes from the same specimen, correlations among measurements from the same flower must be taken into account. Unstructured covariance matrices are common in this multivariate setting. Species comparisons can focus on comparisons of mean response, but comparisons of the variation and covariation are also of interest. In this example, the equivalence of covariance and correlation matrices among the species are examined.

The iris data set is available in the Sashelp library. The following step displays the first 10 observations of the iris data in multivariate format—that is, each observation contains multiple response variables. The DATA step that follows creates a data set in univariate form, where each observation corresponds to a single response variable. This is the form needed by the GLIMMIX procedure.

```
proc print data=Sashelp.iris(obs=10);
run;
```

Output 44.9.1 Fisher (1936) Iris Data

Obs	Species	SepalLength	SepalWidth	PetalLength	PetalWidth
1	Setosa	50	33	14	2
2	Setosa	46	34	14	3
3	Setosa	46	36	10	2
4	Setosa	51	33	17	5
5	Setosa	55	35	13	2
6	Setosa	48	31	16	2
7	Setosa	52	34	14	2
8	Setosa	49	36	14	1
9	Setosa	44	32	13	2
10	Setosa	50	35	16	6

```
data iris_univ;
   set sashelp.iris;
   retain id 0;
   array y (4) SepalLength SepalWidth PetalLength PetalWidth;
   id+1;
   do var=1 to 4;
      response = y{var};
      output;
   end;
   drop SepalLength SepalWidth PetalLength PetalWidth:;
run;
```

The following GLIMMIX statements fit a model with separate unstructured covariance matrices for each species:

```
ods select FitStatistics CovParms CovTests;
proc glimmix data=iris_univ;
   class species var id;
   model response = species*var;
   random _residual_ / type=un group=species subject=id;
   covtest homogeneity;
run;
```

The mean function is modeled as a cell-means model that allows for different means for each species and outcome variable. The covariances are modeled directly (R-side) rather than through random effects. The ID variable identifies the individual plant, so that responses from different plants are independent. The GROUP=SPECIES option varies the parameters of the unstructured covariance matrix by species. Hence, this model has 30 covariance parameters: 10 unique parameters for a (4×4) covariance matrix for each of three species.

The COVTEST statement requests a test of homogeneity—that is, it tests whether varying the covariance parameters by the group effect provides a significantly better fit compared to a model in which different groups share the same parameter.

Output 44.9.2 Fit Statistics for Analysis of Fisher's Iris Data

The GLIMMIX Procedure

Fit Statistics	
-2 Res Log Likelihood	2812.89
AIC (smaller is better)	2872.89
AICC (smaller is better)	2876.23
BIC (smaller is better)	2963.21
CAIC (smaller is better)	2993.21
HQIC (smaller is better)	2909.58
Generalized Chi-Square	588.00
Gener. Chi-Square / DF	1.00

The "Fit Statistics" table shows the –2 restricted (residual) log likelihood in the full model and other fit statistics (Output 44.9.2). The "-2 Res Log Likelihood" sets the benchmark against which a model with homogeneity constraint is compared. Output 44.9.3 displays the 30 covariance parameters in this model.

There appear to be substantial differences among the covariance parameters from different groups. For example, the residual variability of the petal length of the three species is 12.4249, 26.6433, and 40.4343, respectively. The homogeneity hypothesis restricts these variances to be equal and similarly for the other covariance parameters. The results from the COVTEST statement are shown in Output 44.9.4.

Output 44.9.3 Covariance Parameters Varied by Species (TYPE=UN)

Covariance Parameter Estimates

Cov Parm	Subject	Group	Estimate	Standard Error
UN(1,1)	id	Species Setosa	12.4249	2.5102
UN(2,1)	id	Species Setosa	9.9216	2.3775
UN(2,2)	id	Species Setosa	14.3690	2.9030
UN(3,1)	id	Species Setosa	1.6355	0.9052
UN(3,2)	id	Species Setosa	1.1698	0.9552
UN(3,3)	id	Species Setosa	3.0159	0.6093
UN(4,1)	id	Species Setosa	1.0331	0.5508
UN(4,2)	id	Species Setosa	0.9298	0.5859
UN(4,3)	id	Species Setosa	0.6069	0.2755
UN(4,4)	id	Species Setosa	1.1106	0.2244
UN(1,1)	id	Species Versicolor	26.6433	5.3828
UN(2,1)	id	Species Versicolor	8.5184	2.6144
UN(2,2)	id	Species Versicolor	9.8469	1.9894
UN(3,1)	id	Species Versicolor	18.2898	4.3398
UN(3,2)	id	Species Versicolor	8.2653	2.4149
UN(3,3)	id	Species Versicolor	22.0816	4.4612
UN(4,1)	id	Species Versicolor	5.5780	1.6617
UN(4,2)	id	Species Versicolor	4.1204	1.0641
UN(4,3)	id	Species Versicolor	7.3102	1.6891
UN(4,4)	id	Species Versicolor	3.9106	0.7901
UN(1,1)	id	Species Virginica	40.4343	8.1690
UN(2,1)	id	Species Virginica	9.3763	3.2213
UN(2,2)	id	Species Virginica	10.4004	2.1012
UN(3,1)	id	Species Virginica	30.3290	6.6262
UN(3,2)	id	Species Virginica	7.1380	2.7395
UN(3,3)	id	Species Virginica	30.4588	6.1536
UN(4,1)	id	Species Virginica	4.9094	2.5916
UN(4,2)	id	Species Virginica	4.7629	1.4367
UN(4,3)	id	Species Virginica	4.8824	2.2750
UN(4,4)	id	Species Virginica	7.5433	1.5240

Output 44.9.4 Likelihood Ratio Test of Homogeneity

Tests of Covariance Parameters Based on the Restricted Likelihood

Label	DF	-2 Res Log Like	ChiSq	Pr > ChiSq	Note
Homogeneity	20	2959.55	146.66	<.0001	DF

DF: P-value based on a chi-square with DF degrees of freedom.

Denote as $\mathbf{R}_k$ the covariance matrix for species $k = 1, 2, 3$ with elements σ_{ijk}. In processing the COVTEST hypothesis $H_0: \mathbf{R}_1 = \mathbf{R}_2 = \mathbf{R}_3$, the GLIMMIX procedure fits a model that satisfies the constraints

$$\begin{aligned}
\sigma_{111} &= \sigma_{112} = \sigma_{113} \\
\sigma_{211} &= \sigma_{212} = \sigma_{213} \\
\sigma_{231} &= \sigma_{232} = \sigma_{233} \\
&\vdots \\
\sigma_{441} &= \sigma_{442} = \sigma_{443}
\end{aligned}$$

where σ_{ijk} is the covariance between the ith and jth variable for the kth species. The –2 restricted log likelihood of this restricted model is 2959.55 (Output 44.9.4). The change of 146.66 compared to the full model is highly significant. There is sufficient evidence to reject the notion of equal covariance matrices among the three iris species.

Equality of covariance matrices implies equality of correlation matrices, but the reverse is not true. Fewer constraints are needed to equate correlations because the diagonal entries of the covariance matrices are free to vary. In order to test the equality of the correlation matrices among the three species, you can parameterize the unstructured covariance matrix in terms of the correlations and use a COVTEST statement with general contrasts, as shown in the following statements:

```
ods select FitStatistics CovParms CovTests;
proc glimmix data=iris_univ;
   class species var id;
   model response = species*var;
   random _residual_ / type=unr group=species subject=id;
   covtest 'Equal Covariance Matrices'  homogeneity;
   covtest 'Equal Correlation Matrices' general
        0  0  0  0  1  0  0  0  0  0
        0  0  0  0 -1  0  0  0  0  0,
        0  0  0  0  1  0  0  0  0  0
        0  0  0  0  0  0  0  0  0  0
        0  0  0  0 -1  0  0  0  0  0,
        0  0  0  0  0  1  0  0  0  0
        0  0  0  0  0 -1  0  0  0  0,
        0  0  0  0  0  1  0  0  0  0
        0  0  0  0  0  0  0  0  0  0
        0  0  0  0  0 -1  0  0  0  0,
        0  0  0  0  0  0  1  0  0  0
        0  0  0  0  0  0 -1  0  0  0,
        0  0  0  0  0  0  1  0  0  0
        0  0  0  0  0  0  0  0  0  0
        0  0  0  0  0  0 -1  0  0  0,
        0  0  0  0  0  0  0  1  0  0
        0  0  0  0  0  0  0 -1  0  0,
        0  0  0  0  0  0  0  1  0  0
        0  0  0  0  0  0  0  0  0  0
        0  0  0  0  0  0  0 -1  0  0,
        0  0  0  0  0  0  0  0  1  0
        0  0  0  0  0  0  0  0 -1  0,
        0  0  0  0  0  0  0  0  1  0
        0  0  0  0  0  0  0  0  0  0
```

```
          0  0  0  0  0  0  0  0  -1 0,
          0  0  0  0  0  0  0  0  0  1
          0  0  0  0  0  0  0  0  0 -1,
          0  0  0  0  0  0  0  0  0  1
          0  0  0  0  0  0  0  0  0  0
          0  0  0  0  0  0  0  0  0 -1 / estimates;
run;
```

The TYPE=UNR structure is a reparameterization of TYPE=UN. The models provide the same fit, as seen by comparison of the "Fit Statistics" tables in Output 44.9.2 and Output 44.9.5. The covariance parameters are ordered differently, however. In each group, the four variances precede the six correlations (Output 44.9.5). The first COVTEST statement tests the homogeneity hypothesis in terms of the UNR parameterization, and the result is identical to the test in Output 44.9.4. The second COVTEST statement restricts the correlations to be equal across groups. If ρ_{ijk} is the correlation between the ith and jth variable for the kth species, the 12 restrictions are

$$\rho_{211} = \rho_{212} = \rho_{213}$$
$$\rho_{311} = \rho_{312} = \rho_{313}$$
$$\rho_{321} = \rho_{322} = \rho_{323}$$
$$\rho_{411} = \rho_{412} = \rho_{413}$$
$$\rho_{421} = \rho_{422} = \rho_{423}$$
$$\rho_{431} = \rho_{432} = \rho_{433}$$

The ESTIMATES option in the COVTEST statement requests that the GLIMMIX procedure display the covariance parameter estimates in the restricted model (Output 44.9.5).

Output 44.9.5 Fit Statistics, Covariance Parameters (TYPE=UNR), and Likelihood Ratio Tests for Equality of Covariance and Correlation Matrices

The GLIMMIX Procedure

Fit Statistics	
-2 Res Log Likelihood	2812.89
AIC (smaller is better)	2872.89
AICC (smaller is better)	2876.23
BIC (smaller is better)	2963.21
CAIC (smaller is better)	2993.21
HQIC (smaller is better)	2909.58
Generalized Chi-Square	588.00
Gener. Chi-Square / DF	1.00

Output 44.9.5 *continued*

Covariance Parameter Estimates

Cov Parm	Subject	Group	Estimate	Standard Error
Var(1)	id	Species Setosa	12.4249	2.5102
Var(2)	id	Species Setosa	14.3690	2.9030
Var(3)	id	Species Setosa	3.0159	0.6093
Var(4)	id	Species Setosa	1.1106	0.2244
Corr(2,1)	id	Species Setosa	0.7425	0.06409
Corr(3,1)	id	Species Setosa	0.2672	0.1327
Corr(3,2)	id	Species Setosa	0.1777	0.1383
Corr(4,1)	id	Species Setosa	0.2781	0.1318
Corr(4,2)	id	Species Setosa	0.2328	0.1351
Corr(4,3)	id	Species Setosa	0.3316	0.1271
Var(1)	id	Species Versicolor	26.6433	5.3828
Var(2)	id	Species Versicolor	9.8469	1.9894
Var(3)	id	Species Versicolor	22.0816	4.4612
Var(4)	id	Species Versicolor	3.9106	0.7901
Corr(2,1)	id	Species Versicolor	0.5259	0.1033
Corr(3,1)	id	Species Versicolor	0.7540	0.06163
Corr(3,2)	id	Species Versicolor	0.5605	0.09797
Corr(4,1)	id	Species Versicolor	0.5465	0.1002
Corr(4,2)	id	Species Versicolor	0.6640	0.07987
Corr(4,3)	id	Species Versicolor	0.7867	0.05445
Var(1)	id	Species Virginica	40.4343	8.1690
Var(2)	id	Species Virginica	10.4004	2.1012
Var(3)	id	Species Virginica	30.4588	6.1536
Var(4)	id	Species Virginica	7.5433	1.5240
Corr(2,1)	id	Species Virginica	0.4572	0.1130
Corr(3,1)	id	Species Virginica	0.8642	0.03616
Corr(3,2)	id	Species Virginica	0.4010	0.1199
Corr(4,1)	id	Species Virginica	0.2811	0.1316
Corr(4,2)	id	Species Virginica	0.5377	0.1015
Corr(4,3)	id	Species Virginica	0.3221	0.1280

Output 44.9.5 *continued*

Tests of Covariance Parameters Based on the Restricted Likelihood

					Estimates H0									
Label	DF	-2 Res Log Like	ChiSq	Pr > ChiSq	Est1	Est2	Est3	Est4	Est5	Est6	Est7	Est8	Est9	Est10
Equal Covariance Matrices	20	2959.55	146.66	<.0001	26.5004	11.5395	18.5179	4.1883	0.5302	0.7562	0.3779	0.3645	0.4705	0.4845
Equal Correlation Matrices	12	2876.38	63.49	<.0001	16.4715	14.8656	4.8427	1.4392	0.5612	0.6827	0.4016	0.3844	0.4976	0.5219

Tests of Covariance Parameters Based on the Restricted Likelihood

	Estimates H0													
Label	Est11	Est12	Est13	Est14	Est15	Est16	Est17	Est18	Est19	Est20	Est21	Est22	Est23	Est24
Equal Covariance Matrices	26.5004	11.5395	18.5179	4.1883	0.5302	0.7562	0.3779	0.3645	0.4705	0.4845	26.5004	11.5395	18.5179	4.1883
Equal Correlation Matrices	24.4020	9.1566	17.4434	3.0021	0.5612	0.6827	0.4016	0.3844	0.4976	0.5219	35.0544	10.8350	27.3593	8.1395

Tests of Covariance Parameters Based on the Restricted Likelihood

	Estimates H0						
Label	Est25	Est26	Est27	Est28	Est29	Est30	Note
Equal Covariance Matrices	0.5302	0.7562	0.3779	0.3645	0.4705	0.4845	DF
Equal Correlation Matrices	0.5612	0.6827	0.4016	0.3844	0.4976	0.5219	DF

DF: P-value based on a chi-square with DF degrees of freedom.

The result of the homogeneity test is identical to that in Output 44.9.4. The hypothesis of equality of the correlation matrices is also rejected with a chi-square value of 63.49 and a p-value of < 0.0001. Notice, however, that the chi-square statistic is smaller than in the test of homogeneity due to the smaller number of restrictions imposed on the full model. The estimate of the common correlation matrix in the restricted model is

$$\begin{bmatrix} 1 & 0.561 & 0.683 & 0.384 \\ 0.561 & 1 & 0.402 & 0.498 \\ 0.683 & 0.402 & 1 & 0.522 \\ 0.384 & 0.498 & 0.522 & 1 \end{bmatrix}$$

Example 44.10: Multiple Trends Correspond to Multiple Extrema in Profile Likelihoods

Observations for a period of 168 months for the "Southern Oscillation Index," measurements of monthly averaged atmospheric pressure differences between Easter Island and Darwin, Australia (Kahaner, Moler, and Nash 1989, Ch. 11.9; National Institute of Standards and Technology 1998) is available in the data set ENSO in the Sashelp library. These data are also used as an example in Chapter 59, "The LOESS Procedure." The following statements print the first 10 observations of this data set in Output 44.10.1.

```
proc print data=Sashelp.enso(obs=10);
run;
```

Output 44.10.1 El Niño Southern Oscillation Data

Obs	Month	Year	Pressure
1	1	0.08333	12.9
2	2	0.16667	11.3
3	3	0.25000	10.6
4	4	0.33333	11.2
5	5	0.41667	10.9
6	6	0.50000	7.5
7	7	0.58333	7.7
8	8	0.66667	11.7
9	9	0.75000	12.9
10	10	0.83333	14.3

Differences in atmospheric pressure create wind, and the differences recorded in the data set ENSO drive the trade winds in the southern hemisphere. Such time series often do not consist of a single trend or cycle. In this particular case, there are at least two known cycles that reflect the annual weather pattern and a longer cycle that represents the periodic warming of the Pacific Ocean (El Niño).

To estimate the trend in these data by using mixed model technology, you can apply a mixed model smoothing technique such as TYPE=RSMOOTH or TYPE=PSPLINE. The following statements fit a radial smoother to the ENSO data and obtain profile likelihoods for a series of values for the variance of the random spline coefficients:

```
data tdata;
   do covp1=0,0.0005,0.05,0.1,0.2,0.5,
            1,2,3,4,5,6,8,10,15,20,50,
            75,100,125,140,150,160,175,
            200,225,250,275,300,350;
      output;
   end;
run;

ods select FitStatistics CovParms CovTests;
proc glimmix data=sashelp.enso noprofile;
   model pressure = year;
   random year / type=rsmooth knotmethod=equal(50);
   parms (2) (10);
```

```
   covtest tdata=tdata / parms;
   ods output covtests=ct;
run;
```

The tdata data set contains value for the variance of the radial smoother variance for which the profile likelihood of the model is to be computed. The profile likelihood is obtained by setting the radial smoother variance at the specified value and estimating all other parameters subject to that constraint.

Because the model contains a residual variance and you need to specify nonzero values for the first covariance parameter, the NOPROFILE is added to the PROC GLIMMIX statements. If the residual variance is profiled from the estimation, you cannot fix covariance parameters at a given value, because they would be reexpressed during model fitting in terms of ratios with the profiled (and changing) variance.

The PARMS statement determines starting values for the covariance parameters for fitting the (full) model. The PARMS option in the COVTEST statement requests that the input parameters be added to the output and the output data set. This is useful for subsequent plotting of the profile likelihood function.

The "Fit Statistics" table displays the –2 restricted log likelihood of the model (897.76, Output 44.10.2). The estimate of the variance of the radial smoother coefficients is 3.5719.

The "Test of Covariance Parameters" table displays the –2 restricted log likelihood for each observation in the tdata set. Because the tdata data set specifies values for only the first covariance parameter, the second covariance parameter is free to vary and the values for –2 Res Log Like are profile likelihoods. Notice that for a number of values of CovP1 the chi-square statistic is missing in this table. For these values the –2 Res Log Like is *smaller* than that of the full model. The model did not converge to a global minimum of the negative restricted log likelihood.

Output 44.10.2 REML and Profile Likelihood Analysis

The GLIMMIX Procedure

Fit Statistics	
-2 Res Log Likelihood	897.76
AIC (smaller is better)	901.76
AICC (smaller is better)	901.83
BIC (smaller is better)	897.76
CAIC (smaller is better)	899.76
HQIC (smaller is better)	897.76
Generalized Chi-Square	1554.38
Gener. Chi-Square / DF	9.36
Radial Smoother df(res)	153.52

Covariance Parameter Estimates

Cov Parm	Estimate	Standard Error
Var[RSmooth(Year)]	3.5719	3.7672
Residual	9.3638	1.3014

Output 44.10.2 *continued*

Tests of Covariance Parameters Based on the Restricted Likelihood

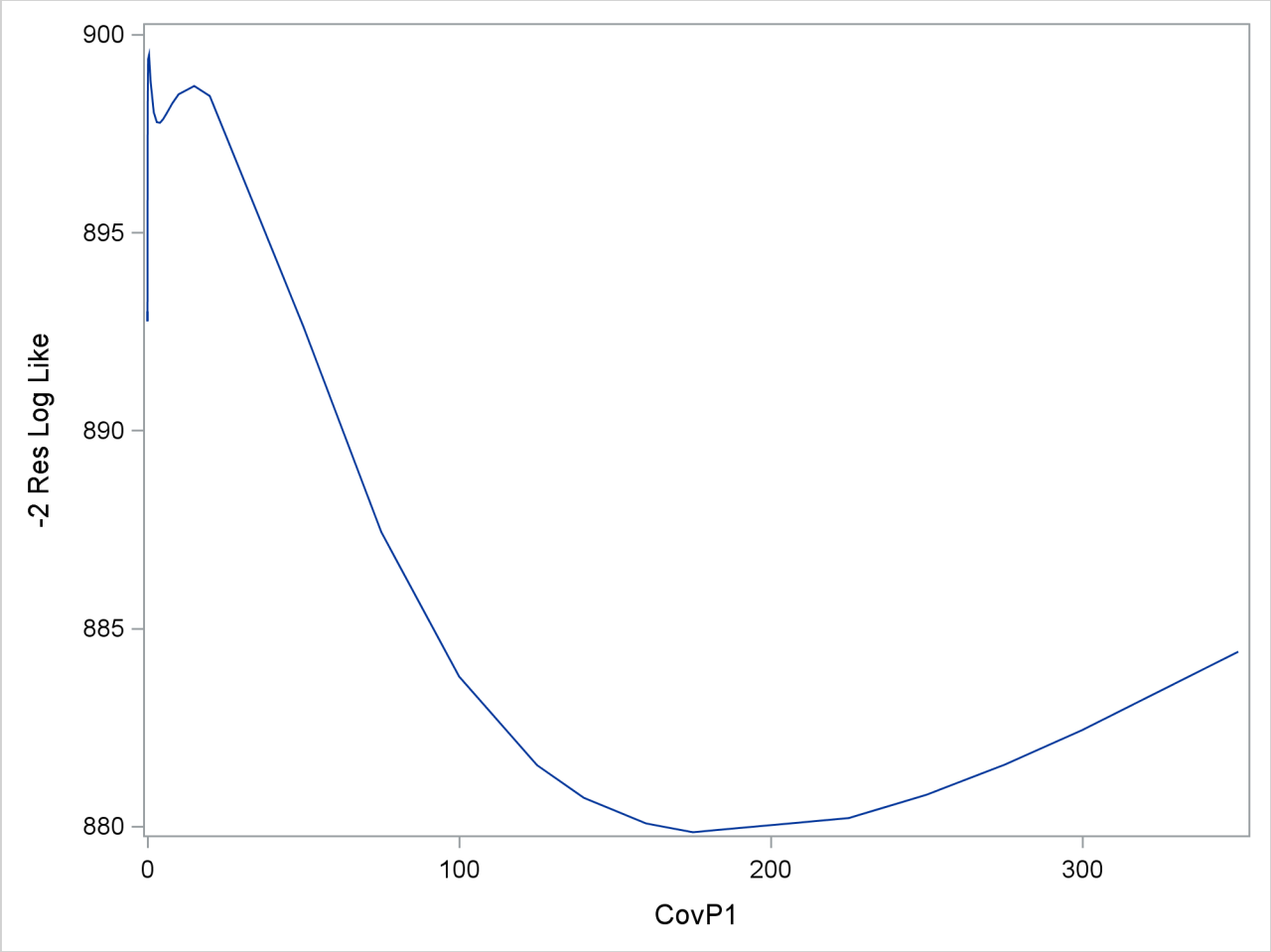
					Input Parameters		
Label	DF	-2 Res Log Like	ChiSq	Pr > ChiSq	CovP1	CovP2	Note
WORK.TDATA	1	893.01	.	1.0000	0	9.3638	MI
WORK.TDATA	1	892.76	.	1.0000	0.000500	9.3638	DF
WORK.TDATA	1	897.34	.	1.0000	0.05000	9.3638	DF
WORK.TDATA	1	898.53	0.77	0.3816	0.1000	9.3638	DF
WORK.TDATA	1	899.38	1.62	0.2038	0.2000	9.3638	DF
WORK.TDATA	1	899.49	1.73	0.1888	0.5000	9.3638	DF
WORK.TDATA	1	898.83	1.07	0.3016	1.0000	9.3638	DF
WORK.TDATA	1	898.04	0.28	0.5967	2.0000	9.3638	DF
WORK.TDATA	1	897.79	0.03	0.8693	3.0000	9.3638	DF
WORK.TDATA	1	897.77	0.01	0.9145	4.0000	9.3638	DF
WORK.TDATA	1	897.86	0.10	0.7517	5.0000	9.3638	DF
WORK.TDATA	1	897.99	0.23	0.6311	6.0000	9.3638	DF
WORK.TDATA	1	898.27	0.51	0.4761	8.0000	9.3638	DF
WORK.TDATA	1	898.49	0.73	0.3919	10.0000	9.3638	DF
WORK.TDATA	1	898.70	0.94	0.3318	15.0000	9.3638	DF
WORK.TDATA	1	898.45	0.69	0.4068	20.0000	9.3638	DF
WORK.TDATA	1	892.63	.	1.0000	50.0000	9.3638	DF
WORK.TDATA	1	887.44	.	1.0000	75.0000	9.3638	DF
WORK.TDATA	1	883.79	.	1.0000	100.00	9.3638	DF
WORK.TDATA	1	881.55	.	1.0000	125.00	9.3638	DF
WORK.TDATA	1	880.72	.	1.0000	140.00	9.3638	DF
WORK.TDATA	1	.	.	.	150.00	9.3638	
WORK.TDATA	1	880.07	.	1.0000	160.00	9.3638	DF
WORK.TDATA	1	879.85	.	1.0000	175.00	9.3638	DF
WORK.TDATA	1	.	.	.	200.00	9.3638	
WORK.TDATA	1	880.21	.	1.0000	225.00	9.3638	DF
WORK.TDATA	1	880.80	.	1.0000	250.00	9.3638	DF
WORK.TDATA	1	881.56	.	1.0000	275.00	9.3638	DF
WORK.TDATA	1	882.44	.	1.0000	300.00	9.3638	DF
WORK.TDATA	1	884.41	.	1.0000	350.00	9.3638	DF

DF: P-value based on a chi-square with DF degrees of freedom.
MI: P-value based on a mixture of chi-squares.

The following statements plot the –2 restricted profile log likelihood (Output 44.10.3):

```
proc sgplot data=ct;
   series y=objective x=covp1;
run;
```

Output 44.10.3 –2 Restricted Profile Log Likelihood for Smoothing Variance



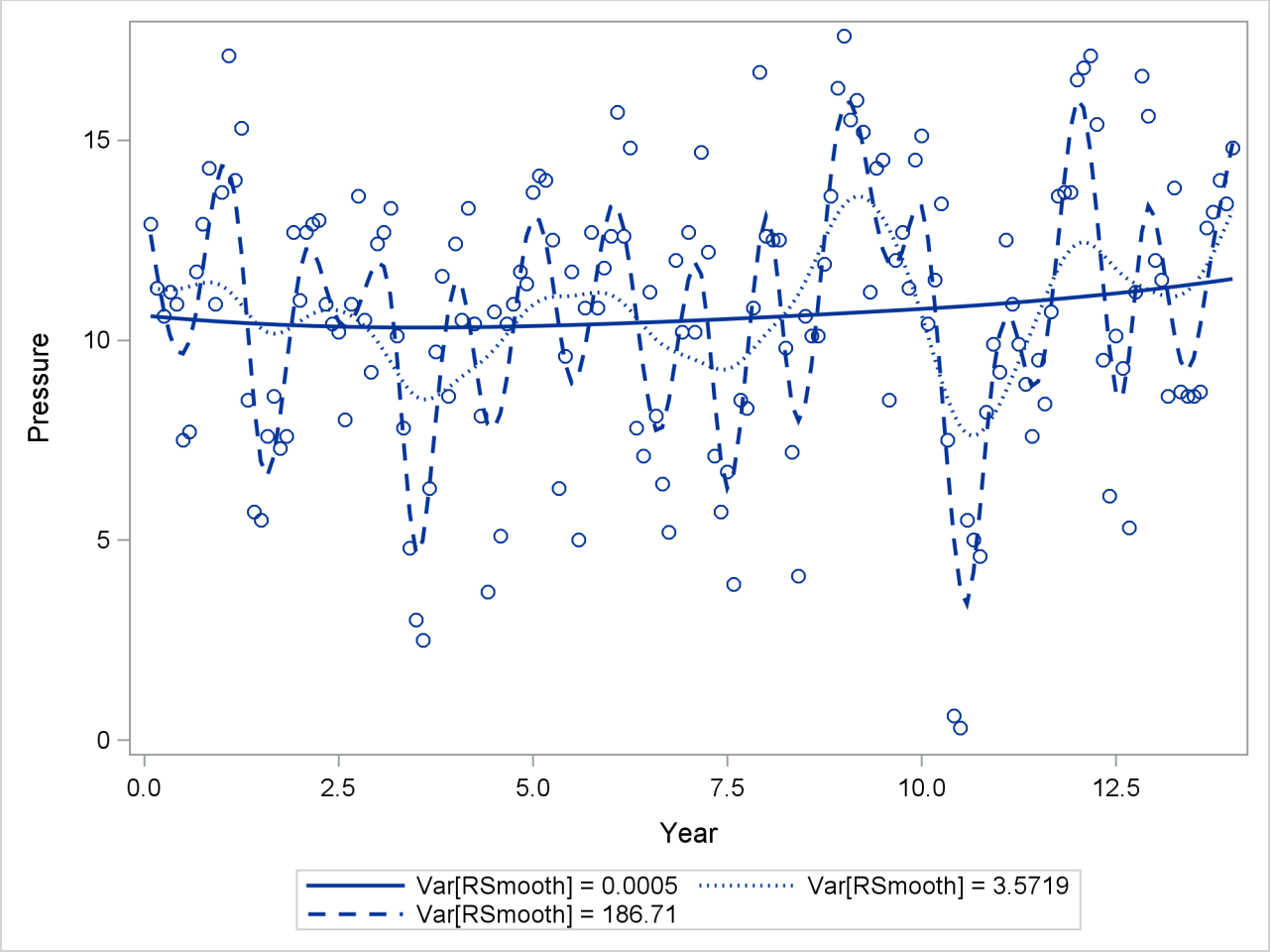
The local minimum at which the optimization stopped is clearly visible, as are a second local minimum near zero and the global minimum near 180.

The observed and predicted pressure differences that correspond to the three minima are shown in Output 44.10.4. These results were produced with the following statements:

```
proc glimmix data=sashelp.enso;
   model pressure = year;
   random year / type=rsmooth knotmethod=equal(50);
   parms (0) (10);
   output out=gmxout1 pred=pred1;
run;
proc glimmix data=sashelp.enso;
   model pressure = year;
   random year / type=rsmooth knotmethod=equal(50);
   output out=gmxout2 pred=pred2;
   parms (2) (10);
run;
proc glimmix data=sashelp.enso;
   model pressure = year;
   random year / type=rsmooth knotmethod=equal(50);
   output out=gmxout3 pred=pred3;
   parms (200) (10);
run;
```

```
data plotthis; merge gmxout1 gmxout2 gmxout3;
run;
proc sgplot data=plotthis;
   scatter x=year y=Pressure;
   series  x=year y=pred1 /
        lineattrs   = (pattern=solid thickness=2)
        legendlabel = "Var[RSmooth] = 0.0005"
        name        = "pred1";
   series  x=year y=pred2 /
        lineattrs   = (pattern=dot thickness=2)
        legendlabel = "Var[RSmooth] = 3.5719"
        name        = "pred2";
   series  x=year y=pred3 /
        lineattrs   = (pattern=dash thickness=2)
        legendlabel = "Var[RSmooth] = 186.71"
        name        = "pred3";
   keylegend "pred1" "pred2" "pred3" / across=2;
run;
```

Output 44.10.4 Observed and Predicted Pressure Differences



The one-year cycle ($\widehat{\sigma}^2_r = 186.71$) and the El Niño cycle ($\widehat{\sigma}^2_r = 3.5719$) are clearly visible. Notice that a larger smoother variance results in larger BLUPs and hence larger adjustments to the fixed-effects model. A

large smoother variance thus results in a more wiggly fit. The third local minimum at $\widehat{\sigma}_r^2 = 0.0005$ applies only very small adjustments to the linear regression between pressure and time, creating slight curvature.

Example 44.11: Maximum Likelihood in Proportional Odds Model with Random Effects

The data for this example are taken from Gilmour, Anderson, and Rae (1987) and concern the foot shape of 2,513 lambs that represent 34 sires. The foot shape of the animals was scored in three ordered categories. The following DATA step lists the data in multivariate form, where each observation corresponds to a sire and contains the outcomes for the three response categories in the variables k1, k2, and k3. For example, for the first sire the first foot shape category was observed for 52 of its offspring, foot shape category 2 was observed for 25 lambs, and none of its offspring was rated in foot shape category 3. The variables yr, b1, b2, and b3 represent contrasts of fixed effects.

```
data foot_mv;
   input yr b1 b2 b3 k1 k2 k3;
   sire = _n_;
   datalines;
  1  1  0  0  52 25  0
  1  1  0  0  49 17  1
  1  1  0  0  50 13  1
  1  1  0  0  42  9  0
  1  1  0  0  74 15  0
  1  1  0  0  54  8  0
  1  1  0  0  96 12  0
  1 -1  1  0  57 52  9
  1 -1  1  0  55 27  5
  1 -1  1  0  70 36  4
  1 -1  1  0  70 37  3
  1 -1  1  0  82 21  1
  1 -1  1  0  75 19  0
  1 -1 -1  0  17 12 10
  1 -1 -1  0  13 23  3
  1 -1 -1  0  21 17  3
 -1  0  0  1  37 41 23
 -1  0  0  1  47 24 12
 -1  0  0  1  46 25  9
 -1  0  0  1  79 32 11
 -1  0  0  1  50 23  5
 -1  0  0  1  63 18  8
 -1  0  0 -1  30 20  9
 -1  0  0 -1  31 33  3
 -1  0  0 -1  28 18  4
 -1  0  0 -1  42 27  4
 -1  0  0 -1  35 22  2
 -1  0  0 -1  33 18  3
 -1  0  0 -1  35 17  4
 -1  0  0 -1  26 13  2
 -1  0  0 -1  37 15  2
 -1  0  0 -1  36 14  1
 -1  0  0 -1  63 20  3
 -1  0  0 -1  41  8  1
;
```

In order to analyze these data as multinomial data with PROC GLIMMIX, the data need to be arranged in univariate form. The following DATA step creates three observations from each record in data set foot_mv and stores the category counts in the variable count:

```
data footshape; set foot_mv;
   array k{3};
   do Shape = 1 to 3;
      count = k{Shape};
      output;
   end;
   drop k:;
run;
```

Because the sires were selected at random, a model for the three-category response with fixed regression effects for yr, b1–b3, and with random sire effects is considered. Because the response categories are ordered, a proportional odds model is chosen (McCullagh 1980). Gilmour, Anderson, and Rae (1987) consider various analyses for these data. The following GLIMMIX statements fit a model with probit link for the cumulative probabilities by maximum likelihood where the marginal log likelihood is approximated by adaptive quadrature:

```
proc glimmix data=footshape method=quad;
   class sire;
   model Shape = yr b1 b2 b3 / s link=cumprobit dist=multinomial;
   random int / sub=sire s cl;
   ods output Solutionr=solr;
   freq count;
run;
```

The number of observations that share a particular response and covariate pattern (variable count) is used in the FREQ statement. The S and CL options request solutions for the sire effects. These are output to the data set solr for plotting.

The "Model Information" table shows that the parameters are estimated by maximum likelihood and that the marginal likelihood is approximated by Gauss-Hermite quadrature (Output 44.11.1).

Output 44.11.1 Model and Data Information

The GLIMMIX Procedure

Model Information	
Data Set	WORK.FOOTSHAPE
Response Variable	Shape
Response Distribution	Multinomial (ordered)
Link Function	Cumulative Probit
Variance Function	Default
Frequency Variable	count
Variance Matrix Blocked By	sire
Estimation Technique	Maximum Likelihood
Likelihood Approximation	Gauss-Hermite Quadrature
Degrees of Freedom Method	Containment

Output 44.11.1 *continued*

Number of Observations Read	102
Number of Observations Used	96
Sum of Frequencies Read	2513
Sum of Frequencies Used	2513

Response Profile		
Ordered Value	**Shape**	**Total Frequency**
1	1	1636
2	2	731
3	3	146

The GLIMMIX procedure is modeling the probabilities of levels of Shape having lower Ordered Values in the Response Profile table.

The distribution of the data is multinomial with ordered categories. The ordering is implied by the choice of a link function for the cumulative probabilities. Because a frequency variable is specified, the number of observations as well as the number of frequencies is displayed. Observations with zero frequency—that is, foot shape categories that were not observed for a particular sire are not used in the analysis. The "Response Profile Table" shows the ordering of the response variable and gives a breakdown of the frequencies by category.

Output 44.11.2 Information about the Size of the Optimization Problem

Dimensions	
G-side Cov. Parameters	1
Columns in X	6
Columns in Z per Subject	1
Subjects (Blocks in V)	34
Max Obs per Subject	3

Optimization Information	
Optimization Technique	Dual Quasi-Newton
Parameters in Optimization	7
Lower Boundaries	1
Upper Boundaries	0
Fixed Effects	Not Profiled
Starting From	GLM estimates
Quadrature Points	1

With METHOD=QUAD, the "Dimensions" and "Optimization Information" tables are particularly important, because for this estimation methods both fixed effects and covariance parameters participate in the optimization (Output 44.11.2). For GLM models the optimization involves the fixed effects and possibly a single scale parameter. For mixed models the fixed effects are typically profiled from the optimization. Laplace and quadrature estimations are exceptions to these rules. Consequently, there are seven parameters in this optimization, corresponding to six fixed effects and one variance component. The variance component has a lower bound of 0. Also, because the fixed effects are part of the optimizations, PROC GLIMMIX initially

performs a few GLM iterations to obtain starting values for the fixed effects. You can control the number of initial iterations with the INITITER= option in the PROC GLIMMIX statement.

The last entry in the "Optimization Information" table shows that—at the starting values—PROC GLIMMIX determined that a single quadrature point is sufficient to approximate the marginal log likelihood with the required accuracy. This approximation is thus identical to the Laplace method that is available with METHOD=LAPLACE.

For METHOD=LAPLACE and METHOD=QUAD, the GLIMMIX procedure produces fit statistics based on the conditional and marginal distribution (Output 44.11.3). Within the limits of the numeric likelihood approximation, the information criteria shown in the "Fit Statistics" table can be used to compare models, and the –2 log likelihood can be used to compare among nested models (nested with respect to fixed effects and/or the covariance parameters).

Output 44.11.3 Marginal and Conditional Fit Statistics

Fit Statistics	
-2 Log Likelihood	3870.12
AIC (smaller is better)	3884.12
AICC (smaller is better)	3884.17
BIC (smaller is better)	3894.81
CAIC (smaller is better)	3901.81
HQIC (smaller is better)	3887.76

Fit Statistics for Conditional Distribution	
-2 log L(Shape | r. effects)	3807.62

The variance of the sire effect is estimated as 0.04849 with estimated asymptotic standard error of 0.01673 (Output 44.11.4). Based on the magnitude of the estimate relative to the standard error, one might conclude that there is significant sire-to-sire variability. Because parameter estimation is based on maximum likelihood, a formal test of the hypothesis of no sire variability is possible. The category cutoffs for the cumulative probabilities are 0.3781 and 1.6435. Except for b3, all fixed effects contrasts are significant.

Output 44.11.4 Parameter Estimates

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error
Intercept	sire	0.04849	0.01673

Solutions for Fixed Effects

Effect	Shape	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept	1	0.3781	0.04907	29	7.71	<.0001
Intercept	2	1.6435	0.05930	29	27.72	<.0001
yr		0.1422	0.04834	2478	2.94	0.0033
b1		0.3781	0.07154	2478	5.28	<.0001
b2		0.3157	0.09709	2478	3.25	0.0012
b3		-0.09887	0.06508	2478	-1.52	0.1289

A likelihood ratio test for the sire variability can be carried out by adding a COVTEST statement to the PROC GLIMMIX statements (Output 44.11.5):

```
ods select FitStatistics CovParms Covtests;
proc glimmix data=footshape method=quad;
   class sire;
   model Shape = yr b1 b2 b3 / link=cumprobit dist=multinomial;
   random int / sub=sire;
   covtest GLM;
   freq count;
run;
```

The statement

```
covtest GLM;
```

compares the fitted model to a generalized linear model for independent data by removing the sire variance component from the model. Equivalently, you can specify

```
covtest 0;
```

which compares the fitted model against one where the sire variance is fixed at zero.

Output 44.11.5 Likelihood Ratio Test for Sire Variance

The GLIMMIX Procedure

Fit Statistics	
-2 Log Likelihood	3870.12
AIC (smaller is better)	3884.12
AICC (smaller is better)	3884.17
BIC (smaller is better)	3894.81
CAIC (smaller is better)	3901.81
HQIC (smaller is better)	3887.76

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error
Intercept	sire	0.04849	0.01673

Tests of Covariance Parameters Based on the Likelihood

Label	DF	-2 Log Like	ChiSq	Pr > ChiSq	Note
Independence	1	3915.29	45.17	<.0001	MI

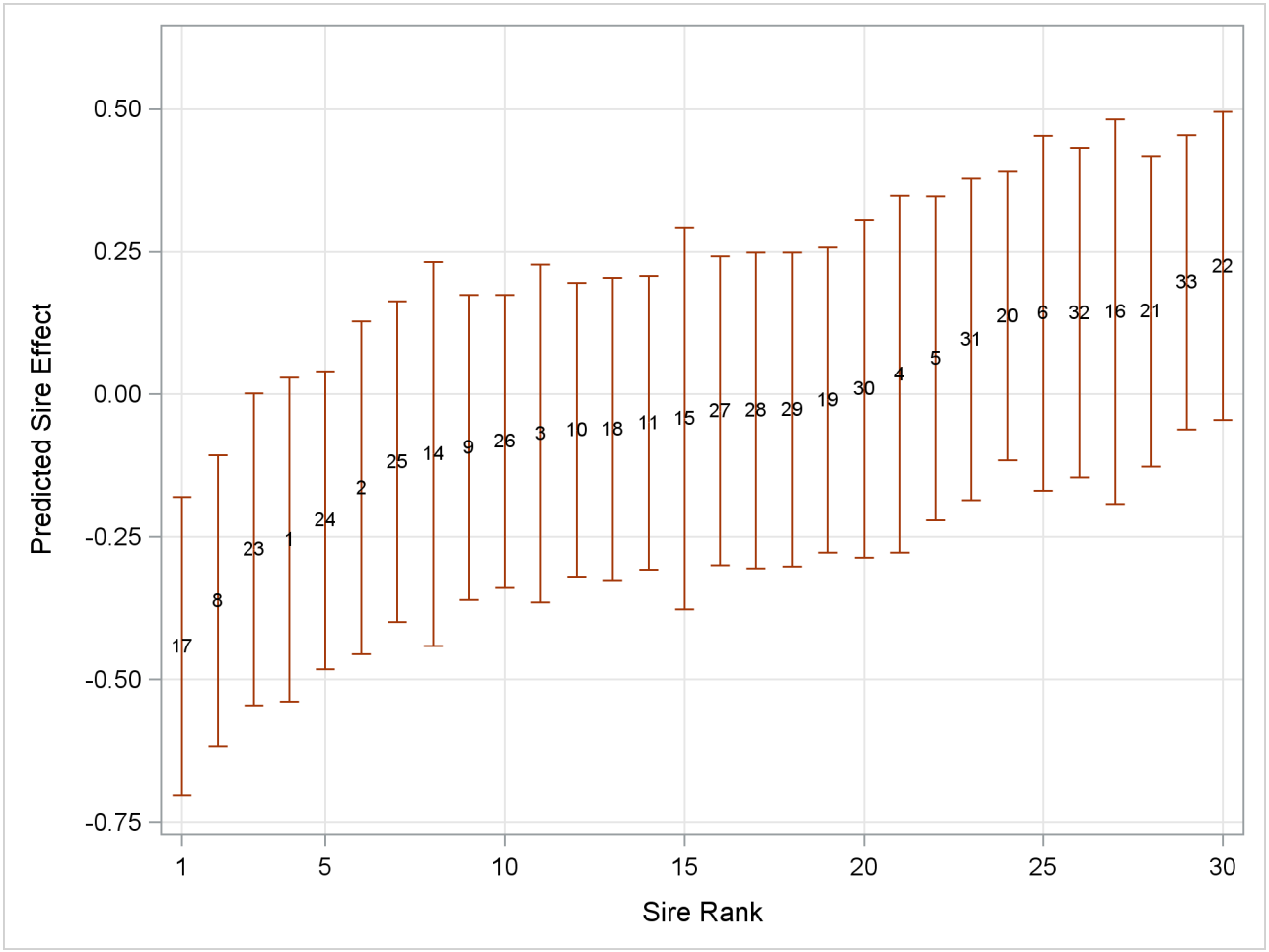
MI: P-value based on a mixture of chi-squares.

The –2 Log Likelihood in the reduced model without the sire effect is 3915.29. Compared to the corresponding marginal fit statistic in the full model (3870.12), this results in a chi-square statistic of 45.17. Because the variance component for the sire effect has a natural lower bound of zero, PROC GLIMMIX performs the likelihood ratio test as a one-sided test. As indicated by the note, the *p*-value for this test is computed from a mixture of chi-square distributions, applying the results of Self and Liang (1987). There is significant evidence that the model without sire random effects does not fit the data as well.

In studies of heritability, one is often interested to rank individuals according to some measure of "breeding value." The following statements display the empirical Bayes estimates of the sire effects from ML estimation by quadrature along with prediction standard error bars (Output 44.11.6):

```
proc sort data=solr;
   by Estimate;
run;
data solr; set solr;
   length sire $2;
   obs  = _n_;
   sire = left(substr(Subject,6,2));
run;
proc sgplot data=solr;
   scatter x=obs y=estimate /
         markerchar  = sire
         yerrorupper = upper
         yerrorlower = lower;
   xaxis grid label='Sire Rank' values=(1 5 10 15 20 25 30);
   yaxis grid label='Predicted Sire Effect';
run;
```

Output 44.11.6 Ranked Predicted Sire Effects and Prediction Standard Errors



Example 44.12: Fitting a Marginal (GEE-Type) Model

A marginal GEE-type model for clustered data is a model for correlated data that is specified through a mean function, a variance function, and a "working" covariance structure. Because the assumed covariance structure can be wrong, the covariance matrix of the parameter estimates is not based on the model alone. Rather, one of the empirical ("sandwich") estimators is used to make inferences robust against the choice of working covariance structure. PROC GLIMMIX can fit marginal models by using R-side random effects and drawing on the distributional specification in the MODEL statement to derive the link and variance functions. The EMPIRICAL= option in the PROC GLIMMIX statement enables you to choose one of a number of empirical covariance estimators.

The data for this example are from Thall and Vail (1990) and reflect the number of seizures of patients suffering from epileptic episodes. After an eight-week period without treatment, patients were observed four times in two-week intervals during which they received a placebo or the drug Progabide in addition to other therapy. These data are also analyzed in Example 43.7 of Chapter 43, "The GENMOD Procedure." The following DATA step creates the data set seizures. The variable id identifies the subjects in the study, and the variable trt identifies whether a subject received the placebo (trt = 0) or the drug Progabide (trt = 1). The variable x1 takes on value 0 for the baseline measurement and 1 otherwise.

```
data seizures;
   array c{5};
   input id trt c1-c5;
   do i=1 to 5;
      x1    = (i > 1);
      ltime = (i=1)*log(8) + (i ne 1)*log(2);
      cnt   = c{i};
      output;
   end;
   keep id cnt x1 trt ltime;
   datalines;
101 1  76 11 14  9  8
102 1  38  8  7  9  4
103 1  19  0  4  3  0
104 0  11  5  3  3  3
106 0  11  3  5  3  3
107 0   6  2  4  0  5
108 1  10  3  6  1  3
110 1  19  2  6  7  4
111 1  24  4  3  1  3
112 1  31 22 17 19 16
113 1  14  5  4  7  4
114 0   8  4  4  1  4
116 0  66  7 18  9 21
117 1  11  2  4  0  4
118 0  27  5  2  8  7
121 1  67  3  7  7  7
122 1  41  4 18  2  5
123 0  12  6  4  0  2
124 1   7  2  1  1  0
126 0  52 40 20 23 12
128 1  22  0  2  4  0
```

```
129 1  13  5  4  0  3
130 0  23  5  6  6  5
135 0  10 14 13  6  0
137 1  46 11 14 25 15
139 1  36 10  5  3  8
141 0  52 26 12  6 22
143 1  38 19  7  6  7
145 0  33 12  6  8  4
147 1   7  1  1  2  3
201 0  18  4  4  6  2
202 0  42  7  9 12 14
203 1  36  6 10  8  8
204 1  11  2  1  0  0
205 0  87 16 24 10  9
206 0  50 11  0  0  5
208 1  22  4  3  2  4
209 1  41  8  6  5  7
210 0  18  0  0  3  3
211 1  32  1  3  1  5
213 0 111 37 29 28 29
214 1  56 18 11 28 13
215 0  18  3  5  2  5
217 0  20  3  0  6  7
218 1  24  6  3  4  0
219 0  12  3  4  3  4
220 0   9  3  4  3  4
221 1  16  3  5  4  3
222 0  17  2  3  3  5
225 1  22  1 23 19  8
226 0  28  8 12  2  8
227 0  55 18 24 76 25
228 1  25  2  3  0  1
230 0   9  2  1  2  1
232 1  13  0  0  0  0
234 0  10  3  1  4  2
236 1  12  1  4  3  2
238 0  47 13 15 13 12
;
```

The model fit initially with the following PROC GLIMMIX statements is a Poisson generalized linear model with effects for an intercept, the baseline measurement, the treatment, and their interaction:

```
proc glimmix data=seizures;
   model cnt = x1 trt x1*trt / dist=poisson offset=ltime
                               ddfm=none s;
run;
```

The DDFM=NONE option is chosen in the MODEL statement to produce chi-square and z tests instead of F and t tests.

Because the initial pretreatment time period is four times as long as the subsequent measurement intervals, an offset variable is used to standardize the counts. If Y_{ij} denotes the number of seizures of subject i in time interval j of length t_j, then Y_{ij}/t_j is the number of seizures per time unit. Modeling the average number per time unit with a log link leads to $\log\{\mathrm{E}[Y_{ij}/t_j]\} = \mathbf{x}'\boldsymbol{\beta}$ or $\log\{\mathrm{E}[Y_{ij}]\} = \mathbf{x}'\boldsymbol{\beta} + \log\{t_j\}$. The logarithm of

time (variable ltime) thus serves as an offset. Suppose that β_0 denotes the intercept, β_1 the effect of x1, and β_2 the effect of trt. Then $\exp\{\beta_0\}$ is the expected number of seizures per week in the placebo group at baseline. The corresponding numbers in the treatment group are $\exp\{\beta_0 + \beta_2\}$ at baseline and $\exp\{\beta_0 + \beta_1 + \beta_2\}$ for postbaseline visits.

The "Model Information" table shows that the parameters in this Poisson model are estimated by maximum likelihood (Output 44.12.1). In addition to the default link and variance function, the variable ltime is used as an offset.

Output 44.12.1 Model Information in Poisson GLM

The GLIMMIX Procedure

Model Information	
Data Set	WORK.SEIZURES
Response Variable	cnt
Response Distribution	Poisson
Link Function	Log
Variance Function	Default
Offset Variable	ltime
Variance Matrix	Diagonal
Estimation Technique	Maximum Likelihood
Degrees of Freedom Method	None

Fit statistics and parameter estimates are shown in Output 44.12.2.

Output 44.12.2 Results from Fitting Poisson GLM

Fit Statistics	
-2 Log Likelihood	3442.66
AIC (smaller is better)	3450.66
AICC (smaller is better)	3450.80
BIC (smaller is better)	3465.34
CAIC (smaller is better)	3469.34
HQIC (smaller is better)	3456.54
Pearson Chi-Square	3015.16
Pearson Chi-Square / DF	10.54

Parameter Estimates

Effect	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept	1.3476	0.03406	Infty	39.57	<.0001
x1	0.1108	0.04689	Infty	2.36	0.0181
trt	-0.1080	0.04865	Infty	-2.22	0.0264
x1*trt	-0.3016	0.06975	Infty	-4.32	<.0001

Because this is a generalized linear model, the large value for the ratio of the Pearson chi-square statistic and its degrees of freedom is indicative of a model shortcoming. The data are considerably more dispersed than is expected under a Poisson model. There could be many reasons for this overdispersion—for example, a misspecified mean model, data that might not be Poisson distributed, an incorrect variance function,

and correlations among the observations. Because these data are repeated measurements, the presence of correlations among the observations from the same subject is a likely contributor to the overdispersion.

The following PROC GLIMMIX statements fit a marginal model with correlations. The model is a marginal one, because no G-side random effects are specified on which the distribution could be conditioned. The choice of the id variable as the SUBJECT effect indicates that observations from different IDs are uncorrelated. Observations from the same ID are assumed to follow a compound symmetry (equicorrelation) model. The EMPIRICAL option in the PROC GLIMMIX statement requests the classical sandwich estimator as the covariance estimator for the fixed effects:

```
proc glimmix data=seizures empirical;
   class id;
   model cnt = x1 trt x1*trt / dist=poisson offset=ltime
                               ddfm=none covb s;
   random _residual_ / subject=id type=cs vcorr;
run;
```

The "Model Information" table shows that the parameters are now estimated by residual pseudo-likelihood (compare Output 44.12.3 and Output 44.12.1). And in this fact lies the main difference between fitting marginal models with PROC GLIMMIX and with GEE methods as per Liang and Zeger (1986), where parameters of the working correlation matrix are estimated by the method of moments.

Output 44.12.3 Model Information in Marginal Model

The GLIMMIX Procedure

Model Information	
Data Set	WORK.SEIZURES
Response Variable	cnt
Response Distribution	Poisson
Link Function	Log
Variance Function	Default
Offset Variable	ltime
Variance Matrix Blocked By	id
Estimation Technique	Residual PL
Degrees of Freedom Method	None
Fixed Effects SE Adjustment	Sandwich - Classical

According to the compound symmetry model, there is substantial correlation among the observations from the same subject (Output 44.12.4).

Output 44.12.4 Covariance Parameter Estimates and Correlation Matrix

Estimated V Correlation Matrix for id 101					
Row	**Col1**	**Col2**	**Col3**	**Col4**	**Col5**
1	1.0000	0.6055	0.6055	0.6055	0.6055
2	0.6055	1.0000	0.6055	0.6055	0.6055
3	0.6055	0.6055	1.0000	0.6055	0.6055
4	0.6055	0.6055	0.6055	1.0000	0.6055
5	0.6055	0.6055	0.6055	0.6055	1.0000

Output 44.12.4 *continued*

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error
CS	id	6.4653	1.3833
Residual		4.2128	0.3928

The parameter estimates in Output 44.12.5 are the same as in the Poisson generalized linear model (Output 44.12.2), because of the balance in these data. The standard errors have increased substantially, however, by taking into account the correlations among the observations.

Output 44.12.5 GEE-Type Inference for Fixed Effects

Solutions for Fixed Effects

Effect	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept	1.3476	0.1574	Infty	8.56	<.0001
x1	0.1108	0.1161	Infty	0.95	0.3399
trt	-0.1080	0.1937	Infty	-0.56	0.5770
x1*trt	-0.3016	0.1712	Infty	-1.76	0.0781

Empirical Covariance Matrix for Fixed Effects

Effect	Row	Col1	Col2	Col3	Col4
Intercept	**1**	0.02476	-0.00115	-0.02476	0.001152
x1	**2**	-0.00115	0.01348	0.001152	-0.01348
trt	**3**	-0.02476	0.001152	0.03751	-0.00300
x1*trt	**4**	0.001152	-0.01348	-0.00300	0.02931

Example 44.13: Response Surface Comparisons with Multiplicity Adjustments

Koch et al. (1990) present data for a multicenter clinical trial testing the efficacy of a respiratory drug in patients with respiratory disease. Within each of two centers, patients were randomly assigned to a placebo (P) or an active (A) treatment. Prior to treatment and at four follow-up visits, patient status was recorded in one of five ordered categories (0=terrible, 1=poor, . . . , 4=excellent). The following DATA step creates the SAS data set clinical for this study.

```
data Clinical;
   do Center =   1,  2;
      do Gender = 'F','M';
         do Drug   = 'A','P';
            input nPatient @@;
            do iPatient = 1 to nPatient;
               input ID Age (t0-t4) (1.) @@;
               output;
            end;
         end;
      end;
   end;
   datalines;
```

```
 2  53 32 12242  18 47 22344
 5   5 13 44444  19 31 21022  25 35 10000  28 36 23322
    36 45 22221
25  54 11 44442  12 14 23332  51 15 02333  20 20 33231
    16 22 12223  50 22 21344   3 23 33443  32 23 23444
    56 25 23323  35 26 12232  26 26 22222  21 26 24142
     8 28 12212  30 28 00121  33 30 33442  11 30 34443
    42 31 12311   9 31 33444  37 31 02321  23 32 34433
     6 34 11211  22 46 43434  24 48 23202  38 50 22222
    48 57 33434
24  43 13 34444  41 14 22123  34 15 22332  29 19 23300
    15 20 44444  13 23 33111  27 23 44244  55 24 34443
    17 25 11222  45 26 24243  40 26 12122  44 27 12212
    49 27 33433  39 23 21111   2 28 20000  14 30 10000
    31 37 10000  10 37 32332   7 43 23244  52 43 11132
     4 44 34342   1 46 22222  46 49 22222  47 63 22222
 4  30 37 13444  52 39 23444  23 60 44334  54 63 44444
12  28 31 34444   5 32 32234  21 36 33213  50 38 12000
     1 39 12112  48 39 32300   7 44 34444  38 47 23323
     8 48 22100  11 48 22222   4 51 34244  17 58 14220
23  12 13 44444  10 14 14444  27 19 33233  47 20 24443
    16 20 21100  29 21 33444  20 24 44444  25 25 34331
    15 25 34433   2 25 22444   9 26 23444  49 28 23221
    55 31 44444  43 34 24424  26 35 44444  14 37 43224
    36 41 34434  51 43 33442  37 52 12122  19 55 44444
    32 55 22331   3 58 44444  53 68 23334
16  39 11 34444  40 14 21232  24 15 32233  41 15 43334
    33 19 42233  34 20 32444  13 20 14444  45 33 33323
    22 36 24334  18 38 43000  35 42 32222  44 43 21000
     6 45 34212  46 48 44000  31 52 23434  42 66 33344
;
```

Westfall and Tobias (2007) define as the measure of efficacy the average of the ratings at the final two visits and model this average as a function of drug, baseline assessment score, and age. Hence, in their model, the expected efficacy for drug $d \in A$, P can be written as

$$\mathrm{E}[Y_d] = \beta_{0d} + \beta_{1d}t + \beta_{2d}a$$

where t is the baseline (pretreatment) assessment score and a is the patient's age at baseline. The age range for these data extends from 11 to 68 years. Suppose that the scientific question of interest is the comparison of the two response surfaces at a set of values $S_t \times S_a = \{0, 1, 2, 3, 4\} \times S_a$. In other words, you want to know for which values of the covariates the average response differs significantly between the treatment group and the placebo group. If the set of ages of interest is $\{10, 13, 16, \cdots, 70\}$, then this involves $5 \times 21 = 105$ comparisons, a massive multiple testing problem. The large number of comparisons and the fact that the set S_a is chosen somewhat arbitrarily require the application of multiplicity corrections in order to protect the familywise Type I error across the comparisons.

When testing hypotheses that have logical restrictions, the power of multiplicity corrected tests can be increased by taking the restrictions into account. Logical restrictions exist, for example, when not all hypotheses in a set can be simultaneously true. Westfall and Tobias (2007) extend the truncated closed testing procedure (TCTP) of Royen (1989) for pairwise comparisons in ANOVA to general contrasts. Their

work is also an extension of the S2 method of Shaffer (1986); see also Westfall (1997). These methods are all *monotonic* in the (unadjusted) p-values of the individual tests, in the sense that if $p_j < p_i$ then the multiple test will never retain H_j while rejecting H_i. In terms of multiplicity-adjusted p-values $\tilde{p}_j$, monotonicity means that if $p_j < p_i$, then $\tilde{p}_j < \tilde{p}_i$.

Analysis as Normal Data with Averaged Endpoints

In order to apply the extended TCTP procedure of Westfall and Tobias (2007) to the problem of comparing response surfaces in the clinical trial, the following convenience macro is helpful to generate the comparisons for the ESTIMATE statement in PROC GLIMMIX:

```
%macro Contrast(from,to,byA,byT);
   %let nCmp = 0;
   %do age = &from %to &to %by &byA;
      %do t0  =  0 %to  4 %by &byT;
         %let nCmp = %eval(&nCmp+1);
      %end;
   %end;
   %let iCmp = 0;
   %do age = &from %to &to %by &byA;
      %do t0  =  0 %to  4 %by &byT;
         %let iCmp = %eval(&iCmp+1);
         "%trim(%left(&age)) %trim(%left(&t0))"
           drug     1    -1
           drug*age &age -&age
           drug*t0  &t0  -&t0
         %if (&icmp < &nCmp) %then %do; , %end;
      %end;
   %end;
%mend;
```

The following GLIMMIX statements fit the model to the data and compute the 105 contrasts that compare the placebo to the active response at 105 points in the two-dimensional regressor space:

```
proc glimmix data=clinical;
   t = (t3+t4)/2;
   class drug;
   model t = drug t0 age drug*age drug*t0;
   estimate %contrast(10,70,3,1)
              / adjust=simulate(seed=1)
                stepdown(type=logical);
   ods output Estimates=EstStepDown;
run;
```

Note that only a single ESTIMATE statement is used. Each of the 105 comparisons is one comparison in the multirow statement. The ADJUST option in the ESTIMATE statement requests multiplicity-adjusted p-values. The extended TCTP method is applied by specifying the STEPDOWN(TYPE=LOGICAL) option to compute step-down-adjusted p-values where logical constraints among the hypotheses are taken into account. The results from the ESTIMATE statement are saved to a data set for subsequent processing. Note also that the response, the average of the ratings at the final two visits, is computed with programming statements in PROC GLIMMIX.

The following statements print the 20 most significant estimated differences (Output 44.13.1):

```
proc sort data=EstStepDown;
   by Probt;
run;
proc print data=EstStepDown(obs=20);
   var Label Estimate StdErr Probt AdjP;
run;
```

Output 44.13.1 The First 20 Observations of the Estimates Data Set

Obs	Label	Estimate	StdErr	Probt	Adjp
1	37 2	0.8310	0.2387	0.0007	0.0071
2	40 2	0.8813	0.2553	0.0008	0.0071
3	34 2	0.7806	0.2312	0.0010	0.0071
4	43 2	0.9316	0.2794	0.0012	0.0071
5	46 2	0.9819	0.3093	0.0020	0.0071
6	31 2	0.7303	0.2338	0.0023	0.0081
7	49 2	1.0322	0.3434	0.0033	0.0107
8	52 2	1.0825	0.3807	0.0054	0.0167
9	40 3	0.7755	0.2756	0.0059	0.0200
10	37 3	0.7252	0.2602	0.0063	0.0201
11	43 3	0.8258	0.2982	0.0066	0.0201
12	28 2	0.6800	0.2461	0.0068	0.0215
13	55 2	1.1329	0.4202	0.0082	0.0239
14	46 3	0.8761	0.3265	0.0085	0.0239
15	34 3	0.6749	0.2532	0.0089	0.0257
16	43 1	1.0374	0.3991	0.0107	0.0329
17	46 1	1.0877	0.4205	0.0111	0.0329
18	49 3	0.9264	0.3591	0.0113	0.0329
19	40 1	0.9871	0.3827	0.0113	0.0329
20	58 2	1.1832	0.4615	0.0118	0.0329

Notice that the adjusted *p*-values (Adjp) are larger than the unadjusted *p*-values, as expected. Also notice that several comparisons share the same adjusted *p*-values. This is a result of the monotonicity of the extended TCTP method.

In order to compare the step-down-adjusted *p*-values to adjusted *p*-values that do not use step-down methods, replace the ESTIMATE statement in the previous statements with the following:

```
estimate %contrast2(10,70,3,1) / adjust=simulate(seed=1);
ods output Estimates=EstAdjust;
```

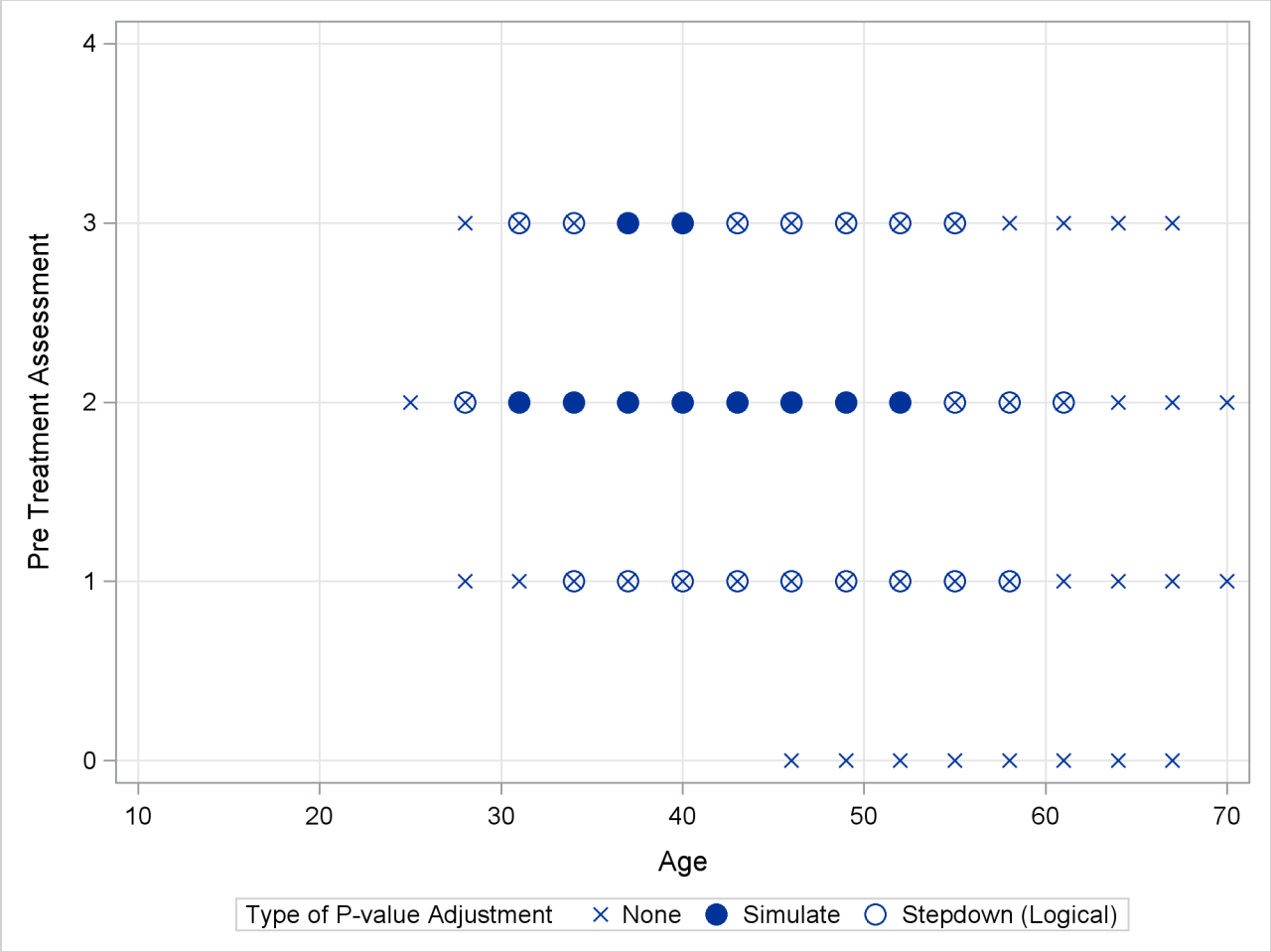
The following GLIMMIX invocations create output data sets named EstAdjust and EstUnAdjust that contain (non-step-down-) adjusted and unadjusted *p*-values:

```
proc glimmix data=clinical;
   t = (t3+t4)/2;
   class drug;
   model t = drug t0 age drug*age drug*t0;
   estimate %contrast(10,70,3,1)
             / adjust=simulate(seed=1);
```

```
   ods output Estimates=EstAdjust;
run;
proc glimmix data=clinical;
   t = (t3+t4)/2;
   class drug;
   model t = drug t0 age drug*age drug*t0;
   estimate %contrast(10,70,3,1);
   ods output Estimates=EstUnAdjust;
run;
```

Output 44.13.2 shows a comparison of the significant comparisons ($p < 0.05$) based on unadjusted, adjusted, and step-down (TCTP) adjusted p-values. Clearly, the unadjusted results indicate the most significant results, but without protecting the Type I error rate for the group of tests. The adjusted p-values (filled circles) lead to a much smaller region in which the response surfaces between treatment and placebo are significantly different. The increased power of the TCTP procedure (open circles) over the standard multiplicity adjustment—without sacrificing Type I error protection—can be seen in the considerably larger region covered by the open circles.

Output 44.13.2 Comparison of Significance Regions



Ordinal Repeated Measure Analysis

The outcome variable in this clinical trial is an ordinal rating of patients in categories 0=terrible, 1=poor, 2=fair, 3=good, and 4=excellent. Furthermore, the observations from repeat visits for the same patients are likely correlated. The previous analysis removes the repeated measures aspect by defining efficacy as the average score at the final two visits. These averages are not normally distributed, however. The response surfaces for the two study arms can also be compared based on a model for ordinal data that takes correlation into account through random effects. Keeping with the theme of the previous analysis, the focus here for illustrative purposes is on the final two visits, and the pretreatment assessment score serves as a covariate in the model.

The following DATA step rearranges the data from the third and fourth on-treatment visits in univariate form with separate observations for the visits by patient:

```
data clinical_uv;
   set clinical;
   array time{2} t3-t4;
   do i=1 to 2; rating = time{i}; output; end;
run;
```

The basic model for the analysis is a proportional odds model with cumulative logit link (McCullagh 1980) and $J = 5$ categories. In this model, separate intercepts (cutoffs) are modeled for the first $J - 1 = 4$ cumulative categories and the intercepts are monotonically increasing. This guarantees ordering of the cumulative probabilities and nonnegative category probabilities. Using the same covariate structure as in the previous analysis, the probability to observe a rating in at most category $k \leq 4$ is

$$\Pr(Y_d \leq k) = \frac{1}{1 + \exp\{-\eta_{kd}\}}$$
$$\eta_{kd} = \alpha_k + \beta_{0d} + \beta_{1d} t + \beta_{2d} a$$

Because only the intercepts are dependent on the category, contrasts comparing regression coefficients can be formulated in standard fashion. To accommodate the random and covariance structure of the repeated measures model, a random intercept γ_i is applied to the observations for each patient:

$$\Pr(Y_{id} \leq k) = \frac{1}{1 + \exp\{-\eta_{ikd}\}}$$
$$\eta_{ikd} = \alpha_k + \beta_{0d} + \beta_{1d} t + \beta_{2d} a + \gamma_i$$
$$\gamma_i \sim \text{iid}\, N(0, \sigma^2_\gamma)$$

The shared random effect of the two observations creates a marginal correlation. Note that the random effects do not depend on category.

The following GLIMMIX statements fit this ordinal repeated measures model by maximum likelihood via the Laplace approximation and compute TCTP-adjusted p-values for the 105 estimates:

```
proc glimmix data=clinical_uv method=laplace;
   class center id drug;
   model rating = drug t0 age drug*age drug*t0 /
                  dist=multinomial link=cumlogit;
   random intercept / subject=id(center);
   covtest 0;
   estimate %contrast(10,70,3,1)
             / adjust=simulate(seed=1)
```

```
                  stepdown(type=logical);
   ods output Estimates=EstStepDownMulti;
run;
```

The combination of DIST=MULTINOMIAL and LINK=CUMLOGIT requests the proportional odds model. The SUBJECT= effect nests patient IDs within centers, because patient IDs in the data set clinical are not unique within centers. (Specifying SUBJECT=ID*CENTER would have the same effect.) The COVTEST statement requests a likelihood ratio test for the significance of the random patient effect.

The estimate of the variance component for the random patient effect is substantial (Output 44.13.3), but so is its standard error.

Output 44.13.3 Model and Covariance Parameter Information

The GLIMMIX Procedure

Model Information	
Data Set	WORK.CLINICAL_UV
Response Variable	rating
Response Distribution	Multinomial (ordered)
Link Function	Cumulative Logit
Variance Function	Default
Variance Matrix Blocked By	ID(Center)
Estimation Technique	Maximum Likelihood
Likelihood Approximation	Laplace
Degrees of Freedom Method	Containment

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error
Intercept	ID(Center)	10.3483	3.2599

Tests of Covariance Parameters Based on the Likelihood

Label	DF	-2 Log Like	ChiSq	Pr > ChiSq	Note
Parameter list	1	604.70	57.64	<.0001	MI

MI: P-value based on a mixture of chi-squares.

The likelihood ratio test provides a better picture of the significance of the variance component. The difference in the –2 log likelihoods is 57.6, highly significant even if one does not apply the Self and Liang (1987) correction that halves the *p*-value in this instance.

The results for the 20 most significant estimates are requested with the following statements and shown in Output 44.13.4:

```
proc sort data=EstStepDownMulti;
   by Probt;
run;
proc print data=EstStepDownMulti(obs=20);
   var Label Estimate StdErr Probt AdjP;
run;
```

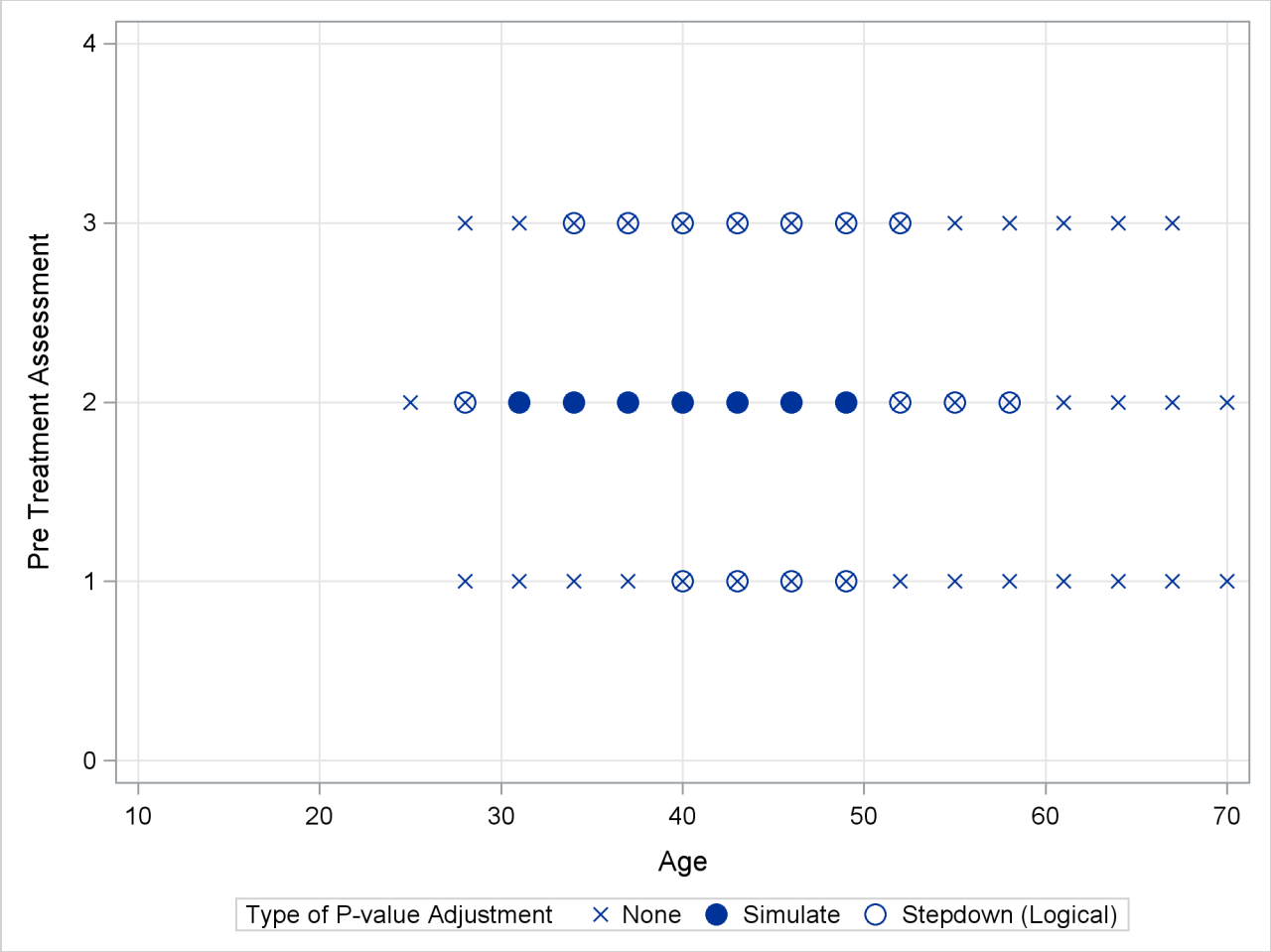
The *p*-values again show the "repeat" pattern corresponding to the monotonicity of the step-down procedure.

Output 44.13.4 The First 20 Estimates in the Ordinal Analysis

Obs	Label	Estimate	StdErr	Probt	Adjp
1	37 2	-2.7224	0.8263	0.0013	0.0133
2	40 2	-2.8857	0.8842	0.0015	0.0133
3	34 2	-2.5590	0.7976	0.0018	0.0133
4	43 2	-3.0491	0.9659	0.0021	0.0133
5	46 2	-3.2124	1.0660	0.0032	0.0133
6	31 2	-2.3957	0.8010	0.0034	0.0133
7	49 2	-3.3758	1.1798	0.0051	0.0164
8	52 2	-3.5391	1.3037	0.0077	0.0236
9	40 3	-2.6263	0.9718	0.0080	0.0267
10	37 3	-2.4630	0.9213	0.0087	0.0267
11	28 2	-2.2323	0.8362	0.0088	0.0278
12	43 3	-2.7897	1.0451	0.0088	0.0278
13	46 3	-2.9530	1.1368	0.0107	0.0291
14	55 2	-3.7025	1.4351	0.0112	0.0324
15	34 3	-2.2996	0.8974	0.0118	0.0337
16	49 3	-3.1164	1.2428	0.0136	0.0344
17	43 1	-3.3085	1.3438	0.0154	0.0448
18	58 2	-3.8658	1.5722	0.0155	0.0448
19	40 1	-3.1451	1.2851	0.0160	0.0448
20	46 1	-3.4718	1.4187	0.0160	0.0448

As previously, the comparisons were also performed with standard p-value adjustment via simulation. Output 44.13.5 displays the components of the regressor space in which the response surfaces differ significantly ($p < 0.05$) between the two treatment arms. As before, the most significant differences occur with unadjusted p-values at the cost of protecting only the individual Type I error rate. The standard multiplicity adjustment has considerably less power than the TCTP adjustment.

Output 44.13.5 Comparison of Significance Regions, Ordinal Analysis



Example 44.14: Generalized Poisson Mixed Model for Overdispersed Count Data

Overdispersion is the condition by which data appear more dispersed than is expected under a reference model. For count data, the reference models are typically based on the binomial or Poisson distributions. Among the many reasons for overdispersion are an incorrect model, an incorrect distributional specification, incorrect variance functions, positive correlation among the observations, and so forth. In short, correcting an overdispersion problem, if it exists, requires the appropriate remedy. Adding an R-side scale parameter to multiply the variance function is not necessarily the adequate correction. For example, Poisson-distributed data appear overdispersed relative to a Poisson model with regressors when an important regressor is omitted.

If the reference model for count data is Poisson, a number of alternative model formulations are available to increase the dispersion. For example, zero-inflated models add a proportion of zeros (usually from a Bernoulli process) to the zeros of a Poisson process. Hurdle models are two-part models where zeros and nonzeros are generated by different stochastic processes. Zero-inflated and hurdle models are described in detail by Cameron and Trivedi (1998) and cannot be fit with the GLIMMIX procedure. See Section 15.5 in Littell et al. (2006) for examples of using the NLMIXED procedure to fit zero-inflated and hurdle models.

An alternative approach is to derive from the reference distribution a probability distribution that exhibits increased dispersion. By mixing a Poisson process with a gamma distribution for the Poisson parameter, for example, the negative binomial distribution results, which is thus overdispersed relative to the Poisson.

Joe and Zhu (2005) show that the generalized Poisson distribution can also be motivated as a Poisson mixture and hence provides an alternative to the negative binomial (NB) distribution. Like the NB, the generalized Poisson distribution has a scale parameter. It is heavier in the tails than the NB distribution and easily reduces to the standard Poisson. Joe and Zhu (2005) discuss further comparisons between these distributions.

The probability mass function of the generalized Poisson is given by

$$p(y) = \frac{\alpha}{y!}(\alpha + \xi y)^{y-1} \exp\{-\alpha - \xi y\}$$

where $y = 0, 1, 2, \cdots, \alpha > 0$, and $0 \leq \xi < 1$ (Joe and Zhu 2005). Notice that for $\xi = 0$ the mass function of the standard Poisson distribution with mean α results. The mean and variance of Y in terms of the parameters α and ξ are given by

$$\mathrm{E}[Y] = \frac{\alpha}{1-\xi} = \mu$$
$$\mathrm{Var}[Y] = \frac{\alpha}{(1-\xi)^3} = \frac{\mu}{(1-\xi)^2}$$

The log likelihood of the generalized Poisson can thus be written in terms of the mean μ and scale parameter ξ as

$$l(\mu, \xi; y) = \log\{\mu(1-\xi)\} + (y-1)\log\{\mu - \xi(\mu - y)\} - (\mu - \xi(\mu - y)) - \log\{\Gamma(y+1)\}$$

The data in the following DATA step are simulated counts. For each of $i = 1, \cdots, 30$ subjects a randomly varying number n_i of observations were drawn from a count regression model with a single covariate and excess zeros (compared to a Poisson distribution).

```
data counts;
   input ni @@;
   sub = _n_;
   do i=1 to ni;
      input x y @@;
      output;
   end;
   datalines;
 1 29 0
 6  2 0 82 5 33 0 15 2 35 0 79 0
19 81 0 18 0 85 0 99 0 20 0 26 2 29 0 91 2 37 0 39 0  9 1 33 0
    3 0 60 0 87 2 80 0 75 0  3 0 63 1
 9 18 0 64 0 80 0  0 0 58 0  7 0 81 0 22 3 50 0
15 91 0  2 1 14 0  5 2 27 1  8 1 95 0 76 0 62 0 26 2  9 0 72 1
   98 0 94 0 23 1
 2 34 0 95 0
18 48 1  5 0 47 0 44 0 27 0 88 0 27 0 68 0 84 0 86 0 44 0 90 0
   63 0 27 0 47 0 25 0 72 0 62 1
13 28 1 31 0 63 0 14 0 74 0 44 0 75 0 65 0 74 1 84 0 57 0 29 0
   41 0
```

```
 9 42 0  8 0 91 0 20 0 23 0 22 0 96 0 83 0 56 0
 3 64 0 64 1 15 0
 4  5 0 73 2 50 1 13 0
 2  0 0 41 0
20 21 0 58 0  5 0 61 1 28 0 71 0 75 1 94 16 51 4 51 2 74 0  1 1
   34 0  7 0 11 0 60 3 31 0 75 0 62 0 54 1
 2 66 1 13 0
 5 83 7 98 1 11 1 28 0 18 0
17 29 5 79 0 39 2 47 2 80 1 19 0 37 0 78 1 26 0 72 1  6 0 50 3
   50 4 97 0 37 2 51 0 45 0
17 47 0 57 0 33 0 47 0  2 0 83 0 74 0 93 0 36 0 53 0 26 0 86 0
    6 0 17 0 30 0 70 1 99 0
 7 91 0 25 1 51 4 20 0 61 1 34 0 33 2
14 60 0 87 0 94 0 29 0 41 0 78 0 50 0 37 0 15 0 39 0 22 0 82 0
   93 0  3 0
16 68 0 26 1 19 0 60 1 93 3 65 0 16 0 79 0 14 0  3 1 90 0 28 3
   82 0 34 0 30 0 81 0
19 48 3 48 1 43 2 54 0 45 9 53 0 14 0 92 5 21 1 20 0 73 0 99 0
   66 0 86 2 63 0 10 0 92 14 44 1 74 0
 8 34 1 44 0 62 0 21 0  7 0 17 0  0 2 49 0
13 11 0 27 2 16 1 12 3 52 1 55 0  2 6 89 5 31 5 28 3 51 5 54 13
   64 0
 9  3 0 36 0 57 0 77 0 41 0 39 0 55 0 57 0 88 1
 7  2 0 80 0 41 1 20 0  2 0 27 0 40 0
18 73 1 66 0 10 0 42 0 22 0 59 9 68 0 34 1 96 0 30 0 13 0 35 0
   51 2 47 0 60 1 55 4 83 3 38 0
17 96 0 40 0 34 0 59 0 12 1 47 0 93 0 50 0 39 0 97 0 19 0 54 0
   11 0 29 0 70 2 87 0 47 0
13 59 0 96 0 47 1 64 0 18 0 30 0 37 0 36 1 69 0 78 1 47 1 86 0
   88 0
15 66 0 45 1 96 1 17 0 91 0  4 0 22 0  5 2 47 0 38 0 80 0  7 1
   38 1 33 0 52 0
12 84 6 60 1 33 1 92 0 38 0  6 0 43 3 13 2 18 0 51 0 50 4 68 0
;
```

The following PROC GLIMMIX statements fit a standard Poisson regression model with random intercepts by maximum likelihood. The marginal likelihood of the data is approximated by adaptive quadrature (METHOD=QUAD).

```
proc glimmix data=counts method=quad;
   class sub;
   model y = x / link=log s dist=poisson;
   random int / subject=sub;
run;
```

Output 44.14.1 displays various informational items about the model and the estimation process.

Output 44.14.1 Poisson: Model and Optimization Information

The GLIMMIX Procedure

Model Information	
Data Set	WORK.COUNTS
Response Variable	y
Response Distribution	Poisson
Link Function	Log
Variance Function	Default
Variance Matrix Blocked By	sub
Estimation Technique	Maximum Likelihood
Likelihood Approximation	Gauss-Hermite Quadrature
Degrees of Freedom Method	Containment

Optimization Information	
Optimization Technique	Dual Quasi-Newton
Parameters in Optimization	3
Lower Boundaries	1
Upper Boundaries	0
Fixed Effects	Not Profiled
Starting From	GLM estimates
Quadrature Points	5

Iteration History

Iteration	Restarts	Evaluations	Objective Function	Change	Max Gradient
0	**0**	4	862.57645728	.	366.7105
1	**0**	5	862.43893582	0.13752147	22.36158
2	**0**	6	854.49131023	7.94762559	28.70814
3	**0**	2	854.47983504	0.01147519	6.036114
4	**0**	4	854.47396189	0.00587315	4.238363
5	**0**	4	854.47006558	0.00389631	0.332454
6	**0**	3	854.47006484	0.00000074	0.003104

The "Model Information" table shows that the parameters are estimated by ML with quadrature. Using the starting values for fixed effects and covariance parameters that the GLIMMIX procedure generates by default, the procedure determined that five quadrature nodes provide a sufficiently accurate approximation of the marginal log likelihood ("Optimization Information" table). The iterative estimation process converges after nine iterations.

The table of conditional fit statistics displays the sum of the independent contributions to the conditional –2 log likelihood (854.47) and the Pearson statistics for the conditional distribution (Output 44.14.2).

Output 44.14.2 Poisson: Fit Statistics and Estimates

Fit Statistics	
-2 Log Likelihood	854.47
AIC (smaller is better)	860.47
AICC (smaller is better)	860.54
BIC (smaller is better)	864.67
CAIC (smaller is better)	867.67
HQIC (smaller is better)	861.81

Fit Statistics for Conditional Distribution	
-2 log L(y | r. effects)	777.90
Pearson Chi-Square	649.58
Pearson Chi-Square / DF	1.97

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error
Intercept	sub	1.1959	0.4334

Solutions for Fixed Effects

Effect	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept	-1.4947	0.2745	29	-5.45	<.0001
x	0.01207	0.002387	299	5.06	<.0001

The departure of the scaled Pearson statistic from 1.0 is fairly pronounced in this case (1.97). If one deems it to far from 1.0, however, the conclusion has to be that the conditional variation is not properly specified. This could be due to an incorrect variance function, for example. The "Solutions for Fixed Effects" table shows the estimates of the slope and intercept in this model along with their standard errors and tests of significance. Note that the slope in this model is highly significant. The variance of the random subject-specific intercepts is estimated as 1.1959.

To fit the generalized Poisson distribution to these data, you cannot draw on the built-in distributions. Instead, the variance function and the log likelihood are computed directly with PROC GLIMMIX programming statements. The CLASS, MODEL, and RANDOM statements in the following PROC GLIMMIX program are as before, except for the omission of the DIST= option in the MODEL statement:

```
proc glimmix data=counts method=quad;
   class sub;
   model y = x / link=log s;
   random int / subject=sub;
   xi = (1 - 1/exp(_phi_));
   _variance_ = _mu_ / (1-xi)/(1-xi);
   if (_mu_=.) or (_linp_ = .) then _logl_ = .;
   else do;
      mustar = _mu_ - xi*(_mu_ - y);
      if (mustar < 1E-12) or (_mu_*(1-xi) < 1e-12) then
         _logl_ = -1E20;
      else do;
         _logl_ = log(_mu_*(1-xi)) + (y-1)*log(mustar) -
                  mustar - lgamma(y+1);
      end;
   end;
run;
```

The assignments to the variables xi and the reserved symbols _VARIANCE_ and _LOGL_ define the variance function and the log likelihood. Because the scale parameter of the generalized Poisson distribution has the range $0 < \xi < 1$, and the scale parameter _PHI_ in the GLIMMIX procedure is bounded only from below (by 0), a reparameterization is applied so that $\phi = 0 \Leftrightarrow \xi = 0$ and ξ approaches 1 as ϕ increases. The statements preceding the calculation of the actual log likelihood are intended to prevent floating-point exceptions and to trap missing values.

Output 44.14.3 displays information about the model and estimation process. The "Model Information" table shows that the distribution is not a built-in distribution and echoes the expression for the user-specified variance function. As in the case of the Poisson model, the GLIMMIX procedure determines that five quadrature points are sufficient for accurate estimation of the marginal log likelihood at the starting values. The estimation process converges after 11 iterations.

Output 44.14.3 Generalized Poisson: Model, Optimization, and Iteration Information

The GLIMMIX Procedure

Model Information	
Data Set	WORK.COUNTS
Response Variable	y
Response Distribution	User specified
Link Function	Log
Variance Function	_mu_ / (1-xi)/(1-xi)
Variance Matrix Blocked By	sub
Estimation Technique	Maximum Likelihood
Likelihood Approximation	Gauss-Hermite Quadrature
Degrees of Freedom Method	Containment

Output 44.14.3 *continued*

Optimization Information	
Optimization Technique	Dual Quasi-Newton
Parameters in Optimization	4
Lower Boundaries	2
Upper Boundaries	0
Fixed Effects	Not Profiled
Starting From	GLM estimates
Quadrature Points	5

Iteration History

Iteration	Restarts	Evaluations	Objective Function	Change	Max Gradient
0	**0**	4	716.12976769	.	161.1184
1	**0**	5	716.07585953	0.05390816	11.88788
2	**0**	4	714.27148068	1.80437884	36.09657
3	**0**	2	711.02643265	3.24504804	108.4615
4	**0**	2	710.26952196	0.75691069	216.9822
5	**0**	2	709.96824991	0.30127205	96.2775
6	**0**	3	709.8419071	0.12634280	19.07487
7	**0**	3	709.83122731	0.01067980	0.649164
8	**0**	3	709.83047646	0.00075085	2.127665
9	**0**	3	709.83046461	0.00001185	0.383319
10	**0**	3	709.83046436	0.00000025	0.010279

The achieved –2 log likelihood is lower than in the Poisson model (compare "Fit Statistics" tables in Output 44.14.4 and Output 44.14.1). The scaled Pearson statistic is now less than 1.0. The fixed slope estimate remains significant at the 5% level, but the test statistics are not as large as in the Poisson model, partly because the generalized Poisson model permits more variation.

Output 44.14.4 Generalized Poisson: Fit Statistics and Estimates

Fit Statistics	
-2 Log Likelihood	709.83
AIC (smaller is better)	717.83
AICC (smaller is better)	717.95
BIC (smaller is better)	723.44
CAIC (smaller is better)	727.44
HQIC (smaller is better)	719.62

Fit Statistics for Conditional Distribution	
-2 log L(y | r. effects)	665.56
Pearson Chi-Square	241.42
Pearson Chi-Square / DF	0.73

Output 44.14.4 *continued*

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error
Intercept	sub	0.5135	0.2400
Scale		0.6401	0.09718

Solutions for Fixed Effects

Effect	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept	-0.7264	0.2749	29	-2.64	0.0131
x	0.003742	0.003537	299	1.06	0.2910

Based on the large difference in the –2 log likelihoods between the Poisson and generalized Poisson models, you conclude that a mixed model based on the latter provides a better fit to these data. From the "Covariance Parameter Estimates" table in Output 44.14.4, you can see that the estimate of the scale parameter is $\widehat{\phi} = 0.6401$ and is considerably larger than 0, taking into account its standard error. The hypothesis $H: \phi = 0$, which articulates that a Poisson model fits the data as well as the generalized Poisson model, can be formally tested with a likelihood ratio test. Adding the following statement to the previous PROC GLIMMIX run compares the model to one in which the variance of the random intercepts (the first covariance parameter) is not constrained and the scale parameter is fixed at zero:

```
covtest 'H: phi = 0' . 0 / est;
```

This COVTEST statement produces Output 44.14.5.

Output 44.14.5 Likelihood Ratio Test for Poisson Assumption

Tests of Covariance Parameters Based on the Likelihood

Label	DF	-2 Log Like	ChiSq	Pr > ChiSq	Estimates H0 Est1	Est2	Note
H:phi = 0	1	854.47	144.64	<.0001	1.1959	1.11E-12	MI

MI: P-value based on a mixture of chi-squares.

Note that the –2 Log Like reported in Output 44.14.5 agrees with the value reported in the "Fit Statistics" table for the Poisson model (Output 44.14.2) and that the estimate of the random intercept under the null hypothesis agrees with the "Covariance Parameter Estimates" table in Output 44.14.2. Because the null hypothesis places the parameter ϕ (or ξ) on the boundary of the parameter space, a mixture correction is applied in the p-value calculation. Because of the magnitude of the likelihood ratio statistic (144.64), this correction has no effect on the displayed p-value.

Example 44.15: Comparing Multiple B-Splines

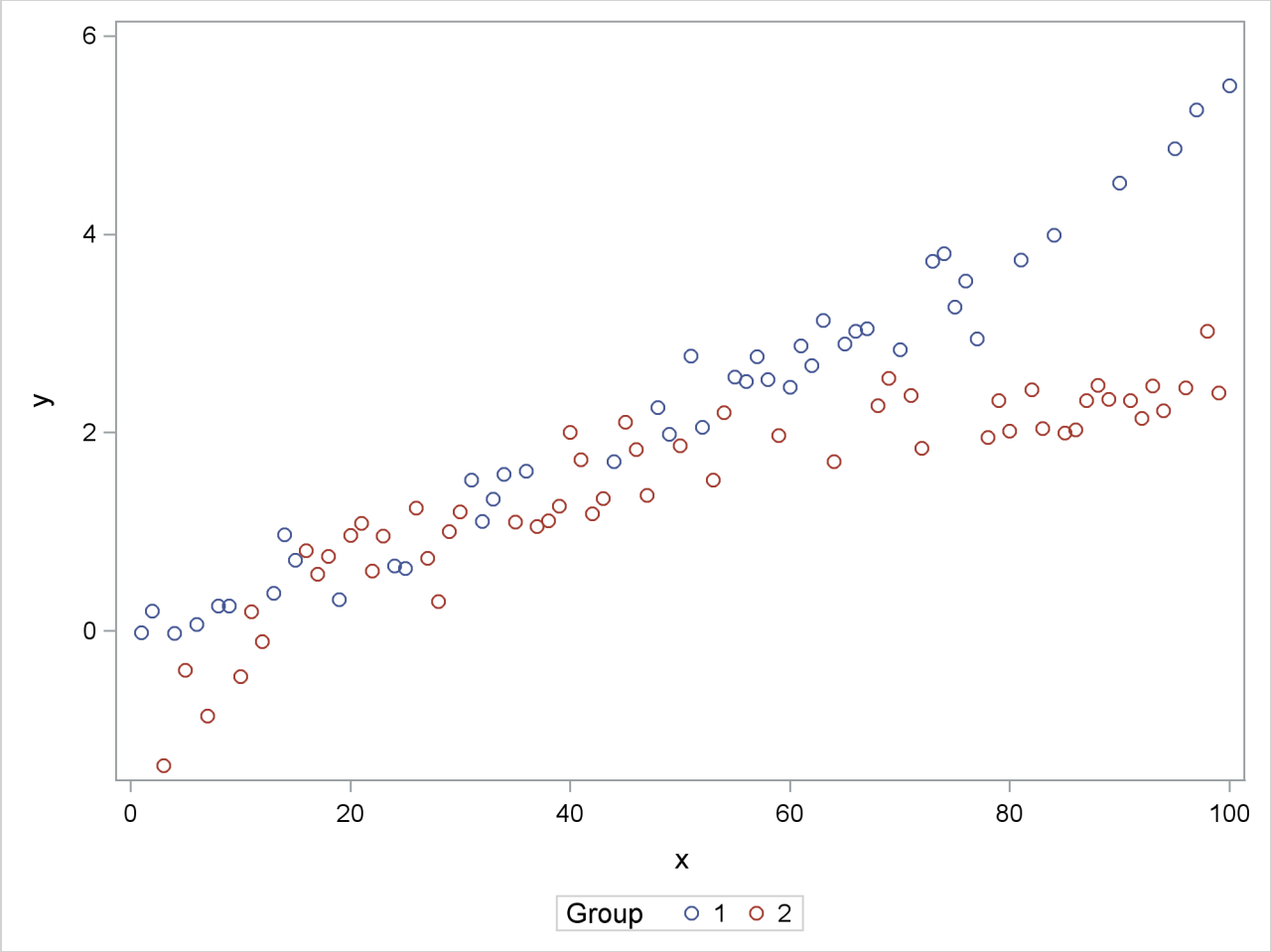
This example uses simulated data to demonstrate the use of the nonpositional syntax (see the section "Positional and Nonpositional Syntax for Contrast Coefficients" on page 3232 for details) in combination with the experimental EFFECT statement to produce interesting predictions and comparisons in models containing fixed spline effects. Consider the data in the following DATA step. Each of the 100 observations for the continuous response variable y is associated with one of two groups.

```
data spline;
   input group y @@;
   x = _n_;
   datalines;
 1    -.020 1    0.199 2    -1.36 1    -.026
 2    -.397 1    0.065 2    -.861 1    0.251
 1    0.253 2    -.460 2    0.195 2    -.108
 1    0.379 1    0.971 1    0.712 2    0.811
 2    0.574 2    0.755 1    0.316 2    0.961
 2    1.088 2    0.607 2    0.959 1    0.653
 1    0.629 2    1.237 2    0.734 2    0.299
 2    1.002 2    1.201 1    1.520 1    1.105
 1    1.329 1    1.580 2    1.098 1    1.613
 2    1.052 2    1.108 2    1.257 2    2.005
 2    1.726 2    1.179 2    1.338 1    1.707
 2    2.105 2    1.828 2    1.368 1    2.252
 1    1.984 2    1.867 1    2.771 1    2.052
 2    1.522 2    2.200 1    2.562 1    2.517
 1    2.769 1    2.534 2    1.969 1    2.460
 1    2.873 1    2.678 1    3.135 2    1.705
 1    2.893 1    3.023 1    3.050 2    2.273
 2    2.549 1    2.836 2    2.375 2    1.841
 1    3.727 1    3.806 1    3.269 1    3.533
 1    2.948 2    1.954 2    2.326 2    2.017
 1    3.744 2    2.431 2    2.040 1    3.995
 2    1.996 2    2.028 2    2.321 2    2.479
 2    2.337 1    4.516 2    2.326 2    2.144
 2    2.474 2    2.221 1    4.867 2    2.453
 1    5.253 2    3.024 2    2.403 1    5.498
;
```

The following statements produce a scatter plot of the response variable by group (Output 44.15.1):

```
proc sgplot data=spline;
   scatter y=y x=x / group=group name="data";
   keylegend "data"  / title="Group";
run;
```

Output 44.15.1 Scatter Plot of Observed Data by Group



The trends in the two groups exhibit curvature, but the type of curvature is not the same in the groups. Also, there appear to be ranges of x values where the groups are similar and areas where the point scatters separate. To model the trends in the two groups separately and with flexibility, you might want to allow for some smooth trends in x that vary by group. Consider the following PROC GLIMMIX statements:

```
proc glimmix data=spline outdesign=x;
   class group;
   effect spl = spline(x);
   model y = group spl*group / s noint;
   output out=gmxout pred=p;
run;
```

The EFFECT statement defines a constructed effect named spl by expanding the x into a spline with seven columns. The group main effect creates separate intercepts for the groups, and the interaction of the group variable with the spline effect creates separate trends. The NOINT option suppresses the intercept. This is not necessary and is done here only for convenience of interpretation. The OUTPUT statement computes predicted values.

The "Parameter Estimates" table contains the estimates of the group-specific "intercepts," the spline coefficients varied by group, and the residual variance ("Scale," Output 44.15.2).

Output 44.15.2 Parameter Estimates in Two-Group Spline Model

The GLIMMIX Procedure

Parameter Estimates

Effect	spl	group	Estimate	Standard Error	DF	t Value	Pr > |t|
group		1	9.7027	3.1342	86	3.10	0.0026
group		2	6.3062	2.6299	86	2.40	0.0187
spl*group	1	1	-11.1786	3.7008	86	-3.02	0.0033
spl*group	1	2	-20.1946	3.9765	86	-5.08	<.0001
spl*group	2	1	-9.5327	3.2576	86	-2.93	0.0044
spl*group	2	2	-5.8565	2.7906	86	-2.10	0.0388
spl*group	3	1	-8.9612	3.0718	86	-2.92	0.0045
spl*group	3	2	-5.5567	2.5717	86	-2.16	0.0335
spl*group	4	1	-7.2615	3.2437	86	-2.24	0.0278
spl*group	4	2	-4.3678	2.7247	86	-1.60	0.1126
spl*group	5	1	-6.4462	2.9617	86	-2.18	0.0323
spl*group	5	2	-4.0380	2.4589	86	-1.64	0.1042
spl*group	6	1	-4.6382	3.7095	86	-1.25	0.2146
spl*group	6	2	-4.3029	3.0479	86	-1.41	0.1616
spl*group	7	1	0	.	.	.	.
spl*group	7	2	0	.	.	.	.
Scale			0.07352	0.01121	.	.	.

Because the B-spline coefficients for an observation sum to 1 and the model contains group-specific constants, the last spline coefficient in each group is zero. In other words, you can achieve exactly the same fit with the MODEL statement

```
model y = spl*group / noint;
```

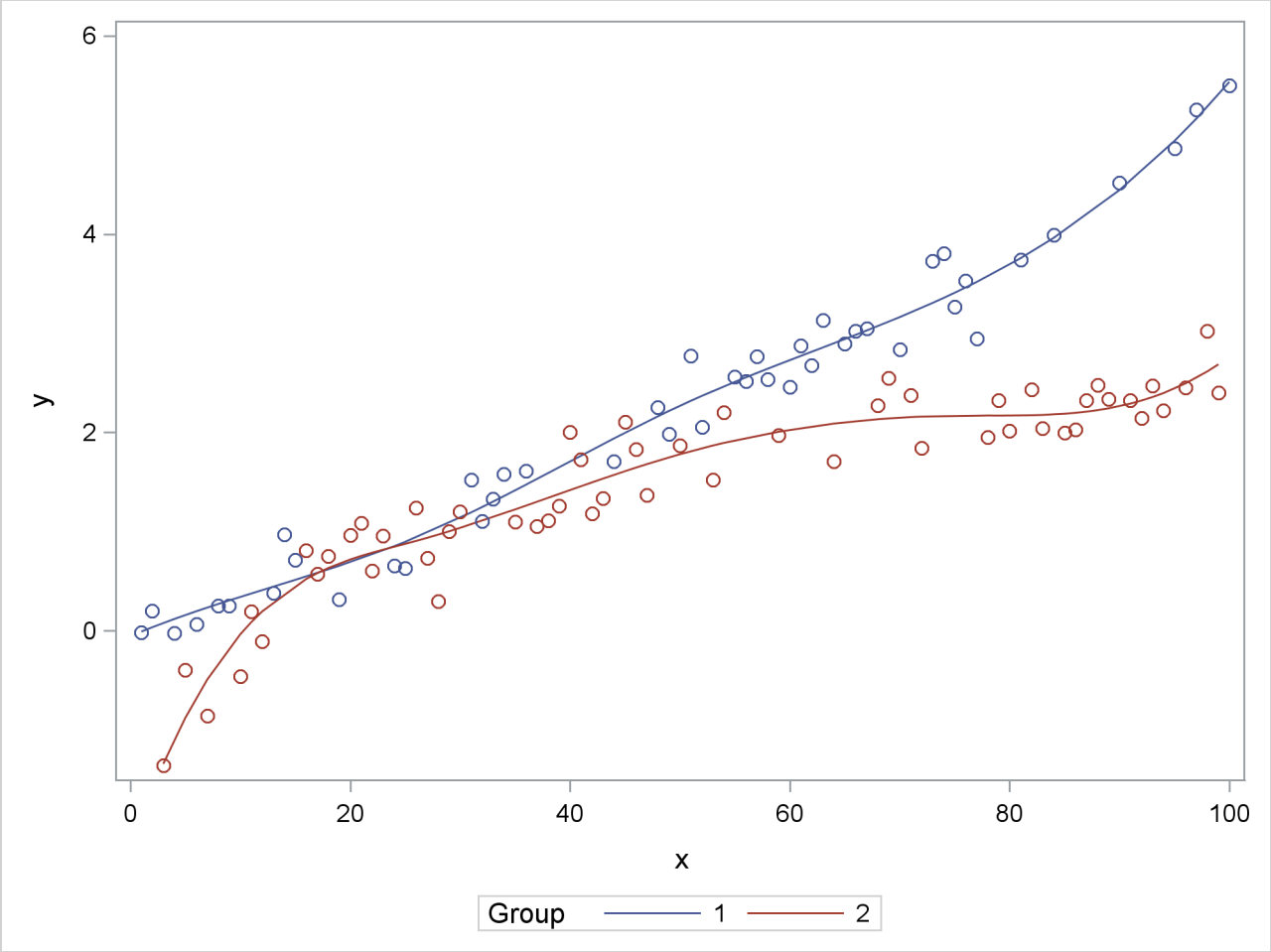
or

```
model y = spl*group;
```

The following statements graph the observed and fitted values in the two groups (Output 44.15.3):

```
proc sgplot data=gmxout;
   series  y=p x=x / group=group name="fit";
   scatter y=y x=x / group=group;
   keylegend "fit" / title="Group";
run;
```

Output 44.15.3 Observed and Predicted Values by Group



Suppose that you are interested in estimating the mean response at particular values of x and in performing comparisons of predicted values. The following program uses ESTIMATE statements with nonpositional syntax to accomplish this:

```
proc glimmix data=spline;
   class group;
   effect spl = spline(x);
   model y = group spl*group / s noint;
   estimate 'Group 1, x=20' group 1    group*spl [1,1 20] / e;
   estimate 'Group 2, x=20' group 0  1 group*spl [1,2 20];
   estimate 'Diff at x=20 ' group 1 -1 group*spl [1,1 20] [-1,2 20];
run;
```

The first ESTIMATE statement predicts the mean response at $x = 20$ in group 1. The E option requests the coefficient vector for this linear combination of the parameter estimates. The coefficient for the group effect is entered with positional (standard) syntax. The coefficients for the group*spl effect are formed based on nonpositional syntax. Because this effect comprises the interaction of a standard effect (group) with a constructed effect, the values and levels for the standard effect must precede those for the constructed effect. A similar statement produces the predicted mean at $x = 20$ in group 2.

The GLIMMIX procedure interprets the syntax

```
group*spl [1,2 20]
```

as follows: construct the spline basis at $x = 20$ as appropriate for group 2; then multiply the resulting coefficients for these columns of the **L** matrix with 1.

The final ESTIMATE statement represents the difference between the predicted values; it is a group comparison at $x = 20$.

Output 44.15.4 Coefficients from First ESTIMATE Statement

The GLIMMIX Procedure

Coefficients for Estimate Group 1, x=20			
Effect	**spl**	**group**	**Row1**
group		1	1
group		2	
spl*group	1	1	0.0021
spl*group	1	2	
spl*group	2	1	0.3035
spl*group	2	2	
spl*group	3	1	0.619
spl*group	3	2	
spl*group	4	1	0.0754
spl*group	4	2	
spl*group	5	1	
spl*group	5	2	
spl*group	6	1	
spl*group	6	2	
spl*group	7	1	
spl*group	7	2	

The "Coefficients" table shows how the value 20 supplied in the ESTIMATE statement was expanded into the appropriate spline basis (Output 44.15.4). There is no significant difference between the group means at $x = 20$ ($p = 0.8346$, Output 44.15.5).

Output 44.15.5 Results from ESTIMATE Statements

Estimates					
Label	**Estimate**	**Standard Error**	**DF**	**t Value**	**Pr > |t|**
Group 1, x=20	0.6915	0.09546	86	7.24	<.0001
Group 2, x=20	0.7175	0.07953	86	9.02	<.0001
Diff at x=20	-0.02602	0.1243	86	-0.21	0.8346

The group comparisons you can achieve in this way are comparable to slices of interaction effects with classification effects. There are, however, no preset number of levels at which to perform the comparisons because x is continuous. If you add further x values for the comparisons, a multiplicity correction is in order to control the familywise Type I error. The following statements compare the groups at values $x = 0, 5, 10, \cdots, 80$ and compute simulation-based step-down-adjusted p-values. The results appear in

Output 44.15.6. (The numeric results for simulation-based *p*-value adjustments depend slightly on the value of the random number seed.)

```
ods select Estimates;
proc glimmix data=spline;
   class group;
   effect spl = spline(x);
   model y = group spl*group / s;
   estimate 'Diff at x= 0' group 1 -1 group*spl [1,1  0] [-1,2  0],
            'Diff at x= 5' group 1 -1 group*spl [1,1  5] [-1,2  5],
            'Diff at x=10' group 1 -1 group*spl [1,1 10] [-1,2 10],
            'Diff at x=15' group 1 -1 group*spl [1,1 15] [-1,2 15],
            'Diff at x=20' group 1 -1 group*spl [1,1 20] [-1,2 20],
            'Diff at x=25' group 1 -1 group*spl [1,1 25] [-1,2 25],
            'Diff at x=30' group 1 -1 group*spl [1,1 30] [-1,2 30],
            'Diff at x=35' group 1 -1 group*spl [1,1 35] [-1,2 35],
            'Diff at x=40' group 1 -1 group*spl [1,1 40] [-1,2 40],
            'Diff at x=45' group 1 -1 group*spl [1,1 45] [-1,2 45],
            'Diff at x=50' group 1 -1 group*spl [1,1 50] [-1,2 50],
            'Diff at x=55' group 1 -1 group*spl [1,1 55] [-1,2 55],
            'Diff at x=60' group 1 -1 group*spl [1,1 60] [-1,2 60],
            'Diff at x=65' group 1 -1 group*spl [1,1 65] [-1,2 65],
            'Diff at x=70' group 1 -1 group*spl [1,1 70] [-1,2 70],
            'Diff at x=75' group 1 -1 group*spl [1,1 75] [-1,2 75],
            'Diff at x=80' group 1 -1 group*spl [1,1 80] [-1,2 80] /
           adjust=sim(seed=1) stepdown;
run;
```

Output 44.15.6 Estimates with Multiplicity Adjustments

The GLIMMIX Procedure

Estimates
Adjustment for Multiplicity: Holm-Simulated

Label	Estimate	Standard Error	DF	t Value	Pr > |t|	Adj P
Diff at x= 0	12.4124	4.2130	86	2.95	0.0041	0.0210
Diff at x= 5	1.0376	0.1759	86	5.90	<.0001	<.0001
Diff at x=10	0.3778	0.1540	86	2.45	0.0162	0.0554
Diff at x=15	0.05822	0.1481	86	0.39	0.6952	0.9043
Diff at x=20	-0.02602	0.1243	86	-0.21	0.8346	0.9578
Diff at x=25	0.02014	0.1312	86	0.15	0.8783	0.9578
Diff at x=30	0.1023	0.1378	86	0.74	0.4600	0.7419
Diff at x=35	0.1924	0.1236	86	1.56	0.1231	0.2890
Diff at x=40	0.2883	0.1114	86	2.59	0.0113	0.0465
Diff at x=45	0.3877	0.1195	86	3.24	0.0017	0.0098
Diff at x=50	0.4885	0.1308	86	3.74	0.0003	0.0022
Diff at x=55	0.5903	0.1231	86	4.79	<.0001	<.0001
Diff at x=60	0.7031	0.1125	86	6.25	<.0001	<.0001
Diff at x=65	0.8401	0.1203	86	6.99	<.0001	<.0001
Diff at x=70	1.0147	0.1348	86	7.52	<.0001	<.0001
Diff at x=75	1.2400	0.1326	86	9.35	<.0001	<.0001
Diff at x=80	1.5237	0.1281	86	11.89	<.0001	<.0001

There are significant differences at the low end and high end of the x range. Notice that without the multiplicity adjustment you would have concluded at the 0.05 level that the groups are significantly different at $x = 10$. At the 0.05 level, the groups separate significantly for $x < 10$ and $x > 40$.

Example 44.16: Diallel Experiment with Multimember Random Effects

Cockerham and Weir (1977) apply variance component models in the analysis of reciprocal crosses. In these experiments, it is of interest to separate genetically determined variation from variation determined by parentage. This example analyzes the data for the diallel experiment in Cockerham and Weir (1977, Appendix C). A diallel is a mating design that consists of all possible crosses of a set of parental lines. It includes reciprocal crossings, but not self-crossings.

The basic model for a cross is $Y_{ijk} = \beta + \alpha_{ij} + \epsilon_{ijk}$, where Y_{ijk} is the observation for offspring k from maternal parent i and paternal parent j. The various models in Cockerham and Weir (1977) are different decompositions of the term α_{ij}, the total effect that is due to the parents. Their "bio model" (model (c)) decomposes α_{ij} into

$$\alpha_{ij} = \eta_i + \eta_j + \mu_i + \phi_j + (\eta\eta)_{ij} + \kappa_{ij}$$

where η_i and η_j are contributions of the female and male parents, respectively. The term $(\eta\eta)_{ij}$ captures the interaction between maternal and paternal effects. In contrast to usual interaction effects, this term must obey a symmetry because of the reciprocals: $(\eta\eta)_{ij} = (\eta\eta)_{ji}$. The terms μ_i and ϕ_j in the decomposition are extranuclear maternal and paternal effects, and the remaining interactions are captured by the κ_{ij} term.

The following DATA step creates a SAS data set for the diallel example in Appendix C of Cockerham and Weir (1977):

```
data diallel;
   label time = 'Flowering time in days';
   do p = 1 to 8;
      do m = 1 to 8;
         if (m ne p) then do;
            sym = trim(left(min(m,p))) || ',' || trim(left(max(m,p)));
            do block = 1 to 2;
               input time @@;
               output;
            end;
         end;
      end;
   end;
   datalines;
14.4 16.2 27.2 30.8 17.2 27.0 18.3 20.2 16.2 16.8 18.6 14.4 16.4 16.0
15.4 16.5 14.8 14.6 18.6 18.6 15.2 15.3 17.0 15.2 14.4 14.8 10.8 13.2
31.8 30.4 21.0 23.0 24.6 25.4 19.2 20.0 29.8 28.4 12.8 14.2 13.0 14.4
16.2 17.8 11.4 13.0 16.8 16.3 12.4 14.2 16.8 14.8 12.6 12.2  9.6 11.2
14.6 18.8 12.2 13.6 15.2 15.4 15.2 13.8 18.0 16.0 10.4 12.2 13.4 20.0
20.2 23.4 14.2 14.0 18.6 14.8 22.2 17.0 14.3 17.3  9.0 10.2 11.8 12.8
14.0 16.6 12.2  9.2 13.6 16.2 13.8 14.4 15.6 15.6 15.6 11.0 13.0  9.8
15.2 17.2 10.0 11.6 17.0 18.2 20.8 20.8 20.0 17.4 17.0 12.6 13.0  9.8
;
```

The observations represent mean flowering times of *Nicotiana rustica* (Aztec tobacco) from crosses of inbred varieties grown in two blocks. The variables p and m identify the eight paternal and maternal lines, respectively. The variable sym is used to model the interaction between the parents, subject to the symmetry condition $(\eta\eta)_{ij} = (\eta\eta)_{ji}$. For example, the first two observations, 14.4 and 16.2 days, represent the observations from blocks 1 and 2 where paternal line 1 was crossed with maternal line 2.

The following PROC GLIMMIX statements fit the "bio model" in Cockerham and Weir (1977):

```
proc glimmix data=diallel outdesign(z)=zmat;
   class block sym p m;
   effect line = mm(p m);
   model  time = block;
   random line sym p m p*m;
run;
```

The EFFECT statement defines the nuclear parental contributions as a multimember effect based on the CLASS variables p and m. Each observation has two nonzero entries in the design matrix for the effect that identifies the paternal and maternal lines. The terms in the RANDOM statement model the variance components as follows: line $\rightarrow \sigma^2_n$, sym $\rightarrow \sigma^2_{(\eta\eta)}$, p $\rightarrow \sigma^2_\phi$, m $\rightarrow \sigma^2_\mu$, p*m $\rightarrow \sigma^2_\kappa$. The OUTDESIGN= option in the PROC GLIMMIX statement writes the $\mathbf{Z}$ matrix to the SAS data set zmat. The EFFECT statement alleviates the need for complex coding, as in Section 2.3 of Saxton (2004).

Output 44.16.1 displays the "Class Level Information" table of the diallel model. Because the interaction terms are symmetric, there are only $8 \times 7/2 = 28$ levels for the 8 lines. The estimates of the variance components and the residual variance in Output 44.16.1 agree with the results in Table 7 of Cockerham and Weir (1977).

Output 44.16.1 Class Levels and Covariance Parameter Estimates in Diallel Example

The GLIMMIX Procedure

Class Level Information

Class	Levels	Values
block	2	1 2
sym	28	1,2 1,3 1,4 1,5 1,6 1,7 1,8 2,3 2,4 2,5 2,6 2,7 2,8 3,4 3,5 3,6 3,7 3,8 4,5 4,6 4,7 4,8 5,6 5,7 5,8 6,7 6,8 7,8
p	8	1 2 3 4 5 6 7 8
m	8	1 2 3 4 5 6 7 8

Covariance Parameter Estimates

Cov Parm	Estimate	Standard Error
line	5.1047	4.0021
sym	2.3856	1.9025
p	3.3080	3.4053
m	1.9134	2.9891
p*m	4.0196	1.8323
Residual	3.6225	0.6908

The following statements print the $\mathbf{Z}$ matrix columns that correspond to the multimember line effect for the first 10 observations in block 1 (Output 44.16.2). For each observation there are two nonzero entries, and their column index corresponds to the index of the paternal and maternal line.

```
proc print data=zmat(where=(block=1) obs=10);
   var p m time _z1-_z8;
run;
```

Output 44.16.2 **Z** Matrix for Line Effect of the First 10 Observations in Block 1

Obs	p	m	time	_Z1	_Z2	_Z3	_Z4	_Z5	_Z6	_Z7	_Z8
1	1	2	14.4	1	1	0	0	0	0	0	0
3	1	3	27.2	1	0	1	0	0	0	0	0
5	1	4	17.2	1	0	0	1	0	0	0	0
7	1	5	18.3	1	0	0	0	1	0	0	0
9	1	6	16.2	1	0	0	0	0	1	0	0
11	1	7	18.6	1	0	0	0	0	0	1	0
13	1	8	16.4	1	0	0	0	0	0	0	1
15	2	1	15.4	1	1	0	0	0	0	0	0
17	2	3	14.8	0	1	1	0	0	0	0	0
19	2	4	18.6	0	1	0	1	0	0	0	0

Example 44.17: Linear Inference Based on Summary Data

The GLIMMIX procedure has facilities for multiplicity-adjusted inference through the ADJUST= and STEPDOWN options in the ESTIMATE, LSMEANS, and LSMESTIMATE statements. You can employ these facilities to test linear hypotheses among parameters even in situations where the quantities were obtained outside the GLIMMIX procedure. This example demonstrates the process. The basic idea is to prepare a data set containing the estimates of interest and a data set containing their covariance matrix. These are then passed to the GLIMMIX procedure, preventing updating of the parameters, essentially moving directly into the post-processing stage as if estimates with this covariance matrix had been produced by the GLIMMIX procedure.

The final documentation example in Chapter 69, "The NLIN Procedure," discusses a nonlinear first-order compartment pharmacokinetic model for theophylline concentration. The data are derived by collapsing and averaging the subject-specific data from Pinheiro and Bates (1995) in a particular—yet unimportant—way that leads to two groups for comparisons. The following DATA step creates these data:

```
data theop;
   input time dose conc @@;
   if (dose = 4) then group=1; else group=2;
   datalines;
 0.00   4  0.1633  0.25   4   2.045
 0.27   4     4.4  0.30   4    7.37
 0.35   4    1.89  0.37   4    2.89
 0.50   4    3.96  0.57   4    6.57
 0.58   4     6.9  0.60   4     4.6
 0.63   4    9.03  0.77   4    5.22
 1.00   4    7.82  1.02   4   7.305
 1.05   4    7.14  1.07   4     8.6
 1.12   4    10.5  2.00   4    9.72
 2.02   4    7.93  2.05   4    7.83
 2.13   4    8.38  3.50   4    7.54
 3.52   4    9.75  3.53   4    5.66
```

```
 3.55   4   10.21  3.62   4     7.5
 3.82   4    8.58  5.02   4   6.275
 5.05   4    9.18  5.07   4    8.57
 5.08   4     6.2  5.10   4    8.36
 7.02   4    5.78  7.03   4    7.47
 7.07   4   5.945  7.08   4    8.02
 7.17   4    4.24  8.80   4    4.11
 9.00   4     4.9  9.02   4    5.33
 9.03   4    6.11  9.05   4    6.89
 9.38   4    7.14 11.60   4    3.16
11.98   4    4.19 12.05   4    4.57
12.10   4    5.68 12.12   4    5.94
12.15   4     3.7 23.70   4    2.42
24.15   4    1.17 24.17   4    1.05
24.37   4    3.28 24.43   4    1.12
24.65   4    1.15  0.00   5   0.025
 0.25   5    2.92  0.27   5   1.505
 0.30   5    2.02  0.50   5   4.795
 0.52   5    5.53  0.58   5    3.08
 0.98   5   7.655  1.00   5   9.855
 1.02   5    5.02  1.15   5    6.44
 1.92   5    8.33  1.98   5    6.81
 2.02   5  7.8233  2.03   5    6.32
 3.48   5    7.09  3.50   5   7.795
 3.53   5    6.59  3.57   5    5.53
 3.60   5    5.87  5.00   5     5.8
 5.02   5  6.2867  5.05   5    5.88
 6.98   5    5.25  7.00   5    4.02
 7.02   5    7.09  7.03   5   4.925
 7.15   5    4.73  9.00   5    4.47
 9.03   5    3.62  9.07   5    4.57
 9.10   5     5.9  9.22   5    3.46
12.00   5    3.69 12.05   5    3.53
12.10   5    2.89 12.12   5    2.69
23.85   5    0.92 24.08   5    0.86
24.12   5    1.25 24.22   5    1.15
24.30   5     0.9 24.35   5    1.57
;
```

In terms of two fixed treatment groups, the nonlinear model for these data can be written as

$$C_{it} = \frac{Dk_{e_i}k_{a_i}}{Cl_i(k_{a_i} - k_{e_i})}[\exp(-k_{e_i}t) - \exp(-k_{a_i}t)] + \epsilon_{it}$$

where C_{it} is the observed concentration in group i at time t, D is the dose of theophylline, k_{e_i} is the elimination rate constant in group i, k_{a_i} is the absorption rate in group i, Cl_i is the clearance in group i, and ϵ_{it} denotes the model error. Because the rates and the clearance must be positive, you can parameterize the model in terms of log rates and the log clearance:

$$Cl_i = \exp\{\beta_{1i}\}$$
$$k_{a_i} = \exp\{\beta_{2i}\}$$
$$k_{e_i} = \exp\{\beta_{3i}\}$$

In this parameterization the model contains six parameters, and the rates and clearance vary by group. The following PROC NLIN statements fit the model and obtain the group-specific parameter estimates:

```
proc nlin data=theop outest=cov;
   parms beta1_1=-3.22 beta2_1=0.47 beta3_1=-2.45
         beta1_2=-3.22 beta2_2=0.47 beta3_2=-2.45;
   if (group=1) then do;
      cl   = exp(beta1_1);
      ka   = exp(beta2_1);
      ke   = exp(beta3_1);
   end; else do;
      cl   = exp(beta1_2);
      ka   = exp(beta2_2);
      ke   = exp(beta3_2);
   end;
   mean = dose*ke*ka*(exp(-ke*time)-exp(-ka*time))/cl/(ka-ke);
   model conc = mean;
   ods output ParameterEstimates=ests;
run;
```

The conditional programming statements determine the clearance, elimination, and absorption rates depending on the value of the group variable. The OUTEST= option in the PROC NLIN statement saves estimates and their covariance matrix to the data set cov. The ODS OUTPUT statement saves the "Parameter Estimates" table to the data set ests.

Output 44.17.1 displays the analysis of variance table and the parameter estimates from this NLIN run. Note that the confidence levels in the "Parameter Estimates" table are based on 92 degrees of freedom, corresponding to the residual degrees of freedom in the analysis of variance table.

Output 44.17.1 Analysis of Variance and Parameter Estimates for Nonlinear Model

The NLIN Procedure

Note: An intercept was not specified for this model.

Source	DF	Sum of Squares	Mean Square	F Value	Approx Pr > F
Model	6	3247.9	541.3	358.56	<.0001
Error	92	138.9	1.5097		
Uncorrected Total	98	3386.8			

Parameter	Estimate	Approx Std Error	Approximate 95% Confidence Limits	
beta1_1	-3.5671	0.0864	-3.7387	-3.3956
beta2_1	0.4421	0.1349	0.1742	0.7101
beta3_1	-2.6230	0.1265	-2.8742	-2.3718
beta1_2	-3.0111	0.1061	-3.2219	-2.8003
beta2_2	0.3977	0.1987	0.00305	0.7924
beta3_2	-2.4442	0.1618	-2.7655	-2.1229

The following DATA step extracts the part of the cov data set that contains the covariance matrix of the parameter estimates in Output 44.17.1 and renames the variables as Col1–Col6. Output 44.17.2 shows the result of the DATA step.

```
data covb;
   set cov(where=(_type_='COVB'));
   rename beta1_1=col1 beta2_1=col2 beta3_1=col3
          beta1_2=col4 beta2_2=col5 beta3_2=col6;
   row = _n_;
   Parm = 1;
   keep parm row beta:;
run;

proc print data=covb;
run;
```

Output 44.17.2 Covariance Matrix of NLIN Parameter Estimates

Obs	col1	col2	col3	col4	col5	col6	row	Parm
1	0.007462	-0.005222	0.010234	0.000000	0.000000	0.000000	1	1
2	-0.005222	0.018197	-0.010590	0.000000	0.000000	0.000000	2	1
3	0.010234	-0.010590	0.015999	0.000000	0.000000	0.000000	3	1
4	0.000000	0.000000	0.000000	0.011261	-0.009096	0.015785	4	1
5	0.000000	0.000000	0.000000	-0.009096	0.039487	-0.019996	5	1
6	0.000000	0.000000	0.000000	0.015785	-0.019996	0.026172	6	1

The reason for this transformation of the data is to use the resulting data set to define a covariance structure in PROC GLIMMIX. The following statements reconstitute a model in which the parameter estimates from PROC NLIN are the observations and in which the covariance matrix of the "observations" matches the covariance matrix of the NLIN parameter estimates:

```
proc glimmix data=ests order=data;
   class Parameter;
   model Estimate = Parameter / noint df=92 s;
   random _residual_ / type=lin(1) ldata=covb v;
   parms (1) / noiter;
   lsmeans parameter / cl;
   lsmestimate Parameter
               'beta1 eq. across groups' 1 0 0 -1,
               'beta2 eq. across groups' 0 1 0  0 -1,
               'beta3 eq. across groups' 0 0 1  0  0  -1 /
               adjust=bon stepdown ftest(label='Homogeneity');
run;
```

In other words, you are using PROC GLIMMIX to set up a linear statistical model

$$\mathbf{Y} = \mathbf{I}\boldsymbol{\alpha} + \boldsymbol{\epsilon}$$
$$\boldsymbol{\epsilon} \sim (\mathbf{0}, \mathbf{A})$$

where the covariance matrix $\mathbf{A}$ is given by

$$\mathbf{A} = \begin{bmatrix} 0.007 & -0.005 & 0.010 & 0 & 0 & 0 \\ -0.005 & 0.018 & -0.011 & 0 & 0 & 0 \\ 0.010 & -0.011 & 0.016 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.011 & -0.009 & 0.016 \\ 0 & 0 & 0 & -0.009 & 0.039 & -0.019 \\ 0 & 0 & 0 & 0.016 & -0.019 & 0.026 \end{bmatrix}$$

The generalized least squares estimate for $\boldsymbol{\alpha}$ in this saturated model reproduces the observations:

$$\begin{aligned} \widehat{\boldsymbol{\alpha}} &= \left(\mathbf{I}'\mathbf{A}^{-1}\mathbf{I}\right)^{-1}\mathbf{I}'\mathbf{A}^{-1}\mathbf{y} \\ &= \left(\mathbf{A}^{-1}\right)^{-1}\mathbf{A}^{-1}\mathbf{y} \\ &= \mathbf{y} \end{aligned}$$

The ORDER=DATA option in the PROC GLIMMIX statement requests that the sort order of the Parameter variable be identical to the order in which it appeared in the "Parameter Estimates" table of the NLIN procedure (Output 44.17.1). The MODEL statement uses the Estimate and Parameter variables from that table to form a model in which the $\mathbf{X}$ matrix is the identity; hence the NOINT option. The DF=92 option sets the degrees of freedom equal to the value used in the NLIN procedure. The RANDOM statement specifies a linear covariance structure with a single component and supplies the values for the structure through the LDATA= data set. This structure models the covariance matrix as $\mathrm{Var}[\mathbf{Y}] = \theta\mathbf{A}$, where the $\mathbf{A}$ matrix is given previously. Essentially, the TYPE=LIN(1) structure forces an unstructured covariance matrix onto the data. To make this work, the parameter θ is held fixed at 1 in the PARMS statement.

Output 44.17.3 displays the parameter estimates and least squares means for this model. Note that estimates and least squares means are identical, since the $\mathbf{X}$ matrix is the identity. Also, the confidence limits agree with the values reported by PROC NLIN (see Output 44.17.1).

Output 44.17.3 Parameter Estimates and LS-Means from Summary Data

The GLIMMIX Procedure

Solutions for Fixed Effects

Effect	Parameter	Estimate	Standard Error	DF	t Value	Pr > |t|
Parameter	beta1_1	-3.5671	0.08638	92	-41.29	<.0001
Parameter	beta2_1	0.4421	0.1349	92	3.28	0.0015
Parameter	beta3_1	-2.6230	0.1265	92	-20.74	<.0001
Parameter	beta1_2	-3.0111	0.1061	92	-28.37	<.0001
Parameter	beta2_2	0.3977	0.1987	92	2.00	0.0483
Parameter	beta3_2	-2.4442	0.1618	92	-15.11	<.0001

Parameter Least Squares Means

Parameter	Estimate	Standard Error	DF	t Value	Pr > |t|	Alpha	Lower	Upper
beta1_1	-3.5671	0.08638	92	-41.29	<.0001	0.05	-3.7387	-3.3956
beta2_1	0.4421	0.1349	92	3.28	0.0015	0.05	0.1742	0.7101
beta3_1	-2.6230	0.1265	92	-20.74	<.0001	0.05	-2.8742	-2.3718
beta1_2	-3.0111	0.1061	92	-28.37	<.0001	0.05	-3.2219	-2.8003
beta2_2	0.3977	0.1987	92	2.00	0.0483	0.05	0.003050	0.7924
beta3_2	-2.4442	0.1618	92	-15.11	<.0001	0.05	-2.7655	-2.1229

The (marginal) covariance matrix of the data is shown in Output 44.17.4 to confirm that it matches the **A** matrix given earlier.

Output 44.17.4 R-Side Covariance Matrix

Estimated V Matrix for Subject 1						
Row	Col1	Col2	Col3	Col4	Col5	Col6
1	0.007462	-0.00522	0.01023			
2	-0.00522	0.01820	-0.01059			
3	0.01023	-0.01059	0.01600			
4				0.01126	-0.00910	0.01579
5				-0.00910	0.03949	-0.02000
6				0.01579	-0.02000	0.02617

The LSMESTIMATE statement specifies three linear functions. These set equal the β parameters from the groups. The step-down Bonferroni adjustment requests a multiplicity adjustment for the family of three tests. The FTEST option requests a joint test of the three estimable functions; it is a global test of parameter homogeneity across groups.

Output 44.17.5 displays the result from the LSMESTIMATE statement. The joint test is highly significant ($F = 30.52$, $p < 0.0001$). From the p-values associated with the individual rows of the estimates, you can see that the lack of homogeneity is due to group differences for β_1, the log clearance.

Output 44.17.5 Test of Parameter Homogeneity across Groups

Least Squares Means Estimates Adjustment for Multiplicity: Holm							
Effect	Label	Estimate	Standard Error	DF	t Value	Pr > |t|	Adj P
Parameter	beta1 eq. across groups	-0.5560	0.1368	92	-4.06	0.0001	0.0003
Parameter	beta2 eq. across groups	0.04443	0.2402	92	0.18	0.8537	0.8537
Parameter	beta3 eq. across groups	-0.1788	0.2054	92	-0.87	0.3862	0.7725

F Test for Least Squares Means Estimates				
Label	Num DF	Den DF	F Value	Pr > F
Homogeneity	3	92	30.52	<.0001

An alternative method of setting up this model is given by the following statements, where the data set pdata contains the covariance parameters:

```
random _residual_ / type=un;
parms / pdata=pdata noiter
```

The following DATA step creates an appropriate PDATA= data set from the data set covb, which was constructed earlier:

```
data pdata;
   set covb;
   array col{6};
   do i=1 to _n_;
```

```
      estimate = col{i};
      output;
   end;
   keep estimate;
run;
```

Example 44.18: Weighted Multilevel Model for Survey Data

Multilevel models are a useful tool for analyzing survey data from multistage sampling designs. In multistage designs, the first-stage clusters (primary sampling units) are selected by using a probability sample from a list of first-stage units. In the second stage, second-stage clusters are selected by using a probability sample from a list of second-stage clusters for every first-stage cluster in the sample. The third and subsequent stages of clusters are selected similarly. You can use multilevel models to analyze data from multistage designs in which each stage of sampling corresponds to one level of random effects in the model. The characteristics of the units at each stage become the explanatory variables at that level. If units are drawn with unequal selection probabilities at each stage, then the unweighted estimators from standard multilevel models can be biased. Pfeffermann et al. (1998) and Rabe-Hesketh and Skrondal (2006) discuss using sampling weights to reduce this bias.

To learn how to fit a two-level model, consider a two-stage design simulation that is similar to the simulation discussed in Rabe-Hesketh and Skrondal (2006).

First, you generate a finite population of 500 level-2 units and 50 level-1 units from a two-level random intercept probit model with dichotomous responses and linear predictor,

$$\eta_{ij} = \beta_0 + \beta_1 x_{1i} + \beta_2 x_{2ij} + \gamma_i$$

where $\beta_0 = \beta_1 = \beta_2 = 1$, $\gamma_i \sim N(0, \sigma^2)$, and $\sigma^2 = 1$. The between-cluster covariate x_1i and within-cluster covariate x_2ij are generated from a Bernoulli distribution with probability 0.5.

Then, you sample subsets of level-2 units and level-1 units from a two-stage sampling design similar to the one discussed in Rabe-Hesketh and Skrondal (2006). The generated sample contains a total of 7,530 observations. The inverse selection probabilities are used as weights. To reduce the bias in the variance parameter estimator, you scale the level-1 weights by using what Pfeffermann et al. (1998) and Rabe-Hesketh and Skrondal (2006) refer to as "Method 2."

The first 30 observations of the data set are shown in Output 44.18.1. In this data set, y is the dichotomous response variable, and x1 and x2 are the between-cluster and within-cluster covariates, respectively. The variables id and w2 identify the first-stage clusters (level-2 units) and their weights, respectively. The observation units are the second-stage clusters (level-1 units), and their weights are scaled using "Method 2" in Pfeffermann et al. (1998) and Rabe-Hesketh and Skrondal (2006) and stored in sw1. The first-stage sampling weights, w2, range from 1.3359 to 4.0513, with an average of 1.6965. The second-stage sampling weights, w1, range from 1.8571 to 2.000, with an average of 1.9588. The scaled weights, sw1, range from 0.9657 to 1.0126, with an average of 1.000.

Output 44.18.1 Simulated Two-Stage Sampling Survey Data (First 30 Observations)

Obs	x1	x2	y	w2	id	sw1
1	1	0	1	1.33594	1	1.00286
2	1	0	1	1.33594	1	1.00286
3	1	1	1	1.33594	1	1.00286
4	1	1	1	1.33594	1	1.00286
5	1	0	1	1.33594	1	1.00286
6	1	1	1	1.33594	1	1.00286
7	1	1	1	1.33594	1	1.00286
8	1	1	1	1.33594	1	1.00286
9	1	0	1	1.33594	1	1.00286
10	1	0	1	1.33594	1	1.00286
11	1	1	1	1.33594	1	1.00286
12	1	0	1	1.33594	1	1.00286
13	1	1	1	1.33594	1	1.00286
14	1	0	1	1.33594	1	1.00286
15	1	0	1	1.33594	1	0.99667
16	1	1	1	1.33594	1	0.99667
17	1	0	1	1.33594	1	0.99667
18	1	0	1	1.33594	1	0.99667
19	1	0	1	1.33594	1	0.99667
20	1	1	1	1.33594	1	0.99667
21	1	0	1	1.33594	1	0.99667
22	1	0	1	1.33594	1	0.99667
23	1	0	1	1.33594	1	0.99667
24	1	0	1	1.33594	1	0.99667
25	1	1	1	1.33594	1	0.99667
26	1	1	1	1.33594	1	0.99667
27	1	0	1	1.33594	2	0.99667
28	1	0	1	1.33594	2	0.99667
29	1	1	1	1.33594	2	0.99667
30	1	1	1	1.33594	2	0.99667

The following statements fit a two-level model without using weights at any level:

```
proc glimmix data=dws method=quadrature;
   class id;
   model y = x1 x2 / dist=binomial link=probit solution;
   random int  / subject=id;
run;
```

The parameter estimates are shown in Output 44.18.2. The estimated fixed-effects parameters are all close to their corresponding true values, but the estimated covariance parameter is almost half of its true value.

Output 44.18.2 Analysis of Two-Stage Sampling Survey Data

The GLIMMIX Procedure

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error
Intercept	id	0.5612	0.08777

Solutions for Fixed Effects

Effect	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept	1.0300	0.07608	293	13.54	<.0001
x1	1.0809	0.1186	7234	9.11	<.0001
x2	0.9934	0.06344	7234	15.66	<.0001

You can incorporate the unequal weights at both levels into the model by using the OBSWEIGHT= and WEIGHT= options as follows:

```
proc glimmix data=dws method=quadrature empirical=classical;
   class id;
   model y = x1 x2 / dist=binomial link=probit obsweight=sw1 solution;
   random int  / subject=id weight=w2;
run;
```

To fit a weighted multilevel model, you should use METHOD=QUAD. The EMPIRICAL=CLASSICAL option in the PROC GLIMMIX statement instructs PROC GLIMMIX to compute the empirical (sandwich) variance estimators for the fixed effect and the variance. The empirical variance estimators are recommended for the inference about fixed effects and variance estimated by pseudo-likelihood.

The OBSWEIGHT= option in the MODEL statement specifies the weight variable for the observation level. The WEIGHT= option in the RANDOM statement specifies the weight variable for the level that is specified by the SUBJECT= option.

The parameter estimates are shown in Output 44.18.3.

Output 44.18.3 Analysis of Two-Stage Sampling Survey Data Using Weights

The GLIMMIX Procedure

Covariance Parameter Estimates

Cov Parm	Subject	Estimate	Standard Error
Intercept	id	1.0106	0.2503

Solutions for Fixed Effects

Effect	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept	1.0322	0.1263	497.3	8.17	<.0001
x1	1.1182	0.2142	12275	5.22	<.0001
x2	1.0074	0.07887	12275	12.77	<.0001

Note that the estimated variance component for the random id effect is much closer to the true value of 1 in the weighted analysis. When you use the raw level-2 weights and the scaled level-1 weights, the multilevel model reduces the bias in the variance parameter estimate. However, the standard errors for the weighted analysis are larger than those for the unweighted analysis.

References

Abramowitz, M. and Stegun, I. A. (1972), *Handbook of Mathematical Functions*, New York: Dover Publications.

Akaike, H. (1974), "A New Look at the Statistical Model Identification," *IEEE Transactions on Automatic Control*, AC-19, 716–723.

Bahadur, R. R. (1961), "A Representation of the Joint Distribution of Responses to n Dichotomous Items," in H. Solomon, ed., *Studies in Item Analysis and Prediction*, Stanford, CA: Stanford University Press.

Beale, E. M. L. (1972), "A Derivation of Conjugate Gradients," in F. A. Lootsma, ed., *Numerical Methods for Nonlinear Optimization*, London: Academic Press.

Bell, R. M. and McCaffrey, D. F. (2002), "Bias Reduction in Standard Errors for Linear Regression with Multi-stage Samples," *Survey Methodology*, 28, 169–181.

Bickel, P. J. and Doksum, K. A. (1977), *Mathematical Statistics*, San Francisco: Holden-Day.

Booth, J. G. and Hobert, J. P. (1998), "Standard Errors of Prediction in Generalized Linear Mixed Models," *Journal of the American Statistical Association*, 93, 262–272.

Bozdogan, H. (1987), "Model Selection and Akaike's Information Criterion (AIC): The General Theory and Its Analytical Extensions," *Psychometrika*, 52, 345–370.

Breslow, N. E. and Clayton, D. G. (1993), "Approximate Inference in Generalized Linear Mixed Models," *Journal of the American Statistical Association*, 88, 9–25.

Breslow, N. E. and Lin, X. (1995), "Bias Correction in Generalised Linear Mixed Models with a Single Component of Dispersion," *Biometrika*, 81, 81–91.

Brinkman, N. D. (1981), "Ethanol Fuel: A Single-Cylinder Engine Study of Efficiency and Exhaust Emissions," *Society of Automotive Engineers Transactions*, 90, 1410–1424.

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

Burdick, R. K. and Graybill, F. A. (1992), *Confidence Intervals on Variance Components*, New York: Marcel Dekker.

Burnham, K. P. and Anderson, D. R. (1998), *Model Selection and Inference: A Practical Information-Theoretic Approach*, New York: Springer-Verlag.

Cameron, A. C. and Trivedi, P. K. (1998), *Regression Analysis of Count Data*, Cambridge: Cambridge University Press.

Clayton, D. G. and Kaldor, J. (1987), "Empirical Bayes Estimates of Age-Standardized Relative Risks for Use in Disease Mapping," *Biometrics*, 43, 671–681.

Cleveland, W. S. and Grosse, E. (1991), "Computational Methods for Local Regression," *Statistics and Computing*, 1, 47–62.

Cockerham, C. C. and Weir, B. S. (1977), "Quadratic Analyses of Reciprocal Crosses," *Biometrics*, 33, 187–203.

Davis, A. W. (1977), "A Differential Equation Approach to Linear Combinations of Independent Chi-Squares," *Journal of the American Statistical Association*, 72, 212–214.

De Boor, C. (2001), *A Practical Guide to Splines*, Rev. Edition, New York: Springer-Verlag.

Dennis, J. E., Gay, D. M., and Welsch, R. E. (1981), "An Adaptive Nonlinear Least-Squares Algorithm," *ACM Transactions on Mathematical Software*, 7, 348–368.

Dennis, J. E. and Mei, H. H. W. (1979), "Two New Unconstrained Optimization Algorithms Which Use Function and Gradient Values," *Journal of Optimization Theory and Applications*, 28, 453–482.

Dennis, J. E. and Schnabel, R. B. (1983), *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*, Englewood Cliffs, NJ: Prentice-Hall.

Diggle, P. J., Liang, K.-Y., and Zeger, S. L. (1994), *Analysis of Longitudinal Data*, Oxford: Clarendon Press.

Dunnett, C. W. (1980), "Pairwise Multiple Comparisons in the Unequal Variance Case," *Journal of the American Statistical Association*, 75, 796–800.

Edwards, D. and Berry, J. J. (1987), "The Efficiency of Simulation-Based Multiple Comparisons," *Biometrics*, 43, 913–928.

Eilers, P. H. C. and Marx, B. D. (1996), "Flexible Smoothing with *B*-Splines and Penalties," *Statistical Science*, 11, 89–121, with discussion.

Eskow, E. and Schnabel, R. B. (1991), "Algorithm 695: Software for a New Modified Cholesky Factorization," *ACM Transactions on Mathematical Software*, 17, 306–312.

Evans, G. (1993), *Practical Numerical Integration*, New York: John Wiley & Sons.

Fai, A. H. T. and Cornelius, P. L. (1996), "Approximate *F*-Tests of Multiple Degree of Freedom Hypotheses in Generalized Least Squares Analyses of Unbalanced Split-Plot Experiments," *Journal of Statistical Computation and Simulation*, 54, 363–378.

Fay, M. P. and Graubard, B. I. (2001), "Small-Sample Adjustments for Wald-Type Tests Using Sandwich Estimators," *Biometrics*, 57, 1198–1206.

Ferrari, S. L. P. and Cribari-Neto, F. (2004), "Beta Regression for Modelling Rates and Proportions," *Journal of Applied Statistics*, 31, 799–815.

Fisher, R. A. (1936), "The Use of Multiple Measurements in Taxonomic Problems," *Annals of Eugenics*, 7, 179–188.

Fletcher, R. (1987), *Practical Methods of Optimization*, 2nd Edition, Chichester, UK: John Wiley & Sons.

Friedman, J. H., Bentley, J. L., and Finkel, R. A. (1977), "An Algorithm for Finding Best Matches in Logarithmic Expected Time," *ACM Transactions on Mathematical Software*, 3, 209–226.

Fuller, W. A. (1976), *Introduction to Statistical Time Series*, New York: John Wiley & Sons.

Games, P. A. and Howell, J. F. (1976), "Pairwise Multiple Comparison Procedures with Unequal *n*'s and/or Variances: A Monte Carlo Study," *Journal of Educational Statistics*, 1, 113–125.

Gay, D. M. (1983), "Subroutines for Unconstrained Minimization," *ACM Transactions on Mathematical Software*, 9, 503–524.

Giesbrecht, F. G. and Burns, J. C. (1985), "Two-Stage Analysis Based on a Mixed Model: Large-Sample Asymptotic Theory and Small-Sample Simulation Results," *Biometrics*, 41, 477–486.

Gilliland, D. and Schabenberger, O. (2001), "Limits on Pairwise Association for Equi-correlated Binary Variables," *Journal of Applied Statistical Sciences*, 10, 279–285.

Gilmour, A. R., Anderson, R. D., and Rae, A. L. (1987), "Variance Components on an Underlying Scale for Ordered Multiple Threshold Categorical Data Using a Generalized Linear Mixed Model," *Journal of Animal Breeding and Genetics*, 104, 149–155.

Golub, G. H. and Welsch, J. H. (1969), "Calculation of Gaussian Quadrature Rules," *Mathematical Computing*, 23, 221–230.

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

Goodnight, J. H. (1978b), *Tests of Hypotheses in Fixed-Effects Linear Models*, Technical Report R-101, SAS Institute Inc., Cary, NC.

Goodnight, J. H. (1979), "A Tutorial on the Sweep Operator," *American Statistician*, 33, 149–158.

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

Gotway, C. A. and Stroup, W. W. (1997), "A Generalized Linear Model Approach to Spatial Data and Prediction," *Journal of Agricultural, Biological, and Environmental Statistics*, 2, 157–187.

Guirguis, G. H. and Tobias, R. D. (2004), "On the Computation of the Distribution for the Analysis of Means," *Communications in Statistics—Simulation and Computation*, 33, 861–888.

Hand, D. J., Daly, F., Lunn, A. D., McConway, K. J., and Ostrowski, E. (1994), *A Handbook of Small Data Sets*, London: Chapman & Hall.

Handcock, M. S. and Stein, M. L. (1993), "A Bayesian Analysis of Kriging," *Technometrics*, 35, 403–410.

Handcock, M. S. and Wallis, J. R. (1994), "An Approach to Statistical Spatial-Temporal Modeling of Meteorological Fields (with Discussion)," *Journal of the American Statistical Association*, 89, 368–390.

Hannan, E. J. and Quinn, B. G. (1979), "The Determination of the Order of an Autoregression," *Journal of the Royal Statistical Society, Series B*, 41, 190–195.

Harville, D. A. and Jeske, D. R. (1992), "Mean Squared Error of Estimation or Prediction under a General Linear Model," *Journal of the American Statistical Association*, 87, 724–731.

Hastie, T. J., Tibshirani, R. J., and Friedman, J. H. (2001), *The Elements of Statistical Learning*, New York: Springer-Verlag.

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

Henderson, C. R. (1984), *Applications of Linear Models in Animal Breeding*, Guelph, ON: University of Guelph.

Hinkley, D. V. (1977), "Jackknifing in Unbalanced Situations," *Technometrics*, 19, 285–292.

Hirotsu, C. and Srivastava, M. (2000), "Simultaneous Confidence Intervals Based on One-Sided Max *t* Test," *Statistics and Probability Letters*, 49, 25–37.

Holm, S. (1979), "A Simple Sequentially Rejective Multiple Test Procedure," *Scandinavian Journal of Statistics*, 6, 65–70.

Hsu, J. C. (1992), "The Factor Analytic Approach to Simultaneous Inference in the General Linear Model," *Journal of Computational and Graphical Statistics*, 1, 151–168.

Hsu, J. C. (1996), *Multiple Comparisons: Theory and Methods*, London: Chapman & Hall.

Hsu, J. C. and Peruggia, M. (1994), "Graphical Representation of Tukey's Multiple Comparison Method," *Journal of Computational and Graphical Statistics*, 3, 143–161.

Huber, P. J. (1967), "The Behavior of Maximum Likelihood Estimates under Nonstandard Conditions," *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, 1, 221–233.

Hurvich, C. M. and Tsai, C.-L. (1989), "Regression and Time Series Model Selection in Small Samples," *Biometrika*, 76, 297–307.

Huynh, H. and Feldt, L. S. (1970), "Conditions Under Which Mean Square Ratios in Repeated Measurements Designs Have Exact F-Distributions," *Journal of the American Statistical Association*, 65, 1582–1589.

Jennrich, R. I. and Schluchter, M. D. (1986), "Unbalanced Repeated-Measures Models with Structured Covariance Matrices," *Biometrics*, 42, 805–820.

Joe, H. and Zhu, R. (2005), "Generalized Poisson Distribution: The Property of Mixture of Poisson and Comparison with Negative Binomial Distribution," *Biometrical Journal*, 47, 219–229.

Johnson, N. L., Kotz, S., and Balakrishnan, N. (1994), *Continuous Univariate Distributions*, volume 1, 2nd Edition, New York: John Wiley & Sons.

Kackar, R. N. and Harville, D. A. (1984), "Approximations for Standard Errors of Estimators of Fixed and Random Effects in Mixed Linear Models," *Journal of the American Statistical Association*, 79, 853–862.

Kahaner, D., Moler, C. B., and Nash, S. (1989), *Numerical Methods and Software*, Englewood Cliffs, NJ: Prentice-Hall.

Karim, M. R. and Zeger, S. L. (1992), "Generalized Linear Models with Random Effects; Salamander Mating Revisited," *Biometrics*, 48, 631–644.

Kass, R. E. and Steffey, D. (1989), "Approximate Bayesian Inference in Conditionally Independent Hierarchical Models (Parametric Empirical Bayes Models)," *Journal of the American Statistical Association*, 84, 717–726.

Kauermann, G. and Carroll, R. J. (2001), "A Note on the Efficiency of Sandwich Covariance Estimation," *Journal of the American Statistical Association*, 96, 1387–1396.

Kenward, M. G. (1987), "A Method for Comparing Profiles of Repeated Measurements," *Applied Statistics*, 36, 296–308.

Kenward, M. G. and Roger, J. H. (1997), "Small Sample Inference for Fixed Effects from Restricted Maximum Likelihood," *Biometrics*, 53, 983–997.

Kenward, M. G. and Roger, J. H. (2009), "An Improved Approximation to the Precision of Fixed Effects from Restricted Maximum Likelihood," *Computational Statistics and Data Analysis*, 53, 2583–2595.

Koch, G. G., Carr, G. J., Amara, I. A., Stokes, M. E., and Uryniak, T. J. (1990), "Categorical Data Analysis," in D. A. Berry, ed., *Statistical Methodology in the Pharmaceutical Sciences*, New York: Marcel Dekker.

Kramer, C. Y. (1956), "Extension of Multiple Range Tests to Group Means with Unequal Numbers of Replications," *Biometrics*, 12, 307–310.

Lange, K. (1999), *Numerical Analysis for Statisticians*, New York: Springer-Verlag.

Liang, K.-Y. and Zeger, S. L. (1986), "Longitudinal Data Analysis Using Generalized Linear Models," *Biometrika*, 73, 13–22.

Lin, X. and Breslow, N. E. (1996), "Bias Correction in Generalized Linear Mixed Models with Multiple Components of Dispersion," *Journal of the American Statistical Association*, 91, 1007–1016.

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

Long, J. S. and Ervin, L. H. (2000), "Using Heteroscedasticity Consistent Standard Errors in the Linear Regression Model," *American Statistician*, 54, 217–224.

Macchiavelli, R. E. and Arnold, S. F. (1994), "Variable Order Ante-dependence Models," *Communications in Statistics—Theory and Methods*, 23, 2683–2699.

MacKinnon, J. G. and White, H. (1985), "Some Heteroskedasticity-Consistent Covariance Matrix Estimators with Improved Finite Sample Properties," *Journal of Econometrics*, 29, 305–325.

Mancl, L. A. and DeRouen, T. A. (2001), "A Covariance Estimator for GEE with Improved Small-Sample Properties," *Biometrics*, 57, 126–134.

Matérn, B. (1986), *Spatial Variation*, 2nd Edition, New York: Springer-Verlag.

McCullagh, P. (1980), "Regression Models for Ordinal Data," *Journal of the Royal Statistical Society, Series B*, 42, 109–142.

McCullagh, P. and Nelder, J. A. (1989), *Generalized Linear Models*, 2nd Edition, London: Chapman & Hall.

McLean, R. A. and Sanders, W. L. (1988), "Approximating Degrees of Freedom for Standard Errors in Mixed Linear Models," in *Proceedings of the Statistical Computing Section*, 50–59, Alexandria, VA: American Statistical Association.

McLean, R. A., Sanders, W. L., and Stroup, W. W. (1991), "A Unified Approach to Mixed Linear Models," *American Statistician*, 45, 54–64.

Milliken, G. A. and Johnson, D. E. (1992), *Designed Experiments*, volume 1 of *Analysis of Messy Data*, New York: Chapman & Hall.

Moré, J. J. (1978), "The Levenberg-Marquardt Algorithm: Implementation and Theory," in G. A. Watson, ed., *Lecture Notes in Mathematics*, volume 30, 105–116, Berlin: Springer-Verlag.

Moré, J. J. and Sorensen, D. C. (1983), "Computing a Trust-Region Step," *SIAM Journal on Scientific and Statistical Computing*, 4, 553–572.

Morel, J. G. (1989), "Logistic Regression under Complex Survey Designs," *Survey Methodology*, 15, 203–223.

Morel, J. G., Bokossa, M. C., and Neerchal, N. K. (2003), "Small Sample Correction for the Variance of GEE Estimators," *Biometrical Journal*, 4, 395–409.

Moriguchi, S., ed. (1976), *Statistical Method for Quality Control*, Tokyo: Japan Standards Association, in Japanese.

Mosteller, F. and Tukey, J. W. (1977), *Data Analysis and Regression*, Reading, MA: Addison-Wesley.

Murray, D. M., Varnell, S. P., and Blitstein, J. L. (2004), "Design and Analysis of Group-Randomized Trials: A Review of Recent Methodological Developments," *American Journal of Public Health*, 94, 423–432.

National Institute of Standards and Technology (1998), "Statistical Reference Data Sets," `http://www.itl.nist.gov/div898/strd/general/dataarchive.html`, accessed June 6, 2011.

Nelder, J. A. and Wedderburn, R. W. M. (1972), "Generalized Linear Models," *Journal of the Royal Statistical Society, Series A*, 135, 370–384.

Nelson, P. R. (1982), "Exact Critical Points for the Analysis of Means," *Communications in Statistics—Theory and Methods*, 11, 699–709.

Nelson, P. R. (1991), "Numerical Evaluation of Multivariate Normal Integrals with Correlations $\rho_{lj} = -\alpha_l \alpha_j$," in A. Öztürk and E. C. van der Meulen, eds., *Frontiers of Statistical Scientific Theory and Industrial Applications: Proceedings of the ICOSCO I Conference*, 97–114, Columbus, OH: American Sciences Press.

Nelson, P. R. (1993), "Additional Uses for the Analysis of Means and Extended Tables of Critical Values," *Technometrics*, 35, 61–71.

Ott, E. R. (1967), "Analysis of Means: A Graphical Procedure," *Industrial Quality Control*, 24, 101–109, reprinted in *Journal of Quality Technology*, 15 (1983), 10–18.

Patel, H. I. (1991), "Analysis of Incomplete Data from a Clinical Trial with Repeated Measurements," *Biometrika*, 78, 609–619.

Pawitan, Y. (2001), *In All Likelihood: Statistical Modelling and Inference Using Likelihood*, Oxford: Clarendon Press.

Pfeffermann, D., Skinner, C. J., Holmes, D. J., Goldstein, H., and Rasbash, J. (1998), "Weighting for Unequal Selection Probabilities in Multilevel Models," *Journal of the Royal Statistical Society, Series B*, 60, 23–40.

Pinheiro, J. C. and Bates, D. M. (1995), "Approximations to the Log-Likelihood Function in the Nonlinear Mixed-Effects Model," *Journal of Computational and Graphical Statistics*, 4, 12–35.

Pinheiro, J. C. and Chao, E. C. (2006), "Efficient Laplacian and Adaptive Gaussian Quadrature Algorithms for Multilevel Generalized Linear Mixed Models," *Journal of Computational and Graphical Statistics*, 15, 58–81.

Polak, E. (1971), *Computational Methods in Optimization*, New York: Academic Press.

Pothoff, R. F. and Roy, S. N. (1964), "A Generalized Multivariate Analysis of Variance Model Useful Especially for Growth Curve Problems," *Biometrika*, 51, 313–326.

Powell, M. J. D. (1977), "Restart Procedures for the Conjugate Gradient Method," *Mathematical Programming*, 12, 241–254.

Prasad, N. G. N. and Rao, J. N. K. (1990), "The Estimation of Mean Squared Error of Small-Area Estimators," *Journal of the American Statistical Association*, 85, 163–171.

Pringle, R. M. and Rayner, A. A. (1971), *Generalized Inverse Matrices with Applications to Statistics*, New York: Hafner Publishing.

Rabe-Hesketh, S. and Skrondal, A. (2006), "Multilevel Modelling of Complex Survey Data," *Journal of the Royal Statistical Society, Series A*, 169, 805–827.

Raudenbush, S. M., Yang, M.-L., and Yosef, M. (2000), "Maximum Likelihood for Generalized Linear Models with Nested Random Effects via Higher-Order, Multivariate Laplace Approximation," *Journal of Computational and Graphical Statistics*, 9, 141–157.

Royen, T. (1989), "Generalized Maximum Range Tests for Pairwise Comparisons of Several Populations," *Biometrical Journal*, 31, 905–929.

Ruppert, D., Wand, M. P., and Carroll, R. J. (2003), *Semiparametric Regression*, Cambridge: Cambridge University Press.

Saxton, A., ed. (2004), *Genetic Analysis of Complex Traits Using SAS*, Cary, NC: SAS Institute Inc.

Schabenberger, O. and Gregoire, T. G. (1996), "Population-Averaged and Subject-Specific Approaches for Clustered Categorical Data," *Journal of Statistical Computation and Simulation*, 54, 231–253.

Schabenberger, O., Gregoire, T. G., and Kong, F. (2000), "Collections of Simple Effects and Their Relationship to Main Effects and Interactions in Factorials," *American Statistician*, 54, 210–214.

Schabenberger, O. and Pierce, F. J. (2002), *Contemporary Statistical Models for the Plant and Soil Sciences*, Boca Raton, FL: CRC Press.

Schall, R. (1991), "Estimation in Generalized Linear Models with Random Effects," *Biometrika*, 78, 719–727.

Schluchter, M. D. and Elashoff, J. D. (1990), "Small-Sample Adjustments to Tests with Unbalanced Repeated Measures Assuming Several Covariance Structures," *Journal of Statistical Computation and Simulation*, 37, 69–87.

Schwarz, G. (1978), "Estimating the Dimension of a Model," *Annals of Statistics*, 6, 461–464.

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

Self, S. G. and Liang, K.-Y. (1987), "Asymptotic Properties of Maximum Likelihood Estimators and Likelihood Ratio Tests under Nonstandard Conditions," *Journal of the American Statistical Association*, 82, 605–610.

Shaffer, J. P. (1986), "Modified Sequentially Rejective Multiple Test Procedures," *Journal of the American Statistical Association*, 81, 826–831.

Shapiro, A. (1988), "Towards a Unified Theory of Inequality Constrained Testing in Multivariate Analysis," *International Statistical Review*, 56, 49–62.

Shun, Z. (1997), "Another Look at the Salamander Mating Data: A Modified Laplace Approximation Approach," *Journal of the American Statistical Association*, 92, 341–349.

Shun, Z. and McCullagh, P. (1995), "Laplace Approximation of High Dimensional Integrals," *Journal of the Royal Statistical Society, Series B*, 57, 749–760.

Silvapulle, M. J. and Sen, P. K. (2004), *Constrained Statistical Inference: Order, Inequality, and Shape Constraints*, New York: John Wiley & Sons.

Silvapulle, M. J. and Silvapulle, P. (1995), "A Score Test against One-Sided Alternatives," *Journal of the American Statistical Association*, 429, 342–349.

Stenstrom, F. H. (1940), *The Growth of Snapdragons, Stocks, Cinerarias, and Carnations on Six Iowa Soils*, Master's thesis, Iowa State College.

Stram, D. O. and Lee, J. W. (1994), "Variance Components Testing in the Longitudinal Mixed Effects Model," *Biometrics*, 50, 1171–1177.

Stram, D. O. and Lee, J. W. (1995), "Correction to 'Variance Components Testing in the Longitudinal Mixed Effects Model'," *Biometrics*, 51, 1196.

Tamhane, A. C. (1979), "A Comparison of Procedures for Multiple Comparisons of Means with Unequal Variances," *Journal of the American Statistical Association*, 74, 471–480.

Thall, P. F. and Vail, S. C. (1990), "Some Covariance Models for Longitudinal Count Data with Overdispersion," *Biometrics*, 46, 657–671.

Verbeke, G. and Molenberghs, G. (2000), *Linear Mixed Models for Longitudinal Data*, New York: Springer.

Verbeke, G. and Molenberghs, G. (2003), "The Use of Score Tests for Inference on Variance Components," *Biometrics*, 59, 254–262.

Vonesh, E. F. (1996), "A Note on Laplace's Approximation for Nonlinear Mixed-Effects Models," *Biometrika*, 83, 447–452.

Vonesh, E. F. and Chinchilli, V. M. (1997), *Linear and Nonlinear Models for the Analysis of Repeated Measurements*, New York: Marcel Dekker.

Vonesh, E. F., Chinchilli, V. M., and Pu, K. (1996), "Goodness-of-Fit in Generalized Nonlinear Mixed-Effects Models," *Biometrics*, 52, 572–587.

Wedderburn, R. W. M. (1974), "Quasi-likelihood Functions, Generalized Linear Models, and the Gauss-Newton Method," *Biometrika*, 61, 439–447.

Westfall, P. H. (1997), "Multiple Testing of General Contrasts Using Logical Constraints and Correlations," *Journal of the American Statistical Association*, 92, 299–306.

Westfall, P. H. and Tobias, R. D. (2007), "Multiple Testing of General Contrasts: Truncated Closure and the Extended Shaffer-Royen Method," *Journal of the American Statistical Association*, 478, 487–494.

Westfall, P. H., Tobias, R. D., Rom, D., Wolfinger, R. D., and Hochberg, Y. (1999), *Multiple Comparisons and Multiple Tests Using the SAS System*, Cary, NC: SAS Institute Inc.

Westfall, P. H. and Young, S. S. (1993), *Resampling-Based Multiple Testing: Examples and Methods for* p-*Value Adjustment*, New York: John Wiley & Sons.

White, H. (1980), "A Heteroskedasticity-Consistent Covariance Matrix Estimator and a Direct Test for Heteroskedasticity," *Econometrica*, 48, 817–838.

White, H. (1982), "Maximum Likelihood Estimation of Misspecified Models," *Econometrica*, 50, 1–25.

Whittle, P. (1954), "On Stationary Processes in the Plane," *Biometrika*, 41, 434–449.

Winer, B. J. (1971), *Statistical Principles in Experimental Design*, 2nd Edition, New York: McGraw-Hill.

Wolfinger, R. D. (1993), "Laplace's Approximation for Nonlinear Mixed Models," *Biometrika*, 80, 791–795.

Wolfinger, R. D. and O'Connell, M. A. (1993), "Generalized Linear Mixed Models: A Pseudo-likelihood Approach," *Journal of Statistical Computation and Simulation*, 48, 233–243.

Wolfinger, R. D., Tobias, R. D., and Sall, J. (1994), "Computing Gaussian Likelihoods and Their Derivatives for General Linear Mixed Models," *SIAM Journal on Scientific Computing*, 15, 1294–1310.

Zeger, S. L. and Liang, K.-Y. (1986), "Longitudinal Data Analysis for Discrete and Continuous Outcomes," *Biometrics*, 42, 121–130.

Subject Index

adaptive Gaussian quadrature
    GLIMMIX procedure, 3073
Akaike's information criterion
    GLIMMIX procedure, 3069
Akaike's information criterion (finite sample corrected version)
    GLIMMIX procedure, 3069
alpha level
    GLIMMIX procedure, 3100, 3106, 3115, 3126, 3134, 3149, 3156
anisotropic power covariance structure
    GLIMMIX procedure, 3170
anisotropic spatial power structure
    GLIMMIX procedure, 3170
ANOM adjustment
    GLIMMIX procedure, 3113
anom plot
    GLIMMIX procedure, 3256
ANTE(1) structure
    GLIMMIX procedure, 3162
ante-dependence structure
    GLIMMIX procedure, 3162
AR(1) structure
    GLIMMIX procedure, 3162
asymptotic covariance
    GLIMMIX procedure, 3066, 3069
automatic variables
    GLIMMIX procedure, 3110, 3178
autoregressive moving-average structure
    GLIMMIX procedure, 3163
autoregressive structure
    GLIMMIX procedure, 3162

banded Toeplitz structure
    GLIMMIX procedure, 3171
Bernoulli distribution
    GLIMMIX procedure, 3137
beta distribution
    GLIMMIX procedure, 3137
Between-Within method
    GLIMMIX procedure, 3210
bias
    GLIMMIX procedure, 3199
binary distribution
    GLIMMIX procedure, 3137
binomial distribution
    GLIMMIX procedure, 3137
BLUP
    GLIMMIX procedure, 3161, 3178, 3246
Bonferroni adjustment
    GLIMMIX procedure, 3113
boundary constraints
    GLIMMIX procedure, 3151, 3154
box plots
    GLIMMIX procedure, 3251

centering
    GLIMMIX procedure, 3142
chi-square mixture
    GLIMMIX procedure, 3102
chi-square test
    GLIMMIX procedure, 3095, 3127, 3134
Cholesky
    covariance structure (GLIMMIX), 3163
    method (GLIMMIX), 3066
    root (GLIMMIX), 3163, 3165
class level
    GLIMMIX procedure, 3078, 3242
comparing splines
    GLIMMIX procedure, 3365
compound symmetry structure
    GLIMMIX procedure, 3164
computed variables
    GLIMMIX procedure, 3110
confidence limits
    adjusted (GLIMMIX), 3105, 3113, 3126, 3320
    adjusted, simulated (GLIMMIX), 3114
    and isotronic contrasts (GLIMMIX), 3320
    and step-down (GLIMMIX), 3109, 3123, 3130
    covariance parameters (GLIMMIX), 3100
    estimate, lower (GLIMMIX), 3109
    estimate, upper (GLIMMIX), 3110
    estimated likelihood (GLIMMIX), 3100
    estimates (GLIMMIX), 3106
    exponentiated (GLIMMIX), 3128
    fixed effects (GLIMMIX), 3134
    in mean plot (GLIMMIX), 3084, 3120
    inversely linked (GLIMMIX), 3117, 3246
    least squares mean estimate (GLIMMIX), 3126
    least squares mean estimate, lower (GLIMMIX), 3130
    least squares mean estimate, upper (GLIMMIX), 3131
    least squares means (GLIMMIX), 3062, 3113, 3116
    least squares means estimates (GLIMMIX), 3127

likelihood-based, details (GLIMMIX), 3208
odds ratios (GLIMMIX), 3079, 3087, 3142, 3226
profile likelihood (GLIMMIX), 3100
random-effects solution (GLIMMIX), 3156
truncation (GLIMMIX), 3247
vs. prediction limits (GLIMMIX), 3246
Wald (GLIMMIX), 3101
constraints
boundary (GLIMMIX), 3151, 3154
constructed effects
GLIMMIX procedure, 3103, 3371
Containment method
GLIMMIX procedure, 3211
contrast-specification
GLIMMIX procedure, 3092, 3104
contrasts
GLIMMIX procedure, 3092
control plot
GLIMMIX procedure, 3256
convergence criterion
GLIMMIX procedure, 3065, 3074, 3081, 3238, 3243, 3244, 3269, 3313
MIXED procedure, 3238
convergence status
GLIMMIX procedure, 3244
covariance
parameters (GLIMMIX), 3056, 3060, 3065
parameters, confidence interval (GLIMMIX), 3096
parameters, testing (GLIMMIX), 3096
covariance parameter estimates
GLIMMIX procedure, 3072, 3081, 3245
covariance structure
anisotropic power (GLIMMIX), 3170
ante-dependence (GLIMMIX), 3162
autoregressive (GLIMMIX), 3055, 3162
autoregressive moving-average (GLIMMIX), 3163
banded (GLIMMIX), 3171
Cholesky type (GLIMMIX), 3163
compound symmetry (GLIMMIX), 3164
equi-correlation (GLIMMIX), 3164
examples (GLIMMIX), 3172
exponential (GLIMMIX), 3169
factor-analytic (GLIMMIX), 3165
G-side (GLIMMIX), 3055, 3097, 3162
Gaussian (GLIMMIX), 3170
general (GLIMMIX), 3053
general linear (GLIMMIX), 3166
heterogeneous autoregressive (GLIMMIX), 3163
heterogeneous compound symmetry (GLIMMIX), 3165
heterogeneous Toeplitz (GLIMMIX), 3171
Huynh-Feldt (GLIMMIX), 3166
Matérn (GLIMMIX), 3170
misspecified (GLIMMIX), 3053
parameter reordering (GLIMMIX), 3151
penalized B-spline (GLIMMIX), 3158, 3160, 3167
positive (semi-)definite, 3163
power (GLIMMIX), 3170
R-side (GLIMMIX), 3055, 3097, 3151, 3155, 3161, 3162
R-side with profiled scale (GLIMMIX), 3151
radial smooth (GLIMMIX), 3158, 3168
simple (GLIMMIX), 3169
spatial (GLIMMIX), 3157, 3169
spherical (GLIMMIX), 3171
Toeplitz (GLIMMIX), 3171
unstructured (GLIMMIX), 3171
unstructured, correlation (GLIMMIX), 3172
variance components (GLIMMIX), 3172
with second derivatives (GLIMMIX), 3213
working, independence (GLIMMIX), 3068
crossed effects
GLIMMIX procedure, 3230

default estimation technique
GLIMMIX procedure, 3241
default output
GLIMMIX procedure, 3241
degrees of freedom
chi-square mixture (GLIMMIX), 3102
GLIMMIX procedure, 3094, 3095, 3105, 3106, 3116, 3127, 3135, 3136, 3210
infinite (GLIMMIX), 3106, 3116, 3127
method (GLIMMIX), 3136
degrees of freedom method
GLIMMIX procedure, 3210
diagnostic plots
GLIMMIX procedure, 3251
diffogram
GLIMMIX procedure, 3256
dimension information
GLIMMIX procedure, 3242
dispersion parameter
GLIMMIX procedure, 3184
doubly iterative algorithm
GLIMMIX procedure, 3239
Dunnett's adjustment
GLIMMIX procedure, 3113

EBE
GLIMMIX procedure, 3161, 3246
EBLUP
GLIMMIX procedure, 3161
effect
name length (GLIMMIX), 3077

empirical Bayes estimates
GLIMMIX procedure, 3074, 3161, 3246
empirical Bayes estimation
GLIMMIX procedure, 3074
empirical estimator
GLIMMIX procedure, 3066, 3213, 3216
estimability
GLIMMIX procedure, 3094
estimates
GLIMMIX procedure, 3104
multiple comparison adjustment (GLIMMIX), 3105
estimation methods
GLIMMIX procedure, 3072
examples, GLIMMIX
adding computed variables to output data set, 3110
analysis of means, ANOM, 3265
analysis of summary data, 3376
anom plot, 3119
anom plots, 3265
binary data, 3304
binary data, GLMM, 3279
binary data, pseudo-likelihood, 3279
binary data, sort order, 3080
binomial data, 3059
binomial data, GLM, 3268
binomial data, GLMM, 3272
binomial data, overdispersed, 3296
binomial data, spatial covariance, 3275
bivariate data; Poisson, binary, 3305
blotch incidence data, 3294
box plots, 3254
bucket size in *k*-d tree, 3222
central *t* distribution, 3139
Cholesky covariance structure, 3322
collection effect, 3231
computed variables, 3287, 3311
constructed random effect, 3372
containment hierarchy, 3073, 3219
contrast, among covariance parameters, 3099, 3100, 3330
contrast, differences of splines, 3235
contrast, nonpositional syntax, 3093, 3233, 3368
contrast, positional syntax, 3093, 3233
contrast, with groups, 3095
contrast, with spline effects, 3234
control plot, 3119, 3263
covariance structure, 3172
covariates in LS-mean construction, 3115
COVTEST statement, 3102
COVTEST with keywords, 3099
COVTEST with no restrictions, 3206
COVTEST with specified values, 3099
cow weight data, 3309
diallel experiment, 3371
diffogram, 3120, 3259, 3261, 3283
diffplot, 3120
empirical Bayes estimates, 3340
epileptic seizure data, 3345
equivalent models, TYPE=VC, 3218
equivalent models, with and without subject, 3218
estimate, multi-row, 3107
estimate, with groups, 3107
estimate, with varied divisors, 3107
ferrite cores data, 3320
FIRSTORDER option for Kenward-Roger method, 3324
foot shape data, 3339
FREQ statement, 3340
G-side spatial covariance, 3157
GEE-type model, 3068, 3203, 3348
generalized logit, 3095, 3106
generalized logit with random effects, 3236
generalized Poisson distribution, 3362
getting started, 3057
GLM mode, 3203
GLMM mode, 3203
graphics, anom plots, 3265
graphics, box plots, 3254
graphics, control plot, 3263
graphics, custom template, 3290
graphics, diffogram, 3259, 3261, 3283
graphics, mean plots, 3257
graphics, Pearson residual panel, 3298
graphics, predicted profiles, 3314
graphics, residual panel, 3251
graphics, studentized residual panel, 3289
group option in contrast, 3095
group option in estimate, 3107
group-specific smoothing, 3316
grouped analysis, 3340
groups in RANDOM statement, 3097, 3327
herniorrhaphy data, 3303
Hessian fly data, 3267
holding covariance parameters fixed, 3151
homogeneity of covariance parameters, 3097, 3099, 3327
identity model, 3376
infinite degrees of freedom, 3346
inverse linking, 3062, 3129
isotonic contrast, 3320
joint model (DIST=BYOBS), 3138
joint model, independent, 3305
joint model, marginal correlation, 3307
joint model, shared random effect, 3306
k-d tree information, 3222
Kenward-Roger method, 3322

knot construction, equal, 3159
knot construction, *k*-d tree, 3159, 3222, 3311
knot construction, optimization, 3159
Laplace approximation, 3194, 3198, 3354
LDATA= option, 3166
least squares mean estimate, 3125, 3320
least squares mean estimate, multi-row, 3127
least squares mean estimate, with varied divisors, 3127
least squares means, 3062
least squares means, AT option, 3115
least squares means, covariate, 3115
least squares means, differences against control, 3116
least squares means, slice, 3121
least squares means, slice differences, 3122
linear combination of LS-means, 3125
linear covariance structure, 3166, 3376
LINP, 3178–3180
logistic model with random effects, binomial data, 3272
logistic model, binomial data, 3268
logistic regression with random intercepts, 3057
logistic regression, binary data, 3226, 3304
logistic regression, binomial data, 3240
LOGL, 3362
marginal variance matrix, 3174
mean plot, sliced interaction, 3121
mean plot, three-way, 3121
mean plots, 3257
MIVQUE0 estimates, 3153
_MU_, 3179, 3180
multicenter clinical trial, 3349
multimember effect, 3231, 3372
multinomial data, 3095, 3106, 3236, 3340, 3354
multiple local minima, 3337
multiple plot requests, 3085
multiplicity adjustment, 3320, 3351, 3354, 3370
multivariate distributions, 3138
multivariate normal model, 3327
nesting v. crossing, 3219
NLIN procedure, 3375
NLOPTIONS statement, 3311
NOFIT option, 3222
NOITER option for covariance parameters, 3152
nonlinear regression, 3375
NOPROFILE option, 3334
odds ratio, 3142, 3227
odds ratio, all pairwise differences, 3143, 3228
odds ratio, with interactions, 3142, 3228
odds ratio, with reference value, 3143, 3228
odds ratio, with specified units, 3143, 3228
ordinal data, 3340, 3354
OUTDESIGN option, 3366, 3372
output statistics, 3146, 3178, 3287, 3311, 3366
overdispersion, 3056, 3068, 3155, 3203, 3296
parallel shifted smooths, 3317
Pearson residual panel, 3298
penalized B-spline, 3167
Poisson model with offset, 3287, 3346
Poisson model with random effects, 3359
Poisson regression, 3305
Pothoff-Roy repeated measures data, 3321
proportional odds model with random effect, 3340, 3354
quadrature approximation, 3196, 3340, 3359
quasi-likelihood, 3299
R-side covariance structure, 3322, 3348
R-side covariance, binomial data, 3275
radial smooth, with parallel shifts, 3317
radial smoothing, 3222, 3311, 3334
radial smoothing, group-specific, 3316
REPEATED in MIXED vs RANDOM in GLIMMIX, 3237
residual panel, 3251
row-wise adjustment of LS-mean differences, 3113
salamander data, 3277
Satterthwaite method, 3113
saturated model, 3376
Scottish lip cancer data, 3285
SGPANEL procedure, 3314
SGPLOT procedure, 3336, 3337, 3344, 3365, 3367
SGRENDER procedure, 3290
simple differences, 3122, 3283
simple differences with control, 3123
simulated *p*-values, 3351, 3354, 3370
simulated survey data, 3379
slice differences, 3122, 3283
slice differences with control, 3123
slice *F* test, 3121
space-filling design, 3159
spatial covariance, binomial data, 3275
specifying lower bounds, 3151
specifying values for degrees of freedom, 3135
spline differences, 3370
spline effect, 3232, 3366
splines in interactions, 3370
standardized mortality rate, 3287
starting values, 3337
starting values and BY groups, 3154
starting values from data set, 3153
step-down *p*-values, 3351, 3354, 3370
studentized maximum modulus, 3113
studentized residual panel, 3289
subject processing, 3073, 3218
subject processing, containment, 3219

subject processing, crossed effects, 3219
subject processing, nested effects, 3219
subject-processing, asymptotics, 3194
syntax, differences to MIXED, 3237, 3238
test for independence, 3343
test for Poisson distribution, 3362
testing covariance parameters, 3098, 3327, 3330, 3334
theophylline data, 3373
TYPE=CS and TYPE=VC equivalence, 3164
user-defined log-likelihood function, 3362
user-defined variance function, 3198
user-specified link function, 3179, 3180
user-specified variance function, 3180, 3299
VARIANCE, 3180, 3299
working independence, 3068, 3203
expansion locus
theory (GLIMMIX), 3192
exponential covariance structure
GLIMMIX procedure, 3169
exponential distribution
GLIMMIX procedure, 3137

factor-analytic structure
GLIMMIX procedure, 3165
finite differences
theory (GLIMMIX), 3197
Fisher's scoring method
GLIMMIX procedure, 3066, 3088
fit statistics
GLIMMIX procedure, 3244
fitting information
GLIMMIX procedure, 3244
fixed effects
GLIMMIX procedure, 3054
frequency variable
GLIMMIX procedure, 3110

G matrix
GLIMMIX procedure, 3155–3157
G-side random effect
GLIMMIX procedure, 3055
gamma distribution
GLIMMIX procedure, 3137
Gaussian covariance structure
GLIMMIX procedure, 3170
Gaussian distribution
GLIMMIX procedure, 3137
GEE, *see also* generalized estimating equations
general linear covariance structure
GLIMMIX procedure, 3166
generalized estimating equations
compound symmetry (GLIMMIX), 3348
working independence (GLIMMIX), 3068, 3203
generalized linear mixed model, *see also* GLIMMIX procedure
generalized linear mixed model (GLIMMIX)
least squares means, 3124
theory, 3186
generalized linear model, *see also* GLIMMIX procedure
generalized linear model (GLIMMIX)
theory, 3180
generalized logit
example (GLIMMIX), 3095, 3106
generalized Poisson distribution
GLIMMIX procedure, 3357
geometric distribution
GLIMMIX procedure, 3137
GLIMMIX procedure
adaptive Gaussian quadrature, 3073
Akaike's information criterion, 3069
Akaike's information criterion (finite sample corrected version), 3069
alpha level, 3100, 3106, 3115, 3126, 3134, 3149, 3156
anisotropic power covariance structure, 3170
anisotropic spatial power structure, 3170
ANOM adjustment, 3113
anom plot, 3256
ANTE(1) structure, 3162
ante-dependence structure, 3162
AR(1) structure, 3162
asymptotic covariance, 3066, 3069
automatic variables, 3110, 3178
autoregressive moving-average structure, 3163
autoregressive structure, 3162
banded Toeplitz structure, 3171
Bernoulli distribution, 3137
beta distribution, 3137
Between-Within method, 3210
bias of estimates, 3199
binary distribution, 3137
binomial distribution, 3137
BLUP, 3161, 3178, 3246
Bonferroni adjustment, 3113
boundary constraints, 3151, 3154
box plots, 3251
BYLEVEL processing of LSMEANS, 3115, 3118, 3127
centering, 3142
chi-square mixture, 3102
chi-square test, 3095, 3127, 3134
Cholesky covariance structure, 3163
Cholesky method, 3066
Cholesky root, 3163, 3165
class level, 3078, 3242
comparing splines, 3365

comparison with the MIXED procedure, 3236
compound symmetry structure, 3164
computed variables, 3110
confidence interval, 3108, 3130, 3156
confidence limits, 3106, 3116, 3127, 3134, 3156
confidence limits, covariance parameters, 3100
constrained covariance parameters, 3103
constructed effects, 3103, 3371
Containment method, 3211
continuous effects, 3162
contrast-specification, 3092, 3104
contrasts, 3092
control plot, 3256
convergence criterion, 3065, 3074, 3081, 3238, 3243, 3244, 3269, 3313
convergence status, 3244
correlations of least squares means, 3116
correlations of least squares means contrasts, 3127
covariance parameter estimates, 3072, 3245
covariance parameters, 3056
covariance structure, 3162, 3172
covariances of least squares means, 3116
covariances of least squares means contrasts, 3127
covariate values for LSMEANS, 3115, 3127
crossed effects, 3230
default estimation technique, 3241
default output, 3241
default variance function, 3177
degrees of freedom, 3094, 3095, 3102, 3104–3106, 3111, 3112, 3116, 3126, 3127, 3135, 3136, 3144, 3210, 3230
degrees of freedom method, 3210
diagnostic plots, 3251
diffogram, 3120, 3256, 3259, 3261, 3283
dimension information, 3242
dispersion parameter, 3184
doubly iterative algorithm, 3239
Dunnett's adjustment, 3113
EBE, 3161, 3246
EBLUP, 3161
effect name length, 3077
empirical Bayes estimates, 3074, 3161, 3246
empirical Bayes estimation, 3074
empirical estimator, 3216
estimability, 3094, 3096, 3108, 3111, 3145, 3231
estimated-likelihood interval, 3100
estimates, 3104
estimation methods, 3072
estimation modes, 3203
examples, *see also* examples, GLIMMIX, 3266
expansion locus, 3192
exponential covariance structure, 3169
exponential distribution, 3137
factor-analytic structure, 3165
finite differences, 3197
Fisher's scoring method, 3066, 3088
fit statistics, 3244
fitting information, 3244
fixed effects, 3054
fixed-effects parameters, 3144
G matrix, 3155–3157
G-side random effect, 3055
gamma distribution, 3137
Gaussian covariance structure, 3170
Gaussian distribution, 3137
general linear covariance structure, 3166
generalized linear mixed model theory, 3186
generalized linear model theory, 3180
generalized Poisson distribution, 3357
geometric distribution, 3137
GLM mode, 3068, 3184, 3203, 3204
GLMM mode, 3068, 3203, 3204
grid search, 3150
group effect, 3157
Hannan-Quinn information criterion, 3069
Hessian matrix, 3066, 3069
heterogeneous AR(1) structure, 3163
heterogeneous autoregressive structure, 3163
heterogeneous compound symmetry structure, 3165
heterogeneous Toeplitz structure, 3171
Hsu's adjustment, 3113
Huynh-Feldt covariance structure, 3166
infinite degrees of freedom, 3095, 3106, 3116, 3127
information criteria, 3069
initial values, 3150
input data sets, 3066
integral approximation, 3186
interaction effects, 3230
intercept, 3230
intercept random effect, 3155
introductory example, 3057
inverse Gaussian distribution, 3137
iteration details, 3071
iteration history, 3243
iterations, 3243
Kackar-Harville-Jeske adjusted estimator, 3216
Kenward-Roger method, 3212
knot selection, 3221
KR adjusted estimator, 3216
L matrices, 3092, 3111
lag functionality, 3176
Laplace approximation, 3072, 3192
least squares means, 3111, 3116, 3122
likelihood ratio test, 3096, 3204

linear covariance structure, 3166
linearization, 3186, 3188
link function, 3054, 3140
log-normal distribution, 3137
marginal residuals, 3149
Matérn covariance structure, 3170
maximum likelihood, 3072, 3241
missing level combinations, 3231
MIVQUE0 estimation, 3153, 3243
mixed model smoothing, 3157, 3158, 3160, 3167, 3168, 3220
model information, 3242
multimember effect, 3371
multimember example, 3371
multinomial distribution, 3137
multiple comparisons of estimates, 3105
multiple comparisons of least squares means, 3112, 3113, 3116, 3122, 3126
multiplicity adjustment, 3105, 3108, 3112, 3113, 3123, 3126, 3130
negative binomial distribution, 3137
Nelson's adjustment, 3113
nested effects, 3230
non-full-rank parameterization, 3230
non-positional syntax, 3094, 3232, 3365
normal distribution, 3137
notation, 3054
number of observations, 3242
numerical integration, 3195
odds estimation, 3225
odds ratio estimation, 3225
odds ratios, 3079
ODS graph names, 3250
ODS Graphics, 3081, 3249
ODS table names, 3247
offset, 3144, 3178, 3287, 3288, 3346
optimization, 3145
optimization information, 3243
output statistics, 3246
overdispersion, 3357
P-spline, 3167
parameterization, 3229
penalized B-spline, 3167
Poisson distribution, 3137
Poisson mixture, 3357
population average, 3192
positive definiteness, 3163
power covariance structure, 3170
profile-likelihood interval, 3100
profiling residual variance, 3078, 3088
programming statements, 3175
pseudo-likelihood, 3072, 3241
quadrature approximation, 3072, 3195
quasi-likelihood, 3241
R-side random effect, 3055, 3161
radial smoother structure, 3168
radial smoothing, 3158, 3168, 3220
random effects, 3054, 3155
random-effects parameter, 3161
reference category, 3235
residual effect, 3155
residual likelihood, 3072
residual maximum likelihood, 3241
residual plots, 3251
response level ordering, 3133, 3235
response profile, 3235, 3242
response variable options, 3133
restricted maximum likelihood, 3241
sandwich estimator, 3216
Satterthwaite method, 3211
scale parameter, 3055, 3056, 3070, 3078, 3088, 3101, 3150, 3151, 3154, 3177, 3180, 3183–3185, 3189, 3190, 3192, 3195, 3208, 3239, 3243, 3245, 3279, 3296, 3299, 3312, 3315, 3341, 3358
Schwarz's Bayesian information criterion, 3069
scoring, 3066
Sidak's adjustment, 3113
simple covariance matrix, 3169
simple effects, 3121
simple effects differences, 3122
simulation-based adjustment, 3114
singly iterative algorithm, 3239
spatial covariance structure, 3169
spatial exponential structure, 3169
spatial Gaussian structure, 3170
spatial Matérn structure, 3170
spatial power structure, 3170
spatial spherical structure, 3171
spherical covariance structure, 3171
spline comparisons, 3365
spline smoothing, 3167, 3168
standard error adjustment, 3066
statistical graphics, 3249, 3250
subject effect, 3161
subject processing, 3218
subject-specific, 3192
t distribution, 3137
table names, 3247
test-specification for covariance parameters, 3097
testing covariance parameters, 3096, 3204
tests of fixed effects, 3245
thin plate spline (approx.), 3220
Toeplitz structure, 3171
Tukey's adjustment, 3113
Type I testing, 3140
Type II testing, 3140
Type III testing, 3140

unstructured covariance, 3171
unstructured covariance matrix, 3165
user-defined link function, 3177
V matrix, 3174
Wald test, 3245
Wald tests of covariance parameters, 3103
weight, 3144, 3174
weighted multilevel models, 3200
weighting, 3175
GLIMMIX procedure
ordering of effects, 3079
GLM, *see also* GLIMMIX procedure
GLMM, *see also* GLIMMIX procedure
group effect
GLIMMIX procedure, 3157

Hannan-Quinn information criterion
GLIMMIX procedure, 3069
Hessian matrix
GLIMMIX procedure, 3066, 3069
heterogeneous AR(1) structure
GLIMMIX procedure, 3163
heterogeneous autoregressive structure
GLIMMIX procedure, 3163
heterogeneous compound symmetry structure
GLIMMIX procedure, 3165
heterogeneous Toeplitz structure
GLIMMIX procedure, 3171
Hsu's adjustment
GLIMMIX procedure, 3113
Huynh-Feldt
stucture (GLIMMIX), 3166

infinite degrees of freedom
GLIMMIX procedure, 3095, 3106, 3116, 3127
information criteria
GLIMMIX procedure, 3069
initial values
GLIMMIX procedure, 3150
integral approximation
theory (GLIMMIX), 3186
interaction effects
GLIMMIX procedure, 3230
intercept
GLIMMIX procedure, 3230
inverse Gaussian distribution
GLIMMIX procedure, 3137
iteration details
GLIMMIX procedure, 3071
iteration history
GLIMMIX procedure, 3243
iterations
history (GLIMMIX), 3243

Kenward-Roger method
GLIMMIX procedure, 3212
knot selection
GLIMMIX procedure, 3221

L matrices
GLIMMIX procedure, 3092, 3111
mixed model (GLIMMIX), 3092, 3111
lag functionality
GLIMMIX procedure, 3176
Laplace approximation
GLIMMIX procedure, 3072
theory (GLIMMIX), 3192
least squares means
Bonferroni adjustment (GLIMMIX), 3113
BYLEVEL processing (GLIMMIX), 3115, 3118, 3127
comparison types (GLIMMIX), 3116, 3122
covariate values (GLIMMIX), 3115, 3127
Dunnett's adjustment (GLIMMIX), 3113
generalized linear mixed model (GLIMMIX), 3111, 3124
Hsu's adjustment (GLIMMIX), 3113
multiple comparison adjustment (GLIMMIX), 3112, 3113, 3126
Nelson's adjustment (GLIMMIX), 3113
observed margins (GLIMMIX), 3118, 3130
Scheffe's adjustment (GLIMMIX), 3113
Sidak's adjustment (GLIMMIX), 3113
simple effects (GLIMMIX), 3121
simple effects differences (GLIMMIX), 3122
simulation-based adjustment (GLIMMIX), 3114
Tukey's adjustment (GLIMMIX), 3113
likelihood ratio test
GLIMMIX procedure, 3096, 3204
linear covariance structure
GLIMMIX procedure, 3166
linearization
theory (GLIMMIX), 3186, 3188
link function
GLIMMIX procedure, 3054, 3140
user-defined (GLIMMIX), 3177
log-normal distribution
GLIMMIX procedure, 3137

marginal residuals
GLIMMIX procedure, 3149
Matérn covariance structure
GLIMMIX procedure, 3170
maximum likelihood
GLIMMIX procedure, 3072, 3241
MBN adjusted sandwich estimators
GLIMMIX procedure, 3215
missing level combinations
GLIMMIX procedure, 3231

MIVQUE0 estimation
GLIMMIX procedure, 3153, 3243
mixed model (GLIMMIX)
parameterization, 3229
mixed model smoothing
GLIMMIX procedure, 3157, 3158, 3160, 3167, 3168, 3220
MIXED procedure
comparison with the GLIMMIX procedure, 3236
convergence criterion, 3238
mixture
chi-square (GLIMMIX), 3096, 3102
chi-square, weights (GLIMMIX), 3103
Poisson (GLIMMIX), 3357
model
information (GLIMMIX), 3242
multimember effect
GLIMMIX procedure, 3371
multimember example
GLIMMIX procedure, 3371
multinomial distribution
GLIMMIX procedure, 3137
multiple comparison adjustment (GLIMMIX)
estimates, 3105
least squares means, 3112, 3113, 3126
multiple comparisons of estimates
GLIMMIX procedure, 3105
multiple comparisons of least squares means
GLIMMIX procedure, 3112, 3113, 3116, 3122, 3126
multiplicity adjustment
Bonferroni (GLIMMIX), 3113, 3126
Dunnett (GLIMMIX), 3113
estimates (GLIMMIX), 3105
GLIMMIX procedure, 3105
Hsu (GLIMMIX), 3113
least squares means (GLIMMIX), 3113
least squares means estimates (GLIMMIX), 3126
Nelson (GLIMMIX), 3113
row-wise (GLIMMIX), 3105, 3112
Scheffe (GLIMMIX), 3113, 3126
Sidak (GLIMMIX), 3113, 3126
Simulate (GLIMMIX), 3126
simulation-based (GLIMMIX), 3114
step-down *p*-values (GLIMMIX), 3108, 3123, 3130
T (GLIMMIX), 3126
Tukey (GLIMMIX), 3113

negative binomial distribution
GLIMMIX procedure, 3137
Nelson's adjustment
GLIMMIX procedure, 3113
nested effects
GLIMMIX procedure, 3230
non-full-rank parameterization
GLIMMIX procedure, 3230
non-positional syntax
GLIMMIX procedure, 3094, 3232, 3365
normal distribution
GLIMMIX procedure, 3137
notation
GLIMMIX procedure, 3054
number of observations
GLIMMIX procedure, 3242
numerical integration
theory (GLIMMIX), 3195

odds estimation
GLIMMIX procedure, 3225
odds ratio estimation
GLIMMIX procedure, 3225
ODS graph names
GLIMMIX procedure, 3250
ODS Graphics
GLIMMIX procedure, 3081, 3249
offset
GLIMMIX procedure, 3144, 3178, 3287, 3288, 3346
optimization
GLIMMIX procedure, 3145
optimization information
GLIMMIX procedure, 3243
options summary
LSMEANS statement, (GLIMMIX), 3112
MODEL statement (GLIMMIX), 3132
PROC GLIMMIX statement, 3064
RANDOM statement (GLIMMIX), 3155
output statistics
GLIMMIX procedure, 3246
overdispersion
GLIMMIX procedure, 3357

P-spline
GLIMMIX procedure, 3167
parameterization
GLIMMIX procedure, 3229
mixed model (GLIMMIX), 3229
penalized B-spline
GLIMMIX procedure, 3167
Poisson distribution
GLIMMIX procedure, 3137
Poisson mixture
GLIMMIX procedure, 3357
positive definiteness
GLIMMIX procedure, 3163
power covariance structure
GLIMMIX procedure, 3170

probability distributions
GLIMMIX procedure, 3137
PROC GLIMMIX procedure
residual variance tolerance, 3089
programming statements
GLIMMIX procedure, 3175
pseudo-likelihood
GLIMMIX procedure, 3072, 3241

quadrature approximation
GLIMMIX procedure, 3072
theory (GLIMMIX), 3195
quasi-likelihood
GLIMMIX procedure, 3241

R-side random effect
GLIMMIX procedure, 3055
radial smoother structure
GLIMMIX procedure, 3168
radial smoothing
GLIMMIX procedure, 3158, 3168, 3220
random effects
GLIMMIX procedure, 3054, 3155
reference category
GLIMMIX procedure, 3235
residual likelihood
GLIMMIX procedure, 3072
residual plots
GLIMMIX procedure, 3251
residual-based sandwich estimators
GLIMMIX procedure, 3213
response level ordering
GLIMMIX procedure, 3133, 3235
response profile
GLIMMIX procedure, 3235, 3242
response variable options
GLIMMIX procedure, 3133
restricted maximum likelihood
GLIMMIX procedure, 3241
reverse response level ordering
GLIMMIX procedure, 3133

sandwich estimator, *see also* empirical estimator
GLIMMIX procedure, 3066, 3213, 3216
Satterthwaite method
GLIMMIX procedure, 3211
scale parameter
GLIMMIX compared to GENMOD, 3184
GLIMMIX procedure, 3055, 3056, 3070, 3078, 3088, 3101, 3150, 3151, 3154, 3177, 3180, 3183, 3185, 3189, 3190, 3192, 3195, 3208, 3239, 3243, 3245, 3279, 3296, 3299, 3312, 3315, 3341, 3358
Schwarz's Bayesian information criterion
GLIMMIX procedure, 3069
scoring
GLIMMIX procedure, 3066
Sidak's adjustment
GLIMMIX procedure, 3113
simple covariance matrix
GLIMMIX procedure, 3169
simple effects
GLIMMIX procedure, 3121
simple effects differences
GLIMMIX procedure, 3122
simulation-based adjustment
GLIMMIX procedure, 3114
singly iterative algorithm
GLIMMIX procedure, 3239
spatial covariance structure
GLIMMIX procedure, 3169
spatial exponential structure
GLIMMIX procedure, 3169
spatial Gaussian structure
GLIMMIX procedure, 3170
spatial Matérn structure
GLIMMIX procedure, 3170
spatial power structure
GLIMMIX procedure, 3170
spatial spherical structure
GLIMMIX procedure, 3171
spherical covariance structure
GLIMMIX procedure, 3171
spline comparisons
GLIMMIX procedure, 3365
spline smoothing
GLIMMIX procedure, 3167, 3168
statistical graphics
GLIMMIX procedure, 3249, 3250
subject effect
GLIMMIX procedure, 3161
subject processing
GLIMMIX procedure, 3218

t distribution
GLIMMIX procedure, 3137
table names
GLIMMIX procedure, 3247
test-specification for covariance parameters
GLIMMIX procedure, 3097
testing covariance parameters
GLIMMIX procedure, 3096, 3204
tests of fixed effects
GLIMMIX procedure, 3245
theophylline data
examples, GLIMMIX, 3373
thin plate spline (approx.)
GLIMMIX procedure, 3220
Toeplitz structure

GLIMMIX procedure, 3171
Tukey's adjustment
GLIMMIX procedure, 3113
Type I testing
GLIMMIX procedure, 3140
Type II testing
GLIMMIX procedure, 3140
Type III testing
GLIMMIX procedure, 3140

unstructured covariance
GLIMMIX procedure, 3171
unstructured covariance matrix
GLIMMIX procedure, 3165

V matrix
GLIMMIX procedure, 3174
variance function
GLIMMIX procedure, 3177
user-defined (GLIMMIX), 3177

Wald test
GLIMMIX procedure, 3245
Wald tests of covariance parameters
GLIMMIX procedure, 3103
weight
GLIMMIX procedure, 3144, 3174
weighting
GLIMMIX procedure, 3175

Syntax Index

ABSPCONV option
    PROC GLIMMIX statement, 3065
ADJDFE= option
    ESTIMATE statement (GLIMMIX), 3105
    LSMEANS statement (GLIMMIX), 3112
    LSMESTIMATE statement (GLIMMIX), 3126
ADJUST= option
    ESTIMATE statement (GLIMMIX), 3105
    LSMEANS statement (GLIMMIX), 3113
    LSMESTIMATE statement (GLIMMIX), 3126
ALLSTATS option
    OUTPUT statement (GLIMMIX), 3149
ALPHA= option
    ESTIMATE statement (GLIMMIX), 3106
    LSMEANS statement (GLIMMIX), 3115
    LSMESTIMATE statement (GLIMMIX), 3126
    MODEL statement (GLIMMIX), 3134
    OUTPUT statement (GLIMMIX), 3149
    RANDOM statement (GLIMMIX), 3156
ASYCORR option
    PROC GLIMMIX statement, 3066
ASYCOV option
    PROC GLIMMIX statement, 3066
AT MEANS option
    LSMEANS statement (GLIMMIX), 3115
    LSMESTIMATE statement (GLIMMIX), 3127
AT option
    LSMEANS statement (GLIMMIX), 3115
    LSMESTIMATE statement (GLIMMIX), 3127

BUCKET= suboption
    RANDOM statement (GLIMMIX), 3158
BY statement
    GLIMMIX procedure, 3090
BYCAT option
    CONTRAST statement (GLIMMIX), 3095
    ESTIMATE statement (GLIMMIX), 3106
BYCATEGORY option
    CONTRAST statement (GLIMMIX), 3095
    ESTIMATE statement (GLIMMIX), 3106
BYLEVEL option
    LSMEANS statement (GLIMMIX), 3115
    LSMESTIMATE statement (GLIMMIX), 3127

CHISQ option
    CONTRAST statement (GLIMMIX), 3095
    LSMESTIMATE statement (GLIMMIX), 3127
    MODEL statement (GLIMMIX), 3134
CHOL option
    PROC GLIMMIX statement, 3066
CHOLESKY option
    PROC GLIMMIX statement, 3066
CL option
    COVTEST statement (GLIMMIX), 3100
    ESTIMATE statement (GLIMMIX), 3106
    LSMEANS statement (GLIMMIX), 3116
    LSMESTIMATE statement (GLIMMIX), 3127
    MODEL statement (GLIMMIX), 3134
    RANDOM statement (GLIMMIX), 3156
CLASS statement
    GLIMMIX procedure, 3091
CLASSICAL option
    COVTEST statement (GLIMMIX), 3102
CODE statement
    GLIMMIX procedure, 3092
CONTRAST option
    COVTEST statement (GLIMMIX), 3099
CONTRAST statement
    GLIMMIX procedure, 3092
CORR option
    LSMEANS statement (GLIMMIX), 3116
    LSMESTIMATE statement (GLIMMIX), 3127
CORRB option
    MODEL statement (GLIMMIX), 3135
COV option
    LSMEANS statement (GLIMMIX), 3116
    LSMESTIMATE statement (GLIMMIX), 3127
COVB option
    MODEL statement (GLIMMIX), 3135
COVBI option
    MODEL statement (GLIMMIX), 3135
COVTEST statement
    GLIMMIX procedure, 3096
CPSEUDO option
    OUTPUT statement (GLIMMIX), 3149

DATA= option
    PROC GLIMMIX statement, 3066
DDF= option
    MODEL statement (GLIMMIX), 3135
DDFM= option
    MODEL statement (GLIMMIX), 3136
DER option
    OUTPUT statement (GLIMMIX), 3149
DERIVATIVES option
    OUTPUT statement (GLIMMIX), 3149
DESCENDING option

MODEL statement, 3133
DF= option
CONTRAST statement (GLIMMIX), 3095
COVTEST statement (GLIMMIX), 3102
ESTIMATE statement (GLIMMIX), 3106
LSMEANS statement (GLIMMIX), 3116
LSMESTIMATE statement (GLIMMIX), 3127
MODEL statement (GLIMMIX), 3135
DIFF option
LSMEANS statement (GLIMMIX), 3116
DIST= option
MODEL statement (GLIMMIX), 3137
DISTRIBUTION= option
MODEL statement (GLIMMIX), 3137
DIVISOR= option
ESTIMATE statement (GLIMMIX), 3107
LSMESTIMATE statement (GLIMMIX), 3127

E option
CONTRAST statement (GLIMMIX), 3095
ESTIMATE statement (GLIMMIX), 3107
LSMEANS statement (GLIMMIX), 3117
LSMESTIMATE statement (GLIMMIX), 3128
MODEL statement (GLIMMIX), 3140
E1 option
MODEL statement (GLIMMIX), 3140
E2 option
MODEL statement (GLIMMIX), 3140
E3 option
MODEL statement (GLIMMIX), 3140
EFFECT statement
GLIMMIX procedure, 3103
ELSM option
LSMESTIMATE statement (GLIMMIX), 3128
EMPIRICAL= option
PROC GLIMMIX statement, 3066
ERROR= option
MODEL statement (GLIMMIX), 3137
ESTIMATE statement
GLIMMIX procedure, 3104
ESTIMATES option
COVTEST statement (GLIMMIX), 3103
EVENT= option
MODEL statement, 3133
EXP option
ESTIMATE statement (GLIMMIX), 3107
LSMESTIMATE statement (GLIMMIX), 3128
EXPHESSIAN option
PROC GLIMMIX statement, 3069

FDIGITS= option
PROC GLIMMIX statement, 3069
FREQ statement
GLIMMIX procedure, 3110
FTEST option
LSMESTIMATE statement (GLIMMIX), 3128

G option
RANDOM statement (GLIMMIX), 3156
GC option
RANDOM statement (GLIMMIX), 3157
GCI option
RANDOM statement (GLIMMIX), 3157
GCOORD= option
RANDOM statement (GLIMMIX), 3157
GCORR option
RANDOM statement (GLIMMIX), 3157
GENERAL option
COVTEST statement (GLIMMIX), 3099
GI option
RANDOM statement (GLIMMIX), 3157
GLIMMIX procedure, 3063
CONTRAST statement, 3092
COVTEST statement, 3096
EFFECT statement, 3103
ESTIMATE statement, 3104
FREQ statement, 3110
ID statement, 3110
LSMEANS statement, 3111
LSMESTIMATE statement, 3124
MODEL statement, 3131
NLOPTIONS statement, 3145
OUTPUT statement, 3146
PARMS statement, 3150
PROC GLIMMIX statement, 3064
Programming statements, 3175
RANDOM statement, 3155
syntax, 3063
WEIGHT statement, 3175
GLIMMIX procedure, BY statement, 3090
GLIMMIX procedure, CONTRAST statement, 3092
BYCAT option, 3095
BYCATEGORY option, 3095
CHISQ option, 3095
DF= option, 3095
E option, 3095
GROUP option, 3095
SINGULAR= option, 3096
SUBJECT option, 3096
GLIMMIX procedure, COVTEST statement, 3096
CL option, 3100
CLASSICAL option, 3102
CONTRAST option, 3099
ESTIMATES option, 3103
GENERAL option, 3099
MAXITER= option, 3103
PARMS option, 3103
RESTART option, 3103

TESTDATA= option, 3098
TOLERANCE= option, 3103
WALD option, 3103
WGHT= option, 3103
GLIMMIX procedure, DF= statement
CLASSICAL option, 3102
GLIMMIX procedure, EFFECT statement, 3103
GLIMMIX procedure, ESTIMATE statement, 3104
ADJDFE= option, 3105
ADJUST= option, 3105
ALPHA= option, 3106
BYCAT option, 3106
BYCATEGORY option, 3106
CL option, 3106
DF= option, 3106
DIVISOR= option, 3107
E option, 3107
EXP option, 3107
GROUP option, 3107
ILINK option, 3108
LOWERTAILED option, 3108
SINGULAR= option, 3108
STEPDOWN option, 3108
SUBJECT option, 3109
UPPERTAILED option, 3110
GLIMMIX procedure, FREQ statement, 3110
GLIMMIX procedure, ID statement, 3110
GLIMMIX procedure, LSMEANS statement, 3111
ADJUST= option, 3113
ALPHA= option, 3115
AT MEANS option, 3115
AT option, 3115
BYLEVEL option, 3115
CL option, 3116
CORR option, 3116
COV option, 3116
DF= option, 3116
DIFF option, 3116
E option, 3117
ILINK option, 3117
LINES option, 3117
OBSMARGINS option, 3118
ODDS option, 3118
ODDSRATIO option, 3118
OM option, 3118
PDIFF option, 3116, 3118
PLOT option, 3118
PLOTS option, 3118
SIMPLEDIFF= option, 3122
SIMPLEDIFFTYPE option, 3122
SINGULAR= option, 3121
SLICE= option, 3121
SLICEDIFF= option, 3122
SLICEDIFFTYPE option, 3122
STEPDOWN option, 3123
GLIMMIX procedure, LSMESTIMATE statement, 3124
ADJUST= option, 3126
ALPHA= option, 3126
AT MEANS option, 3127
AT option, 3127
BYLEVEL option, 3127
CHISQ option, 3127
CL option, 3127
CORR option, 3127
COV option, 3127
DF= option, 3127
DIVISOR= option, 3127
E option, 3128
ELSM option, 3128
EXP option, 3128
FTEST option, 3128
ILINK option, 3129
JOINT option, 3128
LOWERTAILED option, 3130
OBSMARGINS option, 3130
OM option, 3130
SINGULAR= option, 3130
STEPDOWN option, 3130
UPPERTAILED option, 3131
GLIMMIX procedure, MODEL statement, 3131
ALPHA= option, 3134
CHISQ option, 3134
CL option, 3134
CORRB option, 3135
COVB option, 3135
COVBI option, 3135
DDF= option, 3135
DDFM= option, 3136
DESCENDING option, 3133
DF= option, 3135
DIST= option, 3137
DISTRIBUTION= option, 3137
E option, 3140
E1 option, 3140
E2 option, 3140
E3 option, 3140
ERROR= option, 3137
EVENT= option, 3133
HTYPE= option, 3140
INTERCEPT option, 3140
LINK= option, 3140
LWEIGHT= option, 3141
NOCENTER option, 3142
NOINT option, 3142, 3230
OBSWEIGHT= option, 3144
ODDSRATIO option, 3142
OFFSET= option, 3144

ORDER= option, 3133
REFERENCE= option, 3134
REFLINP= option, 3144
SOLUTION option, 3144, 3230
STDCOEF option, 3144
ZETA= option, 3145
GLIMMIX procedure, OUTPUT statement, 3146
ALLSTATS option, 3149
ALPHA= option, 3149
CPSEUDO option, 3149
DER option, 3149
DERIVATIVES option, 3149
keyword= option, 3146
NOMISS option, 3149
NOUNIQUE option, 3149
NOVAR option, 3150
OBSCAT option, 3150
OUT= option, 3146
SYMBOLS option, 3150
GLIMMIX procedure, PARMS statement, 3150
HOLD= option, 3151
LOWERB= option, 3151
NOBOUND option, 3152
NOITER option, 3152
PARMSDATA= option, 3153
PDATA= option, 3153
UPPERB= option, 3154
GLIMMIX procedure, PROC GLIMMIX statement, 3064
ABSPCONV option, 3065
ASYCORR option, 3066
ASYCOV option, 3066
CHOL option, 3066
CHOLESKY option, 3066
DATA= option, 3066
EMPIRICAL= option, 3066
EXPHESSIAN option, 3069
FDIGITS= option, 3069
GRADIENT option, 3069
HESSIAN option, 3069
IC= option, 3069
INFOCRIT= option, 3069
INITGLM option, 3071
INITITER option, 3071
ITDETAILS option, 3071
LIST option, 3071
MAXLMMUPDATE option, 3071
MAXOPT option, 3071
METHOD= option, 3072
NAMELEN= option, 3077
NOBOUND option, 3077
NOBSDETAIL option, 3078
NOCLPRINT option, 3078
NOFIT option, 3078
NOINITGLM option, 3078
NOITPRINT option, 3078
NOPROFILE option, 3078
NOREML option, 3078
ODDSRATIO option, 3079
OUTDESIGN option, 3080
PCONV option, 3081
PLOT option, 3081
PLOTS option, 3081
PROFILE option, 3088
SCOREMOD option, 3088
SCORING= option, 3088
SINGCHOL= option, 3089
SINGULAR= option, 3089
STARTGLM option, 3089
SUBGRADIENT option, 3089
GLIMMIX procedure, programming statements, 3175
ABORT statement, 3175
CALL statement, 3175
DELETE statement, 3175
DO statement, 3175
GOTO statement, 3175
IF statement, 3175
LINK statement, 3175
PUT statement, 3175
RETURN statement, 3175
SELECT statement, 3175
STOP statement, 3175
SUBSTR statement, 3175
WHEN statement, 3175
GLIMMIX procedure, RANDOM statement, 3155
ALPHA= option, 3156
CL option, 3156
G option, 3156
GC option, 3157
GCI option, 3157
GCOORD= option, 3157
GCORR option, 3157
GI option, 3157
GROUP= option, 3157
KNOTINFO option, 3157
KNOTMAX= option, 3157
KNOTMETHOD= option, 3158
KNOTMIN= option, 3160
LDATA= option, 3160
NOFULLZ option, 3161
RESIDUAL option, 3161
RSIDE option, 3161
SOLUTION option, 3161
SUBJECT= option, 3161
TYPE= option, 3162
V option, 3174
VC option, 3174
VCI option, 3174

VCORR option, 3174
VI option, 3174
WEIGHT= option, 3174
GLIMMIX procedure, WEIGHT statement, 3175
GLIMMIX procedure, CLASS statement, 3091
REF= option, 3091
REF= variable option, 3091
TRUNCATE option, 3092
GLIMMIX procedure, CODE statement, 3092
GLIMMIX procedure, PROC GLIMMIX statement
ORDER= option, 3079
GLIMMIX procedure, STORE statement, 3175
GRADIENT option
PROC GLIMMIX statement, 3069
GROUP option
CONTRAST statement (GLIMMIX), 3095
ESTIMATE statement (GLIMMIX), 3107
GROUP= option
RANDOM statement (GLIMMIX), 3157

HESSIAN option
PROC GLIMMIX statement, 3069
HOLD= option
PARMS statement (GLIMMIX), 3151
HTYPE= option
MODEL statement (GLIMMIX), 3140

IC= option
PROC GLIMMIX statement, 3069
ID statement
GLIMMIX procedure, 3110
ILINK option
ESTIMATE statement (GLIMMIX), 3108
LSMEANS statement (GLIMMIX), 3117
LSMESTIMATE statement (GLIMMIX), 3129
INFOCRIT= option
PROC GLIMMIX statement, 3069
INITGLM option
PROC GLIMMIX statement, 3071
INITITER option
PROC GLIMMIX statement, 3071
INTERCEPT option
MODEL statement (GLIMMIX), 3140
ITDETAILS option
PROC GLIMMIX statement, 3071

JOINT option
LSMESTIMATE statement (GLIMMIX), 3128

keyword= option
OUTPUT statement (GLIMMIX), 3146
KNOTINFO option
RANDOM statement (GLIMMIX), 3157
KNOTMAX= option
RANDOM statement (GLIMMIX), 3157
KNOTMETHOD= option
RANDOM statement (GLIMMIX), 3158
KNOTMIN= option
RANDOM statement (GLIMMIX), 3160
KNOTTYPE= suboption
RANDOM statement (GLIMMIX), 3158

LDATA= option
RANDOM statement (GLIMMIX), 3160
LINES option
LSMEANS statement (GLIMMIX), 3117
LINK= option
MODEL statement (GLIMMIX), 3140
LIST option
PROC GLIMMIX statement, 3071
LOWERB= option
PARMS statement (GLIMMIX), 3151
LOWERTAILED option
ESTIMATE statement (GLIMMIX), 3108
LSMESTIMATE statement (GLIMMIX), 3130
LSMEANS statement
GLIMMIX procedure, 3111
LSMESTIMATE statement
GLIMMIX procedure, 3124
LWEIGHT= option
MODEL statement (GLIMMIX), 3141

MAXITER= option
COVTEST statement (GLIMMIX), 3103
MAXLMMUPDATE option
PROC GLIMMIX statement, 3071
MAXOPT option
PROC GLIMMIX statement, 3071
METHOD= option
PROC GLIMMIX statement, 3072
MODEL statement
GLIMMIX procedure, 3131

NAMELEN= option
PROC GLIMMIX statement, 3077
NEAREST suboption
RANDOM statement (GLIMMIX), 3159
NLOPTIONS statement
GLIMMIX procedure, 3145
NOBOUND option
PARMS statement (GLIMMIX), 3152
PROC GLIMMIX statement, 3077
NOBSDETAIL option
PROC GLIMMIX statement, 3078
NOCENTER option
MODEL statement (GLIMMIX), 3142
NOCLPRINT option
PROC GLIMMIX statement, 3078
NOFIT option
PROC GLIMMIX statement, 3078

NOFULLZ option
RANDOM statement (GLIMMIX), 3161
NOINITGLM option
PROC GLIMMIX statement, 3078
NOINT option
MODEL statement (GLIMMIX), 3142, 3230
NOITER option
PARMS statement (GLIMMIX), 3152
NOITPRINT option
PROC GLIMMIX statement, 3078
NOMISS option
OUTPUT statement (GLIMMIX), 3149
NOPROFILE option
PROC GLIMMIX statement, 3078
NOREML option
PROC GLIMMIX statement, 3078
NOUNIQUE option
OUTPUT statement (GLIMMIX), 3149
NOVAR option
OUTPUT statement (GLIMMIX), 3150

OBSCAT option
OUTPUT statement (GLIMMIX), 3150
OBSMARGINS option
LSMEANS statement (GLIMMIX), 3118
LSMESTIMATE statement (GLIMMIX), 3130
OBSWEIGHT= option
MODEL statement (GLIMMIX), 3144
ODDS option
LSMEANS statement (GLIMMIX), 3118
ODDSRATIO option
LSMEANS statement (GLIMMIX), 3118
MODEL statement (GLIMMIX), 3142
PROC GLIMMIX statement, 3079
OFFSET= option
MODEL statement (GLIMMIX), 3144
OM option
LSMEANS statement (GLIMMIX), 3118
LSMESTIMATE statement (GLIMMIX), 3130
ORDER= option
MODEL statement, 3133
PROC GLIMMIX statement, 3079
OUT= option
OUTPUT statement (GLIMMIX), 3146
OUTDESIGN option
PROC GLIMMIX statement, 3080
OUTPUT statement
GLIMMIX procedure, 3146

PARMS option
COVTEST statement (GLIMMIX), 3103
PARMS statement
GLIMMIX procedure, 3150
PARMSDATA= option
PARMS statement (GLIMMIX), 3153
PCONV option
PROC GLIMMIX statement, 3081
PDATA= option
PARMS statement (GLIMMIX), 3153
PDIFF option
LSMEANS statement (GLIMMIX), 3116, 3118
PLOT option
LSMEANS statement (GLIMMIX), 3118
PROC GLIMMIX statement, 3081
PLOTS option
LSMEANS statement (GLIMMIX), 3118
PROC GLIMMIX statement, 3081
PROC GLIMMIX procedure, PROC GLIMMIX statement
SINGRES= option, 3089
PROC GLIMMIX statement, *see* GLIMMIX procedure
GLIMMIX procedure, 3064
PROFILE option
PROC GLIMMIX statement, 3088
Programming statements
GLIMMIX procedure, 3175

RANDOM statement
GLIMMIX procedure, 3155
RANDOM statement (GLIMMIX)
BUCKET= suboption, 3158
KNOTTYPE= suboption, 3158
NEAREST suboption, 3159
TREEINFO suboption, 3159
REF= option
CLASS statement (GLIMMIX), 3091
REFERENCE= option
MODEL statement, 3134
REFLINP= option
MODEL statement (GLIMMIX), 3144
RESIDUAL option
RANDOM statement (GLIMMIX), 3161
RESTART option
COVTEST statement (GLIMMIX), 3103
RSIDE option
RANDOM statement (GLIMMIX), 3161

SCOREMOD option
PROC GLIMMIX statement, 3088
SCORING= option
PROC GLIMMIX statement, 3088
SIMPLEDIFFTYPE option
LSMEANS statement (GLIMMIX), 3122
SIMPLEEDIFF= option
LSMEANS statement (GLIMMIX), 3122
SINGCHOL= option
PROC GLIMMIX statement, 3089
SINGRES= option

PROC GLIMMIX statement (GLIMMIX), 3089
SINGULAR= option
CONTRAST statement (GLIMMIX), 3096
ESTIMATE statement (GLIMMIX), 3108
LSMEANS statement (GLIMMIX), 3121
LSMESTIMATE statement (GLIMMIX), 3130
PROC GLIMMIX statement, 3089
SLICE= option
LSMEANS statement (GLIMMIX), 3121
SLICEDIFF= option
LSMEANS statement (GLIMMIX), 3122
SLICEDIFFTYPE option
LSMEANS statement (GLIMMIX), 3122
SOLUTION option
MODEL statement (GLIMMIX), 3144, 3230
RANDOM statement (GLIMMIX), 3161
STARTGLM option
PROC GLIMMIX statement, 3089
STDCOEF option
MODEL statement (GLIMMIX), 3144
STEPDOWN option
ESTIMATE statement (GLIMMIX), 3108
LSMEANS statement (GLIMMIX), 3123
LSMESTIMATE statement (GLIMMIX), 3130
STORE statement
GLIMMIX procedure, 3175
SUBGRADIENT option
PROC GLIMMIX statement, 3089
SUBJECT option
CONTRAST statement (GLIMMIX), 3096
ESTIMATE statement (GLIMMIX), 3109
SUBJECT= option
RANDOM statement (GLIMMIX), 3161
SYMBOLS option
OUTPUT statement (GLIMMIX), 3150

TESTDATA= option
COVTEST statement (GLIMMIX), 3098
TOLERANCE= option
COVTEST statement (GLIMMIX), 3103
TREEINFO suboption
RANDOM statement (GLIMMIX), 3159
TRUNCATE option
CLASS statement (GLIMMIX), 3092
TYPE= option
RANDOM statement (GLIMMIX), 3162

UPPERB= option
PARMS statement (GLIMMIX), 3154
UPPERTAILED option
ESTIMATE statement (GLIMMIX), 3110
LSMESTIMATE statement (GLIMMIX), 3131

V option
RANDOM statement (GLIMMIX), 3174
VC option
RANDOM statement (GLIMMIX), 3174
VCI option
RANDOM statement (GLIMMIX), 3174
VCORR option
RANDOM statement (GLIMMIX), 3174
VI option
RANDOM statement (GLIMMIX), 3174

WALD option
COVTEST statement (GLIMMIX), 3103
WEIGHT statement
GLIMMIX procedure, 3175
WEIGHT= option
RANDOM statement (GLIMMIX), 3174
WGHT= option
COVTEST statement (GLIMMIX), 3103

ZETA= option
MODEL statement (GLIMMIX), 3145