Maximum Likelihood Algorithms for Generalized Linear Mixed Models

by

Charles E. McCulloch
Biometrics Unit
and
Statistics Center
Cornell University
Ithaca, NY 14853

BU-1272-MB November, 1995

ABSTRACT

Maximum likelihood algorithms are described for generalized linear mixed models. We show how to construct a Monte Carlo version of the EM algorithm, propose a Monte Carlo Newton-Raphson algorithm and evaluate and improve the use of importance sampling ideas. Calculation of the maximum likelihood estimates are shown to be feasible for a wide variety of problems where they were not previously. We also use the Newton-Raphson algorithm as a framework to compare maximum likelihood with the "joint-maximization" or penalized quasi-likelihood methods and explain why the latter can perform poorly.

Keywords: Monte Carlo EM, Newton-Raphson, Metropolis-Hastings algorithm, importance sampling, simulated maximum likelihood, joint-maximization algorithms, penalized quasi-likelihood.

1. INTRODUCTION

Generalized linear mixed models (GLMMs) are a natural outgrowth of both linear mixed models and generalized linear models. As such, they are of wide applicability and practical importance (e.g., Breslow and Clayton, 1993). GLMMs enable the accommodation of non-normally distributed responses, specification of a possibly nonlinear link between the mean of the response and the predictors, and can model overdispersion and correlation by incorporating random effects. While maximum likelihood and variants are standard for both linear mixed models (e.g. REML) and generalized linear models (e.g., logistic regression), its use in GLMMs has been limited to simple models due to the need to numerically evaluate high dimensional integrals.

To avoid these computational problems several approaches have been proposed. McCulloch (1994) describes a Monte Carlo EM (MCEM) approach which can handle complicated fixed and random effects structure but is limited to a binary response with a probit-link. "Joint-maximization" algorithms have been proposed by a number of authors (Gilmour, Anderson and Rae, 1984; Harville and Mee, 1984; Schall, 1991). These are approximate versions of the mixed model equations of Henderson *et al* (1959) which arise from maximizing the joint distribution of the observed data and random effects with respect to the parameters *and* the random effects. Others (Breslow and Clayton, 1993; Wolfinger, 1994) have arrived at essentially the same computational algorithm via different justifications. We compare these methods with ML in Sections 5 and 6. Generalized estimating equations approaches (Diggle, Liang and Zeger, 1994) are useful for longitudinal data situations and have attractive robustness properties, but may not be applicable in other situations and sometimes suffer from a lack of efficiency (Fitzmaurice, 1995). As a general approach to difficult ML problems Tanner (1993) and Diggle, Liang and Zeger (1994) have suggested using a Bayesian paradigm with flat or diffuse priors to approximate ML estimates. This will often be inappropriate for models with random effects (such as we are interested in here) since the posterior may not exist for diffuse priors (Natarajan and McCulloch, 1995; Hobert and Casella, 1996). This may not be detected when using computational techniques such as the Gibbs sampler and wrong estimates can result.

In this paper we show how an MCEM algorithm can be constructed, propose a new procedure, called Monte Carlo Newton-Raphson (MCNR), and evaluate and improve the use of simulated maximum likelihood methods. We also use the Newton-Raphson (NR) algorithm as a framework within which to compare ML with joint-maximization approaches. While MCEM algorithms are not new (Tanner, 1993; Ledholter and Chan, 1994), those which have been proposed are not directly applicable to the class of models considered here. We show how the incorporation of a Metropolis-Hastings step allows construction of an MCEM algorithm for ML in GLMMs. Geyer and Thompson (1992) and Gelfand and Carlin (1993) have developed the use of simulation to directly approximate the likelihood and have suggested but not systematically investigated its use in finding maximum likelihood estimates. We demonstrate that these methods may not work well for GLMMs and suggest an improvement by preceding them with either MCEM or MCNR and a Metropolis step.

In Section 2 we define our GLMM and establish notation. In Section 3, the three basic methods are given: we show how to construct the EM algorithm, develop the Monte Carlo Newton-Raphson algorithm and adapt simulated maximum likelihood (SML) for our class of models. Section 4 considers a data set and a logit-normal model which demonstrates some basic properties of the methods. We then propose a hybrid method combining MCNR with SML. Section 5 uses the NR algorithm to compare ML with the joint-maximization algorithms. Some simulations comparing the methods are given in Section 6 and Section 7 discusses convergence issues and offers conclusions.

2. THE BASIC MODEL AND NOTATION

We consider the following class of models. Let Y be the observed data vector and, conditional on the random effects, u, we assume that the elements of Y are independent and drawn from a distribution in the exponential family, which, for simplicity of exposition, we take with canonical link. To complete the specification we assume a distribution for u, depending on parameters, D:

$$\begin{aligned} &f_{y_i|u}(y|u,\beta,\phi) = \exp\{(y\eta_i - c(\eta_i))/a(\phi) + d(y,\phi)\} \\ &u \sim f_u(u|D) \end{aligned} \tag{1}$$

Here $\eta_i = x_i'\beta + z_i'u$ with x_i' being the i*th* row of X, the model matrix for the fixed effects, and likewise with z_i' being the i*th* row of Z, the model matrix for the random effects. The likelihood for (1) is given by

$$L(\beta,\phi,D|y) = \int \prod_{i=1}^{n} f_{y_i|u}(y_i|u,\beta,\phi) f_u(u|D) du, \tag{2}$$

which cannot usually be evaluated in closed form and has an integral with dimension equal to the number of levels of the random factors, u. Our goal is to develop algorithms to calculate fully parametric ML estimates based on the likelihood (2).

3. THREE ALGORITHMS

In this section we develop the three main algorithms for ML in model (1): MCEM, MCNR, and SML.

3.1 Monte Carlo EM

To set up the EM algorithm we consider the random effects, u, to be the missing data. The complete data, W, is then W=(Y,u) and the complete data loglikelihood is given by

$$\ln L_W = \sum_i \ln f_{y_i|u}(y_i|u,\beta,\phi) + \ln f_u(u|D). \tag{3}$$

This choice of missing data has two advantages. First, upon knowing the u's, the Y_i's are independent. Secondly, The M step of the EM algorithm maximizes (3) with respect to β, ϕ, and D. Since β and ϕ only enter the first term, the M step with respect to β and ϕ uses only $f_{y|u}$ (the generalized linear model portion of the likelihood) and so it is similar to a standard generalized linear model computation with the values of u treated as known. Maximizing with respect to D is just ML using the distribution of u after replacing sufficient statistics (in the case where f_u is in the exponential family) with their conditional expected values. The EM algorithm then takes the following form.

1. Choose starting values $\beta^{(0)}, \phi^{(0)}$, and $D^{(0)}$. Set m=0.

2. Calculate (with expectations evaluated under $\beta^{(m)}, \phi^{(m)}$, and $D^{(m)}$).
   a. $\beta^{(m-1)}$ and $\phi^{(m-1)}$ which maximize $E[\ln f_{y|u}(y|u,\beta,\phi)|y]$,
   b. $D^{(m-1)}$ which maximizes $E[\ln f_u(u|D)|y]$,
   c. Set m=m+1.

3. If convergence is achieved, declare $\beta^{(m+1)}$, $\phi^{(m+1)}$, and $D^{(m+1)}$ to be MLEs, otherwise return to step 2.

In general, neither of the expectations in 2a. or 2b. can be computed in closed form for the model (1). This is because the conditional distribution of $u|y$ involves f_y, i.e., the likelihood which we are trying to avoid calculating directly.

However, it *is* possible to produce random draws from the conditional distribution of $u|y$ by using a Metropolis algorithm (Tanner, 1993), which does not require specification of f_y. One can then form Monte Carlo approximations to the required expectations.

To specify the Metropolis algorithm we specify the candidate distribution, $h_u(u)$, from which potential new values are drawn and the acceptance function which gives the probability of accepting the new value (as opposed to keeping the previous value). If we choose f_u as the candidate distribution then the acceptance function takes a particularly neat form. Let u denote the previous draw from the conditional distribution of $u|y$ and generate a new value, u_k^*, for the k*th* component of u using the candidate distribution. If we denote $u^*=(u_1,u_2,\ldots u_{k-1},u_k^*,u_{k+1},\ldots,u_q)$, then we accept u^* as the new value with probability $A_k(u,u^*)$ and otherwise we retain u. Here $A_k(u,u^*)$ is given by:

$$A_k(u,u^*) = \min\left\{1, \frac{f_{u|y}(u^*|y,\beta,\phi,D)h_u(u)}{f_{u|y}(u|y,\beta,\phi,D)h_u(u^*)}\right\}. \qquad (4)$$

Upon choosing $h_u = f_u$, the second term in braces in (4) simplifies to

$$\frac{f_{u|y}(u^*|y,\beta,\phi,D)h_u(u)}{f_{u|y}(u|y,\beta,\phi,D)h_u(u^*)} = \frac{\prod_{i=1}^{n} f_{y_i|u}(y_i|u,\beta,\phi)f_u(u^*|D)f_u(u|D)}{\prod_{i=1}^{n} f_{y_i|u}(y_i|u^*,\beta,\phi)f_u(u|D)f_u(u^*|D)}$$

$$= \frac{\prod_{i=1}^{n} f_{y_i|u}(y_i|u^*,\beta,\phi)}{\prod_{i=1}^{n} f_{y_i|u}(y_i|u,\beta,\phi)}. \qquad (5)$$

This calculation only involves the specification of the generalized linear model portion of the model, namely the conditional distribution of $y|u$.

Incorporating the Metropolis step into the EM algorithm gives an MCEM algorithm as follows:

1. Choose starting values $\beta^{(0)}, \phi^{(0)}$, and $D^{(0)}$. Set m=0.

2. Generate N values, $u^{(1)}, u^{(2)}, \ldots, u^{(N)}$, from $f_{u|y}(u|y,\beta^{(m)},\phi^{(m)},D^{(m)})$ using the Metropolis algorithm described above.

   a. Choose $\beta^{(m+1)}$ and $\phi^{(m+1)}$ to maximize a Monte Carlo estimate of

   $$\mathrm{E}[\ln f_{y|u}(y|u,\beta,\phi)|y], \text{ i.e., maximize} \frac{1}{N}\sum_{k=1}^{N} \ln f_{y|u}(y|u^{(k)},\beta,\phi), \qquad (6)$$

   b. Choose $D^{(m+1)}$ to maximize $\frac{1}{N}\sum_{k=1}^{N} \ln f_u(u^{(k)}|D)$,

   c. Set m=m+1

3. If convergence is achieved, declare $\beta^{(m+1)}$, $\phi^{(m+1)}$, and $D^{(m+1)}$ to be MLEs, otherwise return to step 2.

While computationally intensive, this approach remains feasible for a variety of data configurations. We demonstrate its performance in Section 6.

3.2 Monte Carlo Newton-Raphson

EM is a standard technique to use for linear mixed models, but generalized linear models are usually fit using a Newton-Raphson or scoring algorithm. It thus makes sense to develop a simulation analog of the Newton-Raphson approach for fitting GLMMs. We start by noting that whenever the marginal density of Y is formed as a mixture as in (2) with separate parameters for $f_{y|u}$ and f_u then the ML equations for $\theta=(\beta, \phi)$ and D take the following form:

$$\mathrm{E}\left[\frac{\partial \ln f_{y|u}(y|U,\theta)}{\partial \theta}|y\right] = 0 \tag{7a}$$

$$\mathrm{E}\left[\frac{\partial \ln f_{u}(U|D)}{\partial D}|y\right] = 0. \tag{7b}$$

Equation (7b) only involves the distribution of u and is often fairly easy to solve, e.g., when the distribution is normal. On the other hand, (7a) is amenable to a Newton-Raphson or scoring approach exactly as it is for a standard generalized linear model. Expanding $\frac{\partial \ln f_{y|u}(y|U,\theta)}{\partial \beta}$ as a function of β around the value β_0 gives:

$$\frac{\partial \ln f_{y|u}(y|U,\theta)}{\partial \beta} \cong \frac{\partial \ln f_{y|u}(y|U,\theta)}{\partial \beta}\bigg|_{\theta=\theta_0} + \frac{\partial^2 \ln f_{y|u}(y|U,\theta)}{\partial \beta \partial \beta'}\bigg|_{\theta=\theta_0} (\beta - \beta_0).$$

Specializing this to model (1), and noting that one term has a conditional expected value of zero just as in the generalized linear model derivation (McCullagh and Nelder, 1989, p.42), the formula for a scoring type algorithm becomes:

$$\frac{\partial \ln f_{y|u}(y|U,\theta)}{\partial \beta} \cong X'W(\theta_0,U)/a(\phi)\frac{\partial \eta}{\partial \mu}\bigg|_{\theta=\theta_0} (Y - \mu(\theta_0,U)) - X'W(\theta_0,U)/a(\phi)X(\beta - \beta_0), \tag{8}$$

where $\mu_i(\theta,u) = \mathrm{E}[Y_i|u]$, $W(\theta.u)^{-1} = diag\{(\partial\eta_i/\partial\mu_i)^2 \mathrm{var}(Y_i|u)\}$ and $\partial\eta/\partial\mu = diag\{\partial\eta_i/\partial\mu_i\}$. Using this approximation in (7a) leads to an iteration equation of the form:

$$\beta^{(m+1)} = \beta^{(m)} + \mathrm{E}[X'W(\theta^{(m)},U)X|y]^{-1}X'(\mathrm{E}[W(\theta^{(m)},U)\frac{\partial \eta}{\partial \mu}\bigg|_{\theta=\theta_0} (y - \mu(\beta^{(m)},U))|y]). \tag{9}$$

This analog of scoring would proceed by iteratively solving (7b), (9), and an equation for ϕ. An advantage of the scoring approach over MCEM is that it makes automatic the maximization step in 2a. of (6).

Again, the expectations cannot typically be evaluated in closed form which leads to our Monte Carlo Newton-Raphson (MCNR) approach:

1. Choose starting values $\beta^{(0)}, \phi^{(0)}$, and $D^{(0)}$. Set m=0.

2. Generate N values, $u^{(1)}, u^{(2)}, \ldots, u^{(N)}$, from $f_{u|y}(u|y, \beta^{(m)}, \phi^{(m)}, D^{(m)})$ using the Metropolis algorithm described above and use them to form Monte Carlo estimates of the expectations (denoted below as $\tilde{\mathrm{E}}[\cdot]$).

a. Calculate

$$\beta^{(m+1)} = \beta^{(m)} + \tilde{\mathrm{E}}[X'W(\theta^{(m)}, U)X|y]^{-1} X'(\tilde{\mathrm{E}}[W(\theta^{(m)}, U)\frac{\partial \eta}{\partial \mu}\Big|_{\theta=\theta_0} (y - \mu(\beta^{(m)}, U))|y]), \tag{10}$$

b. Calculate $\phi^{(m+1)}$ to solve $\mathrm{E}[\frac{\partial \ln f_{y|u}(y|U,\theta)}{\partial \phi}|y] = 0$ or a scoring equation,

c. Choose $D^{(m+1)}$ to maximize $\frac{1}{N}\sum_{k=1}^{N} \ln f_u(u^{(k)}|D)$,

d. Set m=m+1.

3. If convergence is achieved, declare $\beta^{(m+1)}$, $\phi^{(m+1)}$, and $D^{(m+1)}$ to be MLEs, otherwise return to step 2.

3.3 Simulated Maximum Likelihood

While both MCEM and MCNR work on the log of the likelihood, Geyer and Thompson (1992) and Gelfand and Carlin (1993) have suggested simulation to estimate the value of the likelihood directly. Starting from (2),

$$\begin{aligned} L(\beta, \phi, D|y) &= \int f_{y|u}(y|u, \beta, \phi) f_u(u|D) du \\ &= \int \frac{f_{y|u}(y|u, \beta, \phi) f_u(u|D)}{h_u(u)} h_u(u) du \\ &\cong \frac{1}{N}\sum_{k=1}^{N} \frac{f_{y|u}(y|u^{(k)}, \beta, \phi) f_u(u^{(k)}|D)}{h_u(u^{(k)})}, \end{aligned} \tag{11}$$

where the u's are selected from the importance sampling distribution, $h_u(u)$, and N is the number of simulated values. This gives an unbiased estimate of the likelihood no matter the choice of $h_u(u)$. The simulated likelihood is then numerically maximized, either after a single simulation, or using multiple simulations in an iterative process where the importance sampling distribution is allowed to depend on the current parameter values.

4. ILLUSTRATION USING A LOGIT-NORMAL MODEL AND A HYBRID ALGORITHM

In this section we give some computational details using a model chosen to be as simple as possible, yet retaining the GLMM structure. These lead to consideration of a hybrid algorithm which begins with MCNR and concludes with SML.

4.1 A simple logit-normal model.

Consider a logit-normal model with a single, normally distributed random effect and a single fixed effect:

$$\begin{aligned} &Y_{ij}|u \sim \textit{indep}\ \text{Bernoulli}(p_{ij}),\ \text{i} = 1,2,\ldots,\ \text{n};\ \text{j} = 1,2,\ldots,\ \text{q}, \\ &\ln(p_{ij}/(1-p_{ij})) = \beta x_{ij} + u_j, \\ &u_j \sim \textit{iid}\ \text{N}(0,\sigma^2). \end{aligned} \tag{12}$$

With a single random effect the likelihood is relatively easy to evaluate numerically (and hence maximize) and for this example it is given by

$$L(\beta,\sigma^2|y) = \prod_{j=1}^{q}\int_{-\infty}^{\infty}\prod_{i=1}^{n}\frac{\exp\{y_{ij}(\beta x_{ij}+u_j)\}}{1+\exp\{y_{ij}\beta x_{ij}+u_j\}}\frac{e^{-u_j^2/2\sigma^2}}{(2\pi\sigma^2)^{1/2}}du_j. \tag{13}$$

This can be evaluated by Gauss-Hermite quadrature (Abramowitz and Stegun, 1959) and we can thus compare MCEM, MCNR, SML and the MLE.

For the Metropolis algorithm we chose the candidate distribution, $h_u(u)$, in (4) to be $N(0,\sigma^2)$ and the acceptance function is thus

$$A_k(u,u^*) = \min\left\{1, e^{y_{+k}(u_k^*-u_k)}\prod_i\frac{1+e^{\beta x_{ij}+u_k}}{1+e^{\beta x_{ij}+u_k^*}}\right\},$$

where $y_{+k}=\Sigma_i y_{ik}$. To find $\beta^{(m+1)}$ in step 2a. of (6) we maximize

$$\frac{1}{N}\sum_{k=1}^{N}\beta\Sigma_{i,j}y_{ij}x_{ij} + \Sigma_j y_{+j}u_j^{(k)} - \Sigma_{i,j}\ln(1+\exp\{\beta x_{ij}+u_j^{(k)}\})$$

while the Newton-Raphson iteration in step 2a. of (10) is

$$\beta^{(m+1)} = \beta^{(m)} + \text{E}[X'W(\beta^{(m)},U)X|y]^{-1}X'(y - \text{E}[\mu(\beta^{(m)},U)|y]),$$

where $\mu_i(\beta,u) = 1/(1+\exp\{-\beta x_{ij}-u_j\})$ and $W(\beta,u) = diag\{\mu_i(\beta,u)(1-\mu_i(\beta,u))\}$. For both MCEM and MCNR the update for σ^2 (2b. of either (6) or (10)) is

$$\sigma^{2(m+1)} = \frac{1}{N}\sum_{k=1}^{N}(\Sigma_j u_j^{(k)2}) / q .$$

For SML there is the question as to what to use as the importance sampling distribution. Though perhaps giving SML an unfair advantage we chose the importance sampling distribution as $N(0,\sigma^2)$, with σ^2 set equal to the true value. This seemed like it would make a good importance sampling distribution and matches with a common suggestion in simulation based methods (e.g., Tanner, 1993).

Figure 1 shows a plot of the three methods of calculating the MLE for a representative data set simulated from model (12) with $\beta=5$, $\sigma^2=0.5$, $x_{ij}=i/15$, $n=15$, and $q=10$. The MLE of β was found to be 3.50 by direct numerical maximization of the likelihood using the routine OPTMUM from GAUSS (Aptech Systems, 1992). This is indicated in Figure 1 by a solid line. All the methods were started at $\beta=2$. Several facts are clear from the plot and are representative of samples from this model:

1. SML using the true distribution as the importance sampling distribution performs poorly. Using a large number of replications in an attempt to achieve accuracy, SML is much slower than either MCEM or MCNR, but converges to a value much farther from the MLE. The optimal importance sampling distribution (optimal in the sense it estimates the value of the likelihood with zero variance at the MLE) is $f_{u|y}$, evaluated at the MLEs. Clearly this is impossible to use since we do not know the value of the MLE and we cannot calculate the conditional distribution. Unfortunately, importance sampling distributions which are far from optimal usually lead to erroneous estimates as in the example. This has also been noticed by Geyer (1994).

2. MCEM and MCNR reach the neighborhood of the MLEs very quickly (in about a minute), but continue to show random variation. The number of replications required to get MCEM or MCNR to converge with four or three decimal accuracy would be very large.

4.2 A Hybrid Algorithm

The preceding observations suggest that a hybrid algorithm would be advantageous. A preliminary stage of MCEM or MCNR can be run which yields both rough estimates of the MLEs and a sample of observations from $f_{u|y}$ at those estimates. These can also be used to approximate the optimal importance sampling distribution for SML. The added advantage of such a hybrid approach is that an estimate of the value of the likelihood is a byproduct of the final SML round. This would not be available from either MCEM or MCNR.

5. COMPARISON OF ML WITH JOINT MAXIMIZATION METHODS

In this section we use equation (9) of the NR algorithm (exact, not Monte Carlo version) and the ML equation, (7b), to compare ML methods with "joint maximization"

algorithms. These have arisen via several justifications: as an approximation to and by analogy with the mixed model equations of Henderson, *et al* (1959), as penalized quasi-likelihood estimators (Breslow and Clayton, 1993), and as Laplace approximations (Wolfinger, 1993). The quasi-likelihood justification is attractive because it suggests that it is "not necessary to specify the distribution of the random effects beyond weak assumptions on the expectation and variance..." (Schall, 1991).

To understand the performance of the joint-maximization (JM) methods it is instructive to compare them in the case of a GLMM with a logit link. In such a case the NR iterations for β (irrespective of the random effects distribution) are:

$$\beta^{(m+1)} = \beta^{(m)} + \mathrm{E}[X'W(\beta^{(m)},u)X|y]^{-1}X'(y - \mathrm{E}[\mu(\beta^{(m)},u)|y]), \qquad (14)$$

where $\mu_i(\beta,u) = 1/(1+\exp\{-x_i'\beta - z_i'u\})$ and $W(\beta,u) = diag\{\mu_i(\beta,u)(1-\mu_i(\beta,u))\}$. The iterations for JM (e.g., Schall, 1991) are similar and are given by:

$$\beta^{(m+1)} = \beta^{(m)} + (X'W(\beta^{(m)},\tilde{u})X)^{-1}X'(y - \mu(\beta^{(m)},\tilde{u})), \qquad (15)$$

where $\tilde{u}$ is the solution to the approximate joint-maximization equations (presumably an attempt to approximate E[$u|y$]). The equations for random effects parameters are a bit different. In our version of NR we use the full ML equation, (7b), to form an iteration. If the random effects distribution is u~N(0,Iσ^2) then (7b) is given by:

$$\sigma^{2(m+1)} = \mathrm{E}[u'u|y]/q, \qquad (16)$$

while Schall (1991) uses

$$\sigma^{2(m+1)} = \tilde{u}'\tilde{u}/(q - v^*), \qquad (17)$$

with v* being the trace of a matrix defined therein. The form of (17) is based on calculations using formulas for E[$u'u|y$] assuming u and Y are jointly normal. This can be a poor approximation for non-normal data and/or non-normally distributed random effects. In fact, for non-normally distributed random effects equation (7b) would take a completely different form, suggesting that neither (16) nor (17) will perform well.

So, if we think of JM techniques as approximating the ML equations, we can see that JM techniques involve two sorts of approximations. First, they depend on joint normal theory calculations for the form of E[$u'u|y$] and second, they assume that $\tilde{u}$ coming from the JM equations will suffice in simultaneously deriving approximations (comparing (14) to (15)) to E[$u'u|y$], $\mathrm{E}[X'W(\beta^{(m)},u)X|y]$, and $\mathrm{E}[\mu(\beta^{(m)},u)|y]$. For large variances of u it is unlikely that the same value of $\tilde{u}$, no matter how derived, will be sufficient for all these approximations.

By comparison, in the linear mixed model, equation (7a) for β is linear in u. Hence the only conditional expectation needed is E[$u|y$] which is found exactly by solving $\partial \ln f_{y,u}/\partial u = 0$, i.e., one of the JM equations. For GLMMs in general, not only does

solving $\partial \ln f_{y,u} / \partial u = 0$ not give E[$u|y$], but, even if we could easily calculate E[$u|y$], it is not the needed ingredient to use in solving (7a).

Of course, as pointed out by Breslow and Clayton (1993), it may be best to consider JM algorithms merely as new methods (rather than as approximations to the ML equations) and evaluate their merit directly. This is done in the next Section.

6. TWO SIMULATION STUDIES

To evaluate the performance of these estimators we ran two small simulation studies. The first compared all of the estimators, while the second focused on the performance of JM estimators as compared to ML estimation when the random effects distribution was not normal.

The first simulation (with 100 replications) compared MCEM, MCNR, SML, JM, MCNR+SML (i.e., a round of SML to follow MCNR) and MCNR+SML+SML (to see if a second round of SML would further improve the estimates). Since SML with a simple importance sampling distribution performed so poorly (Figure 1), we approximated the optimal importance sampling distribution in the SML routines by assuming it was *iid* normal and by using the Metropolis algorithm to estimate the means and variances. The *iid* normality assumption was decided upon after looking at a large number of histograms and scatterplots of $f_{u|y}$. The simulated data were generated from model (12), but with an intercept term, α, estimated. The true values of the parameters were α=0, β=5, and σ^2=1.5. There were q=15 levels of the random effect and n=8 observations per level of the random effect for a total sample size of 120. All the methods were started at α=1, β=4, and σ^2=1.

Figure 2 shows plots of the estimates of β from the six methods against the MLEs. If the methods were all perfect, all the estimates would fall exactly on the y=x line. The JM and SML methods performed poorly, with JM underestimating the true values consistently (due to underestimation of σ^2 - see Figure 3) and SML showing a very large variance, despite the improved importance sampling distribution over that used in Figure 1. MCEM and MCNR performed quite well and generally the SML methods with a start of MCNR performed well, though every once in a while they gave stray values. Figure 3 shows the estimated versus calculated values of the estimate of the variance component with results being very similar to the estimates of β. Figure 4 shows the estimated versus calculated values of the negative of the loglikelihood for the various methods using SML. SML was quite likely to give stray values, but the other methods were virtually always correct; the mean square difference between the calculated and estimated values was 0.12 for MCNR+SML and 0.15 for MCNR+SML+SML. This shows that the values in Figure 3 which gave values different from the MLEs corresponded to nearly equivalent values of the likelihood.

Table 1 gives a numerical summary of the simulation and confirms the graphs. The JM method is badly biased in estimating both the fixed effect and the variance component. The others give approximately the correct value for β on the average. In terms of MSE the JM and SML methods performed poorly but the other four methods were quite good. To get a more detailed idea of which methods were better, we ranked the six methods for

estimating β and σ^2 and the three methods for estimating -loglikelihood for each replication based on their closeness to the MLEs (lower ranks are closer). This shows that the follow-up rounds of SML generally improved the estimates over the preliminary MCNR round to give closer estimates.

The second study focused more on the JM method and its performance as a method of estimation compared to ML when the random effects distribution was non-normal. In order to have a distribution which was highly non-normal and yet easy to perform calculations we used an exponential distribution for the random effects. The layout was chosen to mimic a matched cases, binary data analysis with four treatments and 100 blocks for a total of 400 observations. The model was:

$$\begin{aligned} &Y_{ij}|u \sim \textit{indep}\ \text{Bernoulli}(p_{ij}),\ \text{i} = 1,2,3,4;\ \text{j} = 1,2,\ldots,\ 100, \\ &\ln(p_{ij}/(1-p_{ij})) = \alpha_i + u_j, \\ &u_j \sim \textit{iid Exponential}(\lambda), \end{aligned} \tag{18}$$

with α=(0,0,-5,-5) and λ=3. Numerical ML was used to form the estimates using 20 point Gauss-Hermite quadrature to approximate the integrals when assuming normally distributed random effects and using 15 point Laguerre quadrature (Abramowitz and Stegun, 1959) for the exponentially distributed random effects. The OPTMUM procedure in GAUSS (Aptech Systems, 1992) was used to maximize the likelihoods.

We focus on the estimates of the variance components and correlation structure, since the marginal means in this simple, balanced setup will be very close to the observed proportions for all three methods. As shown in Table 2, JM performed exceedingly poorly, with the estimates of the variance of the random effect being badly biased downward. Full ML assuming a normal distribution for the random effects performed a bit better. This and the previous simulation shows that the approximations involved in JM perform poorly when the random effects variances are not extremely small. This is not a case with an excessively large random effects variance: the marginal correlations between observations on the four treatments within a block vary between 0.2 and 0.5. Both normal ML and JM perform poorly in estimating the marginal correlation structure with the MSE of JM being as much as 20 times larger than ML using the correct model and with ML using the incorrect (normal) model being as much as 9 times worse. Interestingly, in estimating σ^2, ML assuming the (incorrect) normal model gave a smaller mean square error. This was due to an extremely skewed sampling distribution for the estimate of σ^2 under the exponential model which occasionally gave very large values.

7. DISCUSSION AND CONCLUSIONS

A natural question concerns the convergence properties of these algorithms. For sufficiently large simulation sample sizes, MCEM or MCNR would inherit the properties of the exact versions. So MCEM would inherit the likelihood increasing properties of EM and would, under suitable regularity conditions (e.g., Wu, 1983), converge to a local maximum. Newton-Raphson algorithms do not have guaranteed convergence properties when the surfaces to be maximized are not concave.

Unfortunately, in variance components problems, the likelihood surfaces need not be even unimodal (Searle, Casella, and McCulloch, 1992), hence even exact EM algorithms may converge to local, rather than global, maxima and Newton-Raphson algorithms may not converge at all. However, if they do converge, they solve the ML equations (7) and hence converge to a solution of the ML equations. Of course, simulation sample sizes of the order necessary to make MCEM or MCNR essentially deterministic are usually not feasible. In such cases, they clearly will not converge in the usual sense. They will instead get "close" to the correct answer and then vary in the neighborhood of the correct answer (Chan and Ledholter, 1994). This is one reason for suggesting a follow-up round of SML, in order to avoid the complications of deciding whether the stochastic versions of EM or NR have converged.

The variability associated with the estimates from SML can be evaluated directly by repeating the maximization with a new simulation. This could also be used to determine the required simulation sample size. For our simulation studies we conducted a preliminary components of variance analysis to determine sample sizes for both SML and the preliminary Metropolis step which was needed to estimate the optimal importance sampling distribution.

All of the methods seemed robust to the starting values (though they did not always converge to an accurate answer) and only SML (without a preliminary MCNR round) exhibited some convergence problems. Even though the JM methods performed poorly in general, they are fast and might be used to provide starting values for the ML methods. Being based on a linearization which becomes more accurate as the variances become smaller they can be expected to perform well when the variance components are small. This is a well-known case where methods like EM can have problems since the parameter estimate lies on or near the boundary of the parameter space.

In conclusion, we have demonstrated that calculating ML estimates for generalized linear mixed models is feasible using either a Monte Carlo EM algorithm or a Monte Carlo Newton-Raphson algorithm. To make convergence issues clearer, to achieve a slightly more precise estimator and to estimate the value of the maximized likelihood (e.g. for likelihood ratio tests) MCEM or MCNR can be followed by a round of SML. This usually refines the estimates and also gives accurate estimates of the maximized value of the likelihood. Further iteration of SML did not show much improvement over a single round of SML. SML by itself and JM (or penalized quasi-likelihood) methods did not perform well in our simulations. Some reasons for the poor performance of JM were suggested by comparing them to the NR algorithm.

7. REFERENCES

Abramowitz, M. and Stegun, I.A. (1959), *Handbook of Mathematical Functions*, Washington D.C.: National Bureau of Standards.

Aptech Systems, (1992), *GAUSS Manual Version 3.0*, Kent, Washington.

Breslow, N.E. and Clayton, D.G. (1993), "Approximate Inference in Generalized Linear Mixed Models," *Journal of the American Statistical Association*, 88, 9-25.

Chan, K.S. and Ledholter, J. (1995), "Monte Carlo EM Estimation for Time Series Models Involving Counts," *Journal of the American Statistical Association*, 90, 242-252.

Diggle, P.J., Liang, K.-Y., and Zeger, S.L. (1994). *Analysis of Longitudinal Data*. Oxford: Oxford University Press.

Fitzmaurice, G.M. (1995), "A Caveat Concerning Independence Estimating Equations with Multivariate Binary Data," *Biometrics* 51, 309-317.

Gelfand, A.E. and Carlin, B.P. (1993), "Maximum-likelihood Estimation for Constrained- or Missing-data Problems," *Canadian Journal of Statistics*, 21, 303-311.

Geyer, C.J. (1994), "Estimating Normalizing Constants and Reweighting Mixtures in Markov Chain Monte Carlo," Technical Report No. 568, School of Statistics, University of Minnesota.

Geyer, C.J. and Thompson, E.A. (1992), "Constrained Monte Carlo Maximum Likelihood for Dependent Data," *Journal of the Royal Statistical Society, Series B*, 54, 657-699.

Gilmour, A.R., Anderson, R.D., and Rae, A.L. (1985), "The Analysis of Binomial Data by a Generalized Linear Mixed Model," *Biometrika*, 72, 593-599.

Harville, D.A., and Mee, R.W. (1984), "A Mixed Model Procedure for Analyzing Ordered Categorical Data," *Biometrics*, 40, 393-408.

Henderson, C.R., Kempthorne, O., Searle, S.R., and VonKrosigk, C.N. (1959), "Estimation of Environmental and Genetic Trends from Records Subject to Culling," *Biometrics*, 30, 5830-588.

Hobert, J. and Casella, G., "The Effect of Improper Priors on Gibbs Sampling in Hierarchical Linear Mixed Models," To appear in *Journal of the American Statistical Association.*

McCullagh, P and Nelder, J.A. (1989), *Generalized Linear Models, 2nd Ed*, London: Chapman and Hall.

McCulloch, C.E. (1994), "Maximum Likelihood Variance Components Estimation for Binary Data," *Journal of the American Statistical Association*, 89, 330-335.

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.

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.

Tanner, M.A. (1993). *Tools for Statistical Inference: Observed Data and Data Augmentation, 2nd Ed.*, Berlin: Springer-Verlag.

Wolfinger, R. (1994), "Laplace's Approximation for Nonlinear Mixed Models," *Biometrika*, 80, 791-795.

Wu, C.-F. (1983), "On the Convergence Properties of the EM Algorithm," *Annals of Statistics*, 11, 95-103.

Table Captions

Table 1: Estimated average difference from (SEs in parentheses), mean square difference from (SEs in parentheses) and ranking of closeness to (SEs in parentheses) the MLE for six different methods of estimation for the model given by equation (12) - see text for details. JM is the joint-maximization (or penalized quasi-likelihood) method, MCEM is Monte Carlo EM, MCNR is Monte Carlo Newton-Raphson. SML is simulated ML and "+SML" denotes a follow-up round (or two rounds) of SML. MCEM and MCNR used 50, 200, and 5000 replications for iterations 1-19, 20-39, and 40-50 and were stopped after the 50th iteration. SML used 5000 replications and was preceded by a Metropolis step of 250 replications to estimate the optimal importance sampling distribution. 104 replications were performed in all, however, four replications were excluded from consideration because SML did not converge for three and all the methods failed to converge for the fourth.

Table 2: Estimated bias (SEs in parentheses) and mean square errors (SEs in parentheses) of three methods of estimation for data simulated using exponentially distributed random effects from model (18) - see text for details. JM is the joint-maximization (or penalized quasi-likelihood) method, ML exponential is ML assuming exponentially distributed random effects and ML normal is ML assuming normally distributed random effects.

Table 1: Estimated average difference from (SEs in parentheses), mean square difference from (SEs in parentheses) and ranking of closeness to (SEs in parentheses) the MLE for six different methods of estimation for the model given by equation (12).

	Method of Estimation					
	JM	MCEM	MCNR	SML	MCNR+SML	MCNR+SML+SML
Average difference						
β	-0.37	-0.01	-0.01	-0.08	-0.004	-0.007
	(0.05)	(0.01)	(0.02)	(0.05)	(0.02)	(0.02)
σ^2	-0.54	-0.09	-0.11	-0.36	-0.09	-0.08
	(0.13)	(0.07)	(0.09)	(0.15)	(0.09)	(0.10)
-loglik				0.32	-0.07	-0.08
				(0.10)	(0.03)	(0.04)
Mean square difference						
β	0.36	0.02	0.05	0.25	0.04	0.03
	(0.09)	(0.01)	(0.05)	(0.08)	(0.02)	(0.02)
σ^2	1.87	0.59	0.87	2.45	0.81	0.95
	(0.88)	(0.45)	(0.59)	(1.00)	(0.54)	(0.55)
-loglik				1.03	0.11	0.15
				(0.29)	(0.07)	(0.11)
Average rank of difference						
β	4.6	3.0	3.1	5.0	2.8	2.6
	(0.2)	(0.1)	(0.1)	(0.1)	(0.1)	(0.1)
σ^2	3.6	3.1	3.2	5.4	3.0	2.7
	(0.2)	(0.1)	(0.1)	(0.1)	(0.1)	(0.1)
-loglik				2.7	1.8	1.5
				(0.06)	(0.06)	(0.07)

Table 2: Estimated bias (SEs in parenthesis) and mean square errors (SEs in parentheses) of three methods of estimation for the model given by equation (18).

	Method of Estimation		
	ML Exponential	ML Normal	JM
Bias			
σ^2	0.19 (0.05)	-0.58 (0.03)	-1.53 (0.01)
$Corr(Y_{1j}, Y_{2j})$	0.005 (0.003)	0.204 (0.004)	0.047 (0.002)
$Corr(Y_{3j}, Y_{4j})$	-0.017 (0.005)	-0.183 (0.004)	-0.346 (0.002)
Mean Square Error			
σ^2	0.93 (0.14)	0.56 (0.03)	2.35 (0.03)
$Corr(Y_{1j}, Y_{2j})$	0.003 (0.0005)	0.047 (0.002)	0.003 (0.0002)
$Corr(Y_{3j}, Y_{4j})$	0.006 (0.0006)	0.039 (0.002)	0.121 (0.002)

Figure Captions

Figure 1: Convergence of Monte Carlo EM, Monte Carlo Newton-Raphson and simulated ML to the MLE of β. The SML method used 7000 replications from an importance sampling distribution which was N(0,0.5). MCEM and MCNR used 50, 200, and 5000 replications for iterations 1-19, 20-39 and 40+. All the methods were started using β=2. The value of the MLE for this dataset was 3.50.

Figure 2: Plot of the estimated βs versus the ML estimates for six different methods of estimation. JM is the joint-maximization (or penalized quasi-likelihood) method, MCEM is Monte Carlo EM, MCNR is Monte Carlo Newton-Raphson. SML is simulated ML and "+SML" denotes a follow-up round (or two rounds) of SML. MCEM and MCNR used 50, 200, and 5000 replications for iterations 1-19, 20-39, and 40-50 and were stopped after the 50th iteration. SML used 5000 replications and was preceded by a Metropolis step of 250 replications to estimate the optimal importance sampling distribution. All the iterations were started at β=4.

Figure 3: Plot of the estimated σ^2s versus the ML estimates for six different methods of estimation. JM is the joint-maximization (or penalized quasi-likelihood) method, MCEM is Monte Carlo EM, MCNR is Monte Carlo Newton-Raphson. SML is simulated ML and "+SML" denotes a follow-up round (or two rounds) of SML. MCEM and MCNR used 50, 200, and 5000 replications for iterations 1-19, 20-39, and 40-50 and were stopped after the 50th iteration. SML used 5000 replications and was preceded by a Metropolis step of 250 replications to estimate the optimal importance sampling distribution. All the iterations were started at σ^2=1.

Figure 4: Plot of the estimated -loglikelihoods versus actual values at the MLEs for three different methods of estimation. MCNR is Monte Carlo Newton-Raphson, SML is simulated ML and "+SML" denotes a follow-up round (or two rounds) of SML. MCNR used 50, 200, and 5000 replications for iterations 1-19, 20-39, and 40-50 and was stopped after the 50th iteration. SML used 5000 replications and was preceded by a Metropolis step of 250 replications to estimate the optimal importance sampling distribution.

Figure 1: Convergence of MCEM,MCNR, and SML to the MLE of β (=3.50 for this data set).

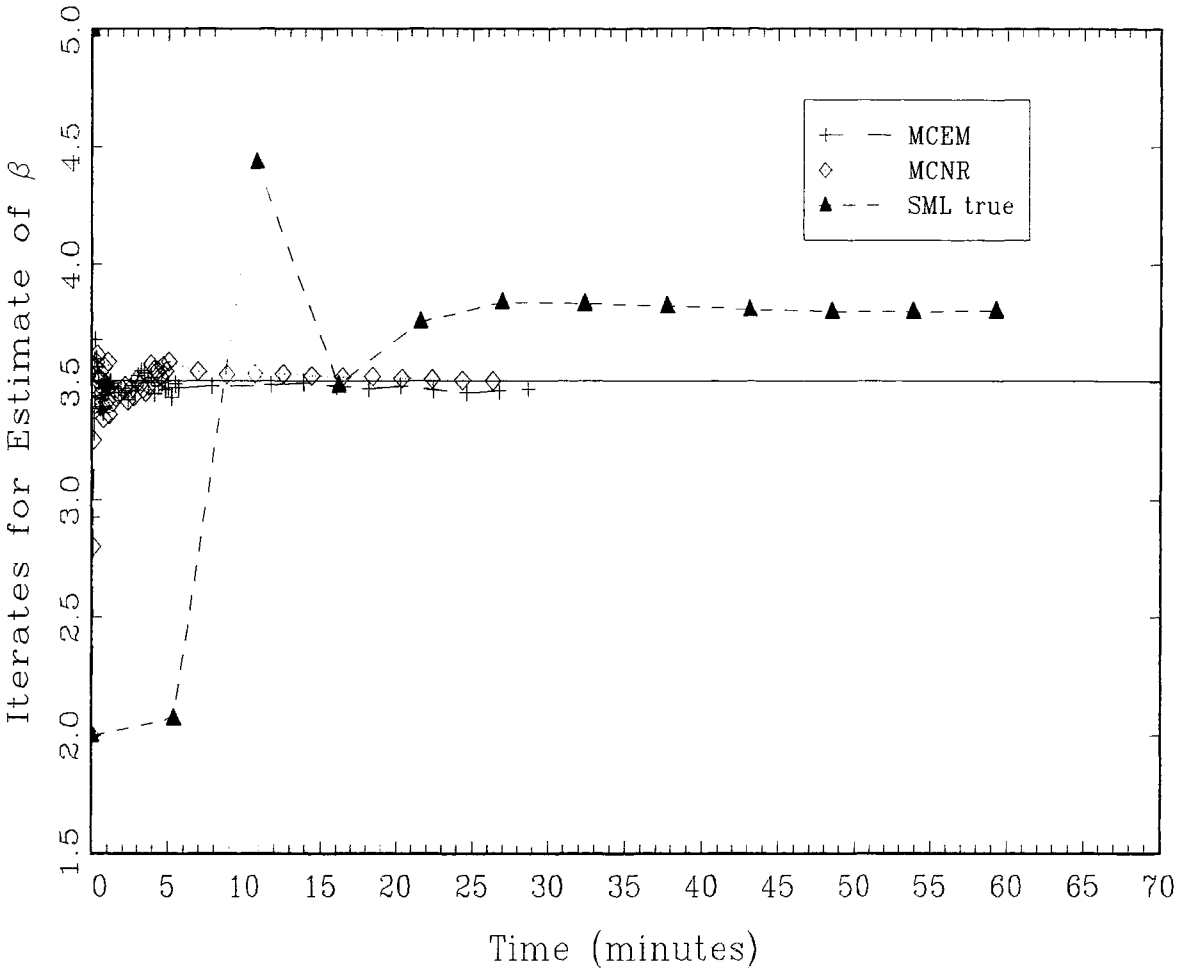

Figure 2: Plot of estimated βs vs ML estimates

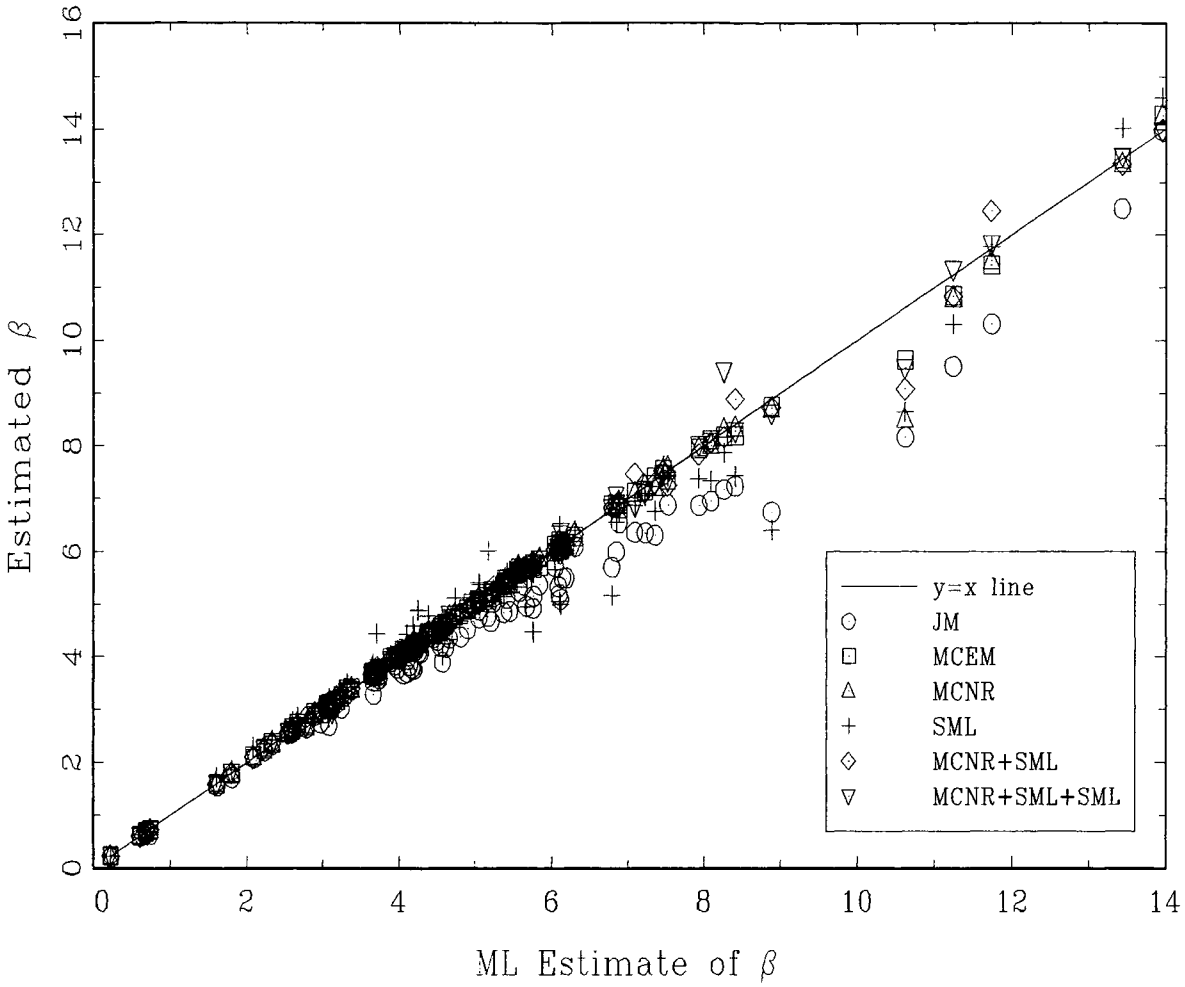

Figure 3: Plot of estimated σ^2s vs ML estimates

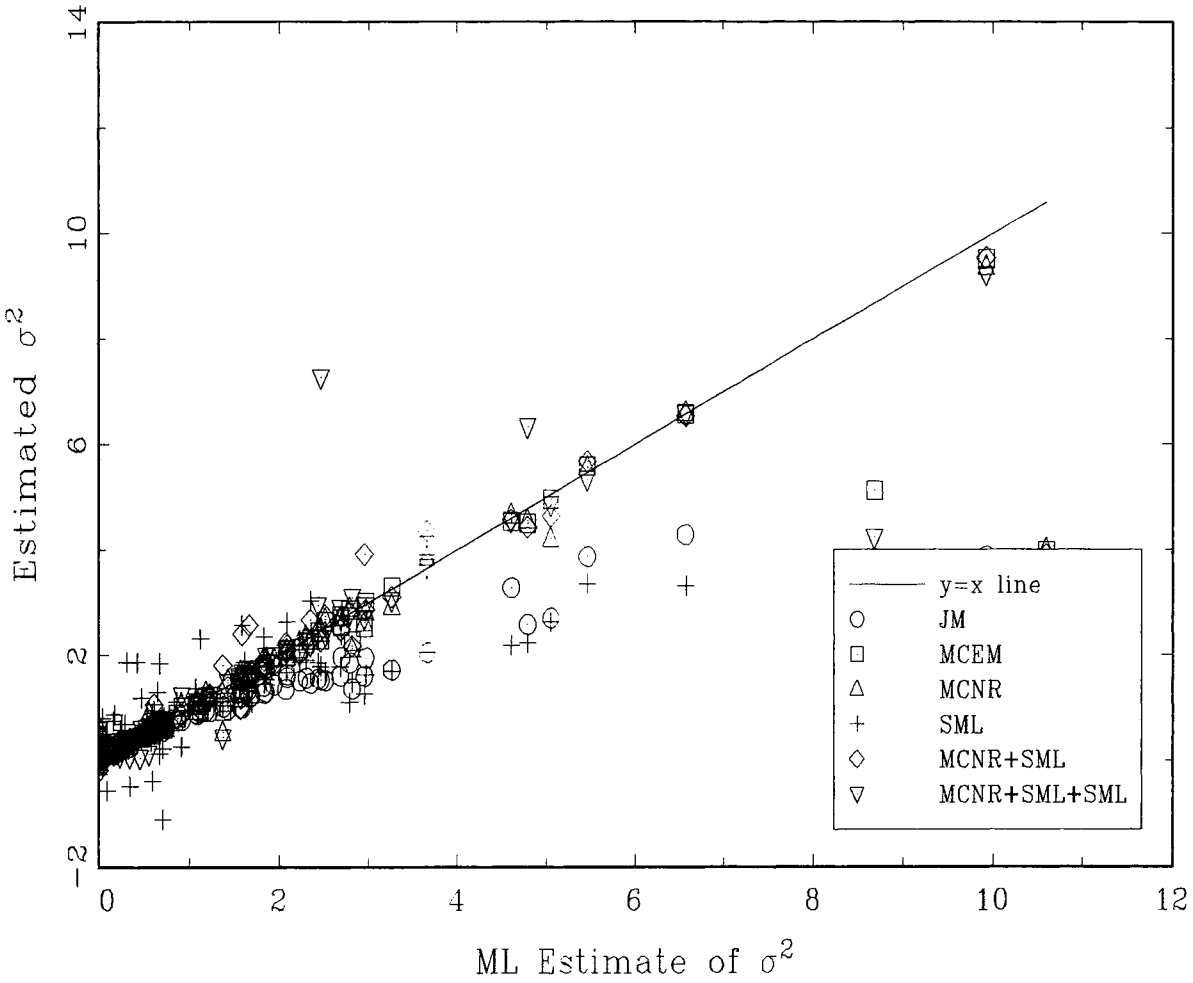

Figure 4: Plot of estimated −loglikelihoods vs ML estimates