Insights into Using the GLIMMIX Procedure to Model Categorical Outcomes with Random Effects

Kathleen Kiernan, SAS Institute Inc.

ABSTRACT

Modeling categorical outcomes with random effects is a major use of the GLIMMIX procedure. Building, evaluating, and using the resulting model for inference, prediction, or both requires many considerations. This paper, written for experienced users of SAS® statistical procedures, illustrates the nuances of the process with two examples: modeling a binary response using random effects and correlated errors and modeling a multinomial response with random effects.

In addition, the paper provides answers to common questions that are received by SAS Technical Support concerning these analyses with PROC GLIMMIX. These questions cover working with events/trials data, handling bias issues in a logistic model, and overcoming convergence problems.

INTRODUCTION

A logistic regression model with random effects or correlated data occurs in a variety of disciplines. For example, subjects are followed over time, are repeatedly treated under different experimental conditions, or are observed in clinics, families, and litters. The LOGISTIC procedure is the standard tool in SAS for estimating logistic regression models with fixed effects. The GLIMMIX procedure provides the capability to estimate generalized linear mixed models (GLMM), including random effects and correlated errors.

For binary response models, PROC GLIMMIX can estimate fixed effects, random effects, and correlated errors models. PROC GLIMMIX also supports the estimation of fixed- and random-effect multinomial response models. However, the procedure does not support the estimation of correlated errors (R-side random effects) for multinomial response models.

This paper provides a brief review of modeling random effects in the GLIMMIX procedure. The paper also illustrates examples of using PROC GLIMMIX to estimate a binomial logistic model with random effects, a binomial model with correlated data, and a multinomial model with random effects. In addition, each example provides a list of commonly asked questions and answers that are related to estimating logistic regression models with PROC GLIMMIX. The final section includes a brief discussion for some of the commonly reported notes, warnings, and errors that are reported in the SAS log when you use PROC GLIMMIX to run an analysis.

MODELING RANDOM EFFECTS IN PROC GLIMMIX

A quick review of modeling random effects in PROC GLIMMIX might be helpful before discussing examples of modeling categorical outcomes with random effects. PROC GLIMMIX distinguishes two types of random effects. Depending on whether the parameters of the covariance structure for random components in your model are contained in the $\mathbf{G}$ matrix or the $\mathbf{R}$ matrix, the procedure distinguishes between G-side and R-side random effects. Consider the following terminology that draws from the common specification of the linear mixed model, where the random effects have a normal distribution with mean 0 and variance matrix $\mathbf{G}$:

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

The distribution of the errors ε is normal with mean 0 and variance matrix $\mathbf{R}$. Modeling with G-side effects, you specify the columns of the Z matrix and the covariance structure of matrix $\mathbf{G}$. Modeling with R-side effects, you directly specify the covariance structure of matrix $\mathbf{R}$. Simply put, if a random effect is an element of γ, 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 with only G-side effects are also known as *conditional (or subject-specific) models,* while models with R-side effects are known as

marginal (or population-averaged) models. Models fit with PROC GLIMMIX can have none, one, or more of each type of random effect.

Note that an R-side effect in PROC GLIMMIX is equivalent to a REPEATED effect in the MIXED procedure. The R-side covariance structure in PROC GLIMMIX is the covariance structure that you formulate with the REPEATED statement in the MIXED procedure. In PROC GLIMMIX, all random effects and their covariance structures are specified through the <u>RANDOM statement</u>.

To understand the model that is estimated in PROC GLIMMIX, it is important to recognize the different specifications for the RANDOM statement. For example, the following RANDOM statement defines a random coefficients model using a G-side random effect that creates a block diagonal matrix for each level of the ID variable with the default TYPE=VC covariance structure:

```
random intercept x1/ subject=ID;
```

Note that TYPE=VC or TYPE=UN are typical covariance structures that are used to model G-side correlation. TYPE=VC defines zero correlation between the random coefficients.

You use the _RESIDUAL_ keyword or the RESIDUAL option in the RANDOM statement to instruct the GLIMMIX procedure that a random effect models an R-side component. The following specification of the RANDOM statement defines an R-side random effect that correlates observations from a given ID with the TYPE=CS covariance structure.

```
random time / subject=ID residual type=cs;
```

You model the correlation of an R-side random effect by selecting a TYPE= covariance structure that is meaningful to your application and data. Most often the correlation for an R-side random effect is more complex than the default TYPE=VC covariance structure.

You can explicitly estimate an additional scale parameter with PROC GLIMMIX by using the following statement:

```
random _residual_;
```

If your code defines a generalized linear model (GLM), you can add the `random _residual_;` statement, and the scale parameter is displayed in the Solutions for the Fixed Effects table.

Keep in mind that the addition of a scale parameter does not change the fixed-effect parameter estimates in a generalized linear model (GLM). The extra scale parameter changes the standard errors of the fixed-effect parameter estimates in a GLM. However, in a generalized linear mixed model (GLMM), the addition of a scale parameter does change the fixed- and random-effect parameter estimates and the covariance parameter estimates. If you add the overdispersion parameter to a model with G-side random effects, then there is a redistribution of variability between R- and G-side variation compared to a model without the extra scale parameter. The extra scale parameter changes the magnitude of the G-side variance component estimates, and because the parameter estimates depend on the random-effects solutions, all associated results change. The addition of the extra scale parameter changes every aspect of the GLMM model.

The following example adds the extra scale parameter:

```
    /* Only the standard errors change. */
proc glimmix data=test;
    model y=x1 x2 / s dist=binomial link=logit;
    random _residual_;
run;
```

In PROC GLIMMIX, if you do not have the `random _residual_;` statement, then the scale parameter is assumed to be 1, regardless of whether you have overdispersion. Once you specify the `random _residual_;` statement, the overdispersion parameter is given by the Pearson chi-square / DF value, and the standard errors and related quantities are adjusted by this overdispersion value.

DATA EXAMPLES

The following sections illustrate specific examples of using PROC GLIMMIX to estimate a binomial logistic model with random effects, a binomial model with correlated data. and a multinomial model with random effects. Procedure code and results of the analysis are provided with respective interpretation. After each example, you will find a list of commonly asked questions and answers related to using PROC GLIMMIX to model categorical outcomes with random effects.

EXAMPLE 1: USING PROC GLIMMIX WITH BINOMIAL AND BINARY DATA

One of the more popular reasons to use PROC GLIMMIX is to model binary (yes/no, 0/1) outcomes with random effects. This first example analyzes the data from Beitler and Landis (1985), which represent results from a multi-center clinical trial that investigates the effectiveness of two topical cream treatments (active drug, control) in curing an infection. For each of eight clinics, the number of trials and favorable cures are recorded for each treatment.

The following DATA step creates the data set for the analysis in two forms (events/trials syntax and individual-level data). This DATA step creates the INFECTION input data set where the binomial response data are grouped such that x represents the number of events and n represents the number of trials.

```
data infection;
   input clinic treatment x n @@;
   datalines
1 1 11 36 3 1 14 19 5 1 6 17 7 1 1 5 2 1 16 20 4 1 2 16 6 1 1 11 8 1 4 6
1 0 10 37 3 0  7 19 5 0 0 12 7 0 1 9 2 0 22 32 4 0 1 17 6 0 0 10 8 0 6 7
   ;
```

To illustrate the use of an alternative form of the input data set, the following DATA step creates the INFECTION2 input data set. This data set expands the data in the INFECTION data set so that it represents the individual-level data. That is, each observation represents each person and the response (cure) in a single line.

```
data infection2(drop=i);
   set infection;
   do i=1 to n;
      if i<= x then cure=1;
      if i>x then cure=0;
      status=2-cure;
        output;
    end;
run;
```

The following code requests that PROC GLIMMIX fit this model for both input data sets using METHOD=QUAD estimation to obtain less biased estimates and goodness-of-fit statistics:

```
proc glimmix data=infection method=quad;
   class clinic treatment(ref='0');
   model x/n= treatment /s dist=binomial link=logit;
   random intercept/subject=clinic;
run;

proc glimmix data=infection2 method=quad;
   class clinic treatment(ref='0');
   model cure(Event='1')=treatment /s dist=binary link=logit;
   random intercept/subject=clinic;
run;
```

ANALYSIS RESULTS USING GROUPED DATA

The results that are generated by using the grouped data with the events/trials response DIST=BINOMIAL, are shown in the tables that follow below.

The Fit Statistics table lists some useful statistics that are based on the maximized value of the log likelihood. These fit statistics are for the marginal model.

Fit Statistics	
-2 Log Likelihood	74.04
AIC (smaller is better)	80.04
AICC (smaller is better)	82.04
BIC (smaller is better)	80.28
CAIC (smaller is better)	83.28
HQIC (smaller is better)	78.44

The Fit Statistics for Conditional Distribution table, shown below, contains the fit statistics for the conditional model given random effects (without integrating the random effects as the procedure does for the marginal model).

Fit Statistics for Conditional Distribution	
-2 log L(x | r. effects)	49.29
Pearson Chi-Square	8.37
Pearson Chi-Square / DF	0.52

The fit statistics for the marginal model is useful for evaluating the model fit for your entire model, whereas the fit statistics for conditional distribution is useful for evaluating the model fit given random effects (the fixed-effect part of the model).

The Covariance Parameter Estimates table displays estimates and asymptotic estimated standard errors for all covariance parameters. The variance of the random clinic intercepts on the logit scale is estimated as σ_c^2 =1.9591.

Covariance Parameter Estimates			
Cov Parm	Subject	Estimate	Standard Error
Intercept	clinic	1.9591	1.1903

The Solutions for Fixed Effects table indicates marginal significance of the two fixed-effects parameters. The positive value of the estimate of treatment parameter indicates that the treatment significantly increases the chance of a favorable cure.

Solutions for Fixed Effects						
Effect	treatment	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept		-1.1974	0.5561	7	-2.15	0.0683
treatment	1	0.7385	0.3004	7	2.46	0.0436
treatment	0	0		.	.	.

The Type III Tests of Fixed Effects table displays significance tests for the fixed effects in the model.

Type III Tests of Fixed Effects				
Effect	Num DF	Den DF	F Value	Pr > F
treatment	1	7	6.04	0.0436

ANALYSIS RESULTS USING INDIVIDUAL DATA

The results that are generated by using the individual data, with a binary response using DIST=BINARY, are shown in the following tables. Notice that the fit and test statistics in these tables differ from those in earlier tables for the group data.

Fit Statistics	
-2 Log Likelihood	302.93
AIC (smaller is better)	308.93
AICC (smaller is better)	309.02
BIC (smaller is better)	309.17
CAIC (smaller is better)	312.17
HQIC (smaller is better)	307.32

Fit Statistics for Conditional Distribution	
-2 log L(cure | r. effects)	278.17
Pearson Chi-Square	248.89
Pearson Chi-Square / DF	0.91

Covariance Parameter Estimates			
Cov Parm	Subject	Estimate	Standard Error
Intercept	clinic	1.9591	1.1903

Solutions for Fixed Effects						
Effect	treatment	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept		-1.1974	0.5561	7	-2.15	0.0683
treatment	1	0.7385	0.3004	264	2.46	0.0146
treatment	0	0	.	.	.	.

Type III Tests of Fixed Effects				
Effect	Num DF	Den DF	F Value	Pr > F
treatment	1	264	6.04	0.0146

EXAMPLE 1: COMMON QUESTIONS

This section contains common questions that might arise based on the Example 1 results for the analysis of the binomial and binary data when you use PROC GLIMMIX.

Why Are the Fit Statistics for Binomial and Binary Data Different?

A constant term is included in the binomial objective function but not in the binary objective function. Therefore, the fit statistics are different. However, a constant term does not affect the optimization. Therefore, you should see similar parameter estimates.

Why are the PROC GLIMMIX Results Different When You Use Events/Trials Syntax versus Individual-Level Data?

For Example 1, the F statistic is based on different denominator degrees of freedom (ddf), which lead to different results. In the events/trials syntax with the DIST=BINOMIAL option, the ddf is associated with the number of clusters. There are eight clinics, so there are seven denominator degrees of freedom. While the individual-level data with the DIST=BINARY option, the number of denominator degrees of freedom is associated with number of observations rather than number of clusters. Note that if the assumption that the conditions for a valid F test are met (for example, the numbers of trials are large enough), the two specifications might lead to the same conclusion (rejection/acceptance) for the Type 3 tests of fixed effects. The t tests that are associated with the parameter estimates also differ because of the different denominator degrees of freedom.

If you specify the CHISQ and DDFM=NONE options in the MODEL statement, you replicate similar test statistics and p-values. Using the CHISQ and DDFM=NONE option is good, especially when you use the events/trials syntax.

```
proc glimmix data=infection method=quad;
    class clinic treatment(ref='0');
    model x/n = treatment/s dist=binomial link=logit ddfm=none chisq;
    random intercept/subject=clinic;
run;

proc glimmix data=infection2 method=quad;
    class clinic treatment(ref='0');
    model cure(event='1')=treatment/s dist=binary link=logit ddfm=none
            chisq;
    random intercept/subject=clinic;
run;
```

The following tables are displayed for both the events/trial data and individual data when you use the DDFM=NONE and CHISQ options.

Solutions for Fixed Effects						
Effect	treatment	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept		-1.1974	0.5561	Infty	-2.15	0.0313
treatment	1	0.7385	0.3004	Infty	2.46	0.0140
treatment	0	0	.	.	.	.

Type III Tests of Fixed Effects						
Effect	Num DF	Den DF	Chi-Square	F Value	Pr > ChiSq	Pr > F
treatment	1	Infty	6.04	6.04	0.0140	0.0140

When Should You Use the DIST=BINOMIAL Option versus the DIST=BINARY Option for a Logistic Regression Model in PROC GLIMMIX?

For the DIST=BINOMIAL option, the dependent variable is typically events/trials syntax. PROC GLIMMIX models the higher response, 1, as the event for DIST=BINOMIAL, as shown in this example:

```
model x/n=treatment/ dist=binomial;
```

However, if you specify DIST=BINARY, the dependent variable is typically 0/1 or character response value Yes/No. The lower response, 0, is modeled, by default, as the event. If you want 1 to be modeled as the event with the DIST=BINARY option, then you specify the EVENT= option, as shown in the following example:

```
model cure(event='1') = treatment /dist=binary;
```

Does Bias Exist When You Estimate a Logistic Regression Model with the DIST=BINARY Option in PROC GLIMMIX?

Lin and Breslow (1996) and Breslow and Lin (1995) discuss the bias in the parameter estimates in binary models that have many non-events and a small number of observations per cluster.

The METHOD=QUAD and METHOD=LAPLACE options in the PROC GLIMMIX statement provide maximum likelihood estimates for the parameters. These methods are similar to the ones that are used in the NLMIXED procedure, and they and produce less biased results for models with the binary response variable. **If you have a binomial response and random effects, it is best to use maximum likelihood methods such as METHOD=QUAD or METHOD=LAPLACE, if possible,** because pseudo-likelihood can produce biased variance estimates for the variance components. Some models cannot be fit with maximum likelihood estimation, in particular, models with correlated errors. In these cases, it is a good idea to specify EMPIRICAL=MBN in the PROC GLIMMIX statement to obtain empirical sandwich estimates for the standard errors of the fixed effects, with small-sample bias correction. The empirical sandwich estimates produce standard errors (and therefore *p*-values and confidence limits) that are robust to the misspecification of the covariance structure. Sometimes, maximum likelihood estimation is very slow, and pseudo-likelihood estimation might give you just what you need in less time.

How Do You Obtain the Predicted Probabilities in PROC GLIMMIX?

Although it is not included in the code for the previous example, the OUTPUT OUT= data set contains the predicted probabilities. To obtain the predicted probabilities, you must apply the inverse link function,

which you can do by using the ILINK option. The predicted probabilities are represented in the keyword PREDICTED with the ILINK option in the OUT= data set, as shown in the following statement:

```
output out=glmxout predicted(blup ilink)=predprob;
```

In the GLMXOUT data set, the variable PREDPROB contains predicted probabilities. If you remove the (ILINK) option, then the PREDPROB variable will contain the linear predictor, XBETA, which is the predicted log odds.

How Do You Obtain ROC Analysis for a Binary Response Model in PROC GLIMMIX?

You can obtain a ROC analysis from PROC GLIMMIX by saving the predicted probabilities from the fitted model and using them in PROC LOGISTIC in the MODEL or ROC statements. Then, PROC LOGISTIC can provide a graph of the ROC curve and the area under the curve (AUC). You can also use PROC LOGISTIC to perform tests that compare the ROC curves from competing models.

As it stands in the example below, PROC GLIMMIX uses the random intercepts from the model when it predicts the probability. This process will almost certainly increase the AUC value relative to an ROC curve based on the fixed effects alone, even when you use the same estimated coefficients. The AUC that you obtain when you use the PROC GLIMMIX estimates but with the NOBLUP option might reflect some clinical practice better.

The following PROC GLIMMIX statements fit a logistic model with a random effect for clinics. Predicted log odds (XBETA) and predicted probabilities (PREDPROB) and predicted probabilities based on fixed effects alone (FIX_PREDPROB) are computed by the OUTPUT statement and saved in data set GMXOUT.

```
proc glimmix data=infection method=quad;
   class clinic treatment(ref='0');
   model x/n=treatment/s dist=binomial link=logit ddfm=none chisq;
   random intercept/subject=clinic;
   output out=gmxout pred=xbeta pred(ilink)=predprob pred(ilink
                                      noblup)=fix_predprob;
run;
```

You can perform the ROC analysis by using either the predicted log odds or the predicted probabilities as the single predictor in the MODEL statement. The PLOTS(ONLY)=ROC option produces a graph of the ROC curve. The area under the ROC curve is displayed at the top of the graph.

The following statements generate the ROC curve.

```
ods graphics on;
proc logistic data=gmxout plots(only)=roc;
   model x/n = predprob;
   ods select roccurve;
run;

ods graphics on;
proc logistic data=gmxout plots(only)=roc;
   model x/n=fix_predprob;
   ods select roccurve;
run;
```

The ROC analysis on the left shows that the AUC is 0.8081 for predicted probabilities using the BLUPS option. As expected, the ROC analysis on the right shows that the AUC is 0.5503 for the predicted probabilities using NOBLUPS option.
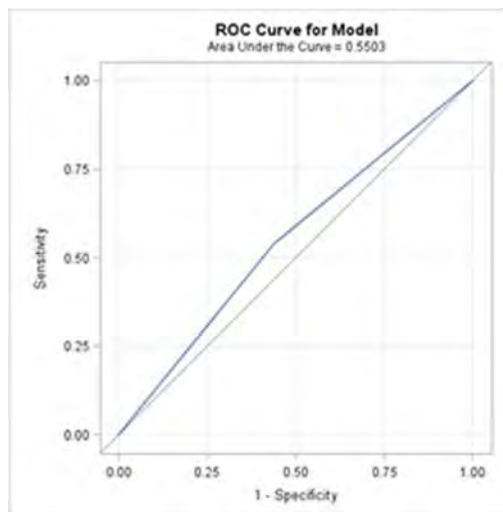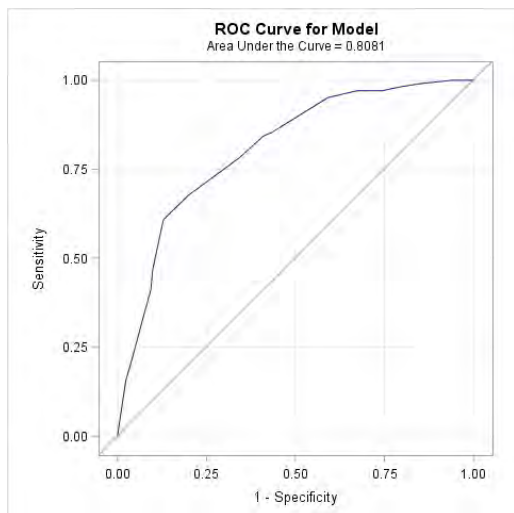
EXAMPLE 2: USING PROC GLIMMIX WITH A BINOMIAL CORRELATED-ERROR MODEL

The next example illustrates how you estimate a marginal, generalized estimating equations (GEE) type of model. A GEE type of model for clustered data is a model for correlated data that you specify 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, you use one of the empirical sandwich estimators to make inferences robust against the choice of the working covariance structure.

As discussed earlier in this paper, PROC GLIMMIX models with only G-side effects are also known as *conditional (or subject-specific) models*, while models with R-side effects are also known as *marginal (or population-averaged) models*. 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 several empirical covariance estimators.

The data for this example is from a clinical trial (Stokes, Davis, and Koch 2012) that was conducted to compare two treatments for a respiratory illness. Patients in each of two centers were randomly assigned to two groups: one group received the active treatment and one group received a placebo.

During treatment, respiratory status was determined for each of four visits and is represented by the variable OUTCOME (coded here as 0=poor, 1=good). The variables CENTER, TREATMENT, SEX, and BASELINE (baseline respiratory status) are classification variables that have two levels. The variable AGE (age at time of entry into the study) is a continuous variable. The variable ID is the patient identification number.

The following statements fit the model:

```
    /* Marginal GEE type of model */
 proc glimmix data=resp empirical;
    class id center sex treatment visit;
    model outcome (event='1')=sex treatment visit age baseline
                          / dist=binary link=logit;
    random _residual_ / subject=id(center) type=cs;
 run;
```

The MODEL statement specifies the model for the fixed effects. The RANDOM statement specifies a compound symmetry structure for the correlations among the (linearized pseudo-) measurements for each patient.

Below are the fit statistics, covariance estimates, and Type III tests from the fitted model. Because a pseudo-likelihood estimation method is used (METHOD=RSPL), no likelihood-based fit statistics are produced in the Fit Statistics table. However, you can use the COVTEST statement to compare appropriate covariance structures.

Fit Statistics	
-2 Res Log Pseudo-Likelihood	1955.83
Generalized Chi-Square	273.30
Gener. Chi-Square / DF	0.63

The Covariance Parameter Estimates table displays the estimates of the covariance parameters. The common covariance is estimated to be 0.3781, as listed in the **CS** row, and the residual variance is estimated to be 0.6268, as listed in the **Residual** row.

Covariance Parameter Estimates			
Cov Parm	Subject	Estimate	Standard Error
CS	ID(Center)	0.3781	0.07448
Residual		0.6268	0.04880

The Type III Tests of Fixed Effects table indicates that the TREATMENT and BASELINE effects are highly significant.

Type III Tests of Fixed Effects				
Effect	Num DF	Den DF	F Value	Pr > F
Sex	1	106	0.43	0.5116
Treatment	1	106	12.99	0.0005
Visit	3	330	1.20	0.3080
Age	1	106	1.01	0.3178
Baseline	1	106	36.56	<.0001

EXAMPLE 2: COMMON QUESTIONS

This section contains common questions that might arise based on the estimation of marginal models in Example 2.

When Is It Best to Use the GLIMMIX Procedure versus the GENMOD Procedure?

Both PROC GENMOD and PROC GLIMMIX are still the procedures to use if you have a generalized linear model. You should use PROC GLIMMIX for generalized linear mixed models, when you have random effects in your model. Both procedures can be used for a generalized linear model with repeated-measures data. There is no recommendation as to the best procedure to use in that situation.

What Is the Difference between Using PROC GENMOD or PROC GLIMMIX to Estimate a Marginal GEE Model?

The GEE implementation in PROC GENMOD is a marginal method that does not incorporate random effects. The GEE estimation in PROC GENMOD relies on R-side covariances only, and the unknown parameters in the **R** matrix are estimated by method of moments.

PROC GLIMMIX fits marginal models by using R-side random effects and pseudo-likelihood estimation. The TYPE= option in the RANDOM statement also provides more flexibility in modeling the correlation structures. PROC GLIMMIX allows G-side random effects and R-side covariances.

When Do You Use a Marginal Model versus a Conditional Model Approach?

The choice of the marginal (population-averaged) model or conditional (subject-specific) model often depends on the goal of your analysis: whether you are interested in population-averaged effects or subject-specific effects.

The GEE model is a marginal, or population-averaged model. If you are interested in making predictions about individuals, then you would use GLIMMIX to the fit the conditional model using G-side random effects and obtain the subject specific estimates. For example,

```
    /* conditional model subject specific estimates */
 proc glimmix data=resp method=quad;
    class id center sex treatment visit;
    model outcome (event='1')=sex treatment visit age baseline
                            /s dist=binary link=logit;
    random intercept /s subject=id(center);
 run;
```

Is It Possible to Reproduce a PROC GENMOD Analysis Using PROC GLIMMIX?

The two procedures are not going to match exactly because the estimation methods are different. PROC GENMOD uses maximum likelihood or method of moments, whereas PROC GLIMMIX uses pseudo-likelihood or maximum likelihood estimation methods.

Using PROC GLIMMIX with these options might help you to obtain results that are similar to those from PROC GENMOD, as follows:

- You can use the EMPIRICAL option in the PROC GLIMMIX statement to obtain results that are similar to estimating a GEE model in PROC GENMOD.

- You can model the overdispersion parameter that you can obtain with PROC GENMOD by adding a RANDOM _RESIDUAL_ statement to PROC GLIMMIX code. When the GLIMMIX and GENMOD procedures fit a generalized linear model where the distribution contains a scale parameter (for example, 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 the scale parameter differs.

- Verify that both procedures are using the GLM parameterization for the classification variables in the CLASS statement.

EXAMPLE 3: USING PROC GLIMMIX IN A MULTINOMIAL MODEL WITH RANDOM EFFECTS

The previous examples are based on a dichotomous response variable. The next example illustrates the analysis of multinomial data where the response variable has more than two categories on the ordinal scale. This example fits the cumulative logit proportional-odds model to the data, assesses the treatment effects, and provides an interpretation of odds ratio.

The data comes from a study to compare active treatment with a control treatment for patients having shoulder pain after rotator-cuff surgery. The two treatments were randomly assigned to patients. The

patients were asked to rate their pain in the morning and afternoon for three days after the surgery. The response-variable pain score is an ordinal measure with five categories where 1=low, 5 =severe.

The following statements fit the model:

```
proc glimmix data=Shoulder_pain empirical=mbn method=quad;
   class subject_id trt gender;
   model y=trt|gender age time/dist=mult link=clogit solution or(label);
   random int / subject=subject_id;
   estimate 'Active Vs Placebo' trt 1 -1 / or;
   store gmxres;
run;
```

Selected results are shown below. First, in the Covariance Parameter Estimates table, you can see that the estimate of variance of the random intercepts σ^2 is 4.7629.

Covariance Parameter Estimates			
Cov Parm	Subject	Estimate	Standard Error
Intercept	subject_id	4.7629	2.0132

Notice there are four intercept terms. Intercept 1 is -5.4149, and it defines the boundary between pain categories 1 and 2. Intercept 2 is -3.8903. This intercept defines the boundary between pain categories 2 and 3, and so on. The intercept terms correspond to the four cumulative logits that are defined on the pain categories in the order shown. The TRT effect estimates show how far up or down the boundaries move under the different treatment groups. In this example, the Active treatment group is positive relative to the Placebo treatment. This means that the Active treatment has a higher probability of **1** (low) pain score and a lower probability of **5** (severe) pain score than the Placebo treatment. The EMPIRICAL=MBN option in the PROC GLIMMIX statement makes small-sample variance corrections to the standard errors.

Solutions for Fixed Effects								
Effect	y	Trt	gender	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept	1			-5.4149	1.8176	36	-2.98	0.0052
Intercept	2			-3.8903	1.8488	36	-2.10	0.0424
Intercept	3			-2.2656	1.8079	36	-1.25	0.2182
Intercept	4			-0.1664	1.8358	36	-0.09	0.9283
trt		Active		2.5634	1.4080	201	1.82	0.0702
trt		Placebo		0	.	.	.	.
gender			Female	0.002896	1.4163	201	0.00	0.9984
gender			Male	0	.	.	.	.
trt*gender		Active	Female	0.5278	1.8341	201	0.29	0.7738
trt*gender		Active	Male	0	.	.	.	.
trt*gender		Placebo	Female	0	.	.	.	.
trt*gender		Placebo	Male	0	.	.	.	.
age				0.04950	0.02727	201	1.82	0.0709
time				0.3814	0.1391	201	2.74	0.0067

PROC GLIMMIX models the probability of the lower end of the pain scale. Therefore, the odds ratio for TIME being greater than one indicates a lessening of pain with TIME. These results appear to support an expected reduction in pain three days after the surgery.

Odds Ratio Estimates				
Comparison	Estimate	DF	95% Confidence Limits	
trt Active vs Placebo	16.901	201	2.829	100.963
gender Female vs Male	1.306	201	0.196	8.691
unit change of age from mean	1.051	201	0.996	1.109
unit change of time from mean	1.464	201	1.113	1.927

Effects of continuous variables are assessed as one unit offsets from the mean. The AT suboption modifies the reference value and the UNIT suboption modifies the offsets.

The TYPE III tests for the TRT effect tests the significance of the log of the odds ratio that is obtained from comparing the Active vs. Placebo treatment group. The treatment and time effects are highly significant. However, the AGE effect is only marginally significant.

Type III Tests of Fixed Effects				
Effect	Num DF	Den DF	F Value	Pr > F
Trt	1	201	9.73	0.0021
gender	1	201	0.08	0.7816
trt*gender	1	201	0.08	0.7738
Age	1	201	3.30	0.0709
Time	1	201	7.51	0.0067

The ESTIMATE statement reports the log odds ratio (2.8273) and the odds ratio estimate (16.9006) indicates the relative differences for the Active Vs Placebo treatment group.

Estimates						
Label	Estimate	Standard Error	DF	t Value	Pr > |t|	Exponentiated Estimate
Active Vs Placebo	2.8273	0.9065	201	3.12	0.0021	16.9006

The odds ratio of 16.9006 indicates that the odds of the Active treatment group being in lower pain categories is approximately 17 times the odds of the Placebo group being in lower pain categories. Simply put, the odds ratio of 16.9006 indicates approximately a 17-fold reduction in pain. Note this is the same odds ratio as reported in the Odds Ratio table that is associated with the Solution of Fixed Effects table.

How Do You Obtain Least Squares Means (LS-MEANS) and Respective Differences for a Multinomial Response Model When You Use PROC GLIMMIX?

For the multinomial response model (DIST=MULTINOMIAL), the least-square means (LS-MEANS) is not available for PROC GLIMMIX. If you can determine that what you want to estimate is some linear combination of the model parameters, then you can use the ESTIMATE statement to estimate that

quantity. If you want to estimate the probability of one of your multinomial events, you can do that by using the ESTIMATE statement with the ILINK option.

Another approach is to create the item store using the STORE statement in PROC GLIMMIX and then use the PLM procedure, as shown below, to compute the LS-MEANS.

```
proc plm restore=gmxres;
   lsmeans trt/ilink diff exp;
   ods select diffs;
run;
```

As you can see in the PROC PLM output below, the log odds ratio and odds ratio estimates match the results of the ESTIMATE statement in PROC GLIMMIX.

trt	_trt	Estimate	Standard Error	DF	t Value	Pr > |t|	Exponentiated
Active	Placebo	2.8273	0.9065	201	3.12	0.0021	16.9006

Differences of trt Least Squares Means

GENERAL QUESTIONS RELATED TO BINOMIAL MODELS IN PROC GLIMMIX

The following commonly asked questions can apply to any of the three examples described earlier as well as any of the other distributions that are available in PROC GLIMMIX.

How Do You Assess Overdispersion for a Binomial or Binary Model?

PROC GLIMMIX does not generate deviance or scaled deviance statistics. The Fit Statistics table lists information about the fitted model. For generalized linear models, PROC GLIMMIX reports the Pearson chi-square statistic. For generalized linear mixed models (GLMMs), the procedure typically reports a generalized chi-square statistic when you use pseudo-likelihood estimation. A GLMM that uses the METHOD=LAPLACE or METHOD=QUAD options also reports the Pearson chi-square statistic.

In many GLMMs, the observed conditional variance can differ from what is expected under the baseline distributions in much the same way in which overdispersion is possible in the generalized linear model.

For example, in the following code, the conditional variance is determined by the binomial distribution:

```
proc glimmix data=weed;
   class Soil Loc;
   model Y/N = soil / dist=binomial;
   random intercept /subject=loc;
run;
```

In other words, the dispersion scale-parameter phi is assumed to be 1 because the binomial distribution does not have a scale parameter. The Generalized Chi-Square statistic is divided by its degrees-of-freedom estimates for the residual variation based on the final pseudo data.

Fit Statistics	
-2 Res Log Pseudo-Likelihood	194.10
Generalized Chi-Square	160.20
Gener. Chi-Square / DF	5.01

Often, people incorrectly assume that if the Generalized Chi-Square statistic divided by its degrees-of-freedom value is larger than 1, then this is indicative of misspecified conditional distribution and requires the addition of an extra scale parameter. However, under pseudo-likelihood estimation, the Generalized Chi-Square statistic divided by its degrees of freedom is not well defined as a measure of fit and it cannot

assess overdispersion. Given that, you would not want to use the Generalized Chi-Square statistic divided by its degrees of freedom as a measure of overdispersion for a GLMM. To diagnose overdispersion using Pearson Chi-Square/DF requires a true likelihood for computation that you can obtain by using METHOD=LAPLACE or METHOD=QUAD.

Fitting the model above using the METHOD=QUAD option in the PROC GLIMMIX statement, the Fit Statistics for Conditional Distribution indicate the following results:

Fit Statistics for Conditional Distribution	
-2 log L(Y | r. effects)	299.37
Pearson Chi-Square	145.12
Pearson Chi-Square / DF	4.27

The likelihood is computed from the true likelihood. The ratio of the Pearson chi-square statistic and its degrees of freedom exceeds **1**, which indicates that the variability in these data has not been properly modeled and that there is residual overdispersion. Keep in mind that you can also examine the Pearson type residuals in PROC GLIMMIX to diagnose whether an overdispersion adjustment is needed. Furthermore, it is important to remember that causes for overdispersion can be a result of using the wrong probability distribution, missing an important predictor, or source of variation in the model.

How Do You Obtain an Odds Ratio Estimate for a Change Other Than One Unit When You Use PROC GLIMMIX?

To obtain an odds ratio estimate for a change other than one unit, use the UNIT suboption with the ODDSRATIO option in the MODEL statement. For example, if your independent variable is X1, then the UNIT suboption is added to the MODEL statement, as shown below:

```
model y=trt x1 x2 / s dist=binary oddsratio(unit x1=-10 label);
```

This statement computes the odds ratio for a ten-unit decrease in X1, and the LABEL option provides a condensed Odds Ratio table.

How Do You Use PROC GLIMMIX to Calculate the Odds Ratio and Relative Risk for a Binary Model?

The following PROC GLIMMIX code fits an odds ratio and a relative-risk binary model, respectively.

```
     /* Odds ratio */
proc glimmix data=test;
    class trt;
    model y=trt x1 /dist=binary link=logit s;
run;

    /* Relative risk */
proc glimmix data=test;
    class trt;
    model y=trt x1 /dist=binary link=log s;
run;
```

How Do You Compute an R-square Statistic in PROC GLIMMIX?

PROC GLIMMIX does not have an option for computing an R-square statistic. You can use the %RSQUAREV macro, based on Zhang (2016), to compute the coefficient of determination for generalized linear models. You can also use the %GLIMMIX_GOF macro, provided by Vonesh and Chinchilli (1997) and Vonesh (1996), to obtain additional model fit statistics when you use PROC GLIMMIX.

How Do You Compute an Intra-Class Correlation Coefficient (ICC) for a Non-Normal Response Model in PROC GLIMMIX (Binary, Multinomial, and so on)?

SAS is not aware of a generally accepted method for calculating an ICC in a logistic model, mainly because there is no concept of a residual variance in a logistic regression model. You can form your own ratios of the variance components in your model, but SAS does not endorse such an ICC calculation for a non-normal response model.

TROUBLESHOOTING

This section discusses troubleshooting techniques in PROC GLIMMIX regarding the following:

- recommendations for handling various notes, warnings, and error messages
- strategies for addressing long run times or insufficient memory
- strategies to try when a model fails to converge
- recommendations for identifying and circumventing a quasi-separation issue in a binomial logistic model

Recommendations for Handling Notes, Warnings, and Errors in PROC GLIMMIX

Recommendations and circumventions for various notes, warnings, and error messages are often data and model dependent and should be evaluated on a case-by-case basis. The next section discusses some common strategies that might circumvent some of the messages that are reported in the SAS log when you use PROC GLIMMIX. If you have run generalized linear mixed models much, you have undoubtedly encountered some version of these messages. Perhaps you have wondered what these messages mean and what you need to do about them. This section provides some insight into these questions for various messages.

Strategies for Addressing Notes That Occur When You Use PROC GLIMMIX

The following PROC GLIMMIX code demonstrates the analysis for a binomial random-coefficients model.

```
proc glimmix data=cohort method=laplace;
    class inst;
    model mort1yr=x1 x2 / dist=binomial solution;
    random intercept x1 x2 / subject=inst type=un;
run;
```

When you submit this code, the following notes are generated in the SAS log:

```
NOTE: Convergence criterion (FCONV=2.220446E-16) satisfied.
NOTE: At least one element of the gradient is greater than 1e-3.
NOTE: Estimated G matrix is not positive definite.
```

The results of the PROC GLIMMIX analysis are shown in the following tables. The results include a condensed table for the Iteration History, Covariance Parameter Estimates, and Solution for Fixed Effects:

Iteration	Restarts	Evaluations	Objective	Change	MaxGradient
0	0	4	163.05149995	.	1247.873
1	0	9	163.04608362	0.00541633	33.6281
2	0	3	163.00850546	0.03757817	21.09995
3	0	3	162.77615314	0.23235231	39.47698
.....more data lines					
15	0	3	162.32747773	0.01020608	84.11765
16	0	3	162.32684028	0.00063745	2.739381
17	0	3	162.32682791	0.00001237	3.038176
18	0	3	162.32682775	0.00000016	0.127753

Convergence criterion (GCONV=1E-8) satisfied.

Estimated G matrix is not positive definite.

Covariance Parameter Estimates			
Cov Parm	Subject	Estimate	Standard Error
UN(1,1)	inst	0.05156	.
UN(2,1)	inst	0.02003	.
UN(2,2)	inst	0.02803	.
UN(3,1)	inst	0.000903	.
UN(3,2)	inst	0.000146	.
UN(3,3)	inst	0.000018	.

Solutions for Fixed Effects					
Effect	Estimate	Standard Error	DF	t Value	Pr > |t|
Intercept	0.002926	0.03797	89	0.08	0.9388
x1	1.0098	0.03243	89	31.14	<.0001
x2	0.01548	0.002329	89	0.08	0.9351

Even though the model converges, the missing values for the estimated standard errors of the covariance parameters are an indication of some sort of modeling difficulty or misspecification. Furthermore, the SAS log displays the following message:

```
NOTE: At least one element of projected gradient is greater than 1e-3.
```

The gradient measures a rate of change in the response versus the change in the associated parameter estimate. Theoretically, the gradient should be 0 at the global maximum estimate. In reality, the gradient

17

should be close to `0`. When the previous note is displayed, you should examine the gradient. As is the case with some other model-fitting difficulties, large gradients might be an indication of a misspecified model or a problematic model fitting.

If misspecification is not the case, the options to control the nonlinear optimization process found in the NLOPTIONS statement might provide a means to avoid this model-fitting problem. In particular, consider a different specification for the TECH= option. You might want to consider using TECH=NRRIDG or TECH=NEWRAP because these optimization techniques tend to work well with categorical data.

When the SAS log displays the following message, you often notice that some covariance parameter estimates are either 0 or have no estimate, or there are no standard errors at all:

```
NOTE: Estimated G matrix is not positive definite.
```

It is important that you do not ignore this message. When the messages above occur in the SAS log, you should not compare results across operating systems or different SAS releases. One item to check is the scaling of your predictor variables. If they are on vastly different scales, the model might have trouble calculating variances. Then you might need to simply change the scaling of a predictor variable.

If the best estimate for a variance is 0, it means that there really is no variation in the data for that effect. For example, in a random coefficients model, perhaps the slopes do not really differ across individuals, and the random intercept captures all the variation.

When the best estimate for a variance is 0, you need to respecify the random parts of the model. This might mean that you need to remove a random effect. Sometimes, even when a random effect ought to be included because of the design, there just is no variation in the data. Or, you might need to use a simpler covariance structure with fewer unique parameters. Another alternative, if the design and your hypothesis allow it, is to consider a marginal model that does not contain a random effect.

Strategies for Addressing Warnings That Occur in PROC GLIMMIX

Occasionally when you use PROC GLIMMIX, you might encounter one of the following warnings:

```
WARNING: Obtaining minimum variance quadratic unbiased estimates as
starting values for the covariance parameters failed.
```

```
WARNING: The initial estimates did not yield a valid objective function.
```

It is important to understand that these warning messages might be data and model dependent regarding when they might occur in a program. Therefore, each warning should be evaluated on case-by-case basis.

Problems Finding Initial Values

For example, consider the following model that is estimated with PROC GLIMMIX:

```
data example;
   input center id treatment sex age visit outcome baseline;
   datalines @@;
1 1 P M 46 1 0 0 1 2 P M 28 3 0 0 1 3 A M 23 4 0 1
1 1 P M 46 2 0 0 1 2 P M 28 4 0 0 1 3 A M 23 4 0 1
1 1 P M 46 3 0 0 1 3 A M 23 1 1 1 1 4 P M 44 1 1 1
1 1 P M 46 4 0 0 1 3 A M 23 2 1 1 1 4 P M 44 2 1 1
1 2 P M 28 1 0 0 1 3 A M 23 3 1 1 1 4 P M 44 3 1 1
1 2 P M 28 2 0 0 1 3 A M 23 4 1 1 1 4 P M 44 4 0 1
...more data lines...
;
```

(code continued)

```
proc glimmix data=example;
   class id center sex treatment visit;
   model outcome (event='1')=sex treatment visit age baseline
                               / dist=binary link=logit;
   random visit/ subject=id(center) type=ar(1) residual;
   random intercept/subject=id(center);
run;
```

When you estimate this model with PROC GLIMMIX, the following warning is generated in the SAS log:

```
WARNING: Obtaining minimum variance quadratic unbiased estimates as
starting values for the covariance parameters failed.
```

When you estimate an R-side random effect, PROC GLIMMIX can only have one observation per level of the repeated effect for each subject. For example, if the data for a specific subject has two observations for the same time point, PROC GLIMMIX issues the warning message shown above and stops processing. You need to check your data and remove any duplicate time points to enable the procedure to run.

In the previous code example, the data set has three observations that contain a value of 4 for the VISIT variable, a value of 3 for the ID variable, and a value of 1 for the CENTER variable. As such, these observations violate the data requirements for PROC GLIMMIX with an R-side random effect.

In other types of analysis or circumstances where the warning shown above occurs, you might specify your own set of starting values for the covariance parameters in a PARMS statement.

If these strategies do not work, it might be that your data does not support your model, and you might have to consider a different approach to resolve this issue.

Problems Evaluating Code for Objective Function

PROC GLIMMIX fits a generalized linear model to obtain the initial fixed-effect estimates. However, at times, the initial estimates might not yield a valid objective function for the generalized linear mixed model. Consider this PROC GLIMMIX example that simulates a conditional model with a binary response:

```
data one;
   do pt=1 to 100;
      do visit=0, 4, 8,12;
         trt=round(ranuni(0)*4);
         ranint=rannor(1234);
         do time=0 to 1;
            x1=rannor(1234);
            linp=-1+ranint;
            pi=1/(1+exp(-linp));
            y=ranbin(0,1,pi);
            base=round(ranuni(0)*12);
            output;
         end;
      end;
   end;
run;

proc glimmix data=one method=laplace;
   class pt trt visit;
   model y(event='1')=trt visit trt*visit base/dist=binary link=identity
   s;
   random intercept /subject = pt;
run;
```

When you submit this code, the following warning is generated in the SAS log:

```
WARNING: The initial estimates did not yield a valid objective function.
```

The first strategy you should try is to verify that your code specification is correct. In this example, if you determine that you should use LINK=LOGIT, make that correction, and the model will converge. However, if you truly want to use LINK=IDENTITY in this example, you can use the following strategies:

- Specify a different method of estimation.
- Specify a different optimization technique.
- Specify starting values for covariance parameters.

You can use these strategies to circumvent various notes, warnings, and error messages that might occur when you try to estimate the previous example by using the DIST=BINARY and LINK=IDENTITY options, as show in this example:

```
proc glimmix data=one method=quad(qpoints=7);
    class pt trt visit;
    model y(event='1')=trt visit trt*visit base / dist=binary
                                        link=identity s;
    random intercept /subject=pt;
    nloptions tech=nrridg;
    parms (0.003);
run;
```

Using the code above, the SAS log now reports the following messages:

```
NOTE: The GLIMMIX procedure is modeling the probability that y='1'.
NOTE: Convergence criterion (GCONV=1E-8) satisfied.
```

You can also use the following strategies when the warning message that is shown above is generated in the SAS log.

- Increase the value for the INITITER= option in the PROC GLIMMIX statement.
- Try the INITGLM option in the PROC GLIMMIX statement.
- Try the NLOPTIONS TECH=NRRIDG; statement to see if that makes any difference.
- Check your data and the scales of variables. If the variables differ in magnitude, you need to rescale them.
- Specify your own starting values for the covariance parameters using the PARMS statement.
- Verify that your PROC GLIMMIX code is specified correctly.

Strategies for Addressing Long Run Times or Insufficient Memory in PROC GLIMMIX

In terms of memory and run time, one of the most resource intensive pieces of estimating a generalized linear mixed model is evaluating the likelihood function. There are six different methods for taking on this calculation in PROC GLIMMIX that vary from pseudo-likelihood evaluation to estimating the exact likelihood. When you choose from these methods, you need to take into consideration the effects of estimation bias with certain response types. You also need to weigh the cost of the likelihood evaluation. Some of these methods require more memory and run time than others. Finally, depending on which type of model you are working with, you might be limited to a subset of these methods.

A very popular method is to use adaptive quadrature by specifying the METHOD=QUAD option in PROC GLIMMIX. This approach can yield very exact evaluations of the log-likelihood through integral approximation, but that exactness comes at a price. Quadrature is an expensive method to use, both in terms of run time and memory. For some multilevel models, quadrature might not be memory-feasible.

Beginning with SAS/STAT® 14.1 software in SAS 9.4 TS1M3, the FASTQUAD suboption for the METHOD=QUAD option can often alleviate long run times and memory constraints. FASTQUAD uses the multilevel adaptive-quadrature algorithm proposed by Pinheiro and Chao (2006). For a multilevel model, this algorithm reduces the number of random effects over which the integration of the marginal log-

likelihood computation is carried out. This reduction in the dimension of the integral can lead to a substantial reduction in the number of conditional log-likelihood evaluations.

If you fit the following model using METHOD=QUAD, the result is an `insufficient resources` message or `out of memory` message from the procedure:

```
proc glimmix data=test method=quad;
   class hospital doctor;
   model y=x1 / dist=binary link=logit;
   random int / subject=hospital;
   random int / subject=doctor(hospital);
run;

NOTE: The GLIMMIX procedure is modeling the probability that y='0'.
ERROR: Insufficient resources to perform adaptive quadrature with 3
quadrature points. METHOD=LAPLACE, corresponding to a single point,
might provide a computationally less intensive possibility.
```

Prior to SAS/STAT 14.1 software, METHOD=LAPLACE might provide a satisfactory alternative. However, with release 14.1, you can try the FASTQUAD suboption, as follows:

```
proc glimmix data=test method=quad(fastquad qpoints=3);
   class hospital doctor;
   model y=x1 / dist=binary link=logit;
   random int / subject=hospital;
   random int / subject=doctor(hospital);
run;
```

This second attempt at PROC GLIMMIX dramatically reduces the amount of memory that is needed to estimate the model, and the procedure completes with no errors. One important note is that to take advantage of the FASTQUAD method, you need to specify one RANDOM statement for each level. This action is shown in the PROC GLIMMIX example above.

The significant reduction in the exponent (for the number of conditional log-likelihood evaluations) allows the use of more quadrature points, increasing the accuracy of the evaluation of the integral and increasing the accuracy of the log-likelihood. For additional strategies to improve performance in PROC GLIMMIX, see Kiernan et al. (2012).

Strategies to Try When the Model Fails to Converge in PROC GLIMMIX

Convergence problems are quite common with models that are fit by iterative optimization methods for maximum likelihood, and such problems can happen in many possible ways that depend on the data and model. Any change to the response or predictors presents a completely new optimization problem that can produce very different results from a seemingly similar scenario. Typically, you cannot determine the cause in any individual case by examining the model specification or data. Often, a solution must be found by experimentation. Usually, the most helpful strategy is to simplify the model (that is, reduce the number of model parameters) in some acceptable way such as by removing higher-order effects (for example, interactions), removing predictors, or dropping or merging categories of CLASS variables. In general, the more parameters there are in the model, the more likely convergence problems become.

In categorical response models, sparseness of the data is common and can cause various fitting errors, though this is not the only possible cause. Generally, as model complexity increases and sample size decreases, the data becomes sparser and more likely to result in convergence problems. Starting with a simple model and adding variables as they can be supported is often a good strategy.

In some cases, a relative-risk model might encounter convergence issues. The defined relative-risk model with the binary response and the LINK=LOG option does not ensure that predicted probabilities are mapped to the [0,1] interval, which leads a variance function(vf) vf=p*(1-p) to be negative. Even if PROC GLIMMIX could compute the starting values, there is still no guarantee of a valid variance function in every iteration of the optimization.

Note that using the DIST=BINOMIAL and LINK=LOGIT options does not enable direct estimation of a relative risk. Instead, those options enable you to estimate the odds ratios. But it is well known that when the event probability is low, the odds ratio is a good estimate of the relative risk. Keep in mind that the issue might be that your data does not support the model and you might have to consider an alternative approach.

However, these strategies are helpful in troubleshooting convergence issues:

- Rescale your data.
- Check data for sufficient variability before estimating a model.
- Examine iteration history.
- Simplify your model.
- Consider an alternative TYPE= covariance structure.
- Use the PARMS statement.
- Consider an alternative method of estimation using the METHOD= option in the PROC GLIMMIX statement.
- Use options in the NLOPTIONS statement. For example, try the TECH=NRRIDG or TECH=NEWRAP options. The TECH= option is especially useful for binary and binomial distributions.

For more details about various strategies to obtain convergence, see Kiernan et al. (2012).

Strategies for Identifying and Circumventing a Quasi-Separation Issue in PROC GLIMMIX

Unlike the LOGISTIC procedure, PROC GLIMMIX does not generate a warning message about a possible quasi-separation issue. Nor does PROC GLIMMIX support a FIRTH option. However, you can examine the Solutions for Fixed Effects table for large parameter estimates, standard errors, odds ratios, and confidence limits that can alert you to a potential problem such as quasi-complete separation in PROC GLIMMIX.

The validity of the model fit is questionable because of quasi-completion separation of data points. With binary data, your chances of this type of problem occurring increases in a model that has many factors or sparse data with little or no variation in the response. Three possible strategies to the separation problem of making the model fit less perfectly are as follows:

- reduce the number of variables or effects in the model
- categorize continuous variables
- merge categories of categorical (CLASS) variables

It is often difficult to know exactly which variables cause the separation, but variables that exhibit large parameter estimates or standard errors are likely candidates.

CONCLUSION

PROC GLIMMIX offers a flexible and powerful procedure for fitting generalized linear mixed models. The focus of this paper is to demonstrate modeling conditional and marginal models for categorical response data by using PROC GLIMMIX. The discussion addresses some of the basic questions that arise when you estimate binomial and multinomial models with PROC GLIMMIX. Finally, the paper discusses strategies for handling various notes, warnings, and error message that might occur in PROC GLIMMIX when you estimate binomial and multinomial models. Each analysis must be evaluated on a case-by-case basis. Various strategies are discussed, and you should not be afraid to try several of the strategies!

REFERENCES

- SAS Institute Inc. 2012. *SAS/STAT® 14.2 User's Guide*. Cary, NC: SAS Institute Inc. Available at **support.sas.com/documentation/onlinedoc/stat/142/statug.pdf**.
- Beitler, Paula. J. and J Richard Landis. 1985. "A Mixed-Effects Model for Categorical Data," *Biometrics* 41:991–1000.
- Breslow, Norman E. and Xihong Lin. 1995. "Bias Correction in Generalized Linear Mixed Models with a Single Component of Dispersion", *Biometrika*, 81:81-91.

- Kiernan, Kathleen. Jill Tao, and Phil Gibbs. 2012. "Tips and Strategies for Mixed Modeling with SAS/STAT Procedures". *Proceedings of the SAS Global Forum 2012 Conference.* Cary, NC: SAS Institute Inc. Available at
  `support.sas.com/resources/papers/proceedings12/332-2012.pdf`.
- Lin, Xihong and Norman E. Breslow. 1996. "Bias Correction in Generalized Linear Mixed Models with Multiple Components of Dispersion.", *Journal of American Statistical Association* 19, 435:1007-1016.
- Pinheiro, J. C. and E.C Chao. (2006). "Efficient Laplacian and Adaptive Gaussian Quadrature Algorithms for Multilevel Generalized Linear Mixed Models*." Journal of Computational and Graphical Statistics* 15:58–81.
- SAS Institute, Inc. 2002. SAS Note 22599, "Understanding and correcting complete or quasi-complete separation problems"". Cary, NC: SAS Institute Inc. Available at
  `support.sas.com/kb/22/599.html`.
- SAS Institute, Inc. 2010. SAS Note 40724, "Comparing covariance structures, testing covariance parameters using the COVTEST statement in PROC GLIMMIX.". Cary, NC: SAS Institute Inc. Available at `support.sas.com/kb/40/724.html`.
- SAS Institute, Inc. 2010. SAS Note 41364, "ROC analysis for binary response models fit in the GLIMMIX, NLMIXED, GAM or other procedures." Cary, NC: SAS Institute Inc. Available at `support.sas.com/kb/41/364.html`.
- SAS Institute, Inc. 2017. SAS Sample 60162, "R-square and partial R-square for generalized linear models based on the variance function." Cary, NC: SAS Institute Inc. Avail *(list continued)* `support.sas.com/kb/60/162.html`.
- Stokes, Maura E., Charles S. Davis, and Gary. G. Koch. 2000. *Categorical Data Analysis Using the SAS System.* 2nd ed. Cary, NC: SAS Institute Inc.
- Stroup, Walter W. 2012. *Generalized Linear Mixed Models: Modern Concepts, Methods and Applications.* New York: CRC Press.
- Vonesh, Edward. F. 1996. "A Note on Laplace's Approximation for Nonlinear Mixed-Effects Models." *Biometrika* 83:447–452.
- Vonesh, Edward. F. and Vernon M. Chinchilli. 1997. *Linear and Nonlinear Models for the Analysis of Repeated Measurements.* New York: Marcel Dekker.
- Vonesh, Edward F., Vernon M. Chinchilli, and KW Pu. 1996. "Goodness-of-Fit in Generalized Nonlinear Mixed-Effects Models*." Biometrics* 52:572–587.
- Vonesh, Edward. F. 2012. *Generalized Linear and Nonlinear Models for Correlated Data: Theory and Applications Using SAS®.* Cary, NC: SAS Institute Inc.
- Zhang, Dabao. 2016. "A Coefficient of Determination for Generalized Linear Models." *The American Statistician.* Tentatively accepted.

ACKNOWLEDGMENTS

CONTACT INFORMATION

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

Kathleen Kiernan
SAS Institute Inc.
Email: `support@sas.com`
Web: `support.sas.com`