A Monte Carlo Likelihood Approximation for Generalized Linear Mixed Models

Bernardo B. de Andrade*
Departamento de Economia
Universidade de Brasília
Brasília - DF 70910-900 Brazil

Charles J. Geyer
School of Statistics
University of Minnesota
Minneapolis, MN 55455 USA

February 19, 2010

Abstract

We introduce the method of Monte Carlo likelihood approximation for the estimation of general missing-data models. An *umbrella sampling* scheme is proposed for the estimation of the likelihood in regions other than the neighborhood of the maximum likelihood estimate thus generalizing previous Monte Carlo methods. A version of Monte Carlo stochastic approximation is also proposed as a fast optimization routine along with a discussion of related Monte Carlo based methods.

Keywords: Monte Carlo, Stochastic approximation, Likelihood approximation, Umbrella sampling

*Corresponding author: Tel. +55(61) 3107-6607 Fax +55(61) 3340-2311, E-mail bbandrade@unb.br

1 Introduction

In this paper we discuss two computationally intensive methods for the estimation of hierarchical models using *Markov chain Monte Carlo* (MCMC) schemes: (i) *Monte Carlo stochastic approximation* (MCSA) and (ii) *Monte Carlo likelihood approximation* (MCLA).

Hierarchical models may arise in the statistical analyses of panel data (when random effects are used) and in state space models. Estimation in such models become especially hard with limited dependent variables or with nonlinear modelling. General missing-data models produce intractable likelihoods in the form of high-dimensional integrals,

$$\ell_{\mathbf{y}}(\boldsymbol{\theta}) = \log \int f_{\boldsymbol{\theta}_1}(\mathbf{y}|\mathbf{u}) g_{\boldsymbol{\theta}_2}(\mathbf{u}) \, \mathrm{d}\mathbf{u}, \tag{1.1}$$

hence the need for Monte Carlo integration. Here $\boldsymbol{\theta} = (\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$, $\mathbf{y}$ is the observed data with conditional density $f_{\boldsymbol{\theta}_1}(\mathbf{y}|\mathbf{u})$ and $\mathbf{u}$ is the vector of unobserved data with marginal $g_{\boldsymbol{\theta}_2}(\mathbf{u})$. Typically both $f_{\boldsymbol{\theta}_1}$ and $g_{\boldsymbol{\theta}_2}$ are in exponential families. A simple example of a potentially intractable hierarchical model follows.

1.1 Panel Data with Random Effects and Binary Response

Consider data on N firms collected at T time points. Let the response for the i-th firm, $\mathbf{y}_i$, be binary and its mean $\mathbf{p}_i = \Pr(\mathbf{y}_i = 1|\mathbf{u})$ be linked to k covariates and the random effects $\mathbf{u}$ by

$$\operatorname{logit}(\mathbf{p}_i) = \mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\mathbf{u}_i. \tag{1.2}$$

The $\mathbf{u}_i$'s are independent zero-mean with variance $\boldsymbol{\Sigma}_i(q \times q)$; $\mathbf{p}_i$ and $\mathbf{y}_i$ are $T \times 1$; $\mathbf{X}_i$ is $T \times k$ and $\mathbf{Z}_i$ is $T \times q$, $i = 1, \ldots, N$. We assume strict exogeneity throughout. A probit link, instead of logit, could also be adopted as in the analyses of Guilkey and Murphy (1993). Here $\boldsymbol{\theta}$ comprises $\boldsymbol{\beta}_{k\times 1}$ and all parameters characterizing $\boldsymbol{\Sigma}_1$, …, $\boldsymbol{\Sigma}_N$. The integral in (1.1) can be very complicated if the $\boldsymbol{\Sigma}$'s are accounting for spatio-temporal correlation and even numerically tractable if $\boldsymbol{\Sigma}_i = \sigma^2\mathbf{I}_q$, $\forall i$, with the $\mathbf{u}_i$'s normally distributed. For a general statistical account of random-effect models we refer to McCulloch et al. (2008) and Wooldridge (1999).

Any model with limited dependent response and random efects would be an example of hierarchical structure for which the Monte Carlo methods discussed here would apply. Other instances of hierarchical modelling with their own estimation challenges are latent-data, empirical Bayes and state space time series. In particular, non-Gaussian state space models and stochatisc volatility models have provided an active area of research for Monte Carlo methodology (Durbin and Koopman, 2000; Jungbacker and Koopman, 2007; Chib et al., 2002).

1.2 Likelihood Approximation

The key to MCLA is to replace (1.1) by a Monte Carlo approximation, $\ell_{n,\mathbf{y}}(\boldsymbol{\theta})$, where n represents the simulation size used. All statistical inference can then be carried out relative to $\ell_{n,\mathbf{y}}$. For instance, in order to estimate $\boldsymbol{\theta}$ we define the Monte Carlo MLE (MCMLE) by

$$\widehat{\boldsymbol{\theta}}_n = \arg\max_{\Theta} \ell_{n,\mathbf{y}}(\boldsymbol{\theta}). \tag{1.3}$$

Monte Carlo versions of the score, Fisher information, likelihood ratios and profile likelihoods can be similarly defined, something other methods such as MCSA and Monte Carlo EM (MCEM), are not designed to do.

Guilkey and Murphy (1993) pointed to difficulties in using Hermite integration to obtain (1.1) for a random-effects probit model and a Monte Carlo alternative was proposed. Other particular versions of MCLA known as Monte Carlo maximum likelihood (MCML) and also simulated maximum likelihood (SML), have been proposed (Pinheiro and Bates, 1995; McCulloch, 1997; Kuk and Cheng, 1999; Kuk, 2003; Booth and Hobert, 1999; Jank and Booth, 2003; Sung and Geyer, 2007) and some of these authors have used a random-effect logit model to illustrate their Monte Carlo methods. Much of their focus has been on approximating the MLE whereas our proposal in this paper aims at generalizing the approximation to regions far from the peak. We will also consider a logit-normal model to illustrate MCLA.

Maximum likelihood estimation based on $\ell_{\mathbf{y}}(\boldsymbol{\theta})$ typically requires us to handle the derivatives $\nabla \ell_{\mathbf{y}}(\boldsymbol{\theta})$ and $\nabla^2 \ell_{\mathbf{y}}(\boldsymbol{\theta})$ which often can only be expressed in terms of high dimensional integrals.

In Section 2 we propose MCSA as an auxiliary method for MCLA which is introduced in Section 3. Though the presentation of Section 3 may suggest that we are only interested in approximating the MLE, the simple method proposed therein not only provides direct ways of estimating other features of the likelihood but also it can be easily generalized. A generalization based on the umbrella sampling idea of Torrie and Valleau (1977) is then given in Section 5.

Stochastic approximation (Younes, 1988; Moyeed and Baddeley, 1991), MCSA, MCEM (Wei and Tanner, 1990), and Monte Carlo Newton-Raphson (MCNR) (Penttinen, 1984) are methods for maximum likelihood which do not provide approximations of the observed data likelihood. MCEM and MCNR are described in Section 4 to illustrate their relation to MCLA. Importance sampling formulas are recurrent in these methods and we derive them in Section 3.2.

We end this introduction by describing a toy random-intercept model and some of its likelihood features that we would like to mimic by Monte Carlo methods.

1.3 A Random-Effect Model

Hereafter, unless the context explicitly requires a distinction to be made, we will suppress the bold face letter and will represent scalars and vectors alike.

Let y be observed binary data on N independent clusters of T observations each, where $\Pr\{y_{ij} = 1|u\} = p_{ij}$ and a random-intercept model for the logit is assumed,

$$\operatorname{logit}(p_{ij}) \equiv \log\left(\frac{p_{ij}}{1 - p_{ij}}\right) = \beta x_j + u_i, \tag{1.4}$$

with $u_1, \ldots, u_N \sim$ iid $N(0, \sigma^2)$; $i = 1, \ldots, N$ and $j = 1, \ldots, T$. The parameter of interest is $\theta = (\beta, \sigma)$ and the MLE can be obtained with arbitrary precision by routine numerical integration and optimization procedures. Using the data from Booth and Hobert (2003, Table 2) generated with $\theta_0 = (5, \sqrt{0.5})$, $T = 15$, $N = 10$ and $x_j = j/T$ we obtain the MLE $\hat{\theta} = (6.13, 1.33)$ and the inverse of observed Fisher information,

$$\mathcal{J}^{-1}(\hat{\theta}) = \begin{pmatrix} 1.80 & 0.42 \\ 0.42 & 0.36 \end{pmatrix}. \tag{1.5}$$

For later reference, Wald and likelihood confidence regions given respectively by

$$\left\{(\hat{\theta} - \theta)^\top \mathcal{J}(\hat{\theta})(\hat{\theta} - \theta) \leq \chi^2_\alpha(2)\right\} \quad \text{and} \quad \left\{2[\ell(\hat{\theta}) - \ell(\theta)] \leq \chi^2_\alpha(2)\right\} \tag{1.6}$$

are depicted as full lines in Figure 2. Non-simultaneous approximate 95%-confidence intervals, with standard errors from (1.5) are (3.50, 8.76) and (0.15, 2.51) for β and σ, respectively. Projecting the curves of the right plot of Figure 2 onto the two axes yield (simultaneous) T^2-intervals (Johnson and Wichern, 2002). Based on the Wald curves, the 95% T^2-intervals are (2.85, 9.42) and (-0.14, 2.80) for β and σ respectively and based on the likelihood contours, the intervals are (3.56, 10.34) and (0.08, 4.00).

We note that methods that only provide point estimates and standard errors (such as MCSA and MCEM) would not allow us to check for the agreement between Wald ellipses and likelihood contours, something MCLA is designed to provide.

2 Monte Carlo Stochastic Approximation

2.1 Introduction

MCLA and other Monte Carlo likelihood methods require the computation of second derivatives (with respect to θ), $\nabla^2 \log f_\theta(u, y)$, which can be demanding and of little gain when far from the solution.

An effective way to get near the MLE is to run a Markov chain with *nonstationary* transition probabilities having state $(u_i, \hat{\theta}_i)$ and a two-part update:

1. Generate u_{i+1} with a Markov update having invariant distribution $f_{\hat{\theta}_i}(u|y)$;
2. Update θ with the deterministic step

$$\hat{\theta}_{i+1} = \hat{\theta}_i + a_i \nabla \log f_{\hat{\theta}_i}(u_{i+1}, y), \tag{2.1}$$

   where $\{a_i\}$ is a sequence of positive constants chosen in advance.

Here $\nabla \log f_{\hat{\theta}_i}(u, y)$ is the vector of first derivatives of $\log f_\theta(u, y)$ with respect to θ evaluated at $\hat{\theta}_i$.

Younes (1999) gives a complete theoretical treatment of MCSA but, to our knowledge, no practical guidelines for choosing the gain sequence $\{a_i\}$ are offered in the literature. We will rely on the conditions for the convergence of the classical Robbins-Munro *stochastic approximation* (SA) algorithm (Wasan, 1969), which are

$$a_i > 0, \quad a_i \to 0, \quad \sum_{i=0}^{\infty} a_i = \infty \quad \text{and} \quad \sum_{i=0}^{\infty} a_i^2 < \infty. \tag{2.2}$$

A common form of the gain sequence is

$$a_i = \frac{a}{(i + 1 + A)^\alpha}, \quad a > 0,\ 0.5 < \alpha \leq 1,\ A \geq 0. \tag{2.3}$$

Then, under appropriate conditions relating to the smoothness of $f(u, y)$,

$$n^{\alpha/2}(\hat{\theta}_n - \theta^*) \xrightarrow{\text{D}} N\left(0, \Sigma_{SA}\right), \tag{2.4}$$

where θ^* is the true maximizer and details on Σ_{SA} are given in Spall (1999, 2003).

The optimal rate of convergence in (2.4) obtained at $\alpha = 1$ is generally not recommended and in practical finite-sample problems a slower decay may be desirable in order to provide the algorithm with larger steps in iterations with large i. Even the non-convergent choice of $\alpha = 0$ has been found of practical use (Spall, 1999, 2003).

The constant A is commonly referred to as the *stability constant* and its choice is combined with that of a in order to control both early and late iterations. A large a may guarantee sizable steps at large i in which case $A > 0$ can stabilize otherwise bumpy early iterations. Setting $A = 0$ may prevent the iterations from offsetting the possibly desirable choice of a large a (a small a may bring stability to early iterations but at the expense of virtual immobility in later iterations). Another use of A is to offset the effect of starting the iterates too far from θ^* (if that much can be known in advance). As for n one may set usual stopping rules such as relative variation in estimates or negligible norm of gradient.

Polyak and Juditsky (1992) give the asymptotics of $\sqrt{n}(\bar{\theta}_n - \theta^*)$ where $\bar{\theta}_n$ $= (1/n) \sum_{i=1}^{n} \hat{\theta}_i$ with further conditions on (2.3). Thus one could use $\bar{\theta}_n$, or a trimmed mean, instead of the last iteration $\hat{\theta}_n$, as the final estimate. For our implementation of MCSA, as described in the next section, the results were very similar whether one used the last iterate, $\hat{\theta}_n$, iterate averaging, or 25% trimming and thus we only report results for $\hat{\theta}_n$ (Table 1).

2.2 MCSA on the Logit-Normal Model

We ran MCSA with $n = 5 \cdot 10^5$ starting at $\hat{\theta}_1 = (4, 2)$ and repeated the process 100 times to analyze the distribution of the estimates.

In the first step of the MCSA algorithm we generate u_{i+1} with a Metropolis scheme using normal proposal so that at the $(i + 1)$th iteration we have:

1.1 Propose $u \sim N\left(u_i, \tau^2 \mathbf{I}_N\right)$.

1.2 Let $u_{i+1} = u$ with probability $\rho = \dfrac{f_{\hat{\theta}_i}(u, y)}{f_{\hat{\theta}_i}(u_i, y)}$ and let $u_{i+1} = u_i$ with probability $1 - \rho$.

We have set $\tau = 0.5$ yielding an acceptance rate in step 1.1 slightly over 30% and similar results were obtained for τ in the range [0.3, 0.6]. For this Markov chain with *nonstationary* transition probabilities there is no theory for what would be a “reasonable” acceptance rate. We thus settled with the 30% mark (cf. Bedard, 2008) and the autocorrelation plots did not indicate any long range dependence. In the second step as given by (2.1) we used (2.3) with $a = 0.3$, $A = 100$ and $\alpha = 0.8$ after experimenting with different values and by visually inspecting the iterations.

Convergence as determined by the magnitude of $\|\hat{\theta}_i - \hat{\theta}_{i-1}\|$ was detected well before the pre-determined number of iterations of $5 \cdot 10^5$. We also ran the algorithm starting further from the MLE: this required smaller A (and slightly larger a) in order for the algorithm to oscillate "more freely" in the early iterations but not be affected much in later iterations. In fact, equally good results were obtained by fixing $a = 1$, $A = 0$ and simply adjusting α. We recommend starting with a constant sequence $a_i = a$, then try (2.3) with $a = 1$ and different values of A and after having a range of seemingly reasonable choices of A, choose a with some more trial and error. It is important to notice that we obtained similar results while experimenting with different starting values, gain sequences and different values of τ. In fact, varied compositions of a, A, α and τ provided similar MCSA estimates for this problem. The performance is a lot more sensitive to α and the starting value than to the other tuning parameters. We have not done anything to elaborate in terms of starting values other than using parameter estimates from fitting a model with fixed effects.

The results of these 100 MCSA trials are summarized in Table 1 and Figure 1. All but one trial finished close to the MLE. This exceptional run got "stuck" early in the iterations and was not able to return to the neighborhood of the MLE yielding a poor estimate $\hat{\theta}_n = (4.66, 0.09)$. Figure 1 depicts the results with this one run excluded for better visualization and the histograms indicate some skewness and very little bias.

We conclude by noting that MCSA is fairly simple to implement and the estimates were surprisingly accurate for this toy example. It thus seems to be an effcient auxiliary method to MCLA. Why one would like to have an idea of the magnitude of the MLE before implementing MCLA will be clear in the next sections.

3 Simple MCLA

3.1 Introduction

Consider a family of joint densities $\{f_\theta(u, y) : \theta \in \Theta\}$ and the likelihood of interest $f_\theta(y) = \int f_\theta(u, y)\, du$. We define the log-likelihood *relative to* f_ψ

as

$$\ell(\theta;\psi) = \log \frac{f_\theta(y)}{f_\psi(y)} = \log \mathrm{E}_\psi \left\{ \frac{f_\theta(U,Y)}{f_\psi(U,Y)} \;\middle|\; Y = y \right\}. \tag{3.1}$$

Assuming differentiation under the integral sign valid we define the gradient

$$\nabla \ell(\theta) = \frac{\mathrm{E}_\psi \left\{ \nabla \log f_\theta(U,Y) \frac{f_\theta(U,Y)}{f_\psi(U,Y)} \;\middle|\; Y = y \right\}}{\mathrm{E}_\psi \left\{ \frac{f_\theta(U,Y)}{f_\psi(U,Y)} \;\middle|\; Y = y \right\}} \tag{3.2}$$

and at $\psi = \theta$,

$$\nabla \ell(\theta) = \mathrm{E}_\theta \{ \nabla \log f_\theta(U,Y) \mid Y = y \}. \tag{3.3}$$

These expectations are typically intractable. They may be approximated by Monte Carlo with simulation of u_1, u_2, …, u_n yielding the following Monte Carlo version of (3.1) (Thompson and Guo, 1991),

$$\ell_n(\theta;\psi) = \log \left(\frac{1}{n} \sum_{i=1}^{n} \frac{f_\theta(u_i,y)}{f_\psi(u_i,y)} \right). \tag{3.4}$$

Suppose u_1, u_2, …, u_n is a Monte Carlo sample (usually a Markov chain) of the conditional $f_\psi(u|y)$. If the sampler is ergodic, then the law of large numbers holds for any integrable function g, as $n \to \infty$,

$$\frac{1}{n} \sum_{i=1}^{n} g(u_i,y) \xrightarrow{\text{a.s.}} \mathrm{E}_\psi \{ g(U,Y) \mid Y = y \}. \tag{3.5}$$

The asymptotics of (3.4) is detailed in Geyer (1994). Asymptotics (when both n and the sample size go to ∞) for the *ordinary* Monte Carlo is treated by Sung and Geyer (2007).

Let $\hat\theta_0$ be an initial estimate for the MLE (say, from MCSA). A *simple* MCLA scheme consists in using (3.4) with $\psi = \hat\theta_0$ and a Monte Carlo sample from $f_{\hat\theta_0}(\cdot|y)$. An *advanced* scheme will be treated in Section 5 where (3.4) is calculated relative to a *mixture* of distributions rather than to just a single distribution f_ψ. We will use an estimate $\hat\theta_0$ of the MLE for ψ but any point in the parameter space can be used. For instance, a point far out in the tails can be useful in p-value approximation.

We now present the *importance weights* which appear in several of the formulas used in MCLA. These formulas resemble importance sampling with *normalized* weights as introduced by Geweke (1989).

3.2 Importance Weights

The weights

$$w_{\theta;\psi}(u) = \frac{\dfrac{f_\theta(u,y)}{f_\psi(u,y)}}{\displaystyle\sum_{i=1}^{n} \frac{f_\theta(u_i,y)}{f_\psi(u_i,y)}} \tag{3.6}$$

define a probability distribution on $\{u_1, \ldots, u_n\}$, shifting the distribution of the u_i from ψ to θ in the sense that (assuming an ergodic sampling scheme) for any integrable function g, as $n \to \infty$,

$$\sum_{i=1}^{n} g(u_i,y) w_{\theta;\psi}(u_i) \xrightarrow{\text{a.s.}} \frac{\mathrm{E}_\psi\left\{ g(U,Y) \dfrac{f_\theta(U,Y)}{f_\psi(U,Y)} \;\middle|\; Y = y \right\}}{\mathrm{E}_\psi\left\{ \dfrac{f_\theta(U,Y)}{f_\psi(U,Y)} \;\middle|\; Y = y \right\}}. \tag{3.7}$$

But since the numerator is

$$\mathrm{E}_\psi\left\{ g(U,Y) \frac{f_\theta(U,Y)}{f_\psi(U,Y)} \;\middle|\; Y = y \right\} = \frac{f_\theta(y)}{f_\psi(y)} \mathrm{E}_\theta\bigl\{ g(U,Y) \bigm| Y = y \bigr\},$$

the limit in (3.7) is $\mathrm{E}_\theta\bigl\{ g(U,Y) \bigm| Y = y \bigr\}$. Thus we can rewrite (3.7) as

$$\sum_{i=1}^{n} g(u_i,y) w_{\theta;\psi}(u_i) \xrightarrow{\text{a.s.}} \mathrm{E}_\theta\bigl\{ g(U,Y) \bigm| Y = y \bigr\}. \tag{3.8}$$

For this reason we consider the left hand side of (3.8) the Monte Carlo estimate of the limit on the right and write it

$$\mathrm{E}_{n,\theta;\psi}\bigl\{ g(U,Y) \bigm| Y = y \bigr\} = \sum_{i=1}^{n} g(u_i,y) w_{\theta;\psi}(u_i) \tag{3.9}$$

with variance analog given by

$$\begin{aligned} \mathrm{Var}_{n,\theta;\psi}\bigl\{ g(U,Y) \bigm| Y = y \bigr\} &= \mathrm{E}_{n,\theta;\psi}\bigl\{ g(U,Y) g(U,Y)' \bigm| Y = y \bigr\} \\ &\quad - \mathrm{E}_{n,\theta;\psi}\bigl\{ g(U,Y) \bigm| Y = y \bigr\} \mathrm{E}_{n,\theta;\psi}\bigl\{ g(U,Y) \bigm| Y = y \bigr\}' \end{aligned} \tag{3.10}$$

3.3 Monte Carlo Likelihood Approximation

In wanting to approximate $\ell(\theta; \psi)$ in (3.1), θ, $\psi \in \Theta$, we will use the approximation (3.4). Thought of as a function of θ for fixed ψ, expression (3.1) *is*, for any practical purpose, the log-likelihood, merely shifted by $\log f_\psi(y)$. The only requirement for validity of (3.1) is that $f_\psi(u|y)$ dominates $f_\theta(u|y)$ (Geyer, 1994), a requirement that holds for the example of interest to us.

The Monte Carlo likelihood approach directly gives estimates of the score and observed Fisher information through differentiation of (3.4) yielding

$$\begin{aligned} \nabla \ell_n(\theta; \psi) &= \mathrm{E}_{n,\theta;\psi}\big\{\nabla \log f_\theta(U, y)\big\} \\ &= \sum_{i=1}^{n} \nabla \log f_\theta(u_i, y) w_{\theta,\psi}(u_i). \end{aligned} \tag{3.11}$$

and

$$\nabla^2 \ell_n(\theta; \psi) = \mathrm{E}_{n,\theta;\psi}\big\{\nabla^2 \log f_\theta(U, y)\big\} + \mathrm{Var}_{n,\theta;\psi}\big\{\nabla \log f_\theta(U, y)\big\} \tag{3.12}$$

where

$$\mathrm{E}_{n,\theta;\psi}\big\{\nabla^2 \log f_\theta(U, y)\big\} = \sum_{i=1}^{n} \nabla^2 \log f_\theta(u_i, y) w_{\theta,\psi}(u_i)$$

and

$$\begin{aligned} \mathrm{Var}_{n,\theta;\psi}\big\{\nabla \log f_\theta(U, y)\big\} = \mathrm{E}_{n,\theta;\psi}\big\{\big(\nabla \log f_\theta(U, y)\big)\big(\nabla \log f_\theta(U, y)\big)'\big\} \\ - \nabla \ell_n(\theta; \psi)(\nabla \ell_n(\theta; \psi))' \end{aligned} \tag{3.13}$$

These formulas allow us to do likelihood inference based on Fisher information by maximizing the Monte Carlo log-likelihood with either safeguarded Newton-Raphson or trust regions. They are reminiscent of a formula attributed to Louis (1982) in the EM context, although appearing earlier in Sundberg (1974). Our formulas, however, follow from direct differentiation of (3.4).

An important special case of the derivative formulas (3.11) and (3.13) occurs when $\theta = \psi$. Then all the importance weights (3.6) are $1/n$ and the $\mathrm{E}_{n,\theta;\theta}$ terms in these equations are plain averages (see Section 4). Thus we do not need importance sampling to calculate first and second derivatives of the log-likelihood *at the parameter point where the sample is generated.* The virtue of our importance sampling formulas (3.11) and (3.13) is that we can calculate the log-likelihood at other points without collecting a new sample.

If we can find a parameter point ψ near the MLE, then one large Monte Carlo sample is all we need to produce an approximation of the log-likelihood around its peak, with no further iteration necessary. MCSA can thus be a useful precursor to simple MCLA.

3.4 Variance Calculations

With one large Monte Carlo sample and the derivative formulas above we can calculate an *internal* variance estimate for $\hat{\theta}_n - \hat{\theta}$. Following Geyer (1994) we have the following asymptotic equivalence in distribution,

$$\hat{\theta}_n - \hat{\theta} \approx \left(-\nabla^2 \ell_{n,y}(\hat{\theta};\psi)\right)^{-1} \nabla \ell_{n,y}(\hat{\theta};\psi).$$

The right hand side is an asymptotically constant matrix multiplied by an asymptotically normal, mean zero, random vector. Thus the right hand side is asymptotically centered normal with variance estimated by

$$\left(-\nabla^2 \ell_{n,y}(\hat{\theta};\psi)\right)^{-1} \operatorname{Var}\left(\nabla \ell_{n,y}(\hat{\theta};\psi)\right) \left(-\nabla^2 \ell_{n,y}(\hat{\theta};\psi)\right)^{-1} \tag{3.14}$$

In the usual asymptotics of maximum likelihood, the middle term is Fisher information and it cancels out with one of the outer terms. Here it isn't and hence the three-term estimator (3.14) analogous to White's (1982) robust estimator under misspecification. One may think that we are deliberately "misspecifying" the likelihood by sampling at ψ.

3.5 Logit-Normal Example

Using the same logit-normal model and data of Section 1.3 we simulated the missing data u_1, u_2, …, u_n having equilibrium distribution $f_\psi(\cdot|y)$ via a (random walk) Metropolis algorithm. Our proposal distribution for the Metropolis algorithm was $N(u, \tau^2\mathbf{I})$ where u is the current state of the chain and τ was set at 0.5 yielding acceptance rates about 30%.

We have set $\psi = (6.15, 1.30)$ in the target distribution $f_\psi(u|y)$ of the Metropolis algorithm. This value of ψ was obtained from an MCSA run. The resulting MCMLE with $n = 10^5$ was $\hat{\theta}_n = (6.11, 1.33)$. For this run the Monte Carlo observed Fisher information (the negative of (3.12) at $\hat{\theta}_n$) was

$$\begin{aligned} \hat{J}_n \equiv J(\hat{\theta}_n) &= -\nabla^2 \ell_n(\hat{\theta}_n; h) \\ &= \begin{pmatrix} 0.78 & -0.90 \\ -0.90 & 3.55 \end{pmatrix}, \end{aligned} \tag{3.15}$$

with inverse (to be compared with (1.5))

$$\hat{J}_n^{-1} = \begin{pmatrix} 1.80 & 0.46 \\ 0.46 & 0.40 \end{pmatrix}.$$

In order to estimate the variance given by (3.14) we need to first estimate $V = \mathrm{Var}\left(\nabla \ell_n(\hat{\theta}; \psi)\right)$. Since we are using a Markov chain u_1, …, u_n (instead of an iid sequence), V cannot be simply estimated by raw sample variances. We use, instead, *overlapping batch means* but other methods (e.g. regeneration) could be employed. We have used batch length $b = 150$ which we considered reasonable by visual inspection of autocorrelation plots of the components of $\nabla \ell_n(\theta; \psi) = \nabla \log f_\theta(u_i, y) w_{\theta,\psi}(u_i)$. We thus obtained

$$\hat{V} = \frac{1}{10^4} \begin{pmatrix} 1.93 & 1.25 \\ 1.25 & 32.74 \end{pmatrix}.$$

And finally the "sandwich" estimate of variance given by (3.14) is

$$\begin{aligned} \hat{J}_n^{-1} \hat{V} \hat{J}_n^{-1} &= \begin{pmatrix} 1.80 & 0.46 \\ 0.46 & 0.40 \end{pmatrix} \frac{1}{10^4} \begin{pmatrix} 1.93 & 1.25 \\ 1.25 & 32.74 \end{pmatrix} \begin{pmatrix} 1.80 & 0.46 \\ 0.46 & 0.40 \end{pmatrix} \\ &= \frac{1}{10^4} \begin{pmatrix} 15.10 & 8.7 \\ 8.7 & 6.0 \end{pmatrix}. \end{aligned}$$

The Monte Carlo standard errors are the square root of the diagonal and under approximate normality the 95% confidence intervals for $\hat{\beta}$ and $\hat{\sigma}$ are (6.03, 6.20) and (1.28, 1.38), respectively.

Monte Carlo Wald regions based on $\hat{J}_n$ and true Wald ellipses are juxtaposed in Figure 2. We recall from Section 1 that the true Wald intervals are (3.98, 9.34) and (0.50, 3.17) for β and σ respectively. Their Monte Carlo counterparts centered at $\hat{\theta}_n$ with errors given by the diagonal of $\hat{J}_n^{-1}$ are (3.48, 8.74) and (0.10, 2.56).

Figure 2 also displays the shifted log-likelihood contours, $\log f_\theta(y) - \log f_\psi(y)$, superimposed by the Monte Carlo approximation. As expected, the goodness of the approximation is not uniform. It is better near the MLE. The advanced approach of next section is designed to improve on this by allowing good approximation in regions of lower density.

We repeated the simulation 100 times using $\psi_1 = (7.5, 2.0)$ (somewhat away from $\hat{\theta}$) and again with $\psi_2 = (6.15, 1.30)$ (close to $\hat{\theta}$). We kept $n = 10^5$ throughout. Figure 3 depicts the 100 MCMLEs thus obtained. The variability and bias of $\hat{\theta}_n$ are noticeably dependent on ψ. As expected, results are far better under ψ_2. We observe underestimation of both $\hat{\beta}$ and $\hat{\sigma}$ with ψ_1 although most runs still yielded good point estimates. The results for ψ_2 are noticeably more accurate and histograms (not shown) suggest approximate normality when ψ is near $\hat{\theta}$.

4 An Overview of Other MCMC Approaches

We now briefly describe and comment on other approaches to maximum likelihood estimation in hierarchical models. We also comment on the *the "acid test" of convergence* and give a brief digression on optimization theory.

4.1 Monte Carlo Newton-Raphson

From the MCLA point of view, MCNR is a hardly justifiable refusal to use importance sampling. When $\theta = \psi$, formulas (3.11) and (3.12) become simple averages,

$$\nabla \ell_n(\theta; \theta) = \frac{1}{n} \sum_{i=1}^{n} \nabla \log f_\theta(u_i, y), \tag{4.1a}$$

and

$$\begin{aligned}\nabla^2 \ell_n(\theta; \theta) = \frac{1}{n} \sum_{i=1}^{n} \nabla^2 \log f_\theta(u_i, y) \\ + \frac{1}{n} \sum_{i=1}^{n} \bigl(\nabla \log f_\theta(u_i, y)\bigr)\bigl(\nabla \log f_\theta(u_i, y)\bigr)' \\ - \left(\frac{1}{n} \sum_{i=1}^{n} \nabla \log f_\theta(u_i, y)\right)\left(\frac{1}{n} \sum_{i=1}^{n} \nabla \log f_\theta(u_i, y)\right)' \end{aligned} \tag{4.1b}$$

Having calculated (by Monte Carlo) first and second derivatives, we can also calculate the (Monte Carlo approximation to the) Newton-Raphson (NR) step

$$\hat{\delta}_n = -\bigl(\nabla^2 \ell_n(\theta; \theta)\bigr)^{-1} \nabla \ell_n(\theta; \theta) \tag{4.2}$$

the NR step itself being, of course,

$$\delta = -\bigl(\nabla^2 \ell(\theta; \theta)\bigr)^{-1} \nabla \ell(\theta; \theta) \tag{4.3}$$

that is, the exact NR step (if it could be calculated) would be from θ to $\theta + \delta$, and the MCNR step is from θ to $\theta + \hat{\delta}_n$.

MCNR is thus an approximation to what MCLA does when $\psi \approx \hat{\theta}$. When ψ is very close to θ, all the importance weights are nearly equal to one, and there is no appreciable difference between the two.

It is well known (as explained in Section 4.3) that NR can be very badly behaved when started far from the solution. It "explodes" to infinity, the computer crashing due to overflow, divide by zero, or something of the sort. This is the known behavior of *deterministic* NR. Monte Carlo doesn't make it worse (it couldn't be worse), but also doesn't make it any better. Since discrete exponential families, like the binomial and Poisson, often used in random-effect models, are a prime example of bad behavior of NR (in maximum likelihood for the binomial in the canonical parameterization, NR explodes to infinity when started far from the solution) it is hard to believe that MCNR is useful for the class of problems we are interested in.

4.1.1 The Acid Test of Convergence

As skeptical as one may be with MCNR when far from solution, we have no objections at all to using it when near the solution. As we have already admitted, MCLA does almost the same thing in this situation.

Since they do almost the same thing, they agree about convergence. There is only one good convergence criterion. Finding the MCMLE when the MCNR step (4.2) is negligible (zero to within Monte Carlo error). This is the acid test. In particular, MCSA and MCEM must be followed by at least one calculation of this "acid test".

4.2 Monte Carlo EM

Monte Carlo EM samples exactly the same distribution as MCLA and uses the samples in a similar way. It just moves the log inside the expectation. The deterministic maximizes

$$q(\theta; \psi) = \mathrm{E}_\psi \left\{ \log \frac{f_\theta(U, Y)}{f_\psi(U, Y)} \;\middle|\; Y = y \right\} \tag{4.4}$$

thought of as a function of θ for fixed ψ to move from the current iterate ψ of the EM algorithm to the next iterate.

Because of the conditional Jensen inequality and concavity of the log function

$$\mathrm{E}_\psi \left\{ \log \frac{f_\theta(U, Y)}{f_\psi(U, Y)} \;\middle|\; Y = y \right\} \le \log \mathrm{E}_\psi \left\{ \frac{f_\theta(U, Y)}{f_\psi(U, Y)} \;\middle|\; Y = y \right\} \tag{4.5}$$

which says $q(\theta; \psi) \le \ell(\theta; \psi)$ and this implies that a step from ψ to θ will increase ℓ if it increases q. The Monte Carlo approximation of the q function is, of course,

$$q_n(\theta; \psi) = \frac{1}{n} \sum_{i=1}^{n} \log \frac{f_\theta(u_i, y)}{f_\psi(u_i, y)} \tag{4.6}$$

Note that (3.4) and (4.6) use exactly the same samples u_1, u_2, $\ldots$, u_n exactly the same functions, exactly the same operations, just in slightly different order. It is no harder or easier to calculate one or the other. Of course, the two expressions (3.4) and (4.6) have rather different Monte Carlo errors and also estimate rather different things. If it works, (3.4) gives the log likelihood. So the only occasion for ever using (4.6) is when (3.4) happens to be a very bad approximation to (1.1).

But also note that the conditional Jensen inequality also applies to our weighted empirical approximations, because the importance weights (3.6) sum to one. Thus

$$q_n(\theta; \psi) \le \ell_n(\theta; \psi) \tag{4.7}$$

for all data y and all Monte Carlo samples u_1, u_2, $\ldots$, u_n Thus, however bad an approximation $\ell_n(\theta; \psi)$ may be, it can never be negative when $q_n(\theta, \psi)$ is positive.

Differentiating (4.6) we get

$$\nabla q_n(\theta; \psi) = \frac{1}{n} \sum_{i=1}^{n} \nabla \log \frac{f_\theta(u_i, y)}{f_\psi(u_i, y)} \tag{4.8}$$

Because this is a simple (no importance weights) average, it is obvious that (assuming an ergodic sampling scheme)

$$\nabla q_n(\theta; \psi) \xrightarrow{\text{a.s.}} \mathrm{E}_\psi\bigl\{\nabla \log f_\theta(U, Y) \bigm| Y = y\bigr\} \tag{4.9a}$$

or, assuming differentiability under the integral sign is possible,

$$\nabla q_n(\theta; \psi) \xrightarrow{\text{a.s.}} \nabla q(\theta; \psi)$$

In contrast, from our discussion of normalized importance sampling, it is clear that

$$\nabla \ell_n(\theta; \psi) \xrightarrow{\text{a.s.}} \mathrm{E}_\theta\bigl\{\nabla \log f_\theta(U, Y) \bigm| Y = y\bigr\} \tag{4.9b}$$

or, again assuming differentiability under the integral sign,

$$\nabla \ell_n(\theta; \psi) \xrightarrow{\text{a.s.}} \nabla \ell(\theta; \psi).$$

In (4.9b) by using importance sampling we get an expectation for parameter value θ (which can vary!) using a sample for parameter value ψ (which is fixed). In (4.9a) by not using importance sampling we get an expectation for parameter value ψ (which is fixed!). Thus a zero of $\nabla \ell_n(\theta; \psi)$ is a close approximation to the MLE and a zero of $\nabla q_n(\theta; \psi)$ isn't (although EM theory implies some progress toward the MLE).

Thus MCEM is unable to make use of importance sampling which are not difficult to calculate. They are functions of the joint density $f_\theta(u, y)$, which is assumed known in MCEM as in every other approach to hierarchical models.

The importance weights can get out of hand when θ is far from ψ, but once one gets ψ close to the MLE, there can be no sound reason for continuing to use MCEM.

4.3 A Digression on Optimization Theory

Optimization theory shows that NR may or may not be a good algorithm. When *near the solution* NR is optimal in a very strong sense. When *far from the solution* NR is problematic. The problems we are considering, involving discrete exponential families, are in the class of problems in which NR is troublesome when far from the solution.

Returning to behavior near the solution, the Dennis-Moré theorem makes a very strong statement about NR. Consider a continuously differentiable function F mapping $\mathbb{R}^d$ to $\mathbb{R}^d$, and suppose x^* is a zero of F, and the derivative $\nabla F(x^*)$ is non-singular. In the likelihood context, F is $\nabla \ell$, where ℓ is the log likelihood, so a zero of F is a "solution of the likelihood equations." And ∇F is $\nabla^2 \ell$, the negative of observed Fisher information.

Now consider any sequence $x_n \to x^*$. The idea is that x_n is the sequence of iterations of some algorithm, but it can be any convergent sequence whatsoever. We say the sequence is *superlinearly convergent* if

$$x_{n+1} - x^* = o(x_n - x^*).$$

This is the weakest condition a good algorithm can satisfy. Weakening to so-called *linear convergence*, meaning

$$x_{n+1} - x^* = O(x_n - x^*),$$

doesn't even guarantee convergence, only that x_n is a bounded sequence. So linear convergence doesn't imply anything. It is the slowest possible form of convergence. The Dennis-Moré theorem says that every superlinearly convergent algorithm is asymptotically equivalent to NR, which means, defining the NR step

$$y_{n+1} = x_n - \bigl(\nabla F(x_n)\bigr)^{-1} F(x_n),$$

that

$$x_{n+1} - x_n = y_{n+1} - x_n + o(y_{n+1} - x_n).$$

Although the theorem is usually stated this way (weakest conditions), the convergence is typically better than superlinear. Another theorem (Theore 6.2.1, Fletcher (1987)) says that if ∇F is Lipschitz, then NR is quadratically convergent, meaning

$$x_{n+1} - x^* = O(\|x_n - x^*\|^2).$$

Thus every superlinearly convergent algorithm for solving nonlinear equations (or for maximizing or minimizing a function) is asymptotically equivalent to NR. Among algorithms that are not superlinearly convergent are some much used in statistical model fitting, including EM, iteratively reweighted least squares, iterative proportional fitting, and Fisher scoring.

Of course, MCNR is not NR. Like all Monte Carlo methods its Monte Carlo error obeys the square root law, meaning it is $O_p(n^{-1/2})$, where we have just switched the meaning of n, from *iteration number* in the discussion of the Dennis-Moré theorem to *Monte Carlo sample size* in this O_p statement. When the "NR error" is equal to the Monte Carlo error, then more iterations are no help. One must increase the Monte Carlo sample size to get more precision.

A similar statement could be made about MCEM. Once the "EM error" is equal to the Monte Carlo error, then more iterations are no help. The difference is that the "EM error" is very slowly converging. So when combined with Monte Carlo error, it is simply impossible to detect convergence from MCEM iterations alone. In contrast, "NR error" is very rapidly (quadratically) converging. If it is about the same size as Monte Carlo error in one MCNR iteration, then it is negligible in the next. When using MCNR or MCLA for the final iterations only Monte Carlo error matters.

This is why we say the only test of convergence is the “acid test” of Section 4.1.1. The only good way to check whether MCEM converges is to switch to MCNR or MCLA and see whether it finds work still left to do. If one stays with MCEM, one may never be sure about convergence.

5 Umbrella Sampling

5.1 Introduction

It is clear from Section 3 that the simple version of MCLA will be of little use when ψ is far from the MLE unless, of course, interest lies in approximating the likelihood in a region far from the peak. This would be the case, for instance, in wanting to calculate a tail probability. In this section we generalize MCLA by introducing a Monte Carlo scheme for approximating the observed data likelihood based on the idea of *umbrella sampling* (Torrie and Valleau, 1977; Marinari and Parisi, 1992; Geyer and Thompson, 1992). The objective is to allow our approximation to be of use in regions other than the peak.

We simulate the missing data from a finite *mixture* (an “umbrella”) of distributions specified by unnormalized densities h_j, $j = 1, \ldots, m$,

$$h_{\text{mix}}(u) = \sum_{j=1}^{m} b_j h_j(u), \tag{5.1}$$

where $b_1, \ldots, b_m$ are fixed. Rather than $u_1, u_2, \ldots, u_n$ being a Markov chain with stationary distribution $f_\psi(u|y)$ it must now be drawn from h_{mix}. The entire theory goes through as in Section 3 with simply h in place of f_ψ in all formulas.

Sung and Geyer (2007) have simulated the random effects, independent of the observed data, using ordinary Monte Carlo (iid sampling) with importance sampling. This may not work for very complicated problems, but it is easier to diagnose than MCMC whose greater generality requires, in principle, more practice, trial and error. Others have simulated the random effects, dependent on the observed data, using either ordinary Monte Carlo (Ott, 1979; Kong, Liu and Wong, 1994) or MCMC (Lange and Sobel, 1991;

Thompson and Guo, 1991; Gelfand and Carlin, 1993; Geyer, 1994b and Thompson, 2003).

The virtue of MCMC is that it can sample any distribution specified by an unnormalized density. Let $h(u, y)$ be any nonnegative function satisfying, for each fixed y,

$$\tilde{h}(y) = \int h(u, y)\, du < \infty.$$

Then for any y the Metropolis-Hastings algorithm can be used to sample u_1, u_2, $\ldots$, u_n with stationary density $\pi(\cdot)$ proportional to $h(\,\cdot\,, y)$, that is, $\pi(\cdot) = h(\,\cdot\,, y)/\tilde{h}(y)$.

Everything in the MCLA approach goes through if we just replace $f_\psi(u, y)$ by $h(u, y)$ everywhere it occurs. In particular,

$$\ell_n(\theta; h) = \log\left(\frac{1}{n}\sum_{i=1}^{n}\frac{f_\theta(u_i, y)}{h(u_i, y)}\right) \tag{5.2}$$

is our approximation to the log-likelihood. As $n \to \infty$ this converges to

$$\begin{aligned} \ell(\theta; h) &= \log \mathrm{E}_h\left\{ \frac{f_\theta(U, Y)}{h(U, Y)} \,\middle|\, Y = y \right\} \qquad (5.3) \\ &= \log\left(\frac{1}{\tilde{h}(y)}\int \frac{f_\theta(u, y)}{h(u, y)} h(u, y)\, du\right) \\ &= \log f_\theta(y) - \log \tilde{h}(y) \end{aligned}$$

which is a version of the log-likelihood. The only difference is the additive constant, which is now $\log \tilde{h}(y)$ instead of $\log f_\psi(y)$ as in (3.4). But additive constants in log-likelihoods have no effect on statistical inference. As before, derivatives of (5.3) with respect to θ approximate the score and minus the observed Fisher information.

As in the simple approach, the only requirement for validity of the method is that the distribution with unnormalized density $h(u, y)$ dominate the distribution with unnormalized density $f_\theta(u, y)$ for every θ of interest. This means that for all u and θ of interest,

$$f_\theta(u, y) > 0 \Rightarrow h(u, y) > 0. \tag{5.4}$$

5.2 The Umbrella Mixture

What distribution h, subject to (5.4), do we use? The idea which has received the most interest has been called *umbrella sampling* by its originators, Torrie and Valleau (1977). The (impossible) ideal is that $h(u, y)$ would look like $f_\theta(u, y)$ for every θ of interest. The Torrie and Valleau suggestion is to use a finite mixture,

$$h(u, y) = \sum_{j=1}^{m} b_j f_{\theta_j}(u, y), \tag{5.5}$$

which they call an "umbrella" distribution because it spreads out over the region of interest. Here θ_1, …, $\theta_m \in \Theta$ must be chosen judiciously according to one's specific interests in the likelihood.

Torrie and Valleau were rather vague about how umbrella sampling was to be accomplished. An effective scheme was independently invented by Marinari and Parisi (1992) in the context of obtaining MCMC samplers with better mixing properties and by Geyer and Thompson (1995) in the context of likelihood approximation, and named *simulated tempering* by the former.

The idea of simulated tempering is to augment the state space of the Markov chain to include the index of θ_j. Thus the state is a pair of random variables (U, J) having unnormalized joint distribution

$$h(u, j, y) = b_j f_{\theta_j}(u, y). \tag{5.6}$$

One option is variable-at-a-time Metropolis-Hastings updating (Geyer, 1994):

1. Update u preserving $f_{\theta_j}(u, y)$ for the current value of j.

2. Update j preserving $b_j f_{\theta_j}(u, y)$ for the current value of u.

The u update in step 1 is the same as the one used for the simple approach and the j update tours the finite set $\{1, \ldots, m\}$. For the j update Geyer and Thompson (1995) proposed using Metropolis-Hastings moves to neighboring states (temperatures) only. We will later propose a modification. As opposed to their simulated tempering situation, the concepts of cold and hot distributions and of neighboring temperatures are somewhat irrelevant in our implementation. Any Metropolis-Hastings update that preserves the unnormalized density (5.6) should work.

Remarks Inference purposes matter when building the umbrella distribution. We recall that MCLA provides us with approximations to all the important elements of likelihood inference such as the MLE, likelihood ratios and also Monte Carlo standard errors. If one is simply interested in a point estimate, $\hat{\theta}$, then there is little point in including values far from $\hat{\theta}$ among $\{\theta_1, \ldots, \theta_m\}$. However if interest lies in calculating p-values or likelihood ratios for which tail probabilities are of interest then the umbrella distribution must include adequate (low likelihood) values of θ in $\{\theta_1, \ldots, \theta_m\}$.

Suppose the distributions $f_{\theta_j}(u, y)$, $j = 1, \ldots, m$, have been specified and that a method for updating u having $f_{\theta_j}(u, y)$ as stationary distribution is available for each j. Note that $f_{\theta_j}(u, y)$ is an unnormalized density for u (and this is all we need for a Metropolis-Hastings update). The fact that the normalizing constant for distribution j is

$$c(j) = \int f_{\theta_j}(u, y)\ \mathrm{d}u = f_{\theta_j}(y) \equiv L(\theta_j)$$

may help us choose $\theta_1, \ldots, \theta_m$ and adjust these *pseudopriors* $b_1, \ldots, b_m$. We note that the marginal distribution of the temperature J is given by

$$\Pr(J = j) \propto b_j \int f_{\theta_j}(u, y)\ \mathrm{d}u = b_j c(j) = b_j L(\theta_j). \tag{5.7}$$

Thus if the sampler is visiting all temperatures quite frequently we may use the occupation numbers $o(j) = n\widehat{\Pr}(J = j)$ to guide us in the selection of parameter points θ_j for which the likelihood attains higher (lower) values by noting that

$$\hat{L}(\theta_j) = \frac{\widehat{\Pr}(J = j)}{b_j},$$

where $\widehat{\Pr}(J = j)$ represents the proportion of time state j is visited.

Given i as the the current state, the simplest proposal distribution to use in updating J is to propose $j = j\prime$ where $j\prime \in \{1, \ldots, m\} \setminus \{i\}$ with probabilities $1/(m-1)$ so that $\Pr\{j^{(t+1)} = i | j^{(t)} = i\} = 0$.

However, this scheme may result in very slow mixing across $\{f_{\theta_j}(u, y);$ $j = 1, \ldots, m\}$ especially if there are values of θ_j very far apart (for instance, when the umbrella distribution is designed to capture the "tails" of the likelihood and also the neighbourhood of the MLE). We suggest a Gibbs update for the auxiliary variable j. More specifically, in Section 5.3 we show how Liu's (1995) improvement of the random scan Gibbs sampler (the *Metropolized Gibbs sampler*) can be adapted. This modification of the Gibbs sampler precludes the possibility of an immediate draw of the current state in the updating which then requires a Metropolis rejection-acceptance step to correct for the possible deviation from the equilibrium distribution. The literature often mentions this Metropolis correction as a means of reducing variance but it is the related concept of faster mixing (for J specially) that we are pursuing. Faster mixing means in practice that our chain does not get "stuck" in high-likelihood regions of the parameter space. In general the larger the asymptotic variance the slower the chain mixes.

A quick note on the asymptotic variance of an MCMC sampler seems in place. If P is the transition kernel (or matrix) being used to generate a Markov chain with invariant distribution π and $u_1, \ldots, u_n$ represents the generated chain, then the asymptotic variance

$$v(f, \pi, P) = \lim_{n \to \infty} n \ \mathrm{var}(\hat{\mu}_n),$$

where $\hat{\mu}_n$ is the Monte Carlo estimate of the functional $f \in L_2(\pi)$ of the Markov chain,

$$\hat{\mu}_n = \frac{1}{n} \sum_{i=1}^{n} f(U_i),$$

can be expressed in terms of the spectrum (or eigenvalues) of P and the convergence to stationarity depends on the second largest eigenvalue in absolute value (for a discrete state space) or on the spectral gap for a general state space (Robert and Rosenthal, 1997).

Brémaud (2001) gives exact expressions for $v(f, \pi, P)$ as a function of the eigenvalues of P for finite state spaces and for general state spaces Chan and Geyer (1994) provide an expression in terms of the spectra of the operator defined by P.

Jones (2004) provides a survey of conditions under which

$$\sqrt{n}\,(\hat{\mu}_n - \mathrm{E}_\pi\{f\}) \xrightarrow{D} N\,(0, v(f, \pi, P))\,.$$

Adapting Liu's (1995) proposal to our sampler makes it mix faster, since it reduces v, than simply having a random draw from $\{1, \ldots, m\}$ in the update of j.

5.3 An Umbrella Sampling Algorithm

Our algorithm updates the pair (U, J) having (unnormalized) joint distribution $h(u, j, y) = b_j f_{\theta_j}(u, y)$ with two steps in each iteration. In step 1 we update u with a Metropolis-Hastings update. In step 2 we use a Gibbs update for j with the full conditional being

$$p(j|u) = \frac{b_j f_{\theta_j}(u, y)}{d(u, y)},$$

where $d(u, y) = \sum_{j=1}^m b_j f_{\theta_j}(u, y)$.

The entire update mechanism is

1. (Metropolis: update u preserving $f_{\theta_j}(u, y)$)

   Given $j^{(t)}, u^{(t)}$ we propose $u \sim N(u^{(t)}, \tau^2\mathbf{I})$ which is accepted with probability

   $$\min\left(\frac{f_{\theta_j^{(t)}}(u, y)}{f_{\theta_j^{(t)}}(u^{(t)}, y)},\ 1 \right).$$

2. (Metropolized Gibbs: update j preserving $b_j f_{\theta_j}(u, y)$)

   Given $j^{(t)}, u^{(t+1)}$:

   (a) simulate $j \neq j^{(t)}$ with probability

   $$\frac{p(j|u^{(t+1)})}{1 - p(j^{(t)}|u^{(t+1)})};$$

(b) accept $j^{(t+1)} = j$ with probability

$$\min\left(\frac{1 - p(j^{(t)}|u^{(t+1)})}{1 - p(j|u^{(t+1)})},\ 1\right).$$

The results of implementing the above algorithm to the logit-normal model and data of previous sections is shown in Figure 4. In order to achieve nearly uniform visitation across temperatures the pseudopriors, b_j, were chosen inversely proportional to the occupation numbers $n\widehat{\Pr}(J = j)$ obtained in a pilot run. The Monte Carlo standard errors obtained were 0.055 and 0.032 for $\hat{\beta}_n$ and $\hat{\sigma}_n$ respectively. These are slightly higher than the errors found with the simple approach (Section 3) but the approximation is good well beyond the high-density contours (Figure 4, cf. Figure 2).

6 Discussion

MCSA has worked very well in getting close to the MLE and much faster than other methods (such as MCEM). It is a simple method to be under consideration as an auxiliary (starting) procedure to MCLA and to slow converging MCEM algorithms.

Our implementation in a random-intercept model based on an umbrella sampler using a Metropolized Gibbs scheme yielded a good approximation of the likelihood surfaces as illustrated in Figure 4. Monte Carlo errors were easily calculated based on overlapping batch means and on derivative formulas using importance weights. The method is not limited to finding the MLE and we provided an approximation of the entire likelihood, likelihood contours and Wald ellipses, thus generalizing other Monte Carlo methods.

Future research can be conducted in designing rules to automate the choice of the umbrella points $\theta_1, \ldots, \theta_m$ in perhaps a fairly general class of problems that would include not only binary responses (with logit or probit links) but also count data (with log link) with random-effects. Another direction is to investigate how much of the umbrella idea can be used in non-linear non-Gaussian state space models where Monte Carlo methods are needed to approximate the likelihood.

References

[1] Bédard, M. (2008) Optimal Acceptance Rates for Metropolis Algorithms: Moving Beyond 0.234. *Stochastic Processes and their Applications*, **118**, 2198-2222.

[2] Brémaud, P. (2001) *Markov Chains, Gibbs Fields, Monte Carlo Simulation and Queues*. Springer.

[3] Booth, J.; Hobert, J. (1999) Maximizing generalised linear mixed model likelihoods with an automated Monte Carlo EM algorithm. *Journal of the Royal Statistical Society, Ser. B*, **62**, 265-285.

[4] Chan, K.S.; Geyer, C.J. (1994). Discussion on 'Markov chains for exploring posterior distributions' by Luke Tierney. *The Annals of Statistics*, **22**, 1747-1758.

[5] Chib, S.; Nardari, F.; Shephard, N. (2002) Markov chain Monte Carlo methods for stochastic volatility models, *Journal of Econometrics*, **108**, 281-316.

[6] Durbin, J.; Koopman, S.J. (2000) Time series analysis of non-Gaussian observations based on state space models from both classical and Bayesian perspectives *Journal of the Royal Statistical Society, Ser. B* **62** 3-56.

[7] Gelfand, A.E.; Carlin, B.P. (1993) Maximum-likelihood estimation for constrained- or missing-data models. *Canadian Journal of Statistics*, **21**, 303-311.

[8] Geweke, J. (1989) Bayesian inference in econometric models using Monte Carlo integration. *Econometrica*, **57**, 1317-1339.

[9] Geyer, C.J. (1994) On the convergence of Monte Carlo maximum likelihood calculations. *Journal of the Royal Statistical Society, Ser. B* **56** 261-274.

[10] Geyer, C.J.; Thompson, E.A. (1992) Constrained Monte Carlo maximum likelihood for dependent data, (with discussion). *Journal of the Royal Statistical Society, Ser. B*, **54**, 657-699.

[11] Geyer, C.J.; Thompson, E.A. (1995) Annealing Markov chain Monte Carlo with applications to ancestral inference. *Journal of the American Statistical Association*, **90**, 909-920.

[12] Guilkey, D.K.; Murphy, J.L. (1993) Estimation and testing in the random-effects probit model. *Journal of Econometrics*, **59**, 301-317.

[13] Jank, W.; Booth, J. (2003) Efficiency of Monte Carlo EM and simulated maximum likelihood in generalised linear mixed models. *Journal of Computational and Graphical Statistics*, **12**, 214-229.

[14] Johnson, R.A.; Wichern, D.W. (2002) *Applied Multivariate Statistical Analysis.* Prentice-Hall.

[15] Jones, G.L. (2004) On the Markov chain central limit theorem. *Probability Surveys*, **1**, 299-320.

[16] Jungbacker, B.; Koopman, S.J. (2007) Monte Carlo estimation for nonlinear non-Gaussian state space models, *Biometrika*, 97, 827-839.

[17] Kong, A.; Liu, J.S.; Wong, W.H. (1994) Sequential Imputations and Bayesian Missing Data Problems. *Journal of the American Statistical Association*, **89**, 278-288

[18] Kuk, Anthony Y.C. (2003) Automatic choice of driving values in Monte Carlo likelihood approximation via posterior simulations. *Statistics and Computing*, **13**, 101-109.

[19] Kuk, A.Y.C.; Cheng, Y.W. (1999) Pointwise and functional approximations in Monte Carlo maximum likelihood estimation. *Statistics and Computing*, **9**, 91-99.

[20] Lange, K.; Sobel, E. (1995) A random walk method for computing genetic location scores. *American Journal of Human Genetics*, **49**, 1320-1334.

[21] Liu, J. (1995) *Metropolized Gibbs Sampler: An Improvement.* Tech. Report, Dept. of Statistics, Stanford University, CA.

[22] Louis, T.A. (1982) Finding the observed information matrix when using the EM algorithm. *Journal of the Royal Statistical Society, Ser. B*, **44**, 226-233.

[23] McCulloch, C.E. (1997) Maximum Likelihood Algorithms for Generalised Linear Mixed Models. *Journal of the American Statistical Association*, **92**, 162-170.

[24] McCulloch, C.E.; Searle, S.R.; Neuhaus, J.M. (1997) *Generalized, Linear, and Mixed Models.* Wiley.

[25] Marinari, E.; Parisi G. (1992) Simulated tempering: A new Monte Carlo scheme. *Europhysics Letters*, **19**, 451-458.

[26] Moyeed, R.A.; Baddeley, A.J. (1991) Stochastic approximation of the MLE for a spatial point pattern. *Scandinavian Journal of Statistics*, **18**, 39-50.

[27] Ott, J. (1979) Maximum likelihood estimation by counting methods under polygenic and mixed models in human pedigrees. *American Journal of Human Genetics*, **31**, 161175.

[28] Pinheiro, J.C.; Bates, D.M. (1995) Approximations to the Log-likelihood function in Nonlinear Mixed-Effects Models. *Journal of Computational and Graphical Statistics*, **4**, 12-35.

[29] Polyak, B.T.; Juditsky, A.B. (1992) Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, **30**, 838-855.

[30] Robert, G.; Rosenthal, J. (1997) Geometric Ergodicity and Hybrid Markov Chains. *Electronic Communications in Probability*, **2**, 13-25.

[31] Spall, J.C. (1999) Stochastic Optimization: Stochastic Approximation and Simulated Annealing, in *Encyclopedia of Electrical and Electronics Engineering* (J. G. Webster, ed.), Wiley, **20**, 529542.

[32] Spall, J.C. (2003) *Introduction to Stochastic Search and Optimization: Estimation, Simulation, and Control.* Wiley.

[33] Sundberg, R. (1974) Maximum likelihood theory for incomplete data from an exponential family. *Scandinavian Journal of Statistics*, **1**, 49-58.

[34] Sung, Y.J.; Geyer, C.J. (2007) Monte Carlo likelihood inference for missing data models. *The Annals of Statistics*, **35**, 990-1011.

[35] Thompson, E.A.; Guo, S.W. (1991) Evaluation of likelihood ratios for complex genetic models. *IMA Journal of Mathematics Applied in Medicine and Biology*, **8**, 149-169.

[36] Torrie, G.M.; Valleau, J.P. (1977) Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling. *Journal of Computational Physics*, **23**, 187-199.

[37] Wasan, M.T. (1969) *Stochastic Approximation.* Cambridge Univ. Press.

[38] White, H. (1982) Maximum likelihood estimation of misspecified model. *Econometrica*, **50**, 1-25.

[39] Wooldridge, J.M. (1999) *Econometric Analysis of Cross Section and Panel Data.* MIT Press.

[40] Younes, L. (1988) Estimation and annealing for Gibbsian fields. *Annales de l'Institut Henri Poincaré. Probabilités et Statistiques*, **24**, 269-294.

[41] Younes, L. (1999) On the convergence of markovian stochastic algorithms with rapidly decreasing ergodicity rates. *Stochastics and Stochastics Models* **65**, 177-228.

Table 1: Summary statistics for 100 MCSA runs, $(\hat{\beta}, \hat{\sigma}) = (6.13, 1.33)$.

	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.	SD
$\hat{\beta}_n$	4.66	6.12	6.12	6.13	6.14	6.18	0.15
$\hat{\sigma}_n$	0.09	1.32	1.33	1.32	1.35	1.37	0.13

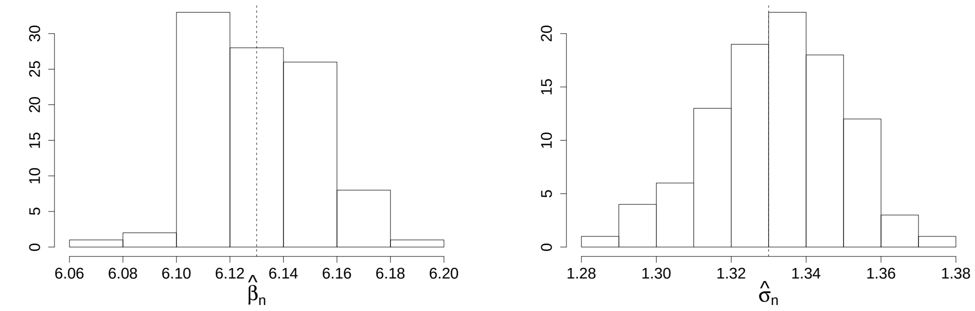

Figure 1: Histograms of $\hat{\beta}_n$ and $\hat{\sigma}_n$ for 99 MCSA runs (each of length $n = 5 \times 10^5$). Dashed lines drawn at the MLE (6.13, 1.33).

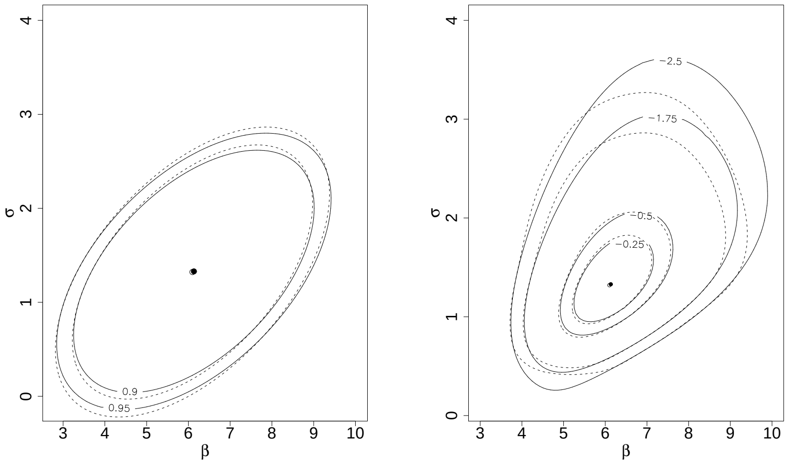

Figure 2: Left panel: Wald regions (solid) and MC approximations (left). Right panel: Contours of the (shifted) log-likelihood surface (solid) and MC versions (dashed) using (3.4) with $\psi = (6.15, 1.30)$. Full circle is MLE (hardly distinguishable from MCMLE).

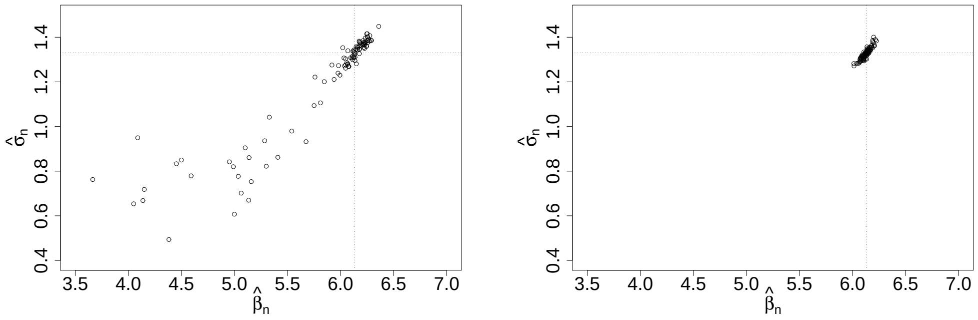

Figure 3: 100 MCMLEs with $\psi = (7.5, 2.0)$ (left) and $\psi = (6.15, 1.3)$ (right) Dashed lines cross at the MLE.

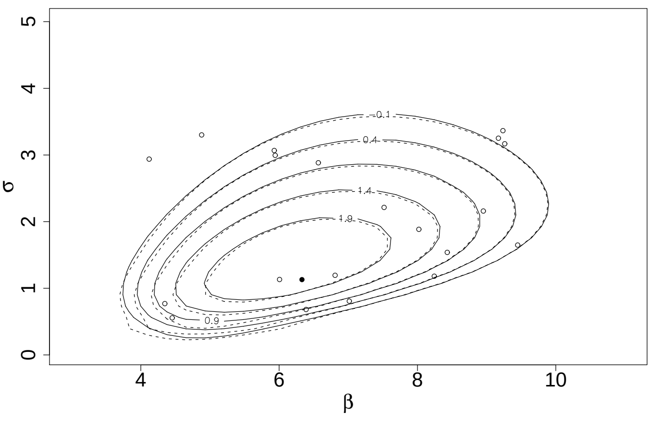

Figure 4: Contours of the log-likelihood surface (full line) and Monte Carlo approximation (5.2) (dashed) with $n = 10^5$. Empty circles are $\theta_1, \ldots, \theta_{20}$. Full circle is MLE which cannot be distinguished from the MCMLE on this scale.