Statistica Sinica **5**(1995), 55-75

MAXIMUM LIKELIHOOD ESTIMATION VIA THE ECM ALGORITHM: COMPUTING THE ASYMPTOTIC VARIANCE

David A. van Dyk, Xiao-Li Meng and Donald B. Rubin*

*University of Chicago and Harvard University**

Abstract. This paper provides detailed theory, algorithms, and illustrations for computing asymptotic variance-covariance matrices for maximum likelihood estimates using the ECM algorithm (Meng and Rubin (1993)). This Supplemented ECM (SECM) algorithm is developed as an extension of the Supplemented EM (SEM) algorithm (Meng and Rubin (1991a)). Explicit examples are given, including one that demonstrates SECM, like SEM, has a powerful internal error detecting system for the implementation of the parent ECM or of SECM itself.

Key words and phrases: Contingency table, convergence rate, EM algorithm, Fisher information, incomplete data, IPF, missing data, SEM algorithm.

1. Introduction

The EM algorithm (Dempster, Laird and Rubin (1977)) is a formalization of an old ad hoc method for handling missing data. If we had the missing data, we could estimate the parameters of a particular model using standard complete-data techniques. On the other hand, if we knew the model parameters, we could impute the missing data according to the model. This leads naturally to an iterative scheme. The advantage of the EM formulation over its ad hoc predecessor is that it recognizes that the correct imputation is through the complete-data sufficient statistics, or more generally through the complete-data loglikelihood function. Since EM separates the complete-data analysis from the extra complications due to missing data, it is both conceptually and computationally simple. When facing an incomplete-data problem, we can first ask what would be done if there were no missing values, and then proceed with the help of EM to deal with the missing data, assuming that the missing data mechanism (Rubin (1976)) has been taken into account. This advantage has helped EM win great populatity among practical users. Meng and Pedlow's (1992) bibliography reveals that there are more than 1,000 EM related articles in almost 300 journals, most of which are outside the field of statistics.

In some cases, the complete-data problem itself may be complicated. For instance, when a model has many parameters, finding maximum likelihood estimates (MLEs) can be a demanding task. A natural strategy, in general, is to break a big problem into several smaller ones. In a case with many parameters, if some of the model parameters were known, it might be easier to estimate the rest. In the complete-data problem, we can partition the parameters into several sets, and estimate one set conditional on all the others. This technique is well-known in the numerical analysis literature as the cyclic coordinate ascent method (e.g. Zangwill (1969)) and is called, in statistical terms, the Conditional Maximization or CM algorithm by Meng and Rubin (1993). The ECM algorithm (Meng and Rubin (1993)) is an efficient combination of the CM and EM algorithms. It replaces the maximization step of EM with a set of conditional maximization steps, and thus splits a difficult maximization problem into several easier ones. Consequently, in many practical applications, ECM extends the flexibility and power of EM and retains the stability of EM in the sense of monotonic convergence of the likelihood along the induced sequence to the MLE.

Besides computing point estimates, statistical inference requires measures of uncertainty, for example (asymptotic) variance-covariance matrix of the estimates. The Supplemented EM (SEM) algorithm (Meng and Rubin (1991a)) computes such matrices using a sequence of EM iterates to obtain the matrix rate of convergence of EM. This rate is then used to inflate the complete-data asymptotic variance matrix to obtain the asymptotic variance matrix for the observed-data MLEs. A key feature of SEM is that it requires only the code for EM and the code for computing the complete-data asymptotic variance matrix.

Here we develop and illustrate an analogous supplemented algorithm for ECM, namely, SECM, which computes the asymptotic variance-covariance matrix of the MLEs. In addition to requiring the computation of both the rate of convergence of ECM and the complete-data variance-covariance matrix, it requires the computation of the rate of convergence of the CM algorithm. The computations of SECM, however, remain as simple as SEM in the sense that they only require the ECM code along with the code for computing the complete-data variance matrix. Although our presentation is focused on the asymptotic variance-covariance matrix of the MLEs, the SECM algorithm can just as easily be applied to compute the asymptotic posterior variance-covariance matrix when ECM is used to find a posterior mode, which includes penalized likelihood models as a special case (e.g. Segal, Bacchetti and Jewell (1994)).

After providing the necessary theoretical development in Section 2, we detail in Section 3 the computational steps of SECM. Section 4 presents two examples to illustrate the SECM algorithm. The last section offers discussion on some practical issues involved in implementing SECM.

2. Methodological Development

2.1. Notation and general setting

As in Meng and Rubin (1991a, 1993) and Meng (1994), let $f(Y|\theta)$ be a density for the complete-data Y, where $\theta = (\theta_1, \ldots, \theta_d)$ is a d-dimensional model parameter with domain Θ. In the presence of missing data we write $Y = (Y_{\rm obs}, Y_{\rm mis})$, where $Y_{\rm obs}$ represents the part of Y that is observed and $Y_{\rm mis}$ the part that is missing. The object is to find the MLE of θ given $Y_{\rm obs}$, $\theta^\star$, as well as the asymptotic variance-covariance matrix of $(\theta - \theta^\star)$. That is, we seek both the $\theta^\star$ that maximizes the observed-data loglikelihood, $L(\theta|Y_{\rm obs}) = \log f(Y_{\rm obs}|\theta)$ and the corresponding observed information matrix

$$I_{\rm o}(\theta^\star|Y_{\rm obs}) = -\frac{\partial^2 L(\theta|Y_{\rm obs})}{\partial\theta\partial\theta^\top}\Big|_{\theta=\theta^\star}. \tag{2.1.1}$$

The use of the inverse of (2.1.1) as an asymptotic variance requires that $\theta^\star$ is in the interior of Θ, a condition that is also needed for the general EM theory developed in Dempster, Laird and Rubin (1977) as well as in Wu (1983). We therefore assume $\theta^\star$ is in the interior of Θ throughout this paper. Because of the missing data, direct analytic computation of $\theta^\star$ and $I_{\rm o}(\theta^\star|Y_{\rm obs})$ can be difficult or impossible. This is the setting in which EM and its extensions are useful.

2.2. The EM and ECM algorithms

Starting with an initial value $\theta^{(0)} \in \Theta$, the EM algorithm finds $\theta^\star$ by iterating between the following two steps ($t = 0, 1, \ldots$):

E step: Impute the unknown complete-data loglikelihood $L(\theta|Y) = \log f(Y|\theta)$ by its conditional expectation given the current estimate $\theta^{(t)}$:

$$Q(\theta|\theta^{(t)}) = \int L(\theta|Y) f(Y_{\rm mis}|Y_{\rm obs}, \theta^{(t)}) {\rm d}Y_{\rm mis}. \tag{2.2.1}$$

M step: Determine $\theta^{(t+1)}$ by maximizing the imputed loglikelihood $Q(\theta|\theta^{(t)})$:

$$Q(\theta^{(t+1)}|\theta^{(t)}) \geq Q(\theta|\theta^{(t)}), \quad \text{for all } \theta \in \Theta. \tag{2.2.2}$$

As we mentioned in Section 1, in some cases the M step is not in closed form. Directly implementing EM then requires undesirable nested iterations. In many such cases, the ECM algorithm, which replaces the maximization of $Q(\theta|\theta^{(t)})$ by several simpler conditional maximizations, is more useful. Specifically, let $G = \{g_s(\theta), s = 1, \ldots, S\}$ be a set of S (≥ 1) preselected (vector) functions that are "space filling" (Meng and Rubin (1993)) in the sense of allowing maximization over the full space Θ. With ECM, the M step is replaced by S Conditional Maximization (CM) steps:

sth *CM step:* Find $\theta^{(t+s/S)}$ such that

$$Q(\theta^{(t+s/S)}|\theta^{(t)}) \geq Q(\theta|\theta^{(t)}), \quad \text{for all } \theta \in \Theta_s^{(t)} \equiv \{\theta \in \Theta : g(\theta) = g(\theta^{(t+(s-1)/S)})\}, \tag{2.2.3}$$

where $s = 1, \ldots, S$, and the next iterate $\theta^{(t+1)} \equiv \theta^{(t+S/S)}$. The rational behind the *CM* steps is that in problems where maximizing $Q(\theta|\theta^{(t)})$ over $\theta \in \Theta$ is difficult, it may be possible to choose G so that it is simple to maximize over $\theta \in \Theta_s^{(t)}$ for $s = 1, \ldots, S$.

For example, a common useful choice of G is to choose $g_s(\theta) = (\vartheta_1, \ldots, \vartheta_{s-1}, \vartheta_{s+1}, \ldots, \vartheta_S)$ for $s = 1, \ldots, S$, where $(\vartheta_1, \ldots, \vartheta_S)$ is a partition of θ. In other words, at the sth CM step, we maximize $Q(\theta|\theta^{(t)})$ over ϑ_s with the rest of the $S-1$ subvectors fixed at their previous estimates. This common special class of ECM is called the partitioned ECM (PECM) algorithm by Meng and Rubin (1992). More complicated choices of G can also be useful in practice, as we will see in Section 4.2.

2.3. The SEM algorithm

Having described EM and ECM, we briefly review the SEM algorithm before extending it to SECM. The SEM algorithm is built upon the following fundamental identity, established in Dempster, Laird and Rubin (1977), under the condition that $Q(\theta|\theta^{(t)})$ is maximized by setting its first derivative equal to zero,

$$DM^{EM} = I_{\text{om}} I_{\text{oc}}^{-1}. \tag{2.3.1}$$

In (2.3.1), DM^{EM} is the Jacobian of the EM mapping, $\theta^{(t+1)} = M^{EM}(\theta^{(t)})$, at $\theta = \theta^\star$,

$$I_{\text{om}} = \int -\frac{\partial^2 \log f(Y_{\text{mis}}|Y_{\text{obs}}, \theta)}{\partial\theta\partial\theta^\top} f(Y_{\text{mis}}|Y_{\text{obs}}, \theta) dY_{\text{mis}} \Big|_{\theta=\theta^\star} \tag{2.3.2}$$

is the expected missing information, and

$$I_{\text{oc}} = \int -\frac{\partial^2 \log f(Y|\theta)}{\partial\theta\partial\theta^\top} f(Y_{\text{mis}}|Y_{\text{obs}}, \theta) dY_{\text{mis}} \Big|_{\theta=\theta^\star} \tag{2.3.3}$$

is the expected complete information (see Meng and Rubin (1991a)). The identity (2.3.1) is fundamental because it directly relates the rate of convergence of EM, namely, DM^{EM}, with the (*matrix*) *fraction of missing information*, $I_{\text{om}} I_{\text{oc}}^{-1}$. It underlies the SEM computations because the desired information matrix (2.1.1) can be written as the difference between the complete and missing information (e.g. Orchard and Woodbury (1972), Meng and Rubin (1991a))

$$I_{\text{o}}(\theta^\star|Y_{\text{obs}}) = I_{\text{oc}} - I_{\text{om}} = [I_d - I_{\text{om}} I_{\text{oc}}^{-1}] I_{\text{oc}} = [I_d - DM^{EM}] I_{\text{oc}}, \tag{2.3.4}$$

where I_d is the $d \times d$ identity matrix. In other words, to compute $I_o(\theta^\star|Y_{obs})$, we need only compute DM^{EM} and I_{oc}. When $f(Y|\theta)$ is from the exponential family, as is typically the case when the E step is tractable, $I_{oc} = I_o(\theta^\star|S^\star(Y_{obs}))$, where $S^\star(Y_{obs}) = E(S(Y)|Y_{obs}, \theta^\star)$, as found at the last E step; we thus can compute I_{oc} using standard complete-data procedures. Computing DM^{EM} can be accomplished by numerical differentiation of the EM mapping. Details of these SEM calculations are provided in Meng and Rubin (1991a) and will be reviewed in Section 3.

2.4. The rate of convergence of CM and ECM

To apply the logic of the SEM procedure to ECM, we need to relate the rate of convergence of ECM to the fraction of missing information. Meng (1994) extended (2.3.1) to the ECM case with the following result:

$$[I_d - DM^{ECM}] = [I_d - DM^{EM}][I_d - DM^{CM}], \tag{2.4.1}$$

where DM^{ECM} is the rate of convergence of ECM at $\theta = \theta^\star$, and DM^{CM} is the rate of convergence of CM at $\theta = \theta^\star$. If we knew DM^{ECM} and DM^{CM}, we could use (2.4.1) and (2.3.4) to calculate the asymptotic variance, $[I_o(\theta^\star|Y_{obs})]^{-1}$.

As will be detailed in Section 3, DM^{ECM} can be computed by numerical differentiation just like DM^{EM}. The rate of convergence of CM, namely DM^{CM}, can be computed in two ways. Meng (1994) shows that it can be calculated analytically as

$$DM^{CM} = P_1 \cdots P_S, \tag{2.4.2}$$

where

$$P_s = \nabla_s[\nabla_s^\top I_{oc}^{-1}\nabla_s]^{-1}\nabla_s^\top I_{oc}^{-1}, \qquad s = 1, \ldots, S, \tag{2.4.3}$$

with I_{oc} given in (2.3.3) and $\nabla_s = \nabla g_s(\theta^\star)$ the gradient of the constraint function $g_s(\theta)$ evaluated at $\theta = \theta^\star$. Notice that all the quantities in (2.4.2) involve only the complete-data information matrix and the g functions, and thus they can be computed once $\theta^\star$ is obtained.

Alternatively, when the complete-data model $f(Y|\theta)$ is from an exponential family, DM^{CM} can be obtained numerically from the output of ECM at convergence, $(\theta^\star, S^\star(Y_{obs}))$, and an additional run of the code for the CM steps. If we take $S^\star(Y_{obs})$ to be the fixed complete-data sufficient statistics, we can obtain $\hat{\theta}(S^\star(Y_{obs}))$, the MLE of θ given $S^\star(Y_{obs})$, using the CM algorithm starting from $\theta^{(0)} \neq \theta^\star$; DM^{CM} is the rate of convergence of this CM algorithm. This can be proved by examining (2.4.3) and noting that if $f(Y|\theta)$ is from an exponential family, $I_{oc} = I_o(\theta^\star|S^\star(Y_{obs}))$ and that $\theta^\star = \hat{\theta}(S^\star(Y_{obs}))$. Thus we can derive DM^{CM} by calculating the rate of convergence of the CM algorithm applied to

$L(\theta|S^\star(Y_{\rm obs}))$. This avoids the matrix inversions and computation of the ∇_s in (2.4.2) and (2.4.3), which are necessary when performing analytical calculations.

With the PECM algorithm described in Section 2.2, the computation of DM^{CM} is particularly easy. Let Υ be the block diagonal matrix of $I_{\rm oc}$ with S blocks corresponding to the S subvectors of θ defined by the partition. Let Γ be the corresponding lower block triangular matrix of $I_{\rm oc}$, that is, $I_{\rm oc} = \Upsilon + \Gamma + \Gamma^\top$. Meng (1990) established that in this case (2.4.2) reduces to

$$DM^{CM} = -\Gamma[\Upsilon + \Gamma^\top]^{-1}, \tag{2.4.4}$$

which makes analytical calculation of DM^{CM} very simple, as illustrated in Section 4.1.

2.5. The basic identity for the SECM algorithm

Having obtained DM^{ECM}, DM^{CM}, and $I_{\rm oc}$, we can combine (2.3.4) and (2.4.1) to obtain

$$I_{\rm o}(\theta^\star|Y_{\rm obs}) = [I_d - DM^{ECM}][I_d - DM^{CM}]^{-1} I_{\rm oc}. \tag{2.5.1}$$

Equivalently, in terms of the variance,

$$V_{\rm obs} \equiv I_{\rm o}^{-1}(\theta^\star|Y_{\rm obs}) = V_{\rm com} + \Delta V, \tag{2.5.2}$$

where $V_{\rm com} = I_{\rm oc}^{-1}$ can be viewed as the variance-covariance matrix of the MLE given the complete-data, and

$$\Delta V = V_{\rm com}[DM^{ECM} - DM^{CM}][I_d - DM^{ECM}]^{-1} \tag{2.5.3}$$

is the increase in variance due to the missing data.

Identity (2.5.3) is the basis for the SECM algorithm, and it reduces to (2.3.6) of Meng and Rubin (1991a) when $DM^{CM} = 0$, which corresponds to EM. An interesting property of SECM (and SEM) is that, although ΔV is mathematically symmetric, the right side of (2.5.3) is not numerically constrained to be symmetric because $V_{\rm com}$, DM^{CM}, and DM^{ECM} are computed separately as described in Section 3. Numerical symmetry is obtained only when all three of these are computed without appreciable numerical imprecision. This property turns out to be a surprisingly powerful tool for detecting implementation or numerical errors, as illustrated in Section 4 and further discussed in Section 5.

3. Implementing the SECM Algorithm

3.1. A schematic

This section is designed to explain how to implement SECM in a step by step manner. Readers not interested in implementational details may wish to skip to the examples in Section 4. We will describe in simple terms exactly how one can compute $\theta^\star$ and $V_{\rm obs}$. The schematic in Figure 1 describes the necessary steps in broad terms. The user must provide routines that perform the E and CM steps, as well as one that computes $I_{\rm oc}$. These are described in Section 3.2. The schematic also references `Algorithms 1, 2` and `3`, which compute $\theta^\star$, DM^{ECM} and DM^{CM} respectively and are described in Section 3.3. The mathematical background for the routines that follow is given in Section 2 of this paper and in Meng and Rubin's (1991a) presentation of the SEM algorithm. Since SEM is a special case of SECM, the algorithms presented here can also be used to implement SEM. The only modification when running SEM is that the `CMSTEPS` routine should compute the global maximum of $Q(\theta|\theta^{(t)})$, that is, use only one CM step, and the DM^{CM} matrix should be set to 0.

3.2. User provided specific subroutines

The computations in the following three subroutines are problem specific; the first two routines are used in the Algorithms in Section 3.3 and the third is used in box 3 of Figure 1. These subroutines are developed assuming that $f(Y|\theta)$ is from an exponential family, beyond which the simplicity of EM-type algorithms is typically lost because the E step typically requires numerical integration or simulations (c.f. Wei and Tanner (1990)).

`Subroutine 1. ESTEP:`

INPUT: $\theta^{(t)}$, $Y_{\rm obs}$

Compute $S^{(t)}(Y_{\rm obs}) = {\rm E}[S(Y)|\theta^{(t)}, Y_{\rm obs}]$, where $S(Y)$ is the complete-data sufficient statistic.

OUTPUT: $S^{(t)}(Y_{\rm obs})$

`Subroutine 2. CMSTEPS:`

INPUT: $S^{(t)}(Y_{\rm obs})$, $\theta^{(t)}$

Compute $\theta^{(t+1)}$ with a sequence of constrained maximization steps, as described in Section 2.2.

OUTPUT: $\theta^{(t+1)}$

`Subroutine 3. ICOM:`

INPUT: $\theta^\star$, $S^\star(Y_{\rm obs})$

Compute $I_{\rm oc} = I_{\rm o}(\theta^\star|S^\star(Y_{\rm obs}))$, the observed Fisher information matrix based on the complete-data model, evaluated at $\theta^\star$ and $S^\star(Y_{\rm obs})$.

OUTPUT: $I_{\rm oc}$ and $V_{\rm com} = I_{\rm oc}^{-1}$

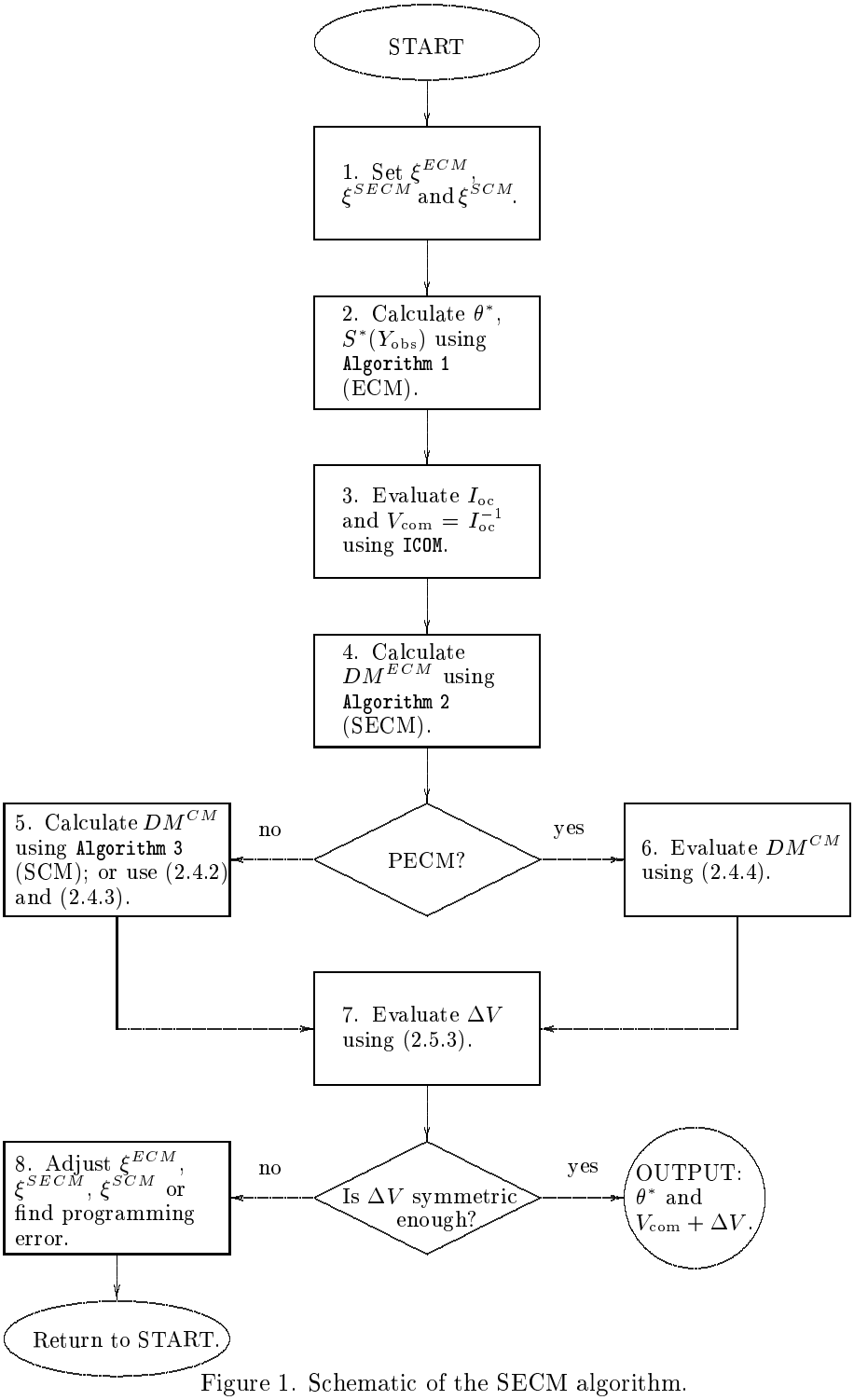

Figure 1. Schematic of the SECM algorithm.

3.3. General algorithms

`Algorithm 1`: Calculate $\theta^\star$ and $S^\star(Y_{\rm obs})$ using ECM.
Repeat the ECM steps:
INPUT: $\theta^{(t)}$

Step 1: Calculate $S^{(t)}(Y_{\rm obs})$ with `ESTEP`;

Step 2: Calculate $\theta^{(t+1)}$ with `CMSTEPS`;

OUTPUT: $\theta^{(t+1)}$

Continue until

$$\delta(\theta^{(t)}, \theta^{(t+1)}) < \xi^{ECM} \tag{3.3.1}$$

for some convergence criterion δ and threshold ξ^{ECM}. A discussion on how to choose δ and ξ^{ECM}, as well as ξ^{SECM} and ξ^{SCM} appears in Section 3.4.
FINAL OUTPUT : Set $S^\star(Y_{\rm obs})$ equal to the output from the final `ESTEP`, and set $\theta^\star$ equal to the output from the final `CMSTEPS`.

`Algorithm 2`: Calculate DM^{ECM} using SECM.
Let r_{ij} be the (i,j)th element of the $d \times d$ matrix DM^{ECM} and define $\theta^{(t)}(i)$ as

$$\theta^{(t)}(i) = (\theta_1^\star, \ldots, \theta_{i-1}^\star, \theta_i^{(t)}, \theta_{i+1}^\star, \ldots, \theta_d^\star), \qquad i = 1, \ldots, d. \tag{3.3.2}$$

That is, $\theta^{(t)}(i)$ is $\theta^\star$ with the ith component active, i.e. replaced by the ith component of $\theta^{(t)}$.

Repeat the SECM steps:
INPUT: $\theta^\star$ and $\theta^{(t)}$
Repeat Step 1 and Step 2 for each i

Step 1: Calculate $\theta^{(t)}(i)$ from (3.3.2), treat it as input for `Algorithm 1`, and run one iteration of `Algorithm 1` (that is, one `ESTEP` and one `CMSTEPS`) to obtain $\tilde{\theta}^{(t+1)}(i)$;

Step 2: Obtain the ratio

$$r_{ij}^{(t)} = \frac{\tilde{\theta}_j^{(t+1)}(i) - \theta_j^\star}{\theta_i^{(t)} - \theta_i^\star} \quad \text{for} \quad j = 1, \ldots, d; \tag{3.3.3}$$

OUTPUT: $\{r_{ij}^{(t)}, i, j = 1, \ldots, d\}$.
FINAL OUTPUT: $DM^{ECM} = \{r_{ij}^\star\}$, where $r_{ij}^\star = r_{ij}^{(t_{ij})}$ is such that

$$\delta(r_{ij}^{(t_{ij})}, r_{ij}^{(t_{ij}+1)}) < \xi^{SECM} \tag{3.3.4}$$

for some suitable convergence threshold ξ^{SECM}.

`Algorithm 3`: Calculate DM^{CM} using SCM.

For notational simplicity, the same notation is used for the elements of DM^{CM} as was used for DM^{ECM}.

Repeat the SCM (e.g. supplemented CM) steps:
INPUT: $\theta^\star$, $\theta^{(t)}$ and $S^\star(Y_{\rm obs})$
Repeat Step 1 and Step 2 for each i

Step 1: Calculate $\theta^{(t)}(i)$ from (3.3.2) and run `CMSTEPS` once using $S^\star(Y_{\rm obs})$ and $\theta^{(t)}(i)$ as input to obtain $\tilde{\theta}^{(t+1)}(i)$;

Step 2: Obtain the ratio $r_{ij}^{(t)}$ as in (3.3.3);

OUTPUT: $\{r_{ij}^{(t)}, i,j = 1,\ldots,d\}$.
FINAL OUTPUT: $DM^{CM} = \{r_{ij}^\star\}$ where all the $r_{ij}^\star = r_{ij}^{t_{ij}}$ are such that (3.3.4) is satisfied for some convergence threshold for SCM, ξ^{SCM}.

When implementing the PECM algorithm, `Algorithm 3` can be replaced by a simple evaluation of (2.4.4). For the more general ECM algorithm, it can be computationally advantageous to replace `Algorithm 3` with analytical calculations described in (2.4.2) and (2.4.3). Finally, the outputs of `ICOM`, and `Algorithms 2` and `3` (i.e. $V_{\rm com}$, DM^{ECM}, DM^{CM}) are put together to calculate ΔV using (2.5.3), and then (2.5.2) is used to obtain the desired variance-covariance matrix $V_{\rm obs}$.

3.4. Notes on implementation

The convergence criterion $\delta(a,b)$ is a discrepancy measure between a and b. Common choices are (i) $\delta(a,b) = \max_i |a_i - b_i|$, (ii) $\delta(a,b) = \max_i |(a_i - b_i)/a_i|$ or $\max_i |(a_i - b_i)/(a_i + b_i)|$, and (iii) $\delta(a,b) = ||a - b||$, where $||\cdot||$ denotes the standard Euclidean norm, and a_i and b_i are the components of a and b. The first of these, (i) was used in the examples in Section 4 and is generally fine unless the magnitudes of the components vary greatly, in which case (ii) is preferred. The same holds for SECM and SCM except that a and b are scalars, in which case (i) and (iii) are the same.

When ECM is run alone, the convergence threshold ξ^{ECM} can be set to obtain whatever level of precision is desired for $\theta^\star$. When SECM is used, however, ξ^{ECM} must be quite small (compared to the magnitude of $\theta^\star$) to insure convergence of $\theta^{(t)}$ as well as $r_{ij}^{(t)}$ and thereby to insure satisfactory symmetry in ΔV. Generally ξ^{SECM} and ξ^{SCM} should be about equal to the square root of ξ^{ECM}, as is discussed further in Section 5.1.

Finally, note that `Algorithms 2` and `3` assume that the ECM iterates were saved when `Algorithm 1` was run. This saves computational time, but requires extra storage. For some users, it may be better to recompute the iterates than to save them. To do this, change the input in `Algorithms 2` and `3` to "*INPUT:* $\theta^\star$ and $\theta^{(t-1)}$" and add "*Step 0:* Run one ECM iteration on $\theta^{(t-1)}$ to obtain $\theta^{(t)}$."

Generally, it is not necessary to start the SECM or SCM algorithms at $\theta^{(0)}$ or to run them for as many steps as ECM was run. Thus, saving all the iterates may not be economical, and it may be computationally more efficient to recompute only the iterates that are needed.

4. Examples illustrating SECM

4.1. Bivariate normal stochastic censoring model

Suppose $(y_{i1}, y_{i2})^\top$ are independent observations from a bivariate normal distribution, where y_2 is never observed and y_1 is observed only if $y_2 > 0$. For each unit, the density is specified by

$$\begin{pmatrix} y_{i1} \\ y_{i2} \end{pmatrix} \overset{\text{indep}}{\sim} N\left[\begin{pmatrix} \beta_{11}x_{i1} + \beta_{12}x_{i2} + 0 \cdot x_{i3} \\ \beta_{21}x_{i1} + 0 \cdot x_{i2} + \beta_{23}x_{i3} \end{pmatrix}, \begin{pmatrix} \sigma_1^2 & \rho\sigma_1 \\ \rho\sigma_1 & 1 \end{pmatrix} \right], \quad i = 1, \ldots, n.$$

Here the x_{ij} are completely observed, and we set $\beta = (\beta_{11}, \beta_{12}, \beta_{21}, \beta_{23})$ and $\Sigma = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1 \\ \rho\sigma_1 & 1 \end{pmatrix}$.

This is an example of the so called seemingly unrelated regression model (Zellner (1962)), also known in economics as the Type II Tobit model (e.g., Amemiya (1984)). When the active covariates for y_{i1} and y_{i2} overlap but are not identical (in our example, $\beta_{13} = \beta_{22} = 0$), even if all the y's are observed, the MLEs of β and Σ are not in closed form. Consequently, implementing EM would require nested iterations within the M step. However, given β, the conditional MLE of Σ is simply the sum of squares of the residuals divided by n. On the other hand, given Σ, the conditional MLE for β can be easily obtained by weighted least squares. ECM replaces the iterative M step with these two CM steps (detailed formulas are given in Meng and Rubin (1995)).

To compute the E step, we need to find the conditional expectation of $(y_{i1}, y_{i1}^2, y_{i2}, y_{i2}^2, y_{i1}y_{i2})$ for $i = 1, 2, \ldots, n$, given the observed data and the parameters. These calculations follow from the properties of the bivariate normal distribution and are given explicitly in Little and Rubin (1987), p. 225. There is, however, an error in that presentation. When $y_{i2} > 0$, we also observe y_{i1}, and must thus find $E(y_{i2}|y_{i1}, y_{i2} > 0)$ and $E(y_{i2}^2|y_{i1}, y_{i2} > 0)$, not $E(y_{i2}|y_{i2} > 0)$ and $E(y_{i2}^2|y_{i2} > 0)$ as presented there. This error will lead to incorrect results, in particular it tends to underestimate the magnitude of ρ. For the data described below, the true $\rho = 0.5$, the MLE $\rho^\star = 0.482$, but the incorrect procedure gives 0.200. We discovered this error only after we found that the resulting variance-covariance matrix from SECM was clearly asymmetric, which demonstrates the power of SECM as a tool for detecting errors in implementing ECM. To correct the E step, we need to substitute the following two expectations for the second

and fifth equations given in Little and Rubin (1987), p. 225,

$$\mathrm{E}(y_{i2}|y_{i1}, y_{i2} > 0, \beta^{(t)}, \Sigma^{(t)}) = \eta_{i2}^{(t)} + \tau_{i2}^{(t)} \lambda\left(\frac{\eta_{i2}^{(t)}}{\tau_{i2}^{(t)}}\right),$$

$$\mathrm{E}(y_{i2}^2|y_{i1}, y_{i2} > 0, \beta^{(t)}, \Sigma^{(t)}) = [\tau_{i2}^{(t)}]^2 + [\eta_{i2}^{(t)}]^2 + \tau_{i2}^{(t)} \eta_{i2}^{(t)} \lambda\left(\frac{\eta_{i2}^{(t)}}{\tau_{i2}^{(t)}}\right),$$

where $\lambda(z) = \phi(z)/\Phi(z)$ is the inverse Mill's ratio, and

$$\eta_{i2} = E(y_{i2}|y_{i1}, \beta, \Sigma) = \mu_2 + \rho\frac{(y_{i1} - \mu_1)}{\sigma_1}, \quad \tau_{i2} = \sqrt{\mathrm{Var}(y_{i2}|y_{i1}, \beta, \Sigma)} = \sqrt{1 - \rho^2}.$$

We ran SECM using the variance stabilizing transformations $(\log(\sigma_1), Z_\rho)$ in place of (σ_1, ρ), where $Z_\rho = 0.5 \log\{(1+\rho)/(1-\rho)\}$ is the Fisher Z transformation of ρ. This transformation is used, not only to ensure better normality when invoking large sample approximations, but also to enhance the computational stability of SECM since $DM^{ECM}(\theta)$ is more nearly constant (as a matrix function of θ) near $\theta^\star$ when the loglikelihood is more nearly quadratic. The sample data set of size 12 in Table 1 was simulated using the parameters in the first row of Table 2. The observed data are the 8 observations of Y_1 for which the corresponding observation of Y_2 is positive. None of the values of Y_2 are observed.

Table 1. The data for Example 4.1

x_1	x_2	x_3	Y_1	Y_2
−1	1	−1	−0.4443346	−2.9841022
−1	1	−1	−0.4038098	−0.9029128
−1	−1	−1	−0.4457312	−0.1776825
−1	−1	0	−0.1966688	0.4006104
0	1	0	0.5583971	0.3723503
0	−1	0	−0.7892194	1.1994856
0	1	2	−0.2868998	−0.5555625
0	−1	2	−0.4309087	0.7991658
1	1	2	1.2447119	1.4188357
1	1	3	1.3696260	2.1091285
1	−1	3	−0.4198308	0.0973109
1	−1	3	−0.3999554	1.1703623

ECM was run with $\xi^{ECM} = 10^{-12}$, resulting in 249 iterations, from a starting value of all zeros. Table 2 contains the MLEs of β, ρ and σ_1 in the second row

and the corresponding asymptotic variances, which were found using SECM as described below, in the third row.

Since the complete-data distribution is from a standard exponential family, $I_{\rm oc}$ is just the complete-data information matrix evaluated at $\theta^\star$ and $S^\star(Y_{\rm obs})$, which yields

$$V_{\rm com} = \begin{array}{c} \\ \beta_{11} \\ \beta_{12} \\ \beta_{21} \\ \beta_{23} \\ \\ \log(\sigma_1) \\ Z_\rho \end{array} \begin{array}{c} \begin{array}{cccccccc} \beta_{11} & \beta_{12} & \beta_{21} & \beta_{23} & & \log(\sigma_1) & Z_\rho \end{array} \\ \begin{pmatrix} 0.013512 & 0.000000 & 0.001210 & 0.007259 & \vdots & -0.002169 & -0.008234 \\ 0.000000 & 0.009502 & 0.013557 & 0.000000 & \vdots & 0.000000 & 0.000000 \\ 0.001210 & 0.013557 & 0.083980 & 0.003880 & \vdots & -0.000009 & -0.002012 \\ 0.007259 & 0.000000 & 0.003880 & 0.023278 & \vdots & -0.000051 & -0.012073 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ -0.002169 & 0.000000 & -0.000009 & -0.000051 & \vdots & 0.039834 & 0.016296 \\ -0.008234 & 0.000000 & -0.002012 & -0.012073 & \vdots & 0.016296 & 0.081700 \end{pmatrix} \end{array}.$$

Since this is a PECM algorithm, using (2.4.4) we can quickly calculate DM^{CM} from $I_{\rm oc}$. In applying (2.4.4), Υ is the block diagonal matrix indicated in the above display, and the non zero portion of Γ is the lower left 2×4 submatrix of $V_{\rm com}$. SECM was run with $\xi^{SECM} = 10^{-7}$, and $V_{\rm obs}$ is found by a simple application of (2.5.2):

$$V_{\rm obs} = \begin{array}{c} \\ \beta_{11} \\ \beta_{12} \\ \beta_{21} \\ \beta_{23} \\ \log(\sigma_1) \\ Z_\rho \end{array} \begin{array}{c} \begin{array}{cccccc} \beta_{11} & \beta_{12} & \beta_{21} & \beta_{23} & \log(\sigma_1) & Z_\rho \end{array} \\ \begin{pmatrix} 0.022862 & -0.002858 & -0.001767 & 0.007392 & -0.000422 & -0.004376 \\ -0.002858 & 0.014679 & 0.008360 & 0.002188 & -0.002109 & -0.015259 \\ -0.001765 & 0.008360 & 0.260939 & -0.081708 & 0.005719 & 0.115485 \\ 0.007390 & 0.002187 & -0.081704 & 0.161360 & -0.003444 & -0.143232 \\ -0.000421 & -0.002109 & 0.005719 & -0.003445 & 0.062933 & 0.029605 \\ -0.004369 & -0.015253 & 0.115469 & -0.143233 & 0.029602 & 0.494789 \end{pmatrix} \end{array}.$$

We see here the symmetry holds to at least 4 decimal places, indicating accurate computation. Comparing $V_{\rm com}$ and $V_{\rm obs}$, we can also easily find the increase in variance due to missing data, as recorded in the fourth row of Table 2 (we have applied a Jacobian transformation for the variances of $\sigma_1^\star$ and $\rho^\star$).

Table 2. The results from Example 4.1

	β_{11}	β_{12}	β_{21}	β_{23}	σ_1	ρ
θ_{true}	0.2000	0.5000	−0.3000	0.3000	0.5000	0.5000
$\theta^\star$	0.2643	0.6248	−0.5263	0.5274	0.3377	0.4818
$\text{Var}(\theta^\star)$	0.0229	0.0147	0.2609	0.1614	0.0072	0.2918
ΔV	0.0094	0.0052	0.1769	0.1381	0.0027	0.2435

4.2. A 2×2×2 contingency table

Table 3(a) presents a $2 \times 2 \times 2$ contingency table on infant survival (Bishop, Fienberg and Holland (1975), Table 2.4-2). The supplementary data in Table 3(b) was added by Little and Rubin (1987, p. 187) to form a partially classified table. Suppose we wish to fit a log-linear model with no three way interaction:

$$\begin{aligned}\log(\theta_{ijk}) = u_0 &+ (-1)^{i-1}u_P + (-1)^{j-1}u_C + (-1)^{k-1}u_S \\ &+ (-1)^{i+j}u_{PC} + (-1)^{j+k}u_{CS} + (-1)^{i+k}u_{PS}, \qquad (4.2.1)\end{aligned}$$

where θ_{ijk} is the cell probability for cell (i, j, k) for $i, j, k = 1, 2$, with i corresponding to P, j to C and k to S. We will derive the MLE of $U = (u_0, u_P, \ldots, u_{PS})^\top$ and V_{obs}, the asymptotic variance matrix of $U^\star$.

Table 3. A $2 \times 2 \times 2$ contingency table with partially classified observations

		Survival (S)		
Clinic (C)	Prenatal care (P)	Died	Suvivied	
(a) Completely classified cases				
A	Less	3	176	
	More	4	293	
B	Less	17	197	
	More	2	23	$N^{(a)} = 715$ cases
(b) Partially classified cases				
?	Less	10	150	
	More	5	90	$N^{(b)} = 255$ cases

Meng and Rubin (1991b, 1993) describe an ECM algorithm with three CM steps for this problem. Specifically, since the loglikelihood is linear in the cell

counts, $Y = \{y_{ijk}\}$, the E step simply involves imputing the missing data,

$$E : y_{ijk}^{(t)} = \tilde{y}_{ijk}^{(a)} + \tilde{y}_{ik}^{(b)} \frac{\theta_{ijk}^{(t)}}{\sum_j \theta_{ijk}^{(t)}}, \tag{4.2.2}$$

where $\tilde{y}_{ijk}^{(a)}$ are the cell counts in Table 3(a), and $\tilde{y}_{ik}^{(b)}$ are the marginal counts classified only according to parental care and survival (see Table 3(b)). The CM steps make use of IPF. Given the current estimated cell probabilities $\{\theta_{ijk}^{(t)}\}$, the three CM steps are

$$CM_1 : \theta_{ijk}^{(t+\frac{1}{3})} = \theta_{ij(k)}^{(t)} \frac{y_{ij+}}{N},\ CM_2 : \theta_{ijk}^{(t+\frac{2}{3})} = \theta_{i(j)k}^{(t+\frac{1}{3})} \frac{y_{i+k}}{N},\ CM_3 : \theta_{ijk}^{(t+\frac{3}{3})} = \theta_{(i)jk}^{(t+\frac{2}{3})} \frac{y_{+jk}}{N},$$

where N is the total count, $\theta_{ij(k)} = \theta_{ijk} / \sum_k \theta_{ijk}$ is the conditional probability of the third factor given the first two, and $y_{ij+} = \sum_k y_{ijk}$, etc. It is easy to see that CM_1 maximizes $L(\theta|Y)$ subject to $\theta_{ij(1)} = \theta_{ij(1)}^{(t)}$ for each i and j, so that the constraint function $g_1(\theta) = \{\theta_{ij(1)}\}$. Likewise $g_2(\theta) = \{\theta_{i(1)k}\}$ and $g_3(\theta) = \{\theta_{(1)jk}\}$. It is clear that this is not a PECM algorithm.

We start ECM with $\theta_{ijk} = 1/8$ for each i, j and k, which satisfies the constraint of no three way interaction, and cycle according to ECM : $E \to CM_1 \to CM_2 \to CM_3$. At each iteration, $U^{(t)}$ can then be calculated by regression

$$U^{(t)} = (X^\top X)^{-1} X^\top \log \theta^{(t)}, \quad t = 1, 2, 3, \ldots, \tag{4.2.3}$$

where the design matrix X is derived from (4.2.1) with elements either $+1$ or -1. Note that we are defining two mappings, $M_\theta : \theta^{(t)} \to \theta^{(t+1)}$ and $M_U : U^{(t)} \to U^{(t+1)}$. The θ parameterization is more natural in the context of the E and CM steps. The U parameterization is more convenient in the context of the log-linear model, and is a stable parameterization for the SECM calculations.

To compute $V_{\rm obs}$ on the U scale we need to derive DM^{ECM}, DM^{CM}, and $V_{\rm com} \equiv V(U^\star|Y^\star)$, where $Y^\star = E(Y|\theta^\star, Y_{\rm obs})$. Implementing `Algorithm 2` on the mapping induced by M_U will yield DM^{ECM}. Since this is not a PECM algorithm, we cannot use (2.4.4) to derive DM^{CM}. Instead, SCM in `Algorithm 3` is used, with $Y^\star$ being the complete data. That is, we use the standard IPF procedure, which induces the mapping given by $CM_1 \to CM_2 \to CM_3$, to bit (4.2.1) to $Y^\star$; `Algorithm 3` differentiates this mapping and produces DM^{CM}. Finally, we compute $V_{\rm com}$ via a log-linear models package. Since all standard programs use the sufficient statistics as their input, and $I_{\rm oc}$ is linear in the complete-data sufficient statistics, fitting (4.2.1) using $Y^\star$ as the data will yield $V_{\rm com}$. For example, in *'S'* (AT&T Bell Laboratories) $V_{\rm com}$ can be computed with

```
> model<-glm(formula=Y*~ P+C+S+PC+PS+CS, family=poisson(link=log))
> summary(model)$cov.unscaled
```

where `P`, `C`, `S`, etc. are vectors of $+1$ and -1 determined by (4.2.1) and are the columns of the design matrix X. The parameter u_0 in (4.2.1) is just a scale parameter that insures that $\sum \theta_{ijk} = 1$ and it should be ignored in the calculation of V_{obs} as there are only six free parameters. Replace DM^{ECM}, DM^{CM} and V_{com} with the 6×6 submatrices corresponding to the other six parameters before computing V_{obs} using (2.5.2).

The results are presented in Table 4. The calculated matrix V_{obs} was symmetric to nine places beyond the decimal (due to space limitations, only five are shown in Table 4), which is more accurate than expected since the algorithm was run with $\xi^{ECM} = 10^{-16}$, $\xi^{SECM} = 10^{-8}$ and $\xi^{SCM} = 10^{-7}$. The ECM algorithm required 70 iterations to converge.

Table 4. The MLE $U^\star$ and V_{obs} for Model (4.2.1) with data given in Table 3

	u_P	u_S	u_C	u_{PS}	u_{CS}	u_{PC}
$U^\star$†	0.40694	−1.56568	0.18153	−0.04442	−0.42478	−0.66167
sd($U^\star$)	0.11761	0.09274	0.13516	0.11749	0.13267	0.05847
	0.01383	−0.00182	0.01026	0.01225	0.00902	−0.00119
	−0.00182	0.00860	0.00312	−0.00232	0.00247	−0.00034
V_{obs}	0.01026	0.00312	0.01827	0.00867	0.01618	−0.00160
	0.01225	−0.00232	0.00867	0.01380	0.00988	0.00115
	0.00902	0.00247	0.01618	0.00988	0.01760	0.00053
	−0.00119	−0.00034	−0.00160	0.00115	0.00053	0.00342

† $u_0^\star = -3.32944$

Table 5. 95% marginal intervals for $\theta^\star$ for Model (4.2.1) with data given in Table 3

		Survival (S)		
Clinic (C)	Prenatal care (P)	Died	Suvivied	
A	Less	[0.0016, 0.0115]	[0.2244, 0.2876]	
	More	[0.0040, 0.0156]	[0.3566, 0.4200]	
B	Less	[0.0180, 0.0391]	[0.2531, 0.3181]	
	More	[0.0013, 0.0090]	[0.0205, 0.0457]	$N = 970$ cases

The information in Table 4 can be used to construct confidence intervals. For example, we can derive the Jacobian J of the transformation form logit(θ) to U in order to calculate the observed Fisher information matrix for logit($\theta^\star$): $I_{\text{o}}(\text{logit}(\theta^\star)|Y_{\text{obs}}) = J^\top I_{\text{o}}(U^\star|Y_{\text{obs}})J$. Assuming approximate normality on the logit(θ) scale, we can construct confidence intervals for each of logit(θ_{ijk}), and

then transform to the θ scale. Table 5 is an illustration.

The calculations presented in the context of this example, are in fact quite general. Bishop, Fienberg and Holland (1975) describe how either IPF or closed form solutions can be used to fit any hierarchical log-linear model to contingency tables with complete data. This means that the CM steps can easily be identified for any such model. Since the E step in (4.2.2) can easily be generalized to any table with incomplete data, the SECM algorithm for fitting a log-linear model to any table can easily be formulated. These calculations are described more fully by van Dyk (1993).

5. Diagnostics and Variations

5.1. Checking the symmetry of V_{obs}

One of the most valuable properties of SECM is that like SEM, it has a built in diagnostic. Section 3 describes all the steps required by SECM. It is the last step, computation of ΔV that helps us know if mistakes have been made in any of these steps. The variance matrix, $V_{\text{obs}} = V_{\text{com}} + \Delta V$, and thus ΔV, must be symmetric, but if any of $\theta^\star$, DM^{ECM}, DM^{CM}, or I_{oc} are not calculated correctly, it is practically certain that the resulting ΔV and hence V_{obs} will be asymmetric. The example in Section 4.1 documents that this diagnostic not only checks the computation of V_{obs} but also detects errors in implementing the E and CM steps. Convergence of the $\theta^{(t)}$ sequence does not insure that the convergent value is the MLE, $\theta^\star$. Many erroneous algorithms converge. In fact, we had an instance in which our algorithm increased the likelihood at each step and converged, but the resulting V_{obs} was asymmetric. In this case, careful checking led to the discovery of some subtle errors in implementation. There is no other diagnostic known to us that can automatically detect these errors, and one perhaps would never find them without the detection power of this tool. If V_{obs} is symmetric, however, we are virtually assured that both $\theta^\star$ and V_{obs} are correct because it seems practically impossible to make separate errors in SECM that cancel appropriately.

Even when SECM is implemented correctly, the convergence threshold ξ^{ECM} needs to be quite small in order to obtain a V_{obs} matrix with satisfactory symmetry; this implies an increase in the number of iterations required, especially when θ is of high dimension. The more precisely we calculate $\theta^\star$, the more accurately we are able to compute V_{obs}, because we are able to compute DM^{ECM} and DM^{CM} more accurately. An example demonstrating this is given in Section 4.5 of van Dyk, Meng and Rubin (1994). In general ξ^{SECM} and ξ^{SCM} are chosen to be about equal to the square root of ξ^{ECM}. They should be chosen as small as possible, however, so that (3.3.4) is satisfied for some t_{ij} for each i and j. In order to increase the accuracy of DM^{ECM} and DM^{CM}, ξ^{SECM} and ξ^{SCM} may even be different for different components. When deciding on convergence thresholds, a

good rule of thumb is that V_{obs} will be symmetric to about half as many digits as $\theta^\star$ is precise; for example, roughly, when ξ^{ECM} is 10^{-8}, we can expect 3 or 4 digits of accuracy in V_{obs}. The accuracy of V_{obs} can always be judged by its symmetry, however, and gross asymmetry always indicates either errors in implementation or numerical imprecision; also see Section 5 of Meng and Rubin (1991a).

5.2. Computing V_{obs} when implementing MCECM

The multi-cycle ECM or MCECM algorithm (Meng and Rubin (1993)) is a variation of the ECM algorithm in which extra E steps are added to each iteration, in the hope that adding E steps will speed the convergence. Consider, for example, the three CM-step ECM algorithm : $E \to CM_1 \to CM_2 \to CM_3$. The MCECM algorithm adds one or more E steps to each iteration, for example,

$$\text{MCECM}: \quad E \to CM_1 \to E \to CM_2 \to E \to CM_3. \tag{5.2.1}$$

Like ECM, this algorithm increases $L(\theta|Y_{\text{obs}})$ at each iteration and converges to $\theta^\star$ under the same conditions that guarantee that ECM does. The rate of convergence of MCECM is, however, more complicated than that of ECM (see Meng (1994)). Consequently, direct implementation of the supplemented MCECM algorithm would be quite involved. There is, however, an easy solution. In `Algorithms 2 and 3`, we evaluate the ratio r_{ij} using the ECM iterates in order to calculate DM^{ECM} and DM^{CM}. But there is nothing that requires the use of the ECM *iterates* in these algorithms, just the ECM code, and we can actually use any sequence $\theta^{(t)}$ converging to $\theta^\star$. In particular, we can use the MCECM iterates as input for these algorithms and compute V_{obs} just as described in Section 3. That is, the sequence of steps in (5.2.1) can be used when we calculate $\theta^\star$ in `Algorithm 1`, but the standard ECM algorithm should be used in `Algorithms 2` and `3` in which `ESTEP` will consist of one E step, and `CMSTEPS` will consists of the three CM steps. If the MCECM iterations were not saved, we can simply run the regular ECM algorithm (i.e., drop the added E steps) when implementing the supplemented algorithm to calculate V_{obs}. Dropping the added E step in this round will not slow the convergence, since we can start at initial values that are closer to $\theta^\star$ than the original $\theta^{(0)}$.

5.3. When some components have no missing information

In certain situations, missing data only affect estimates for some components of θ, that is, there is no missing information for the rest of θ. For example, with $\theta = (\vartheta_1, \vartheta_2)$, there might be no missing data for the estimate of ϑ_1, in which case we can compute the MLE $\vartheta_1^\star$ without using EM or ECM; see example 4.4 of Meng and Rubin (1991a). An efficient implementation of ECM in such cases fixes

$\vartheta_1 = \vartheta_1^\star$, and only updates $\vartheta_2^{(t)}$. Since this implementation of ECM conditions on $\vartheta_1 = \vartheta_1^\star$, the corresponding SECM algorithm can be used to calculate $\Delta V(\vartheta_2^\star|\vartheta_1^\star)$, the increase in asymptotic conditional variance of $\vartheta_2^\star$ (conditioning on $\vartheta_1^\star$) due to missing information. Specifically, we can compute $\Delta V(\vartheta_2^\star|\vartheta_1^\star)$ (see (2.5.3)) as

$$\Delta V(\vartheta_2^\star|\vartheta_1^\star) = \{[I_{\rm oc}]_{22}\}^{-1}\,[DM^{ECM} - DM^{CM}][I_{d_2} - DM^{ECM}]^{-1}, \qquad (5.3.1)$$

where the ECM and CM algorithms are run with ϑ_1 fixed at $\vartheta_1^\star$, $[I_{\rm oc}]_{22}$ is the submatrix of $I_{\rm o}(\theta^\star|Y_{\rm obs})$ corresponding to ϑ_2, and d_2 is the dimension of ϑ_2. It turns out that (5.3.1) is all we need to compute the increase in variance, since

$$\Delta V = \begin{pmatrix} 0 & 0 \\ 0 & \Delta V(\vartheta_2^\star|\vartheta_1^\star) \end{pmatrix}. \qquad (5.3.2)$$

This identity holds because (i) there is no increase in variance or covariance of $\vartheta_1^\star$, and (ii) there is no increase in the part of the variance of $\vartheta_2^\star$ that can be explained by $\vartheta_1^\star$ (see Meng and Rubin (1991a)). When there is no missing information for ϑ_1, we can therefore calculate ΔV using (5.3.2) and then calculate $V_{\rm obs}$ using (2.5.2). It is, however, worth remarking that fixing ϑ_1 at $\vartheta_1^\star$ increases the efficiency of ECM and SECM, but is not a required step since the standard ECM and SECM algorithms will produce the desired estimates. This is in contrast to the standard SEM algorithm, which fails in this situation because some of the denominators of (3.3.3) are zero and therefore a special version of SEM *must* be implemented. An example illustrating these points is given in van Dyk, Meng and Rubin (1994). Also, see Meng and Rubin (1991a), especially sections 3.4 and 5, for a related discussion, as well as other implementational considerations (e.g. how computational effort increases with the dimension of θ).

Acknowledgement

This manuscript was prepared using computer facilities supported in part by several NSF Grants awarded to the Department of Statistics at the University of Chicago, by the Fairchild Foundation, and by The University of Chicago Block Fund. The research was supported in part by the NSF Grants DMS 92-04504 and SES 92-07456. It is also supported in part by the U.S. Census Bureau through joint statistical agreements with Harvard University, and through a contract with the National Opinion Research Center at the University of Chicago. We thank Jeff Wu and the referees for helpful comments that improved our presentation. The computer code for the specific algorithms used in the examples (not for the general SECM algorithm) is available upon request.

References

Amemiya, T. (1984). Tobit models: A survey. J. Econometrics **24**, 3-61.

Bishop, Y. M. M., Fienberg, S. E. and Holland, P. W. (1975). *DiscreteMultivariateAnalysis. TheoryandPractice*▷ MIT Press, Cambridge, Massachusetts.

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm (with discussion). J. Roy. Statist. Soc. Ser.B **39,** 1-38.

van Dyk, D. A. (1993). Fitting log-linear models to contingency tables with incomplete data. Technical Report 381. Department of Statistics, University of Chicago.

van Dyk, D. A., Meng, X. L. and Rubin, D. B. (1994). Maximum likelihood estimation via the ECM algorithm: Computing the asymptotic variance. Technical Report 380, Department of Statistics, University of Chicago.

Little, R. J. A. and Rubin, D. B. (1987). Statistical Analysis with Missing Data. John Wiley, New York.

Meng, X. L. (1990). Towards complete results for some incomplete-data problems. Ph.D. Thesis, Harvard University, Department of Statistics. (Printed by U.M.I., Ann Arbor, MI.)

Meng, X. L. (1994). On the rate of convergence of the ECM algorithm. Ann. Statist. **22,** 326-339.

Meng, X. L. and Pedlow, S. (1992). EM: A bibliographic review with missing articles. Proc. Statist. Comp. Sect., 24-27. American Statistical Association, Washington, DC.

Meng, X. L. and Rubin, D. B. (1991a). Using EM to obtain asymptotic variance-covariance matrices: The SEM algorithm. J. Amer. Statist. Assoc. **86,** 899-909.

Meng, X. L. and Rubin, D. B. (1991b). IPF for contingency tables with missing data via the ECM algorithm. Proc. Statist. Comp. Sect., 244-247. American Statistical Association, Washington, DC.

Meng, X. L. and Rubin, D. B. (1992). Recent extensions to the EM algorithm (with discussion). In Bayesian Statistics 4 (Edited by J. M. Bernardo, J. O. Berger, A. P. Dawid and A. F. M. Smith), 307-320. Oxford University Press.

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

Meng, X. L. and Rubin, D. B. (1995). Efficient methods for estimating and testing with seemingly unrelated regressions in the presence of latent variables and missing data. To appear in a special volume in honor of Arnold Zellner. John Wiley, New York.

Orchard, T. and Woodbury, M. A. (1972). A missing information principle theory and application. Proc. 6th Berkeley Symp. Math. Statist. Probab. **1,** 697-715.

Rubin, D. B. (1976). Inference and missing data. Biometrika **63**, 581-592.

Segal, M. R., Bacchetti, P. and Jewell, N. P. (1994). Variances for maximum penalized likelihood estimates obtained via the EM algorithm. J. Roy. Statist. Soc. Ser.B **56**, 345-352.

Wei, G. C. G. and Tanner, M. A. (1990). A monte carlo implementation of the EM algorithm and the poor man's data augmentation algorithms. J. Amer. Statist. Assoc. **85,** 699-704.

Wu, C. F. J. (1983). On the convergence properties of the EM algorithm. Ann. Statist. **11,** 95-103.

Zangwill, W. I. (1969). Nonlinear Programming: A Unified Approach. Englewood Cliffs, Prentice-Hall, New Jersey.

Zellner, A. (1962). An efficient method of estimating seemingly unrelated regressions and tests for aggregation bias. J. Amer. Statist. Assoc. **57,** 348-368.

Department of Statistics, University of Chicago, Chicago, IL 60637, U.S.A.
Department of Statistics, Harvard University, Cambridge, MA 02138, U.S.A.

(Received January 1994; accepted August 1994)