A Stochastic Approximation Algorithm for Maximum Likelihood Estimation with Incomplete Data

ABSTRACT

We propose a new stochastic approximation (SA) algorithm for maximum likelihood estimation (MLE) in the incomplete data setting. This algorithm is most useful for problems when the EM algorithm is not possible due to an intractable E-step or M-step. Compared to other algorithms that have been proposed for intractable EM problems such as the MCEM algorithm of Wei and Tanner (1990), our proposed algorithm appears more generally applicable and efficient. The approach we adopt is inspired by the Robbins-Monro (1951)'s stochastic approximation procedure and we show that the proposed algorithm can be used to solve some of the long standing problems in computing an MLE with incomplete data. We prove that in general $O(n)$ simulation steps are required in computing the MLE with the SA algorithm and $O(n\log(n))$ simulation steps are required in computing the MLE using the MCEM and/or the MCNR algorithm, where n is the sample size of the observations. Examples include computing the MLE in the nonlinear error-in-variable model and nonlinear regression model with random effects.

AMS 1985 subject classification: Primary 62A10; secondary 62F10, 60H12.

Keywords: EM algorithm, Incomplete data, Maximum likelihood, Missing data, Stochastic approximation.

1 Introduction.

One of the most significant achievements in modern statistical computation has been the development of the EM algorithm (Dempster *et al.*, 1977; Little and Rubin, 1987) for finding the maximum likelihood estimate (MLE) in the presence of incomplete data. While widely useful, for certain incomplete data problems the EM algorithm has difficulties because either the E-step or the M-step of the EM algorithm becomes intractable. Extensions of the EM algorithm to the case when the M-step becomes intractable have been proposed by Meng and Rubin (1993) and Liu and Rubin (1994). These are respectively, the ECM and ECME algorithms. Both algorithms involve breaking up an M-step in the EM algorithm into several simple steps. Lange (1995a, 1995b) introduced a Gradient algorithm and an improved version of it, where he argues that if one replaces the M-step by a one-step Newton-Raphson iteration, the algorithm still converges to the MLE. Other algorithms proposed by Green (1990) and Titterington (1985) are also discussed in Lange (1995a). Extensions of the EM algorithm to the case when the E-step in the EM algorithm becomes intractable, include the development of the Gibbs sampling method (Geman and Geman, 1984, Gelfand and Smith 1990), the MCEM algorithm proposed by Wei and Tanner (1990), the MCNR algorithm proposed by Geyer and Thompson (1992) and Gelfand and Carlin (1993). In the MCEM or the MCNR algorithms, the intractable expectation step is approximated by Monte Carlo simulation. As indicated in the original paper by Wei and Tanner (1990), the problem associated with the MCEM algorithm is that if one wishes to use the algorithm efficiently and also to achieve some predetermined accuracy, one has to increase the number of simulations in each iteration during the latter stages of the calculation. As we shall see in section 2, the MCNR algorithm faces the same problem in applications. Thus while the algorithm is theoretically valid, sufficient accuracy can, in practice, be hard to achieve for certain problems.

In this paper, we present an algorithm for finding the MLE in the presence of incomplete data. We do not require closed form representations in either steps of the EM algorithm. Therefore our algorithm is applicable to a wide range of difficult incomplete data problems. Examples include the nonlinear error-in-variable model and nonlinear regression model with random effects. The algorithm is based on the general idea of the stochastic approximation procedure first proposed by Robbins and Monro (1951) and will be called the stochastic approximation (SA) algorithm. The key idea of the SA algorithm in the incomplete data setting is to simulate one or a limited number of complete data sets from the conditional distribution given the incomplete data and the current estimate of the parameter and then, specify the next estimate by "shrinking" the current estimate towards the true MLE. Following the general rule of stochastic approximation, we can show that such estimates converge to the MLE.

The rest of this paper is arranged in the following way. In section 2, we review the EM, the MCEM and the Monte Carlo Newton-Raphson (MCNR) algorithms since MCEM and the MCNR are in principle competitors to our approach. In section 3, we propose the SA algorithm for the incomplete data problem and formulate a square root convergence theorem. In section 4, we compare the SA algorithm to the MCEM and MCNR algorithms in terms of the number of simulations needed to achieve the same accuracy. We show that in general the SA algorithm needs $O(n)$ simulation steps in computing a MLE, and the MCEM and MCNR algorithms need $O(n \log(n))$ simulation steps in computing a MLE. Comparison is also made with the Motorette data set where both MCEM and SA algorithms are applied. In section 5, we show that the SA algorithm can be applied to a logistic regression model with covariate measurement error and nonlinear mixed effects model. Note that finding MLE in such models is a long standing problem (Carroll *et al.*, 1984; Stiratelli, Laird and Ware, 1984; Zeger, Liang and Albert, 1988). In section 6, we provide some possible variations, extensions of the SA algorithm and discuss its relation to Gibbs sampling.

2 EM, MCEM and MCNR Algorithms

Let $f(\boldsymbol{x}|\boldsymbol{\psi})$ be a family of sampling densities, and let $\boldsymbol{y} = \boldsymbol{y}(\boldsymbol{x})$ be the incomplete data, where $\boldsymbol{x} \to \boldsymbol{y}$ is a many to one random mapping. In this paper, n denotes the dimension of the observed data vector $\boldsymbol{y}$. As in Dempster *et al.* (1977), we let $p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\psi})$ denote the conditional distribution of $\boldsymbol{X}$ given $\boldsymbol{y}$ and the parameter $\boldsymbol{\psi}$. Let $l(\boldsymbol{\psi}) = l(\boldsymbol{\psi}|\boldsymbol{y})$ be the log-likelihood function based on the observation $\boldsymbol{y}$ and assume $l(\boldsymbol{\psi})$ has a unique maximum at $\boldsymbol{\psi} = \boldsymbol{\psi}_0$. Throughout we use $\boldsymbol{\psi}_0$ or ψ_0 to denote the MLE since it is treated as a constant in this paper.

The key idea of the EM algorithm is that for a given current estimate $\boldsymbol{\psi}_k$ of $\boldsymbol{\psi}_0$, the log-likelihood function $l(\boldsymbol{\psi})$ in the neighborhood of $\boldsymbol{\psi}_k$ has the same behavior as the function

$$Q(\boldsymbol{\psi}, \boldsymbol{\psi}_k) = \int \log f(\boldsymbol{x}|\boldsymbol{\psi}) p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\psi}_k) d\boldsymbol{x}. \tag{1}$$

Therefore maximizing $Q(\boldsymbol{\psi}, \boldsymbol{\psi}_k)$ with respect to $\boldsymbol{\psi}$ provides a reasonable next step estimate of $\boldsymbol{\psi}_0$. The integration in (1) is the E-step and the maximization is the M-step. The EM algorithm is comprised of iterating these two steps. Note that the EM algorithm works best for solving problems where the two steps are relatively easy to implement, for example, if they both yield explicit analytic expressions.

In extending the EM algorithm to the case where the E-step fails to give an explicit expression, Wei and Tanner (1990) suggested substituting the integration in (1) by Monte-Carlo simulation, *i.e.* replace $Q(\boldsymbol{\psi}, \boldsymbol{\psi}_k)$ by

$$Q_m(\boldsymbol{\psi}, \boldsymbol{\psi}_k) = \frac{1}{m} \sum_{i=1}^{m} \log f(\boldsymbol{X}_i|\boldsymbol{\psi}), \tag{2}$$

where $\boldsymbol{X}_1, \cdots, \boldsymbol{X}_m$ is an i.i.d. simulated sample from the conditional distribution $p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\psi}_k)$. This is known as the Monte Carlo EM or MCEM algorithm. To carry out this algorithm, a new sample of m $\boldsymbol{X}$'s needs to be generated for each iteration. Note that in order for the Monte Carlo approximation to be effective, m has to be rather large at the later stages of the iterations. The MCEM algorithm is best applied to problems where as a function of $\boldsymbol{\psi}$, $Q_m(\boldsymbol{\psi}, \boldsymbol{\psi}_k)$ can be maximized in closed form. This is the case for the genetic linkage model, first studied by Rao (1973), and for the linear regression with the Motorette data (Wei and Tanner, 1990).

If simulation is going to be used, one can directly simulate the derivative of the log-likelihood function, *i.e.* approximate $(\partial/\partial\boldsymbol{\psi})l(\boldsymbol{\psi})$ by

$$\frac{\partial}{\partial\boldsymbol{\psi}} l_m(\boldsymbol{\psi}) = \frac{1}{m}\sum_{i=1}^{m}\frac{\partial}{\partial\boldsymbol{\psi}}\log f(\boldsymbol{X}_i|\boldsymbol{\psi}), \tag{3}$$

where $\boldsymbol{X}_1, \cdots, \boldsymbol{X}_m$ is an i.i.d. simulated sample from the conditional distribution $p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\psi})$. Approximation of $(\partial/\partial\boldsymbol{\psi})l(\boldsymbol{\psi})$ by $(\partial/\partial\boldsymbol{\psi})l_m(\boldsymbol{\psi})$ is based on the observation that

$$\frac{\partial}{\partial\boldsymbol{\psi}} l(\boldsymbol{\psi}) = E\left[\frac{\partial}{\partial\boldsymbol{\psi}}\log f(\boldsymbol{X}|\boldsymbol{\psi})\Big|\boldsymbol{y}, \boldsymbol{\psi}\right] \tag{4}$$

if $\boldsymbol{X}$ is distributed as $p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\psi})$. More specifically, equation (4) can be derived by noting that

$$\log f(\boldsymbol{X}|\boldsymbol{\psi}) = l(\boldsymbol{\psi}) + \log p(X|y, \boldsymbol{\psi}). \tag{5}$$

Equation (5) is exactly equation (2.6) of Dempster *et al.* (1977) but with the term $\log p(\cdot|\cdot)$ on the other side of the equation. Taking the derivative with respect to $\boldsymbol{\psi}$ on both sides of equation (5) and then taking expectation with respect to $\boldsymbol{X}$, conditional on $\boldsymbol{y}$ and $\boldsymbol{\psi}$ and under the condition that the order of integration and differentiation can be interchanged, we get (4) by observing that

$$E\left[\frac{\partial}{\partial\boldsymbol{\psi}}\log p(\boldsymbol{X}|\boldsymbol{y}, \boldsymbol{\psi})\Big|\boldsymbol{y}, \boldsymbol{\psi}\right] = 0.$$

Once we approximate $(\partial/\partial\boldsymbol{\psi})l(\boldsymbol{\psi})$ using (3), we can use the Newton-Raphson algorithm to approximate the zeros of $(\partial/\partial\boldsymbol{\psi})l(\boldsymbol{\psi})$. In each Newton-Raphson step, m draws from the conditional distribution are used to estimate the function and its derivative. This algorithm was first proposed by Geyer and Thompson (1992) in an exponential distribution setting and generalized to general constrained likelihood estimation by Gelfand and Carlin (1993). This procedure has been called the Monte Carlo Newton-Raphson algorithm and it is similar to the idea of the MCEM algorithm in the sense that one replaces the integration by Monte-Carlo simulation. MCNR algorithm can also be viewed as Lange (1995a)'s Maximum Gradient algorithm with the Monte Carlo approximation (3) in the E-step.

Since the goal is to find the maximum of $l(\boldsymbol{\psi})$, or the zero of $(\partial/\partial\boldsymbol{\psi})l(\boldsymbol{\psi})$ and not the $l(\boldsymbol{\psi})$ (or derivative of it) itself, the stochastic approximation procedure can also be applied.

3 Stochastic Approximation Procedures

3.1 Introduction to Stochastic Approximation and Motivation of our Algorithm

The Robbins-Monro (1951) procedure is a sequential procedure designed to find the root of a regression function observed with error by sequentially specifying design points $\psi_1, \psi_2, \cdots$ that converge to the root ψ_0. Consider the following regression model

$$Z(\psi) = M(\psi) + \epsilon(\psi), \tag{6}$$

where $M(\cdot)$ is an unknown mean response curve and at each design point ψ, $\epsilon(\psi)$ has mean zero and finite variance. The $\epsilon(\psi)$'s are independent given the ψ's. Suppose that for any fixed design point ψ, we are only able to observe $Z(\psi)$. Suppose also that $M(\psi) = 0$ has a unique solution at $\psi = \psi_0$ and that $(\psi - \psi_0)M(\psi) > 0$ for $\psi \neq \psi_0$. The Robbins-Monro (1951) procedure provides an efficient way to find ψ_0 in this setting.

Let $\{\gamma_k, k \geq 1\}$ be a sequence of positive numbers such that

$$(i) \quad \sum \gamma_k = \infty; \qquad (ii) \quad \sum \gamma_k^2 < \infty. \tag{7}$$

Let ψ_1 be some fixed number (or a random variable with $\mathrm{E}\psi_1^2 < \infty$), and define $\{\psi_k, k \geq 1\}$ by the recursion

$$\psi_{k+1} = \psi_k - \gamma_k Z(\psi_k). \tag{8}$$

Note that if ψ_k lies to the right of ψ_0, then $Z(\psi_k)$ is likely positive and thus ψ_{k+1} lies to the left of ψ_k. The amount γ_k is the shrinking size which goes to zero but cannot converge too fast. The first condition of (7) ensures that the sequence of ψ's reaches the root no matter where it starts and the second condition of (7) ensures the convergence of the sequence.

If the parameter ψ in an incomplete data problem is one dimensional, then the above procedure can be adapted to find the MLE for the incomplete data problem. Going back to the notation defined in Section 2, we can express the problem of maximizing $l(\psi)$ in the regression form by setting

$$Z(\psi) = -\frac{\partial}{\partial \psi} \log f(\boldsymbol{X}_j|\psi), \tag{9}$$

which is observed, and

$$M(\psi) = -\frac{\partial}{\partial \psi} l(\psi), \epsilon(\psi) = -\frac{\partial}{\partial \psi} \log p(\boldsymbol{X}_j|\boldsymbol{y}, \psi), \tag{10}$$

which are treated as unobserved in the regression setting of (6). To be more specific, we can define a sequence $\{\psi_k, k \geq 1\}$ by starting with some initial estimate ψ_1 to ψ_0, the MLE, and define sequentially for $k = 1, 2, \cdots$

$$\psi_{k+1} = \psi_k + \gamma_k \frac{\partial}{\partial \psi} \log f(\boldsymbol{X}^k|\psi)\Big|_{\psi=\psi_k}, \tag{11}$$

where $\boldsymbol{X}^k$ is a random variable generated from the conditional distribution $p(\boldsymbol{x}|\boldsymbol{y}, \psi_k)$ and $\{\gamma_k\}$ is a sequence of constants satisfying (7).

This approach is most useful when $l(\psi)$ is difficult to compute, and one can simulate $\boldsymbol{X}$ from $p(\boldsymbol{x}|\boldsymbol{y}, \psi)$ and can easily evaluate $(\partial/\partial\psi) \log f(\boldsymbol{X}|\psi)$. It requires no integration nor maximization and most importantly, no analytic expressions of the related functions, such as $l(\psi)$ and $p(\boldsymbol{x}|\boldsymbol{y}, \psi)$.

In practice, γ_k is usually taken to be $b_k\ k^{-1}$ where $b_k \to b$ as $k \to \infty$ for some positive value b. For this choice of γ_k, Chung (1954) has shown that $k^{1/2}(\psi_k - \psi_0)$ is asymptotically normal with mean zero for procedure (11) as long as $-b\ (\partial^2/\partial\psi^2) l(\psi)|_{\psi=\psi_0} > 1/2$. Chung (1954) showed that the variance in the limiting distribution is minimized if $-b\ (\partial^2/\partial\psi^2) l(\psi)|_{\psi=\psi_0} = 1$. See also Sacks (1958) for details.

3.2 Stochastic Approximation Algorithm for Maximum Likelihood Estimation with General Incomplete Data

In general incomplete data problems, especially those where simulation based procedures are needed, the parameter $\boldsymbol{\psi}$ is multi-dimensional, and $M(\boldsymbol{\psi}) = -(\partial/\partial\boldsymbol{\psi}) l(\boldsymbol{\psi})$ in (6) is a vector of functions from $\mathcal{R}^d \to \mathcal{R}^d$, where d is the dimensionality of the parameter space. A multi-dimensional version of stochastic approximation procedure can be used to find the MLE of $\boldsymbol{\psi}$. The literature on how one should generalize Robbins-Monro procedure to the multi-dimensional case starts with Blum (1954). We refer the reader to Lai (1985) for a review of work in this direction.

Even though a simple generalization of (11), *i.e.* updating the estimate sequentially with ψ_k, ψ_{k+1} and ψ replaced by $\boldsymbol{\psi}_k$, $\boldsymbol{\psi}_{k+1}$ and $\boldsymbol{\psi}$ respectively in (11), provides a consistent estimate of $\boldsymbol{\psi}_0$, the MLE (see Lai, 1985), we find that the following version of the SA algorithm works much better in practice. It is essentially based on the work of Nevel'son and Has'minski (1973a,b) and Wei (1987).

The algorithm allows the user to choose m and estimates for the information matrix, such as the inverse of the average of the second derivative matrix of the complete data log-likelihood at the previously simulated values, or Louis's estimate (see Remark 2 below) and the Meilijson's estimate (see Remark 3 below). All these estimates mentioned can be updated sequentially in terms of k and therefore in the coding, we do not have to keep all the previously simulated values from the conditional distributions. The m in the algorithm could be any fixed positive constant.

The SA Algorithm for Incomplete data with Multi-dimensional Parameter:

a) Find an initial estimate $\boldsymbol{\psi}_1$;

b) At the kth iteration, $k = 1, 2, \cdots$, generate $\boldsymbol{X}_1^k, \ldots, \boldsymbol{X}_m^k$ from the conditional distribution $p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\psi}_k)$ and define the estimate $\boldsymbol{\psi}_{k+1}$ as

$$\boldsymbol{\psi}_{k+1} = \boldsymbol{\psi}_k + \frac{B_k^{-1}}{km} \sum_{j=1}^{m} \frac{\partial}{\partial \boldsymbol{\psi}} \log f(\boldsymbol{X}_j^k | \boldsymbol{\psi}) \Big|_{\boldsymbol{\psi}=\boldsymbol{\psi}_k}, \tag{12}$$

where $\{B_k, k \geq 1\}$ is a sequence of $d \times d$ positive definite matrices which converges to a positive definite matrix B.

Remark 1. The above algorithm is consistent in the sense that under very general conditions on $M(\boldsymbol{\psi})$ and $\epsilon(\boldsymbol{\psi})$, which relate to $l(\boldsymbol{\psi})$ and $p(\boldsymbol{\psi})$ through (9) and (10), $\boldsymbol{\psi}_k$

is strongly consistent for $\boldsymbol{\psi}_0$. See Lai (1985) and Nevel'son and Has'minski (1973a). The speed of convergence of $\boldsymbol{\psi}_k$ to $\boldsymbol{\psi}_0$ depends on the closeness of the matrix B to the matrix $\mathcal{I}$, the information matrix of the log-likelihood function $l(\boldsymbol{\psi})$ at $\boldsymbol{\psi}_0$. To be more precise, if all the eigenvalues of the matrix $I/2 - B\,\mathcal{I}$ have negative real parts, then we have $\boldsymbol{\psi}_k - \boldsymbol{\psi}_0 = O(k^{-1/2})$. See Nevel'son and Has'minski (1973b).

Remark 2. A sensible choice for the matrix B_k in practice is the estimate based on Louis's missing information principle. The principle provides the identity

$$-\frac{\partial^2}{\partial \boldsymbol{\psi}^2} l(\boldsymbol{\psi}) = -\mathrm{E}\left[\frac{\partial^2}{\partial \boldsymbol{\psi}^2} \log f(\boldsymbol{X}|\boldsymbol{\psi})|\boldsymbol{y}, \boldsymbol{\psi}\right] - \mathrm{Var}\left[\frac{\partial}{\partial \boldsymbol{\psi}} \log f(\boldsymbol{X}|\boldsymbol{\psi})|\boldsymbol{y}, \boldsymbol{\psi}\right].$$

See also Tanner (1993, page 49). Based on this, we take B_k to be A_k when k is small and adapt B_k by $B_k = A_k - D_k$ when k is large, where

$$\begin{aligned} A_k &= -\frac{1}{km} \sum_{i=1}^{k} \sum_{j=1}^{m} \frac{\partial^2}{\partial \boldsymbol{\psi}^2} \log f(\boldsymbol{X}_j^i|\boldsymbol{\psi}_i) \qquad (13) \\ D_k &= \frac{1}{k} \sum_{i=1}^{k} \left(\frac{1}{m} \sum_{j=1}^{m} \frac{\partial}{\partial \boldsymbol{\psi}} \log f(\boldsymbol{X}_j^i|\boldsymbol{\psi}_i)\right)^{\otimes 2} - \left[\frac{1}{km} \sum_{i=1}^{k} \sum_{j=1}^{m} \frac{\partial}{\partial \boldsymbol{\psi}} \log f(\boldsymbol{X}_j^i|\boldsymbol{\psi}_i)\right]^{\otimes 2}, \end{aligned}$$

and $\boldsymbol{X}_1^k, \cdots, \boldsymbol{X}_m^k$ is the sample generated in the step b) of the SA algorithm. We have used the notation $\boldsymbol{a}^{\otimes 2} = \boldsymbol{a} \cdot \mathrm{transpose}(\boldsymbol{a})$ for a vector $\boldsymbol{a}$. The problem of taking B_k as the inverse of $A_k - D_k$ when k is small is that the latter term may not be positive.

Remark 3. In the case that the observation $\boldsymbol{y}$ is composed of independent observations, the moment estimator of the information matrix proposed by Meijlijson (1989) can be used for B_k. Suppose that $\boldsymbol{y} = (y^1, \cdots, y^n)$ and that $\boldsymbol{X}^i = (X^{1,i}, \cdots, X^{n,i})$ is a copy of the simulated complete data at the ith iteration of (12). Here $X^{j,i}$ is a copy of simulated complete data of y^j at ψ_i. Then

$$\frac{\partial}{\partial \boldsymbol{\psi}} \log f(\boldsymbol{X}^i|\boldsymbol{\psi}_i) = \sum_{j=1}^{n} \frac{\partial}{\partial \boldsymbol{\psi}} \log f_j(X^{j,i}|\boldsymbol{\psi}_i).$$

Define

$$\hat{\boldsymbol{s}}_j = \frac{1}{k} \sum_{i=1}^{k} (\partial/\partial \boldsymbol{\psi}) \log f(X^{j,i}|\boldsymbol{\psi}_i).$$

Then we can take B_k as

$$B_k^{-1} = \sum_{j=1}^{n} (\hat{\boldsymbol{s}}_j - \bar{\hat{\boldsymbol{s}}})^{\otimes 2}, \tag{14}$$

where $\bar{\hat{\boldsymbol{s}}}$ is the average of the $\hat{\boldsymbol{s}}_j$'s.

Next we state a central limit theorem for the above proposed procedures. Assumptions 1-3 are needed for the theorem to hold.

Assumption 1. The log-likelihood function $l(\boldsymbol{\psi})$ has Hessian matrix $H(\boldsymbol{\psi})$ which is bounded and satisfies the Hölder condition, *i.e.*, for any vectors $\boldsymbol{\psi}$, $\boldsymbol{\phi}$ in $\mathcal{R}^d$

$$\|H(\boldsymbol{\psi}) - H(\boldsymbol{\phi})\| \leq \delta \|\boldsymbol{\psi} - \boldsymbol{\phi}\|,$$

where $\delta > 0$ is a constant.

Assumption 2. The minus of the Hessian matrix $-H(\boldsymbol{\psi})$ is positive definite in a neighborhood of $\boldsymbol{\psi}_0$.

Assumption 3. The complete data log-likelihood function $\log f(\boldsymbol{x}|\boldsymbol{\psi})$ has bounded second derivatives.

Let $\mathcal{I}$ be the information matrix of the log-likelihood function $l(\boldsymbol{\psi})$ at $\boldsymbol{\psi}_0$, that is $\mathcal{I} = -H(\boldsymbol{\psi}_0)$. Let Σ denote the information matrix of the conditional distribution at $\boldsymbol{\psi} = \boldsymbol{\psi}_0$, that is $\Sigma = \mathrm{E}_{\boldsymbol{\psi}_0}\left[(\partial/\partial\boldsymbol{\psi}) \log p(\boldsymbol{X}|\boldsymbol{y}, \boldsymbol{\psi})|_{\boldsymbol{\psi}=\boldsymbol{\psi}_0}\right]^2$.

Theorem 1 *Under the assumptions 1-3, if the matrix sequence B_k used in (12) satisfies $[\log\log(k)]^{1/2}(B_k - \mathcal{I}) \to 0$ almost surely as $k \to \infty$, then for the $\boldsymbol{\psi}_k$ defined by (12), $k^{1/2}(\boldsymbol{\psi}_k - \boldsymbol{\psi}_0)$ is asymptotically normal with mean 0 and covariance matrix $\mathcal{I}^{-1}\Sigma\mathcal{I}^{-1}/m$ as $k \to \infty$.*

The proof of Theorem 1 is deferred to the appendix.

Remark 4. The condition that $(\log\log(k))^{1/2}(B_k - \mathcal{I}) \to 0$ almost surely as $k \to \infty$ is satisfied for the estimate based on Louis's missing information principle in Remark 2. Based on the result of Nevel'son and Has'minski (1973b) (see also Wei, 1987), we first have $\boldsymbol{\psi}_k - \boldsymbol{\psi}_0 = O(k^{-\alpha})$ almost surely for some $0 < \alpha < 1/2$. Then under Assumptions 1-3 it is not hard to see that $A_k - (\mathcal{I} + \Sigma) = O(k^{-\alpha})$ almost surely and $D_k - \Sigma = O(k^{-\alpha})$ almost surely.

4 On the Number of Simulations

Since the MCEM and the simulated Newton-Raphson (MCNR) algorithm do not require integration, in principle, one can also use these algorithms to solve problems where the SA algorithm would apply. In this section, we give a theoretical analysis of how the SA algorithm reduces the number of simulation steps from the conditional distribution. Our experience shows that where the usual EM algorithm does not apply, simulating from the conditional distribution usually is the most time consuming part of the computation. Note that one simulation step here means simulating once from the conditional distribution $p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\psi})$. This usually requires that the number of simulations be of order n, where n is the dimension of the observed data vector $\boldsymbol{y}$ (see Section 5 for examples).

4.1 Precision of approximations

From Theorem 1, we see that to achieve precision δ in $\boldsymbol{\psi}_k$ to $\boldsymbol{\psi}_0$, k needs to be of the order $1/\delta^2$.

To consider the precision of the final estimate of $\boldsymbol{\psi}_0$ using the MCEM or the MCNR, we first observe that by taking second derivatives on both sides of (5) and then taking expectation,

$$-\frac{\partial^2 Q(\boldsymbol{\phi}, \boldsymbol{\psi}_0)}{\partial \boldsymbol{\phi}^2}\Big|_{\boldsymbol{\phi}=\boldsymbol{\psi}_0} = \mathcal{I} + \Sigma.$$

In the MCEM or the simulated Newton-Raphson algorithm, one needs to approximate $(\partial/\partial\boldsymbol{\phi})Q(\boldsymbol{\phi}, \boldsymbol{\psi})$ by

$$\frac{\partial}{\partial \boldsymbol{\phi}} Q_m(\boldsymbol{\phi}, \boldsymbol{\psi}) = \frac{1}{m}\sum_{i=1}^{m} \frac{\partial}{\partial \boldsymbol{\phi}} \log f(\boldsymbol{X}_i|\boldsymbol{\phi}),$$

where $\boldsymbol{X}_i$ are i.i.d. simulated from the conditional distribution $p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{\psi})$. By the Louis's (1982) missing information principle and the Central Limit Theorem, the variance in this approximation is about Σ/m. To achieve the precision δ in the approximation in the M-step (the variance for the approximation to the estimated parameter in the M-step is approximately $(\mathcal{I}+\Sigma)^{-1}\Sigma(\mathcal{I}+\Sigma)^{-1}/m$, the sample size m needs to be of the order $1/\delta^2$. If one wishes to achieve the precision δ in the final approximation to the MLE, then at some point one needs to have $m = O(1/\delta^2)$ simulations in each iteration for both MCEM and MCNR.

Since the MLE usually estimates the true parameter to the precision of order $n^{-1/2}$ by the Central Limit Theorem, where n is the sample size, it is common to take the precision δ as $cn^{-1/2}$, where c is some small number. If this is the case, SA algorithm requires a total of $O(n)$ simulation steps while the MCEM and MCNR requires $O(n)$ simulation steps in each of its iteration.

The number of iterations for the Newton-Raphson procedure to reach precision $cn^{-1/2}$ can be calculated by the following argument. The Newton-Raphson procedure has quadratic convergence rate, which means that the precision ξ_{k+1} at the $(k+1)$st iteration is $\xi_{k+1} = c\xi_k^2 = \cdots = c^k\xi_1^{2k}$ (see for example Thisted (1988), page 164). Therefore the number of iterations to reach precision $cn^{-1/2}$ is $O(\log(n))$. Even though the MCEM algorithm has a linear convergence rate (as does the EM), the number of iterations required is also $O(\log(n))$.

To summarize, we note that if in an implementation of SA algorithm, it takes $O(n)$ simulation steps, then $O(n\log(n))$ number of simulation steps are expected for the MCEM or the MCNR algorithm to reach the same results.

In the above analysis we have assumed that in the M-step of the MCEM algorithm a closed form is possible (like for the genetic linkage problem in Section 3). If the M-step itself requires some iterative routines (such as logistic in error-in-variable model in Section 5), the actual number of simulations for the MCEM algorithm can be much larger.

It is clear that in problems where the sample size is large and each simulation step requires many simulations from the conditional distribution, there is a great savings in computation time by adopting the SA algorithm.

4.2 Example: linear regression with the Motorette data

We present this example to compare the SA algorithm to the MCEM algorithm on the same data set. The Motorette data were used by Tanner and Wei (1990) where the original idea of MCEM was introduced.

The data set is from Schmee and Hahn (1979). At each of four temperatures, ten motorettes were tested and the time to failure in hours was recorded. The data set can be modeled by the right-censored linear regression model:

$$t_i = \beta_0 + \beta_1 v_i + \sigma\epsilon_i$$

where $\epsilon_i \sim N(0,1)$; $v_i = 1000/(\text{temperature} + 273.2)$ and $t_i = \log_{10}(i\text{th failure time})$.

Reorder the data so that the first q observations are uncensored and the remaining $n-q$ are censored. The log-likelihood of the complete data is given by:

$$-n\log\sigma - \sum_{i=1}^{q}(t_i - \beta_0 - \beta_1 v_i)^2/2\sigma^2 - \sum_{i=q+1}^{n}(T_i - \beta_0 - \beta_1 v_i)^2/2\sigma^2,$$

where T_j is the unknown failure time for case j and T_j is known to be greater than c_j. The conditional distribution $p_j(t_j|\boldsymbol{\beta}, \sigma^2, c_i)$ is the normal distribution $N(\beta_0 + \beta_1 v_i, \sigma^2)$ conditional on T_j being greater than c_j. Having sampled T_j^k, $j = q, \ldots, n$, from the predictive conditional distribution, the kth iteration is calculated by (12) with $\boldsymbol{X}^k = (t_1, \cdots, t_q, T_{q+1}^k, \cdots, T_n^k)$.

To apply the MCEM algorithm, the function $Q(\boldsymbol{\beta}, \sigma^2; \boldsymbol{\beta}_k, \sigma_k^2)$ in the kth iteration is approximated by

$$Q_m(\boldsymbol{\beta}, \sigma^2; \boldsymbol{\beta}_k, \sigma_k^2) = -n\log\sigma - \sum_{i=1}^{l}(t_i - \beta_0 - \beta_1 v_i)^2/2\sigma^2 - \sum_{i=l+1}^{n}\frac{1}{m}\sum_{j=1}^{m}(T_i^j - \beta_0 - \beta_1 v_i)^2/2\sigma^2,$$

where $T_i^1, \ldots, T_i^m$ is an i.i.d. sample from the conditional distribution $p(t_i|\boldsymbol{\beta}_k, \sigma_k^2, c_i)$. The $\boldsymbol{\beta}_{k+1} = (\beta_{0,k+1}, \beta_{1,k+1}), \sigma_{k+1}^2$ can be obtained by solving the equations $(\partial/\partial\boldsymbol{\beta})Q_m(\boldsymbol{\beta}, \sigma^2; \boldsymbol{\beta}_k, \sigma_k^2) = 0$ and $(\partial/\partial\sigma^2)Q_m(\boldsymbol{\beta}, \sigma^2; \boldsymbol{\beta}_k, \sigma_k^2) = 0$, which gives

$$\sum_{i=1}^{q}(t_i - \beta_0 - \beta_1 v_i) + \sum_{i=q+1}^{n}(\bar{T}_i - \beta_0 - \beta_1 v_i) = 0,$$
$$\sum_{i=1}^{q}v_i(t_i - \beta_0 - \beta_1 v_i) + \sum_{i=q+1}^{n}v_i(\bar{T}_i - \beta_0 - \beta_1 v_i) = 0,$$

and

$$\sigma^2_{k+1} = \frac{1}{n}\left[\sum_{i=1}^{q}(t_i - \beta_{0,k+1} - \beta_{1,k+1}v_i)^2 + \sum_{i=q+1}^{n}(\bar{T}_i - \beta_{0,k+1} - \beta_{1,k+1}v_i)^2 + \sum_{i=q+1}^{n}\frac{1}{m}\sum_{j=1}^{m}(T_i^j - \bar{T}_i)^2\right],$$

where $\bar{T}_i$ is the average of $T_i^1, \ldots, T_i^m$ for each i. Note here the M-step in the MCEM algorithm has a closed form representation.

We also note that the EM algorithm also applies to this data set since $\mathrm{E}[T_i|T_i > c_i, \beta_k, \sigma_k^2]$ has an explicit expression. See Tanner (1993) for details.

The SA algorithm was initiated with $m = 1$, $\beta_0 = -4.931$, $\beta_1 = 3.747$, and $\sigma = \sqrt{0.0247}$. The B_k in (12) was taken as in (13) of Remark 2. Figure 1 shows the history of an implementation of the SA algorithm. The dotted lines represent the history of the MCEM algorithm, which is from Tanner (1993, page 55).

$\langle$ Figure 1 comes here $\rangle$

The horizontal lines in Figure 1 represent the true MLE's. This shows that the SA algorithm applied to the Motorette data converges to the MLE. The dotted lines are constructed from the values given by Wei and Tanner (1990) which were calculated by the MCEM algorithm under the non-informative prior $p(\sigma^2, \boldsymbol{\beta}) \propto \sigma^{-2}$. (The direct MLE can be considered in a Bayesian setting as using a flat prior). To apply the MCEM algorithm, one has to determine how large m should be to achieve the precision required. As indicated in section 4.1, this is related to the Hessian matrix of $Q(\boldsymbol{\psi}, \cdot)$ at the MLE. The advantage of the SA algorithm is that the number of iterations does not have to be determined in advance. One can sequentially plot the estimates and stop the algorithm when little change is seen. Since the SA algorithm (12) with B_k as the Louis's estimate also gives an estimate of the information matrix $\mathcal{I}$, one can also use that combined with Theorem 1 to determine the number of iterations to achieve a specific precision.

5 Application to Error-in-Variable Logistic Regression and other Models.

5.1 Error-in-Variable Logistic Regression.

The problem of covariate measurement error in a logistic regression model has been investigated by many authors, notably Carroll *et al.* (1984), Stefanski and Carroll (1987) and Schafer (1987). The reader may also refer to the recent monograph by Carroll, Stefanski and Rupert (1995).

We consider the structural error-in-variable model. In this model, the true, unobservable covariate $\boldsymbol{Z}$ follows a density function $f_{\boldsymbol{Z}}$. Assume that $(Y_j, \boldsymbol{X}_j)$, $j = 1, \ldots, n$ are an i.i.d. sample, where Y_i is a binary response with

$$\Pr\{Y_i = 1 | \boldsymbol{Z}_i = \boldsymbol{z}\} = g(\boldsymbol{\beta}'\boldsymbol{z}) = (1 + \exp(\boldsymbol{z}'\boldsymbol{\beta}))^{-1} \tag{15}$$

and $\boldsymbol{X}_i$ and $\boldsymbol{Z}_i$ are related through $\boldsymbol{X}_i = \boldsymbol{Z}_i + \boldsymbol{\epsilon}_i$, where $\boldsymbol{\epsilon}_i$, $i = 1, \cdots, n$ are i.i.d error variables with distribution function $f_{\boldsymbol{\epsilon}}$. For simplicity, we shall assume here that both distributions $f_{\boldsymbol{Z}}$ and $f_{\boldsymbol{\epsilon}}$ are known. In general, they may be assumed known upto an unknown scale parameter. If we try to use the EM algorithm to find the MLE of β, the E-step requires evaluation of the following multi-dimensional integral

$$Q(\boldsymbol{\beta}, \boldsymbol{\beta}_k) = \sum_{j=1}^{n} \int \cdots \int \left[(1 - y_j)\boldsymbol{z}_j'\boldsymbol{\beta} - \log\left(1 + \exp(\boldsymbol{z}_j'\boldsymbol{\beta})\right)\right] f(\boldsymbol{z}_j | y_j, \boldsymbol{x}_j, \boldsymbol{\beta}_k) d\boldsymbol{z}_j, \tag{16}$$

where $f(\boldsymbol{z}|y, \boldsymbol{x}, \boldsymbol{\beta}_k)$ denotes the conditional density of $\boldsymbol{Z}_j$ given $Y_j = y$, $\boldsymbol{X}_j = \boldsymbol{x}$ and $\boldsymbol{\beta} = \boldsymbol{\beta}_k$. Note that the M-step requires the above integral to be calculated for many points of $\boldsymbol{\beta}$ if a numerical procedure is used to to maximize $Q(\boldsymbol{\beta}, \boldsymbol{\beta}_k)$ with respect to $\boldsymbol{\beta}$.

The problem was circumvented in Carroll *et al.* (1984) by using a probit regression with normal distributions for $\boldsymbol{Z}$ and $\boldsymbol{\epsilon}$. Tackling the problem directly, Schafer (1987) suggested an approximation for the integration in (16) for the EM algorithm (approximate EM): assume that $\mathrm{E}\log(g(\boldsymbol{Z})) = \log(g(\mathrm{E}\boldsymbol{Z}))$, where g is the regression function in (15) and the expectation is with respect to the conditional distribution $f(\boldsymbol{z}|y, \boldsymbol{x}, \boldsymbol{\beta}_k)$. The accuracy of such an approximation, of course, depends on the closeness of $(d/dt)\log(g(t))$ to linearity or the magnitude of the error $\boldsymbol{\epsilon}$. In the present case, $\log(g())$ is nonlinear. Still when $\boldsymbol{\epsilon}$ is small, Schaffer's method works very well (See Table 5.1). Schaffer's method could produce an estimate far from the true MLE when $\boldsymbol{\epsilon}$ is large. In general, it is not clear when it will work well and when it will not.

Another way to implement the EM algorithm is to use numerical integration (usually the method of Gaussian quadrature) to calculate the value in (16). Schafer (1987) has shown that such an approach often leads to non-convergence of the EM algorithm.

To apply the SA algorithm to find the MLE in the error-in-variable logistic regression model above, we only need to generate random variables from the conditional distribution $f(\boldsymbol{z}|y, \boldsymbol{x}, \boldsymbol{\beta}_k)$ in (16), which is proportional to

$$h(\boldsymbol{z}) = f_{\boldsymbol{Z}}(\boldsymbol{z}) f_{\boldsymbol{\epsilon}}(\boldsymbol{x} - \boldsymbol{z}) \exp((1-y)\boldsymbol{z}'\boldsymbol{\beta})/(1 + \exp(\boldsymbol{z}'\boldsymbol{\beta})). \tag{17}$$

A simple way of generating random variables from such a density is to use the acceptance/rejection method, or, first generate $\boldsymbol{Z}$ according to $f_{\boldsymbol{Z}}$ and then accept this $\boldsymbol{Z}$ as generated from the conditional distribution with probability $h(\boldsymbol{z})/(C \cdot f(\boldsymbol{z}))$, where C can be any constant no less than $\sup_{\boldsymbol{Z}}(h(\boldsymbol{z})/f(\boldsymbol{z}))$. An easy choice of C would be $\sup_{\boldsymbol{Z}}(f_{\boldsymbol{\epsilon}}(\boldsymbol{z}))$.

If the first two terms of $h(\boldsymbol{z})$ in (17) can be combined into a single density, then more efficient ways of generating random variables from the conditional density can be devised. This is the case when both $f_{\boldsymbol{Z}}$ and $f_{\boldsymbol{\epsilon}}$ are normal densities which we assume below.

Another way of generating random variables from the conditional distribution proportional to $h(\boldsymbol{z})$ is by the method of the Metroplis algorithm (Metroplis *et al.* 1953). In particular, the method of independence chain (Hastings, 1970) can be used with the Markov transition density $q(z', z) = q(z) = c f_{\boldsymbol{Z}}(\boldsymbol{z}) f_{\boldsymbol{\epsilon}}(\boldsymbol{x} - \boldsymbol{z})$. With this choice, the resulting rejection probability in each iteration of the Metroplis algorithm only involves the logistic function, the last factor of $h(\boldsymbol{z})$ in (17).

Table 5.1 shows the simulation results of the mean square error of the MLE's obtained by the three different methods mentioned above with the model in (15). 2000 simulated samples are used and the starting value β_1 is set to be .5 in the case of SA algorithm. We have run the simulation with $m = 1, 10, 30, 50$ and both with the rejection method and the Metroplis algorithm for the SA algorithm. The results are simular to each other as long as the SA algorithms are used. The reported results in the SA column are from the case with $m = 30$ and the Metroplis algorithm. The last column is the theoretical variance in the same scale calculated based on the Louis estimate in (13) with $\boldsymbol{\psi}_i$ replaced by the true parameter and $k = 50,000$. $\boldsymbol{Z}$'s are generated from $N(0,1)$ and $\beta = 1$. The distribution of ϵ's is assumed to be N(0, σ^2). The table shows the results for a sample of size $n = 300$ with $\sigma^2 = 0.1, 0.4$ and 0.8. The numbers in the first two columns are directly copied from Schafer (1987) based on 100 simulations.

Table 5.1. Average of $(\hat{\beta} - \beta)^2 \times 1000$

σ^2	Numerical Integration	Schafer's Approximation	SA Algorithm	Theoretical Variance
0.1	33	26	27.0	26.5
0.4	116	40	39.4	37.4
0.8	781	107	55.1	50.1

The SA algorithm gives very similar mean squared errors to the approximation for the cases $\sigma^2 = .1$ and $\sigma^2 = .4$. Bothe methods work very well in these two cases. The SA algorithm gives a better result than Schafer's approximation for the case $\sigma^2 = .8$. The closeness of the mean square errors to the theoretical variance indicates that the SA algorithm gives values which are essentially the MLEs. Interestingly, numerical integration, which theoretically is consistent, performs much worse than the approximate EM and the SA algorithm. Note that in theory, the SA algorithm involves no approximation to the integration.

Theoretically speaking, when the Metropolis algorithm is used to simulate from the conditional distribution, one has to iterate the Markov chain long enough before the random deviates can be considered as sampled from the conditional distribution. For the simulation results reported above, the following have been done. For each step of the the SA algorithm, we first iterate the specific Markov chain 50 times and then take the next m iterates of the Markov chain as the simulated values $\boldsymbol{X}_1^k, \cdots, \boldsymbol{X}_m^k$ and use them in the updating formulae (12). This can be justified by the Ergodic property of Markov chains. We have also investigated whether iterating the Markov chain 50 times in starting a new step of SA algorithm is enough. The answers we got from extensive simulation studies (not reported here) are quite encouraging: it actually does not matter whether one iterates the Markov chain 1, 2, 5, 10, 30, 50 or 200 times as long as one uses the last Markov deviate in the previous SA step as the staring value in the present Markov chain. An intuitive explanation for this phenomenon can be given as follows: when k is large, γ_k is small, which implies that the changes in $\boldsymbol{\psi}_k$ are small. If $\boldsymbol{\psi}_k$ does not change, then we are iterating the same Markov chain, which converges to its invariant distribution, the conditional distribution. More study on this aspect is certainly necessary.

5.2 Nonlinear Random Effects model.

Another class of problems which is similar to that considered above is nonlinear random effects model. Such models were first considered by Stiratelli, Laird and Ware (1984). See also Zeger, Liang and Albert (1988) and Lindstrom and Bates (1990). It is a long standing problem to solve for MLE's in this model. For the purpose of illustration, we consider a simple logit-normal model in the following. Suppose that $Y_{i,j}$ ($i = 1, \ldots, q$; $j = 1, \ldots, n_i$) follows a binomial distribution with probability $p_{i,j}$, where $p_{i,j}$ is related to the observed covariate $x_{i,j}$ and unobserved random effect u_i through

$$p_{i,j} = \left(1 + \exp(\beta x_{i,j} + u_i)\right)^{-1}. \tag{18}$$

Assume that u_i, $i = 1, \cdots, q$, is an i.i.d. sample from $N(0, \sigma^2)$. For the E-step of the EM algorithm, one has to calculate the function

$$Q(\beta, \sigma^2; \beta_k, \sigma_k^2) \;=\; \sum_{i=1}^{q} \int \sum_{j=1}^{n_i} \left[(1 - y_{i,j})(\beta x_{i,j} + u_i) - \log\left\{1 + \exp(\beta x_{i,j} + u_i)\right\}\right] \tag{19}$$

$$\frac{1}{\sigma}\phi\left(\frac{u_i}{\sigma}\right) f(u_i|y_{i,1},\cdots,y_{i,n_i},\beta_k,\sigma_k^2)du_i,$$

where $\phi(u)$ is the standard normal density and $f(u|y_{i,1},\cdots,y_{i,n_i},\beta_k,\sigma_k^2)$ is the conditional density of U_i given the $y_{i,j}$'s and the current estimates of β and σ^2. This conditional density is proportional to

$$\frac{1}{\sigma}\phi\left(\frac{u}{\sigma}\right)\prod_{i=1}^{n_i}\frac{\exp\{(1-y_{i,j})(\beta x_{i,j}+u)\}}{1+\exp\{\beta x_{i,j}+u\}}. \tag{20}$$

As we may see from (19) and (20), the calculation and simulation in this case are very similar to the error-in-variable case and the SA algorithm can be applied to find the MLE in this setting.

Although our example here is for logistic regression model, in general, the SA algorithm applies to problems of estimating the MLE's in a generalized linear model setting with measurement errors or random effects.

6 Discussion

We have presented a new method for computing an MLE based on incomplete data. It is most advantageous when the E or M step of the EM algorithm can not be performed in closed form. In contrast to the MCEM and the MCNR algorithms which are also appropriate for the incomplete data problems, the SA algorithm is more efficient. The main requirement to implement the algorithm is the ability to generate deviates from the conditional distribution of the complete data given the incomplete data and the current estimate of the parameter(s). This requirement becomes less stringent if one considers the developments on the Metroplis-Hastings algorithms (Metroplis *et al.* 1953; Hastings, 1970).

Another variation of the algorithm in Section 3 for the multi-dimensional parameter case is to use a diagonal matrix for B_k in (12). Of course, a reasonable choice for the ith element of B_k will be some variance estimate of the ith element of $\boldsymbol{\psi}_k$. Note that if we use this modification, there will be no matrix inversion in each iteration. Moreover, if we use a different random deviate $\boldsymbol{X}^k$ for each component of $\boldsymbol{\psi}_k$ and keep updating the components, we would have something very similar to the Gibbs sampler (Geman and Geman, 1984, Gelfand & Smith, 1990). These approaches are certainly interesting and worth investigating.

We have based our SA algorithm for incomplete data on the foundation of the likelihood theory. The SA algorithm can also be used to find the maximum posterior estimates (posterior mode) in a Bayesian setting and Huber's M-estimator (Huber, 1973). Note that in the latter case, the extension naturally follows the missing information principle of Lai and Ying (1995) when the data are censored and/or truncated.

Finally, we emphasize that the SA algorithm is best used to tackle problems with intractable E-step or M-step by the EM algorithm. We do not recommend its use on problems where the usual EM algorithm can be directly applied. The example of Motorette data in section 4 is for illustrative purposes. The proposed SA algorithm shines in the hard problems of sections 5.

Appendix: Proof of Theorem 1

The proof follows the proof of Theorem 2 of Wei (1987). We first note that with Assumption 1-3, conditions (4.9), (4.10) are satisfied. Theorem 2 of Wei (1987) requires the errors to be i.i.d. which is not the case in our proof. A close examination of Wei's proof reveals that the part of the proof which requires this condition is so that the Central Limit Theorem (1.7) holds. Wei's (1.7) is based on the work of Dvoretzky (1977), which gives a martingale version of the Central Limit Theorem (CLT). Our conditions guarantee that the CLT still holds in this case. To prove the CLT, we note that (4.24) is replaced by our assumption $(\log\log(n))^{1/2}(B_n - \mathcal{I}) \to 0$ almost surely and $\delta_n = 0$. The rest of the proof is exactly the same as that in Wei (1987). We omit the details.

REFERENCES

Blum, J. R. (1954). Multidimensional stochastic approximation methods. *Annals of Mathematical Statistics,* 25, 382-386.

Carroll, R. J., Ruppert, D. and Stefanski, L. A. (1995). *Measurement Error in Nonlinear Models*, New York: Chapman and Hall.

Carroll, R. J., Spiegelman, C. H., Lan, K. K. G., Bailey, K. T. and Abbott, R. D. (1984). On error-in-variables for binary regression models. *Biometrika,* 71, 19-25.

Chung, K. L. (1954). On a stochastic approximation method. *Annals of Mathematical Statistics,* 25, 463-483.

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977). "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society,* **Ser. B,** 39, 1-38.

Dvoretzky, A. (1977). Asymptotic normality for sums of dependent random vectors. in *Multivariate Analysis* (P. R. Krishnaiah, ed) 4, 23-34. North-Holland, Amsterdam.

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

Gelfand, A. E. and Smith, A. F. M. (1990). Sampling based approaches to calculating marginal densities. *Journal of the American Statistical Association*, 85, 398-409.

Geman, S. and Geman. D. (1984). Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6, 721-741.

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

Green, P. J. (1990). On use of EM algorithm for penalized likelihood estimation. *Journal of the Royal Statistical Society*, **Ser. B,** 52, 443-52.

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications," *Biometrika,* 57, 97-109.

Huber, P. J. (1973). "Robust regression: Asymptotics, conjecture and Monte Carlo. *The Annals of Statistics*, 1, 799-821.

Lai, T. L. (1985). Stochastic approximation and sequential search for optimum. *Proceedings of the Berkeley conference in honor of J. Neyman and J. Kiefer, V. II,* Wadsworth: Hayward, California.

Lai, T. L. and Ying, Z. (1995). A missing information principle and M-estimators in regression analysis with censored and truncated data. *The Annals of Statistics,* 22, 1222-1255.

Lange, K. (1995a). A gradient algorithm locally equivalent to the EM algorithm. *Journal of the Royal Statistical Society,* **Ser. B**, 57, 425-437.

Lange, K. (1995b). A Quasi-Newton acceleration of the EM Algorithm. *Statistica Sinica*, 5, 1-18.

Lindstrom, M. J. and Bates, D. M. (1990). Nonlinear mixed effects models for repeated measures data. *Biometrics,* 46, 673-687.

Little, R. J. A. & Rubin, D.B. (1987). *Statistical Analysis with Missing data,* New York: John 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.

Louis, T. A. (1982). Finding observed information using the EM algorithm. *Journal of the Royal Statistical Society* , **Ser. B**,44, 98-130.

Meilijson, I. (1989). A faster improvement to the EM algorithm on its own terms. *Journal of the Royal Statistical Society,* **Ser. B,** 51, 127-138.

Meng, X. L. and Rubin, D. B. (1993). Maximum likelihood estimation via the ECM algorithm: A general framework. *Biometrika,* 80, 267-278.

Metroplis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. and Teller, E. (1953). Equations of state calculations by fast computing machines. *Journal of Chemical Physics,* 21, 1087-1091.

Nevel'son, M. B. and Has'minski, R. Z. (1973a). *Stochastic Approximation and Recursive Estimation.* Translations of Mathematical Monographs, Vol. 47, American Mathematical Society, Providence, R.I.

Nevel'son, M. B. and Has'minski, R. Z. (1973b). An adaptive Robbins-Monro procedure. *Automation and Remote Control,* 34, 1594-1607.

Robbins, H. and Monro, S. (1951), A stochastic approximation method. *Annals of Mathematical Statistics,* 22, 400-407.

Sacks, J. (1958). Asymptotic distribution of stochastic approximation procedures. *Annals of Mathematical Statistics,* 29, 373-405.

Schafer, D. W. (1987). Covariate measurement error in generalized linear models. *Biometrika,* 74, 385-391.

Schmee, J., and Hahn, G. L. (1979). A simple Method for Regression Analysis With Censored Data. *Technometrics*, 21, 417-432.

Stefanski, L. A. and Carroll, R. J. (1987). Conditional scores and optimal scores for generalized linear measurement error models. *Biometrika,* 74, 703-716.

Stiratelli, R., Laird, N. and Ware, J. H. (1984). Random-effects models for serial observation. *Biometrics,* 40, 961-971.

Tanner, M. A. (1993). *Tools for Statistical Inference*(2nd e.d.), New York: Springer-Verlag.

Thisted R. A. (1988). *Elements of Statistical Computing- Numerical computation*, New York: *Chapman and Hall*.

Titterington, D. M. (1984). Recursive parameter estimation using incomplete data. *Journal of the Royal Statistical Society* , **Ser. B,** 46, 257-267.

Wei, C. Z. (1987). Multivariate Adaptive stochastic approximation. *The Annals of Statistics,* 15, 1115-1130.

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

Zeger, S. L., Liang, K.-Y. and Albert, P. (1988). Models for longitudinal data: A generalized estimating equation approach. *Biometrics*, 44, 1049-1060.

Figure 1. Histories of applying the SA and the MCEM algorithms to the Motorette data