Monte Carlo Likelihood Approximation for Generalized Linear Mixed Models

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Christina Knudson

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Charles Geyer and
Galin Jones, Advisers

January 2016

Contents

Chapter 1

Introduction

The generalized linear mixed model (GLMM) is an extension of both the generalized linear model and the linear mixed model; the model incorporates fixed and random effects as well as a response from an exponential family. GLMMs were first discussed by Stiratelli et al. (1984) and are now used in a variety of disciplines. Despite their widespread use, traditional methods of frequentist likelihood-based inference are not generally available. The challenge lies in the likelihood function for GLMMs: because the likelihood cannot depend on the random effects, the likelihood is an integral that is often intractable (details in Section 1.2). Due to this challenge, most methodology and software for GLMMs perform little more than maximum likelihood (details in Section 2.3) or do not perform likelihood-based inference at all (details in Section 2.2). Our goal is to enable all types of frequentist likelihood-based inference. This includes (but is not limited to) calculating Fisher information, performing hypothesis tests, and constructing confidence intervals. To perform all types of frequentist likelihood-based inference, the entire likelihood function is necessary.

With this goal in mind, we have created an R package `glmm` that approximates the entire likelihood function for a GLMM with either a Bernoulli or Poisson response. This package uses the method of Monte Carlo likelihood approximation (MCLA) (Geyer, 1994; Geyer and Thompson, 1992; Sung and Geyer, 2007). This method

relies on the choice of an importance sampling distribution, a distribution from which simulated random effects are drawn. These generated random effects are then used to approximate the entire likelihood function. Because MCLA approximates the entire likelihood, this procedure enables every type of likelihood-based inference, not solely maximum likelihood. For more details on MCLA, see Section 2.4.

Before the release of R package `glmm`, no publicly-available software implementing MCLA produced accurate maximum likelihood estimates for models of non-trivial complexity. For example, the `bernor` package (Sung and Geyer, 2007) implements MCLA but cannot accurately perform maximum likelihood for the benchmark data set from a salamander mating experiment (described in Section 1.3). The absence of MCLA software highlights a challenge from which the procedure suffers: finding an importance sampling distribution that works well in practice.

We propose an importance sampling distribution to be used in implementing MCLA for GLMMs (Equation (4.1)); establish its theoretical validity (Section 4.2); implement it in R package `glmm` (Knudson, 2015); and demonstrate how to use the package to perform maximum likelihood, test hypotheses, and calculate confidence intervals (Chapter 5). Before discussing these new developments, we review background information on GLMMs, the likelihood of GLMMs, and the benchmark salamander data set. Additionally, Chapter 2 summarizes a few methods of inference for GLMMs.

1.1 Generalized Linear Mixed Models

Let $Y \in \mathbb{R}^n$ be a response vector. Let X be a design matrix for observed predictors and let $\beta \in \mathbb{R}^p$ be its coefficient vector. Let Z be the model matrix for the random effects and let $U \in \mathbb{R}^q$ be a vector of unobservable random effects. Let $\nu = (\nu_1, \ldots, \nu_K)^T$ be a vector of variance components such that each component is nonnegative. Let D be a variance matrix dependent on ν. Although D is a function

of ν, we suppress the parameter for cleaner notation. The general description of a GLMM does not restrict the form of variance matrix D, but this thesis focuses on the case in which D is diagonal; that is, we consider the setting in which the random effects are independent of one another. We assume the random effects follow a multivariate normal distribution centered at 0 with variance matrix D, and we denote this distribution's density by $f_\nu(u)$. We also assume the distribution of the response vector given the random effects has density $f_\beta(y|u)$. If we let $\theta = (\beta^T, \nu^T)^T$, we can express the joint density of the random effects vector and the response vector as

$$f_\theta(u, y) = f_\beta(y|u)\, f_\nu(u). \tag{1.1}$$

Define random vectors η and μ such that $\eta = X\beta + Zu$ and $\mu = E(Y|U = u)$. Let $g(\cdot)$ be a monotone, differentiable function so that $g(\mu) = \eta$. This function is called the "link" function. Though η and μ could be linked through many functions, we focus on the canonical link function and the case in which η is the canonical random vector. In particular, the canonical random vector for the Bernoulli distribution is the log odds of succss and the canonical link function is the logit function. The canonical random vector for the Poisson distribution is the log of the expected response and the canonical link is the natural logarithm.

1.2 The Likelihood Function

The likelihood is a function of the parameter given the data. Constants with respect to the parameter can be dropped from the likelihood. Random effects are part of the model but cannot be part of the likelihood because they are not data. Therefore, the

likelihood for a GLMM is

$$L(\,\theta\,|\,y\,) = \int f_\beta(y|u)\, f_\nu(u)\, du = \int f_\theta(u, y)\, du \tag{1.2}$$

and the log likelihood is $l(\theta|y) = \log L(\theta|y)$. To be explicit, the integral is evaluated over $\mathbb{R}^q$. Often, this integral cannot be expressed in closed form.

We now discuss how the random effect structure relates to the integral. This informs the discussion in Chapter 2 on whether inferential methods are appropriate for a certain model. We will see that in special cases (described in Section 2.1), numerical integration can produce the likelihood and enable likelihood-based inference. If numerical integration is not possible, researchers must either resort to methods that perform specific types of inference (such as Monte Carlo EM, discussed in Section 2.3, which performs maximum likelihood) or methods that approximate the likelihood (such as PQL and MCLA, discussed in Section 2.2 and Section 2.4).

We start with the simplest random effects structure: each component of the response vector depends on a single random effect. Then the likelihood can be factored into a product of one-dimensional integrals. As an example of a likelihood that can be factored into a product of one-dimensional integrals, consider the model created by McCulloch (1997) with data simulated by Booth and Hobert (1999). Each component of the binary response vector depends on a single fixed effect predictor and a single random effect.

When each component of the response vector depends on more than one random effect, we need to define "crossed" random effects. First, recall that the model has K variance components. Construct a graph with a node for every component of the response vector and a node for every random effect, and connect each component of the response vector to the random effects it depends on. If each maximal connected subgraph contains multiple random effects from a variance component, then we say

the random effects are crossed. An example of a GLMM with crossed random effects is described in Section 1.3. A graph for this example would have six maximal connected subgraphs, and each maximal connected subgraph would have 20 random effect nodes.

The structure of the crossed random effects determines how to factor the integral in Equation (1.2). The likelihood can be factored into a product of integrals with dimension determined by the number of random effect nodes in the maximal connected subgraph. The salamander likelihood in Section 1.3 can be factored in a product of six 20-dimensional integrals but no further.

1.3 The Salamander Data and Model

Researchers at the University of Chicago conducted an experiment on a single species of salamanders in 1986. McCullagh and Nelder (1989, Section 14.5) presented the experiment and data, and Karim and Zeger (1992) proposed a model they call "Model A." The salamander data and model have become a benchmark in the GLMM universe; the data have been modeled by many researchers including Booth and Hobert (1999), Breslow and Clayton (1993), Karim and Zeger (1992), McCullagh and Nelder (1989), Schall (1991), Sung and Geyer (2007), and Wolfinger and O'Connell (1993), .

1.3.1 The Salamander Data

Before the experiment began, female salamanders and male salamanders of the same species were collected from two locations. The salamanders were categorized into populations named "Rough Butt" and "White Side" based on their location of origin. The scientific goal was to determine whether salamanders were more likely to mate with those from their own population or whether they were just as likely to mate with salamanders from either population. More specifically, scientsts sought to compare the odds of mating for each type of cross: female Rough Butt salamanders and male

Rough Butt salamanders (denoted RR), female Rough Butt salamanders and male White Side salamanders (denoted RW), female White Side salamanders and male Rough Butt salamanders (denoted WR), and female White Side salamanders and male White Side salamanders (denoted WW).

Scientists ran an experiment three times. Each experiment consisted of trials conducted on two closed groups of 20 salamanders. That is, each experiment was conducted on 40 salamanders that were split into two closed groups. Each group contained 5 female Rough Butts, 5 female White Sides, 5 male Rough Butts, and 5 male White Sides. Each trial consisted of placing a female salamander and a male salamander in an isolated space together, then observing the binary response of interest: whether the salamanders mated. Each female salamander participated in six trials with male salamanders from her closed group: three trials with male White Side salamanders and three trials with male Rough Butt salamanders. Scientists paired salamanders from the same group; inter-group trials were not conducted. Thus, 60 trials were conducted on each closed group, each experiment consisted of 120 trials, and the overall dataset contains binary responses from 360 trials.

1.3.2 The Salamander Model

To model these data with a GLMM, Karim and Zeger's (1992) "Model A" proposes a random effect for each female salamander; a random effect for each male salamander; a fixed effect predictor for each of the four types of cross; and a Bernoulli response of whether the pair of salamanders mated, dependent on the type of cross, the female random effect, and the male random effect. Each salamander's random effect is assumed to be independent of the others. The male salamanders' random effects share one variance component while the female salamanders' random effects share another variance component. These two variance components are later referred to as ν_M and ν_F. The males' random effects are crossed with the females' random

effects, where "crossed" is defined in Section 2.1. Though the same salamanders were used in the first two experiments, the data are traditionally modeled assuming that new salamanders were used in each experiment (Booth and Hobert, 1999; Karim and Zeger, 1992; McCullagh and Nelder, 1989).

1.3.3 The Salamander Model's Likelihood

Because the data set consists of 360 trials with four crosses, the model matrix for the fixed effects X is 360×4. Because the experiment involved 120 salamanders, the model matrix for the random effects Z is 360×120. Let $x_{i\cdot}$ denote the ith row of X and let $z_{i\cdot}$ denote the ith row of Z. Let $\beta = (\beta_{RR}, \beta_{RW}, \beta_{WR}, \beta_{WW})^T$ denote the fixed effects vector, where β_{RW} denotes the log odds of a female Rough Butt salamander mating with a male White Side salamander. Because the elements of the response vector are conditionally Bernoulli given the random effects,

$$f_\beta(y|u) = \exp\left(y^T (X\beta + Zu) - \sum_{i=1}^{n} \log\left(1 + \exp\left(x_{i\cdot}\beta + z_{i\cdot}u\right)\right)\right).$$

The females' random effects are assumed to be independent, identically distributed draws from a normal distribution with mean 0 and unknown variance ν_F. Similarly, the males' random effects are assumed to be independent, identically distributed draws from a normal distribution with mean 0 and unknown variance ν_M. Let u^F and u^M denote the vectors of females' random effects and males' random effects, respectively. Each vector is of length 60. Then

$$f_\nu(u) = \exp\left(-60\log(2\pi) - 30\log\nu_F - 30\log\nu_M - \frac{(u^F)^T u^F}{2\nu_F} - \frac{(u^M)^T u^M}{2\nu_M}\right).$$

Then the likelihood is

$$\int \exp\left(y^T (X\beta + Zu) - \sum_{i=1}^{n} \log\left(1 + \exp\left(x_{i\cdot}\beta + z_{i\cdot}u\right)\right) - 60\log(2\pi) + \right.$$
$$\left. -30\log\nu_F - 30\log\nu_M - \frac{(u^F)^T u^F}{2\nu_F} - \frac{(u^M)^T u^M}{2\nu_M} \right) du.$$

This integral is of dimension 120, the number of salamanders in the experiment. However, there were six closed groups with 20 salamanders per group, so the likelihood factors into a product of six 20-dimensional integrals. Since the random effects from within a closed group of 20 salamanders are crossed, we cannot factor the 20-dimensional integrals into integrals of any smaller dimension. These integrals are too high-dimensional for numerical integration, a method described in Section 2.1.

Because numerical integration cannot produce the likelihood, this has become a benchmark data set; researchers have analyzed this data using a variety of approaches. Karim and Zeger (1992) analyzed the data with a Bayesian approach relying on Markov chain Monte Carlo. Wolfinger and O'Connell (1993) performed inference based on a pseudo-likelihood. Booth and Hobert (1999) were able to perform maximum likelihood using Monte Carlo EM (a method described in Section 2.3) and calculate standard errors for the maximum likelihood estimates using an additional method (Louis, 1982). More recently, Sung (2003) and Sung and Geyer (2007) implemented MCLA with an importance sampling distribution chosen independently of the observed data, but they were not able to find MLEs for this model. Penalized quasi-likelihood (a method discussed in Section 2.2) can approximate the likelihood, but the approximation does not converge to the likelihood. Before the advent of the R package `glmm`, no researcher or software has constructed an asymptotically-valid approximation to the entire likelihood function for this benchmark data set. The `glmm` analysis of this model is found in Section 5.2.

Chapter 2

Methods of Likelihood-Based Inference

In this section, we explore different methods of performing frequentist likelihood-based inference for GLMMs. Some methods – such as PQL and MCLA – approximate the likelihood function; other methods – such as Monte Carlo EM – have the specific goal of maximum likelihood. Some methods – such as numerical integration, Monte Carlo EM, and MCLA – are able to perform maximum likelihood while other methods – such as penalized quasi-likelihood – produce point estimates that are not MLEs.

2.1 Numerical Integration

Numerical integration produces an approximation to the likelihood. The approximation contains deterministic (rather than Monte Carlo) error. As a result, maximizing the likelihood approximation produces estimates of MLEs. Because numerical integration produces an approximation to the entire likelihood, it enables all types of frequentist likelihood-based inference.

The limitation of numerical integration is that it is viable for low-dimensional integrals only. For example, numerical integration can be used either if each component of the response depends on a single random effect or if the random effects are not crossed

(since then the multi-dimensional integral in Equation (1.2) can be split into a product of one-dimensional integrals). Numerical integration can run into difficulties with crossed random effects because the likelihood often cannot be factored into integrals of low enough dimension. For example, in the salamander model (Section 1.3), the likelihood is a product of six 20-dimensional integrals. Using numerical integration for a 20-dimensional integral is much more complicated and computationally-expensive than using numerical integration for a one-dimensional integral.

From a user's point of view, the salamander model does not seem complicated or contrived; after all, there are only two random effects per mating. We can imagine many real-world GLMM experiments with crossed random effects, and many of these would be too complicated for numerical integration. Because numerical integration is limited in this way, we do not consider it a competitor to MCLA.

2.2 Penalized Quasi-Likelihood

PQL is a method of approximating the likelihood for GLMMs and is usually associated with parameter estimation and random effect prediction (Breslow and Clayton, 1993). Notably, PQL does not produce MLEs; instead of maximizing the likelihood, it maximizes a function known as a penalized quasi-likelihood. Many variations of PQL exist, but most begin by first approximating the log likelihood with a second-order Taylor polynomial. Additional approximations further simplify the expression and yield the penalized quasi-likelihood.

PQL software includes the `GLIMMIX` procedure in SAS and the `glmmPQL` command in the `MASS` R library (Venables and Ripley, 2002). The `glmer` command in the `lme4` R library (Bates et al., 2014) performs PQL for models with multiple variance components (it performs numerical integration for problems with a single variance component).

PQL's popularity stems from its computational speed. However, the assumptions and approximations that provide its speed come at a price. PQL's assumptions and approximations lack theoretical grounding and have an "air of ad hocery," according to McCulloch and Searle (2001). Moreover, most PQL-based software is a black box that is difficult for users to understand. For example, fully understanding the `glmer` function in `lme4` would require reading the source code itself because the package documentation does not provide details on the approximations and assumptions. Finally, PQL estimates are biased when the elements of the response vector are binary (Breslow and Lin, 1995 and Lin and Breslow, 1996, McCulloch and Searle, 2001). Breslow (1993) defends PQL by emphasizing that it is meant for "approximate inference" and subsequently claims that PQL has been subject to "abuse and misinterpretation." However, McCulloch and Searle (2001) sees PQL's problems as so severe that they discourage the use of PQL.

While both MCLA and PQL approximate the likelihood, MCLA's approximation converges to the likelihood as the Monte Carlo sample size increases while PQL's approximation does not. Because our goal is to find a likelihood approximation that converges to the likelihood, we do not consider PQL a competitor to MCLA.

2.3 Monte Carlo EM

The EM algorithm was named by Dempster et al. (1977) but invented independently by several researchers. The EM algorithm performs maximum likelihood for models with missing data. Since random effects are missing data, GLMMs fit into the EM framework. Starting with an initial parameter value, EM iterates between two steps: the "E" step and the "M" step. The tth "E" step calculates

$$Q_t(\theta) = E_{\theta_t}\left[\log f_\theta(U, Y)|Y = y\right]$$

and the "M" step finds the parameter value that maximizes $Q_t(\theta)$.

Monte Carlo EM (MCEM), the Monte Carlo extension of the EM algorithm, was invented to calculate the expectation in $Q_t(\theta)$ using Monte Carlo (Wei and Tanner, 1990). Using the last iteration's parameter value, Monte Carlo generates random effects. These are used to approximate the expectation in the "E" step of ordinary EM. The "M" step remains unchanged from ordinary EM: $Q_t(\theta)$ is maximized. Since many Monte Carlo and Markov chain Monte Carlo methods exist, many implementations of MCEM have been proposed.

MCEM is limited to maximum likelihood; since MCEM only evaluates the likelihood at certain points but does not find the entire likelihood function, MCEM cannot be used for other likelihood-based inference. Moreover, MCEM does not produce Fisher information; it must be calculated separately using the Monte Carlo version of the method published by Louis (1982). Because our goal is to approximate the entire likelihood function in order to perform any type of frequentist likelihood-based inference, we do not consider Monte Carlo EM a competitor to MCLA.

2.4 Monte Carlo Likelihood Approximation

MCLA is a Monte Carlo method for approximating the entire likelihood function; it was first proposed by Geyer (1990) for models with unnormalized densities, then extended to models with normalized densities and random effects (Thompson and Guo, 1991), and then extended again to models with unnormalized densities and random effects (Gelfand and Carlin, 1993). We focus on MCLA for GLMMs with random effects.

MCLA is powerful because it enables any type of likelihood-based inference. Maximum likelihood is of special interest; the maximizer of the Monte Carlo likelihood approximation is called the Monte Carlo maximum likelihood estimate (MCMLE).

Because MCLA is a Monte Carlo method, the accuracy of the likelihood approximation and inferences based on the likelihood approximation can be improved by increasing the size of the Monte Carlo sample used to calculate the likelihood approximation.

The asymptotic properties of the likelihood approximation as the Monte Carlo sample size increases have been studied in generality (Geyer, 1994; Geyer and Thompson, 1992) and for the special case of an importance sampling distribution chosen independently of the data (Sung and Geyer, 2007). Geyer (1994) presents conditions under which the Monte Carlo likelihood approximation converges almost surely to the exact likelihood for any single parameter value. In addition to studying the pointwise convergence, we can also consider the convergence of the entire likelihood function. Geyer (1994) shows the Monte Carlo likelihood approximation converges almost surely to the likelihood function for GLMMs specified with unnormalized densities as long as a Wald-like integrability condition is met. Additionally, if the parameter space can be compactified, the MCMLE converges to the MLE almost surely (Geyer, 1994). Geyer (1994) also shows that the Monte Carlo profile likelihoods converge to the exact profile likelihoods almost surely, and no additional regularity conditions are required.

More recently, MCLA for GLMMs has been studied by Sung and Geyer (2007), who focus on an MCLA implementation with an importance sampling distribution constructed independently of the observed data. Under this framework, Sung and Geyer (2007) provide conditions under which the MCMLE is asymptotically normal and calculate MCMLE variance, taking into account both the usual sampling error (related to the observed sample size) and the Monte Carlo error (related to the Monte Carlo sample size). Sung and Geyer (2007) also produce an approximate distribution for the MLE, which can be used for the MCMLE when the Monte Carlo sample size is large. Additionally, Sung and Geyer (2007) discuss the role of the importance

sampling distribution in calculating the MCMLE variance. Since these theorems and calculations are only appropriate when the importance sampling distribution is chosen independently of the observed sample size, these topics would need to be revisited and updated for importance sampling distributions that depend on the observed data.

In addition to their theoretical work, Sung and Geyer (2007) prepared the R package `bernor` that fits maximum likelihood estimates for a GLMM with a Bernoulli response; their package uses an importance sampling distribution that does not depend on the observed data. Though their package can perform maximum likelihood for simpler models, it cannot find MLEs for a model as complicated as the salamander model (Section 1.3).

This highlights the challenge of finding an importance sampling distribution that performs well in practice. Though MCLA theory indicates any importance sampling distribution should suffice as long as its support contains the support of the target distribution, many importance sampling distributions require so much computing power that modern computers cannot perform maximum likelihood in practice. This practical problem is illustrated by the absence of MCLA software that can produce MLEs for nontrivial problems. In response, we present an importance sampling distribution (Equation (4.1)) and the R package `glmm`.

Chapter 3

GLMM and MCLA Calculations

In Section 3.1, we express the Monte Carlo log likelihood approximation and its first two derivatives for GLMMs. The second derivative (the Hessian matrix) is used for estimating Fisher information, as shown in Section 3.2.1. In addition to sampling error, Section 3.2 discusses Monte Carlo error for GLMM inference performed using MCLA. This chapter also details the distribution of the random effects (Section 3.3) and the distribution of the response vector conditional on the random effects (Section 3.4).

3.1 MCLA Calculations and Derivatives

The MCLA calculation requires the density for the joint distribution of the observed data and random effects, $f_\theta(u, y)$. Because this joint density can be expressed as the product shown in Equation (1.1), we require expressions for $\log f_\nu(u)$ and its derivatives (found in Section 3.3) and expressions for $\log f_\beta(y|u)$ and its derivatives (found in Section 3.4).

Let u_k, $k = 1, \ldots, m$, be vectors of length q drawn from a distribution with density $\tilde{f}(u_k)$ where $\tilde{f}(u_k)$ does not depend on θ. The specific importance sampling distribution used in R package `glmm` is described in Equation (4.1).

Then the Monte Carlo log likelihood approximation is

$$l_m(\theta|y) = \log\left(\frac{1}{m}\sum_{k=1}^{m}\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)}\right). \tag{3.1}$$

Again, this function can be used for any likelihood-based inference and its maximizer is the MCMLE.

Let ∇ represent differentiating with respect to θ and ∇^2 represent differentiating a second time. The chain rule provides a start to the gradient calculation:

$$\begin{aligned}\nabla l_m(\theta|y) &= \nabla\left[\log\left(\frac{1}{m}\sum_{k=1}^{m}\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)}\right)\right] \\ &= \frac{\nabla\left[\frac{1}{m}\sum_{k=1}^{m}\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)}\right]}{\frac{1}{m}\sum_{k=1}^{m}\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)}}.\end{aligned}$$

Because our choice of $\tilde{f}$ does not depend on θ,

$$\begin{aligned}\nabla l_m(\theta|y) &= \frac{\frac{1}{m}\sum_{k=1}^{m}\frac{\nabla f_\theta(u_k,y)}{\tilde{f}(u_k)}}{\frac{1}{m}\sum_{k=1}^{m}\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)}} \\ &= \frac{\sum_{k=1}^{m}\left[\nabla \log f_\theta(u_k,y)\right]\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)}}{\sum_{k=1}^{m}\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)}}.\end{aligned} \tag{3.2}$$

The Hessian of the MCLA is the derivative of the MCLA gradient:

$$\nabla^2 l_m(\theta|y) = \nabla\left[\frac{\sum_{k=1}^{m}\left[\nabla \log f_\theta(u_k,y)\right]\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)}}{\sum_{k=1}^{m}\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)}}\right].$$

Because the MCLA gradient is a product of three functions of θ, the product rule yields three terms:

$$\begin{aligned}\nabla^2 l_m(\theta|y) = &\frac{\sum_{k=1}^m \left[\nabla^2 \log f_\theta(u_k, y)\right] \dfrac{f_\theta(u_k, y)}{\tilde{f}(u_k)}}{\sum_{k=1}^m \dfrac{f_\theta(u_k, y)}{\tilde{f}(u_k)}} \\ &+ \frac{\sum_{k=1}^m \left[\nabla \log f_\theta(u_k, y)\right] \left[\nabla \dfrac{f_\theta(u_k, y)}{\tilde{f}(u_k)}\right]}{\sum_{k=1}^m \dfrac{f_\theta(u_k, y)}{\tilde{f}(u_k)}} \\ &- \left[\frac{\sum_{k=1}^m \left[\nabla \log f_\theta(u_k, y)\right] \dfrac{f_\theta(u_k, y)}{\tilde{f}(u_k)}}{\left(\sum_{k=1}^m \dfrac{f_\theta(u_k, y)}{\tilde{f}(u_k)}\right)^2}\right] \left[\nabla \sum_{k=1}^m \frac{f_\theta(u_k, y)}{\tilde{f}(u_k)}\right].\end{aligned}$$

Completing the derivative for each term gives

$$\begin{aligned}\nabla^2 l_m(\theta|y) = &\frac{\sum_{k=1}^m \left[\nabla^2 \log f_\theta(u_k, y)\right] \dfrac{f_\theta(u_k, y)}{\tilde{f}(u_k)}}{\sum_{k=1}^m \dfrac{f_\theta(u_k, y)}{\tilde{f}(u_k)}} \\ &+ \frac{\sum_{k=1}^m \left[\nabla \log f_\theta(u_k, y)\right] \left[\nabla \log f_\theta(u_k, y)\right]^T \dfrac{f_\theta(u_k, y)}{\tilde{f}(u_k)}}{\sum_{k=1}^m \dfrac{f_\theta(u_k, y)}{\tilde{f}(u_k)}} \\ &- \left[\frac{\sum_{k=1}^m \left[\nabla \log f_\theta(u_k, y)\right] \dfrac{f_\theta(u_k, y)}{\tilde{f}(u_k)}}{\left(\sum_{k=1}^m \dfrac{f_\theta(u_k, y)}{\tilde{f}(u_k)}\right)^2}\right] \left[\nabla \log f_\theta(u_k, y)\right]^T.\end{aligned} \tag{3.3}$$

We recognize the last term contains an expression for the MCLA gradient and rewrite

the Hessian as

$$\begin{aligned}\nabla^2 l_m(\theta|y) = & \frac{\sum_{k=1}^m \left[\nabla^2 \log f_\theta(u_k, y)\right] \frac{f_\theta(u_k, y)}{\tilde{f}(u_k)}}{\sum_{k=1}^m \frac{f_\theta(u_k, y)}{\tilde{f}(u_k)}} \\ & + \frac{\sum_{k=1}^m \left[\nabla \log f_\theta(u_k, y)\right] \left[\nabla \log f_\theta(u_k, y)\right]^T \frac{f_\theta(u_k, y)}{\tilde{f}(u_k)}}{\sum_{k=1}^m \frac{f_\theta(u_k, y)}{\tilde{f}(u_k)}} \\ & - \left[\nabla l_m(\theta|y)\right] \left[\nabla l_m(\theta|y)\right]^T.\end{aligned} \tag{3.4}$$

To reduce the risk of catastrophic cancellation, we combine the last two terms of the Hessian:

$$\begin{aligned}\nabla^2 l_m(\theta|y) = & \frac{\sum_{k=1}^m \left[\nabla^2 \log f_\theta(u_k, y)\right] \frac{f_\theta(u_k, y)}{\tilde{f}(u_k)}}{\sum_{k=1}^m \frac{f_\theta(u_k, y)}{\tilde{f}(u_k)}} \\ & + \frac{\sum_{k=1}^m \left[\nabla \log f_\theta(u_k, y) - \nabla l_m(\theta|y)\right] \left[\nabla \log f_\theta(u_k, y) - \nabla l_m(\theta|y)\right]^T \frac{f_\theta(u_k, y)}{\tilde{f}(u_k)}}{\sum_{k=1}^m \frac{f_\theta(u_k, y)}{\tilde{f}(u_k)}}.\end{aligned} \tag{3.5}$$

We are able to combine the last two terms because $\nabla l_m(\theta|y)$ is a weighted mean of $\nabla \log f_\theta(u_k, y)$. This is easily understood if we let

$$Z = \nabla \log f_\theta(u_k, y)$$

and use the following equality:

$$E(ZZ^T) - E(Z)E(Z)^T = E\left[(Z - EZ)(Z - EZ)^T\right]. \tag{3.6}$$

3.2 Sources of Error

Two sources of error arise when using MCLA for GLMMs. The first source of error is from the process of sampling the observed data while the second source of error is from the Monte Carlo process.

The first source is the variation of the MLE about the true parameter value. We call this "sampling error." Let $\hat{\theta}$ represent the MLE resulting from data with sample size n and θ^* represent the true parameter value. Then this source of variability is the variance of the asymptotic distribution of $\sqrt{n}\left(\hat{\theta} - \theta^*\right)$. When calculating the sampling error, the data are treated as random, which implies the MLE is random as well.

The second source is the variation of the MCMLE about the MLE. We call this "Monte Carlo error." Let $\hat{\theta}_m$ represent the MCMLE resulting from a Monte Carlo sample size m. Then this source of variability is the variance of the asymptotic distribution of $\sqrt{m}\left(\hat{\theta}_m - \hat{\theta}\right)$. When calculating the Monte Carlo error, the data are treated as fixed (not random), which implies the MLE is fixed as well.

We consider these two sources of error separately, following the lead of Geyer (1994). Sung (2003) found that the asymptotic errors can be added when the importance sampling distribution is chosen independently of the data, but these results cannot be applied in this thesis because the importance sampling distribution used in `glmm` is constructed based on the data.

3.2.1 Sampling Error and Fisher Information

To measure the sampling error, we use Fisher information. Let $J(\hat{\theta}_m)$ indicate our estimate of observed Fisher information. Then

$$J(\hat{\theta}_m) = -\nabla^2 l_m(\hat{\theta}_m|y) \tag{3.7}$$

and a consistent estimator of the asymptotic variance of the MLE is $\left[J(\hat{\theta}_m)\right]^{-1}$, as long as the regularity conditions in Geyer (2013) are met. If the user is unsure as to whether the sample sizes are large enough for this approximation to hold, the user can calculate bootstrap standard deviations and compare them to the standard errors calculated using estimated Fisher information. However, bootstrapping a Monte Carlo calculation is very slow.

We can also calculate expected Fisher information, but we focus on observed Fisher information because it is less computationally expensive.

3.2.2 Monte Carlo Error

Equation (4.41) shows the Monte Carlo error of the gradient of the log likelihood at the MLE $\hat{\theta}$ is

$$V = \frac{1}{\gamma_1^2} \int \left[\nabla \log f_{\hat{\theta}}(u, y)\right] \left[\nabla \log f_{\hat{\theta}}(u, y)\right]^T \frac{f_{\hat{\theta}}(u, y)^2}{\tilde{f}(u)}\, du, \tag{3.8}$$

where γ_1 is defined in Equation (4.17). We can estimate this with the MCMLE $\hat{\theta}_m$:

$$\hat{V} = \frac{\frac{1}{m}\sum_{k=1}^{m} \left(\left[\nabla \log f_{\hat{\theta}_m}(u_k, y)\right] \left[\nabla \log f_{\hat{\theta}_m}(u_k, y)\right]^T \frac{f_{\hat{\theta}_m}(u_k, y)^2}{\tilde{f}(u_k)} \right)}{\left(\frac{1}{m}\sum_{k=1}^{m} \frac{f_{\hat{\theta}_m}(u_k, y)}{\tilde{f}(u_k)} \right)^2} \tag{3.9}$$

.

Let

$$\hat{U} = \nabla^2 l_m(\hat{\theta}_m|y). \tag{3.10}$$

In other words, $\hat{U}$ is the MCLA Hessian evaluated at the MCMLE. Let U be the limit of the MCLA Hessian as the Monte Carlo sample size increases to infinity. Then the Monte Carlo error

$$U^{-1} V U^{-1} \tag{3.11}$$

has plug in estimator

$$\hat{U}^{-1} \hat{V} \hat{U}^{-1} \tag{3.12}$$

under the regularity conditions listed in Geyer (1994). An importance sampling distribution for MCLA is proposed in Section 4.1. Section 4.2.6 proves that the Monte Carlo error resulting from the proposed MCLA implementation is finite.

3.3 The Density of the Random Effects

This section expresses the log density of the random effects and its first two derivatives. These expressions are necessary for Equations (3.1), (3.2), and (3.5).

For $t = 1, \ldots, K$, let E_t be a diagonal matrix with ones and zeroes on the diagonal so that $\sum_{t=1}^{K} E_t$ is an identity matrix and so that $D = \sum_{t=1}^{K} \nu_t E_t$. That is, the diagonal of E_t indicates which random effects have variance ν_t. Let q_t be the trace of E_t; then q_t is the number of random effects associated with variance component ν_t. Since q is the number of random effects in the model, $q = \sum_{t=1}^{K} q_t$.

Before we begin with derivatives, we write the log density of the random effects:

$$\log f_\nu(u) = -\frac{q}{2}\log(2\pi) - \frac{1}{2}\log|D| - \frac{1}{2}u^T D^{-1}u.$$

Because D is diagonal, the determinant of D is

$$|D| = \nu_1^{q_1} \ldots \nu_K^{q_K}.$$

Therefore, we can write the log density of the random effects as

$$\log f_\nu(u) = -\frac{q}{2}\log(2\pi) - \frac{1}{2}\left[\sum_{t=1}^{K} q_t \log \nu_t\right] - \frac{1}{2}u^T D^{-1}u.$$

Next we use the fact that $D = \sum_{t=1}^{K} \nu_t E_t$:

$$\begin{aligned}\log f_\nu(u) &= -\frac{q}{2}\log(2\pi) - \frac{1}{2}\left[\sum_{t=1}^{K} q_t \log \nu_t\right] - \frac{1}{2}u^T\left[\sum_{t=1}^{K}\frac{1}{\nu_t}E_t\right]u \\ &= -\frac{q}{2}\log(2\pi) - \frac{1}{2}\left[\sum_{t=1}^{K} q_t \log \nu_t\right] - \frac{1}{2}\sum_{t=1}^{K}\frac{1}{\nu_t}u^T E_t u.\end{aligned} \tag{3.13}$$

The first and second derivatives of $\log f_\nu(u)$ with respect to ν_t are:

$$\frac{\partial}{\partial \nu_t}\log f_\nu(u) = -\frac{q_t}{2\nu_t} + \frac{1}{2\nu_t^2}u^T E_t u \tag{3.14}$$

and

$$\frac{\partial^2}{\partial \nu_t^2}\log f_\nu(u) = \frac{q_t}{2\nu_t^2} - \frac{1}{\nu_t^3}u^T E_t u. \tag{3.15}$$

All mixed partial derivatives are 0. That is, for all $s \neq t$,

$$\frac{\partial^2 \log f_\nu(u)}{\partial \nu_s \, \partial \nu_t} = 0.$$

3.4 The Density of the Data

This section expresses the density of the observed data and its first two derivatives. These espressions are necessary for Equations (3.1), (3.2), and (3.5).

Recall that η represents the canonical random vector. Define $c(\eta)$ to be the cumulant function for $\eta \in \mathbb{R}^n$ where

$$\log f_\beta(y|u) = y^T \eta - c(\eta). \tag{3.16}$$

Details for the cumulant function are in Section 3.5. The gradient and Hessian of the MCLA require derivatives of $\log f_\beta(y|u)$. The first derivative of $\log f_\beta(y|u)$ with respect to β is a vector of length p with jth component

$$\frac{\partial}{\partial \beta_j} \log f_\beta(y|u).$$

To condense the derivative, we write it with respect to the entire vector β:

$$\begin{aligned} \frac{\partial}{\partial \beta} \log f_\beta(y|u) &= \frac{\partial}{\partial \beta} \left[y^T \eta - c(\eta) \right] \\ &= \frac{\partial}{\partial \beta} \left[y^T (X\beta + Zu) - c(\eta) \right] \\ &= X^T y - \frac{\partial}{\partial \beta} c(\eta). \end{aligned}$$

We use $c'(\eta)$ to indicate the derivative of $c(\eta)$ with respect to η and $c''(\eta)$ to represent the second derivative of $c(\eta)$ with respect to η. In particular, $c'(\eta)$ is a vector as

defined in Section 3.5. The multivariate chain rule gives

$$\begin{aligned}\frac{\partial}{\partial\beta}\log f_\beta(y|u) &= X^T y - \frac{\partial\eta}{\partial\beta}\,\frac{\partial c(\eta)}{\partial\eta} \\ &= X^T y - X^T\left[c'(\eta)\right] \\ &= X^T\left[y - c'(\eta)\right],\end{aligned} \tag{3.17}$$

where $y - c'(\eta)$ is a vector with components as described in (3.28). The second derivative is

$$\begin{aligned}\frac{\partial^2}{\partial\beta^2}\log f_\beta(y|u) &= \frac{\partial^2}{\partial\beta^2}\left[y^T\eta - c(\eta)\right] \\ &= \frac{\partial}{\partial\beta}X^T\left[y - c'(\eta)\right] \\ &= -X^T\left[\frac{\partial}{\partial\beta}c'(\eta)\right],\end{aligned} \tag{3.18}$$

and the multivariate chain rule gives

$$\begin{aligned}-\frac{\partial^2}{\partial\beta^2}\log f_\beta(y|u) &= -X^T\left[\frac{\partial\, c'(\eta)}{\partial\eta}\right]\frac{\partial\eta}{\partial\beta} \\ &= -X^T\left[c''(\eta)\right]X.\end{aligned} \tag{3.19}$$

3.5 The Cumulant Function

This section expresses the cumulant function and its first two derivatives. We distinguish between the cumulant function for a vector η and for a scalar η_i. Let $c_i(\eta_i)$ denote the cumulant function for η_i. Because the components of y are conditionally independent given the random effects in a GLMM, we can write

$$c(\eta) = \sum_{i=1}^{n} c_i(\eta_i).$$

Define $c'(\eta)$ to be a vector with $c_i'(\eta_i)$ as the ith entry. Define $c''(\eta)$ to be a diagonal matrix with $c_i''(\eta_i)$ as the ith diagonal entry. Note that $c'(\eta)$ is the expected value of the responses conditional on the random effects. That is, $c_i'(\eta_i) = E(Y_i|U = u)$.

The exact specification of the cumulant function and its derivatives depends on the conditional distribution of the response vector's elements given the random effects. We perform Bernoulli and Poisson GLMM: the elements of the response vector are all conditionally Bernoulli (with responses being 0 or 1) or conditionally Poisson (with responses being $0, 1, 2, \ldots$) given the random effects.

When the elements of the response vector are conditionally Bernoulli-distributed given the random effects, then

$$c_i(\eta_i) = \log(1 + e^{\eta_i}). \tag{3.20}$$

$$c_i'(\eta_i) = \frac{e^{\eta_i}}{1 + e^{\eta_i}} \tag{3.21}$$

$$c_i''(\eta_i) = \frac{e^{\eta_i}}{1 + e^{\eta_i}} - \frac{e^{2\eta_i}}{(1 + e^{\eta_i})^2}. \tag{3.22}$$

We rewrite these expressions to improve their computational stability. The following representation of the cumulant avoids overflow resulting from large values of η_i.

$$c_i(\eta_i) = \begin{cases} \log(1 + e^{\eta_i}) & \text{if } \eta_i \leq 0, \\ \eta_i + \log(e^{-\eta_i} + 1) & \text{if } \eta_i > 0, \end{cases} \tag{3.23}$$

To reduce loss of precision in the cumulant when η_i is near zero, calculation of the function $\eta_i \mapsto \log(1 + \eta_i)$ should be performed with the R function `log1p()`. This function uses the Taylor series expansion for $\log(1 + \eta_i)$ around $\eta_i = 0$.

Similarly, we rewrite the cumulant's derivative by considering two cases for η_i:

$$c_i'(\eta_i) = \begin{cases} \dfrac{e^{\eta_i}}{1+e^{\eta_i}} & \text{if } \eta_i \leq 0 \\ \dfrac{1}{1+e^{-\eta_i}} & \text{if } \eta_i > 0 \end{cases} \tag{3.24}$$

Equation (3.22) is in danger of "catastrophic cancellation," a loss of precision resulting from subtraction of two numbers. Let

$$p(\eta_i) = \begin{cases} \dfrac{e^{\eta_i}}{1+e^{\eta_i}} & \text{if } \eta_i \leq 0 \\ \dfrac{1}{1+e^{-\eta_i}} & \text{if } \eta_i > 0 \end{cases} \tag{3.25}$$

and

$$q(\eta_i) = \begin{cases} \dfrac{e^{-\eta_i}}{1+e^{-\eta_i}} & \text{if } \eta_i \geq 0 \\ \dfrac{1}{1+e^{\eta_i}} & \text{if } \eta_i < 0 \end{cases} \tag{3.26}$$

Then

$$c''(\eta_i) = p(\eta_i)q(\eta_i) \tag{3.27}$$

and

$$y_i - c'(\eta_i) = \begin{cases} -p(\eta_i) & \text{if } y_i = 0 \\ q(\eta_i) & \text{if } y_i = 1. \end{cases} \tag{3.28}$$

When the elements of the response vector are conditionally Poisson-distributed

given the random effects, then

$$\begin{aligned} c_i(\eta_i) &= e^{\eta_i} \\ c_i'(\eta_i) &= e^{\eta_i} \\ c_i''(\eta_i) &= e^{\eta_i}. \end{aligned} \tag{3.29}$$

These expressions overflow for η_i large and underflow for η_i small. No precautions are taken to improve these expressions' computationally stability because little can be done. There is a Taylor series expansion for the function $\eta_i \mapsto e^{\eta_i} - 1$, which can help in the case when $y_i = 1$, but there are not Taylor series expansions around integers other than 1.

Both Bernoulli and Poisson cumulant functions are strictly positive and have strictly positive first derivatives.

Chapter 4

An MCLA Implementation

MCLA requires an importance sampling distribution for generating random effects. In this chapter, we propose an importance sampling distribution for MCLA implementation. We then characterize the asymptotic behavior of the MCLA gradient, which is used for calculating the Monte Carlo error of the MCMLE (Equation (3.12)).

4.1 The Proposed Importance Sampling Distribution

Let s be a vector of length q that represents the random effects on the standard normal scale. That is, s is defined such that $u = D^{1/2}s$. Let σ be a vector of length K with components $\sqrt{\nu_t}$, $t = 1, \ldots, K$. Let β^*, s^* and σ^* denote the PQL estimates for β, s and σ, respectively. Let

$$A^* = \sum_{t=1}^{K} E_t \, \sigma_t^*$$

and

$$D^* = A^* A^*.$$

We can "unstandardize" our PQL-predicted random effects:

$$u^* = A^* s^*.$$

Let p_1, p_2, p_3 be proportions such that $p_1 + p_2 + p_3 = 1$. Let $f(u|\mu, \Sigma)$ denote the density for a multivariate normal distribution with mean μ and variance matrix Σ. Let $\grave{f}(u|0, D^*)$ denote the density for a q-dimensional multivariate t distribution with mean 0, scale matrix D^*, and $\zeta = 5$ degrees of freedom. Then we propose the following importance sampling distribution:

$$\begin{aligned}\tilde{f}(u) = p_1 \grave{f}(u|0, D^*) + p_2 f(u \,|\, u^*, D^*) + \\ + p_3 f(u \,|\, u^*, (Z^T c''(X\beta^* + Zu^*)Z + (D^*)^{-1})^{-1}).\end{aligned} \tag{4.1}$$

The first component $\grave{f}(\,\cdot\,|0, D^*)$ of Equation (4.1) is chosen to ensure the gradient of the MCLA has a central limit theorem (as shown in Section 4.2.6). The second component is chosen because it is centered at u^* and has variance D^*. The last component is centered at u^* and has a variance based on the Hessian of the penalized likelihood from PQL. More specifically, the Hessian of the log density of the last distribution matches the Hessian of the log density of the target distribution $f_\theta(u, y)$.

4.2 Asymptotic Behavior of the MCLA Gradient

In this section, we focus on the asymptotic behavior of the MCLA gradient resulting from use of our proposed importance sampling distribution. We focus on the cases of Bernoulli- and Poisson-distributed elements of the response vector. All limits are taken as the Monte Carlo sample size increases to infinity. For every θ and every y,

we want to show

$$\nabla l_m(\theta|y) \longrightarrow \nabla l(\theta|y)$$

almost surely as the Monte Carlo sample size m increases to infinity. We also want to show

$$\sqrt{m}\,(\nabla l_m(\theta|y) - \nabla l(\theta|y))$$

converges in distribution to a normal distribution with mean 0 and finite variance for every θ and every y as the Monte Carlo sample size increases to infinity. This is analogous to showing the gradient of the Monte Carlo likelihood approximation is consistent and asymptotically normal. These results ensure that our Monte Carlo errors are finite.

4.2.1 Recognizing a Normal Density

The following lemma about normal densities will be used several times in this chapter's proofs.

Lemma 4.2.1

Let u be a vector. Let a be a vector of the same length as u. Let b be a constant. If D is a variance matrix (positive definite and symmetric), then the function

$$u \mapsto \exp\left(-u^T D^{-1} u + a^T u + b\right)$$

is proportional to a normal density with mean $\frac{1}{2}Da$ and variance matrix $\frac{1}{2}D$.

Proof

First, we multiply our expression by a constant and rearrange the terms:

$$\exp\left(-u^T D^{-1}u + a^T u + b\right) \propto \exp\left(-u^T D^{-1}u + a^T u + b\right) \exp\left(-b - \frac{a^T Da}{4}\right)$$
$$= \exp\left(-u^T D^{-1}u + a^T u - \frac{(Da)^T a}{4}\right).$$

In the last term, we multiply by the identity matrix in the form of $D^{-1}D$:

$$\exp\left(-u^T D^{-1}u + a^T u - \frac{(Da)^T a}{4}\right) = \exp\left(-u^T D^{-1}u + a^T u - \frac{(Da)^T D^{-1} Da}{4}\right).$$

Next, we factor the expression:

$$\exp\left(-u^T D^{-1}u + a^T u - \frac{(Da)^T D^{-1} Da}{4}\right)$$
$$= \exp\left(-\left(u - \frac{1}{2}Da\right)^T D^{-1}\left(u - \frac{1}{2}Da\right)\right).$$

Now, we multiply and divide by the same number:

$$\exp\left(-\left(u - \frac{1}{2}Da\right)^T D^{-1}\left(u - \frac{1}{2}Da\right)\right)$$
$$= \exp\left(-\frac{1}{2}\left(u - \frac{1}{2}Da\right)^T \left(2D^{-1}\right)\left(u - \frac{1}{2}Da\right)\right).$$

We recognize this expression as the kernel for a multivariate normal density with mean $\frac{1}{2}Da$ and variance matrix $\frac{1}{2}D$.

□

The converse of Lemma 4.2.1 is also true, but we do not prove it.

4.2.2 Passing the Derivative under the Integral

This lemma will be used in Section 4.2.3.

Lemma 4.2.2

$$E_{f_\theta}\left[\nabla \log f_\theta(u,y)|y\right] = \nabla l(\theta|y). \tag{4.2}$$

Proof

We begin with the right-hand side of Equation (4.2):

$$\begin{aligned} \nabla l(\theta|y) &= \frac{\nabla L(\theta|y)}{L(\theta|y)} \\ &= \frac{\nabla \int f_\theta(u,y)\,du}{L(\theta|y)} \end{aligned} \tag{4.3}$$

To pass the derivative in Equation (4.3) under the integral sign, we use a lemma from Ferguson (1996, p. 124). We provide separate arguments for derivatives with respect to elements of β and for derivatives with respect to elements of ν.

Let k be given. Let $a_1 < a_2$. If $\partial f_\theta(u,y)/\partial\beta_k$ exists and is continuous in β_k for all u and for all $\beta_k \in (a_1, a_2)$, and if

$$\left|\frac{\partial}{\partial\beta_k} f_\theta(u,y)\right| \leq \kappa_1(u) \tag{4.4}$$

on (a_1, a_2) where

$$\int \kappa_1(u)\, du < \infty,$$

and if

$$\int f_\theta(u, y)\, du < \infty, \tag{4.5}$$

then

$$\frac{\partial}{\partial \beta_k} \int f_\theta(u, y)\, du = \int \frac{\partial}{\partial \beta_k} f_\theta(u, y)\, du. \tag{4.6}$$

Condition (4.5) is clearly met because the integral is the marginal probability mass function for y, which can also be thought of as the likelihood $L(\theta|y)$. For any θ and any y, $L(\theta|y)$ exists.

Next, we show that $\partial f_\theta(u, y)/\partial \beta_k$ exists and is continuous in β_k for all u. We start by writing the joint density as the product of marginal and conditional:

$$\frac{\partial}{\partial \beta_k} f_\theta(u, y) = \frac{\partial}{\partial \beta_k} \left[f_\beta(y|u) f_\nu(u) \right].$$

Since $f_\nu(u)$ is free of β_k, this derivative can be rewritten as

$$\begin{aligned} \frac{\partial}{\partial \beta_k} f_\theta(u, y) &= \left[\frac{\partial}{\partial \beta_k} f_\beta(y|u) \right] f_\nu(u) \\ &= f_\beta(y|u) \left[\frac{\partial}{\partial \beta_k} \log f_\beta(y|u) \right] f_\nu(u). \end{aligned}$$

The density $f_\nu(u)$ does not contain β_k and so it is automatically continuous in β_k for every u. The density $f_\beta(y|u)$ is an exponential family. Equation (3.16) clearly shows that $f_\beta(y|u)$ is continuous in β_k for every u. The derivative of $\log f_\beta(y|u)$ is

(3.17), and the kth component of this vector represents $\partial \log f_\beta(y|u)/\partial\beta_k$. Since $c'(\eta)$ is continuous in β_k for all u, so is $\partial \log f_\beta(y|u)/\partial\beta_k$.

To prove Equation (4.6), the final step is to find $\kappa_1(u)$ and show that it is integrable. The function to be dominated by $\kappa_1(u)$ is

$$\left|\frac{\partial \log f_\beta(y|u)}{\partial\beta_k}\ f_\beta(y|u)\ f_\nu(u)\right| = \left|\frac{\partial \log f_\beta(y|u)}{\partial\beta_k}\ \exp\left(y^T\eta\right)\ \exp\left(-c(\eta)\right)\ f_\nu(u)\right|.$$

Letting b be a constant such that

$$f_\nu(u) = b\ \exp\left(-u^T D^{-1} u\right),$$

we see

$$\begin{aligned}&\left|\frac{\partial \log f_\beta(y|u)}{\partial\beta_k}\ f_\beta(y|u)\ f_\nu(u)\right| \\ &\qquad = \left|b\ \left[\frac{\partial}{\partial\beta_k}\log f_\beta(y|u)\right]\ \exp\left(y^T\eta\right)\ \exp\left(-c(\eta)\right)\ \exp\left(-u^T D^{-1} u\right)\right|.\end{aligned}$$

When elements of the response vector are conditionally Poisson or Bernoulli given the random effects, $c(\eta) > 0$, which implies $0 < \exp\left(-c(\eta)\right) < 1$. Therefore,

$$\begin{aligned}&\left|b\ \left[\frac{\partial}{\partial\beta_k}\log f_\beta(y|u)\right]\ \exp\left(y^T\eta\right)\ \exp\left(-c(\eta)\right)\ \exp\left(-u^T D^{-1} u\right)\right| \\ &< \left|b\ \left[\frac{\partial}{\partial\beta_k}\log f_\beta(y|u)\right]\ \exp\left(y^T\eta\right)\ \exp\left(-u^T D^{-1} u\right)\right| \\ &= \left|b\ \left[\frac{\partial}{\partial\beta_k}\log f_\beta(y|u)\right]\ \exp\left(y^T X\beta\right)\ \exp\left(y^T Z u\right)\ \exp\left(-u^T D^{-1} u\right)\right|.\end{aligned}$$

We must now dominate $\exp\left(y^T X\beta\right)$. The function

$$\beta_k \mapsto \exp\left(y^T X\beta\right)$$

is continuous over the compact set $[a_1, a_2]$. Therefore, it achieves its maximum. Let M_1 denote this maximum. Then

$$\exp\left(y^T X\beta\right) \leq M_1$$

for β_k in (a_1, a_2). This implies

$$\begin{aligned}\left| b \left[\frac{\partial}{\partial\beta_k} \log f_\beta(y|u)\right] \exp\left(y^T X\beta\right) \exp\left(y^T Zu\right) \exp\left(-u^T D^{-1} u\right)\right| \\ \leq \left| b \left[\frac{\partial}{\partial\beta_k} \log f_\beta(y|u)\right] M_1 \exp\left(y^T Zu\right) \exp\left(-u^T D^{-1} u\right)\right|.\end{aligned}$$

We now want to dominate

$$\left|\frac{\partial}{\partial\beta_k} \log f_\beta(y|u)\right|.$$

Recall that $\partial \log f_\beta(y|u)/\partial\beta_k$ is the kth component of (3.17). If the elements of the response vector are conditionally Bernoulli given the random effects, then elements of the response vector are either 0 or 1 and the first derivative of the cumulant is a probability. Therefore, $|y_i - c_i(\eta_i)| \leq 1$ for all i, which implies

$$\left|\frac{\partial}{\partial\beta_k} \log f_\beta(y|u)\right| \leq 1 \sum_{j=1}^{n} |x_{jk}|. \tag{4.7}$$

Let M_2 be a constant such that

$$\sum_{j=1}^{n} |x_{jk}| < M_2. \tag{4.8}$$

This implies

$$\begin{aligned} &\left| b \left[\frac{\partial}{\partial \beta_k} \log f_\beta(y|u) \right] M_1 \exp\left(y^T Z u\right) \exp\left(-u^T D^{-1} u\right) \right| \\ &\qquad < \left| b\, M_2\, M_1 \exp\left(y^T Z u\right) \exp\left(-u^T D^{-1} u\right) \right| \end{aligned}$$

as long as the elements of the response vector are conditionally Bernoulli given the random effects. Therefore, for the Bernoulli case,

$$\begin{aligned} \kappa_1(u) &= \left| b\, M_2\, M_1 \exp\left(y^T Z u\right) \exp\left(-u^T D^{-1} u\right) \right| \\ &= b\, M_2\, M_1 \exp\left(y^T Z u\right) \exp\left(-u^T D^{-1} u\right) \\ &= b\, M_2\, M_1 \exp\left(-u^T D^{-1} u + y^T Z u\right). \end{aligned} \tag{4.9}$$

Since $M_1 > 0$, $M_2 > 0$, and $b > 0$, we recognize $\kappa_1(u)$ for the Bernoulli case as proportional to the density of a multivariate normal distribution by the lemma in Section 4.2.1. Therefore, $\kappa_1(u)$ is integrable.

We now turn our attention to expressing $\partial \log f_\beta(y|u)/\partial \beta_k$ for the case in which the elements of the response vector are conditionally Poisson given the random effects. This derivative is the kth element shown in Equation (3.17), where elements of $c'(\eta)$ are given in Equation (3.29). Let $x_{j\cdot}$ denote the jth row of X and let $z_{j\cdot}$ denote the

jth row of Z. Then, for the Poisson case, we use the triangle inequality:

$$\begin{aligned}\left|\frac{\partial \log f_\beta(y|u)}{\partial \beta_k}\right| &= \left|\sum_{j=1}^{n} x_{jk}y_j - x_{jk}\exp\left(x_{j\cdot}\beta + z_{j\cdot}u\right)\right| \\ &\leq \sum_{j=1}^{n}|x_{jk}y_j| + \sum_{j=1}^{n}|x_{jk}|\exp\left(x_{j\cdot}\beta + z_{j\cdot}u\right) \qquad (4.10)\\ &= \sum_{j=1}^{n}|x_{jk}y_j| + \sum_{j=1}^{n}|x_{jk}|\exp\left(x_{j\cdot}\beta\right)\exp\left(z_{j\cdot}u\right).\end{aligned}$$

The function

$$\beta_k \mapsto \exp\left(x_{j\cdot}\beta\right)$$

is continuous on the compact set $[a_1, a_2]$. Therefore, it attains its maximum, which we denote by N_j. Then

$$\sum_{j=1}^{n}|x_{jk}|\ \exp\left(x_{j\cdot}\beta\right)\exp\left(z_{j\cdot}u\right) \leq \sum_{j=1}^{n}|x_{jk}|\ N_j\exp\left(z_{j\cdot}u\right).$$

Then we have dominated our function for the Poisson case and found $\kappa_1(u)$:

$$\kappa_1(u) = b\left[\sum_{j=1}^{n}|x_{jk}y_j| + \sum_{j=1}^{n}|x_{jk}|\ N_j\exp\left(z_{j\cdot}u\right)\right]M_1\ \exp\left(y^TZu\right)\ \exp\left(-u^TD^{-1}u\right).$$

We must now show $\kappa_1(u)$ is integrable. The expression

$$\begin{aligned}
&\int b \left[\sum_{j=1}^{n} |x_{jk} y_j| + \sum_{j=1}^{n} |x_{jk}| \, N_j \exp\left(z_{j\cdot} u\right) \right] M_1 \exp\left(y^T Z u\right) \exp\left(-u^T D^{-1} u\right) \, du \\
&= b \, M_1 \, \sum_{j=1}^{n} |x_{jk} y_j| \, \int \exp\left(-u^T D^{-1} u + y^T Z u\right) \, du + \\
&\quad + b \, M_1 \, \int \sum_{j=1}^{n} |x_{jk}| \, N_j \, \exp\left(-u^T D^{-1} u + y^T Z u + z_{j\cdot} u\right) \, du
\end{aligned}$$

exists because $\exp\left(-u^T D^{-1} u + y^T Z u + z_{j\cdot} u\right)$ and $\exp\left(-u^T D^{-1} u + y^T Z u\right)$ are each proportional to a normal density by Section 4.2.1.

We have now shown condition (4.4) for the Bernoulli case and the Poisson case. We have now met all the requirements to prove Equation (4.6).

We now prove a similar statement about differentiating with respect to a variance component. Let t be given. Let $a_2 > a_1 > 0$. If $\partial f_\theta(u, y)/\partial \nu_t$ exists and is continuous in ν_t for all u and for all $\nu_t \in (a_1, a_2)$, and if

$$\left| \frac{\partial}{\partial \nu_t} f_\theta(u, y) \right| \leq \kappa_2(u) \tag{4.11}$$

on (a_1, a_2) where

$$\int \kappa_2(u) \, du < \infty,$$

then, for $a_1 < \nu_t < a_2$,

$$\frac{\partial}{\partial \nu_t} \int f_\theta(u, y) \, du = \int \frac{\partial}{\partial \nu_t} f_\theta(u, y) \, du. \tag{4.12}$$

To show $\partial f_\theta(u, y)/\partial \nu_t$ exists and is continuous in ν_t for all u, we start by expressing

our joint density as the product of marginal and conditional and take the derivative:

$$\begin{aligned}\frac{\partial}{\partial \nu_t} f_\theta(u,y) &= \frac{\partial}{\partial \nu_t}\left[f_\beta(y|u) f_\nu(u)\right] \\ &= \frac{\partial}{\partial \nu_t}\left[f_\nu(u)\right] f_\beta(y|u) \\ &= \left[\frac{\partial}{\partial \nu_t}\log f_\nu(u)\right]\ f_\nu(u)\ f_\beta(y|u).\end{aligned} \tag{4.13}$$

The density $f_\beta(y|u)$ does not contain ν_t and therefore it is continuous in ν_t for all u. The density $f_\nu(u)$ is a multivariate normal distribution; therefore, it is continuous in ν_t for all u. The partial derivative is shown in (3.14), and it is continuous in ν_t for all u.

To prove (4.12), the final step is to find $\kappa_2(u)$ and show that it is integrable. We need $\kappa_2(u)$ to dominate

$$\left| f_\beta(y|u)\ f_\nu(u)\ \frac{\partial \log f_\nu(u)}{\partial \nu_t}\right| = \left|\exp\left(y^T\eta - c(\eta)\right)\ f_\nu(u)\ \frac{\partial \log f_\nu(u)}{\partial \nu_t}\right|.$$

Using Equation (3.14) leads to

$$\begin{aligned}&\left|\exp\left(y^T\eta - c(\eta)\right)\ f_\nu(u)\ \frac{\partial \log f_\nu(u)}{\partial \nu_t}\right| \\ &= \left|\exp\left(y^T\eta - c(\eta)\right)\ f_\nu(u)\ \left[\frac{-q_t}{2\nu_t} + \frac{u^T E_t u}{2\nu_t^2}\right]\right| \\ &= \exp\left(y^T\eta - c(\eta)\right)\ f_\nu(u)\ \left|\frac{-q_t}{2\nu_t} + \frac{u^T E_t u}{2\nu_t^2}\right|.\end{aligned} \tag{4.14}$$

Let b be a constant such that

$$f_\nu(u) \le b\left(\prod_{j=1}^K \nu_j^{-q_j/2}\right)\ \exp\left(-\frac{1}{2}\sum_{j=1}^K \frac{1}{\nu_j} u^T E_j u\right).$$

Then

$$\begin{aligned}
&\exp\left(y^T\eta - c(\eta)\right)\ f_\nu(u)\ \left|\frac{-q_t}{2\nu_t} + \frac{u^T E_t u}{2\nu_t^2}\right| \\
&\quad \le b\exp\left(y^T\eta - c(\eta)\right)\ \left(\prod_{j=1}^{K}\nu_j^{-q_j/2}\right)\exp\left(\frac{1}{2}\sum_{j=1}^{K}\frac{1}{\nu_j}u^T E_j u\right)\ \left|\frac{-q_t}{2\nu_t} + \frac{u^T E_t u}{2\nu_t^2}\right|.
\end{aligned}$$

Using the triangle inequality on the last term gives

$$\begin{aligned}
&b\ \exp\left(y^T\eta - c(\eta)\right)\left(\prod_{j=1}^{K}\nu_j^{-q_j/2}\right)\exp\left(-\frac{1}{2}\sum_{j=1}^{K}\frac{1}{\nu_j}u^T E_j u\right)\left|\frac{-q_t}{2\nu_t} + \frac{u^T E_t u}{2\nu_t^2}\right| \\
&\quad \le b\ \exp\left(y^T\eta - c(\eta)\right)\left(\prod_{j=1}^{K}\nu_j^{-q_j/2}\right)\exp\left(-\frac{1}{2}\sum_{j=1}^{K}\frac{1}{\nu_j}u^T E_j u\right)\left[\frac{q_t}{2\nu_t} + \frac{u^T E_t u}{2\nu_t^2}\right].
\end{aligned}$$

The function

$$\nu_t \mapsto \frac{u^T E_t u}{2\nu_t^2} + \frac{q_t}{2\nu_t}$$

is continuous and decreasing on $[a_1, a_2]$ and therefore attains its maximum at a_1. Then for $a_1 < \nu_t < a_2$,

$$\frac{u^T E_t u}{2\nu_t^2} + \frac{q_t}{2\nu_t} \le \frac{u^T E_t u}{2a_1^2} + \frac{q_t}{2a_1},$$

which implies

$$\exp\left(y^T\eta - c(\eta)\right)\ \left(\prod_{j=1}^{K}\nu_j^{-q_j/2}\right)\ \exp\left(-\frac{1}{2}\sum_{j=1}^{K}\frac{1}{\nu_j}u^TE_ju\right)\ \left[\frac{u^TE_tu}{2\nu_t^2}+\frac{q_t}{2\nu_t}\right]$$
$$\leq \exp\left(y^T\eta - c(\eta)\right)\ \left(\prod_{j=1}^{K}\nu_j^{-q_j/2}\right)\ \exp\left(-\frac{1}{2}\sum_{j=1}^{K}\frac{1}{\nu_j}u^TE_ju\right)\ \left[\frac{u^TE_tu}{2a_1^2}+\frac{q_t}{2a_1}\right].$$

The function

$$\nu_t \mapsto \exp\left(-\frac{1}{2}\sum_{j=1}^{K}\frac{1}{\nu_j}u'E_ju\right)$$

is continuous on $[a_1, a_2]$ and attains its maximum at a_2. Therefore, the function is bounded on (a_1, a_2):

$$\exp\left(-\frac{1}{2}\sum_{j=1}^{K}\frac{1}{\nu_j}u'E_ju\right) = \exp\left(-\frac{1}{2}\frac{1}{\nu_t}u'E_tu\right)\exp\left(-\frac{1}{2}\sum_{j\neq t}\frac{1}{\nu_j}u'E_ju\right)$$
$$\leq \exp\left(-\frac{1}{2}\frac{1}{a_2}u'E_tu\right)\exp\left(-\frac{1}{2}\sum_{j\neq t}\frac{1}{\nu_j}u'E_ju\right).$$

Let D_2 be a diagonal matrix such that

$$\exp\left(-u^TD_2^{-1}u\right) = \exp\left(-\frac{1}{2}\frac{1}{a_2}u'E_tu\right)\exp\left(-\frac{1}{2}\sum_{j\neq t}\frac{1}{\nu_j}u'E_ju\right).$$

In particular, note that D_2 is constructed to be free of ν_t. Then

$$\exp\left(y^T\eta - c(\eta)\right)\ \left(\prod_{j=1}^{K}\nu_j^{-q_j/2}\right)\ \exp\left(-\frac{1}{2}\sum_{j=1}^{K}\frac{1}{\nu_j}u^TE_ju\right)\ \left[\frac{u^TE_tu}{2a_1^2}+\frac{q_t}{2a_1}\right]$$
$$\leq \exp\left(y^T\eta - c(\eta)\right)\ \left(\prod_{j=1}^{K}\nu_j^{-q_j/2}\right)\ \exp\left(-u^TD_2^{-1}u\right)\ \left[\frac{u^TE_tu}{2a_1^2}+\frac{q_t}{2a_1}\right].$$

Next, note the function

$$\nu_t \mapsto \prod_{j=1}^{K}\nu_j^{-q_j/2}$$

is continuous on $[a_1, a_2]$ and attains its maximum. Denote this maximum by M. Then, for $a_1 < \nu_t < a_2$,

$$\prod_{j=1}^{K}\nu_j^{-q_j/2} \leq M,$$

which implies

$$\exp\left(y^T\eta - c(\eta)\right)\ \left(\prod_{j=1}^{K}\nu_j^{-q_j/2}\right)\ \exp\left(-u^TD_2^{-1}u\right)\ \left[\frac{u^TE_tu}{2a_1^2}+\frac{q_t}{2a_1}\right]$$
$$\leq \exp\left(y^T\eta - c(\eta)\right)\ M\ \exp\left(-u^TD_2^{-1}u\right)\ \left[\frac{u^TE_tu}{2a_1^2}+\frac{q_t}{2a_1}\right].$$

Define

$$\kappa_2(u) = \exp\left(y^T\eta - c(\eta)\right)\ M\ \exp\left(-u^TD_2^{-1}u\right)\ \left[\frac{u^TE_tu}{2a_1^2}+\frac{q_t}{2a_1}\right].$$

We must now show that $\kappa_2(u)$ is integrable:

$$\begin{aligned}
&\int \kappa_2(u)\, du \\
&= \int \exp\left(y^T\eta - c(\eta)\right)\ M\ \exp\left(-u^T D_2^{-1} u\right)\ \left[\frac{u^T E_t u}{2a_1^2} + \frac{q_t}{2a_1}\right] du \\
&= \int \exp\left(y^T\eta\right)\exp\left(-c(\eta)\right)\ M\ \exp\left(-u^T D_2^{-1} u\right)\ \left[\frac{u^T E_t u}{2a_1^2} + \frac{q_t}{2a_1}\right] du
\end{aligned}$$

When the elements of the response vector are conditionally Poisson or Bernoulli given the random effects, $c(\eta) \geq 0$, which implies $0 \leq \exp\left(-c(\eta)\right) \leq 1$. Using this gives

$$\begin{aligned}
&\int \exp\left(y^T\eta\right)\exp\left(-c(\eta)\right)\ M\ \exp\left(-u^T D_2^{-1} u\right)\ \left[\frac{u^T E_t u}{2a_1^2} + \frac{q_t}{2a_1}\right] du \\
&= \int \exp\left(y^T\eta\right)\ M\ \exp\left(-u^T D_2^{-1} u\right)\ \left[\frac{u^T E_t u}{2a_1^2} + \frac{q_t}{2a_1}\right] du \\
&= \int \exp\left(y^T X\beta\right)\exp\left(y^T Z u\right)\ M\ \exp\left(-u^T D_2^{-1} u\right)\ \left[\frac{u^T E_t u}{2a_1^2} + \frac{q_t}{2a_1}\right] du.
\end{aligned}$$

Rearranging the terms gives

$$\begin{aligned}
&\int \exp\left(y^T X\beta\right)\exp\left(y^T Z u\right)\ M\ \exp\left(-u^T D_2^{-1} u\right)\ \left[\frac{u^T E_t u}{2a_1^2} + \frac{q_t}{2a_1}\right] du \\
&\qquad = \exp\left(y^T X\beta\right)\ M \int \exp\left(-u^T D_2^{-1} u + y^T Z u\right)\ \left[\frac{u^T E_t u}{2a_1^2} + \frac{q_t}{2a_1}\right] du \quad (4.15)
\end{aligned}$$

We recognize

$$\exp(-u^T D_2^{-1} u + y^T Z u)$$

as proportional to a normal density by Section 4.2.1. Therefore, Equation (4.15) is proportional to the expectation of $\frac{u^T E_t u}{2a_1^2} + \frac{q_t}{2a_1}$ with respect to a normal distribution, and this expectation exists. Therefore, we have proven Equation (4.12).

Combining Equation (4.6) and Equation (4.12) yields

$$\nabla \int f_\theta(u, y)\, du = \int \nabla f_\theta(u, y)\, du.$$

Therefore, continuing with Equation (4.3), we see

$$\begin{aligned}
\nabla l(\theta|y) &= \frac{\nabla \int f_\theta(u, y)\, du}{L(\theta|y)} \\
&= \frac{\int \nabla f_\theta(u, y)\, du}{L(\theta|y)} \\
&= \frac{\int \left[\nabla \log f_\theta(u, y)\right] f_\theta(u, y)\, du}{L(\theta|y)}.
\end{aligned}$$

However, $L(\theta|y) = f_\theta(y)$, which is a constant with respect to u and can be passed into the integral. Therefore,

$$\begin{aligned}
\nabla l(\theta|y) &= \frac{\int \left[\nabla \log f_\theta(u, y)\right] f_\theta(u, y)\, du}{f_\theta(y)} \\
&= \int \frac{\left[\nabla \log f_\theta(u, y)\right] f_\theta(u, y)}{f_\theta(y)}\, du.
\end{aligned}$$

Dividing a joint density by a marginal density creates a conditional density. Therefore, this expression is an expectation with respect to a conditional density:

$$\begin{aligned}
\nabla l(\theta|y) &= \int \left[\nabla \log f_\theta(u, y)\right] f_\theta(u|y)\, du \\
&= E_{f_\theta}\left[\nabla \log f_\theta(u, y) \,|\, Y = y\right].
\end{aligned}$$

□

4.2.3 Law of Large Numbers for the MCLA Gradient

Lemma 4.2.3

For every θ,

$$\nabla l_m(\theta|y) \longrightarrow \nabla l(\theta|y) \tag{4.16}$$

almost surely as $m \to \infty$.

Proof

Define

$$\int f_\theta(u, y)\, du = f_\theta(y) = \gamma_1. \tag{4.17}$$

Because we fix y at its observed value, we consider γ_1 to be constant and finite. Additionally, note that

$$\int \tilde{f}(u)\, du = 1.$$

Recall the calculation for the MCLA gradient originally stated in Equation (3.2). Both the numerator and denominator of Equation (3.2) are sample means. Since γ_1 is finite, the law of large numbers applies to the denominator of Equation (3.2) as $m \to \infty$:

$$\begin{aligned} \frac{1}{m}\sum_{k=1}^{m} \frac{f_\theta(u_k, y)}{\tilde{f}(u_k)} &\xrightarrow{a.s.} E_{\tilde{f}}\left[\frac{f_\theta(u, y)}{\tilde{f}(u)}\right] \\ &= \int \frac{f_\theta(u, y)}{\tilde{f}(u)} \tilde{f}(u)\, du \\ &= \int f_\theta(u, y)\, du \\ &= \gamma_1. \end{aligned} \tag{4.18}$$

Section 4.2.2 proves

$$E_{f_\theta}\left[\nabla \log f_\theta(u,y)|Y=y\right] = \nabla l(\theta|y)$$

and shows the existence of $\nabla l(\theta|y)$. By the law of large numbers, as $m \to \infty$,

$$\begin{aligned}
\frac{1}{m}\sum_{k=1}^{m}\left[\nabla \log f_\theta(u_k,y)\right]\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)} &\xrightarrow{a.s.} E_{\tilde{f}}\left[\left[\nabla \log f_\theta(u,y)\right]\frac{f_\theta(u,y)}{\tilde{f}(u)}\right] \\
&= \int \left[\nabla \log f_\theta(u,y)\right]\frac{f_\theta(u,y)}{\tilde{f}(u)}\tilde{f}(u)\,du \\
&= \gamma_1 \int \left[\nabla \log f_\theta(u,y)\right]\frac{f_\theta(u,y)}{\gamma_1}\,du \\
&= \gamma_1 E_{f_\theta}\left[\nabla \log f_\theta(u,y)|Y=y\right] \\
&= \gamma_1 \nabla l(\theta|y).
\end{aligned} \tag{4.19}$$

Then, by Slutsky's theorem,

$$\frac{\sum_{k=1}^{m}\left[\nabla \log f_\theta(u_k,y)\right]\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)}}{\sum_{k=1}^{m}\left(\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)}\right)} \xrightarrow{a.s.} \frac{\gamma_1 \nabla l(\theta|y)}{\gamma_1}.$$

More simply,

$$\nabla l_m(\theta|y) \longrightarrow \nabla l(\theta|y)$$

almost surely.

□

4.2.4 Central Limit Theorem for the Numerator of the MCLA Gradient

Lemma 4.2.4

The quantity

$$\frac{1}{m}\sum_{k=1}^{m}\left[\nabla \log f_\theta(u_k, y)\right]\frac{f_\theta(u_k, y)}{\tilde{f}(u_k)} \tag{4.20}$$

has a central limit theorem. That is,

$$\sqrt{m}\left(\frac{1}{m}\sum_{k=1}^{m}\left[\nabla \log f_\theta(u_k, y)\right]\frac{f_\theta(u_k, y)}{\tilde{f}(u_k)} - \gamma_1\,\nabla l(\theta|y)\right)$$

converges to a normal distribution with mean zero and finite variance.

Proof

Section 4.2.3 shows

$$E_{\tilde{f}}\left[\left[\nabla \log f_\theta(u,y)\right]\frac{f_\theta(u,y)}{\tilde{f}(u)}|Y=y\right]=\gamma_1\ \nabla l(\theta|y).$$

To prove the lemma in Section 4.2.4, we require

$$\mathrm{Var}\left(\left[\nabla \log f_\theta(u,y)\right]\frac{f_\theta(u,y)}{\tilde{f}(u)}\right)<\infty. \tag{4.21}$$

This is true if all of the following conditions are true for every j and for every t:

$$\mathrm{Var}\left(\left[\frac{\partial \log f_\beta(y|u)}{\partial \beta_j}\right]\frac{f_\theta(u,y)}{\tilde{f}(u)}\right)<\infty \tag{4.22}$$

$$\mathrm{Var}\left(\left[\frac{\partial \log f_\nu(u)}{\partial \nu_t}\right]\frac{f_\theta(u,y)}{\tilde{f}(u)}\right)<\infty \tag{4.23}$$

$$\left|\mathrm{Cov}\left(\left[\frac{\partial \log f_\beta(y|u)}{\partial \beta_j}\right]\frac{f_\theta(u,y)}{\tilde{f}(u)},\left[\frac{\partial \log f_\nu(u)}{\partial \nu_t}\right]\frac{f_\theta(u,y)}{\tilde{f}(u)}\right)\right|<\infty. \tag{4.24}$$

We start with Condition (4.22). Let j be given. Because u_k, $k=1,\ldots,m$ are drawn from a distribution with density $\tilde{f}(u_k)$,

$$\begin{aligned}
&\mathrm{Var}\left(\left[\frac{\partial \log f_\beta(y|u)}{\partial \beta_j}\right]\frac{f_\theta(u,y)}{\tilde{f}(u)}\right)\\
&<E_{\tilde{f}}\left[\left(\left[\frac{\partial \log f_\beta(y|u)}{\partial \beta_j}\right]\frac{f_\theta(u,y)}{\tilde{f}(u)}\right)^2\right]\\
&=\int\left[\frac{\partial \log f_\beta(y|u)}{\partial \beta_j}\right]^2\frac{\left[f_\theta(u,y)\right]^2}{\tilde{f}(u)}\,du\\
&=\int\left[\frac{\partial \log f_\beta(y|u)}{\partial \beta_j}\right]^2\frac{\left[f_\beta(y|u)\right]^2\left[f_\nu(u)\right]^2}{\tilde{f}(u)}\,du.
\end{aligned}$$

Since $\tilde{f}(u) \geq p_1 \grave{f}(u|0, D^*)$ and we require $p_1 > 0$,

$$\begin{aligned}
&\int \left[\frac{\partial \log f_\beta(y|u)}{\partial \beta_j}\right]^2 \frac{[f_\beta(y|u)]^2 [f_\nu(u)]^2}{\tilde{f}(u)}\, du \\
&\leq \int \left[\frac{\partial \log f_\beta(y|u)}{\partial \beta_j}\right]^2 \frac{[f_\beta(y|u)]^2 [f_\nu(u)]^2}{p_1 \grave{f}(u|0, D^*)}\, du.
\end{aligned}$$

Let b_1 and b_2 be constants with respect to u such that

$$\grave{f}(u|0, D^*) = b_1 \left[1 + u^T (D^*)^{-1} u/\zeta\right]^{(\zeta+q)/2}$$

and

$$f_\nu(u) = \sqrt{b_2} \exp\left(-u^T D^{-1} u\right).$$

Entering the densities for $\grave{f}(u|0, D^*)$ and $f_\nu(u)$ yields

$$\begin{aligned}
&\int \left[\frac{\partial \log f_\beta(y|u)}{\partial \beta_j}\right]^2 \frac{[f_\beta(y|u)]^2 [f_\nu(u)]^2}{p_1 \grave{f}(u|0, D^*)}\, du \\
&= \frac{b_2}{b_1} \int \left[\frac{\partial \log f_\beta(y|u)}{\partial \beta_j}\right]^2 \frac{[f_\beta(y|u)]^2 \exp(-u^T D^{-1} u)}{[1 + u^T (D^*)^{-1} u/\zeta]^{-(\zeta+q)/2}}\, du \\
&= \frac{b_2}{b_1} \int \left[\frac{\partial \log f_\beta(y|u)}{\partial \beta_j}\right]^2 \frac{[f_\beta(y|u)]^2 \left[1 + u^T (D^*)^{-1} u/\zeta\right]^{(\zeta+q)/2}}{\exp(u^T D^{-1} u)}\, du.
\end{aligned}$$

Since $f_\beta(y|u)$ is an exponential family,

$$\begin{aligned}
&\frac{b_2}{b_1} \int \left[\frac{\partial \log f_\beta(y|u)}{\partial \beta_j}\right]^2 \frac{[f_\beta(y|u)]^2 \left[1 + u^T (D^*)^{-1} u/\zeta\right]^{(\zeta+q)/2}}{\exp(u^T D^{-1} u)}\, du \\
&= \frac{b_2}{b_1} \int \left[\frac{\partial \log f_\beta(y|u)}{\partial \beta_j}\right]^2 \frac{\exp(2y^T \eta) \left[1 + u^T (D^*)^{-1} u/\zeta\right]^{(\zeta+q)/2}}{\exp(2c(\eta)) \exp(u^T D^{-1} u)}\, du.
\end{aligned} \tag{4.25}$$

The partial derivative is the jth element of Equation (3.17). Letting $x_{\cdot j}$ indicate the

jth column of X, we can write

$$\frac{\partial \log f_\beta(y|u)}{\partial \beta_j} = x_{\cdot j}^T \left(y - c'(\eta)\right).$$

Substituting this into Equation (4.25) yields

$$\begin{aligned}
&\frac{b_2}{b_1} \int \left[\frac{\partial \log f_\beta(y|u)}{\partial \beta_j}\right]^2 \frac{\exp(2y^T\eta)\ \left[1 + u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(2c(\eta))\ \exp(u^T D^{-1} u)}\, du \\
&\qquad = \frac{b_2}{b_1} \int \left[x_{\cdot j}^T \left(y - c'(\eta)\right)\right]^2 \frac{\exp(2y^T\eta)\ \left[1 + u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(2c(\eta))\ \exp(u^T D^{-1} u)}\, du.
\end{aligned}$$

Because $c(\eta) > 0$ when elements of the response vector are conditionally Bernoulli or Poisson given the random effects, $0 \leq \exp(-2c(\eta)) \leq 1$. This implies

$$\begin{aligned}
&\frac{b_2}{b_1} \int \left[x_{\cdot j}^T \left(y - c'(\eta)\right)\right]^2 \frac{\exp(2y^T\eta)\ \left[1 + u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(2c(\eta))\ \exp(u^T D^{-1} u)}\, du \\
&\qquad \leq \frac{b_2}{b_1} \int \left[x_{\cdot j}^T \left(y - c'(\eta)\right)\right]^2 \frac{\exp(2y^T\eta)\ \left[1 + u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(u^T D^{-1} u)}\, du.
\end{aligned}$$

Writing out η gives

$$\begin{aligned}
&\frac{b_2}{b_1} \int \left[x_{\cdot j}^T \left(y - c'(\eta)\right)\right]^2 \frac{\exp(2y^T\eta)\ \left[1 + u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(u^T D^{-1} u)}\, du \\
&= \frac{b_2}{b_1} \exp(2y^T X\beta) \int \left[x_{\cdot j}^T \left(y - c'(\eta)\right)\right]^2 \frac{\exp(2y^T Zu)\ \left[1 + u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(u^T D^{-1} u)}\, du.
\end{aligned} \tag{4.26}$$

Next, we write out the sum and use the triangle inequality:

$$\begin{aligned}\left[x_{\cdot j}^T\left(y-c'(\eta)\right)\right]^2 &= \left[\sum_{i=1}^n x_{ij}(y_i - c_i'(\eta_i))\right]^2 \\ &\leq \left[\sum_{i=1}^n |x_{ij}|\ \ |y_i - c_i'(\eta_i)|\right]^2.\end{aligned} \tag{4.27}$$

We now consider the effect of y and $c'(\eta)$ on (4.26) for the Bernoulli case and Poisson case separately. When elements of the response vector are conditionally Bernoulli given the random effects, then elements of y and elements of $c'(\eta)$ are bounded between 0 and 1 and we can use a similar argument to that in Equation (4.7):

$$\begin{aligned}\left[\sum_{i=1}^n |x_{ij}|\ \ |y_i - c_i'(\eta_i)|\right]^2 &\leq \left[1\sum_{i=1}^n |x_{ij}|\right]^2 \\ &= \left[\sum_{i=1}^n |x_{ij}|\right]^2.\end{aligned} \tag{4.28}$$

Substituting this into Equation (4.26) and rearranging the terms of the integrand gives

$$\begin{aligned}&\frac{b_2}{b_1}\exp(2y^TX\beta)\int \left[x_{\cdot j}^T\left(y-c'(\eta)\right)\right]^2 \frac{\exp(2y^TZu)\ \left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(u^TD^{-1}u)}\,du \\ &\leq \frac{b_2}{b_1}\exp(2y^TX\beta)\left[\sum_{i=1}^n |x_{ij}|\right]^2 \int \frac{\exp(2y^TZu)\ \left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(u^TD^{-1}u)}\,du \\ &= \frac{b_2}{b_1}\exp(2y^TX\beta)\left[\sum_{i=1}^n |x_{ij}|\right]^2 \times \\ &\qquad \int \left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}\exp(-u^TD^{-1}u+2y^TZu)\,du.\end{aligned} \tag{4.29}$$

By Section 4.2.1, $\exp(-u^T D^{-1}u + 2y^T Zu)$ is proportional to a normal density. Therefore, this integral is proportional to the expectation of $\left[1 + u^T (D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}$ with respect to a normal density. This moment exists. Therefore, we have shown Condition (4.22) for the Bernoulli case.

Now we move on to the Poisson case for Equation (4.27).

$$\begin{aligned}\left[\sum_{i=1}^{n} x_{ij}(y_i - c_i(\eta_i))\right]^2 &= \left[\sum_{i=1}^{n} x_{ij}(y_i - \exp(\eta_i))\right]^2 \\ &= \left[\sum_{i=1}^{n} x_{ij}(y_i - \exp(x_i\beta + z_i u))\right]^2\end{aligned}$$

Substituting this into Equation (4.26) gives

$$\begin{aligned}&\frac{b_2}{b_1}\exp(2y^T X\beta)\int \left[x_{\cdot j}^T\left(y - c'(\eta)\right)\right]^2 \frac{\exp(2y^T Zu)\ \left[1 + u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(u^T D^{-1}u)}\, du \\ &= \frac{b_2}{b_1}\exp(2y^T X\beta)\times \\ &\qquad \int \left[\sum_{i=1}^{n} x_{ij}(y_i - \exp(x_i\beta + z_i u))\right]^2 \frac{\exp(2y^T Zu)\ \left[1 + u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(u^T D^{-1}u)}\, du.\end{aligned}$$

Now we rearrange the terms of the integrand:

$$\begin{aligned}
&\frac{b_2}{b_1}\exp(2y^TX\beta)\times \\
&\qquad\int\left[\sum_{i=1}^{n}x_{ij}(y_i-\exp(x_i\beta+z_iu))\right]^2\frac{\exp(2y^TZu)\ \left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(u^TD^{-1}u)}\,du \\
&=\frac{b_2}{b_1}\exp(2y^TX\beta)\times \\
&\qquad\int\left[\sum_{i=1}^{n}x_{ij}(y_i-\exp(x_i\beta+z_iu))\right]^2\frac{\left[\exp(y^TZu)\right]^2\ \left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\left[\exp(u^TD^{-1}u/2)\right]^2}\,du \\
&=\frac{b_2}{b_1}\exp(2y^TX\beta)\int\left[\sum_{i=1}^{n}-x_{ij}\exp\left(-u^TD^{-1}u/2+y^TZu+x_i\beta+z_iu\right)+\right. \\
&\qquad\left.+x_{ij}y_i\exp\left(-u^TD^{-1}u/2+y^TZu\right)\right]^2\left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}\,du.
\end{aligned}$$

Next, we use the triangle inequality:

$$\begin{aligned}
&\frac{b_2}{b_1}\exp(2y^TX\beta)\int\left[\sum_{i=1}^{n}-x_{ij}\exp\left(-u^TD^{-1}u/2+y^TZu+x_i\beta+z_iu\right)+\right. \\
&\qquad\left.+x_{ij}y_i\exp\left(-u^TD^{-1}u/2+y^TZu\right)\right]^2\left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}\,du \\
&\leq\frac{b_2}{b_1}\exp(2y^TX\beta)\int\left[\sum_{i=1}^{n}|x_{ij}|\exp\left(-u^TD^{-1}u/2+y^TZu+x_i\beta+z_iu\right)+\right. \\
&\qquad\left.+|x_{ij}y_i|\exp\left(-u^TD^{-1}u/2+y^TZu\right)\right]^2\left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}\,du.
\end{aligned}\tag{4.30}$$

We could expand the sum of squares and each term would be proportional to a normal density. Thus, the integral in Equation (4.30) represents the expectation of $\left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}$ with respect to a normal density. Therefore, the integral exists and we have shown Condition (4.22) for the Poisson case.

Therefore, Condition (4.22) is met whether the elements of the response vector are conditionally Bernoulli or conditionally Poisson given the random effects.

Next, we move on to showing Condition (4.23). We start with

$$\begin{aligned}\operatorname{Var}\left(\left[\frac{\partial \log f_\nu(u)}{\partial \nu_t}\right]\frac{f_\theta(u,y)}{\tilde{f}(u)}\right) &\leq E_{\tilde{f}}\left[\left(\frac{\partial \log f_\nu(u)}{\partial \nu_t}\frac{f_\theta(u,y)}{\tilde{f}(u)}\right)^2\right] \\ &= \int \left[\frac{\partial \log f_\nu(u)}{\partial \nu_t}\right]^2 \frac{[f_\theta(u,y)]^2}{\tilde{f}(u)}\,du \\ &= \int \left[\frac{\partial \log f_\nu(u)}{\partial \nu_t}\right]^2 \frac{[f_\beta(y|u)]^2\,[f_\nu(u)]^2}{\tilde{f}(u)}\,du.\end{aligned} \tag{4.31}$$

Using the definition of $\tilde{f}(u)$ and the requirement that $0 < p_1 \leq 1$, we see

$$\begin{aligned}&\int \left[\frac{\partial \log f_\nu(u)}{\partial \nu_t}\right]^2 \frac{[f_\beta(y|u)]^2\,[f_\nu(u)]^2}{\tilde{f}(u)}\,du \\ &\leq \int \left[\frac{\partial \log f_\nu(u)}{\partial \nu_t}\right]^2 \frac{[f_\beta(y|u)]^2\,[f_\nu(u)]^2}{p_1\grave{f}(u|0,D^*)}\,du \\ &\leq \int \left[\frac{\partial \log f_\nu(u)}{\partial \nu_t}\right]^2 \frac{[f_\beta(y|u)]^2\,[f_\nu(u)]^2}{\grave{f}(u|0,D^*)}\,du.\end{aligned}$$

Next, we use the fact that $f_\beta(y|u)$ is a density for an exponential family:

$$\begin{aligned}&\int \frac{\left[\frac{\partial}{\partial \nu_t}\log f_\nu(u)\right]^2 [f_\beta(y|u)]^2\,[f_\nu(u)]^2}{\grave{f}(u|0,D^*)}\,du \\ &\qquad = \int \frac{\left[\frac{\partial}{\partial \nu_t}\log f_\nu(u)\right]^2 \exp(2y^T(X\beta+Zu))\,[f_\nu(u)]^2}{\exp(2c(\eta))\grave{f}(u|0,D^*)}\,du.\end{aligned}$$

Next, we substitute in the densities for $\grave{f}(u|0, D^*)$ and $f_\nu(u)$:

$$\int \frac{\left[\frac{\partial}{\partial \nu_t} \log f_\nu(u)\right]^2 \exp(2y^T(X\beta + Zu)) \left[f_\nu(u)\right]^2}{\exp(2c(\eta))\grave{f}(u|0, D^*)} du$$
$$= \frac{b_2}{b_1} \int \frac{\left[\frac{\partial}{\partial \nu_t} \log f_\nu(u)\right]^2 \exp(2y^T(X\beta + Zu)) \left[1 + u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(2c(\eta)) \exp(u^T D^{-1}u)} du.$$

Entering the partial derivative yields

$$\frac{b_2}{b_1} \int \frac{\left[\frac{\partial}{\partial \nu_t} \log f_\nu(u)\right]^2 \exp(2y^T(X\beta + Zu)) \left[1 + u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(2c(\eta)) \exp(u^T D^{-1}u)} du$$
$$= \frac{b_2}{b_1} \int \frac{\left[-\frac{q_t}{2\nu_t} + \frac{u^T E_t u}{2\nu_t^2}\right]^2 \exp(2y^T(X\beta + Zu)) \left[1 + u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(2c(\eta)) \exp(u^T D^{-1}u)} du.$$

If we use the fact that both the Bernoulli and Poisson cumulant functions are strictly positive, we see

$$\frac{b_2}{b_1} \int \frac{\left[-\frac{q_t}{2\nu_t} + \frac{u^T E_t u}{2\nu_t^2}\right]^2 \exp(2y^T(X\beta + Zu)) \left[1 + u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(2c(\eta)) \exp(u^T D^{-1}u)} du$$
$$< \frac{b_2}{b_1} \int \frac{\left[-\frac{q_t}{2\nu_t} + \frac{u^T E_t u}{2\nu_t^2}\right]^2 \exp(2y^T(X\beta + Zu)) \left[1 + u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(u^T D^{-1}u)} du. \tag{4.32}$$

Rearranging the terms in the integrand, we see

$$\begin{aligned}
&\frac{b_2}{b_1}\int \frac{\left[-\frac{q_t}{2\nu_t}+\frac{u^TE_tu}{2\nu_t^2}\right]^2 \exp(2y^T(X\beta+Zu))\,\left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}}{\exp(u^TD^{-1}u)}\,du \\
&=\frac{b_2}{b_1}\exp(2y^TX\beta)\times \\
&\int\left[-\frac{q_t}{2\nu_t}+\frac{u^TE_tu}{2\nu_t^2}\right]^2\,\left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}\exp(-u^TD^{-1}u+2y^TZu)\,du.
\end{aligned} \tag{4.33}$$

We recognize that $\exp(-u^TD^{-1}u+2y^TZu)$ is proportional to a normal density by Section 4.2.1. Therefore, Equation (4.33) is proportional to the expectation of

$$\left[-\frac{q_t}{2\nu_t}+\frac{u^TE_tu}{2\nu_t^2}\right]^2\,\left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}$$

with respect to a normal distribution. Therefore, Equation (4.33) exists. This satisfies Condition (4.23).

By the Cauchy-Schwartz inequality, Conditions (4.22) and (4.23) imply Condition (4.24). Now we have shown Conditions (4.22), (4.23), and (4.24). This is enough to prove Equation (4.21). Therefore, we can conclude that

$$\frac{1}{m}\sum_{k=1}^{m}\left[\nabla\log f_\theta(u_k,y)\right]\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)} \tag{4.34}$$

has a central limit theorem.

□

4.2.5 Central Limit Theorem for the Denominator of the MCLA Gradient

Lemma 4.2.5

The quantity

$$\frac{1}{m}\sum_{k=1}^{m}\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)} \tag{4.35}$$

has a central limit theorem. That is,

$$\sqrt{m}\left[\frac{1}{m}\sum_{k=1}^{m}\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)}-\gamma_1\right]$$

converges to a normal distribution with mean 0 and finite variance.

Proof

Equation (4.18) shows

$$E_{\tilde{f}}\left[\frac{f_\theta(u,y)}{\tilde{f}(u)}\,|\,Y=y\right]=\gamma_1.$$

We now need proof that

$$\mathrm{Var}\left(\frac{f_\theta(u,y)}{\tilde{f}(u)}\right)<\infty. \tag{4.36}$$

We begin with

$$\begin{aligned}\operatorname{Var}\left(\frac{f_\theta(u,y)}{\tilde{f}(u)}\right) &< E_{\tilde{f}}\left[\left(\frac{f_\theta(u,y)}{\tilde{f}(u)}\right)^2\right] \\ &= \int \frac{f_\theta(u,y)^2}{\tilde{f}(u)}\,du \\ &= \int \frac{f_\beta(y|u)^2 f_\nu(u)^2}{\tilde{f}(u)}\,du.\end{aligned}$$

Next, we use the definition of $\tilde{f}(u)$ and the requirement that $0 < p_1 \leq 1$:

$$\begin{aligned}\int \frac{f_\beta(y|u)^2 f_\nu(u)^2}{\tilde{f}(u)}\,du &\leq \int \frac{f_\beta(y|u)^2 f_\nu(u)^2}{p_1 \grave{f}(u|0,D^*)}\,du \\ &\leq \int \frac{f_\beta(y|u)^2 f_\nu(u)^2}{\grave{f}(u|0,D^*)}\,du.\end{aligned}$$

Entering the densities for $\grave{f}(u|0,D^*)$ and $f_\nu(u)$ yields

$$\int \frac{f_\beta(y|u)^2 f_\nu(u)^2}{\grave{f}(u|0,D^*)}\,du = \frac{b_2}{b_1}\int \frac{\exp(-u^T D^{-1}u)\, f_\beta(y|u)^2}{[1+u^T(D^*)^{-1}u/\zeta]^{-(\zeta+q)/2}}\,du.$$

Because $f_\beta(y|u)$ is the density for an exponential family,

$$\begin{aligned}&\frac{b_2}{b_1}\int \frac{\exp(-u^T D^{-1}u)\, f_\beta(y|u)^2}{[1+u^T(D^*)^{-1}u/\zeta]^{-(\zeta+q)/2}}\,du \\ &\qquad = \frac{b_2}{b_1}\int \frac{\exp(-u^T D^{-1}u)\,\exp(2y^T\eta)}{[1+u^T(D^*)^{-1}u/\zeta]^{-(\zeta+q)/2}\,\exp(2c(\eta))}\,du\end{aligned}$$

Expanding η, we see

$$\frac{b_2}{b_1}\int \frac{\left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}\exp(2y^T\eta)}{\exp(u^TD^{-1}u)\,\exp(2c(\eta))}\,du$$
$$=\frac{b_2}{b_1}\exp\left(2y^TX\beta\right)\int \frac{\left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}\exp(2y^TZu)}{\exp(u^TD^{-1}u)\,\exp(2c(\eta))}\,du.$$

Now we use the fact that both the Bernoulli and Poisson cumulant functions are strictly positive:

$$\frac{b_2}{b_1}\exp\left(2y^TX\beta\right)\int \frac{\left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}\exp(2y^TZu)}{\exp(u^TD^{-1}u)\,\exp(2c(\eta))}\,du$$
$$\leq\frac{b_2}{b_1}\exp\left(2y^TX\beta\right)\int \frac{\left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}\exp(2y^TZu)}{\exp(u^TD^{-1}u)}\,du.$$

Rearranging the terms of the integrand, we see

$$\frac{b_2}{b_1}\exp\left(2y^TX\beta\right)\int \frac{\left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}\exp(2y^TZu)}{\exp(u^TD^{-1}u)}\,du$$
$$=\frac{b_2}{b_1}\exp\left(2y^TX\beta\right)\int \left[1+u^T(D^*)^{-1}u/\zeta\right]^{(\zeta+q)/2}\exp(-u^TD^{-1}u+2y^TZu)\,du. \tag{4.37}$$

Since $\exp\left(-u^TD^{-1}u+2y^TZu\right)$ is proportional to a normal density by Section 4.2.1, Equation (4.37) is proportional to an expectation with respect to a normal distribution. Therefore, Equation (4.37) exists. This satisfies Condition (4.36) whether the elements of the response vector are conditionally Bernoulli or conditionally Poisson distributed given the random effects.

Therefore,

$$\frac{1}{m}\sum_{k=1}^{m}\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)} \tag{4.38}$$

has a central limit theorem.

□

4.2.6 Central Limit Theorem for the MCLA Gradient

Theorem 4.2.6

$\nabla l_m(\theta|y)$ has a central limit theorem. That is,

$$\sqrt{m}\left[\nabla l_m(\theta|y) - \nabla l(\theta|y)\right]$$

converges to a normal distribution with mean zero and finite variance.

Proof

First, we rewrite our expression:

$$\begin{aligned}
&\sqrt{m}\left[\nabla l_m(\theta|y) - \nabla l(\theta|y)\right] \\
&= \sqrt{m}\left[\frac{\frac{1}{m}\sum_{k=1}^{m}\left[\nabla \log f_\theta(u_k,y)\right]\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)}}{\frac{1}{m}\sum_{k=1}^{m}\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)}} - \nabla l(\theta|y)\right] \\
&= \sqrt{m}\left[\frac{\frac{1}{m}\sum_{k=1}^{m}\left[\nabla \log f_\theta(u_k,y)\right]\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)} - \nabla l(\theta|y)\,\frac{1}{m}\sum_{k=1}^{m}\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)}}{\frac{1}{m}\sum_{k=1}^{m}\frac{f_\theta(u_k,y)}{\tilde{f}(u_k)}}\right].
\end{aligned} \tag{4.39}$$

Using Section 4.2.4, we see the the numerator of Equation (4.39) converges to a normal distribution with mean 0 and variance

$$\operatorname{Var}\left(\left[\nabla \log f_\theta(u,y)\right]\frac{f_\theta(u,y)}{\tilde{f}(u)}\right).$$

By the law of large numbers, the denominator of Equation (4.39) converges almost surely to γ_1. Therefore, by Slutsky's theorem, Equation (4.39) converges to a normal distribution with mean zero and variance

$$\frac{1}{\gamma_1^2}\operatorname{Var}\left(\left[\nabla \log f_\theta(u,y)\right]\frac{f_\theta(u,y)}{\tilde{f}(u)}\right). \tag{4.40}$$

Because $\gamma_1 > 0$, the variance is finite.

□

If we wish, we can express the gradient's variance in terms of an integral:

$$\frac{1}{\gamma_1^2}\int \left[\nabla \log f_\theta(u,y)\right]\left[\nabla \log f_\theta(u,y)\right]^T \frac{f_\theta(u,y)^2}{\tilde{f}(u)}\,du. \tag{4.41}$$

Chapter 5

R package `glmm`

The R package `glmm` (Knudson, 2015) approximates the entire likelihood function for GLMMs with a canonical link. The importance sampling distribution used is listed in (4.1). `glmm` calculates and maximizes the MCLA to find MCMLEs for the fixed effects and variance components. Additionally, the value, gradient vector, and Hessian matrix of the MCLA are calculated at the MCMLEs. Observed Fisher information is estimated and used to calculate standard errors for the MCMLEs.

In this chapter, we instruct users on how to format their data to use `glmm` in Section 5.1, we discuss the analysis of the salamander data set in Section 5.2, we discuss the analysis of a data set with a Poisson response in Section 5.3, and we compare the `glmm` MCMLEs to the point estimates from other R packages in Section 5.4.

5.1 Formatting the Data

The following vectors can be used to fit a generalized linear mixed model using the `glmm` package. These vectors can be contained in a data frame, but they do not need to be.

1. A response vector. If your response is Poisson, then the entries in the response vector must be natural numbers. If your response is Bernoulli, then the entries

in the response vector must be 0 and 1. (For this version of `glmm`, these are the only two response types possible.)

2. At least one vector that will be used for defining the random effects' design matrix. For this version of `glmm`, the vector(s) should be class `factor`.

3. Vector(s) that will be used for defining the fixed effects' design matrix. The vector(s) can be of class `factor` or `numeric`.

The first two types of vectors described in the list are required. The last type is optional. That is, the minimum requirement to fit a `glmm` model is the response vector and one vector for defining the random effects' design matrix.

5.2 Analyzing the Salamander Data

Consider the salamander data described in Section 1.3. For your convenience, this data set is included in the `glmm` package. The variable `Mate` tells us whether the pair of salamanders mated: the value is 1 if they successfully mated and 0 if they did not. The variable `Cross` describes the type of female and male salamander. For example, `Cross = W/R` indicates a White Side female was crossed with a Rough Butt male. The variable `Female` contains the identification number of the female salamander, and the variable `Male` contains the identification number of the male salamander.

The first R command shown below gives us access to the `glmm` package and and all of its commands. The second line of code gives us access to the `salamander` data frame. The next three commands help us begin to understand the data. We have four variables: `Mate`, `Cross`, `Female`, and `Male`. The summary shows us `Mate` is numeric, `Cross` is a factor with four levels, `Female` is a factor, and `Male` is a factor.

```
library(glmm)
data(salamander)
names(salamander)
```

```
[1] "Mate"   "Cross"  "Female" "Male"
```

```
head(salamander)
```

```
  Mate Cross Female Male
1    1   R/R     10   10
2    1   R/R     11   14
3    1   R/R     12   11
4    1   R/R     13   13
5    1   R/R     14   12
6    1   R/W     15   28
```

```
summary(salamander)
```

```
      Mate       Cross        Female         Male
 Min.   :0.000   R/R:90   10     :  6   10     :  6
 1st Qu.:0.000   R/W:90   11     :  6   11     :  6
 Median :1.000   W/R:90   12     :  6   12     :  6
 Mean   :0.525   W/W:90   13     :  6   13     :  6
 3rd Qu.:1.000            14     :  6   14     :  6
 Max.   :1.000            15     :  6   15     :  6
                          (Other):324   (Other):324
```

5.2.1 Fitting the Model

We now fit Model A as described in Section 1.3.2 and originally proposed by Karim and Zeger (1992). In the following code, we fit the model using the `glmm` command and save the model under the name `sal`. Because `Mate` is our response, it is on the left of the $\sim$ symbol. We want to have a fixed effect for each of the four levels of `Cross`, so we type `Mate ~ 0 + Cross`. Because `Cross` is a factor, typing `Mate ~ Cross` would fit an equivalent model.

Next, the `random` list creates the design matrices for the random effects. Since we want two random effects for each cross (one from the female salamander and one from the male salamander), we type `list(~ 0 + Female, ~ 0 + Male)`. We include the `0` because we want our random effects to be centered at 0. Almost always, you will want your random effects to have mean 0.

Following the `random` list, the argument `varcomps.names` allows us to name the list of variance components. In the `random` list, we have placed the females first. Therefore, the order of the variance components names are first "F" and then "M."

Next, we specify the name of our data set. This is an optional argument. If the data set is not specified, `glmm` looks to the parent environment for the variables you have referenced.

After the name of the data set, we need to specify the type of the response. In the salamander mating example, the family is `bernoulli.glmm` because the response is binary. If your response is a count, then the family is `poisson.glmm`.

Next, we specify our Monte Carlo sample size `m`. The general rule is a larger Monte Carlo sample size results in a more accurate Monte Carlo likelihood approximation and more accurate MCMLEs. Ideally, you want the largest `m` that time allows. For this vignette, we have chosen a Monte Carlo sample size that allows for quick computation. If you are interested in accuracy in the resulting estimates for the salamander model, we suggest a larger Monte Carlo sample size.

We put these function arguments together in the following commands. We set the seed so that we can have reproducible results. In other words, if you set your seed to the same number and type the exact command listed below, your results should be identical to those listed here. Additionally, the `proc.time` commands have been used to give you an idea of how quickly the model can be fit. The times shown here are from fitting a model on an ultrabook that cost 500 USD in 2013.

```
set.seed(1234)
ptm<-proc.time()
sal <- glmm(Mate ~ 0 + Cross, random = list(~ 0 + Female,
~ 0 + Male), varcomps.names = c("F", "M"), data = salamander,
family.glmm = bernoulli.glmm, m = 10^4, debug = TRUE)
proc.time() - ptm
```

```
   user  system elapsed
 58.210   0.251  59.036
```

5.2.2 Adding Optional Arguments

Additional arguments may be added for more control over the model fit. These options are intended for advanced users.

Setting Variance Components Equal

By default, `glmm` assumes each variance component should be distinct. Suppose we want to set $\nu_F = \nu_M$. Then we would add the argument `varcomps.equal` to indicate the equality. Since the list of random effects has two entries and we want those entries to share a variance component, we would set `varcomps.equal = c(1,1)`. In

this scenario, we would only have one variance component, so we only need one entry in `varcomps.names`. Thus, the new command to fit this updated model with one variance component could be the following:

```
sal2 <- glmm(Mate ~ 0 + Cross, random = list(~ 0 + Female,
~ 0 + Male), varcomps.equal = c( 1, 1), varcomps.names =
c("Only Varcomp"), data = salamander, family.glmm =
bernoulli.glmm, m = 10^4, debug = TRUE)
```

As another example, suppose the list `random` has three entries, indicating three variance components ν_1, ν_2, ν_3. To set $\nu_1 = \nu_3$, we write `varcomps.equal = c(1,2,1)`. Thus, the shared variance component would be listed first in any output, and ν_2 would be listed second. The entries in the `varcomps.equal` vector must start at 1, then continue through the integers. The order of the names of the variance components listed in `varcomps.names` must correspond to the integers in `varcomps.equal`. In this problem, the names could be `varcomps.names = c("shared", "two")`.

Altering the Importance Sampling Distribution

The following default arguments can be adapted to alter the importance sampling distribution: `doPQL`, `p1`, `p2`, `p3`, and `zeta`.

By default, penalized quasi-likelihood estimates are used to form the importance sampling distribution for the generated random effects. To skip PQL, add the argument `doPQL=FALSE`. If PQL is skipped, then the importance sampling distribution uses arbitrary estimates of 0 for the random effects, 0 for the fixed effects, and 1 for the variance components. Sometimes the examples in the `glmm` documentation skip the PQL step so that the package can load more quickly. Most of the time, the model will fit more accurately and efficiently if PQL estimates are used in the importance sampling distribution.

The importance sampling distribution is a mixture of three distributions, as shown in Equation (4.1). By default, the mixture is evenly weighted, with each component's contribution set at 1/3. If you wish to change the mixture, you can alter `p1`, `p2`, and `p3` from the default of `p1 = 1/3`, `p2 = 1/3`, and `p3 = 1/3`. The only restrictions are that the three probabilities must sum to 1 and `p1` must be non-zero.

Recall the first component of the importance sampling distribution is a scaled multivariate t-distribution with `zeta` degrees of freedom. Therefore, another way to alter the importance sampling distribution is by changing `zeta` from its default of 5.

Adjusting Optimization Arguments

It may be useful to adjust the `trust` arguments `rmax` and `iterlim`. The argument `rmax` is the maximum allowed trust region radius. By `glmm` default, this is arbitrarily set to 1000. The smaller this number is, the longer the optimization time.

The argument `iterlim` must be a positive integer that limits the length of the optimization. If `iterlim` is too small, then the `trust` optimization will end before the MCMLA has been maximized. If `iterlim` is reached, then `trust` has not converged to the MCMLE. When the `summary` command is called, a warning will be printed telling the user that the parameter values are not MCMLEs, but `glmm` can be rerun starting at these outputted parameter values. To do this, use the `par.init` argument.

Starting at a Specified Parameter Value

Rather than using the PQL estimates, you can provide parameter values to `glmm` using the argument `par.init`. The `glmm` argument `par.init` is a vector that specifies the user-supplied values of the fixed effects and variance components. The parameters must be inputted in the order that `summary` outputs them, with fixed effects followed by variance components.

If `par.init` is provided, then PQL estimates will not be computed. The `par.init`

estimates will be used instead to form the importance sampling distribution. Then, `trust` will use `par.init` as the starting point for the optimization. This argument may be useful for very hard problems that require iteration.

5.2.3 Reading the Model Summary

The `summary` command displays

- the function call (to remind you of the model you fit).
- the link function.
- the fixed effect estimates, their standard errors (calculated using observed Fisher information), their `z value` test statistics (testing whether the coefficients are significantly different from zero), the corresponding p-values, and the R-standard significance stars (optional).
- the variance component estimates, their standard errors (calculated using observed Fisher information), their `z value` test statistics (testing whether the variance components are significantly greater than zero), the corresponding p-values, and the R-standard significance stars (optional).

The p-value for the fixed effects is calculated using a two-sided alternative hypothesis ($H_A : \beta \neq 0$) while the p-value for the variance components is calculated using a one-sided alternative hypothesis ($H_A : \nu > 0$) because variance components must be nonnegative.

To view the model summary, we use the `summary` command.

```
Call:
glmm(fixed = Mate ~ 0 + Cross, random = list(~0 + Female, ~0 + Male),
     varcomps.names = c("F", "M"), data = salamander,
     family.glmm = bernoulli.glmm, m = 10^4, debug = TRUE)

Link is: "logit (log odds)"

Fixed Effects:
         Estimate Std. Error z value Pr(>|z|)
CrossR/R   0.9560     0.3503   2.729  0.00634 **
CrossR/W   0.2805     0.3660   0.766  0.44347
CrossW/R  -1.8968     0.4223  -4.492 7.05e-06 ***
CrossW/W   0.9723     0.3580   2.716  0.00661 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Variance Components for Random Effects (P-values are one-tailed):
  Estimate Std. Error z value Pr(>|z|)/2
F   1.2878     0.4435   2.904    0.00184 **
M   1.0840     0.4131   2.624    0.00435 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The output tells us the type of cross affects the salamanders' odds of mating. Additionally, both the variance components are significantly different from zero and should be retained in the model.

The summary provides the estimates needed to write our model. First, we establish a little notation. Let π_i represent the probability of successful mating for salamander pair i. Let $I()$ be an indicator function, so that $I(\text{Cross=R/R})$ is 1 when a rough butt female is paired with a rough butt male and 0 otherwise. Let u_i^F represent the random effect from the female salamander in the ith pair. Let u_i^M represent the random effect from the male salamander in the ith pair. Using this notation, we write the model as follows.

$$\begin{aligned}
\log\left(\frac{\pi_i}{1-\pi_i}\right) &= 0.956 * I(\text{Cross=R/R}) + 0.2805 * I(\text{Cross=R/W}) \\
&\quad + -1.8968 * I(\text{Cross=W/R}) + 0.9723 * I(\text{Cross=W/W}) \\
&\quad + u_i^F + u_i^M \\
u_i^F &\overset{i.i.d.}{\sim} N(0, 1.288) \\
u_i^M &\overset{i.i.d.}{\sim} N(0, 1.084)
\end{aligned}$$

Recall that the Monte Carlo sample size `m` in the above model was chosen for convenience to save time. We can reduce the Monte Carlo error to yield more accurate MCMLEs by increasing the Monte Carlo sample size.

5.2.4 Isolating the Parameter Estimates

If we wish to extract the estimates for the fixed effect coefficients or the variance components, we use the commands `coef` and `varcomps`, respectively. These commands isolate the estimates that are shown in the summary (as displayed in section Section 5.2.3).

To extract the fixed effect coefficients, the only argument needed is the model. The commands `coef` and `coefficients` are interchangeable. We can type either of the following:

```
coef(sal)
```

```
  CrossR/R   CrossR/W   CrossW/R   CrossW/W
 0.9560113  0.2804932 -1.8968316  0.9722904
```

```
coefficients(sal)
```

```
  CrossR/R   CrossR/W   CrossW/R   CrossW/W
 0.9560113  0.2804932 -1.8968316  0.9722904
```

To extract the variance components, the only argument needed is the model.

```
varcomps(sal)
```

```
       F        M
1.287848 1.083975
```

To further isolate variance components or fixed effects, use indexing. The following demonstrates how to extract the last two fixed effects and the first variance component.

```
coef(sal)[c(3,4)]
```

```
  CrossW/R   CrossW/W
-1.8968316  0.9722904
```

```
varcomps(sal)[1]
```

```
       F
1.287848
```

5.2.5 Calculating Confidence Intervals

We can calculate confidence intervals for parameters using the `confint` command. (Prediction is not yet possible in this version of the package). If we wish to calculate 95% confidence intervals for all of our parameters, the only argument is the model name.

```
confint(sal)
```

```
              0.025      0.975
CrossR/R  0.9647676  1.2975071
CrossR/W  0.2896434  0.6373526
CrossW/R -1.8862749 -1.4851238
CrossW/W  0.9812412  1.3213731
F         1.2989347  1.7202402
M         1.0943035  1.4867789
```

The output is a matrix. Each row represents one parameter. The first column is the lower bound of the confidence interval, and the second column is the upper bound of the confidence interval.

If we wish to change the level of confidence from the default of 95%, we use the argument `level` and specify a number between 0 and 1. For example, the following produces 90% confidence intervals and 99% confidence intervals:

```
confint(sal,level=.9)
```

```
               0.05       0.95
CrossR/R  0.9735239  1.2887508
CrossR/W  0.2987937  0.6282024
```

```
CrossW/R -1.8757183 -1.4956804
CrossW/W  0.9901921  1.3124222
F         1.3100217  1.7091532
M         1.1046318  1.4764506
```

```
confint(sal,level=.99)
```

```
              0.005      0.995
CrossR/R  0.9577625  1.3045121
CrossR/W  0.2823232  0.6446728
CrossW/R -1.8947202 -1.4766785
CrossW/W  0.9740806  1.3285338
F         1.2900651  1.7291098
M         1.0860409  1.4950415
```

We can calculate 90% confidence intervals for the first and third fixed effects through indexing or by listing the names of the fixed effects:

```
confint(sal,level=.9,c(1,3))
```

```
               0.05      0.95
CrossR/R  0.9735239  1.288751
CrossW/R -1.8757183 -1.495680
```

```
confint(sal,level=.9,c("CrossR/R","CrossW/R"))
```

```
               0.05      0.95
CrossR/R  0.9735239  1.288751
CrossW/R -1.8757183 -1.495680
```

To calculate a 95 percent confidence interval for the variance component for the female salamanders, we can again either use indexing or list the name of the variable. There are four fixed effects so ν_F is the fifth parameter in this model. (Similarly, ν_M is the sixth parameter in this model).

```
confint(sal, c(5))
```

```
     0.025   0.975
F 1.298935 1.72024
```

```
confint(sal, c("F"))
```

```
     0.025   0.975
F 1.298935 1.72024
```

All confidence intervals are calculated using the observed Fisher information from the Monte Carlo likelihood approximation.

5.2.6 Estimating the Variance-Covariance Matrix

The variance-covariance matrix for the parameter estimates can be found using the `vcov` function. The only input is the model name.

```
(myvcov <- vcov(sal))
```

```
            CrossR/R     CrossR/W    CrossW/R     CrossW/W           F
CrossR/R  0.122676531  0.024012058  0.02086939 -0.010709735 -0.009392339
CrossR/W  0.024012058  0.133963096 -0.02178909  0.014823950  0.001149926
CrossW/R  0.020869390 -0.021789085  0.17830719  0.021167259 -0.027846771
CrossW/W -0.010709735  0.014823950  0.02116726  0.128187999  0.026420898
```

```
F        -0.009392339  0.001149926 -0.02784677  0.026420898  0.196674003
M        -0.002885214 -0.021014244 -0.01669001 -0.003236856 -0.033417152
                    M
CrossR/R -0.002885214
CrossR/W -0.021014244
CrossW/R -0.016690009
CrossW/W -0.003236856
F        -0.033417152
M         0.170678041
```

The variance-covariance matrix can be useful for some hypothesis testing. For example, suppose we want to test the hypotheses:

$$H_0 : \beta_{RR} - \beta_{WW} = 0$$
$$H_0 : \beta_{RR} - \beta_{WW} \neq 0.$$

The Wald test statistic is

$$\frac{\hat{\beta}_{RR} - \hat{\beta}_{WW} - 0}{\sqrt{\text{Var}\left(\hat{\beta}_{RR} - \hat{\beta}_{WW}\right)}} \sim N(0, 1).$$

To calculate

$$\text{Var}\left(\hat{\beta}_{RR} - \hat{\beta}_{WW}\right) = \text{Var}\left(\hat{\beta}_{RR}\right) + \text{Var}\left(\hat{\beta}_{WW}\right) - 2\,\text{Cov}\left(\hat{\beta}_{RR}, \hat{\beta}_{WW}\right)$$

we use the variances and covariances from the variance-covariance matrix:

```
myvar <- myvcov[1,1] + myvcov[4,4] - 2* myvcov[1,4]
SE <- sqrt(myvar)
SE
```

```
[1] 0.5218084
```

Then the test statistic and its associated p-value can be calculated:

```
test.stat <- (coef(sal)[1] - coef(sal)[4]) / SE
as.numeric(2 * pnorm(test.stat))
```

```
[1] 0.975112
```

Therefore, we do not have evidence to reject $H_0 : \beta_{RR} = \beta_{WW}$. The probability of two White Side salamanders mating is not significantly different from the probability of two Rough Butt salamanders mating.

Similarly, we could do a Wald-style hypothesis test to find the two variance components ν_F and ν_M are not significantly different.

5.2.7 Accessing Additional Output

The model produced by `glmm` has information that is not displayed by the `summary` command. The `names` command helps us see what we can access.

```
names(sal)
```

```
 [1] "beta"                "nu"                  "likelihood.value"
 [4] "likelihood.gradient" "likelihood.hessian"  "trust.converged"
 [7] "mod.mcml"            "fixedcall"           "randcall"
[10] "x"                   "y"                   "z"
```

```
[13] "family.glmm"         "call"                "varcomps.names"
[16] "varcomps.equal"      "debug"
```

The first two items are `beta` and `nu`. These are the MCMLEs for the fixed effects and variance components.

The third item is `likelihood.value`, the value of the MCLA evaluated at the MCMLEs. The fourth item is `likelihood.gradient`, the gradient vector of the MCLA evaluated at the MCMLEs. The fifth item is `likelihood.hessian`, the Hessian matrix of the MCLA evaluated at the MCMLEs.

Next is `trust.converged`, which tell us whether the `trust` function in the `trust` package converged to the optimizer of the MCLA. If trust did not converge, then the summary will print the following warning: "WARNING: the optimizer trust has not converged to the MCMLE. The following estimates are not maximum likelihood estimates, but they can be used in the argument par.init when rerunning glmm."

Items 7 through 16 relate to the original function call. The list `mod.mcml` contains the model matrix for the fixed effects, a list of model matrices for the random effects, and the response vector. These are also displayed in `x`, `z`, and `y`, respectively. Then, the call (the original formula representations of the fixed and random effects) are contained in `fixedcall`, `randcall`, and `call`.

The last argument is `debug`. If the model was fit with the default `debug = FALSE`, then this argument is just `FALSE`. If the model was fit with `debug = TRUE`, then `debug` contains a list of output for advanced users and programmers. In particular, this contains the matrix of random effects generated from the importance sampling distribution.

5.3 Analyzing the Radish Data

We now fit a model with a Poisson response. The data in this example are a subset of the data collected by Ridley and Ellstrand (2010). The scientists were interested in whether non-native radishes had adapted to the climate they had been growing in for the last 150 years. In other words, they wanted to compare two types of radish to see whether each type would grow just as well in their own climate as they would in the other climate.

In this dataset, the response is the number of radish flowers. This is assumed to have a Poisson distribution. `Site` is a categorical variable with two categories representing the two sites where plants were grown. The variable `Region` is categorical with two categories representing the two places in California from which the plants were taken. The variable `Pop` is a categorical variable representing the population of radish, and `Pop` is nested in `Region`. The variable `Block` is a categorical blocking variable nested in `Site`. Following the example of Ridley and Ellstrand (2010), `Block` and `Pop` are random while `Site` and `Region` are fixed. The scientists were interested in the interaction between `Site` and `Region`, since that would indicate that radishes grow better in the area they have been grown during recent history.

We load the data and fit the model using `glmm`:

```
library(glmm)
load("radish2.rda")


set.seed(1234)
mod<-glmm(resp~Site*Region,random=list(~0+Block,~0+Pop),
family.glmm="poisson.glmm",varcomps.names =c("block","pop"),
m=10^4,debug=TRUE,data=radish2)
summary(mod)
```

```
Call:
glmm(fixed = resp ~ Site * Region, random = list(~0 + Block,
    ~0 + Pop), varcomps.names = c("block", "pop"), data = radish2,
    family.glmm = "poisson.glmm", m = 10^4, debug = TRUE)


Link is: "log"

Fixed Effects:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)           5.911667   0.006972  847.92   <2e-16 ***
SiteRiverside         0.268346   0.009247   29.02   <2e-16 ***
RegionS              -0.423975   0.010583  -40.06   <2e-16 ***
SiteRiverside:RegionS 0.507595   0.012190   41.64   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


Variance Components for Random Effects (P-values are one-tailed):
      Estimate Std. Error z value Pr(>|z|)/2
block 0.514197   0.230077   2.235     0.0127 *
pop   0.011086   0.006607   1.678     0.0467 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Notably, the Site-Region interaction is statistically significant, which indicates local adaptation.

5.4 Comparing the Results

The following sections compare the MCMLEs from `glmm` (produced in Section 5.2 and Section 5.3) to point estimates from other methods.

5.4.1 Salamander Results

We used `glmm` (Knudson, 2015) to fit Model A Karim and Zeger's (1992) with a Monte Carlo sample size of $m = 10^5$. The resulting MCMLEs are listed in the following table along with Booth and Hobert's (1999) MCEM point estimates and the PQL-based point estimates from `lme4` (Bates et al., 2014) for comparison.

	$\hat{\beta}_{RR}$	$\hat{\beta}_{RW}$	$\hat{\beta}_{WR}$	$\hat{\beta}_{WW}$	$\hat{\nu}_F$	$\hat{\nu}_M$
Knudson (2015) (`glmm`)	1.03	.34	-1.94	1.00	1.36	1.23
Booth and Hobert (1999) (MCEM)	1.03	.32	-1.95	.99	1.4	1.25
Bates et al. (2014) (`lme4`)	1.01	.31	-1.89	.99	1.17	1.04

We can see that the results produced by Knudson (2015) match those produced by Booth and Hobert (1999). Additionally, the Booth and Hobert (1999) point estimates and the Knudson (2015) point estimates were checked using Markov chain Monte Carlo. These checks confirmed that the sets of point estimates are MLEs.

The fixed effect estimates from Bates et al. (2014) match the Knudson (2015) and Booth and Hobert (1999) results, but the `lme4` variance component estimates are smaller than those found by Knudson (2015) and Booth and Hobert (1999). This is consistent with research that has found a downward bias in PQL variance component estimates (Breslow and Lin, 1995; Lin and Breslow, 1996).

5.4.2 Radish Results

We used `glmm` (Knudson, 2015) to fit the model described in Section 5.3. The resulting MCMLEs are listed in the first row. The second row contains the point estimates from `lme4`. The first four columns represent estimates of the fixed effects. The last two columns represent estimates of the variance components.

	Intercept	SiteRiverside	RegionS	Site:Region	Block	Pop
`glmm`	5.91	.27	−.42	.51	.51	.011
`lme4`	5.92	.26	−.41	.51	.50	.012

The similarities between these sets of estimates indicate `lme4` works well. The `lme4` package appears to perform better for Poisson-distributed responses than for Bernoulli-distributed responses.

References

Bates, D., Maechler, M., Bolker, B., and Walker, S. (2014). *lme4: Linear mixed-effects models using Eigen and S4*. R package version 1.1-6.

Booth, J. G. and Hobert, J. P. (1999). Maximizing generalized linear mixed model likelihoods with an automated Monte Carlo EM algorithm. *Journal of the Royal Statistical Society, Series B*, 61:265–285.

Breslow, N. (1993). Whither PQL? (preprint). *UW Biostatistics Working Paper Series*, 192:ii–xix.

Breslow, N. and Clayton, D. (1993). Approximate inference in generalized linear mixed models. *Journal of the American Statistical Association*, 88:9–25.

Breslow, N. and Lin, X. (1995). Bias correction in generalized linear mixed models with a single component of dispersion. *Biometrika*, 82:81–91.

Dempster, A., Laird, N., and Rubin, D. (1977). Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39:1–38.

Ferguson, T. S. (1996). *A course in large sample theory*. Chapman and Hall/CRC, New York.

Gelfand, A. and Carlin, B. (1993). Maximum-likelihood estimation for constrained- or missing-data models. *Canadian Journal of Statistics*, 21:303–311.

Geyer, C. (1990). *Likelihood and Exponential Families*. PhD thesis, University of Washington. http://purl.umn.edu/56330.

Geyer, C. J. (1994). On the convergence of Monte Carlo maximum likelihood calculations. *Journal of the Royal Statistical Society, Series B*, 61:261–274.

Geyer, C. J. (2013). Asymptotics of maximum likelihood without the LLN or CLT or sample size going to infinity. In Jones, G. L. and Shen, X., editors, *Advances in Modern Statistical Theory and Applications: A Festschrift in honor of Morris L. Eaton*, volume 10, pages 1–24. Institute of Mathematical Statistics, Hayward, CA.

Geyer, C. J. and Thompson, E. (1992). Constrained Monte Carlo maximum likelihood for dependent data. *Journal of the Royal Statistical Society, Series B*, 54:657–699.

Karim, M. and Zeger, S. (1992). Generalized linear models with random effects; salamander mating revisited. *Biometrics*, 48:631–644.

Knudson, C. (2015). *glmm: Generalized Linear Mixed Models via Monte Carlo Likelihood Approximation*. R package version 1.0.2.

Lin, X. and Breslow, N. (1996). Bias correction in generalized linear mixed models with multiple components of dispersion. *Journal of the American Statistical Association*, 91:1007–1016.

Louis, T. A. (1982). Finding the observed information matrix when using the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 44:226–233.

McCullagh, P. and Nelder, J. (1989). *Generalized Linear Models*. Chapman and Hall/CRC.

McCulloch, C. (1997). Maximum likelihood algorithms for generalized linear mixed models. *Journal of the American Statistical Association*, 92:162–170.

McCulloch, C. and Searle, S. (2001). *Generalized, Linear, and Mixed Models*. John Wiley and Sons, New York.

Ridley, C. and Ellstrand, N. (2010). Rapid evolution of morphology and adaptive life history in the invasive California wild radish (Raphanus sativus) and the implications for management. *Evolutionary Applications*, 3(1):64–76.

Schall, R. (1991). Estimation in generalized linear mixed models with random effects. *Biometrika*, 78:719–727.

Stiratelli, R., Laird, N., and Ware, J. H. (1984). Random-effects models for serial observations with binary response. *Biometrics*, 40:961–971.

Sung, Y. J. (2003). *Monte Carlo likelihood inference for missing data models*. PhD thesis, University of Minnesota.

Sung, Y. J. and Geyer, C. J. (2007). Monte Carlo likelihood inference for missing data models. *Annals of Statistics*, 35:990–1011.

Thompson, E. and Guo, S. (1991). Evaluation of likelihood ratios for complex genetic models. *IMA Journal of Mathematics Applied in Medical Biology*, 8:149–169.

Venables, W. N. and Ripley, B. D. (2002). *Modern Applied Statistics with S*. Springer, New York, fourth edition. ISBN 0-387-95457-0.

Wei, G. C. G. and Tanner, M. A. (1990). A Monte Carlo implementation of the EM algorithm and the poor man's data augmentation algorithms. *Journal of the American Statistical Association*, 85:699–704.

Wolfinger, R. and O'Connell, M. (1993). Generalized linear mixed models: a pseudo-likelihood approach. *Journal of Statistical Computation and Simulation*, 4:233–243.