J. R. Statist. Soc. B (1999)
61, *Part* 1, *pp.* 265–285

Maximizing generalized linear mixed model likelihoods with an automated Monte Carlo EM algorithm

James G. Booth and James P. Hobert

University of Florida, Gainesville, USA

[Received April 1997. Final revision May 1998]

Summary. Two new implementations of the EM algorithm are proposed for maximum likelihood fitting of generalized linear mixed models. Both methods use random (independent and identically distributed) sampling to construct Monte Carlo approximations at the E-step. One approach involves generating random samples from the exact conditional distribution of the random effects (given the data) by rejection sampling, using the marginal distribution as a candidate. The second method uses a multivariate t importance sampling approximation. In many applications the two methods are complementary. Rejection sampling is more efficient when sample sizes are small, whereas importance sampling is better with larger sample sizes. Monte Carlo approximation using random samples allows the Monte Carlo error at each iteration to be assessed by using standard central limit theory combined with Taylor series methods. Specifically, we construct a sandwich variance estimate for the maximizer at each approximate E-step. This suggests a rule for automatically increasing the Monte Carlo sample size after iterations in which the true EM step is swamped by Monte Carlo error. In contrast, techniques for assessing Monte Carlo error have not been developed for use with alternative implementations of Monte Carlo EM algorithms utilizing Markov chain Monte Carlo E-step approximations. Three different data sets, including the infamous salamander data of McCullagh and Nelder, are used to illustrate the techniques and to compare them with the alternatives. The results show that the methods proposed can be considerably more efficient than those based on Markov chain Monte Carlo algorithms. However, the methods proposed may break down when the intractable integrals in the likelihood function are of high dimension.

Keywords: Confidence ellipsoid; Hastings–Metropolis algorithm; Importance sampling; Laplace approximation; Markov chain Monte Carlo method; Rejection sampling; Salamander data; Sandwich variance estimate

1. Introduction

Generalized linear mixed models are extensions of generalized linear models (McCullagh and Nelder, 1989) that allow for additional components of variability due to unobservable effects. Typically, the unobserved effects are modelled by the inclusion of random effects in the linear predictor of the generalized linear model. The (marginal) likelihood function for the generalized linear mixed model is then obtained by integration of a generalized linear model likelihood with respect to the mixing distribution — the assumed distribution of the random effects.

Although generalized linear mixed models are a rich class of models for statistical analysis, their use in practice has been limited by the complexity of the likelihood function. This has

Address for correspondence: James P. Hobert, Department of Statistics, 203 Griffin-Floyd Hall, University of Florida, Gainesville, FL 32611, USA.
E-mail: jhobert@stat.ufl.edu

 1369–7412/99/61265

led to the development of several methods using analytical approximations to the likelihood. The main approaches involve integrating a first-order Taylor series expansion of the likelihood integrand (Goldstein, 1991; Schall, 1991; Breslow and Clayton, 1993; Wolfinger and O'Connell, 1993; Longford, 1994; McGilchrist, 1994). In particular, the methods of Breslow and Clayton (1993) and Wolfinger and O'Connell (1993) involve iterative fitting of normal theory linear mixed models and can be implemented by using the `%GLIMMIX` macro in SAS (see Littell *et al.* (1996)). Unfortunately, approximate maximum likelihood estimates of this kind have some unsatisfactory properties. In particular, they are known to be inconsistent under standard (small domain) asymptotic assumptions and the size of the asymptotic bias can be substantial if the variance components are not small (Kuk, 1995; Breslow and Lin, 1995; Lin and Breslow, 1996). Thus, there is ample motivation for investigating methods for finding the exact maximum likelihood estimate.

In this paper we discuss the use of the EM algorithm for finding exact maximum likelihood estimates in the generalized linear mixed models setting. Because the E-step of the algorithm involves an integral that cannot be evaluated analytically, we consider computer-intensive alternatives. We propose two different implementations of the Monte Carlo EM algorithm (Wei and Tanner, 1990) in which simulation methods are used to evaluate the intractable integral at the E-step. The first method uses simulated random samples from the exact conditional distribution of the random effects vector $\mathbf{u}$ given the data $\mathbf{y}$, obtained via rejection sampling, using the marginal distribution of $\mathbf{u}$ as the candidate. The second method uses an importance sampling approximation involving simulated random samples from a multivariate t-density with approximately the same mean and covariance as the true conditional distribution of $\mathbf{u}$ given $\mathbf{y}$. A key point is that both methods use *random* (independent and identically distributed) samples. This fact allows us to construct a sandwich variance estimate for assessing Monte Carlo error at each iteration of the EM algorithm. The assessment of Monte Carlo error is essential for producing a fully automated implementation. In particular, we suggest a rule for increasing the number of simulations after iterations in which the change in the parameter value is swamped by Monte Carlo error. Thus, the number of simulations used for approximation automatically increases as the algorithm approaches convergence.

The use of random samples separates our implementations of the Monte Carlo EM algorithm from those of McCulloch (1994, 1997). In each iteration of McCulloch's algorithm, a Markov chain, whose stationary distribution is the exact conditional distribution of $\mathbf{u}$ given $\mathbf{y}$, is used to approximate the E-step. Specifically, McCulloch (1994) used a Gibbs chain at each iteration to fit a probit–binomial model with normal random effects, whereas McCulloch (1997) used the Hastings–Metropolis algorithm to fit more general models.

The use of random samples has significant advantages over dependent samples arising from Markov chains. First, as noted earlier, the assessment of Monte Carlo error is straightforward when random samples are used. Specifically, this involves establishing moment conditions for validity of the central limit theorem combined with first-order Taylor series methods. In contrast, the validity of a corresponding central limit theorem for approximations based on Markov chains involves, for example, establishing that the chain is geometrically ergodic. This is a technically difficult problem even for the simplest generalized linear mixed model (see, for example, Hobert and Geyer (1997)). Moreover, even if such a condition does hold, variance formulae for Monte Carlo error assessment, such as the window estimator discussed by Geyer (1992), are far more complicated than those based on independent samples. This is presumably why McCulloch used *predetermined* values for the number of iterations and for the lengths of Markov chains and pointed to '. . . the complications of deciding whether the stochastic versions of EM . . . have converged'.

Moreover, we show in Section 7 that Markov chain Monte Carlo EM algorithms can be computationally inefficient relative to Monte Carlo EM algorithms based on random samples. This is because the variance inflation due to the use of (positively) dependent samples far outweighs the inefficiency of rejection sampling or the inexactness of the importance distribution. For example, our rejection sampling method is about 2.5 times faster than Hastings–Metropolis sampling for the simple example considered by McCulloch (1997), section 4, and our importance sampling method is more than 30 times faster than Hastings–Metropolis sampling in a more realistic example with three-dimensional random effects (see Section 7).

A possible exception to our recommendation against using Markov chain Monte Carlo EM algorithms occurs in problems involving very high dimensional integrals. In Section 7.3 we consider a logit–binomial model with normal random effects for the salamander data of McCullagh and Nelder (1989). The likelihood function in this case involves six intractable 20-dimensional integrals. McCulloch's Hastings–Metropolis method appears to be slightly faster than our importance sampling technique in this example. We note, however, that this comparison presumes that an adequate estimate of Monte Carlo variance can be constructed for use with McCulloch's method, something that has not yet been accomplished.

Other references that are related to this paper include Chan and Kuk (1997) who assessed Monte Carlo error *after a predetermined number of iterations* using several independent runs of McCulloch's (1994) Gibbs implementation of a Markov chain Monte Carlo EM algorithm. McCulloch (1997) also proposed a Markov chain Monte Carlo Newton–Raphson algorithm as an alternative to a Markov chain Monte Carlo EM algorithm. Our methods using independent samples could clearly be applied in this context as well, although we have not looked into the details of this application. Zeger and Karim (1991) used Gibbs sampling to approximate Bayesian posterior expectations of generalized linear mixed model parameters based on vague prior information. Note, however, that the use of a flat prior will often result in an improper posterior in this setting (Natarajan and McCulloch, 1995) and a diffuse prior need not result in a posterior mode (or mean) that is close to the maximum likelihood estimate (Kass and Wasserman, 1996; Natarajan and McCulloch, 1998). Also, it is worth re-emphasizing that Markov chain Monte Carlo EM algorithms require a Markov chain to be generated *at every iteration*. Thus, the amount of computation is perhaps one or two orders of magnitude greater than a typical Bayesian application aimed at approximating posterior expectations.

The remainder of this paper is laid out as follows. A formal description of the generalized linear mixed model that we consider, as well as an example, is given in the next section. Section 3 contains a brief introduction to Monte Carlo EM algorithms. Details regarding our sampling schemes are provided in Section 4. The theory behind our method of Monte Carlo error assessment appears in Section 5. Section 6 concerns convergence criteria. In Section 7, we illustrate our method and compare it with alternative methods by using three data sets. Some concluding remarks are given in Section 8.

2. The generalized linear mixed model

Let $\mathbf{y} = (y_1, \ldots, y_n)^{\mathrm{T}}$ denote an observed vector of responses and let $\mathbf{x}_i$ and $\mathbf{z}_i$ be p- and q-vectors of covariates associated with the ith response, $i = 1, \ldots, n$. We assume that, conditionally on an unobservable effects vector, $\mathbf{u} = (u_1, \ldots, u_q)^{\mathrm{T}}$, the data $\mathbf{y}$ arise from a generalized linear model with linear predictors, $\eta_i = \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta} + \mathbf{z}_i^{\mathrm{T}}\mathbf{u}$, where $\boldsymbol{\beta}$ is a p-vector of unknown regression coefficients, and $\mu_i = E(y_i|\mathbf{u})$ satisfying $g(\mu_i) = \eta_i$, for some link function

g, i.e. the responses are conditionally independent with density functions of the form

$$f(y_i|\mathbf{u};\, \boldsymbol{\beta},\, \sigma_0^2) = \exp\left[\frac{w_i}{\sigma_0^2}\{y_i\theta_i - b(\theta_i)\} + c\left(y_i,\, \frac{\sigma_0^2}{w_i}\right)\right], \tag{1}$$

where the w_i are known weights and the conditional mean and canonical parameters are related through the equation $\mu = b'(\theta)$ (see McCullagh and Nelder (1989), chapter 2). The likelihood function for the conditional generalized linear model, in which the effects vector is treated as a fixed but unknown parameter, is given by

$$f(\mathbf{y}|\mathbf{u};\, \boldsymbol{\beta},\, \sigma_0^2) = \prod_{i=1}^{n} f(y_i|\mathbf{u};\, \boldsymbol{\beta},\, \sigma_0^2). \tag{2}$$

The specification of the generalized linear mixed model is completed by assuming that $\mathbf{u}$ is a q-variate random variable with a parametric density $e(\mathbf{u};\, \boldsymbol{\sigma}_1^2)$ that depends on an $s \times 1$ vector of variance components $\boldsymbol{\sigma}_1^2$. A common assumption is that $\mathbf{u}$ is a mean 0 multivariate normal random variable with covariance matrix $\mathbf{G} = \mathbf{G}(\boldsymbol{\sigma}_1^2)$. Let $\boldsymbol{\psi}^{\mathrm{T}} = (\boldsymbol{\beta}^{\mathrm{T}},\, \sigma_0^2,\, (\boldsymbol{\sigma}_1^2)^{\mathrm{T}})$ denote the complete vector of unknown parameters. The likelihood function for $\boldsymbol{\psi}$ is given by

$$L(\boldsymbol{\psi};\, \mathbf{y}) = \int f(\mathbf{y}|\mathbf{u};\, \boldsymbol{\beta},\, \sigma_0^2)\, e(\mathbf{u};\, \boldsymbol{\sigma}_1^2)\, \mathrm{d}\mathbf{u}. \tag{3}$$

Outside the normal theory mixed model, $L(\boldsymbol{\psi};\, \mathbf{y})$ nearly always involves intractable integrals whose dimension depends on the structure of the random effects.

As an example, consider the data in Table 1 concerning 14 retrospective studies on the association between smoking and lung cancer (Dorn, 1954; Cox and Snell, 1988). Let n_{ij} denote the number of people in the ith study who were smokers ($j = 1$) or non-smokers ($j = 2$) and let y_{ij} be the corresponding number of people with lung cancer. The assumption of a common odds ratio for all 14 studies is clearly not appropriate in this instance. A model that allows for random variability in the log-odds ratios between studies is obtained by assuming that the y_{ij} are conditionally independent with

$$y_{ij}|u_i,\, v_{ij} \sim \text{binomial}(n_{ij},\, \pi_{ij}) \tag{4}$$

Table 1. Binomial response classified by treatment and study

Study	*Smokers*		*Non-smokers*		*Sample odds ratio*
	Lung cancer	*Total*	*Lung cancer*	*Total*	
1	83	155	3	17	5.38
2	90	317	3	46	5.68
3	129	210	7	26	4.32
4	70	467	12	137	1.84
5	412	711	32	163	5.64
6	597	1263	8	122	12.77
7	88	262	5	17	1.21
8	1350	2646	7	68	9.08
9	60	166	3	30	5.09
10	459	993	18	99	3.87
11	724	970	4	58	39.73
12	499	961	19	75	3.18
13	451	2180	39	675	4.25
14	260	519	5	33	5.62

where

$$\eta_{ij} = \log\left(\frac{\pi_{ij}}{1-\pi_{ij}}\right) = \beta_0 + \beta_1 x_{ij} + u_i + v_{ij}. \tag{5}$$

The βs in equation (5) are unknown regression parameters and x_{ij} is an indicator for the smokers. The u_i are assumed to be a random sample from the $N(0, \sigma_u^2)$ distribution, and independent of the v_{ij}, which are assumed to be a random sample from the $N(0, \sigma_v^2)$ distribution. Note that the elimination of the interaction terms implies a common log-odds ratio. This is a fairly simple model, yet the (marginal) likelihood function contains one intractable three-dimensional integral for each of the 14 studies. In Section 7.2 we illustrate our methods by finding the maximum likelihood estimates for this model.

Our model for the lung cancer data treats the data as though they were collected prospectively. This is appropriate if the probability of inclusion in the study was independent of smoking status and study location given disease status (see McCullagh and Nelder (1989), section 4.3.3). Although it seems reasonable that inclusion was independent of smoking status, it is not so clear why it would be independent of location, since different protocols were probably used in the different studies. However, since we are using this example only to illustrate computational methods, we shall assume that identical protocols were used in all 14 studies.

3. Monte Carlo EM algorithm

A popular method for finding maximum likelihood estimates in normal theory mixed models is an EM algorithm in which the random effects are treated as *missing data* (Searle *et al.* (1992), chapter 8). This idea can be extended to the broader generalized linear mixed model setting. Let $f(\mathbf{y}, \mathbf{u}; \boldsymbol{\psi})$ represent the joint density of the *complete data*, $(\mathbf{y}^{\mathrm{T}}, \mathbf{u}^{\mathrm{T}})^{\mathrm{T}}$, which is the same as the integrand in equation (3). Each iteration consists of an E-step and an M-step. The $(r+1)$th E-step entails the calculation of

$$Q(\boldsymbol{\psi}|\boldsymbol{\psi}^{(r)}) = E[\log\{f(\mathbf{y}, \mathbf{u}; \boldsymbol{\psi})\}|\mathbf{y}; \boldsymbol{\psi}^{(r)}] \tag{6}$$

where $\boldsymbol{\psi}^{(r)}$ denotes the value of $\boldsymbol{\psi}$ from the rth iteration. The new value $\boldsymbol{\psi}^{(r+1)}$ is defined as the maximizer of $Q(\boldsymbol{\psi}|\boldsymbol{\psi}^{(r)})$, i.e.

$$Q(\boldsymbol{\psi}^{(r+1)}|\boldsymbol{\psi}^{(r)}) \geqslant Q(\boldsymbol{\psi}|\boldsymbol{\psi}^{(r)})$$

for all $\boldsymbol{\psi}$ in the parameter space. This maximization is known as the M-step. Given a starting value $\boldsymbol{\psi}^{(0)}$, iteration between the two steps of the EM algorithm produces a sequence that, under regularity conditions (Dempster *et al.*, 1977; Wu, 1983; Little and Rubin, 1987), converges to the maximum likelihood estimate $\hat{\boldsymbol{\psi}}$.

When the density of the complete data is a member of an exponential family, it is often possible to write both steps of the EM algorithm in closed form, which leads to a very simple implementation. However, even in situations where one or both of the steps are very complicated, the EM algorithm may still be worthwhile if direct maximization is impossible. Indeed, several variants have been created to deal with situations in which iterative maximization is required at the M-step (see Rai and Matthews (1993), Meng and Rubin (1993), Liu and Rubin (1994), McLachlan and Krishnan (1997) and Meng and van Dyk (1997)). In the generalized linear mixed model context, it is the E-step that causes problems.

As the notation suggests, the expectation in equation (6) is with respect to the distribution of $\mathbf{u}$ given $\mathbf{y}$ with parameter value $\boldsymbol{\psi}^{(r)}$, whose density will be written as $h(\mathbf{u}|\mathbf{y}; \boldsymbol{\psi}^{(r)})$. This density is given by

$$h(\mathbf{u}|\mathbf{y};\ \psi) \propto f(\mathbf{y}|\mathbf{u};\ \boldsymbol{\beta},\ \sigma_0^2)\, e(\mathbf{u};\ \boldsymbol{\sigma}_1^2) \tag{7}$$

where the normalizing constant is the (marginal) likelihood function $L(\psi;\ \mathbf{y})$ given in equation (3). Thus, an analytical evaluation of equation (6) will be impossible outside the normal theory mixed model. Wei and Tanner's (1990) Monte Carlo EM algorithm avoids this difficulty by replacing the expectation in the E-step with a Monte Carlo approximation (see also Celeux and Diebolt (1985)). Specifically, let $\mathbf{u}_{r,1}, \ldots, \mathbf{u}_{r,m}$ denote a random sample from $h(\mathbf{u}|\mathbf{y};\ \psi^{(r)})$. A Monte Carlo approximation of $Q(\psi|\psi^{(r)})$ is given by

$$Q_m(\psi|\psi^{(r)}) = \frac{1}{m} \sum_{l=1}^{m} \log\{f(\mathbf{y},\ \mathbf{u}_{r,l};\ \psi)\}. \tag{8}$$

The Monte Carlo EM algorithm involves the use of Q_m in place of Q. Because of the introduction of Monte Carlo error at the E-step, the incomplete data log-likelihood is not guaranteed to increase at every iteration. However, the Monte Carlo EM algorithm still converges to the maximum likelihood estimate under suitable regularity conditions (see, for example, Chan and Ledolter (1995)).

As McCulloch (1997) has pointed out, the Monte Carlo M-step is usually relatively simple in the generalized linear mixed model context. The reason is that $Q_m(\psi|\psi^{(r)})$ is the sum of a log-likelihood from a generalized linear model involving only $\boldsymbol{\beta}$ and σ_0^2 and a second term involving only $\boldsymbol{\sigma}_1^2$. The first term can be maximized via iteratively reweighted least squares and, depending on the distribution of the random effects, the maximizer of the second term can sometimes be written in closed form.

Although it is often possible to generate a random sample from $h(\mathbf{u}|\mathbf{y};\ \psi^{(r)})$, as shown in Section 4, we now describe two alternatives to equation (8) that can be used when such a random sample is difficult to obtain. The first is based on importance sampling. Let $\mathbf{u}^*_{r,1}, \ldots, \mathbf{u}^*_{r,m}$ be a random sample from the importance density $h^*(\mathbf{u})$, which we assume has the same support as $h(\mathbf{u}|\mathbf{y};\ \psi^{(r)})$. The importance sampling Monte Carlo estimate of $Q(\psi|\psi^{(r)})$ is given by

$$Q_m(\psi|\psi^{(r)}) = \frac{1}{m} \sum_{l=1}^{m} w_{r,l}\, \log\{f(\mathbf{y},\ \mathbf{u}^*_{r,l};\ \psi)\},$$

where $w_{r,l} = h(\mathbf{u}^*_{r,l}|\mathbf{y};\ \psi^{(r)})/h^*(\mathbf{u}^*_{r,l})$ are importance weights. Since h involves an unknown normalizing constant, so do the weights. However, the normalizing constant depends on the known value $\psi^{(r)}$ and not on ψ, which means that it has no effect on the M-step and is therefore irrelevant. (Sinha *et al.* (1994) suggested this importance sampling estimate for the analysis of grouped survival data.) This estimate is discussed further in Section 4.

The second alternative to equation (8) is based on Markov chain Monte Carlo techniques. It is not imperative that $\mathbf{u}_{r,1}, \ldots, \mathbf{u}_{r,m}$ be independent. Indeed, equation (8) is still an unbiased estimate of $Q(\psi|\psi^{(r)})$ when $\mathbf{u}_{r,1}, \ldots, \mathbf{u}_{r,m}$ is a random sequence from a stationary Markov chain that is aperiodic and irreducible, and has invariant density $h(\mathbf{u}|\mathbf{y};\ \psi^{(r)})$. Even if the Markov chain is not stationary, i.e. the starting value is not drawn from the invariant density, equation (8) is still a consistent estimate of $Q(\psi|\psi^{(r)})$ (see Tierney (1994)). In McCulloch's (1997) Monte Carlo EM algorithm for generalized linear mixed models, the sequence $\mathbf{u}_{r,1}, \ldots, \mathbf{u}_{r,m}$ is from a Markov chain constructed by using a Metropolis algorithm (see also McCulloch (1994) and Chan and Kuk (1997)).

There have been other suggestions concerning the intractable E-step. Steele (1996) suggested replacing the exact conditional expectation in the E-step with a second-order Laplace approximation. He argued that this modified EM algorithm will

'. . . produce more accurate estimates of the fixed effects, and perhaps of the random effects, than Breslow and Clayton's (1993) PQL algorithm'.

Although this may be true, the error in the Laplace approximation precludes this modified EM algorithm from finding the maximum likelihood estimate. Sammel *et al.* (1997) noted that the right-hand side of equation (6) is the ratio of two expectations with respect to $e(\mathbf{u}; \boldsymbol{\sigma}_1^2)$. They proposed to estimate Q with a ratio of Monte Carlo approximations but stated that the method is quite slow in practice.

4. Monte Carlo approximation of the E-step using random samples

4.1. Rejection sampling

Outside the normal mixed model setting, the conditional density h is typically a non-standard multivariate density depending on an unknown (normalizing) constant. Suppressing dependence on the parameters and the data, the characterization in expression (7) can be written as

$$h(\mathbf{u}) = a\, f(\mathbf{u})\, e(\mathbf{u}),$$

where a is the normalizing constant. A random sample from h can be selected as follows by multivariate rejection sampling (Geweke (1996), section 3.2).

Step 1: sample $\mathbf{u}$ from e and, independently, sample w from the uniform(0, 1) distribution.
Step 2: if $w \leqslant f(\mathbf{u})/\tau$ where $\tau = \sup_{\mathbf{u}}\{f(\mathbf{u})\}$, then accept $\mathbf{u}$; if not, go to step 1.

Finding the supremum τ required in step 2 is equivalent to finding the maximum likelihood estimate of the regression parameter for a generalized linear model with an *offset* (McCullagh and Nelder (1989), p. 206), which can be accomplished by using iteratively reweighted least squares.

Although finding a new τ at each iteration of a Monte Carlo EM algorithm is not a great burden, it is sometimes unnecessary as we now explain. Let $\mathbf{X}$ and $\mathbf{Z}$ denote the $n \times p$ and $n \times q$ covariate matrices with ith rows equal to $\mathbf{x}_i^{\mathrm{T}}$ and $\mathbf{z}_i^{\mathrm{T}}$ respectively. Assume without loss of generality that $\mathbf{X}$ has full rank p. In addition, let $\mathbf{C}$ denote the combined $n \times (p+q)$ covariate matrix $(\mathbf{X}, \mathbf{Z})$. Suppose that $\mathbf{C}$ has k distinct rows, $\mathbf{c}_j^* = (\mathbf{x}_j^{*\mathrm{T}}, \mathbf{z}_j^{*\mathrm{T}})^{\mathrm{T}}$, $j = 1, \ldots, k$, where $1 \leqslant k \leqslant n$, and let S_j denote the set of indices of the rows of $\mathbf{C}$ equal to $\mathbf{c}_j^*$. Let $\eta_j^* = \mathbf{x}_j^{*\mathrm{T}}\boldsymbol{\beta} + \mathbf{z}_j^{*\mathrm{T}}\mathbf{u}$ and let θ_j^* and μ_j^* denote the canonical parameter and mean parameter values respectively corresponding to $\mathbf{c}_j^*$ in the conditional model. Then the conditional log-likelihood can be written in the form

$$\log\{f(\mathbf{y}|\mathbf{u};\, \boldsymbol{\beta},\, \sigma_0^2)\} = \sum_{j=1}^{k} \sum_{i \in S_j} \left[\frac{w_i}{\sigma_0^2}\{y_i\theta_j^* - b(\theta_j^*)\} + c\left(y_i,\, \frac{\sigma_0^2}{w_i}\right)\right].$$

This log-likelihood is maximized, for any fixed value of σ_0^2, when $\theta_j^* = (b')^{-1}(\mu_j^*)$ and $\eta_j^* = g^{-1}(\mu_j^*)$, where

$$\mu_j^* = \sum_{i \in S_j} w_i y_i \Big/ \sum_{i \in S_j} w_i.$$

Denote this maximum value by $\gamma(\sigma_0^2)$.

Let $\mathbf{X}^*$ $(k \times p)$ and $\mathbf{Z}^*$ $(k \times q)$ denote the matrices with jth rows equal to $\mathbf{x}_j^{*\mathrm{T}}$ and $\mathbf{z}_j^{*\mathrm{T}}$ respectively and let $\boldsymbol{\eta}^* = (\eta_1^*, \ldots, \eta_k^*)^{\mathrm{T}}$, so that

$$\mathbf{Z}^*\mathbf{u} = \boldsymbol{\eta}^* - \mathbf{X}^*\boldsymbol{\beta}.$$

A solution to this equation is guaranteed, for all $\boldsymbol{\beta}$ and $\boldsymbol{\eta}^*$, if $\text{rank}(\mathbf{Z}^*) = k$. Therefore, when $\text{rank}(\mathbf{Z}^*) = k$, $\tau = \exp(\gamma)$ which depends only on the data. An example of a model for which this condition is satisfied is given in Section 7.2.

This simple method of simulating from $h(\mathbf{u}|\mathbf{y}; \boldsymbol{\psi})$ is often very fast even if the acceptance rate is quite low provided that it is easy to simulate from the assumed random effects density. It is interesting that McCulloch's (1997) Metropolis algorithm, which produces a Markov chain with invariant density h, also uses the marginal random effects density as a candidate.

4.2. Importance sampling

When the acceptance rate for the rejection sampler is very low, it may be more efficient to use importance sampling. We propose a multivariate Student t importance density whose mean and variance match the mode and curvature of h. (A multivariate normal importance density is less likely to yield the finite moments necessary for validity of the central limit theorem discussed in Section 5.) More specifically, suppressing the dependence on $\mathbf{y}$ and $\boldsymbol{\psi}$, we write $h(\mathbf{u}|\mathbf{y}; \boldsymbol{\psi}) = a \exp\{l(\mathbf{u})\}$, where a is the unknown normalizing constant. Let $l^{(1)}(\mathbf{u})$ denote the vector of first derivatives of $l(\mathbf{u})$ and $l^{(2)}(\mathbf{u})$ the second-derivative matrix. Suppose that $\tilde{\mathbf{u}}$ is the maximizer of $l(\mathbf{u})$ satisfying the equation $l^{(1)}(\mathbf{u}) = \mathbf{0}$. The Laplace approximations of the mean and variance (de Bruijn (1981), chapter 4) are $\tilde{\mathbf{u}}$ and $-l^{(2)}(\tilde{\mathbf{u}})^{-1}$ respectively. Booth and Hobert (1998) give formulae for these derivatives when the random effects are normal.

5. Normal approximation of Monte Carlo error

In this section, some theory is developed concerning the size of the Monte Carlo error in $\boldsymbol{\psi}^{(r+1)}$. These results are useful for choosing m and for deciding when the Monte Carlo EM algorithm has converged. Throughout this section we assume that Q is estimated by using a random sample from $h(\mathbf{u}|\mathbf{y}; \boldsymbol{\psi}^{(r)})$ or importance sampling. However, our arguments are not restricted to the generalized linear mixed model situation.

Define

$$Q^{(1)}(\boldsymbol{\psi}|\boldsymbol{\psi}') = \frac{\partial}{\partial \boldsymbol{\psi}} Q(\boldsymbol{\psi}|\boldsymbol{\psi}')$$

and

$$Q^{(2)}(\boldsymbol{\psi}|\boldsymbol{\psi}') = \frac{\partial^2}{\partial \boldsymbol{\psi} \mathrm{d} \boldsymbol{\psi}^{\mathrm{T}}} Q(\boldsymbol{\psi}|\boldsymbol{\psi}')$$

and define $Q_m^{(j)}$ similarly. Now,

$$\mathbf{0} = Q_m^{(1)}(\boldsymbol{\psi}^{(r+1)}|\boldsymbol{\psi}^{(r)}) \approx Q_m^{(1)}(\boldsymbol{\psi}^{*(r+1)}|\boldsymbol{\psi}^{(r)}) + (\boldsymbol{\psi}^{(r+1)} - \boldsymbol{\psi}^{*(r+1)})^{\mathrm{T}} Q_m^{(2)}(\boldsymbol{\psi}^{*(r+1)}|\boldsymbol{\psi}^{(r)}),$$

where $\boldsymbol{\psi}^{*(r+1)}$ satisfies $Q^{(1)}(\boldsymbol{\psi}^{*(r+1)}|\boldsymbol{\psi}^{(r)}) = \mathbf{0}$. It follows that, conditionally on $\boldsymbol{\psi}^{(r)}$, $\boldsymbol{\psi}^{(r+1)}$ is approximately normal with mean $\boldsymbol{\psi}^{*(r+1)}$ and variance

$$\text{var}(\boldsymbol{\psi}^{(r+1)}|\boldsymbol{\psi}^{(r)}) \approx Q_m^{(2)}(\boldsymbol{\psi}^{*(r+1)}|\boldsymbol{\psi}^{(r)})^{-1} \text{var}\{Q_m^{(1)}(\boldsymbol{\psi}^{*(r+1)}|\boldsymbol{\psi}^{(r)})\} Q_m^{(2)}(\boldsymbol{\psi}^{*(r+1)}|\boldsymbol{\psi}^{(r)})^{-1}. \quad (9)$$

A sandwich estimate of $\text{var}(\boldsymbol{\psi}^{(r+1)}|\boldsymbol{\psi}^{(r)})$ is obtained by substituting $\boldsymbol{\psi}^{(r+1)}$ in place of $\boldsymbol{\psi}^{*(r+1)}$ on the right-hand side of expression (9) and using the estimate

$$\widehat{\text{var}}\{Q_m^{(1)}(\psi^{*(r+1)}|\psi^{(r)})\} = \frac{1}{m^2}\sum_{l=1}^{m}\left(w_{r,l}\frac{\partial}{\partial\psi}\log\{f(\mathbf{y}, \mathbf{u}_{r,l}; \psi^{(r+1)})\}\right)\left(w_{r,l}\frac{\partial}{\partial\psi}\log\{f(\mathbf{y}, \mathbf{u}_{r,l}; \psi^{(r+1)})\}\right)^{\mathrm{T}}$$

where the importance weights $w_{r,l}$ are all set equal to 1 when direct random sampling is used. Thus, after the $(r+1)$th iteration of the Monte Carlo EM algorithm has been performed, the new value $\psi^{(r+1)}$ is approximately normally distributed with mean $\psi^{*(r+1)}$ and a covariance matrix that can be estimated.

Now consider attempting to generalize these arguments to McCulloch's (1997) version of the Monte Carlo EM algorithm which is based on Markov chain Monte Carlo sampling. Suppose that $X_1, X_2, \ldots$ is an irreducible, aperiodic, positive recurrent Markov chain produced by using a Markov chain Monte Carlo technique and that $s(\cdot)$ is some function. Conditions ensuring that a properly normalized version of $n^{-1}\Sigma_{i=1}^{n} s(X_i)$ converges in distribution to a normal random variable involve the mixing properties of the Markov chain and are generally much more difficult to verify than the standard (moment) conditions for random samples (Geyer, 1992; Tierney, 1994; Chan and Geyer, 1994; Roberts and Rosenthal, 1997; Hobert and Geyer, 1997). Thus, constructing a normal approximation for $\psi^{(r+1)}$ is a much more difficult problem when Q_m is based on a Markov chain and, given what is currently known, it is not clear that it is possible to derive such an approximation in that case. Furthermore, even if a central limit theorem does hold, estimators for the Monte Carlo variance are far more complicated than our sandwich formula (see Geyer (1992)).

6. Stopping rules

An important issue in the implementation of the Monte Carlo EM algorithm is the choice of the Monte Carlo sample size m. Because simulation is time consuming, there is an obvious trade-off between accurate approximation of Q and speed. Wei and Tanner (1990) noted that

> '. . . it is inefficient to start with a large value of m when the current approximation to the [maximum likelihood estimate] may be far from the true value'.

They went on to suggest that m should increase with the number of iterations but did not say exactly how this should be done. Clearly, at the $(r+1)$th iteration of the Monte Carlo EM algorithm, the larger the value of m, the closer $\psi^{(r+1)}$ will be to $\psi^{*(r+1)}$ (on average). If the Monte Carlo error associated with $\psi^{(r+1)}$ (as an estimate of $\psi^{*(r+1)}$) is not small relative to $\|\psi^{*(r+1)} - \psi^{(r)}\|$, then the $(r+1)$th iteration of the Monte Carlo EM algorithm is wasted because the EM step has been 'swamped' by Monte Carlo error. Of course, it is also wasteful to use a very large value of m when $\|\psi^{*(r+1)} - \psi^{(r)}\|$ is large, which is typically the case at the start.

Thus, automation of Monte Carlo EM algorithms requires the ability to assess the Monte Carlo error associated with $\psi^{(r+1)}$. However, as noted earlier, recently proposed Monte Carlo EM algorithms for generalized linear mixed models make no attempt to assess this Monte Carlo error and use *ad hoc* methods for increasing m. For example, in McCulloch (1994), m is increased linearly with the number of iterations, whereas in McCulloch (1997) $m = 50$ for iterations 1–19, $m = 200$ for iterations 20–39 and $m = 5000$ for iterations 40 and over. Chan and Kuk (1997) used $m = 1000$ at each of their 300 iterations. The examples in Section 7 illustrate that many Monte Carlo EM iterations will be wasted when such *ad hoc* schemes are employed for choosing m.

A standard stopping rule (or convergence criterion) for deterministic EM algorithms is to stop (and claim convergence) when the relative change in the parameter values from

successive iterations is small. For example, the algorithm is stopped when

$$\max_i \left(\frac{|\psi_i^{(r+1)} - \psi_i^{(r)}|}{|\psi_i^{(r)}| + \delta_1} \right) < \delta_2 \tag{10}$$

where δ_1 and δ_2 are predetermined constants. (Stopping rules that depend on changes in the unknown incomplete data log-likelihood are much less convenient.) Searle *et al.* (1992), p. 296, reported that several researchers suggest using $\delta_1 = 0.001$ and $\delta_2 = 0.0001$. It is not possible to use the stopping rule (10) unless the Monte Carlo EM steps can be resolved into the true EM steps and Monte Carlo error. This is presumably why McCulloch (1994, 1997) and Chan and Kuk (1997) ran their Monte Carlo EM algorithms for a *predetermined* number of iterations. The alternative that we propose is automated in that an appropriate value for m is chosen after each iteration, and the algorithm is stopped when changes in the parameter estimates are small after taking Monte Carlo error into account.

Our updating scheme for m is as follows. After the $(r+1)$th iteration, construct an approximate $100(1-\alpha)\%$ confidence ellipsoid for $\psi^{*(r+1)}$ by using the normal approximation derived in Section 5. If the previous value $\psi^{(r)}$ lies in that region, then the EM step was swamped by Monte Carlo error, and m should be increased, e.g. $m \leftarrow m + m/k$, where k is a positive constant. We have been successful using this method with $\alpha = 0.25$ and $k \in \{3, 4, 5\}$ (see Section 7), but the optimal choice of α and k is a topic that needs further investigation.

It is well known that the EM algorithm may stall temporarily before reaching the maximizer. The smaller the value of δ_2, the smaller the chance that the algorithm will be stopped prematurely. However, at present it is not feasible to use the value $\delta_2 = 0.0001$ that is recommended for deterministic algorithms because of the excessive Monte Carlo sample size required for such precision. (More powerful computers will change this situation.) In our implementation of the Monte Carlo EM method we used values of δ_2 between 0.002 and 0.005. Thus, the price that we pay for the inability to calculate the E-step in closed form is running a larger risk of stopping prematurely.

Another problem with this stopping rule in the Monte Carlo EM context is that $\psi^{(r+1)}$ can be very close to $\psi^{(r)}$ simply because a large amount of Monte Carlo error is associated with $\psi^{(r+1)}$. To reduce the risk of stopping prematurely because of an unlucky Monte Carlo sample, we did not stop the algorithm in our examples until rule (10) was satisfied for three consecutive iterations.

Finally, it is well known that the EM algorithm has problems when the parameter is close to the boundary of the parameter space. This often occurs in variance components models when one of the variance components is close to 0 and the model is essentially overparameterized. In such cases an extremely large number of EM iterations may be required to attain the convergence criterion (10) even though the likelihood function is effectively maximized after a few iterations. One way of diagnosing this problem is to compare the maximum likelihood estimate of the parameters values with their standard errors. Intuitively, there is little point in demanding the relative precision of rule (10) when it is clear that the estimate is very small relative to its standard error. For this reason, we suggest using a second convergence criterion in conjunction with criterion (10) based on the change in the parameter estimates relative to their standard errors. More specifically, the algorithm is stopped if

$$\max_i \left\{ \frac{|\psi_i^{(r+1)} - \psi_i^{(r)}|}{\sqrt{\text{var}(\hat{\psi}_i)} + \delta_1} \right\} < \delta_2 \tag{11}$$

for suitable small values of δ_1 and δ_2. (These need not be the same as in criterion (10).) This convergence criterion is similar to one described by Bates and Watts (1981) for non-linear least squares. The variance of $\hat{\psi}_i$ in the denominator of expression (11) can be estimated by using an estimate of the observed Fisher information evaluated at the current parameter estimate. The observed information matrix is essentially a by-product of our algorithm. Indeed, Louis (1982) showed that the observed information matrix can be written as the sum of $-Q^{(2)}(\psi|\hat{\psi})$ and

$$-\text{var}\left[\frac{\partial}{\partial\psi}\log\{f(\mathbf{y}, \mathbf{u}; \psi)\}|\mathbf{y}; \hat{\psi}\right]$$

both evaluated at $\hat{\psi}$. The matrix $Q_m^{(2)}$ from the final iteration of our algorithm can be used to approximate $-Q^{(2)}(\psi|\hat{\psi})$, and a Monte Carlo estimate of

$$\text{var}\left[\frac{\partial}{\partial\psi}\log\{f(\mathbf{y}, \mathbf{u}; \psi)\}|\mathbf{y}; \hat{\psi}\right]$$

can be constructed by using the simulations from the last iteration (Tanner (1996), chapter 4).

7. Examples

In Section 7.1 we consider a data set simulated according to the model that McCulloch (1997) used to demonstrate his Monte Carlo EM algorithm. Because the random effects are univariate, the likelihood can be evaluated by using one-dimensional numerical integration, and the maximum likelihood estimates can be found without the need for a Monte Carlo EM algorithm. Thus, we can compare our algorithm with McCulloch's in a situation where the maximum likelihood estimates are known *a priori*. The lung cancer data introduced in Section 2 are considered in Section 7.2 These data are modelled by using a generalized linear mixed model with trivariate random effects. In Section 7.3 we fit a binary response model with a logit link and normal random effects to the infamous salamander data of McCullagh and Nelder (1989), chapter 14.

All our computations were done in Fortran on a SPARC Ultra 1 workstation.

7.1. McCulloch's model

Suppose that the y_{ij} are conditionally independent with

$$y_{ij}|u_i \sim \text{Bernoulli}(\pi_{ij}) \tag{12}$$

for $i = 1, 2, \ldots, q$ and $j = 1, 2, \ldots, n$ where

$$\eta_{ij} = \log\left(\frac{\pi_{ij}}{1-\pi_{ij}}\right) = \beta_1 x_{ij} + u_i \tag{13}$$

The u_i are assumed to be a random sample from the $N(0, \sigma^2)$ distribution. McCulloch (1997) used data that were simulated according to this model with $q = 10$, $n = 15$, $\beta = 5$, $\sigma^2 = \frac{1}{2}$ and $x_{ij} = j/15$ but did not report the data. Using the same settings, we generated the data in Table 2. The exact maximum likelihood estimate, $(\hat{\beta}, \hat{\sigma}^2) = (6.132, 1.766)$, was calculated by using numerical integration. Our rejection sampling algorithm with $k = 3$, $\alpha = 0.25$, $\delta_1 = 0.001$ and $\delta_2 = 0.002$ converged in 47 iterations after 4.2 min using $(\beta, \sigma^2) = (2, 1)$ as the starting value. The value of m increased from $m = 100$ at the start to 17536 at the final iteration. Fig. 1

Table 2. Data simulated according to McCulloch's model

i	*Values for the following values of j:*														
	1	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*	*11*	*12*	*13*	*14*	*15*
1	1	0	0	0	0	1	1	0	1	1	1	1	1	1	1
2	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
5	0	1	1	1	1	1	1	1	1	1	0	1	1	1	1
6	0	0	0	1	0	1	1	1	0	1	1	1	1	1	1
7	0	1	0	0	1	1	1	1	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
9	1	0	0	1	1	0	1	1	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

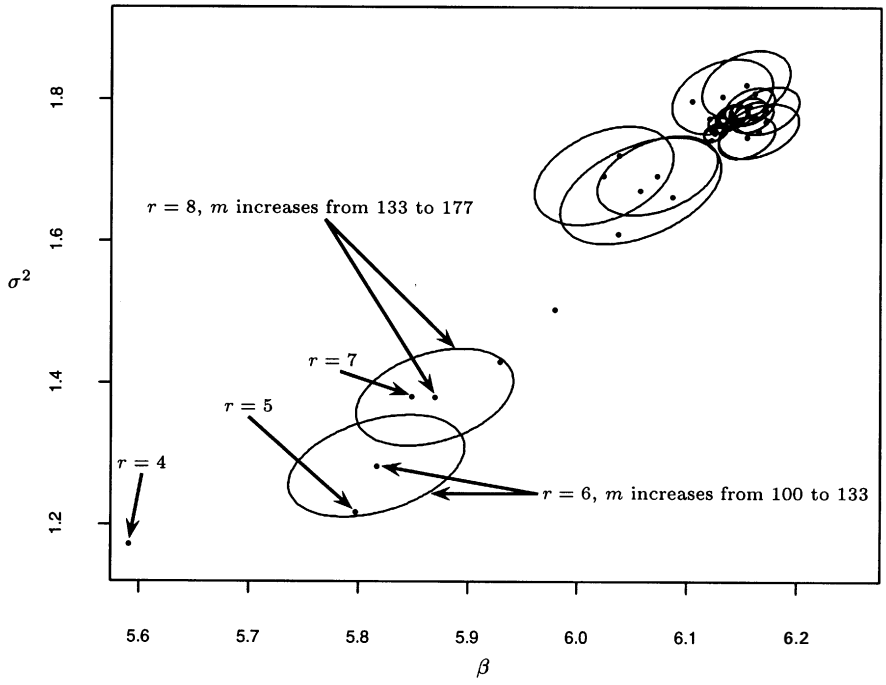

Fig. 1. Iterations 4–47 of the Monte Carlo EM algorithm: iteration numbers are given for only five of the points; the approximate 75% confidence ellipse for $\psi^{\star(r+1)}$ contained $\psi^{(r)}$ 18 times, resulting in 18 increases in m; for example, note that the ellipse centred at $\psi^{(6)}$, which is an approximate 75% confidence ellipse for $\psi^{\star(6)}$, contains $\psi^{(5)}$; this resulted in m being increased from 100 to 133; all 18 such ellipses are shown (the starting value and first three iterations are absent for aesthetic reasons)

provides a graphical representation of the iteration history and shows all the ellipses that resulted in an increase in m.

The most direct way of comparing our algorithm with McCulloch's (1997) Markov chain Monte Carlo method would be to run his algorithm until convergence and to compare the time with 4.2 min. However, for the reasons given in Section 5, Monte Carlo error assessment (at least, of the type needed to assess convergence) is not yet possible for McCulloch's method. In lieu of such a direct comparison, we ran McCulloch's algorithm for 4.2 min using

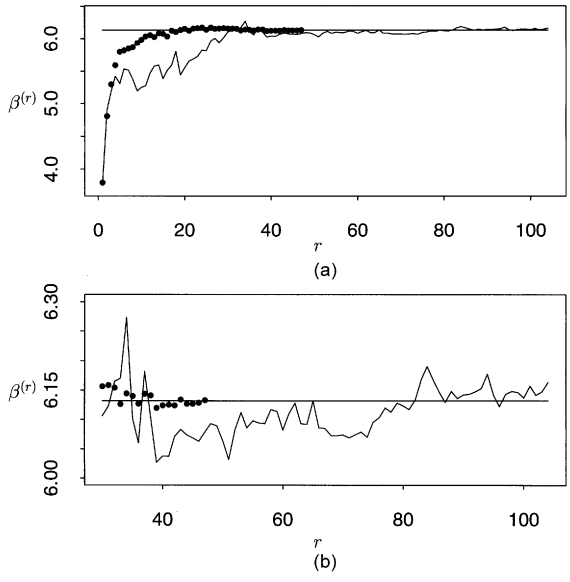

Fig. 2. Comparison of the histories of two Monte Carlo EM algorithms: (a) iteration r *versus* $\beta^{(r)}$ (•, our algorithm; ———, McCulloch's (1997) algorithm with $m = 50$, 200, 5000 for iterations 1–19, 20–39 and 40–104); the starting value $(\beta, \sigma^2) = (2, 1)$ was used in both cases; the maximum likelihood estimate, found by numerical integration, is $(\hat{\beta}, \hat{\sigma}^2) = (6.132, 1.766)$; the horizontal line represents the maximum likelihood estimate of β; the corresponding plot for σ^2 is very similar; (b) as for (a) except only iterations 30–104 are shown

his *predetermined* sequence of m-values: $m = 50$, 200, 5000 for iterations 1–19, 20–39 and 40 and over. In 4.2 min, 104 iterations were performed. Thus, iterations 40–104 were performed using $m = 5000$. Fig. 2 shows the values of β after each iteration for both methods. This plot demonstrates the advantage of updating m after each iteration. Clearly, McCulloch's predetermined sequence increases much too slowly and stops prematurely at $m = 5000$. Our method is superior because we can resolve each Monte Carlo EM step into the true EM step and Monte Carlo error. From this comes the ability to update m automatically and to diagnose convergence.

Fig. 3 shows what happens when McCulloch's algorithm is run using our sequence of ms. It took only about 2 min to do this, but it is clear that to achieve convergence according to criterion (10) using McCulloch's algorithm would require much larger values of m.

The true EM algorithm does not move from the maximum likelihood estimate. Thus, we can observe *pure* Monte Carlo error by starting at the maximum likelihood estimate and performing a single iteration of the Monte Carlo EM algorithm. We performed this experiment 1000 times for each of three values of m (100, 500 and 1000) for both algorithms and the results are given in Table 3. The average value of β over the 1000 replications is denoted $\bar{\beta}$ and the standard deviation of those 1000 values of β is denoted s_β. The corresponding quantities for σ^2 are denoted $\bar{\sigma}^2$ and s_{σ^2}. Table 3 has several interesting features. Note that the values of $\bar{\beta}$ and $\bar{\sigma}^2$ for McCulloch's algorithm begin some distance from the maximum likelihood estimates but appear to move closer with increasing m. This is the effect of *burn-in*, i.e. of not starting the Markov chain according to its stationary distribution (see Section 7.2). The stability of the values of $s\sqrt{m}$ is empirical evidence that, at least in this example, there is a central limit theorem for the maximizer of the Hastings–Metropolis estimate of the E-step. Finally, it appears that m independent samples are

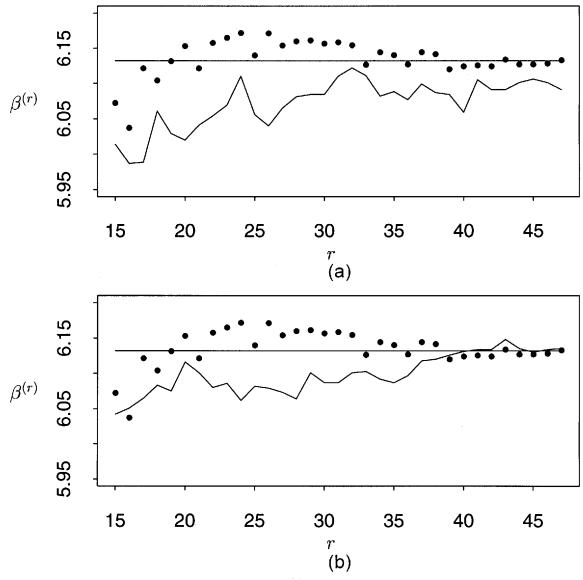

Fig. 3. McCulloch's algorithm with different multiples of our sequence of ms: (a) iteration r *versus* $\beta^{(r)}$ (•, our algorithm; ———, McCulloch's algorithm); the starting value $(\beta, \sigma^2) = (2, 1)$ was used in both cases; the maximum likelihood estimate, found by numerical integration, is $(\hat{\beta}, \hat{\sigma}^2) = (6.132, 1.766)$; the horizontal line represents the maximum likelihood estimate of β; the corresponding plot for σ^2 is very similar; (b) as for (a) except that McCulloch's algorithm was run with our sequence of ms multiplied by 5

Table 3. Measuring pure Monte Carlo error†

m	*Results from rejection sampling*				*Results from Hastings–Metropolis sampling*			
	$\bar{\beta}$	$\bar{\sigma}^2$	$s_\beta/\sqrt{m}$	$s_{\sigma^2}/\sqrt{m}$	$\bar{\beta}$	$\bar{\sigma}^2$	$s_\beta/\sqrt{m}$	$s_{\sigma^2}/\sqrt{m}$
100	6.130	1.766	0.584	0.643	6.091	1.711	1.260	1.393
500	6.132	1.766	0.563	0.657	6.125	1.755	1.241	1.501
1000	6.132	1.765	0.569	0.658	6.129	1.762	1.255	1.510

†Recall that the exact maximum likelihood estimate is $(\hat{\beta}, \hat{\sigma}^2) = (6.132, 1.766)$. The standard error of $\bar{\beta}$ is given by $s_\beta/\sqrt{m}/\sqrt{(1000m)}$ and similarly for $\bar{\sigma}^2$. The standard error of $s_\beta/\sqrt{m}$ is $s_\beta/\sqrt{m}/\sqrt{2000}$ and similarly for $s_{\sigma^2}/\sqrt{m}$.

equivalent to about $(1.25/0.57)^2 m \approx (1.5/0.66)^2 m \approx 5m$ dependent samples from the Hastings–Metropolis algorithm.

Fig. 3 also shows the result of running McCulloch's algorithm using our sequence of ms multiplied by 5. This took about 10 min. It appears that there is a similar amount of Monte Carlo error in the two sequences, which means that it is fair to compare 10 min with the 4.2 min that it took to run our Monte Carlo EM algorithm. Therefore, in this example, our algorithm is about 2.5 times faster than the Markov chain Monte Carlo based algorithm. Actually, this comparison is unfair to our algorithm since the 4.2 min includes the time that it took to assess the Monte Carlo error at each step and thus to determine the sequence of ms that we used in the comparison.

It is not surprising that the rejection sampler outperforms Hastings–Metropolis sampling when the random effects are univariate. The example in the next section involves trivariate random effects, and rejection sampling again outperforms Hastings–Metropolis sampling. Furthermore, importance sampling is 30 times faster than rejection sampling!

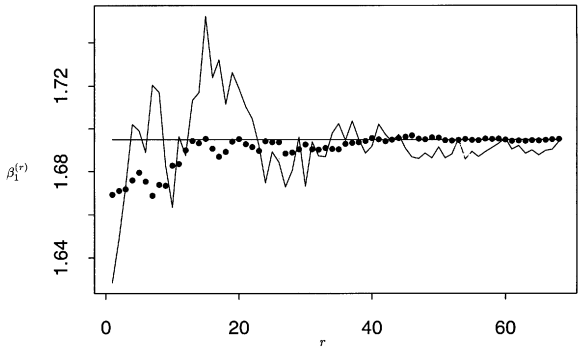

Fig. 4. McCulloch's Monte Carlo EM algorithm with our sequence of *m*s: iteration *r versus* $\beta_1^{(r)}$ (•, our algorithm; ———, McCulloch's algorithm); the starting value $(\beta_0, \beta_1, \sigma_u^2, \sigma_v^2) = (-1.898, 1.660, 0.2012, 0.2502)$ was used in both cases; the maximum likelihood estimate of β_1 is 1.695; the horizontal line represents the maximum likelihood estimate of β_1; the corresponding plots for the other three parameters are similar

7.2. Lung cancer data

In this section, we illustrate our Monte Carlo EM methodology by using the lung cancer data described in Section 2. Rejection sampling and importance sampling were both applied, and we report the results based on the rejection sampler first. The pseudolikelihood estimates produced by the SAS `%GLIMMIX` macro (Littell *et al.*, 1996; Wolfinger and O'Connell, 1993) were $(\beta_0, \beta_1, \sigma_u^2, \sigma_v^2) = (-1.898, 1.660, 0.2012, 0.2502)$. Using these starting values along with $\alpha = 0.25$, $k = 5$ and $\delta_2 = 0.002$, the algorithm converged in 68 iterations after 150 min to $(\hat{\beta}_0, \hat{\beta}_1, \hat{\sigma}_u^2, \hat{\sigma}_v^2) = (-1.932, 1.695, 0.1896, 0.2318)$. In this example, the value of m increased from the initial value of 100 to 58 263 at the final iteration.

The number of patients within each study is fairly large, which means that the integral approximations used in pseudolikelihood should be very accurate in this situation. Thus, it is not surprising that the pseudolikelihood estimates are quite close to the maximum likelihood estimates. An estimate of the inverse of the observed information matrix (Tanner (1996), chapter 4) is

$$I^{-1}(\hat{\psi}) = \begin{pmatrix} 7 \times 10^{-2} & & & \\ -2 \times 10^{-2} & 4 \times 10^{-2} & & \\ -5 \times 10^{-4} & -4 \times 10^{-4} & 4 \times 10^{-3} & \\ 5 \times 10^{-4} & 5 \times 10^{-4} & -4 \times 10^{-4} & 3 \times 10^{-3} \end{pmatrix}.$$

Running McCulloch's (1997) algorithm with our sequence of *m*s took only 6.5 min. (The large time difference is because the overall acceptance rate of the rejection sampler in this example is only about 1.5%). However, it is clear from Fig. 4 that to achieve convergence by using McCulloch's algorithm would require much larger values of m. To ascertain the appropriate multiple of our sequence of *m*s, we performed an experiment to measure the pure Monte Carlo error similar to the experiment in the previous section. The results are given in Table 4.

The most interesting aspect of Table 4 is that the values of $s\sqrt{m}$ for the Hastings–Metropolis algorithm seem to be increasing with m, and starting the Markov chain at stationarity does not alleviate the problem. This could be an indication that there is no central limit theorem in

Table 4. Measuring pure Monte Carlo error†

m	$\bar{\beta}_0$	$\bar{\beta}_1$	$\bar{\sigma}_u^2$	$\bar{\sigma}_v^2$	$s_{\beta_0}\sqrt{m}$	$s_{\beta_1}\sqrt{m}$	$s_{\sigma_u^2}\sqrt{m}$	$s_{\sigma_v^2}\sqrt{m}$
Rejection sampling								
500	−1.932	1.695	0.1896	0.2318	0.074	0.075	0.058	0.051
1000	−1.932	1.695	0.1896	0.2318	0.075	0.078	0.057	0.049
5000	−1.932	1.695	0.1897	0.2317	0.075	0.076	0.059	0.050
Hastings–Metropolis sampling								
500	−1.938	1.702	0.1865	0.2202	0.304	0.318	0.874	0.513
	(−1.932)	(1.695)	(0.1886)	(0.2331)	(0.347)	(0.365)	(0.902)	(0.614)
1000	−1.937	1.700	0.1865	0.2235	0.391	0.407	1.011	0.648
	(−1.932)	(1.695)	(0.1876)	(0.2330)	(0.414)	(0.433)	(1.053)	(0.800)
5000	−1.934	1.697	0.1883	0.2294	0.521	0.544	1.399	1.037
	(−1.932)	(1.695)	(0.1894)	(0.2313)	(0.526)	(0.548)	(1.392)	(1.009)
Importance sampling								
500	−1.932	1.695	0.1897	0.2318	0.082	0.084	0.062	0.053
1000	−1.932	1.695	0.1896	0.2318	0.082	0.084	0.062	0.053
5000	−1.932	1.695	0.1896	0.2317	0.084	0.085	0.062	0.054

†We performed the experiment twice for the Hastings–Metropolis algorithm. The first time we used fixed starting values for the Markov chain, and the second time the starting values were generated according to the invariant distribution, i.e. the Markov chain was started at stationarity. The results under stationarity are given in parentheses. The standard errors of the entries can be calculated by using formulae that are analogous to those given in Table 3.

this case, or that there is a central limit theorem, but $s\sqrt{m}$ does not stabilize until $m > 5000$.

The instability of the values of $s\sqrt{m}$ makes it difficult to decide what multiple of our sequence of ms is appropriate. This issue is also complicated by the lack of proportionality of Monte Carlo variances from rejection sampling and Hastings–Metropolis sampling. For example, when $m = 5000$ the ratios of variances vary from $(0.544/0.076)^2 = 52.6$ to $(1.40/0.059)^2 = 563$, which suggests that m needs to be at least 50 times greater for the Hastings–Metropolis algorithm. Multiplying 6.5 min by this factor results in a computing time far in excess of the 150 min for our rejection sampling algorithm with an acceptance rate of only 1.5%. We note here that the maximization step of the Monte Carlo EM algorithm involves fitting a generalized linear model with an offset term, requiring results from all m simulations to be saved. Thus, extremely large values of m can lead to storage problems. For this reason we could not even come close to convergence by using McCulloch's method.

We also applied our Monte Carlo EM algorithm based on importance sampling. A multivariate Student t importance density with 40 degrees of freedom was used. (The optimal degrees of freedom is another topic requiring further investigation.) As before we used $\alpha = 0.25$, $k = 5$ and $\delta_2 = 0.002$, and started with $m = 100$. The importance sampling algorithm converged in 48 iterations and took only about 5 min! At the final iteration, m was 28099 and $\psi^{(48)} = (-1.934 \quad 1.694 \quad 0.1891 \quad 0.2316)$. Thus, in this example, the Monte Carlo EM algorithm using importance sampling is about 30 times faster than the algorithm that uses rejection sampling (and much more than 30 times faster than McCulloch's algorithm). Table 4 also includes a section for our importance sampling method. Note that the asymptotic standard errors for importance sampling are almost the same as for 'exact' rejection sampling. Thus, the huge time difference is not surprising since generating multivariate Student t-variates is much faster than using the rejection sampler to make draws from h.

An explanation for the efficiency of importance sampling in this setting is that the large binomial sample sizes make the Laplace approximations to the conditional mean and

variance of $\mathbf{u}$ given $\mathbf{y}$ extremely accurate. Thus the importance distribution closely resembles the true conditional distribution in this setting. The use of the marginal random effects distribution as a candidate for rejection sampling is inefficient for the same reason; because of the large sample sizes, the marginal and conditional distributions are not similar. Thus, in many cases our rejection sampling and importance sampling methods complement one another. The former is efficient when the sample sizes are small, whereas the latter is more efficient with large sample sizes. However, when the likelihood function involves high dimensional integrals, this informal argument breaks down because the acceptance rate for rejection sampling can be very low. Such is the case with the salamander data, which are analysed in the next section.

7.3. Salamander data

In this section, the salamander data of McCullagh and Nelder (1989), p. 439, are considered. These data have been analysed by many researchers (e.g. Karim and Zeger (1992), McCulloch (1994) and Lee and Nelder (1996)) and consist of three separate experiments, each performed according to the design given in McCullagh and Nelder (1989), Table 14.3. Each experiment involved matings among salamanders in two closed groups. Both groups contained five species R females, five species W females, five species R males and five species W males. Within each group, only 60 of the possible 100 heterosexual crosses were observed owing to time constraints. Thus, each experiment resulted in 120 binary observations indicating which matings were successful and which were not.

Following McCullagh and Nelder (1989), p. 441, we model the data as if different sets of 40 salamanders were used in each experiment, i.e. we ignore the fact that the same salamanders were used in the first two experiments. Let y_{ij} be the indicator of a successful mating between female i and male j for $i, j = 1, 2, \ldots, 60$, where only 360 ($= 120 \times 3$) of the (i, j) pairs are relevant. Let u_i^{f} denote the (random) effect that the ith female salamander has across matings in which she is involved, and define u_j^{m} similarly for the jth male. We assume that the y_{ij} are conditionally independent and that $y_{ij}|u_i^{\mathrm{f}}, u_j^{\mathrm{m}} \sim \mathrm{Bernoulli}(\pi_{ij})$ where

$$\eta_{ij} = \log\left(\frac{\pi_{ij}}{1-\pi_{ij}}\right) = \mathbf{x}_{ij}^{\mathrm{T}}\boldsymbol{\beta} + u_i^{\mathrm{f}} + u_j^{\mathrm{m}}, \tag{14}$$

$\mathbf{x}_{ij}^{\mathrm{T}}$ being a 1×4 row vector indicating the type of cross and $\boldsymbol{\beta} = (\beta_{\mathrm{R/R}} \;\; \beta_{\mathrm{R/W}} \;\; \beta_{\mathrm{W/R}} \;\; \beta_{\mathrm{W/W}})^{\mathrm{T}}$ is an unknown regression parameter. (The symbol W/R, for example, denotes a cross between a species W female and a species R male.) Finally we assume that the u_i^{f} are a random sample from the $N(0, \sigma_{\mathrm{f}}^2)$ distribution and independent of the u_j^{m} which are assumed to be a random sample from the $N(0, \sigma_{\mathrm{m}}^2)$ distribution. This model is the logit–binomial analogue of McCulloch's (1994) probit–binomial model, which he fitted by using a Gibbs sampling implementation of the Monte Carlo EM algorithm. It is also the frequentist version of Karim and Zeger's (1992) 'model A'.

With obvious notation, let $\mathbf{y}$, $\mathbf{u}^{\mathrm{f}}$ and $\mathbf{u}^{\mathrm{m}}$ denote the full data vector and the female and male random effects vectors. Together, the three experiments involve six closed groups of 20 salamanders, and correspondingly $f(\mathbf{y}, \mathbf{u}^{\mathrm{f}}, \mathbf{u}^{\mathrm{m}}; \boldsymbol{\beta}, \sigma_{\mathrm{f}}^2, \sigma_{\mathrm{m}}^2)$ factors into six pieces. The random effects within each of these six groups are crossed, however, which means that the likelihood involves six intractable 20-dimensional integrals.

The multivariate rejection sampler is very inefficient in this case, which is not surprising given the dimension of the random effects. Hence, we turn to our Monte Carlo EM algorithm based on importance sampling. As in the previous example, we use a multivariate Student t

Table 5. Comparison of estimates for the salamander data

Estimate	$\beta_{R/R}$	$\beta_{R/W}$	$\beta_{W/R}$	$\beta_{W/W}$	σ_f^2	σ_m^2
Maximum likelihood	1.03	0.32	−1.95	0.99	1.40	1.25
Bayesian (Karim and Zeger, 1992)	1.03	0.34	−1.98	1.07	1.50	1.36
Pseudolikelihood (SAS)	0.87	0.28	−1.69	0.95	1.35	0.93

Table 6. Measuring pure Monte Carlo error†

m	$\bar{\beta}_{R/R}$	$\bar{\beta}_{R/W}$	$\bar{\beta}_{W/R}$	$\bar{\beta}_{W/W}$	$\bar{\sigma}_f^2$	$\bar{\sigma}_m^2$	$s_{\beta_{R/R}}\sqrt{m}$	$s_{\beta_{R/W}}\sqrt{m}$	$s_{\beta_{W/R}}\sqrt{m}$	$s_{\beta_{W/W}}\sqrt{m}$	$s_{\sigma_f^2}\sqrt{m}$	$s_{\sigma_m^2}\sqrt{m}$
Importance sampling												
500	1.022	0.322	−1.946	0.994	1.394	1.248	0.44	0.49	0.48	0.45	0.64	0.55
1000	1.023	0.323	−1.948	0.995	1.395	1.248	0.47	0.46	0.47	0.44	0.69	0.55
5000	1.022	0.322	−1.947	0.995	1.396	1.249	0.48	0.49	0.51	0.45	0.71	0.60
Hastings–Metropolis sampling												
500	1.022	0.323	−1.946	0.993	1.389	1.240	0.30	0.28	0.37	0.33	0.67	0.56
1000	1.022	0.323	−1.946	0.994	1.393	1.244	0.30	0.28	0.37	0.33	0.67	0.57
5000	1.023	0.323	−1.947	0.994	1.395	1.246	0.31	0.28	0.39	0.32	0.65	0.57

†The Laplace approximation of the mean of the invariant distribution was used as the starting value for the Hastings–Metropolis Markov chain. The standard errors of the entries can be calculated by using formulae that are analogous to those given in Table 3.

importance density with 40 degrees of freedom. We used $\alpha = 0.25$, $k = 5$, $\delta_2 = 0.005$ and starting values

$$(\beta_{R/R}, \beta_{R/W}, \beta_{W/R}, \beta_{W/W}, \sigma_f^2, \sigma_m^2) = (0, 0, 0, 0, 1, 1).$$

The algorithm converged in 51 iterations and took 80 min. In this example m increased from 1000 at the start to 66169 at the final iteration.

Table 5 shows the maximum likelihood estimates along with the Bayesian estimates based on the non-informative priors reported by Karim and Zeger (1992) and the pseudolikelihood estimates produced by the SAS `%GLIMMIX` macro. Karim and Zeger's (1992) estimates are fairly close to the maximum likelihood estimates, whereas the SAS estimates are not.

Finally, Table 6 contains a pure Monte Carlo error comparison of our algorithm and McCulloch's (1997) algorithm. The values of $s\sqrt{m}$ are stable for the Hastings–Metropolis algorithm, indicating that the central limit theorem probably holds. It appears that our method requires larger values of m to obtain the same level of precision. The algorithms are similar in terms of speed for the same sequence of ms in this example. Thus, McCulloch's algorithm could be faster in this case, but the problem of diagnosing convergence still remains for his method.

8. Discussion

Our Monte Carlo EM algorithm is not restricted to models with normally distributed random effects. For example, Lee and Nelder (1996) suggested an alternative to normality for the distribution of the random effects when the data are binomial. Specifically, suppose that z has a symmetric beta distribution with parameter λ, i.e. the density function of z is

proportional to $\{z(1-z)\}^{\lambda-1}$ in the interval (0, 1). Then we say that $z^* = \log\{z(1-z)^{-1}\}$ has a logistic–beta(λ) distribution. The random variable z^* has a density that is symmetric about 0, and its cumulant-generating function is given by

$$K(t) = -2\log\{\Gamma(\lambda)\} + \log\{\Gamma(\lambda+t)\} + \log\{\Gamma(\lambda-t)\}.$$

It follows that $\mathrm{var}(z^*) = 2\psi'(\lambda)$ where $\psi(\cdot)$ is the digamma function.

Consider the model for the lung cancer data. Suppose that in place of normality we assume that the u_i are a random sample from the logistic–beta(λ_u) distribution and independent of the v_{ij}, which are assumed to be a random sample from the logistic–beta(λ_v) distribution. In terms of the Monte Carlo EM algorithm, moving from normal to logistic–beta random effects requires only two minor changes. First, maximization of Q_m with respect to the variance components can be done in closed form in the normal model, whereas a Newton–Raphson algorithm involving the digamma function and its derivative (McCullagh, 1981) is required in the logistic–beta case. Second, the distribution from which we simulate to estimate Q must be altered. If the Monte Carlo EM algorithm based on rejection sampling is used, then we must find a way to simulate from the logistic–beta distribution. An obvious method is to transform deviates from the appropriate beta distribution. (A more efficient method is to use a rejection sampler with a Cauchy candidate.) If the Monte Carlo EM algorithm based on importance sampling is used, then the Laplace approximations of the mean and variance of $h(\mathbf{u}|\mathbf{y}, \psi^{(r)})$ must be recalculated.

In conclusion, normal theory linear mixed models have a rich history and continue to be tremendously important tools for modelling unobservable random effects and for prediction. Generalized linear mixed models greatly extend the range of applications that can be analysed but involve far more difficult computational issues than the normal theory case. We believe that recent advances in computer hardware combined with the development of new statistical methodology, such as that described in this paper, will create a bright future for generalized linear mixed models. However, there is still a way to go before exact Monte Carlo methods for fitting generalized linear mixed models are competitive with approximate analytical methods in terms of computing speed, although presumably this gap will close as computers become faster. Until such time, approximate analytical methods will remain an attractive practical alternative to the Monte Carlo methods. A potential strategy that combines the strengths of the two approaches is to use approximate analytical methods such as `%GLIMMIX` for model selection and exact Monte Carlo methods for fitting the final model.

Acknowledgements

The authors are grateful to Alan Agresti, Brent Coull, Ralitza Gueorguieva, Charles E. McCulloch, the Associate Editor and three referees for helpful comments and suggestions.

References

Bates, D. M. and Watts, D. G. (1981) A relative offset orthogonality convergence criterion for nonlinear least squares. *Technometrics*, **23**, 179–183.

Booth, J. G. and Hobert, J. P. (1998) Standard errors of prediction in generalized linear mixed models. *J. Am. Statist. Ass.*, **93**, 262–272.

Breslow, N. E. and Clayton, D. G. (1993) Approximate inference in generalized linear mixed models. *J. Am. Statist. Ass.*, **88**, 9–25.

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

de Bruijn, N. G. (1981) *Asymptotic Methods in Analysis*. New York: Dover Publications.

Celeux, G. and Diebolt, J. (1985) The SEM algorithm: a probabilistic teacher algorithm derived from the EM algorithm for the mixture problem. *Comput. Statist. Q.*, **2**, 73–82.
Chan, J. S. K. and Kuk, A. Y. C. (1997) Maximum likelihood estimation for probit-linear mixed models with correlated random effects. *Biometrics*, **53**, 86–97.
Chan, K. S. and Geyer, C. J. (1994) Comments on "Markov chains for exploring posterior distributions". *Ann. Statist.*, **22**, 1747–1757.
Chan, K. S. and Ledolter, J. (1995) Monte Carlo EM estimation for time series models involving counts. *J. Am. Statist. Ass.*, **90**, 242–252.
Cox, D. R. and Snell, E. J. (1988) *Analysis of Binary Data*, 2nd edn. London: Chapman and Hall.
Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977) Maximum likelihood from incomplete data via the EM algorithm (with discussion). *J. R. Statist. Soc.* B, **39**, 1–38.
Dorn, H. F. (1954) The relationship of cancer of the lung and the use of tobacco. *Am. Statistn*, **8**, 7–13.
Geweke, J. (1996) *Handbook of Computational Economics*, ch. 15. Amsterdam: North-Holland.
Geyer, C. J. (1992) Practical Markov chain Monte Carlo. *Statist. Sci.*, **7**, 473–482.
Goldstein, H. (1991) Nonlinear multilevel models, with an application to discrete response data. *Biometrika*, **78**, 45–51.
Hobert, J. P. and Geyer, C. J. (1997) Geometric ergodicity of Gibbs and block Gibbs samplers for a hierarchical random effects model. *J. Multiv. Anal.*, to be published.
Karim, M. R. and Zeger, S. L. (1992) Generalized linear models with random effects; salamander mating revisited. *Biometrics*, **48**, 631–644.
Kass, R. E. and Wasserman, L. (1996) The selection of prior distributions by formal rules. *J. Am. Statist. Ass.*, **91**, 1343–1370.
Kuk, A. Y. C. (1995) Asymptotically unbiased estimation in generalized linear models with random effects. *J. R. Statist. Soc.* B, **57**, 395–407.
Lee, Y. and Nelder, J. A. (1996) Hierarchical generalized linear models (with discussion). *J. R. Statist. Soc.* B, **58**, 619–678.
Lin, X. and Breslow, N. E. (1996) Bias correction in generalized linear mixed models with multiple components of dispersion. *J. Am. Statist. Ass.*, **91**, 1007–1016.
Littell, R. C., Milliken, G. A., Stroup, W. W. and Wolfinger, R. D. (1996) *SAS System for Mixed Models*. Cary: SAS Institute.
Little, R. J. A. and Rubin, D. B. (1987) *Statistical Analysis with Missing Data*. New York: Wiley.
Liu, C. and Rubin, D. B. (1994) The ECME algorithm: a simple extension of EM and ECM with faster monotone convergence. *Biometrika*, **81**, 633–648.
Longford, N. T. (1994) Logistic regression with random coefficients. *Comput. Statist. Data Anal.*, **17**, 1–15.
Louis, T. A. (1982) Finding the observed information matrix when using the EM algorithm. *J. R. Statist. Soc.* B, **44**, 226–233.
McCullagh, P. (1981) A rapidly convergent series for computing $\psi(z)$ and its derivatives. *Math. Computn*, **36**, 247–248.
McCullagh, P. and Nelder, J. A. (1989) *Generalized Linear Models*, 2nd edn. London: Chapman and Hall.
McCulloch, C. E. (1994) Maximum likelihood variance components estimation for binary data. *J. Am. Statist. Ass.*, **89**, 330–335.
——— (1997) Maximum likelihood algorithms for generalized linear mixed models. *J. Am. Statist. Ass.*, **92**, 162–170.
McGilchrist, C. A. (1994) Estimation in generalized mixed models. *J. R. Statist. Soc.* B, **56**, 61–69.
McLachlan, G. J. and Krishnan, T. (1997) *The EM Algorithm and Extensions*. New York: Wiley.
Meng, X.-L. and van Dyk, D. (1997) The EM algorithm — an old folk-song sung to a fast new tune (with discussion). *J. R. Statist. Soc.* B, **59**, 511–567.
Meng, X.-L. and Rubin, D. B. (1993) Maximum likelihood estimation via the ECM algorithm: a general framework. *Biometrika*, **80**, 267–278.
Natarajan, R. and McCulloch, C. E. (1995) A note on the existence of the posterior distribution for a class of mixed models for binomial responses. *Biometrika*, **82**, 639–643.
——— (1998) Gibbs sampling with diffuse priors: a valid approach to data-driven inference? *J. Comput. Graph. Statist.*, to be published.
Rai, S. N. and Matthews, D. E. (1993) Improving the EM algorithm. *Biometrics*, **49**, 587–591.
Roberts, G. O. and Rosenthal, J. S. (1997) Geometric ergodicity and hybrid Markov chains. *Electron. Communs Probab.*, **2**, 13–25.
Sammel, M. D., Ryan, L. M. and Legler, J. M. (1997) Latent variable models for mixed discrete and continuous outcomes. *J. R. Statist. Soc.* B, **59**, 667–678.
Schall, R. (1991) Estimation in generalized linear models with random effects. *Biometrika*, **78**, 719–727.
Searle, S. R., Casella, G. and McCulloch, C. E. (1992) *Variance Components*. New York: Wiley.
Sinha, D., Tanner, M. A. and Hall, W. J. (1994) Maximization of the marginal likelihood of grouped survival data. *Biometrika*, **81**, 53–60.
Steele, B. M. (1996) A modified EM algorithm for estimation in generalized mixed models. *Biometrics*, **52**, 1295–1310.
Tanner, M. A. (1996) *Tools for Statistical Inference: Methods for the Exploration of Posterior Distributions and Likelihood Functions*, 3rd edn. New York: Springer.

Tierney, L. (1994) Markov chains for exploring posterior distributions (with discussion). *Ann. Statist.*, **22**, 1701–1762.
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. *J. Am. Statist. Ass.*, **85**, 699–704.
Wolfinger, R. and O'Connell, M. (1993) Generalized linear mixed models: a pseudo-likelihood approach. *J. Statist. Computn Simuln*, **48**, 233–243.
Wu, C. F. J. (1983) On the convergence properties of the EM algorithm. *Ann. Statist.*, **11**, 95–103.
Zeger, S. L. and Karim, R. M. (1991) Generalized linear models with random effects: a Gibbs sampling approach. *J. Am. Statist. Ass.*, **86**, 79–86.