GENERALIZED LINEAR MIXED MODEL ESTIMATION USING PROC GLIMMIX: RESULTS FROM SIMULATIONS WHEN THE DATA AND MODEL MATCH, AND WHEN THE MODEL IS MISSPECIFIED

Debbie Boykin[1], Mary J. Camp[2], LuAnn Johnson[3], Matthew Kramer[4], David Meek[5], Debra Palmquist[6], Bryan Vinyard[7], and Mark West[8]

[1] Mid-South Area Office, ARS/USDA, Stoneville, MS 38776
[2] Biometrical Consulting Service, Beltsville Area, ARS/USDA, Beltsville, MD 20705
[3] University of North Dakota, Grand Forks, ND 58203
[4] Biometrical Consulting Service, Beltsville Area, ARS/USDA, Beltsville, MD 20705 (email: matt.kramer@ars.usda.gov for reprints)
[5] NLAE, Midwest Area, ARS/USDA, Ames, IA 50011
[6] Biometrical Services, Midwest Area, ARS/USDA, Peoria, IL 61604
[7] Biometrical Consulting Service, Beltsville Area, ARS/USDA, Beltsville, MD 20705
[8] Northern Plains Area Office, ARS/USDA, Fort Collins, CO 80526

Abstract

A simulation study was conducted to determine how well SAS® PROC GLIMMIX (SAS Institute, Cary, NC), statistical software to fit generalized linear mixed models (GLMMs), performed for a simple GLMM, using its default settings, as a naïve user would do. Data were generated from a wide variety of distributions with the same sets of linear predictors, and under several conditions. Then, the data sets were analyzed by using the correct model (the generating model and estimating model were the same) and, subsequently, by misspecifying the estimating model, all using default settings. The data generation model was a randomized complete block design where the model parameters and sample sizes were adjusted to yield 80% power for the *F*-test on treatment means given a 30 block experiment with block-by-treatment interaction and with additional treatment replications within each block. Convergence rates were low for the exponential and Poisson distributions, even when the generating and estimating models matched. The normal and lognormal distributions converged 100% of the time; convergence rates for other distributions varied. As expected, reducing the number of blocks from 30 to five and increasing replications within blocks to keep total N the same reduced power to 40% or less. Except for the exponential distribution, estimates of treatment means and variance parameters were accurate with only slight biases. Misspecifying the estimating model by omitting the block-by-treatment random effect made *F*-tests too liberal. Since omitting that term from the model, effectively ignoring a process involved in giving rise to the data, produces symptoms of over-dispersion, several potential remedies were investigated. For all distributions, the historically recommended variance stabilizing transformation was applied, and then the transformed data were fit using a linear mixed model. For one-parameter members of the exponential family an over-dispersion parameter was included in the estimating model. The negative binomial distribution was also examined as the estimating model distribution. None of these remedial steps corrected the over-dispersion problem created by misspecifying the linear predictor, although using a variance stabilizing transformation did improve convergence rates on most distributions investigated.

1. Introduction

Researchers in agriculture frequently collect data that do not satisfy the linear mixed model assumption that the response variables are normally distributed. The recent addition of procedures in some commonly used statistical software packages (e.g. SAS®, R[1]) expand their analysis choices to a more general class of models, referred to as generalized linear mixed models (GLMMs). GLMMs provide a way to fit responses to predictors that include counts and proportions, which arise from distributions that are not necessarily normal but are included in the exponential distribution family. There is also a sense of parsimony when modeling data based on the distribution one believes one is sampling from, rather than relying on transformations as a vehicle to move data into the familiar normal distribution framework due to software limitations. The GLMM procedures are extensions of software already developed for estimating linear mixed models. However, by allowing for estimation of many GLMMs, the estimation process becomes more complicated and time consuming. The SAS software designers have made three non-Bayesian algorithms available that allow models to be estimated quickly (pseudo-likelihood—the default method, Laplace, and quadrature). Statistics computed from the fitted models for testing and estimating model effects are based on large sample (asymptotic) results and first order Taylor's series approximations. SAS does not give guidelines on the sample size necessary to ensure accurate parameter estimates and unbiased tests. Because this kind of modeling is in its infancy, there do not appear to be general rules one can apply, and sample size requirements will vary greatly by sampling distribution and experimental design. Even if the design has the power to detect a given difference, that does not ensure that the algorithms used in Glimmix will provide accurate parameter estimates. For large data sets, estimating cell variances may be possible, which can be used to help determine the appropriate sampling distribution and determine if over-dispersion is present (see example for a generalized Poisson distribution in SAS Institute, Inc. 2010, p. 2955).

One way to determine how well the estimating algorithms work is to simulate data for a given model with known properties and parameters, and ask how well the software can reproduce them. We can also ask how useful are the statistics produced from the fitted model for testing hypotheses about the model's parameters. Our approach was to simulate data from the same linear mixed model effects, varying the response type for a wide variety of the distributions supported by SAS PROC GLIMMIX. The model was a randomized complete block design that included a block × treatment interaction, with additional replications for each treatment within the blocks. This is a simple design, which made it easier to interpret results. It is also a regularly used design in agriculture. Initially, data were generated from known models with sufficient power (80%) to detect specified treatment effects for a given number of blocks (30 blocks) when the true model was fitted. Using the very same data, both correctly specified models and those

[1] The mention of a trade-name is for informational purposes only; it does not imply an endorsement by the USDA-ARS.

that were incorrectly specified (e.g. omitting a term) were fit. Omission of a term is probably one of the most common errors when developing a statistical model, and may occur when predictor variable data are not collected, restrictions on randomization are not acknowledged, or there is incomplete understanding of the processes that generate the data. By including a true nonzero block × treatment interaction effect, we could evaluate the effect of misspecifying the model by ignoring it, which we believe is typically done in analyses that include blocking effects. Additionally, data were generated from the same distributions, but with the treatment effects equal, to evaluate the Type I error rates.

Residual variance estimation

The mixed models software differs from that used to estimate parameters of linear models in a fundamental way. In addition to being able to specify a factor as being fixed or random, the variance parameters are estimated differently. In the familiar linear model, there is only one variance parameter, and it is estimated by subtraction. From the total variance (as a sum of squares) of the dependent variable, one subtracts the sum of squares that is accounted for by the linear model, and the remaining sum of squares is used to estimate the residual variance. Now, if a term is missing from the model, and it is orthogonal to all other effects, the portion of the variance in the dependent variable that the missing variable explains goes into the residual. If the missing variable is the blocking effect, the residual variance would be increased by the amount that corresponds to the blocking effect's missing variance. Thus, if one is missing an important effect, the linear model's error variance is inflated, which makes *F*-tests on the remaining effects conservative.

Contrast this with what occurs for mixed models. Here, the variance terms are estimated directly, so the variance due to a missing random effect is, at best, only partly accounted for by changes in the variance components that remain. That is, if one summed the variances accounted for by the fixed and random effects that were specified in the model, that total would be less than the variance of the dependent variable. Since the variance for the missing effect is largely ignored (and does not get to contribute to the standard errors of the fixed effects), *F*-tests on the fixed effects become too liberal. Thus, a missing (random) linear predictor has opposite effects, depending on whether one is misspecifying the model as a linear model (ignoring the random effect), or misspecifying it as a linear mixed model, but with a missing linear predictor (the missing random effect).

Things become more complicated when moving into the generalized linear mixed models framework because a 'residual' may or may not make sense, depending on whether one is using a two-parameter versus a one-parameter distribution. For example, for the binomial and Poisson (one-parameter) distributions, the differences between what the model predicts (on the original scale) and the data values should be only sampling error (the samples will differ because one is sampling from a large population, so one may get four positives in one sample and seven positives in another, even if the true mean is 5.5 positives for a sample of that size), with the amount of sampling error depending strictly on the mean. For two-parameter distributions, the residual variance is estimated directly (as for mixed linear models), so *F*-tests on the fixed effects

become too liberal if there is a missing random effect. For the one-parameter distributions, a missing linear predictor affects the mean, and through that, the estimate of sampling error for each observation.

Over-dispersion

Since a missing linear predictor in the model generates over-dispersion, we also evaluated the usefulness of estimating an over-dispersion parameter, an option offered with PROC GLIMMIX. We wanted to see if such an approach might be used as a scaling factor to compensate for inaccuracies of significance tests and estimated standard errors resulting from the deliberate model misspecification. This is especially important for generalized linear models (GLMs) involving the one-parameter members of the exponential family of distributions because the marginal variance of an observation is a function of its mean (the variance function for the distribution involves the mean). When effects are missing in a model, the marginal variance of an observation, as modeled by its variance function, will be too small by a multiplicative scalar if there is over-dispersion. Thus, for real data, where the model may have missing effects, either because data were not collected (e.g. a covariate), or they involve interactions between fixed and random effects, having some way to compensate for these model deficiencies is important. Over-dispersion can arise for reasons other than missing linear predictors; Young et al. (1999) assess how well an over-dispersion parameter works for a different kind of over-dispersion in the GLM framework. An additional (possibly common) model misspecification is to give the wrong distribution family when coding the model. We investigated these potential model misspecifications by (1) assuming the generating data were normally distributed regardless of the true generating distribution, (2) using a variance stabilizing transformation, and then analyzing the transformed data as if they were normal, and, (3) for one-parameter distributions, fitting a (two-parameter) negative binomial distribution.

Other issues

PROC GLIMMIX differs from most other SAS/STAT procedures (PROCs) in that the options do far more than simply affect what is output, as in PROC TTEST or PROC FREQ. We anticipate an important conclusion from our study: knowing which options need to be specified for the model of interest (among the diversity of models that can be estimated) is critical for obtaining a valid analysis, and, in our opinion, requires both training on this PROC and reading of the (statistical) literature on GLMM models. Without this background, a user is likely to make serious errors while using the software; default settings are often not advisable.

An important issue that surfaced during our investigation was that of non-convergence (i.e. PROC GLIMMIX stopped the estimation process because, using default settings, the algorithm often ran into problems and was unable to provide what it determined to be good parameter estimates). While there are simple fixes for some scenarios, it was nevertheless instructive to know under what situations non-convergence occurred when estimating parameters in our simple models.

In summary, we investigated how well PROC GLIMMIX performed for data sets both when the fitted model was correctly specified, and with the same data sets, how well PROC GLIMMIX performed when the model was deliberately missing terms. For one parameter members of the exponential family, we also evaluated how useful an estimate for over-dispersion was when the model was misspecified. All of these modeling scenarios were examined for 30-block data sets, and more practically achievable 5-block data sets, to examine robustness of PROC GLIMMIX's asymptotic algorithms. Our primary objectives were to identify: (1) conditions under which PROC GLIMMIX appears to produce "reasonable answers" (i.e. estimated parameters are close to those used when generating the data), (2) when frequent model non-convergence is innate to algorithms at default settings (i.e., naïve user), and (3) identify important strengths or weaknesses of PROC GLIMMIX when applied to common modeling scenarios.

We provide examples of scenarios for each distribution below as a reminder of how the various distributions serve to model biological processes generating real data. All examples are based on the same linear predictor, which includes a three-level fixed treatment effect, a random block effect (i.e. a restriction on randomization such that experimental units are grouped physically or temporally into "blocks"), and, potentially, a covariate. They differ by the conditional distribution of the experimental units.

Agricultural Examples

Beta. Prevalence of disease is tested by measuring the proportion of total surface area of a tuber covered by scab. The blocking effect is environmental chamber and the treatments are three CFU (Colony Forming Unit) levels of soil inoculation, with two replicates of each CFU level in each chamber (six tubers were harvested and measured from each of 5 chambers). The soil in each plot was inoculated with one of the three levels every two weeks from planting until harvest, with the objective of seeing how various levels of inocula (i.e., CFU) affected the development of the disease. The covariate is tuber weight. Since the proportion of the total surface covered by scab is a continuous variable, these data can be regarded as samples from a beta distribution.

Binomial. Repellency of compounds are tested using ticks by putting the compound across the middle of a strip of paper, letting it dry, and observing whether ticks will pass through the treated middle zone (ticks of many species will climb up a paper strip if they sense a potential host above them). Eleven ticks (n) are initially placed at the bottom of the paper strip, and the number completely crossing the middle zone within 5 min. is noted. The blocking effect is day, and the treatments are two different compounds and a control. There are six trials (with 11 ticks per trial) per day. The covariate is time of day. This is a classic example of data arising from a binomial distribution, where a proportion of subjects behave in one way, and the rest in another.

Exponential. Many non-desirable invasive weed species in the Great Basin of Nevada are able to germinate faster at lower temperatures than more desirable native plants. To determine the competitiveness of native species used in restoration projects (after fires or mining reclamation), cumulative percent germination in soils from five goldmine reclamation sites was measured for

three treatments over a range of day and night temperature regimes. Soils from the goldmine reclamation sites were utilized as blocks. There were two batches of seeds for each treatment × block combination. The three treatments consisted of seeds from *Oryzopsis hymenoides* (a desirable native species, Indian ricegrass), *Bromus tectorum* (an invasive weed species, cheatgrass), and cold stratified pretreated seeds from *Oryzopsis hymenoides*. The covariate was the pre-trial weight of the batch of seeds. Time required to achieve 50% germination is the exponentially-distributed dependent variable.

Gamma. In an arid or semiarid climate, a desired end point is to increase the infiltration or seasonal total water flux to improve irrigation water availability in the soil profile. The treatments are two alternative crops versus a conventional one. Blocks are areas within a large field, in each block there are six subplots, each subplot planted with one of the three crops. Soil hydrological properties, such as hydraulic conductivity, infiltration, or soil water flux, are measured at the center of a treatment subplot. The initial water content in the profile is the covariate. Soil hydraulic properties are often right skewed; in this example they are considered to be samples from a gamma distribution.

Lognormal. The effect of three different types of dietary fat on body composition, e.g. lean body mass, is studied using an obese-prone rat as a model. The primary endpoint is lean body mass after eating the diets for four months. Because the investigators can only measure body composition in a few rats each week, the study will have a staggered start. The rats will be grouped so that six rats, two per treatment, start the study each week. Groups are used as the blocking factor. Initial body weight is used as a covariate. Previous research has shown lean body mass to have a lognormal distribution in this breed of obese rats.

Normal. A study of the effect of three different soil chemicals on tomato plant root growth is conducted. Sixty pots of soil are individually mixed with one of the three chemicals. Then, each of 180 tomato seedlings of a commercial cultivar is randomly assigned to one of the 180 pots. When the tomato seedling is first planted in the pot its height is measured. The pots are placed in different areas of a field under 30 insect-netting covers that hold six pots each, two pots per chemical. After a month, the plants are removed from the pots and the root dry mass weight is measured for each plant. The treatment effect is the soil chemical and the blocking effect is the insect-netting covered groups within the field. The plant height at time of potting is a covariate. Previous studies have shown root dry mass weight to have a normal distribution for this cultivar.

Poisson. *Bacillus thuringiensis* (BT) produces a toxin that is often used as a biological alternative to an insecticide. One method of use is to genetically modify crops to include the BT gene. Three BT Corn Varieties are evaluated for resistance to tarnished plant bugs. Corn varieties are planted in plots two feet wide and six feet long, and each block contains 15 of these plots (five per variety). Measurements of resistance are taken by counting the number of tarnished plant bugs on an ear of corn, one ear is sampled at random from each plot (i.e., 15 ears are used per block). These counts follow a Poisson distribution. The covariate is the size of the ear of corn.

2. Methods

Data were simulated for the seven distributions in Table 1 under different conditions (Table 2). The data were generated in SAS using code developed by M.J. Camp; the code also estimated the requested GLIMMIX models (Table 3) and saved the results. Table 4 provides the anticipated consequences on model estimates when misspecifying the linear predictor by leaving out a term, along with a brief explanation.

The data were generated to simulate conditions for a randomized complete block design (possibly including a covariate on individual observations), with three treatments, a block effect, a block × treatment interaction, and replications of treatments within each block. We attempted to maintain a similar effect size for both fixed and random effects (i.e., all the effect sizes were similar), based on link-scale means differing by 10% from each other (e.g., true means of 9, 10, and 11; typically the effect size in which researchers say they are interested). Actual values differed among the distributions in order to maintain similar effect sizes and create sensible data, and are given in Table 1. We used simulation to set and verify that effect sizes were as desired, and adjusted the sample size (total number of experimental units) to yield a power of about 80% for simulation sets with 30 blocks. For those simulations where a covariate was also generated, its effect size was adjusted to yield about 80% power. Thus the effect sizes of all random and fixed effects were approximately the same, for all the distributions examined, although "true" means and variances differed among the distributions, as did the sample size per treatment-block combination.

The motivating experimental design underlying our simulation data can be described as a generalized randomized block design (Wilk, 1955) with blocks being considered random effects and with each block consisting of enough physical units so that each of three treatments could be replicated multiple times. For example, blocks might be fields from randomly selected locations where each field is subdivided into six plots and where the three treatments are randomized to plots within each field so that each treatment is replicated two times. We chose to simulate data from a generalized randomized block design so that (for the case of simulating data from the Normal distribution) variance within block and treatment could be separated from the variance of the block × treatment interaction. We also considered a modification so that (as in some of our simulations) an additional observation on each experimental unit could be used as a covariate.

Because we wanted to see if variance estimates were biased, we started with 30 blocks in our simulations to provide a sufficient number of samples (of blocks) to get reasonably good estimates of the block variance. However, most of the experiments we are familiar with are designed with far fewer blocks, so we also looked at results using only five blocks.

As an example of how we determined parameters, the following steps were used for the gamma distribution. The data set was generated (starting on the link scale) from log means of 4.5, 5, and 5.5 (to give the 10% difference among means), the variances for the block and block × treatment interaction were both 1.9 (so the effect sizes of the random and fixed effects were approximately

the same). Data for treatment 2 (log mean = 5) were generated from a gamma distribution with scale parameter $c = 5$ and shape parameter $b = 29.68$ (with $c = 5$, an integer value, the distribution is actually an Erlang, which is a special case for the gamma). The untransformed mean then is $bc = (29.68)(5) = 148.41$ and the variance is $b^2c = 4405.3$; the corresponding untransformed mean for treatment 1 is 90.02, and that for treatment 3 is 244.69. There were problems using smaller mean values, resulting in not achieving the desired power value. If the block or block × treatment interaction effects were larger, then for smaller means the distribution sets looked and tested exponential. In the end, even with the selected means, some sets or subsets tested more lognormal than gamma using goodness-of-fit tests.

Variation in observed data is determined by the type of distribution one is sampling from. In members of the one-parameter exponential family, it is solely a function of the observation's mean. For members of the two-parameter exponential family, it is associated with a distinct scale parameter (or some function of the distribution's two parameters). Thus, if one is sampling from the one parameter Poisson distribution, the mean is estimated by the count, and so is the sampling error (variance), i.e. they are the same. Having two replicates from the same block-by-treatment combination is not required to estimate the variance of that cell; though averaging the two counts should be a better estimate of that cell's variance than using either single count. However, if one was sampling from a normal distribution, one could not estimate the cell sampling error (variance) from a single replicate, one would need at least two replicates. Since similar code was used for all the sampling distributions and we needed to estimate a within cell sampling error (residual variance) for the two-parameter distributions, the situation is somewhat unnatural for the one-parameter distributions, with two or more observations for each block × treatment combination. If one was sampling from a Poisson distribution when collecting real data, typically each block × treatment combination would only be counted once. If two counts or replicates were taken rather than just one, additional variation would be expected to be present, and might include a replicate within (block × treatment) variance parameter in the estimated model. Because in our simulations there was no additional variance at this level (i.e., only the usual sampling error that would accompany sampling from a Poisson distribution with a particular mean), we did not include that term in the model (and, indeed, when we did, it was estimated to be near zero). For all distributions, the covariate was introduced at the replicate level, so the two replicates from the same block × treatment combination each had a unique covariate value. We gave examples of how data might arise for each sampling distribution used in our study above.

Under different conditions (Table 2), 5000 data sets were simulated and fit using PROC GLIMMIX to four different models (Table 3) that differed by the effects included in or omitted from the model, i.e., the data were the same but the estimating models differed. These models were as follows: (1) a model that excluded the block × treatment interaction random term (i.e. $g(\mu_{ijk}|B_j) = \alpha + \tau_i + B_j$), where $g(\cdot)$ is a suitable link function for the response Y_{ijk}, $B_j \sim N(0, \sigma_B^2)$, τ_i is the *i*th treatment effect, and B_j is the *j*th block effect, (2) the "true" model, the model used to generate the data, $g(\mu_{ijk}|B_j, \tau B_{ij}) = \alpha + \tau_i + B_j + \tau B_{ij}$, $\tau B_{ij} \sim N(0, \sigma_{\tau B}^2)$, τB_{ij} is the *ij*th interaction effect of block and treatment, (3) the "true" model included a scale

parameter to be estimated with the random statement by using the "_residual_" option (i.e. $g(\mu_{ijk}|B_j, \tau B_{ij}) = \alpha + \tau_i + B_j + \tau B_{ij}$ with $Var(Y_{ijk}) = \phi \cdot V(\mu_i)$ where ϕ is a free scale parameter and $V(\mu_i)$ the appropriate variance function associated with the distribution function for Y_{ijk}), and (4) a model that excluded the block × treatment interaction random term but included a scale parameter estimated with the random statement using the "_residual_" option. Representations of these models, and random statements used to estimate these models, are given in Table 3. Table 4 summarizes these models and also provides a brief explanation of the anticipated consequences from misspecifying the estimating model. For models where a covariate was also generated, we again had a "true" model, a second where the covariate term was omitted from the estimating model, and a third where the covariate term was omitted but an over-dispersion parameter was added to the estimating model. These models are also given in Tables 3 and 4 (denoted as models 5, 6 and 7, respectively). Including an over-dispersion parameter for a two- parameter distribution (which already has a scale parameter) does not make sense; over-dispersion parameter results are not given for these distributions.

Historically, data were often treated as normally distributed, even if it was obvious that standard errors of means increased with the mean. This common model misspecification was included, i.e., we analyzed the data sets as if they were generated by a Gaussian process. We also investigated the effects of applying a variance stabilizing transformation and of specifying a different distribution (possibly remedial) in the model to try to compensate for the over-dispersion created by the missing terms. The variance stabilizing transformations, which differ by distribution, are given in Table 1. While the intent of these transformations is to rescale the data in a way that makes them suitable for analysis in a mixed linear models framework, the rescaling affects relationships among groups (e.g., if treatments A and B were most similar on the original scale, treatments B and C might be most similar on the transformed scale). In addition, back-transformation of means and coverage intervals may produce estimates that are far from those produced by the appropriate GLMM analysis. We did not investigate back-transformation issues in our study.

For one-parameter members of the exponential distribution family, missing terms leading to over-dispersion might be remedied by designating a two-parameter distribution. Thus, for these distributions, we also fit a model specifying a (two-parameter) negative binomial distribution. For the binomial, this can be justified (for $p < 0.5$) by noting that the variance will increase with the mean, as it does in a Poisson distribution, and with over-dispersion, the distributions may be difficult to distinguish. Further, n may not be collectible in some experiments (males are eaten, adults disperse). It is a longer stretch to consider the negative binomial as appropriate to model an over-dispersed exponential distribution. Both the exponential (continuous) and the negative binomial (discrete) can be conceptualized as waiting time distributions for an event. The one-parameter exponential distribution has the 'lack of memory property', where the probability of an event is invariant to time. When $X \sim$ Neg Bin (r, p), $0 \leq p \leq 1$, the mean is r/p and the variance is rq/p^2, with r = number of successes before the first failure, and $q = 1 - p$. When $X \sim$ Exp (λ), the mean is $1/\lambda$ and the variance is $1/\lambda^2$. If $1/\lambda = r/p$, then the means for the negative binomial and exponential distributions are equal. When $q = r$, the variance for the two distributions are equal. If $rq > r^2$, the variance of the negative binomial exceeds that for the

corresponding exponential. However, since r takes on only integer values and $0 \leq q \leq 1$, this is unlikely to occur (however this does show that the negative binomial can be an appropriate model for an under-dispersed exponential distribution). If one relaxes the restriction on r to allow non-integer values with $r > 0$, then the negative binomial distribution can model an over-dispersed exponential distribution; we use this justification to motivate our use of the negative binomial distribution.

All models were fit using the default options in PROC GLIMMIX, as a naïve user (someone who is comfortable using SAS, specifically many of the STAT PROCs, but doesn't know much about PROC GLIMMIX) might do, certainly on the first modeling attempt. Thus, we did not choose starting parameters, bounds, estimation method, etc. Part of the reason for this is that we felt that this should be the usual approach for using the software, since SAS/STAT products are used by non-statisticians (such as the researchers we work with), as well as by applied statisticians. New users will not recognize that understanding the options and a good deal of the theory behind them is key to producing a correct analysis; this is not true for many of the other SAS/STAT PROCs.

The PROC GLIMMIX default options were: (1) Estimation METHOD=RSPL (Subject-Specific Residual Pseudo-Likelihood), (2) DDFM=CONTAIN, and 3) LINK=LOGIT (for Beta and Binomial), LOG (for Exponential, Gamma, and Poisson), and IDENTITY (for Lognormal and Normal). Many users with some training, e.g., with PROC MIXED, will have learned that setting DDFM=KR is recommended. We initially tried that option but found that the estimated degrees of freedom varied from one simulated data set to another; using the default DDFM=CONTAIN produced the same degrees of freedom for every simulated data set for a given condition.

Computing Methods

The data for the distributions were simulated with SAS v. 9.2 (TS Level 2M0) (SAS Institute Inc. 2010) software. Using SAS's macro language, macros were written to generate the data, run the PROC GLIMMIX models under the various conditions, and save and manage the output from PROC GLIMMIX. Following SAS's recommendation, the RAND function was used to simulate the distributions. Based on an efficiency study by Novikov (2003), an index variable was created with a unique value for each data sample. This allowed the data sets to be modeled in PROC GLIMMIX using by-group processing, which greatly reduces the time needed to model the data. Options in the Output Delivery System were used to keep the output from filling the computer's random access memory during modeling.

Estimated Values

One common difficulty with software of this nature is that the model estimation may not converge under default settings. Typically, it is unclear whether this is due to a data or model problem (potentially solved by specifying particular options) or related to the estimating algorithm. If one knows the true model used to generate the data and specifies it for estimation, then poor convergence cannot be due to data or model issues, but can only come from the

algorithm if the data set is sufficiently large (and the definition of 'large' may differ by distribution). Surprisingly, there were many instances of low convergence rates with some conditions and distributions when the model used was known to be the true one.

In addition to convergence rates, for those models that did converge, we saved parameter estimates from the various fixed and random effects specified in the model, the over-dispersion parameter (if present), the *F* values for the treatment effects, and the associated *p* values. In some instances, we examined the correlations between parameter values. We examined the output parameter estimates to see if they matched those used to generate the data, and what, if anything, changed when the model was misspecified. We examined changes in *p* values for misspecified models (power and Type I error), in particular, we were interested in whether omitting terms grossly affects *p* values (since, with real data, every model will be somewhat misspecified), and how *p* values change as one reallocates observations to, say fewer blocks, but more observations per block. Table 2 gives the various conditions that were examined for all distributions. Since variance stabilizing transformations are commonly used for data when the variance appears to vary with the mean, as it does for most members of the exponential family, we were also interested in whether *p* values from transformed data, subsequently analyzed as if normally distributed, were similar to those based on the appropriate GLMM model. This is important because there are far fewer convergence issues if one models (transformed) data using the normal distribution than for other distributions. So a reasonable route, if one has convergence problems, is to try a variance stabilizing transformation; yet other important limitations of reliance on a transformation approach are discussed below.

Over-dispersion

The term "over-dispersion" refers to more variation displayed by data than what is expected under an assumed model. For generalized linear models (GLMs), an assumed model may imply a relationship between the mean and variance of the marginal distribution for the random variable being modeled. This relationship has the form $Var(Y) = V(\mu)$ where Y is the random variable, μ is its mean and $V(\cdot)$ is some function. For example, for a random variable Y that is a binomial proportion, $\mu = \pi$ where π is the probability of success for a given trial, and the relationship between the mean and variance is $Var(Y) = \mu \cdot (1 - \mu)$. When modeling proportions assumed to be binomial but the data suggest that $Var(Y) \gg \mu \cdot (1 - \mu)$, a free scale parameter ϕ is used as a simple solution to correct the discrepancy so that $Var(Y) = \phi \cdot \mu \cdot (1 - \mu)$. Similar corrections apply for other one-parameter distributions such as the Poisson and exponential distributions. For GLMs, Wedderburn (1974) suggests estimating ϕ with the sum of squares of the 'Pearson' residuals divided by the residual degrees of freedom.

The problem of over-dispersion is carried over to GLMMs, as they are GLMs with random effects included in the linear predictor, and possibly a structured residual covariance (e.g. due to, say, repeated measures on the same experimental unit). Our simulations generated over-dispersed data by virtue of the random effects we included in the simulations (but did not specify in the estimating model). The data were necessarily over-dispersed for fitting GLMs to them because of the model we used to generate them. PROC GLIMMIX provides this over-dispersion

correction through the random statement when used with the keyword "_residual_". In our study, we fitted GLMMs with the "random _residual_" statement to see how well the scale parameter ϕ was estimated and to explore whether it could be used for correcting test statistics for misspecified models, in the same way it is used to correct for over-dispersion for GLMs that are misspecified. PROC GLIMMIX uses the sum of squares of "Pearson" residuals divided by the residual degrees of freedom to estimate ϕ. We computed residual degrees of freedom as $N - g - 1$, where N is the number of cases in the data set and g is the number of G-side parameters that are estimated. For more details of this computation, refer to SAS Help and Documentation for PROC GLIMMIX (SAS Institute, Inc. 2010). The Wald F-statistics reported in the output of the fitted model using PROC GLIMMIX with the "random _residual_" statement are scaled versions of the F-statistics that would be reported without the "random _residual_" statement. That is, the F-statistics reported for testing a treatment effect for a fitted model using PROC GLIMMIX with the "random _residual_" statement is $F/\hat{\phi}$ where F is the F-statistic computed when fitting the model without the "random _residual_" statement and $\hat{\phi}$ is the estimate for ϕ when fitting the same model with the "random _residual_" statement.

3. Results

Below we give general results for the models when the estimating models match the true models generating the data (i.e. Models 2 and 5, Table 3) and, subsequently, when the estimating model is misspecified (i.e. Models 1, 3, 4, 6, 7). Following that are detailed results for each distribution (additional information is found in Tables 5 – 8 and Fig. 1).

Results when generating and estimating models match (Models 2 and 5, Table 3)

In general, the proportion of fitted models that converged (referred to as convergence rate in this paper) was high for all the distributions except the beta, exponential, and Poisson (Tables 5 and 6). The number of blocks had a large effect on convergence rates for these latter distributions; convergence rates were higher for the five block models than for the 30-block models. For the 30-block model, the exponential distribution was particularly difficult for PROC GLIMMIX to fit with the default options—less than 30% of the models converged. Surprisingly, the Poisson distribution (with slightly over 40% converging for the 30 block model) was also problematic using the default options (though not when options were changed—see details for the Poisson distribution below).

Since not all simulations converged, our definition of power is based on only those data sets where the model converged (Table 7). We created the data sets to have approximately 80% power (for the F-test on treatment) for the 30-block model and investigated how the power changed as we reallocated observations to fewer blocks (power will decrease since the block and block × treatment interaction variances are less well estimated, increasing uncertainty in the fixed effect parameter estimates). Power decreased markedly from Condition 1 to Condition 2 (Fig. 1) for all the distributions tested; in most cases decreasing to less than 40%, even for the same total number of observations. Thus, the number of blocks in an experiment has a substantial impact on how well one can detect fixed-effect treatment differences. While for

linear mixed models the relationship between the number of blocks (as a random effect) and the power to detect treatment differences is well known to statisticians, researchers (who typically design their own experiments) are too often ignorant of this relationship, and focus instead on total *N*. Our results confirm that this is also a design issue in the GLMM framework for all distributions examined.

If all the treatment means are set equal, Type I error rates should be about 5%. For the beta distribution, Type I error was slightly higher than 5%, for the binomial and Poisson distributions it was lower (Table 8). The exponential distribution had only a 2% Type I error rate for Condition 5 (5 blocks), but was somewhat greater than 5% (5.6%) for the 30-block simulation set, though with low convergence rates, these percents are not always accurately estimated. Based on the binomial distribution, two standard errors on 5000 simulations for $p = 0.05$ is about 0.006, thus, if everything was working correctly, Type I error rates should have ranged from 4.4% to 5.6%. The Type I error rate fell well outside these limits for some of the distributions (Table 8). Thus, it appears that Type I error rates are not correct under certain conditions, even when the model is known to be "true". We do not know the reason for this.

In general, the estimates of the variance parameters and treatment means were very close to those used to generate the data; some of the variance parameters showed a small but consistent bias. One exception to the generally good parameter estimates was that both the block and block × treatment variance components were estimated to be zero in almost all simulations which converged for the exponential distribution. For this distribution, the algorithm seemed to have difficulty keeping some variance components away from zero. Some additional bias results for other distributions are given below in the summaries for each distribution. Littell et al. (2006) suggested using the 'nobounds' option (to allow variance components to become negative during the estimating iterations), which may produce better final variance estimates (and avoid premature termination of the estimation routine). However, as explained above, default options were used in this study.

Results when generating and estimating models do not match (Models 1, 3, 4, 6, 7, Table 3)

A likely scenario when analyzing real data in the GLMM framework occurs when predictor variables, either fixed or random, are missing. We looked at examples of both. We modeled the data without the block × treatment interaction effect and found that, for every distribution, the proportion of significant tests on the fixed treatment effect increased, that is, the tests for treatment became too liberal when the model was missing a random effect (Table 7). This erroneous (but expected) apparent increase in power was most noticeable for Condition 3 (5 blocks; Table 2), where the power for the true model was low. For most distributions in this set of simulations, the power increased from 20% for the true model to more than 80% for the model missing this variance component. This problem is particularly apparent for the sets of simulations where the means were set equal (Table 8); Type I errors increased from about 5% to over 60% for all but the binomial and exponential distributions (in the latter two, Type I errors increased to 34% and 8%, respectfully). The variability associated with this missing variance component was, at best, only partially absorbed by other variance components in the model; the

block variance did tend to be slightly overestimated when this interaction term was missing. The *F*-tests were distorted because the data were over-dispersed for the fitted model. In contrast, modeling the data without the covariate predictor had essentially no effect on the *F*-test for the treatment fixed effect, i.e., power was not affected nor did it introduce bias into the means of the treatment effects. In general, the treatment means and variance parameter estimates for models missing either a random or fixed linear predictor were accurate, with only a slight upward bias of the block variance parameter, as noted above, when the random term was missing from the estimating model.

Models can also be misspecified by furnishing the wrong conditional distribution (i.e. the wrong distribution family). We found that, by misspecifying a distribution as normal, convergence rates were nearly all 100% and power was largely unaffected (i.e. *F*-tests on treatment effects were not affected by this type of misspecification), though it did decrease (was more conservative) for the exponential and lognormal distributions (Table 7). Even though an important rationale for using PROC GLIMMIX is to account for the relationship between the mean and variance for non-normal distributions, which is ignored if the distribution is specified as normal, this did not appear to be an important issue in our simulations; certainly not in comparison to the very large problem created when dropping a random effect from the model. We are aware that the chosen basic model and how the parameters of that model were populated were likely responsible for the observed small effect of loss of power due to conditional distribution misspecification. Parameters and models could certainly be chosen to demonstrate that conditional distribution misspecification has serious negative consequences (in addition to affecting *F*-tests, estimated means can be badly biased). What is unclear (and not investigated in this paper) is under which conditions distribution misspecification does matter. We show only that under the conditions investigated this type of model misspecification did not greatly affect *F*-tests.

For many distributions in the exponential family, there is a historically recommended variance stabilizing transformation designed to allow the transformed data to be analyzed as if they were normally distributed. For example, for the binomial distribution, the arcsine-square root transformation approximately stabilizes the variance (to meet the homogeneity of variance assumption of linear models). We applied the recommended variance stabilizing transformations to our simulated data and found that, except for the exponential distribution, power was preserved (Table 7) and convergence rates were at or near 100% (Table 6). The variance stabilizing transformation did not work well for the exponential distribution; power dropped by about 25%. Our results suggest that if convergence is an issue, except perhaps for the exponential distribution, a variance stabilizing transformation can be used, but we recommend it only as an initial step to obtain starting parameter estimates and to see which effects are important because, as mentioned above, relationships among means on the transformed scale may be quite different from those on the original scale. Our results also suggest that the variance stabilizing transformation is not remedial for a missing random effect; that is, power for Model 1 (missing the random block by treatment interaction effect) was about the same if the generating distribution was used or if the data were transformed and the normal distribution was used (except for the exponential distribution).

A possible alternative remedial distribution for over-dispersed data from a one-parameter member of the exponential distribution family is to use a two-parameter distribution. The negative binomial distribution has been suggested (SAS Institute, Inc. 2010) as a possible remedial distribution for data coming from an over-dispersed Poisson distribution, thus we tried it for the Poisson distribution and for other one-parameter members for Model 1. For no distribution did the negative binomial distribution decrease power to the correct level (80%). Thus, at least for the situations we examined, the negative binomial does not compensate for over-dispersion resulting from a missing random effect.

Below are our observations of the simulation results for each distribution.

Beta Distribution

Using true Model 2 (Table 3), parameters were selected (Table 1) to simulate beta-distributed data ($0 < p_i < 1$) in the neighborhood of 0.20 ($\mu_1 = 0.175$, $\mu_2 = 0.20$, $\mu_3 = 0.225$); a region of the range where there is a strong relationship between the mean and variance and the benefit from using an arcsine($\sqrt{p_i}$) transformation is realized. Beta-distributed variates, p_i , occur as proportions whose n (i.e., denominator) on which the proportion is based is not known; or when there is need to model over or under-dispersed binomially-distributed data. For Models 2 and 5 ("true" models) the block variance estimates were accurate, ranging from $\sigma^2_{block} = 0.156$ to $\sigma^2_{block} = 0.164$ (true link-scale value = 0.16) and the block × treatment variance (true link-scale value = 0.09) were also accurate, ranging from $\sigma^2_{block \times trt} = 0.085$ to 0.097. The 3 treatment means were consistently biased upward for all estimating models (1, 2 and 5; Table 3). For Model 1, the average bias (over 5,000 simulations) ranged (across all conditions examined) from 2 to 4% above the true data scale treatment mean values. For Models 2 and 5, the average bias ranged from 1 to 2.7% above the true treatment mean values.

Convergence rates (Table 5) for the true beta model, with two replicates per block-treatment combination, were comparable for 30 blocks (83.1%) and for 5 blocks (81.4%), but low; yet improved (87%) for 5 blocks with 12 replicates. Under the same three conditions, the convergence rate was better for Model 1 (excluding a random effect) than for (true) Model 2; PROC GLIMMIX algorithms (at default settings) converged substantially (2 to 10%) more often for an incorrect model (due to omission of a random effect) than for the correct model. Under normality assumptions, convergence rates were 100% (Table 6) regardless of whether a normal distribution (Table 1) was fit directly to beta-distributed data (Condition 6) or was fit to transformed data (Condition 7). When fitting a model that omitted a linear covariate term that was present in the generated data (Model 6), the convergence rate (79.5%) exhibited no appreciable change from the convergence rate (81%) of the true model, Model 5 (Table 5). The average time required to fit each of the 5,000 beta models ranged from 0.07 to 7.4 seconds, even with 13% to 19% non-convergence rates.

The observed power (Table 7) of the true beta model for detecting significant differences among the treatment means was very close to the 81% true power; regardless of condition (1, 6, 7 or 9;

Table 2). For each condition, the observed power for Model 1 was ~12% greater than for the true model; indicating that failure to model all random effects consistently resulted in liberal *F*-tests for the fixed treatment effect. Highly inflated observed Type I Errors (Table 8) corroborated this finding. The treatment effect *F*-tests were also too liberal for the true model. Even though the data were simulated with true α=0.05, false positive treatment effects were observed in 5.4% of the 5,000 simulations, regardless of number of blocks used (Conditions 4 or 5); and with a slightly higher (5.6%) frequency using transformation (Condition 8).

Binomial Distribution

Treatment means for the one-parameter binomial distribution were centered on zero (on the logit scale, 0.5 on the proportion scale), not a region where the arcsine-square root transformation is needed. Convergence rates were uniformly high and estimations rapid. Like the other distributions examined, power and Type I error erroneously increased when the model was misspecified by omitting the random block by treatment interaction term.

The negative binomial distribution should not be used as a remedial distribution, even if all *p*'s are small (see above); the power estimate dropped from 80% (estimating distribution as binomial) to 46% (same data, estimating distribution as negative binomial) for Model 2 (Table 7). However, assuming that the data were normal or using the arcsine-square root transformation did give reasonable results for power for the simulations settings used. As noted above, this should be expected for means close to $p = 0.5$, as they were for the simulations done for this distribution. Inclusion of the over-dispersion parameter (which was frequently estimated to be < 1) did not compensate for the missing random effect.

Exponential Distribution

Unexpected results were obtained when using a simple, one-parameter exponential distribution for generating simulated data. Lack of convergence was the major stumbling block for obtaining accurate estimates of model parameters, the over-dispersion parameter, and bias. The true model consistently had the worst convergence rates of all the models under all tested conditions, ranging from 25% to 86%. Anomalous results occurred when the number of blocks was reduced while keeping sample size low—convergence rates almost doubled (27% to 48%) at a cost of severe power loss (86% to 11%). This same doubling of convergence rates (27% to 53%) was seen when more replicates were reallocated to fewer blocks without as much loss in power (86% to 61%). The strangest convergence issue related to exponentially distributed data was that the true model had the covariance parameter estimates, block and block × treatment, of zero for those simulations that converged, whereas simulations that did not converge overestimated those same covariance parameters such that averaging the converged and non-converged estimates produced better estimates of all parameters.

Power was ill-defined since it could only be calculated for those simulations that converged, and convergence was generally quite low. Type I error estimates tended to be slightly overestimated for all models tested, ranging from 5.4% to 6.8%. When an over-dispersion parameter was

included in the model, model parameter estimates tended to be lower for the data sets that did not converge, and was most inaccurate for Model 3 (an over-dispersion parameter added to the true model) due to it having the lowest convergence rates. There was more bias present (both positive and negative) for the random effects of block and block × treatment in all models than for the fixed effect parameters.

When a normal modeling distribution was assumed for exponentially generated data without a log(*Y*) transformation, 100% convergence was obtained with no great loss of power (86% to 81%), at a cost of overestimating the random effects. When using a normal modeling distribution with a log(*Y*) transformation, convergence was still 100% but resulted in a 25% decrease in power (from 86% to 61%), as well as overestimation of random effects and underestimation of Type I error. When a negative binomial modeling distribution was assumed and no data transformation performed, there was no change in low convergence rates (27% to 25%), so power was still ill-defined, and resulted in underestimation of the random block effect.

Gamma distribution

There were severe consequences for model misspecification, often resulting in low convergence, poor power and poor Type I error rates, and bias in estimates of treatment means and covariance parameters. These problems were especially pronounced for Model 1 and Model 4 results; hence, any inferences on results from these models are from a relatively small number of cases and must be considered with caution (we intentionally omit some results from our tables for this reason). Simulations with five blocks lowered power by about 55%. For log-transformed data under normal distribution assumptions, convergence was 100% and power and Type I error results were reasonable only for Model 2, but also produced somewhat larger biases in covariance parameters and estimates of treatment means. Results with an added covariate were very close to those of Model 2, but only when the covariate was included in the analysis, and the convergence rate was almost 100%. Adjusting convergence criteria and iteration options can markedly improve the convergence rate. For example, in Tables 5 and 6, when the options for Model 1 were set to allow for up to 50 iterations, the convergence rate changed to 98.4%. See the Poisson results for more discussion on this topic.

Lognormal Distribution

All simulations converged for the lognormal distribution when the estimating distribution was correctly specified as lognormal or when it was specified as normal. Power and Type I error were identical when the response, *Y*, was modeled using the lognormal distribution and when log(*Y*) was modeled using the normal distribution. When the data were modeled using the normal distribution without transforming the responses, power decreased to 60% for the true model and the estimates of the treatment means were biased by a factor of $\exp(\sigma^2)$. No other bias in treatment means was observed. When the estimating distribution was correctly specified as lognormal and the block by treatment interaction was omitted from the model, power increased from 79% to 93% and the Type I error increased from 5% to 15%. The estimates of the block variances were inflated by approximately 20%.

Normal Distribution

The normal distribution always converged under all conditions and models. The estimated power depended primarily on the number of blocks, less so on the number of replicates. For 30 blocks and two replicates, the power for Model 2 was 81.1%; under Model 1 (missing the treatment- by- block interaction) it was 93.4%. When the number of blocks was reduced to five (1/6 *N*), the power fell to 15.5% and 41.8% respectively. Even when the number of replicates was set to 12 with five blocks, (thus providing the same number of experimental units as with 30 blocks and 2 replicates) the power did not increase greatly for Model 2 (rising only from 15.5% to 20.0%). However, there was a larger increase, to 89.0%, for Model 1. The Type I error rate for Condition 1 was 4.8% for Model 2 (within expectations) but 16.8% for Model 1 (with the missing term). When blocks were reduced to five and replicates increased to 12, the error rate was 5.5% for Model 2 and 69.7% for Model 1. Type I error inaccuracies were thus more pronounced for fewer blocks. Under all conditions and models, the average treatment mean estimates were very close to their true value. On average, the block variance estimate increased when the block × treatment effect was omitted (Model 1). For data generated under the various simulation conditions the average block variance estimate for Model 1 was 12.2% to 18.4% higher than the true block variance. When normally distributed data were modeled as lognormal the power was 4.3% for Model 1 and 17.3% for Model 2. The back-transformed treatment means were, on average, positively biased by one unit (2%).

Poisson Distribution

For Model 2, when the generating and estimating models matched, PROC GLIMMIX had a poor convergence rate with a large number of blocks; this convergence rate improved greatly when the number of blocks was decreased. The convergence rate was 45% with 30 blocks, but 87% (n = 30, Condition 3) and 96% (n = 5, Condition 2) when blocks were decreased to five. We are unclear as to why this occurred. As expected, power decreased as the sample size and the number of blocks decreased.

The simulated value for the variance components were chosen so that power would be approximately 80% for blocks = 30 and n = 5. However, when the true model was fit to the data, the Type I error rate was too low (3.97%). The true means for the 3 treatments were 3.5, 4.0 and 4.5. The average treatment means from the 5000 simulations were slightly larger (one to three percent increase) than the true means for all conditions and all models (except for Conditions 7 and 8).

When the generating and estimating models did not match, the convergence rate improved when the negative binomial was used to model the data (86% for 30 blocks and 94% for 5 blocks). Specifying the normal distribution yielded 100% convergence. Modeling the data as if they had come from a negative binomial or normal distribution gave approximately 80% power and improved the Type I error rate (but they were still low). As expected, leaving the block × treatment variance component out of the model resulted in increased Type I error rates. Adding

an over-dispersion parameter in the model did not appear to help correct for the model misspecification.

Using Laplace estimation method increased the convergence rate when using the correct model. For 30 blocks, 5 samples and Model 2 (correct model), the convergence rate increased from 45.1% (see Table 5) to 100% with 'method = Laplace'. In order to improve convergence with the pseudo-likelihood default method, a special set of runs were used to check out options to assist with convergence for a Poisson distribution. One thousand data sets were simulated with 30 blocks and 5 replications per block-treatment combination. When PROC GLIMMIX fit the true model with no additional options, the convergence rate was 49%. Changing the 'pconv' options had the biggest effect on improving convergence rate. This option sets the criteria for deciding when the iterative process shows little change in estimated parameter values. Decreasing the sensitivity, by changing pconv from 1e-08 (default) to 1e-06, increased convergence to 98.8%. Increasing maximum iterations to 50 with the 'maxopt' option (with pconv at its default) increased convergence slightly to 51.4%. Using the 'subject=options' on the random statement also increased convergence slightly to 51.4%. Using all three options increased convergence to 99.3%.

Figure 2 compares results between the default and LaPlace estimation methods for Model 2 (the correct model). As indicated above, using LaPlace estimation greatly improved percent convergence. However, when the means were set equal (Conditons 4 and 5), Type I error rates were consistently greater than 5%, especially noticeable for simulations with five blocks. Figure 2 also depicts values (or estimates) for the variance parameters, which were consistently underestimated using the LaPlace estimation method. These results are consistent with the kinds of biases accompanying maximum likelihood methods (of which LaPlace is one).

Over-dispersion

Adding an over-dispersion parameter only makes sense for one-parameter members of the exponential family, since the two- parameter members do not have the tight linkage between the mean and variance. We included the "random _residual_" option to fit simulated data with Model 3 for the Poisson, binomial and exponential distributions to test whether the mean estimate of ϕ would be 1, as should be expected, and with Model 4 (the intentionally misspecified model with the block × treatment term omitted) to evaluate if its inclusion would produce unbiased *F*-tests for the treatment effect. The mean estimate of ϕ when fitting Model 3 was approximately 0.98 for all three distributions investigated. The slight effect of the underestimation of ϕ is revealed by comparing Models 2 and 3 in Table 7 where the observed power for Model 3 is slightly larger than that for Model 2 for each distribution. The reason for this is because the *F* values for Model 3 are the same *F* values of Model 2 but multiplied by $1/\hat{\phi}$. We also included the "random _residual_" option to fit simulated data with Model 4 for the Poisson, binomial and exponential distributions to see whether the resulting *F*-tests would perform similarly to the *F*-tests corresponding to the fitted Model 2. Inspection of Table 7 reveals that the power for the binomial and Poisson distributions to be substantially larger for

fitting Model 4 (with a missing term) than for Model 2, suggesting the over-dispersion parameter is severely underestimated when random effects are unaccounted for in the model.

It appears that $\hat{\phi}$ will generally underestimate ϕ when random effects, not accounted for in the model, do have a substantial impact on the model. In a simulation to illustrate this for the binomial distribution, we conducted 500 Monte Carlo simulations under Model 2 for the binomial distribution and compared all pairs of *F* statistics resulting from fitting both Models 2 and 4 to each data set (Figure 3). These *F*-statistics in Fig. 3 indicate that the *F* values computed with the scale parameter adjustment for Model 4 are too large and could be easily scaled to accurately approximate the *F* value computed for Model 2 by adjusting $\hat{\phi}$ by a multiple scalar. This suggests that the degrees of freedom used for computing $\hat{\phi}$ should be some smaller amount than what is presently used with PROC GLIMMIX. We do not recommend using the 'random _residual_' statement to adjust for over-dispersed data when fitting a GLMM (i.e., when random effects are deemed a necessary part of the linear predictor.)

4. Conclusions

We investigated how well SAS PROC GLIMMIX handled data simulated from a variety of distributions, with models that either matched those used to simulate the data or were misspecified in some way. The reason to look at misspecified models was because the "true" model is typically unknown, so models for real data are misspecified in one way or another; we thought it useful to know what kinds of model misspecification were most risky, and which were least important.

However, to even start looking at whether model estimations were reasonable, the estimation algorithm must converge (i.e., the best parameter estimates for the specified model have been identified), and PROC GLIMMIX had trouble with convergence under default options for some distributions, particularly the exponential and Poisson distributions. One should expect high convergence rates when the generating and estimating model match, total *N* is reasonably large, and the statistical model simple; this suggests that PROC GLIMMIX default settings often do not allow PROC GLIMMIX to produce a correct model fit, even when the correct model is correctly specified. It may simply be a matter of using distribution-dependent defaults, or may be due to effects of the approximations used in the estimating algorithm. In any case, other analyses, such as power and parameter biases, became complicated due to successful and failed estimations having different distributions for final parameter estimates. We base our conclusions on only those simulations that converged, but that clearly biases some findings. Our basic convergence finding is that, for some distributions, lack of convergence does not necessarily indicate that there is a problem with the model. If one believes the model to be close to "truth", rather than tinker with the model it is probably a better strategy to tinker with the options to try to get convergence. If that does not work, use a variance stabilizing transformation, which should at least provide a ball-park estimate of the importance of the various effects in the model. Our results suggest that, for the kinds of data we simulated, significance tests on fixed effects for transformed data in a LMM framework are close to those produced by PROC GLIMMIX for a GLMM. This kind of comparison does not validate the use of variance stabilizing

transformations; as mentioned above, their use creates other problems (that we did not investigate).

Omitting an important random effect appears to be the most consequential model misspecification we found, affecting all distributions, and not remedied by including an over-dispersion parameter, variance stabilizing transformation, or modeling using another distribution with a scale parameter. Since the fixed effect tests become too liberal, this misspecification could result in making claims unsupported by the data. We currently have no suggestions for a remedy. Very likely the missing random effect is excluded because the researcher is not aware of its existence (or importance) so it is never entered into the model to be checked. What we do suggest is that the analyst checks the importance of all potential random effects for the data set. These include constraints on randomization, blocking, and interactions between these random effects and fixed effects. Even though they make interpretations problematic, as a block × treatment interaction may, a problematic interpretation is less serious than falsely concluding a significant treatment effect.

We found that the omission of a fixed covariate effect had little influence on testing another fixed effect (orthogonal to the covariate), even though this missing term also results in over-dispersion. The different results we found are due to how uncertainty about fixed effects is impacted by random effects versus by other fixed effects. Recall that the fixed and random effects were approximately the same in effect size. If we want to compare the effect of dropping a random effect on power, we start with the model that includes the random effect. For this latter model, there is a large contribution to the uncertainty in a treatment mean due to the inclusion of the random effect; which is an often cited reason for considering an effect to be random (because one is sampling from a population of effects, so that additional sampling error due to blocks has to be incorporated into the uncertainty about the fixed treatment means). If one drops that random effect (or considers it a fixed effect), the estimated variance of the mean decreases, which leads to excessively liberal *F*-tests. However, including or not including a fixed effect in the model has much less effect on the uncertainty of the mean of another orthogonal fixed effect, so we saw small or no changes in power in those simulations. Another way of thinking about it is that the standard error of a fixed effect mean is influenced more by a random effect than by another fixed effect of the same magnitude.

The effect of reducing the number of blocks, even if total *N* is unchanged, can have a dramatic effect on power since the additional uncertainty in estimating random effects inflates the standard errors of the fixed effects. While this decrease in power should be familiar to statisticians and can be determined analytically, many or most agricultural researchers are not aware that the number of blocks affects power. For determining sample sizes for real experiments it is probably easier to run simulations with the design options and treatment differences of interest to see how power is affected. Researchers pay a high price for having few blocks. Our simulations results are consistent with theory: it is better to have more blocks with fewer observations per block.

Along with better defaults (or perhaps situation dependent defaults), new users to PROC GLIMMIX would benefit from improved documentation that would steer them away from model/option combinations that make little sense (e.g., including the 'random _residual_;' statement when the conditional distribution is normal) and give warnings when default options are likely to create problems (as we found for the pconv default for the Poisson distribution). As this kind of software becomes easier to use (which is good), it also creates the situation where naïve users will simply plunge into estimating models without wading through the lengthy documentation and without having had training on the procedure. As we have demonstrated, this procedure often yields problematic analyses using default settings. We believe an appropriate analysis can only be obtained if one has both the background and the knowledge of the underlying statistics and the complications unique to GLMM model estimation, as well as training in PROC GLIMMIX to learn how to specify models and options. Moving from PROC MIXED to PROC GLIMMIX is not as straightforward as it might seem, especially in one-parameter cases where the familiar concept of a "residual" in linear models (and MS (error) and normal residual diagnostics) is no longer relevant, and the conceptual leap needed is not trivial.

5. Summary

A simulation study was conducted to determine how well a commonly used statistical software procedure for fitting generalized linear mixed models (GLMMs), SAS PROC GLIMMIX, performed for a simple GLMM. Data were generated from a wide variety of distributions for the same model under several conditions. Then, the generated data sets were analyzed by using the correct model (the generating model and estimating model were the same) and, subsequently, by misspecifying the estimating model. PROC GLIMMIX default options were used in all cases. We simulated data from a complete block design, including a block × treatment interaction, and, for some sets, a covariate. The effect sizes of the fixed and random effects were approximately the same, and model parameters and sample sizes were adjusted to yield 80% power for the *F*-test on treatment means for a 30 block experiment.

Convergence rates were low for the exponential and Poisson distributions, even when the generating and estimating models matched. The normal and lognormal distributions converged 100% of the time, but convergence rates for other distributions varied, affected by whether and how the models were misspecified. The number of blocks had a large effect on power, reducing the number of blocks from 30 to five (with the same total *N*) reduced power to 40% or less. Omitting the block × treatment random effect in the estimating model made *F*-tests too liberal. This was most obvious when treatment means were set equal to estimate Type I error rates, some of which increased to about 70% for the five-block condition. Since omitting a term makes the data over-dispersed relative to the estimating model, several potential remedies were investigated. For all distributions, we used the appropriate variance stabilizing transformation and fit the transformed data using a linear mixed model. For one-parameter members of the exponential family we included an over-dispersion parameter in the estimating model. We also tried changing the estimating model distribution to the negative binomial. None of these remedial steps fixed this problem, though using a variance stabilizing transformation did improve convergence rates.

Fitting a GLMM can be difficult. Although current software such as SAS PROC GLIMMIX is available and offers tremendous flexibility to fit a wide range of statistical models from a wide range of response types, it requires knowledge of experimental design, mathematical statistics and numerical methods, and model diagnostics. Our project started with the intent of focusing on model diagnostics to help a naïve user, but we quickly learned that this was precluded by the complications involved in successfully fitting a GLMM, i.e., a naïve user would run into trouble using the default options, with model diagnostics taking a back seat. A full evaluation of SAS's GLMM fitting software would require better planning of the simulations, choosing more realistic scenarios, and including a much wider sampling of models and parameters than used in our study.

Acknowledgements

This paper was much improved by comments from an anonymous reviewer and by the extensive suggestions provided by Walt Stroup, not all of which we were able to incorporate.

References

Faraway, J. J. 2006. Extending the Linear Model with R: Generalized Linear, Mixed Effects and Nonparametric Regression Models. Chapman and Hall. New York.

Littell, R.C., Milliken, G.A., Stroup, W.W., Wolfinger, R.D., Schabenberger, O. 2006. SAS for mixed models, second edition. Cary, NC. SAS Institute Inc.

McCullagh, P. and Nelder, J.A. 1989. Generalized linear models, second edition: Monographs on statistics and applied probability, 37. Chapman and Hall. New York, NY.

Mead, R. 1988. The design of experiments: Statistical principles for practical applications. Cambridge Univ. Press.

Novikov, I. 2003. A remark on efficient simulations in SAS. The Statistician **52** (Part1): 83-86

SAS Institute Inc. 2010. *SAS/STAT® 9.22 User's Guide* (PROC GLIMMIX). Cary, NC. SAS Institute Inc.

Wedderburn, R.W.M. 1974. Quasi-likelihood functions, generalised linear models, and the Gauss-Newton method. Biometrika **61**: 439-447.

Wilk, M.B. (1955). The randomization analysis of a generalized randomized block design. Biometrika **42**, 70-79.

Young, L.J., Campbell, N.L., Capuano, G.A. 1999. Analysis of overdispersed count data from single-factor experiments: A comparative study. J. of Agric., Biol., and Envir. Stat. **4**, 258-275.

Figures

Figure 1. Change in power when the number of blocks is decreased. Condition 1 is a randomized complete block design with 30 blocks, Condition 2 has the same number of replicates per block but five blocks (so 1/6 of total *N*), Condition 3 has five blocks but with total *N* the same as Condition 1.

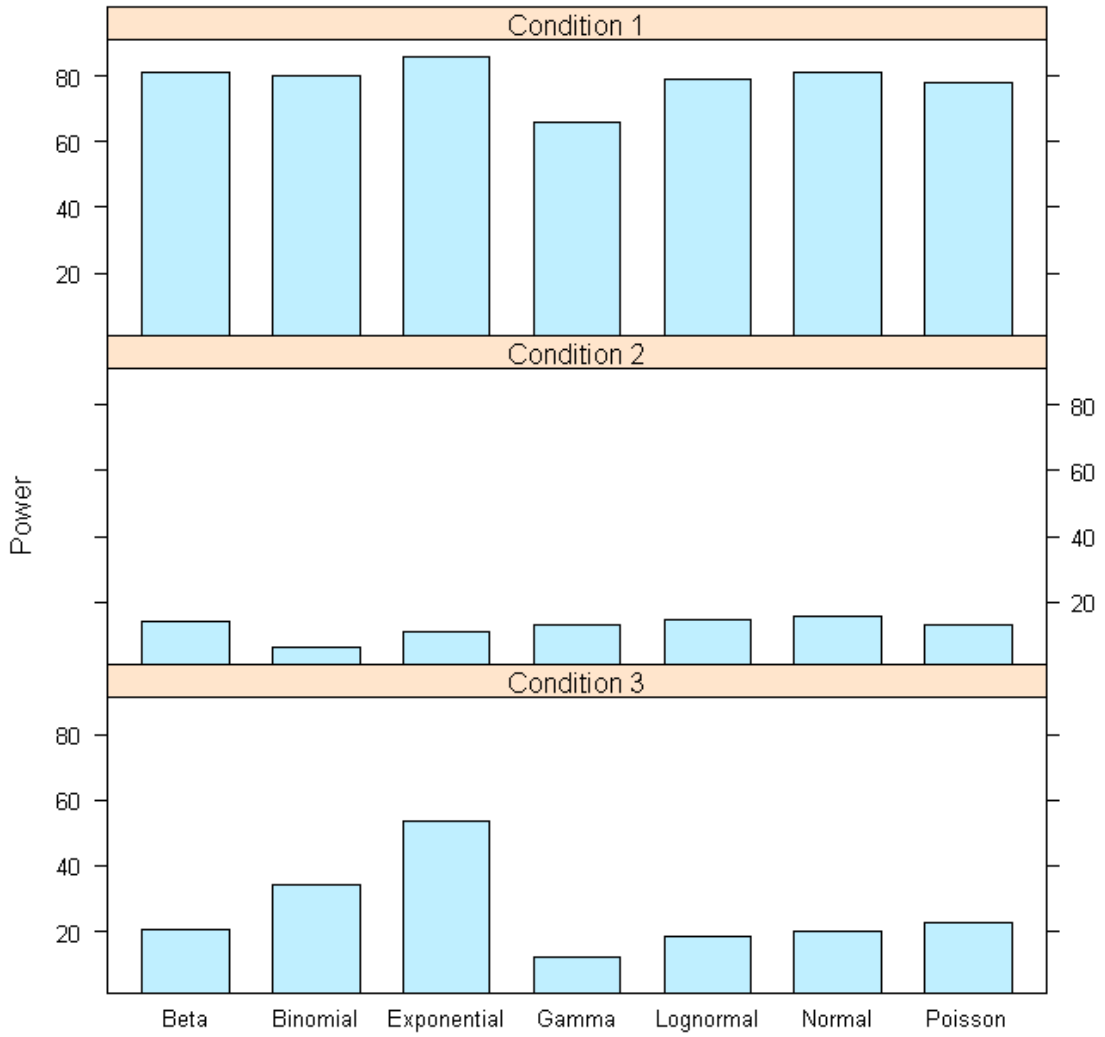


Figure 2. Convergence rates (light blue) and power (red) for the simulations using the Poisson distribution for default and LaPlace estimation options; and parameters (× 10) for the block variance (green) and the block × treatment interaction variance (dark blue) for the true, and the average estimates using the default and LaPlace options.

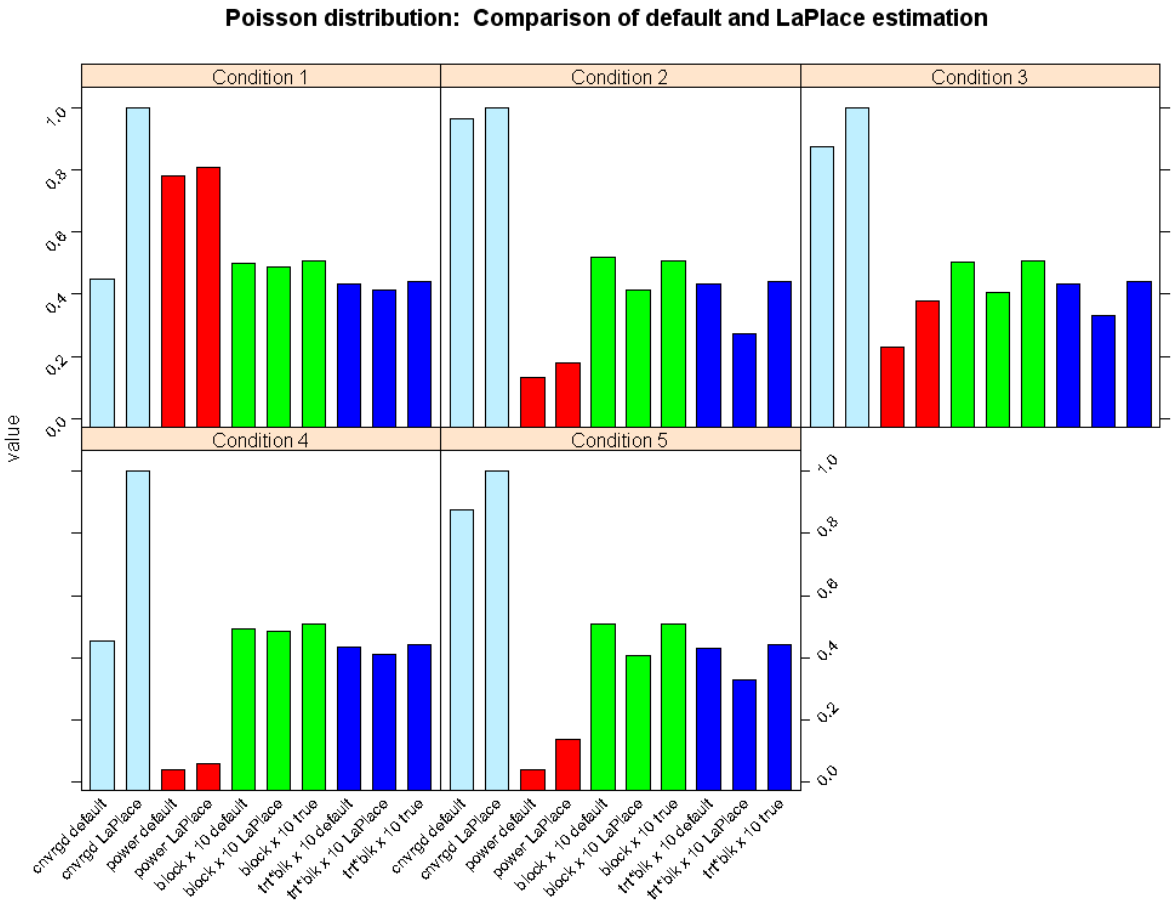
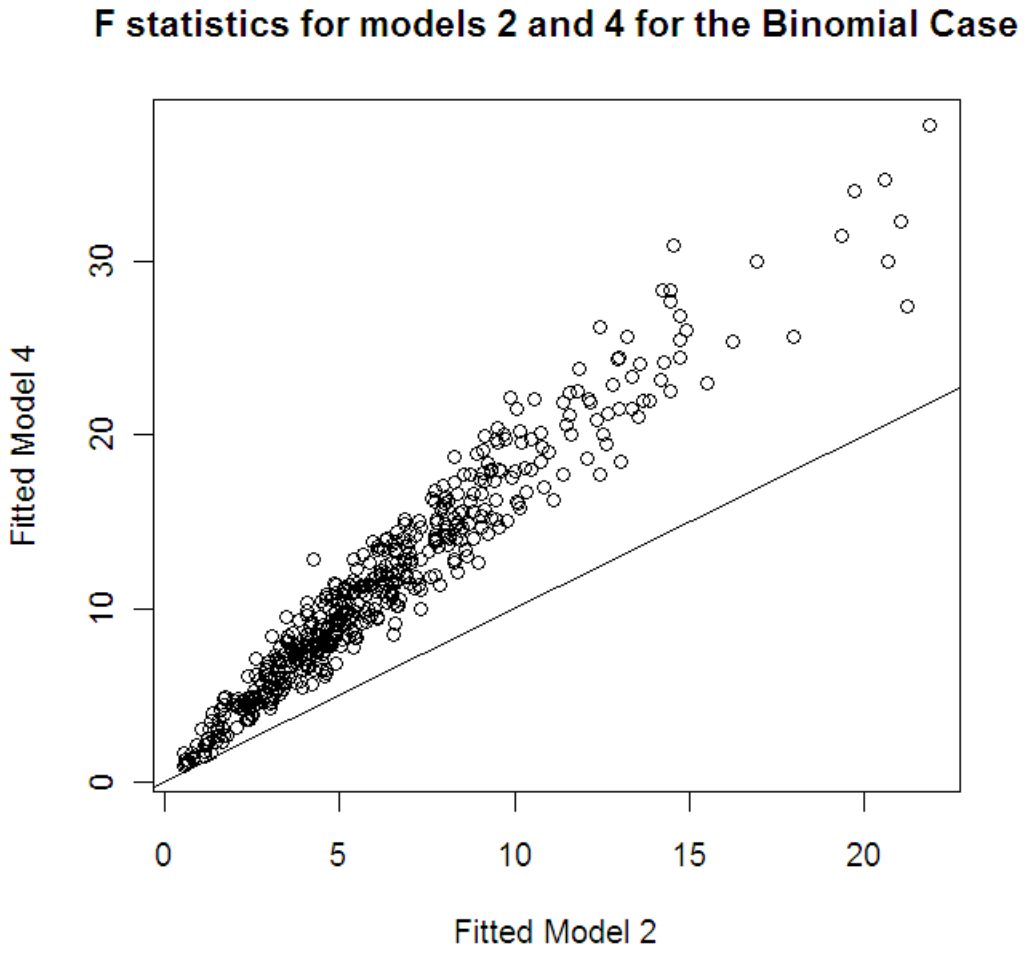


Figure 3. Comparison of *F* statistics for the binomial distribution when fitting Model 2 (generating and estimating models match) and Model 4 (estimating model missing block-by-treatment interaction, over-dispersion parameter added).



Tables (on following pages)

Table 1. Statistical distributions and parameters (in link-scale) examined in simulations[a]

Generating Distribution	Link Function	Treatment Means	σ^2_{Block}	$\sigma^2_{Trt*Block}$	β (SD)[b]	Additional Parameters	n[c]	Variance Stabilizing Transformation
Beta (μ,Φ)	logit	-1.551, -1.386, -1.237	0.16	0.09	-0.5 (0.16)	Φ = a+b = 63	2	arcsine ($Y^{1/2}$)
Binomial (μ)	logit	-0.44, 0.0, 0.44	1.0	0.25	1.0 (0.33)		2, 11[d]	arcsine ($Y^{1/2}$)
Exponential (μ)	log	3.5, 3.75, 4.0	0.01	0.01	1.0 (0.2)		3	log(Y)
Gamma (μ,Φ)	log	4.5, 5, 5.5	1.90	1.90	1.0 (0.80)	Φ = 29.68	3	log(Y)
Lognormal (μ,Φ)	identity	2.25, 2.5, 2.75	0.5	0.5	0.6 (0.25)	σ^2_{error} = 0.5	2	log(Y)
Normal (μ,Φ)	identity	59, 60, 61	6.25	3.8025	2.0 (0.25)	σ^2_{error} = 3.8025	2	
Poisson (μ)	log	3.5,4.0,4.5	0.0506	0.0441	1.0 (0.2)		5	log(Y)

[a] Shaded area indicates that the parameter or transformation is not applicable for this statistical distribution.
[b] Slope of covariate (SD = standard deviation).
[c] Number of replicates.
[d]Two replicates, each a draw from a binomial distribution with n = 11, same *p*.

Table 2. Conditions examined in simulations

Condition	Blocks	Replicates (n ≥ 2)	Total N	Means	Estimating Distribution	Estimating Response
1	30	n	N	$\mu_1 \neq \mu_2 \neq \mu_3$	True	*Y*
2	5	n	⅙N	$\mu_1 \neq \mu_2 \neq \mu_3$	True	*Y*
3	5	6n	N	$\mu_1 \neq \mu_2 \neq \mu_3$	True	*Y*
4	30	n	N	$\mu_1 = \mu_2 = \mu_3$	True	*Y*
5	5	6n	N	$\mu_1 = \mu_2 = \mu_3$	True	*Y*
6	30	n	N	$\mu_1 \neq \mu_2 \neq \mu_3$	Normal	*Y*
7	30	n	N	$\mu_1 \neq \mu_2 \neq \mu_3$	Normal	Transform(*Y*)
8	30	n	N	$\mu_1 = \mu_2 = \mu_3$	Normal	Transform(*Y*)
9	30	n	N	$\mu_1 \neq \mu_2 \neq \mu_3$	Negative Binomial	*Y*

Table 3. Models examined in PROC GLIMMIX

Fixed Effects in Generating Model	Model	Representation of Estimated Model	Glimmix Random Statements
Treatment	1	$f(y) = \mu + \tau_i + B_j$	Random Block;
	2[a]	$f(y) = \mu + \tau_i + B_j + \tau B_{(ij)}$	Random Block Trt*Block;
	3	$f(y) = \mu + \tau_i + B_j + \tau B_{(ij)}$ (+ Over-dispersion[b])	Random Block Trt*Block; Random _residual_;
	4	$f(y) = \mu + \tau_i + B_j$ (+ Over-dispersion[b])	Random Block; Random _residual_;
Treatment + Covariate	5[a]	$f(y) = \mu + \tau_i + \beta \cdot x_1 + B_j + \tau B_{(ij)}$	Random Block Trt*Block;
	6[c]	$f(y) = \mu + \tau_i + B_j + \tau B_{(ij)}$	Random Block Trt*Block;
	7[c]	$f(y) = \mu + \tau_i + B_j + \tau B_{(ij)}$ (+ Over-dispersion[b])	Random Block Trt*Block; Random _residual_;

[a] True model used to generate simulated data; τ_i is the ith treatment effect and B_j is the jth block effect. All generating models included the same block and block × treatment interaction random effects.
[b] Estimating an over-dispersion parameter is appropriate for 1-parameter distributions only.
[c] Different from Model 2 in that data were generated with covariate but model was misspecified by omitting the covariate effect.

Table 4. Impact of models and their predictable consequences for one-parameter distributions (Binomial, Exponential, and Poisson) and two-parameter distributions (Beta, Gamma, Log-normal, and Normal).

Model	What it Does	Predictable Consequences
1	Leaves out $Blk \times Trt$	Mean estimate (μ) may be biased since $E(X) \neq \mu$ Over-dispersion symptoms: inflated Type I error rate, inadequate confidence interval (CI) coverage (due to underestimates of standard errors)
2	Consistent with generating data (true model)	Population average (mean) estimate is unbiased, i.e. $E(X) = \mu$ Over-dispersion symptoms: none expected
3	Includes both $Blk \times Trt$ and scale parameter ϕ	Redundant since no within-plot (subsampling) variation exists when data are generated from simulations and residual variance is unchanged. Scale parameter $\phi \cong 1$ so $\phi \cdot$ (variance) would be unchanged. There is little difference between Models 2 and 3.
4	Leaves out $Blk \times Trt$ and adds a scale parameter ϕ	Mean estimate (μ) may be biased, $E(X) \neq \mu$ Scale parameter known to be an inadequate "fix" for over-dispersion, with symptoms: inflated Type I error rate, inadequate CI coverage (although not expected to be as severe as Model 1 since standard error underestimate should not be as bad)
5	Consistent with new generating data that includes an added fixed covariate	Population average (mean) estimate is unbiased, i.e. $E(X) = \mu$ Over-dispersion symptoms: none expected
6	Leaves out covariate fixed effect	As long as the missing fixed covariate effect is orthogonal to the other linear predictors, then the mean and variance component estimates are little affected when the covariate is omitted.
7	Leaves out covariate fixed effect and adds a scale parameter ϕ	Like Model 6, omitting the orthogonally generated covariate has little effect on the mean and variance component estimates. Adding a scale parameter as an over-dispersion "fix" is also ineffective in picking up information from the missing covariate since we expect $\phi \cong 1$ so $\phi \cdot$ (variance) would be unchanged.

Table 5. Rates of convergence obtained for various sample allocations where $\mu_1 \neq \mu_2 \neq \mu_3$ and the distribution used to generate the data was specified as the estimating distribution. Simulations were designed such that power ≈ 0.8 for 30 blocks.[a,b]

Conditions Examined in Simulations				Model	Beta (μ,Φ)	Binomial (μ)	Exponential (μ)	Gamma (μ,Φ)	Lognormal (μ,Φ)	Normal (μ,Φ)	Poisson (μ)
Condition[c]	Blocks	Reps	Fixed Effects								
1	30	n	Treatment	1	85.3	97.2	39.9	0.02	100	100	94.2
				2[d]	83.1	99.7	26.6	99.9	100	100	45.1
				3		96.3	50.1				51.6
				4		96.5	81.8				91.9
			Treatment + Covariate	5[d]	81.0	99.8	25.4	99.9	100	100	44.7
				6	79.5	99.8	16.4	77.0	100	100	44.7
				7		96.6	48.1				49.7
2	5	n	Treatment	1	91.5	99.9	67.3	7.0	100	100	95.3
				2[d]	81.4	99.9	48.5	99.4	100	100	87.5
				3		99.0	68.3				87.8
				4		99.2	85.5				98.0
3	5	6n	Treatment	1	92.0	99.6	76.0	4.4	100	100	98.6
				2[d]	87.0	97.6	53.4	99.7	100	100	96.4
				3		89.4	71.2				96.6
				4		99.1	80.9				97.9

[a] Shaded area indicates that the model is not applicable for this statistical distribution.
[b] The true power is given in Table 7 (Condition 1:Model 2 or Model 5,when the estimating distribution was the generating distribution).
[c] See Table 2 for complete descriptions of simulation conditions.
[d] True model used to generate simulated data (see Table 3).

Table 6. Rates of convergence obtained when the generating distribution, the normal distribution or the negative binomial distribution are specified as the estimating distribution where $\mu_1 \neq \mu_2 \neq \mu_3$ and the number of blocks = 30.[a]

Estimating Distribution, Response[b]	Condition[c]	Fixed Effects	Model	Generating Distribution's Convergence Rate (%)						
				Beta (μ,Φ)	Binomial (μ)	Exponential (μ)	Gamma (μ,Φ)	Lognormal (μ,Φ)	Normal (μ,Φ)	Poisson (μ)
Generating, *Y*	1	Treatment	1	85.3	97.2	39.9	0.02	100	100	94.2
			2[d]	83.1	99.7	26.6	99.9	100	100	45.1
			3		96.3	50.1				51.6
			4		96.5	81.8				91.9
		Treatment + Covariate	5[d]	81.0	99.8	25.4	78.6	100	100	44.7
			6	79.5	99.8	16.4	77.0	100	100	44.7
			7		96.6	48.1				49.7
Normal, *Y*	6	Treatment	1	100	100	100	100	100		100
			2[d]	100	100	99.9	100	100		100
Normal, g(*Y*)	7	Treatment	1	100	100	100	100	100		100
			2[d]	100	100	99.9	100	100		100
Negative Binomial, *Y*	9	Treatment	1		92.6	44.5				85.9
			2[d]		92.6	25.0				61.9

[a] Shaded area indicates that the model is not applicable for this statistical distribution.
[b] g(*Y*) is the transformed value of the response. See Table 1 for the variance stabilizing transformation used for each distribution.
[c] See Table 2 for complete descriptions of simulation conditions.
[d] True model used to generate simulated data (see Table 3).

Table 7. Power of treatment comparisons obtained when the generating distribution, the normal distribution or the negative binomial distribution are specified as the estimating distribution where $\mu_1 \neq \mu_2 \neq \mu_3$ and the number of blocks = 30.[a]

Estimating Distribution, Response[b]	Condition[c]	Fixed Effects	Model	Generating Distribution's Power (%)						
				Beta (μ,Φ)	Binomial (μ)	Exponential[d] (μ)	Gamma (μ,Φ)	Lognormal (μ,Φ)	Normal (μ,Φ)	Poisson (μ)
Generating, *Y*	1	Treatment	1	93.4	87.9	86.3	—[e]	92.9	93.4	94.9
			2[f]	81.0	80.3	85.6	77.1	79.2	81.1	78.0
			3		80.7	86.3				78.4
			4		87.4	81.1				93.8
		Treatment + Covariate	5[f]	81.0	80.3	89.4	77.1	79.1	80.7	77.7
			6	79.0	78.2	84.0	71.7	78.1	80.2	77.1
			7		78.3	82.0				78.9
Normal, *Y*	6	Treatment	1	93.5	87.4	84.3	81.8	80.2		93.7
			2[f]	80.8	81.2	81.0	24.4	59.6		79.6
Normal, g(*Y*)	7	Treatment	1	93.1	86.5	64.8	97.2	92.9		92.2
			2[f]	81.3	80.3	60.8	77.2	79.2		78.3
Negative Binomial, *Y*	9	Treatment	1		48.0	86.3				93.8
			2[f]		46.3	85.8				79.2

[a] Shaded area indicates that the model is not applicable for this statistical distribution.
[b] g(*Y*) is the transformed value of the response. See Table 1 for the variance stabilizing transformation used for each distribution.
[c] See Table 2 for complete descriptions of simulation conditions
[d] Because not all simulations converged, our definition of power is based on only those data sets where the model converged.
[e] Convergence rate < 1% therefore power was not calculated.
[f] True model used to generate simulated data (see Table 3)

Table 8. Type I error for treatment comparisons (as a percentage) obtained when the response, *Y*, was modeled using the generating distribution, as Normally distributed data, or was transformed and modeled using the Normal distribution. Treatment was the only fixed effect included in the generating model.[a]

Estimating Distribution, Response[b]	Condition[c]	Blocks	Model	Generating Distribution's Type I Error Rate (%)						
				Beta (μ,Φ)	Binomial (μ)	Exponential (μ)	Gamma (μ,Φ)	Lognormal (μ,Φ)	Normal (μ,Φ)	Poisson (μ)
Generating, *Y*	4	30	1	17.3	8.0	6.3	—[d]	16.4	16.8	22.0
			2	5.4	4.6	5.6	9.8	5.0	4.8	4.0
	5	5	1	69.4	34.0	7.8	98.6	69.0	69.7	63.7
			2	5.4	4.3	2.0	10.2	5.0	5.5	4.1
Normal, g(*Y*)	8	30	1	16.0	8.0	5.1	53.5	16.4		15.9
			2	5.6	5.0	4.1	9.7	5.0		4.9

[a] Shaded area indicates that the model is not applicable for this statistical distribution.
[b] g(*Y*) is the transformed value of the response. See Table 1 for the variance stabilizing transformation used for each distribution.
[c] N = 90*n* (3 treatments × 30 blocks × *n* replicates) for all conditions (See Table 2).
[d] Convergence rate < 1% therefore Type 1 error was not calculated