On simulated EM algorithms

Soren Feodor Nielsen

Department of Theoretical Statistics, Institute of Mathematical Sciences, University of Copenhagen, Universitetsparken 5, 2100 Copenhagen 0, Denmark

Received 1 August 1997; received in revised form 1 May 1999

Abstract

The EM algorithm is a popular and useful algorithm for finding the maximum likelihood estimator in incomplete data problems. Each iteration of the algorithm consists of two simple steps: an E-step, in which a conditional expectation is calculated, and an M-step, where the expectation is maximized. In some problems, however, the EM algorithm cannot be applied since the conditional expectation required in the E-step cannot be calculated. Instead the expectation may be estimated by simulation. We call this a simulated EM algorithm. The simulations can, at least in principle, be done in two ways. Either new independent random variables are drawn in each iteration, or the same uniforms are re-used in each iteration. In this paper the properties of these two versions of the simulated EM algorithm are discussed and compared. © 2000 Elsevier Science S.A. All rights reserved.

Keywords: Simulation; EM algorithm; Incomplete data

1. Simulated EM algorithms

The EM algorithm (cf. Dempster et al., 1977) has two steps, an E-step and an M-step. The E-step is the calculation of the conditional expectation of the complete data log-likelihood given the observed data. The M-step is a maximization of this expression. These two steps are then iterated. Each iteration of the algorithm increases the observed data log-likelihood, and though the general
convergence theory is rather vague, the algorithm often works well in practice. Ruud (1991) gives an overview of the general theory and some applications to econometrics. For more on the EM algorithm, see McLachlan and Krishnan (1997).

However in some cases the E-step of the algorithm is not practically feasible, because the conditional expectation cannot be calculated. This happens for instance when the expectation is a large sum or when the expectation corresponds to a high-dimensional integral without a closed-form expression. In this case, the expectation could be replaced by an estimate obtained by simulation. We shall call the resulting algorithm a *simulated EM algorithm*, a name taken from a paper by Ruud (1991). More or less different ways of constructing this estimate have been suggested by Celeux and Diebolt (1985), Wei and Tanner (1990), and McFadden and Ruud (1994). The difference between these ideas are discussed below after a more formal definition of the simulated EM algorithm.

Let $X_1, X_2, \ldots, X_n$ be iid random variables with density $f_h$. Instead of observing $X_i$ suppose we observe $Y_i = Y(X_i)$. Then the E-step of the EM-algorithm is the calculation of

$$\theta \to Q(\theta|\theta') = \frac{1}{n} \sum_{i=1}^{n} E_{\theta'}(\log f_{\theta}(X_i)|Y_i)$$

for a fixed value of $\theta'$.

In the M-step this function is maximized over $\theta$, and the maximizer is then used as a new value of $\theta'$ in the subsequent iteration. In this way we get a sequence of $\theta$-values, $\theta_1, \theta_2, \ldots$ say, which converge under weak assumptions to the local maximizer (cf. Wu, 1983).

One iteration of the EM algorithm thus corresponds to calculating the conditional expectation (1) and maximizing it as a function of $\theta$. We denote the EM update, i.e. the $\theta$-value given by one iteration of the EM algorithm starting in $\theta'$, by $M(\theta')$. The maximum likelihood estimator, $\hat{\theta}_n$, is a fixed point of $M$, i.e. a solution to $M(\theta) = \theta$. The EM algorithm finds fixed points of $M$ by the method of successive substitution; from a starting value, $\theta_1$, of $\theta$, $\theta_2 = M(\theta_1)$ is calculated and used to calculate $\theta_3 = M(\theta_2)$, etc.

In the simulated EM algorithm the expectation (1) is replaced by an estimate in the following way: Let $X_{ij}$ for $j = 1, \ldots, m$ be a random variable drawn from the conditional distribution of $X_i$ given $Y_i = y_i$ with parameter $\theta'$. Then

$$\tilde{Q}(\theta|\theta') = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{m} \sum_{j=1}^{m} \log f_{\theta}(\tilde{X}_{ij})$$

is an unbiased estimate of $Q(\theta|\theta')$. In practice, $\tilde{X}_{ij}$ is simulated using one or more uniform (pseudo-) random numbers. Thus $\tilde{X}_{ij} = g(U_{ij}|Y_i, \theta')$ for some function $g(\cdot|y, \theta)$, where the $U_{ij}$'s are independent (vectors of) uniform random variables, independent of the $Y_i$'s. Whether the $U_{ij}$'s are one-dimensional or vectors of iid
uniform random variables, depends on the dimension of the $X_i$’s as well as on
the method of generating the $X_{ij}$’s. We shall not go further into the actual
generation of the $X_{ij}$’s, since all that matters for the results of this paper is the
fact that the simulations are based on a sequence of (pseudo-) random numbers;
they need not even be uniform, but generally they are.

We get two fundamentally different simulated EM algorithms according to
whether we draw new independent random variables in each iteration or we
re-use the uniforms, $U_{ij}$, in each iteration.

Drawing new uniforms in each iteration, the sequence of $h$-values obtained
from the algorithm, $(\tilde{\theta}_n(k))_{k=0}^\infty$, is a Markov chain, which is typically ergodic. As
$k \to \infty$, the distribution of $\tilde{\theta}_n(k)$ approaches the stationary initial distribution of
the Markov chain. The estimator obtained, when the algorithm has converged,
is thus a random variable drawn from the stationary initial distribution of the
chain. This version has been discussed in detail by Nielsen (2000). It is essentially
the ‘stochastic EM algorithm’ suggested by Celeux and Diebolt (1985) (see
Diebolt and Ip, 1996 for a review), which also uses new uniforms in each step of
the algorithm but only allow $m = 1$.

Reusing the uniforms, we estimate the function $\theta \to M(\theta)$ once and for all
and search for fixed points of the estimated function by the method of successive
substitutions. Since a fixed point, $\theta^*$, of $M$ has $0 = D_\theta Q(\theta|\theta^*)_{|\theta = \theta^*}$, this essentially
corresponds to finding a root of

$$G_n(\theta^*) = \frac{1}{n} \sum_{i=1}^n \frac{1}{m} \sum_{j=1}^m D_\theta \log f_\theta(g(U_{ij}|Y_i, \theta^*))_{|\theta = \theta^*}. $$

This is done by the method of successive substitutions: Starting with a given
value of $\theta^*$, the next value is found by finding the root of

$$D_\theta \bar{Q}(\theta|\theta^*) = \frac{1}{n} \sum_{i=1}^n \frac{1}{m} \sum_{j=1}^m D_\theta \log f_\theta(g(U_{ij}|Y_i, \theta^*))$$

which is then used to find the next value, etc. In practice of course there may well
be more than one root of $D_\theta \bar{Q}(\theta|\theta^*)$; the root we need to pick is the one
maximizing $\bar{Q}(\theta|\theta^*)$. Notice that $G_n(\theta^*)$ is an unbiased estimate of $D_\theta Q(\theta|\theta^*)_{|\theta = \theta^*}$ if
differentiation and expectation are interchangeable. This version is discussed by
McFadden and Ruud (1994) as a special case of the method of simulated
moments.

So we have two versions of the simulated EM algorithm; in one we keep
drawing new random numbers (leading to a Markov chain of $\theta$-values), and one
in which we re-use the uniforms.

In order to distinguish between these two versions of the algorithm we will use
the acronym StEM for the version, where new independent random variables
are drawn in each iteration, and call the second version, where the uniforms are
re-used, the SimEM algorithm. One should note that in the literature the acronym SEM is used by McFadden and Ruud (1994) for the SimEM algorithm, but it is also the commonly used acronym for the stochastic EM algorithm suggested by Celeux and Diebolt (1985). For this reason we avoid the acronym SEM. Also the supplementary EM algorithm (cf. Meng and Rubin, 1991), which is not a simulated EM algorithm, is denoted SEM in the literature.

It should be clear that at least for moderate values of \( m \) the two versions differ significantly. The purpose of this paper is to compare these two simulated EM algorithms.

As noted above StEM is essentially just the stochastic EM algorithm suggested by Celeux and Diebolt (1985). The only difference is that we here allow \( m > 1 \). Diebolt and Celeux (1993) discuss estimation of mixing proportions using the stochastic EM algorithm and give some asymptotic results for this case. Nielsen (2000) gives asymptotic results under general smoothness assumptions.

It is not clearly stated by Ruud (1991) which version of the simulated EM algorithm he considers. However, the suggested asymptotic results and the paper by McFadden and Ruud (1994) indicates that it is the SimEM version rather than the StEM version, he has in mind. However, his definition would allow StEM as well as SimEM as a simulated EM algorithm, and we therefore use the term simulated EM algorithm to denote both StEM and SimEM.

Wei and Tanner (1990) suggest to calculate the conditional expectation needed in the E-step of the EM algorithm by Monte Carlo integration, leading to the so-called MCEM algorithm. With our notation this corresponds to \( m = \infty \). In this case it does not matter whether we re-use uniforms or draw new uniforms in each iteration. In practice, of course, \( m < \infty \) and the results discussed in this paper thus also applies to the MCEM algorithm. Ignoring that \( m < \infty \) will lead to underestimation of the asymptotic variance of the estimator, since we then ignore the noise added by the simulations.

Other ways of using simulation in estimation exist. One method is the method of Monte Carlo likelihood, where we attempt to estimate the observed data log-likelihood function using simulation and then maximize this estimated function (see Geyer, 1996 for a recent review). A closely related method is the method of simulated scores (MSS) suggested by Hajivassiliou and McFadden (1997). Another possibility is to specify a Bayesian model and use Gibbs sampling or other MCMC methods to estimate the parameters. These methods are discussed further in Section 6.

2. Asymptotic results

We begin this section by introducing some notation. All score functions defined below are assumed to have expectation zero and finite variance.
Let $f_\theta$ be the density of $X$. Let $s_\theta(\theta)$ be the corresponding score function and put $V(\theta) = E_\theta(s_\theta(\theta)s_\theta(\theta)^t)$; the superscript $t$ denotes transpose. Let $\theta_0$ denote the true unknown value of $\theta \in \Theta \subseteq \mathbb{R}^d$.

The score function corresponding to the conditional distribution of $X$ given $Y = y$ with density $x \rightarrow g_\theta(x|y)$ is denoted $s_{\theta|y}(\theta)$. Let $I_\theta(\theta) = E_\theta(s_\theta(\theta)s_\theta(\theta)^t|Y = y)$.

Let $s_\theta(\theta)$ be the score function corresponding to the distribution of $Y$ and put $I(\theta) = E_\theta(s_Y(\theta)s_Y(\theta)^t)$.

Notice that $s_\theta(\theta) = s_\theta(\theta) - s_{\theta|y}(\theta)$ and that $I(\theta) = V(\theta) - E_\theta I_Y(\theta)$. We assume that $V(\theta_0)$ and $I(\theta_0)$ are positive definite. This means that the unknown parameter $\theta$ is locally identified from the complete data as well as from the incomplete data. Of course this is necessary in order to estimate the parameter.

Finally we put $F(\theta) = E_\theta I_Y(\theta)V(\theta)^{-1}$. This can be interpreted as the expected fraction of missing information (cf. Dempster et al., 1977). We will need the following facts about $F(\theta)$:

**Lemma 1.** The eigenvalues of $F(\theta_0)$ are real, non-negative, and smaller than 1. $F(\theta_0)$ has $d$ linearly independent eigenvectors corresponding to the $d$ eigenvalues. The rank of $F(\theta_0)$ equals the number of non-zero eigenvalues.

**Proof.** Since $V(\theta_0)$ is positive-definite, it has a positive-definite square-root, $V(\theta_0)^{1/2}$. Let $V(\theta_0)^{-1/2}$ be the inverse of this square root.

Since $F(\theta_0)$ is similar to the positive-semi-definite matrix $V(\theta_0)^{-1/2}F(\theta_0)V(\theta_0)^{1/2} = V(\theta_0)^{-1/2}E_{\theta_0}I_Y(\theta_0)V(\theta_0)^{-1/2}$, it has $d$ linearly independent eigenvectors and real, non-negative eigenvalues (cf. Lancaster, 1969). Furthermore, $F(\theta_0)$ has the same rank as this symmetric matrix, which has rank equal to the number of non-zero eigenvalues.

If $\lambda$ is an eigenvalue and $e$ the corresponding eigenvector, then $F(\theta_0)e = \lambda e$, which is equivalent to $E_{\theta_0}I_Y(\theta_0)\tilde{e} = \lambda V(\theta_0)\tilde{e}$ with $\tilde{e} = V(\theta_0)^{-1}e$. Since the difference $V(\theta_0) - E_{\theta_0}I_Y(\theta_0) = I(\theta_0)$ is positive definite, we must have $\tilde{e} V(\theta_0)\tilde{e} > \tilde{e} E_{\theta_0}I_Y(\theta_0)\tilde{e} = \lambda^2 V(\theta_0)\tilde{e} \geq 0$, which implies that $\lambda < 1$. \qed

Asymptotic results for the StEM algorithm were discussed by Nielsen (2000, Theorem 1), who proved the following result under general regularity conditions:

**Theorem 1** (Asymptotic results for the StEM algorithm). If the Markov chain is ergodic and tight, then $\sqrt{n}(\tilde{\theta}_n - \theta_0) \overset{D}{\rightarrow} N(0, \Sigma_m(\theta_0))$ where

$$
\Sigma_m(\theta_0) = I(\theta_0)^{-1} + \frac{1}{m} \sum_{k=0}^{\infty} F(\theta_0)^k V(\theta_0)^{-1} E_{\theta_0} I_Y(\theta_0) V(\theta_0)^{-1} F(\theta_0)^k
$$

$$
= I(\theta_0)^{-1} + \frac{1}{m} V(\theta_0)^{-1} E_{\theta_0} I_Y(\theta_0) V(\theta_0)^{-1}(I - F(\theta_0)^2)^{-1}
$$

(3) where $I$ is the identity matrix.
Remark. Ergodicity and tightness can be checked as in Nielsen (2000).

Recall that $\tilde{\theta}_n$ is a random variable drawn from the stationary initial distribution of the Markov chain constructed in the StEM algorithm. The theorem states that the stationary distribution of the Markov chain converges weakly to a normal distribution as the sample size increases.

We shall not give a detailed proof of Theorem 1 here; see Nielsen (2000) for a precise statement of the regularity conditions as well as a general proof of the result. We will however give a proof for the case of exponential family models, which includes many models of practical interest as noted by Ruud (1991). Since this proof is rather long, it is deferred to the appendix. Example 1 in the following section proves this theorem and Theorem 2 below in a special case with Gaussian distributions, where the results hold exactly and not only asymptotically.

In order to obtain asymptotic results for the SimEM version, we apply Corollary 3.2 and Theorem 3.3 in Pakes and Pollard (1989). McFadden and Ruud (1994) give similar results under stronger assumptions.

Using $E_{\theta_0}$ to denote expectation under the true distribution of the $X_{ij}$’s and the $\tilde{X}_{ij}$’s, we note that

$$G(\theta) = E_{\theta_0}(G_n(\theta)) = E_{\theta_0}(D_\theta \log f_\theta(\tilde{X}_{ij})) = E_{\theta_0}E_{\theta_0}(D_\theta \log f_\theta(\tilde{X}_{ij})|Y_i)$$

$$= E_{\theta_0}E_{\theta}(s_X(\theta)|Y_i) = E_{\theta_0}(s_Y(\theta) + E_{\theta}(s_{X|Y}(\theta)|Y))$$

$$= E_{\theta_0}(s_Y(\theta)).$$

In the fourth equality note that $E_{\theta_0}(D_\theta \log f_\theta(\tilde{X}_{ij})|Y_i) = E_{\theta}(s_X(\theta)|Y)$; this is how the $\tilde{X}_{ij}$ are simulated. $G(\theta)$ is differentiable under usual regularity conditions with

$$D_\theta G(\theta)_{\theta=\theta_0} = I(\theta_0).$$

Furthermore, by the central limit theorem

$$\sqrt{n}G_n(\theta_0) \convergesP N\left(0, I(\theta_0) + \frac{1}{m} E_{\theta_0} I_Y(\theta_0)\right).$$

By assumption $G(\theta_0) = 0$ and we will assume that there is a neighbourhood of $\theta_0$ such that $\theta_0$ is the only root of $G$ inside this neighbourhood.

From the law of large numbers, we know that $G_n(\theta) \convergesP G(\theta)$ for all $\theta$. This needs to be extended to uniform convergence to obtain consistency. Also, by the central limit theorem, $\sqrt{n}(G_n(\theta) - G(\theta))$ is asymptotically normal; we need this
to be close to \( \sqrt{n}G_n(\theta) \) when \( \theta \) is close to \( \theta_0 \). To be precise we need these two assumptions:

(A1) \( \sup_{\theta \in \Theta} ||G_n(\theta) - G(\theta)||/(1 + ||G_n(\theta)|| + ||G(\theta)||) \overset{p}{\to} 0 \), for a compact neighbourhood, \( C \), of \( \theta_0 \).

(A2) 
\[
\sup_{||\theta - \theta_0|| < \Delta} (\sqrt{n}G_n(\theta) - G(\theta) - G_n(\theta_0))/(1 + \sqrt{n}||G_n(\theta)|| + \sqrt{n}||G(\theta)||) \overset{p}{\to} 0,
\]
for any \( \delta_n \to 0 \).

Stronger assumptions are obtained if the numerators are ignored. A sufficient condition for both assumptions is given in this lemma (see Pakes and Pollard, 1989 for weaker conditions):

**Lemma 2.** Suppose that in a neighbourhood of \( \theta_0 \) for some \( z > 0 \)
\[
||\log f_\theta(g(U|Y, \theta)) - \log f_{\theta'}(g(U|Y, \theta'))|| \leq \psi(U, Y)||\theta - \theta'||^z
\]

(i) If \( E_{\theta_0}||\psi(U, Y)|| < \infty \) then (A1) holds.

(ii) If \( E_{\theta_0}||\psi(U, Y)||^2 < \infty \) then (A2) holds.

**Proof.** See Lemmas 2.13, 2.8, and 2.17 in Pakes and Pollard (1989).

We say that \( \tilde{\theta}_n \) is an asymptotic local minimum of \( ||G_n(\theta)|| \) if for some open set \( W \subseteq \Theta \) \( \inf_{\theta \in W} ||G_n(\theta)|| = \inf_{\theta \in W} ||G_n(\theta_0)|| + o_p(n^{-1/2}) \). From Corollary 3.2 and Theorem 3.3 in Pakes and Pollard (1989) we get the following result:

**Theorem 2** (Asymptotic results for the SimEM algorithm). Under assumption (A1), there is an asymptotic local minimum, \( \tilde{\theta}_n \), of \( ||G_n(\theta)|| \) such that \( \tilde{\theta}_n \overset{P}{\to} \theta_0 \).

Under assumption (A2), if \( \tilde{\theta}_n \) is consistent, then
\[
\sqrt{n}(\tilde{\theta}_n - \theta_0) \overset{D}{\to} N(0, I(\theta_0)^{-1} + (1/m)I(\theta_0)^{-1}E_{\theta_0}I_Y(\theta_0)I(\theta_0)^{-1}).
\]

### 3. A comparison

It is clear that both the SimEM and the StEM version of the simulated EM algorithm approximates the EM algorithm as \( m \) tends to infinity. Both versions lead to estimators with asymptotic variances tending to \( I(\theta_0)^{-1} \) as \( m \to \infty \). For finite values of \( m \), however, the asymptotic variances differ.

Recalling that \( F(\theta_0) = E_{\theta_0}I_Y(\theta_0)V(\theta_0)^{-1} \) and \( V(\theta_0) = I(\theta_0) + E_{\theta_0}I_Y(\theta_0) \) we find (with the usual ordering of positive definite matrices) that
\[
I(\theta_0)^{-1}E_{\theta_0}I_Y(\theta_0)I(\theta_0)^{-1} > V(\theta_0)^{-1}E_{\theta_0}I_Y(\theta_0)V(\theta_0)^{-1}(I - F(\theta_0)^2)^{-1}
\]
\[
\Leftrightarrow (I - F(\theta_0)^2)V(\theta_0)E_{\theta_0}I_Y(\theta_0)^{-1}V(\theta_0) > I(\theta_0)E_{\theta_0}I_Y(\theta_0)^{-1}I(\theta_0)
\]
\[ \iff V(\theta_0)E_{\theta_0} I_Y(\theta_0)^{-1}V(\theta_0) - E_{\theta_0} I_Y(\theta_0) > I(\theta_0)E_{\theta_0} I_Y(\theta_0)^{-1}I(\theta_0) \]

\[ \iff E_{\theta_0} I_Y(\theta_0) + I(\theta_0)E_{\theta_0} I_Y(\theta_0)^{-1}I(\theta_0) + 2I(\theta_0) - E_{\theta_0} I_Y(\theta_0) \]

\[ > I(\theta_0)E_{\theta_0} I_Y(\theta_0)^{-1}I(\theta_0) \]

\[ \iff 2I(\theta_0) > 0. \]

In these calculations we have implicitly assumed that \( E_{\theta_o} I_Y(\theta_0) \) is non-singular. If \( E_{\theta_o} I_Y(\theta_0) \) is singular, then some of the eigenvalues of \( F(\theta_0) \) are zero (cf. the proof of Lemma 1). Suppose that \( r \) of the \( d \) eigenvalues of \( F(\theta_0) \) are zero. Then we can write \( F(\theta_0) = z\beta^t \), where \( z \) and \( \beta \) are \( d \times r \)-matrices of full rank, \( \text{span} \beta^t = \text{span} F(\theta_0) \) and \( \beta^t z \) is non-singular. Let \( z_\perp \) be a \( d \times (d - r) \)-matrix of full rank such that \( z_\perp z = 0 \). Then

\[
\theta \rightarrow \begin{bmatrix} x_\perp \theta \\ \beta \theta \end{bmatrix}
\] (6)

is a bijection of \( \theta \) (cf. Johansen, 1995). Thus \((x_\perp \theta, \beta \theta)\) is a reparameterization of \( \theta \). The fraction of missing information for the parameters \( x_\perp \theta \) and \( \beta \theta \) are

\[
(\beta^t E_{\theta_o} I_Y(\theta_0)\beta)(\beta^t V(\theta_0)\beta)^{-1} = (\beta^t F(\theta_0)V(\theta_0)\beta)(\beta^t V(\theta_0)\beta)^{-1} = \beta^t z,
\]

\[
(x_\perp E_{\theta_o} I_Y(\theta_0)x_\perp)(x_\perp V(\theta_0)x_\perp)^{-1} = (x_\perp F(\theta_0)V(\theta_0)x_\perp)(x_\perp V(\theta_0)x_\perp)^{-1} = 0.
\]

The second line means that there is no missing information about \( x_\perp \theta \), and the asymptotic variance of \( x_\perp \theta \) is easily seen (cf. the proof of Theorem 6) to be 0 for both versions of the simulated EM algorithm. To handle the parameter \( \beta \theta \), note that the missing information about \( \beta \theta \), \( \beta^t E_{\theta_o} I_Y(\theta_0)\beta \), is non-singular, since \( \beta^t z \) is non-singular. Thus we can repeat the argument given above for a non-singular \( E_{\theta_o} I_Y(\theta_0) \) applied to the information matrices corresponding to \( \beta \theta \), leading to the conclusion that \( \beta \theta \) is strictly better estimated using StEM rather than SimEM.

Hence, we have shown the following result:

**Theorem 3.** The estimator derived from the StEM algorithm has smaller asymptotic variance than the estimator derived from the SimEM algorithm, when the same value of \( m \), i.e. the same number of simulated values per iteration, is used in both algorithms.

This result may seem counterintuitive as new random noise is added in each iteration of the StEM algorithm. The following example offers an explanation.

**Example 1.** Let \( X_1, X_2, \ldots, X_n \) be iid bivariate random variables, normally distributed with common expectation \( \theta \) and variances 1 and known correlation \( \rho \in [\rho] - 1; 1[. \) Suppose only the first coordinate, \( X_{1i} \), of each \( X_i = (X_{1i}, X_{2i})' \) is observed. The observed data MLE, \( \hat{\theta}_n \), is just the average of the \( X_{1i} \)'s.
The simulated EM algorithm (with $m = 1$) corresponds to simulating $X_{2i}$ from the conditional distribution of $X_{2i}$ given $X_{1i}$ and then averaging all the $X_{ji}$’s, the observed as well as the simulated. Thus, in the $(k + 1)$th iteration, we simulate $\tilde{X}_{2i} = (1 - \rho)\bar{\theta}_n(k) + \rho X_{1i} + \varepsilon_{k,i}$, where $\varepsilon_{k,i} \sim N(0, 1 - \rho^2)$, and in the M-step we put

$$\tilde{\theta}_n(k + 1) = \frac{1}{2n} \sum_{i=1}^{n} (X_{1i} + \tilde{X}_{2i}).$$

This leads to

$$\sqrt{n}(\tilde{\theta}_n(k + 1) - \hat{\theta}_n) = \frac{1 - \rho}{2} \sqrt{n}(\bar{\theta}_n(k) - \hat{\theta}_n) + \varepsilon_k,$$

where $\varepsilon_k = \sqrt{n}(1/2n)\sum_{i=1}^{n} \varepsilon_{k,i} \sim N(0, \sigma^2)$ with $\sigma^2 = (1 - \rho^2)/4$.

In the SimEM case, $\varepsilon_k = \varepsilon_1$ for all $k$ and as $k \to \infty$

$$\sqrt{n}(\tilde{\theta}_n(k + 1) - \hat{\theta}_n) \to \frac{2}{1 + \rho} \varepsilon_1 \sim N\left(0, \left(\frac{4}{(1 + \rho)^2}\right) \sigma^2\right).$$

In the StEM case the $\varepsilon_k$’s are iid, and (7) defines an Gaussian AR(1) process, so that

$$\sqrt{n}(\tilde{\theta}_n(k + 1) - \hat{\theta}_n) \overset{\mathcal{D}}{\sim} N\left(0, \left(1 - \left(\frac{1 - \rho}{2}\right)^2\right)^{-1} \sigma^2\right)$$

when $k \to \infty$.

The (asymptotic) distribution of $\sqrt{n}(\bar{\theta}_n - \theta_0) = \sqrt{n}(\bar{\theta}_n - \hat{\theta}_n) + \sqrt{n}(\hat{\theta}_n - \theta_0)$ in the SimEM and the StEM case is thus normal with expectation 0 and variance given by the variance of the MLE plus the variance specified above (in (8) and (9), respectively).

It is not difficult to show directly that the variance of the StEM estimator is smaller than the variance of the SimEM estimator, and we leave these calculations to the reader. Instead, looking at (7), we observe that

$$\text{Var}(\sqrt{n}(\tilde{\theta}_n(k + 1) - \hat{\theta}_n)) = \frac{(1 - \rho)^2}{4} \text{Var}(\sqrt{n}(\bar{\theta}_n(k) - \hat{\theta}_n)) + \sigma^2$$

$$+ 2 \text{Cov}(\varepsilon_k, \sqrt{n}(\bar{\theta}_n(k) - \hat{\theta}_n)) \frac{1 - \rho}{2}.$$

For the StEM version the covariance term is 0, whereas it is positive for the SimEM version. Thus, we see that the variance of the SimEM estimator is larger exactly because ‘the uniforms are re-used’.

Furthermore, the StEM estimator can easily be improved — in the sense of reducing the variance — by averaging over the last 5–20, say, iterations. Since the iterations constitute a Markov chain, averaging will decrease the variance (see Nielsen, 2000 for details). Obviously, the only way to improve the SimEM estimator is to increase $m$, i.e. to increase the simulation burden.
The Markov chain of the StEM algorithm is irreducible under weak assumptions, and the StEM algorithm does therefore not ‘get stuck’ in a wrong estimate. If it converges, i.e. if the Markov chain is ergodic, the StEM algorithm converges stochastically in the sense that the sequence of distributions of the $\theta$-values obtained from the iterations converge in total variation. There is a growing literature on convergence detection for MCMC methods. Many of these methods for detecting convergence (or rather lack of convergence) of MCMC calculations can be used to detect (lack of) convergence for any Markov chain. Hence, there exist methods which can be applied to ascertain convergence of the StEM algorithm; see Brooks and Roberts (1998) for a review of these methods.

The SimEM algorithm is a deterministic algorithm – an application of the method of successive substitutions – searching for a root of the (random) function $G(\theta)$. Hence, if it converges, it converges deterministically. There is no apparent reason to believe that $G_n(\theta)$ only has one root, and the SimEM algorithm stops when one is found. Thus multiple runs are needed to ensure that the correct root is found. There may even be no root at all as in the following example. Furthermore, the method of successive substitutions is typically slow (requires many iterations), as is well-known in the case of the EM algorithm. Hence, it is not clear which algorithm is faster in practice.

**Example 2.** Let $X_1, X_2, \ldots, X_n$ be iid normally distributed with expectation 0 and variance $\theta$. Observe $X_i$ if $X_i \geq 0$. The SimEM algorithm (with $m = 1$) corresponds to iterating

$$
\bar{\theta}_n(k + 1) = \frac{1}{n} \sum_{i=1}^{n} \left( X_i^2 \cdot 1_{\{X_i \geq 0\}} + \bar{\theta}_n(k) \cdot e_i^2 \cdot 1_{\{X_i < 0\}} \right)
$$

where $e_i$ are iid random variables from the standard normal distribution given that they are negative. We shall assume that at least one $X_i$ is observed so that $\theta$ can be estimated from the observed data. Of course, as $n \to \infty$ this happens with probability 1 by the Borel–Cantelli lemma.

If $\sum_{i=1}^{n} e_i^2 \cdot 1_{\{X_i < 0\}} \geq n$ (which happens with positive – albeit small – probability), then $\bar{\theta}_n(k) \to \infty$. If $\sum_{i=1}^{n} e_i^2 \cdot 1_{\{X_i < 0\}} < n$ then $\bar{\theta}_n(k)$ converges to $\sum_{i=1}^{n} X_i^2 \cdot 1_{\{X_i \geq 0\}} / (n - \sum_{i=1}^{n} e_i^2 \cdot 1_{\{X_i < 0\}})$ which may be arbitrarily far from the maximum likelihood estimator; the MLE is obtained when $\sum_{i=1}^{n} e_i^2 \cdot 1_{\{X_i < 0\}} = \sum_{i=1}^{n} 1_{\{X_i < 0\}}$.

Using a contraction principle (cf. Letac, 1986), it is not difficult to show that the Markov chain of the StEM algorithm for this problem is ergodic so that the StEM algorithm converges.

Assumptions (A1) and (A2) (in the SimEM case) and tightness (in the StEM case) can be shown to hold, but the details will not be given here. Consequently, the asymptotic results of Section 2 hold.
Using (adaptive) rejection sampling or Markov chain Monte Carlo techniques, we can generally simulate the \( \tilde{X}_{ij} \)'s. However, neither technique will give us simulations suitable for the SimEM version of the simulated EM algorithm, as neither method ‘re-uses uniforms’ even if we start each iteration with the same random seed. This is due to the random number of iterations needed with these methods. Thus, unless we can simulate the \( \tilde{X}_{ij} \)'s in a non-iterative manner, the SimEM algorithm is not really implementable. In Section 4 we shall discuss a way of overcoming this problem. Obviously, this problem does not occur for the StEM algorithm.

To summarize, the StEM algorithm has some theoretical advantages compared to the SimEM algorithm: The asymptotic variance is smaller, and the algorithm does not get stuck. Notice also that the SimEM algorithm is not always implementable. On the other hand, convergence is more difficult to detect in the StEM algorithm than in the SimEM algorithm.

4. Importance sampling

Sampling from the exact conditional distribution of \( X_i \) given \( Y_i = y_i \) may in some cases be impossible. As mentioned previously, iterative methods such as (adaptive) rejection sampling and MCMC methods can often be used to overcome this problem in the StEM algorithm but not in the SimEM algorithm since we would then use a random number of uniforms making it impossible to ‘re-use’ the uniforms in each iteration.

Another way of overcoming this problem is to simulate \( \tilde{X}_{ij} \) from another distribution, and then weight the complete data log-likelihoods \( \log f_{\theta}(\tilde{X}_{ij}) \), to give an unbiased estimate of the expectation \( Q(\theta|\theta') \). This is known as importance sampling.

Suppose that instead of drawing \( \tilde{X}_{ij} \) from the conditional distribution of \( X_i \) given \( Y_i = y_i \) with parameter \( \theta' \), we simulate \( \tilde{X}_{ij} \) from a distribution with density \( k_{\theta'}(\cdot|y_i) \). Assume that \( k_{\theta'}(x|y_i) > 0 \) whenever the correct conditional density of \( X_i \) given \( Y_i = y_i \) with parameter \( \theta' \), \( g_{\theta'}(x|y_i) \), is positive. Then

\[
\frac{1}{n} \sum_{i=1}^{n} \frac{1}{m} \sum_{j=1}^{m} \log f_{\theta}(\tilde{X}_{ij}) c_{\theta'}(\tilde{X}_{ij}|y_i)
\]

is an unbiased estimator of \( Q(\theta|\theta') \), when \( c_{\theta'}(\cdot|y_i) = g_{\theta'}(\cdot|y_i)/k_{\theta'}(\cdot|y_i) \), since

\[
\int \log f_{\theta}(x)c_{\theta'}(x|y_i)k_{\theta'}(x|y_i) \, d\mu(x) = \int \log f_{\theta}(x)g_{\theta'}(x|y_i) \, d\mu(x) = E_{\theta'}(\log f_{\theta}(X)|Y = y_i),
\]

where \( d\mu(x) \) denotes integration wrt the appropriate reference measure. Typically this corresponds to either ordinary integration (i.e. Lebesgue measure), when \( X_i \) is continuous, or a sum (i.e. counting measure), when \( X_i \) is discrete.
This suggests that in the simulated EM algorithms discussed so far, we may replace the assumption of $\tilde{X}_{ij}$ being drawn from the correct conditional distribution by the less restrictive assumption that $\tilde{X}_{ij}$ is drawn from a distribution with a density, which is positive whenever the density of the correct conditional distribution is.

Importance sampling may be useful for other reasons as well. It obviously changes the variance, and it may be used to reduce the variance of the resulting estimator. Also, especially in the SimEM case, there may be a considerable computational gain in simulating from a distribution, which does not depend on $h$. In this case we can in fact simulate once and re-use the simulated $\tilde{X}_{ij}$'s in each iteration, rather than re-using the uniforms.

It should be noted however that in order to use importance sampling, we must be able to calculate the importance weights $c_{\theta_0}(\tilde{X}_{ij}|y_i)$, which depend on the conditional density of $X$ given $Y = y_i$. Thus we will typically need to know this density, and hence the density of the observed data. There will still be situations, where knowing the observed data likelihood does not help us find the observed data maximum likelihood estimator; here a simulated EM algorithm may be useful. In one special case, the conditional density is not needed: If we can choose a density, $f_{\theta_0}(x)$, on the complete data sample space so that $h_{\theta_0}(y_i) = f_{\theta_0}(x)/h_{\theta_0}(x|y_i)$ is the correct density of the distribution of the observed data (with $\theta = \theta'$), then the importance weights are given as

$$
c_{\theta_0}(x|y_i) = \frac{f_{\theta_0}(x)/h_{\theta_0}(y_i)}{f_{\theta_0}(x)/h_{\theta_0}(x)} = \frac{f_{\theta_0}(x)}{f_{\theta_0}(x)/h_{\theta_0}(y_i)}.
$$

We can easily generalize Theorem 2 to give

**Theorem 4** (SimEM algorithm with importance sampling). Under assumptions similar to assumptions (A1) and (A2), there is a local asymptotic minimum, $\tilde{\theta}_n$, of

$$
\theta' \rightarrow \left\| \frac{1}{n} \sum_{i=1}^n \frac{1}{m} \sum_{j=1}^m D_\theta \log f_{\theta}(\tilde{X}_{ij}|\theta = \theta') \cdot c_{\theta}(\tilde{X}_{ij}|y_i) \right\|
$$

so that

$$
\sqrt{n}(\tilde{\theta}_n - \tilde{\theta}_0) \overset{\circ}{\sim} N\left(0, I(\theta_0)^{-1} + \frac{1}{m} I(\theta_0)^{-1} \right)
E_{\theta_0} \text{Var}_{\theta_0}(s_{X|Y}(\theta_0)c_{\theta_0}(X|Y)|Y)I(\theta_0)^{-1}.
$$

The only change to the argument given in Section 2 is that the variance in (5) is replaced by $I(\theta_0) + E_{\theta_0} \text{Var}_{\theta_0}(s_{X|Y}(\theta_0)c_{\theta_0}(X|Y)|Y)I(\theta_0)^{-1}$.
Also the proof of Theorem 1 can be generalized. Again it is a variance that is changed; in the proof in the appendix it is the variance in (14), which is replaced by \((1/m)V(\theta)^{-1}E_{\theta_0} \text{Var}_{\theta_0}(s_{X|Y}(\theta_0)c_{\theta_0}(X|Y)|Y)V(\theta_0)^{-1}\). Hence, we find

**Theorem 5** (StEM algorithm with importance sampling). Assuming ergodicity and tightness as in Theorem 1,

\[
\sqrt{n} (\hat{\theta} - \theta_0) \xrightarrow{D} \text{N}(0, \Sigma_m^w(\theta_0)),
\]

where

\[
\Sigma_m^w(\theta_0) = I(\theta_0)^{-1} + \frac{1}{m} \sum_{k=0}^{\infty} F(\theta_0)^k V(\theta_0)^{-1} E_{\theta_0} \text{Var}_{\theta_0}(s_{X|Y}(\theta_0)c_{\theta_0}(X|Y)|Y)V(\theta_0)^{-1} F(\theta_0)^k.
\]

Having seen that the results concerning the asymptotic distributions generalize, the obvious question to ask now is whether Theorem 3 can be generalized too. The proof of Theorem 3 does not generalize, since it relies crucially on the relationship between the involved information matrices, a relationship which \(E_{\theta_0} \text{Var}_{\theta_0}(s_{X|Y}(\theta_0)c_{\theta_0}(X|Y)|Y)\) plays no part in. On the other hand, it is quite easily shown that no matter which distribution we choose to sample the \(\tilde{X}_{ij}\)’s from, there are parameters for which the StEM estimators have lower variance than the SimEM estimators:

**Theorem 6.** Let \(e\) be an eigenvector of \(F(\theta_0)\) and consider the estimators \(e^t \hat{\theta}_n\) of \(e^t \theta\) derived from the two versions of the simulated EM algorithm.

Then the estimator derived from the StEM algorithm with importance sampling has smaller asymptotic variance than the estimator derived from the SimEM algorithm with importance sampling, when the same value of \(m\) and the same importance sampling distribution is used in both algorithms.

**Remark.** The choice of \(e^t \hat{\theta}_n\) as an estimator of \(e^t \theta\) reflects the fact that \(e^t \hat{\theta}_n\) is the MLE of \(e^t \theta\) when \(\hat{\theta}_n\) is the MLE of \(\theta\). Since the simulated EM algorithms approximate the EM algorithm and thus maximum likelihood estimation, \(e^t \hat{\theta}_n\) is the natural choice.

Notice also that the asymptotic variance of \(e^t \hat{\theta}_n\) is \(e^t \text{asympt Var}(\hat{\theta}_n)e\), where \(\text{asympt Var}(\hat{\theta}_n)\) is the asymptotic variance given in Theorems 4 and 5, respectively. In order to show that one of these matrices is smaller (in the usual ordering of positive-definite matrices) we have to show the similar ordering of \(u^t \text{asympt Var}(\hat{\theta}_n)u\) for any vector \(u\), i.e. that the ordering holds in all directions \(u\). Thus, we show this desired inequality for a specific choice of a direction \(u\), namely \(u = e\).
Proof. Let $\lambda$ be the eigenvalue corresponding to the eigenvector $e$. Recall from Lemma 1 that $0 \leq \lambda < 1$. Clearly $(F(\theta_i))^k e = \lambda^k e$ and

$$
e^t \sum_{k=0}^{\infty} F(\theta_i)^k V(\theta_i)^{-1} E_{\theta_i} Var_{\theta_i}(s_{X|Y}(\theta_i)c_{\theta_i}(X|Y)|Y) V(\theta_i)^{-1} F(\theta_i)^k e$$

$$= \sum_{k=0}^{\infty} \lambda^k e^t V(\theta_i)^{-1} E_{\theta_i} Var_{\theta_i}(s_{X|Y}(\theta_i)c_{\theta_i}(X|Y)|Y) V(\theta_i)^{-1} \lambda^k e$$

$$= \frac{1}{1-\lambda^2} e^t V(\theta_i)^{-1} E_{\theta_i} Var_{\theta_i}(s_{X|Y}(\theta_i)c_{\theta_i}(X|Y)|Y) V(\theta_i)^{-1} e.$$ 

Now $I(\theta_0)^{-1} = (I - F(\theta_0))^k V(\theta_0)^{-1} = \sum_{k=0}^{\infty} (F(\theta_0))^k V(\theta_0)^{-1}$ so that

$$e^t I(\theta_0)^{-1} E_{\theta_i} Var_{\theta_i}(s_{X|Y}(\theta_i)c_{\theta_i}(X|Y)|Y) I(\theta_0)^{-1} e$$

$$= \sum_{k=0}^{\infty} \lambda^k e^t V(\theta_i)^{-1} E_{\theta_i} Var_{\theta_i}(s_{X|Y}(\theta_i)c_{\theta_i}(X|Y)|Y) V(\theta_i)^{-1} \sum_{k=0}^{\infty} \lambda^k e$$

$$= \frac{1}{(1-\lambda)^2} e^t V(\theta_i)^{-1} E_{\theta_i} Var_{\theta_i}(s_{X|Y}(\theta_i)c_{\theta_i}(X|Y)|Y) V(\theta_i)^{-1} e.$$ 

Since $0 < (1 - \lambda)^2 \leq 1 - \lambda^2$, the required inequality follows. $\square$

Notice that the result holds for $\lambda = 0$, even though it is more interesting when $\lambda > 0$, in which case the inequality is strict. The case $\lambda = 0$ corresponds to a direction $e$ in which there is no missing information: $e^t E_{\theta_i} I_Y(\theta_i)e = 0$ and consequently $e^t \tilde{\theta}_n$ has asymptotic variance 0 for both algorithms.

The theorem implies that the StEM algorithm yields better estimates than the SimEM algorithm in the directions given by the eigenvectors. Since the eigenvectors span $\mathbb{R}^d$ (cf. Lemma 1), Theorem 6 shows that if we re-parameterize $\theta \rightarrow (\theta_i^t)_{i=1,\ldots,d}$ with $e_1, \ldots, e_d$ the eigenvectors of $F(\theta_i)$, then each component of this new parameter is estimated better by using StEM than by using SimEM. It is tempting to conjecture that the estimators obtained from the StEM algorithm are better than the SimEM algorithm. We have not found any counter-examples to this conjecture. However, the theorem does not prove this conjecture, since it does not take the covariances between $e_i^t \tilde{\theta}_n$ and $e_j^t \tilde{\theta}_n$ into account. The theorem only implies that the estimators obtained from the SimEM algorithm are not better in the usual (partial) ordering of positive-definite matrices than the estimators obtained from the StEM algorithm.

5. Monte Carlo experiment

We illustrate the differences between the two versions of the simulated EM algorithm with a small simulation study. McFadden and Ruud (1994) consider the following trivariate tobit model.
Let

\[
X = \begin{pmatrix} X_1 \\ X_2 \\ X_3 \end{pmatrix} \sim \mathcal{N} \left( \begin{pmatrix} \beta t_1 \\ \beta t_2 \\ \beta t_3 \end{pmatrix}, \begin{pmatrix} \sigma^2 + \rho^2 & \rho & 0 \\ \rho & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \right).
\]

We observe \( Y_i = X_i 1_{\{X > 0\}} \). The observed covariates, \( t_i \), are simulated as iid normal variates with mean 1 and variance 2. We simulate 50 replications of \( X \) with \( \beta = 1, \rho = 0.7 \) and \( \sigma^2 = 1.51 \). The parameter \( \sigma^2 \) is the conditional variance of \( X_1 \) given \((X_2, X_3)\). This parameterization is chosen because it makes the parameters variation independent and, consequently, the results easier to interpret.

To investigate the behaviour of the two versions of the simulated EM algorithm we have obtained 1000 estimates from each algorithm, for \( m = 1, 5, 10 \). To ensure convergence of the StEM algorithms a burn-in of 15,000 iterations has been used. The gbsit-software (Raftery and Lewis, 1992), which in spite of the name can be used for checking convergence of any Markov chain, suggests that a much shorter burn-in is sufficient. After burn-in, 1000 iterations have been performed, and these are used in the results reported below.

The complete data MLE does not have a closed form, and the M-step must be performed numerically. We have used the method of scoring with a fixed tolerance.

The simulated E-step involves drawing values of \((X_1, X_2)\) given that they are both negative. This cannot be done in a non-iterative manner, and as mentioned above this means that the SimEM algorithm is not really implementable. As McFadden and Ruud (1994), we have run a Gibbs sampler for 10 cycles to do this simulation in the SimEM version. In the StEM version, this simulation is done by acceptance-rejection sampling.

The distribution of the simulation estimators, \( \tilde{\theta} - \theta_0 \), obtained from the SimEM and the StEM algorithms can be decomposed into a contribution from the simulations and a contribution from the data, \( \tilde{\theta} - \hat{\theta} \) and \( \hat{\theta} - \theta_0 \), respectively. The latter – the distribution of the MLE – is a function of the data only and gives no information on how well the two simulation estimators behave. The simulation estimators attempt to approximate the maximum likelihood estimator (cf. Example 1), since the simulated EM algorithm approximates the EM algorithm. Therefore, it is the distribution of \( \tilde{\theta} - \hat{\theta} \) that is of interest when evaluating the performance of the simulation estimators. Hence, we look at the empirical distribution of the difference between the simulation estimators, \( \tilde{\theta} \), obtained from the SimEM and the StEM algorithms and the maximum likelihood estimator, \( \hat{\theta} \).

We give summary statistics for the estimators of each parameter (\( \beta, \rho, \) and \( \sigma^2 \)) in Tables 1–3. The MLE based on the observed data is shown in the tables for completeness. Histograms and QQ-plots of the simulated distribution of \( \tilde{\theta} - \hat{\theta} \)
Table 1
Simulation results for $\beta - \hat{\beta}$

<table>
<thead>
<tr>
<th>$\beta$ : ($\hat{\beta} = 1.042$)</th>
<th>Mean</th>
<th>S.D.</th>
<th>25%</th>
<th>Median</th>
<th>75%</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m = 1$:</td>
<td>StEM</td>
<td>0.0117</td>
<td>0.0196</td>
<td>-0.0249</td>
<td>-0.0115</td>
</tr>
<tr>
<td></td>
<td>SimEM</td>
<td>-0.0798</td>
<td>0.0122</td>
<td>-0.0878</td>
<td>-0.0808</td>
</tr>
<tr>
<td>$m = 5$:</td>
<td>StEM</td>
<td>-0.0635</td>
<td>0.0104</td>
<td>-0.0702</td>
<td>-0.0637</td>
</tr>
<tr>
<td></td>
<td>SimEM</td>
<td>-0.0798</td>
<td>0.0055</td>
<td>-0.0838</td>
<td>-0.0799</td>
</tr>
<tr>
<td>$m = 10$:</td>
<td>StEM</td>
<td>-0.0118</td>
<td>0.0065</td>
<td>-0.0162</td>
<td>-0.0117</td>
</tr>
<tr>
<td></td>
<td>SimEM</td>
<td>-0.0797</td>
<td>0.0038</td>
<td>-0.0822</td>
<td>-0.0798</td>
</tr>
</tbody>
</table>

S.D. is the standard deviation, 25% and 75% are the lower and upper quartiles.

Table 2
Simulation results for $\rho - \hat{\rho}$

<table>
<thead>
<tr>
<th>$\rho$ : ($\hat{\rho} = 1.101$)</th>
<th>Mean</th>
<th>S.D.</th>
<th>25%</th>
<th>Median</th>
<th>75%</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m = 1$:</td>
<td>StEM</td>
<td>0.0386</td>
<td>0.0977</td>
<td>-0.0261</td>
<td>0.0417</td>
</tr>
<tr>
<td></td>
<td>SimEM</td>
<td>-0.2122</td>
<td>0.0917</td>
<td>-0.2733</td>
<td>-0.2118</td>
</tr>
<tr>
<td>$m = 5$:</td>
<td>StEM</td>
<td>0.0276</td>
<td>0.0408</td>
<td>-0.0003</td>
<td>0.0269</td>
</tr>
<tr>
<td></td>
<td>SimEM</td>
<td>-0.2179</td>
<td>0.0393</td>
<td>-0.2444</td>
<td>-0.2168</td>
</tr>
<tr>
<td>$m = 10$:</td>
<td>StEM</td>
<td>0.0372</td>
<td>0.0318</td>
<td>0.0148</td>
<td>0.0371</td>
</tr>
<tr>
<td></td>
<td>SimEM</td>
<td>-0.2185</td>
<td>0.0277</td>
<td>-0.2361</td>
<td>-0.2180</td>
</tr>
</tbody>
</table>

S.D. is the standard deviation, 25% and 75% are the lower and upper quartiles.

Table 3
Simulation results for $\sigma^2 - \hat{\sigma}^2$

<table>
<thead>
<tr>
<th>$\sigma^2$ : ($\hat{\sigma}^2 = 1.538$)</th>
<th>Mean</th>
<th>S.D.</th>
<th>25%</th>
<th>Median</th>
<th>75%</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m = 1$:</td>
<td>StEM</td>
<td>-0.1198</td>
<td>0.1594</td>
<td>-0.2287</td>
<td>-0.1347</td>
</tr>
<tr>
<td></td>
<td>SimEM</td>
<td>0.2712</td>
<td>0.1927</td>
<td>0.1390</td>
<td>0.2526</td>
</tr>
<tr>
<td>$m = 5$:</td>
<td>StEM</td>
<td>-0.1703</td>
<td>0.0623</td>
<td>-0.2112</td>
<td>-0.1750</td>
</tr>
<tr>
<td></td>
<td>SimEM</td>
<td>0.2792</td>
<td>0.0836</td>
<td>0.2227</td>
<td>0.2772</td>
</tr>
<tr>
<td>$m = 10$:</td>
<td>StEM</td>
<td>-0.1052</td>
<td>0.0523</td>
<td>-0.1424</td>
<td>-0.1066</td>
</tr>
<tr>
<td></td>
<td>SimEM</td>
<td>0.2809</td>
<td>0.0584</td>
<td>0.2402</td>
<td>0.2799</td>
</tr>
</tbody>
</table>

S.D. is the standard deviation, 25% and 75% are the lower and upper quartiles.

for each parameter when $m = 1$ are shown in Figs. 1–3. Each figure contains two histograms and two QQ-plots. The top row in each figure corresponds to the StEM estimator, while the bottom row corresponds to the SimEM estimator. The two histograms in each figure have the same axes and are thus directly
Fig. 1. Histograms and QQ-plots for $\beta - \hat{\beta}$, $m = 1$.

Fig. 2. Histograms and QQ-plots for $\delta - \hat{\delta}$, $m = 1$. 
comparable. The displayed part of the abscissae of the histograms all contain 0, the point corresponding to 'no simulation-bias', i.e. $\bar{\theta} = \hat{\theta}$. Thus, a good method will give a narrow histogram centered at 0. In the QQ-plots a straight line indicates that $\bar{\theta} = \hat{\theta}$ is approximately normal.

Looking first at the results for $\beta$ (Table 1 and Fig. 1) we notice that the variance of the StEM estimator is larger than the variance of the SimEM estimator. This is probably due to the small sample size, as the results of Section 2 indicate that the asymptotic variance is smaller for the StEM estimator.

The bias of the SimEM estimator is however large compared to the bias of the StEM estimator, and the StEM estimator clearly performs better in this example in spite of the larger variance. The fact that the bias of the StEM estimator in the case $m = 5$ is larger than when $m = 1$ or 10 is surprising but also occurs in additional runs of the StEM algorithm for this data (not shown here).

The QQ-plot suggests that both estimators are reasonably close to normally distributed. The tails of the distributions may be a bit too heavy, and the StEM estimator seems to be slightly closer to normal than the SimEM estimator.

For $\rho$ we see that the bias is significantly larger for the SimEM estimator than for the StEM estimator, but the variances are now roughly equal. Again the StEM estimator is superior. Fig. 2 suggest that the estimators are not too far from normally distributed.
We may note that the SimEM estimator is less biased than the StEM estimator when looking at the empirical distribution of \( \tilde{\rho} - \rho_0 \) rather than \( \hat{\rho} - \hat{\rho} \). This is due to the large positive bias in the MLE and is thus an effect of the data and not the simulations. Consequently, we would not expect the SimEM estimator to perform better than the StEM estimator on all data sets, though it may perform better when \( \hat{\rho} \) is positively biased. In particular, in large samples, where we would expect the bias of the MLE to be negligible, we would expect the StEM estimator to be superior. Incidentally, this is the only parameter, \( \theta \), for which \( \tilde{\theta} - \theta_0 \) is more biased in the StEM case than in the SimEM case.

When estimating \( \sigma^2 \), the StEM estimator again performs better; the bias is larger for the SimEM estimator, and the variances are here smaller for the StEM estimator.

Fig. 3 shows that the distribution of \( \tilde{\sigma}^2 - \hat{\sigma}^2 \) is far from normal for \( m = 1 \), but this is what we would expect. Ignoring the multivariate nature of the data and the fact that \( \sigma^2 \) is not the only parameter, the problem of estimating \( \sigma^2 \) is very similar to Example 2. Thus, we would expect the SimEM estimator to be distributed approximately as \( a/(50 - \chi^2_{12m}) \) for some constant \( a \) depending on the observed data, since only 12 \( X_i \)’s are censored. In particular, \( \tilde{\sigma}^2 - \hat{\sigma}^2 \) is not approximately normal for small values of \( m \) in the SimEM case. The distribution of the StEM estimator is more difficult to describe but since the conditional distribution of the next step of the Markov chain in Example 2 given the past is an affine transformation of a \( \chi^2_{12m} \)-distribution, we would not expect \( \tilde{\sigma}^2 - \hat{\sigma}^2 \) to be approximately normal for small values of \( m \) in the StEM case, either. QQ plots for larger values of \( m \) (not shown here) shows that the distribution of \( \tilde{\sigma}^2 - \hat{\sigma}^2 \) gets closer to the normal distribution, when \( m \) increases. This does not necessarily mean that the distribution of \( \tilde{\sigma}^2 - \sigma^2 \) gets closer to normal as \( m \) increases, since \( n \) is kept fixed.

For all parameters, the standard deviations decrease as \( m \) increases. The decrease is roughly \( 1/\sqrt{m} \approx 0.44 \) when \( m \) increases from 1 to 5 and \( 1/\sqrt{2} \approx 0.71 \) when going from \( m = 5 \) to 10, as we would expect from the asymptotic results. This fact and the distribution of \( \tilde{\sigma}^2 - \hat{\sigma}^2 \) indicates that as \( m \) increases the empirical distribution of the simulation estimators minus the MLEs gets closer to normality, as we would expect. As noted above, this does not mean that the asymptotic results of Section 2 hold for this sample size, since the sample size (\( n \)) is small.

The bias of the SimEM estimator is unaffected by the choice of \( m \), whereas the mean of the StEM estimator is always lower for \( m = 5 \). This leads to larger bias when \( m = 5 \) in the estimators of \( \beta \) and \( \hat{\sigma}^2 \), but smaller in the positively biased estimator of \( \rho \). As mentioned previously this shows up in additional simulations and is thus not ‘accidental’. Whether it is an effect of the observations or a more general phenomenon remains to be seen.

The components of the StEM estimator appears to be independent whereas the SimEM estimators of \( \rho \) and \( \sigma^2 \) are negatively correlated; the correlation
coefficient is approximately $-0.5$, independent of $m$. Hence, some care should be exercised when interpreting the results for $\rho$ and $\sigma^2$.

In conclusion, the StEM algorithm performs better than the SimEM algorithm in this example, even though the sample size is too small for the asymptotic results to hold. The SimEM algorithm is faster than the StEM algorithm in this example, but the number of iterations used in the StEM version is a lot larger than what is necessary for convergence. We might hope to improve the SimEM algorithm by increasing $m$ but as we have seen the main problem with the SimEM estimator is bias rather than variance, and the bias does not seem to decrease for moderate values of $m$. One could also hope to improve the SimEM algorithm in this example by allowing more iterations in the Gibbs sampler used for simulating $(X_1, X_2)$ given that they are both negative. However, since only 7 observations have both $X_1$ and $X_2$ censored, the effect of this will probably not be very large.

6. Other simulation estimators

We conclude this paper with a brief discussion of some other simulation estimators, namely estimators derived from the Monte Carlo likelihood method, the Method of Simulated Scores, and Gibbs sampling. We will avoid technical details and only mention similarities and differences between these algorithms and the simulated EM algorithm.

Monte Carlo likelihood methods (cf. Geyer, 1996) resemble the SimEM algorithm in trying to estimate an entire function and then maximizing this estimated function. Rather than estimating the EM update, $M$, it is the observed data score function, which is estimated. This is done by simulating $\tilde{X}_{ij}$ from the conditional distribution of $X_i$ given $Y_i = y_i$ with parameter $\theta'$ and calculating the function

$$
\theta \to \frac{1}{n} \sum_{i=1}^{n} D_{\theta} \log \left( \frac{1}{m} \sum_{j=1}^{m} f_{\theta}(\tilde{X}_{ij}) \right). \tag{11}
$$

The expectation of the inner sum in (11) is the observed data likelihood function at $\theta$. Thus the estimated function (11) approximates $(1/n) \sum_{i=1}^{n} s_{y_i}(\theta)$, the observed data score function. The estimator for $\theta$ is a root of this function, typically found using Newton–Raphson or a similar optimization method.

However, since the logarithm is a strictly concave function, the simulated score function (11) is a biased estimate of the observed data score function. Moreover, the bias does not disappear as $n \to \infty$ if $m$ is kept fixed. Thus in order to obtain consistent estimators, we need to let $m$ increase with $n$. In this respect the MC likelihood method differs from the simulated EM algorithms which yield consistent estimators for any fixed value of $m$. In other words, for the MC likelihood estimator to be consistent, we need to calculate the observed data likelihood function (using Monte Carlo integration) rather than just estimate it.
We note that MC likelihood calculations are amenable to importance sampling as discussed in Section 4 as well as to MCMC methods; the $\hat{X}_{ij}$'s can be simulated from a Markov chain with the correct conditional distribution of $X$ given $Y = y_i$ as stationary initial distribution. Furthermore, the entire chain (after burn-in is discarded) can be used to calculate the inner sum in (11).

The method of simulated scores (MSS) suggested by Hajivassiliou and McFadden (1997) is very similar to the Monte Carlo likelihood method mentioned above. In both methods the observed data score function is estimated by simulation. One could say that MC likelihood as given by Geyer is a special case of MSS; with missing data a particularly useful special case since it is based on the complete data likelihood rather than the observed data likelihood. Hajivassiliou and McFadden (1997) suggest different estimators of the score function for LDV models and give asymptotic results. Most of these suggestions are biased for finite values of $m$, and it is discussed how fast $m$ should increase with $n$ in order to get asymptotically unbiased estimators.

There is an obvious Gibbs sampling analogue – often referred to as the Data Augmentation algorithm (cf. Wei and Tanner, 1990) – to StEM. Here we draw $\hat{X}_{ij}$ from the conditional distribution of $X$ given $Y = y_i$ with parameter $\theta'$ and then – rather than finding the next value of $\theta$ by maximization – a new value of $\theta'$ is drawn from a given distribution dependent on the simulated values, $\hat{X}_{ij}$. This new value of the parameter is then used as $\theta'$ in the next iteration. Using new random numbers in each iteration we get a Markov chain which is ergodic under weak conditions. In order to decide which distribution to simulate the parameter values from, we need to specify a Bayesian prior distribution on the parameter space. The distribution used for simulating $\theta$ in the iterations is then specified so that the stationary initial distribution of the Markov chain made by the simulated $\hat{X}_{ij}$'s and $\theta$-values is the posterior distribution of the parameter and the unobserved data.

The usual Gibbs sampler – data augmentation with $m = 1$ – can be seen as a Bayesian version of StEM with $m = 1$. Though Bayesian in nature, the posterior mean of $\theta$ (i.e. the mean of the posterior distribution) typically is a good approximation to the observed data MLE. This posterior mean can be estimated by taking an average of the simulated $\theta$-values (after burn-in). Notice that since this average in practice is always a finite average (of say $M$ values) there will be additional variance due to simulations with this method as well. The StEM analogue to this estimator is to run the StEM algorithm for $M$ iterations after convergence and then average the simulated $\theta$-values. There is one major difference here, though; the posterior mean of $\theta$ is typically known to exist, but the mean of the StEM Markov chain only exists in special cases, for instance when the parameter space is compact. Moreover, it still remains to be shown if this mean generally is a good (consistent, say) estimator of $\theta$. Diebolt and Celeux (1993) show this for the stochastic EM algorithm (StEM with $m = 1$) for the special case of estimation of mixing proportions.
From a (strict) Bayesian point of view the posterior distribution is the aim of the inference. From this we can get for instance the posterior mean and its distribution. As the number of iterations tend to infinity the Gibbs sampler gives us the posterior distribution, regardless of the sample size. Thus in this sense, the justification for using the Gibbs sampler is not asymptotic in \( n \), only in the number of iterations. However, if we wish to use the posterior mean as an approximation to the MLE, we need \( n \) to be large. Similarly, by running StEM sufficiently long, we can find the distribution of the StEM estimator given the data (as in Section 5) but unless the sample size is large we cannot know in general if the estimator is ‘good’.

Which estimator – the StEM or the Gibbs sampling estimator – has the lowest asymptotic variance will depend on the distributions chosen in the Gibbs sampler. The (prior) distribution of \( \theta \) is subjective, and we can choose this to decrease the variance. Typically, however we would rather run the algorithm for more iterations, since the choice of the prior is usually preferred to be left flat in order to minimize the influence of the prior on the resulting estimator. The StEM ‘analogue’ to choosing a better prior distribution of \( \theta \) is importance sampling.

Chib (1996) compares the Gibbs sampling method to the stochastic EM algorithm (StEM with \( m = 1 \)) and the MCEM algorithm (StEM with \( m = 1000 \)) in Markov mixture models. He finds that the estimates obtained by the MCEM algorithm are close to the estimates obtained using Gibbs sampling and a fully Bayesian model. The estimates obtained using the stochastic EM algorithm are more variable, as one should expect.

All algorithms have their advantages and disadvantages. The StEM algorithm has better asymptotic properties than SimEM (lower variance) but may take longer to run. The MC likelihood method requires Monte Carlo integration rather than just estimation but will typically give better estimates, since the additional simulation brings down the variance. On the other hand the simulated EM algorithms are – like the EM algorithm – ‘derivative-free’, which makes each iteration of the algorithm less complicated and thus possibly faster. The Gibbs sampling method is obviously preferable for Bayesian inference, but whether it is preferable generally is more questionable; for instance MC likelihood calculates the observed data MLE (subject to simulation error) rather than the posterior mean found by Gibbs sampling (again subject to simulation error), and the MLE will typically be more appealing in non-Bayesian statistics.

Acknowledgements

I am grateful to Richard Gill for calling my attention to the subject of simulated EM algorithms. Comments from an associate editor and two referees have been very stimulating.
Appendix. Proof of Theorem 1

We shall here prove Theorem 1 in the special case of exponential family models. The EM algorithm is particularly simple when the complete data model is an exponential family. Furthermore, this class contains many models of common interest (cf. Ruud, 1991). We stress, however, that the proof can be carried through under general smoothness assumptions (see Nielsen, 2000). Thus the theorem holds more generally, but for simplicity we restrict our attention to this simpler case. For details on incomplete observations from an exponential family, see Sundberg (1971).

Suppose that the model for the complete data, \( X \), is an exponential family, i.e.

\[
f_\theta(x) = \frac{1}{\phi(\theta)} \exp(\theta t(x)) b(x).
\]

The distribution of \( X \) given \( Y = y \) is given by the density

\[
k_\theta(x|y) = \frac{1}{\phi_y(\theta)} \exp(\theta t(x)) b(x) \quad \text{for } x \in \{ x : Y(x) = y \}
\]

with \( \phi_y(\theta) \) given as the appropriate normalization constant or, equivalently, as \( \phi(\theta) \) times the observed data likelihood. This is again an exponential family.

The observed data likelihood equations are

\[
E_\theta(t(X)) = \frac{1}{n} \sum_{i=1}^{n} E_\theta(t(X_i)|Y_i = y_i).
\]

The observed data maximum likelihood estimator, \( \hat{\theta}_n \), is a solution to (12). Let \( \tau(\theta) \) denote the left-hand side of this equation. From standard exponential family theory, it is known that \( \tau \) has a continuously differentiable inverse. Hence, the EM-update, \( M \), is given by

\[
M(\theta) = \tau^{-1}\left( \frac{1}{n} \sum_{i=1}^{n} E_\theta(t(X_i)|Y_i = y_i) \right).
\]

In StEM, the right-hand side of (12) is estimated, leading to the simulated likelihoods equations

\[
\tau(\theta) = \frac{1}{n} \sum_{i=1}^{n} \frac{1}{m} \sum_{j=1}^{m} t(\tilde{X}_{ij}).
\]

Having thus defined the necessary ingredients, we shall have a look at what happens in a single iteration of StEM starting from \( \theta_0 \). We start by simulating \( \tilde{X}_{ij} \) using the observed data and \( \theta_0 \). For each \( i \) and every \( j \), we get \( E(t(\tilde{X}_{ij})) = E_{\theta_0}(t(X_i)|Y_i = y_i) \) so that the right-hand side of (13) is an unbiased
estimate of the right-hand side of (12). We then calculate the next $\theta$-value as

$$\theta' = \tau^{-1} \left( \frac{1}{n} \sum_{i=1}^{n} \frac{1}{m} \sum_{j=1}^{m} t(X_{ij}) \right).$$

Now, we see that

$$\sqrt{n}(\theta' - \hat{\theta}_n) = \sqrt{n}(M(\theta_0) - M(\hat{\theta}_n)) + \sqrt{n}(\theta' - M(\theta_0))$$

$$= D_\theta M(\theta|\theta = \hat{\theta}) + o(||\theta_0 - \hat{\theta}_n||) + \sqrt{n}(\theta' - M(\theta_0))$$

using first the fact that $M(\hat{\theta}_n) = \hat{\theta}_n$ and then a Taylor expansion. By standard exponential family theory $D_\theta \tau(\theta) = V(\theta)$ and $D_\theta E_\theta(t(X_i)|Y_i = y_i) = I_{Y_i}(\theta)$. Now since $\hat{\theta}_n \to \theta_0$

$$D_\theta M(\theta|\theta = \theta_0) = \left( V(\theta_0) \right)^{-1} \left( \frac{1}{n} \sum_{i=1}^{n} I_{Y_i}(\theta_0) \right) \to F(\theta_0).$$

This convergence holds for almost every observed sequence of $y_i$’s by the uniform strong law of large numbers. Secondly, by the central limit theorem and continuous mapping

$$\sqrt{n}(\theta' - M(\theta_0)) \to N\left( 0, \frac{1}{m} V(\theta_0)^{-1} E_{\theta_0} I_Y(\theta_0) V(\theta_0)^{-1} \right)$$

for almost every sequence of observed $y_i$’s, since $D_\theta \tau(\theta) = V(\theta)$. Thus we can write

$$\sqrt{n}(\theta' - \hat{\theta}_n) = F(\theta_0)^\top \sqrt{n}(\theta_0 - \hat{\theta}_n) + \varepsilon + o_p(1),$$

where $\varepsilon \sim N(0, (1/m)V(\theta_0)^{-1} E_{\theta_0} I_Y(\theta_0) V(\theta_0)^{-1})$.

We can show (using the smoothness properties of exponential families) that (15) holds even if we replace $\theta_0$ by any sequence $\theta_n = \hat{\theta}_n + O_p(1/\sqrt{n})$. This implies that the transition probabilities of the Markov chain defined by the StEM algorithm converge weakly to the transition probabilities of the Gaussian multivariate AR(1) process. By Lemma 1, this is a stationary process.

Suppose now that we start the Markov chain defined by StEM in its stationary initial distribution; denote this chain by $(\tilde{\theta}_n^k)_{k \in \mathbb{N}}$. Then the weak convergence of the transition probabilities implies that this Markov chain converge weakly as a discrete time process to the Gaussian AR(1) process (defined as the limit of (15)) if it is tight. In particular, the marginal distribution of $\tilde{\theta}_n^k$ (for any $k$) converge in distribution to the stationary distribution of the Gaussian AR(1) process.

In practice, we cannot start StEM from the stationary distribution; if we could there would be little reason for using the StEM algorithm at all. Thus we need to
consider \((\bar{\theta}_n(k))_{k\in\mathbb{N}}\), the Markov chain obtained when starting the StEM algorithm at an arbitrary point.

Since Markov chains ‘forget’ where they started, the total variation distance between \(\bar{\theta}_n(k)\) and \(\bar{\theta}_n^0(k)\) will go to zero as \(k \to \infty\). Notice that ergodicity of the StEM Markov chain is necessary here. Thus for any \(\varepsilon > 0\) and any continuous, bounded function \(\kappa: \Theta \to \mathbb{R}\) we can first choose \(n\) (using the weak convergence) so that (with \(Z\) distributed according to the limiting stationary Gaussian AR(1) process) for any \(k\)
\[
|\mathbb{E}_\kappa(\bar{\theta}_n^0(k)) - \mathbb{E}_\kappa(Z)| < \varepsilon/2
\]
and then \(k_n\) (using the convergence in total variation) so that for any \(k > k_n\)
\[
|\mathbb{E}_\kappa(\bar{\theta}_n^0(k)) - \mathbb{E}_\kappa(\bar{\theta}_n(k))| < \varepsilon/2.
\]
This proves that \(\bar{\theta}_n(k_n)\) converges weakly to the stationary initial distribution of the ‘limiting’ Gaussian AR(1)-process for some sequence \(k_n\). In particular, this holds for \(k_n = \infty\), i.e. in the limit, if the sequence \((\bar{\theta}_n)_n\) is tight as assumed in the theorem.

Thus for almost every observed \(y_i\)-sequence
\[
\sqrt{n}(\bar{\theta}_n - \hat{\theta}_n) \xrightarrow{D} \mathcal{N}\left(0, \frac{1}{m} \sum_{k=0}^{\infty} F(\theta_0)^t V(\theta_0)^{-1} E_{\theta_0} I_Y(\theta_0) V(\theta_0)^{-1} F(\theta_0)^k\right).
\]
Hence unconditionally (see for instance Lemma 1 in Schenker and Welsh, 1988)
\[
\sqrt{n}(\bar{\theta}_n - \theta_0) = \sqrt{n}(\bar{\theta}_n - \hat{\theta}_n) + \sqrt{n}(\hat{\theta}_n - \theta_0)
\]
\[
\xrightarrow{D} \mathcal{N}\left(0, I(\theta_0)^{-1} + \frac{1}{m} \sum_{k=0}^{\infty} F(\theta_0)^t V(\theta_0)^{-1} E_{\theta_0} I_Y(\theta_0) V(\theta_0)^{-1} F(\theta_0)^k\right)
\]
as stated in Theorem 1. \(\square\)

References


