The EM Algorithm

S.K. Ng[1], T. Krishnan[2], and G.J. McLachlan[3]

[1] Department of Mathematics, University of Queensland, Brisbane, QLD 4072, Australia skn@maths.uq.edu.au
[2] Systat Software Asia-Pacific Ltd., Floor 5, C Tower, Golden Enclave, Airport Road, Bangalore krishnan@systat.com
[3] Department of Mathematics, University of Queensland, Brisbane, QLD 4072, Australia gjm@maths.uq.edu.au

1 Introduction

1.1 Maximum Likelihood Estimation

The Expectation-Maximization (EM) algorithm is a broadly applicable approach to the iterative computation of maximum likelihood (ML) estimates, useful in a variety of incomplete-data problems. Maximum likelihood estimation and likelihood-based inference are of central importance in statistical theory and data analysis. Maximum likelihood estimation is a general-purpose method with attractive properties. It is the most-often used estimation technique in the frequentist framework; it is also relevant in the Bayesian framework (Chapter III.11). Often Bayesian solutions are justified with the help of likelihoods and maximum likelihood estimates (MLE), and Bayesian solutions are similar to penalized likelihood estimates. Maximum likelihood estimation is an ubiquitous technique and is used extensively in every area where statistical techniques are used.

We assume that the observed data $\boldsymbol{y}$ has probability density function (p.d.f.) $g(\boldsymbol{y}; \boldsymbol{\Psi})$, where $\boldsymbol{\Psi}$ is the vector containing the unknown parameters in the postulated form for the p.d.f. of $\boldsymbol{Y}$. Our objective is to maximize the likelihood $L(\boldsymbol{\Psi}) = g(\boldsymbol{y}; \boldsymbol{\Psi})$ as a function of $\boldsymbol{\Psi}$, over the parameter space $\boldsymbol{\Omega}$. That is,

$$\partial L(\boldsymbol{\Psi})/\partial \boldsymbol{\Psi} = \mathbf{0},$$

or equivalently, on the log likelihood,

$$\partial \log L(\boldsymbol{\Psi})/\partial \boldsymbol{\Psi} = \mathbf{0}. \tag{1}$$

The aim of ML estimation is to determine an estimate $\hat{\boldsymbol{\Psi}}$, so that it defines a sequence of roots of (1) that is consistent and asymptotically efficient. Such a sequence is known to exist under suitable regularity conditions (Cramér,

1946). With probability tending to one, these roots correspond to local maxima in the interior of $\boldsymbol{\Omega}$. For estimation models in general, the likelihood usually has a global maximum in the interior of $\boldsymbol{\Omega}$. Then typically a sequence of roots of (1) with the desired asymptotic properties is provided by taking $\hat{\boldsymbol{\Psi}}$ to be the root that globally maximizes $L(\boldsymbol{\Psi})$; in this case, $\hat{\boldsymbol{\Psi}}$ is the MLE. We shall henceforth refer to $\hat{\boldsymbol{\Psi}}$ as the MLE, even in situations where it may not globally maximize the likelihood. Indeed, in some of the examples on mixture models (McLachlan and Peel, 2000, Chapter 3), the likelihood is unbounded. However, for these models there may still exist under the usual regularity conditions a sequence of roots of (1) with the properties of consistency, efficiency, and asymptotic normality (McLachlan and Basford, 1988, Chapter 12).

When the likelihood or log likelihood is quadratic in the parameters as in the case of independent normally distributed observations, its maximum can be obtained by solving a system of linear equations in parameters. However, often in practice the likelihood function is not quadratic giving rise to nonlinearity problems in ML estimation. Examples of such situations are: (a) models leading to means which are nonlinear in parameters; (b) despite a possible linear structure, the likelihood is not quadratic in parameters due to, for instance, non-normal errors, missing data, or dependence.

Traditionally ML estimation in these situations has been carried out using numerical iterative methods of solution of equations such as the Newton–Raphson (NR) method and its variants like Fisher's method of scoring. Under reasonable assumptions on $L(\boldsymbol{\Psi})$ and a sufficiently accurate starting value, the sequence of iterates $\{\boldsymbol{\Psi}^{(k)}\}$ produced by the NR method enjoys local quadratic convergence to a solution $\boldsymbol{\Psi}^*$ of (1). Quadratic convergence is regarded as the major strength of the NR method. But in applications, these methods could be tedious analytically and computationally even in fairly simple cases; see McLachlan and Krishnan (1997, Section 1.3) and Meng and van Dyk (1997). The EM algorithm offers an attractive alternative in a variety of settings. It is now a popular tool for iterative ML estimation in a variety of problems involving missing data or incomplete information.

1.2 EM Algorithm: Incomplete-Data Structure

In the application of statistical methods, one is often faced with the problem of estimation of parameters when the likelihood function is complicated in structure resulting in difficult-to-compute maximization problems. This difficulty could be analytical or computational or both. Some examples are grouped, censored or truncated data, multivariate data with some missing observations, multiway frequency data with a complex cell probability structure, and data from mixtures of distributions. In many of these problems, it is often possible to formulate an associated statistical problem with the same parameters with "augmented data" from which it is possible to work out the MLE in an analytically and computationally simpler manner. The augmented data could be called the "complete data" and the available data

could be called the "incomplete data", and the corresponding likelihoods, the "complete-data likelihood" and the "incomplete-data likelihood", respectively, and the corresponding ML estimations, the "complete-data problem" and the "incomplete-data problem". The EM Algorithm is a generic method for computing the MLE of an incomplete-data problem by formulating an associated complete-data problem, and exploiting the simplicity of the MLE of the latter to compute the MLE of the former. The augmented part of the data could also be called "missing data", with respect to the actual incomplete-data problem on hand. The missing data need not necessarily be missing in the practical sense of the word. It may just be a conceptually convenient technical device. Thus the phrase "incomplete data" is used quite broadly to represent a variety of statistical data models, including mixtures, convolutions, random effects, grouping, censoring, truncated and missing observations.

The EM algorithm is an iterative algorithm, in each iteration of which there are two steps, the Expectation Step (E-step) and the Maximization Step (M-step). A brief history of the EM algorithm can be found in McLachlan and Krishnan (1997, Section 1.8). The name EM algorithm was coined by Dempster et al. (1977), who synthesized earlier formulations of this algorithm in many particular cases and presented a general formulation of this method of finding MLE in a variety of problems and provided an initial catalogue of problems where this method could be profitably applied. Since then the EM algorithm has been applied in a staggering variety of general statistical problems such as resolution of mixtures, multiway contingency tables, variance components estimation, factor analysis, as well as in specialized applications in such areas as genetics, medical imaging, and neural networks.

1.3 Overview of the Chapter

In Section 2, the basic theory of the EM algorithm is presented. In particular, the monotonicity of the algorithm, convergence, and rate of convergence properties are systematically examined. In Section 3, the EM methodology presented in this chapter is illustrated in some commonly occurring situations such as the fitting of normal mixtures and missing observations in terms of censored failure times. We also provide an example in which the EM algorithm may not be applicable. Consideration is given also to the two important issues associated with the use of the EM algorithm, namely the initialization of the EM and the provision of standard errors.

We discuss further modifications and extensions to the EM algorithm in Section 4. In particular, the extensions of the EM algorithm known as the Monte Carlo EM, ECM, ECME, AECM, and PX–EM algorithms are considered. With the considerable attention being given to the analysis of large data sets, as in typical data mining applications, recent work on speeding up the implementation of the EM algorithm is discussed. These include the IEM, SPIEM, the scalable EM algorithms, and the use of multiresolution kd-trees.

In Section 5, the relationship of the EM algorithm to other data augmentation techniques, such as the Gibbs sampler and MCMC methods is presented briefly. The Bayesian perspective is also included by showing how the EM algorithm and its variants can be adapted to compute the maximum *a posteriori* (MAP) estimate. We conclude the chapter with a brief account of the applications of the EM algorithm in such topical and interesting areas as Bioinformatics and Image Analysis.

2 Basic Theory of the EM Algorithm

2.1 The E- and M-steps

Within the incomplete-data framework of the EM algorithm, we let $\boldsymbol{x}$ denote the vector containing the complete data and we let $\boldsymbol{z}$ denote the vector containing the missing data. Even when a problem does not at first appear to be an incomplete-data one, computation of the MLE is often greatly facilitated by artificially formulating it to be as such. This is because the EM algorithm exploits the reduced complexity of ML estimation given the complete data. For many statistical problems the complete-data likelihood has a nice form.

We let $g_c(\boldsymbol{x}; \boldsymbol{\Psi})$ denote the p.d.f. of the random vector $\boldsymbol{X}$ corresponding to the complete-data vector $\boldsymbol{x}$. Then the complete-data log likelihood function that could be formed for $\boldsymbol{\Psi}$ if $\boldsymbol{x}$ were fully observable is given by

$$\log L_c(\boldsymbol{\Psi}) = \log g_c(\boldsymbol{x}; \boldsymbol{\Psi}).$$

The EM algorithm approaches the problem of solving the incomplete-data likelihood equation (1) indirectly by proceeding iteratively in terms of $\log L_c(\boldsymbol{\Psi})$. As it is unobservable, it is replaced by its conditional expectation given $\boldsymbol{y}$, using the current fit for $\boldsymbol{\Psi}$. On the $(k+1)$th iteration of the EM algorithm,
E-Step: Compute $Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)})$, where

$$Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)}) = E_{\boldsymbol{\Psi}^{(k)}}\{\log L_c(\boldsymbol{\Psi})|\boldsymbol{y}\}. \tag{2}$$

M-Step: Choose $\boldsymbol{\Psi}^{(k+1)}$ to be any value of $\boldsymbol{\Psi} \in \boldsymbol{\Omega}$ that maximizes $Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)})$:

$$Q(\boldsymbol{\Psi}^{(k+1)}; \boldsymbol{\Psi}^{(k)}) \geq Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)}) \quad \forall \boldsymbol{\Psi} \in \boldsymbol{\Omega}. \tag{3}$$

The E- and M-steps are alternated repeatedly until convergence, which may be determined, for instance, by using a suitable stopping rule like $\|\boldsymbol{\Psi}^{(k+1)} - \boldsymbol{\Psi}^{(k)}\| < \varepsilon$ for some $\varepsilon > 0$ with some appropriate norm $\|\cdot\|$ or the difference $L(\boldsymbol{\Psi}^{(k+1)}) - L(\boldsymbol{\Psi}^{(k)})$ changes by an arbitrarily small amount in the case of convergence of the sequence of likelihood values $\{L(\boldsymbol{\Psi}^{(k)})\}$.

It can be shown that both the E- and M-steps will have particularly simple forms when $g_c(\boldsymbol{x}; \boldsymbol{\Psi})$ is from an exponential family:

$$g_c(\boldsymbol{x};\boldsymbol{\Psi}) = b(\boldsymbol{x})\exp\{\boldsymbol{c}^\top(\boldsymbol{\Psi})\boldsymbol{t}(\boldsymbol{x})\}/a(\boldsymbol{\Psi}), \tag{4}$$

where $\boldsymbol{t}(\boldsymbol{x})$ is a $k\times 1$ ($k\geq d$) vector of complete-data sufficient statistics and $\boldsymbol{c}(\boldsymbol{\Psi})$ is a $k\times 1$ vector function of the $d\times 1$ parameter vector $\boldsymbol{\Psi}$, and $a(\boldsymbol{\Psi})$ and $b(\boldsymbol{x})$ are scalar functions. Members of the exponential family include most common distributions, such as the multivariate normal, Poisson, multinomial and others. For exponential families, the E-step can be written as

$$Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)}) = E_{\boldsymbol{\Psi}^{(k)}}(\log b(\boldsymbol{x})|\boldsymbol{y}) + \boldsymbol{c}^\top(\boldsymbol{\Psi})\boldsymbol{t}^{(k)} - \log a(\boldsymbol{\Psi}),$$

where $\boldsymbol{t}^{(k)} = E_{\boldsymbol{\Psi}^{(k)}}\{\boldsymbol{t}(\boldsymbol{X})|\boldsymbol{y}\}$ is an estimator of the sufficient statistic. The M-step maximizes the Q-function with respect to $\boldsymbol{\Psi}$; but $E_{\boldsymbol{\Psi}^{(k)}}(\log b(\boldsymbol{x})|\boldsymbol{y})$ does not depend on $\boldsymbol{\Psi}$. Hence it is sufficient to write:

E–Step: Compute

$$\boldsymbol{t}^{(k)} = E_{\boldsymbol{\Psi}^{(k)}}\{\boldsymbol{t}(\boldsymbol{X})|\boldsymbol{y}\}.$$

M–Step: Compute

$$\boldsymbol{\Psi}^{(k+1)} = \arg\max_{\boldsymbol{\Psi}}[\boldsymbol{c}^\top(\boldsymbol{\Psi})\boldsymbol{t}^{(k)} - \log a(\boldsymbol{\Psi})].$$

In Example 2 of Section 3.2, the complete-data p.d.f. has an exponential family representation. We shall show how the implementation of the EM algorithm can be simplified.

2.2 Generalized EM Algorithm

Often in practice, the solution to the M-step exists in closed form. In those instances where it does not, it may not be feasible to attempt to find the value of $\boldsymbol{\Psi}$ that globally maximizes the function $Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)})$. For such situations, Dempster et al. (1977) defined a generalized EM (GEM) algorithm for which the M-Step requires $\boldsymbol{\Psi}^{(k+1)}$ to be chosen such that

$$Q(\boldsymbol{\Psi}^{(k+1)};\boldsymbol{\Psi}^{(k)}) \geq Q(\boldsymbol{\Psi}^{(k)};\boldsymbol{\Psi}^{(k)}) \tag{5}$$

holds. That is, one chooses $\boldsymbol{\Psi}^{(k+1)}$ to increase the Q-function, $Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)})$, over its value at $\boldsymbol{\Psi}=\boldsymbol{\Psi}^{(k)}$, rather than to maximize it over all $\boldsymbol{\Psi}\in\boldsymbol{\Omega}$ in (3).

It is of interest to note that the EM (GEM) algorithm as described above implicitly defines a mapping $\boldsymbol{\Psi}\rightarrow\boldsymbol{M}(\boldsymbol{\Psi})$, from the parameter space $\boldsymbol{\Omega}$ to itself such that

$$\boldsymbol{\Psi}^{(k+1)} = \boldsymbol{M}(\boldsymbol{\Psi}^{(k)}) \qquad (k=0,1,2,\ldots).$$

The function $\boldsymbol{M}$ is called the EM mapping. We shall use this function in our subsequent discussion on the convergence property of the EM algorithm.

2.3 Convergence of the EM Algorithm

Let $k(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\Psi}) = g_c(\boldsymbol{x};\boldsymbol{\Psi})/g(\boldsymbol{y};\boldsymbol{\Psi})$ be the conditional density of $\boldsymbol{X}$ given $\boldsymbol{Y} = \boldsymbol{y}$. Then the complete-data log likelihood can be expressed by

$$\log L_c(\boldsymbol{\Psi}) = \log g_c(\boldsymbol{x};\boldsymbol{\Psi}) = \log L(\boldsymbol{\Psi}) + \log k(\boldsymbol{x}|\boldsymbol{y};\boldsymbol{\Psi}). \tag{6}$$

Taking expectations on both sides of (6) with respect to the conditional distribution $\boldsymbol{x}|\boldsymbol{y}$ using the fit $\boldsymbol{\Psi}^{(k)}$ for $\boldsymbol{\Psi}$, we have

$$Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)}) = \log L(\boldsymbol{\Psi}) + H(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)}), \tag{7}$$

where $H(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)}) = E_{\boldsymbol{\Psi}^{(k)}}\{\log k(\boldsymbol{X}|\boldsymbol{y};\boldsymbol{\Psi})|\boldsymbol{y}\}$. It follows from (7) that

$$\begin{aligned}\log L(\boldsymbol{\Psi}^{(k+1)}) - \log L(\boldsymbol{\Psi}^{(k)}) = \{&Q(\boldsymbol{\Psi}^{(k+1)};\boldsymbol{\Psi}^{(k)}) - Q(\boldsymbol{\Psi}^{(k)};\boldsymbol{\Psi}^{(k)})\} \\ &-\{H(\boldsymbol{\Psi}^{(k+1)};\boldsymbol{\Psi}^{(k)}) - H(\boldsymbol{\Psi}^{(k)};\boldsymbol{\Psi}^{(k)})\}.\end{aligned} \tag{8}$$

By Jensen's inequality, we have $H(\boldsymbol{\Psi}^{(k+1)};\boldsymbol{\Psi}^{(k)}) \leq H(\boldsymbol{\Psi}^{(k)};\boldsymbol{\Psi}^{(k)})$. From (3) or (5), the first difference on the right-hand side of (8) is nonnegative. Hence, the likelihood function is not decreased after an EM or GEM iteration:

$$L(\boldsymbol{\Psi}^{(k+1)}) \geq L(\boldsymbol{\Psi}^{(k)}) \qquad (k = 0, 1, 2, \ldots). \tag{9}$$

A consequence of (9) is the self-consistency of the EM algorithm. Thus for a bounded sequence of likelihood values $\{L(\boldsymbol{\Psi}^{(k)})\}$, $L(\boldsymbol{\Psi}^{(k)})$ converges monotonically to some L^*. Now questions naturally arise as to the conditions under which L^* corresponds to a stationary value and when this stationary value is at least a local maximum if not a global maximum. Examples are known where the EM algorithm converges to a local *minimum* and to a saddle point of the likelihood (McLachlan and Krishnan, 1997, Section 3.6). There are also questions of convergence of the sequence of EM iterates, that is, of the sequence of parameter values $\{\boldsymbol{\Psi}^{(k)}\}$ to the MLE.

Before the general formulation of the EM algorithm in Dempster et al. (1977), there have been convergence results for special cases, notable among them being those of Baum et al. (1970) for what is now being called the hidden Markov model. In the article of Dempster et al. (1977) itself, there are some convergence results. However, it is Wu (1983) who investigates in detail several convergence issues of the EM algorithm in its generality. Wu examines these issues through their relationship to other optimization methods. He shows that when the complete data are from a curved exponential family with compact parameter space, and when the Q-function satisfies a certain mild differentiability condition, then any EM sequence converges to a stationary point (not necessarily a maximum) of the likelihood function. If $L(\boldsymbol{\Psi})$ has multiple stationary points, convergence of the EM sequence to either type (local or global maximizers, saddle points) depends upon the starting value $\boldsymbol{\Psi}^{(0)}$ for $\boldsymbol{\Psi}$. If $L(\boldsymbol{\Psi})$ is unimodal in $\boldsymbol{\Omega}$ and satisfies the same differentiability

condition, then any sequence $\{\boldsymbol{\Psi}^{(k)}\}$ will converge to the unique MLE of $\boldsymbol{\Psi}$, irrespective of its starting value.

To be more specific, one of the basic convergence results of the EM algorithm is the following:

$$\log L(\boldsymbol{M}(\boldsymbol{\Psi})) \geq \log L(\boldsymbol{\Psi})$$

with equality if and only if

$$Q(\boldsymbol{M}(\boldsymbol{\Psi}); \boldsymbol{\Psi}) = Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}) \quad \text{and} \quad k(\boldsymbol{x}|\boldsymbol{y}; \boldsymbol{M}(\boldsymbol{\Psi})) = k(\boldsymbol{x}|\boldsymbol{y}; \boldsymbol{\Psi}).$$

This means that the likelihood function increases at each iteration of the EM algorithm, until the condition for equality is satisfied and a fixed point of the iteration is reached. If $\hat{\boldsymbol{\Psi}}$ is an MLE, so that $\log L(\hat{\boldsymbol{\Psi}}) \geq \log L(\boldsymbol{\Psi}),\ \forall\ \boldsymbol{\Psi} \in \boldsymbol{\Omega}$, then $\log L(\boldsymbol{M}(\hat{\boldsymbol{\Psi}})) = \log L(\hat{\boldsymbol{\Psi}})$. Thus MLE are fixed points of the EM algorithm. If we have the likelihood function bounded (as might happen in many cases of interest), the EM sequence $\{\boldsymbol{\Psi}^{(k)}\}$ yields a bounded nondecreasing sequence $\{\log L(\boldsymbol{\Psi}^{(k)})\}$ which must converge as $k \to \infty$.

The theorem does not quite imply that fixed points of the EM algorithm are in fact MLEs. This is however true under fairly general conditions. For proofs and other details, see McLachlan and Krishnan (1997, Section 3.5) and Wu (1983). Furthermore, if a sequence of EM iterates $\{\boldsymbol{\Psi}^{(k)}\}$ satisfy the conditions

1. $[\partial Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)})/\partial \boldsymbol{\Psi}]_{\boldsymbol{\Psi}=\boldsymbol{\Psi}^{(k+1)}} = \mathbf{0}$, and
2. the sequence $\{\boldsymbol{\Psi}^{(k)}\}$ converges to some value $\boldsymbol{\Psi}^*$ and $\log k(\boldsymbol{x}|\boldsymbol{y}; \boldsymbol{\Psi})$ is sufficiently smooth,

then we have $[\partial \log L(\boldsymbol{\Psi})/\partial \boldsymbol{\Psi}]_{\boldsymbol{\Psi}=\boldsymbol{\Psi}^*} = \mathbf{0}$; see Little and Rubin (2002) and Wu (1983). Thus, despite the earlier convergence results, there is no guarantee that the convergence will be to a global maximum. For likelihood functions with multiple maxima, convergence will be to a local maximum which depends on the starting value $\boldsymbol{\Psi}^{(0)}$.

Nettleton (1999) extends Wu's convergence results to the case of constrained parameter spaces and establishes some stricter conditions to guarantee convergence of the EM likelihood sequence to some local maximum and the EM parameter iterates to converge to the MLE.

2.4 Rate of Convergence of the EM Algorithm

The rate of convergence of the EM algorithm is also of practical interest. The convergence rate is usually slower than the quadratic convergence typically available with Newton-type methods. Dempster et al. (1977) show that the rate of convergence of the EM algorithm is linear and the rate depends on the proportion of information in the observed data. Thus in comparison to the formulated complete-data problem, if a large portion of data is missing, convergence can be quite slow.

Recall the EM mapping $\boldsymbol{M}$ defined in Section 2.2. If $\boldsymbol{\Psi}^{(k)}$ converges to some point $\boldsymbol{\Psi}^*$ and $\boldsymbol{M}(\boldsymbol{\Psi})$ is continuous, then $\boldsymbol{\Psi}^*$ is a fixed point of the algorithm; that is, $\boldsymbol{\Psi}^*$ must satisfy $\boldsymbol{\Psi}^* = \boldsymbol{M}(\boldsymbol{\Psi}^*)$. By a Taylor series expansion of $\boldsymbol{\Psi}^{(k+1)} = \boldsymbol{M}(\boldsymbol{\Psi}^{(k)})$ about the point $\boldsymbol{\Psi}^{(k)} = \boldsymbol{\Psi}^*$, we have in a neighborhood of $\boldsymbol{\Psi}^*$ that

$$\boldsymbol{\Psi}^{(k+1)} - \boldsymbol{\Psi}^* \approx \boldsymbol{J}(\boldsymbol{\Psi}^*)(\boldsymbol{\Psi}^{(k)} - \boldsymbol{\Psi}^*),$$

where $\boldsymbol{J}(\boldsymbol{\Psi})$ is the $d \times d$ Jacobian matrix for $\boldsymbol{M}(\boldsymbol{\Psi}) = (M_1(\boldsymbol{\Psi}), \ldots, M_d(\boldsymbol{\Psi}))^\top$, having (i,j)th element $r_{ij}(\boldsymbol{\Psi})$ equal to

$$r_{ij}(\boldsymbol{\Psi}) = \partial M_i(\boldsymbol{\Psi})/\partial \Psi_j,$$

where $\Psi_j = (\boldsymbol{\Psi})_j$ and d is the dimension of $\boldsymbol{\Psi}$. Thus, in a neighborhood of $\boldsymbol{\Psi}^*$, the EM algorithm is essentially a linear iteration with rate matrix $\boldsymbol{J}(\boldsymbol{\Psi}^*)$, since $J(\boldsymbol{\Psi}^*)$ is typically nonzero. For this reason, $\boldsymbol{J}(\boldsymbol{\Psi}^*)$ is often referred to as the matrix rate of convergence. For vector $\boldsymbol{\Psi}$, a measure of the actual observed convergence rate is the global rate of convergence, which is defined as

$$r = \lim_{k \to \infty} \| \boldsymbol{\Psi}^{(k+1)} - \boldsymbol{\Psi}^* \| / \| \boldsymbol{\Psi}^{(k)} - \boldsymbol{\Psi}^* \|,$$

where $\| \cdot \|$ is any norm on d-dimensional Euclidean space $\Re^d$. It is noted that the observed rate of convergence equals the largest eigenvalue of $\boldsymbol{J}(\boldsymbol{\Psi}^*)$ under certain regularity conditions (Meng and van Dyk, 1997). As a large value of r implies slow convergence, the global speed of convergence is defined to be $s = 1 - r$ (Meng, 1994).

2.5 Properties of the EM Algorithm

The EM algorithm has several appealing properties, some of which are:

1. It is numerically stable with each EM iteration increasing the likelihood.
2. Under fairly general conditions, it has reliable global convergence.
3. It is easily implemented, analytically and computationally. In particular, it is generally easy to program and requires small storage space. By watching the monotone increase in likelihood (if evaluated easily) over iterations, it is easy to monitor convergence and programming errors (McLachlan and Krishnan, 1997, Section 1.7).
4. The cost per iteration is generally low, which can offset the larger number of iterations needed for the EM algorithm compared to other competing procedures.
5. It can be used to provide estimates of missing data.

Some of its drawbacks are:

1. It does not automatically provide an estimate of the covariance matrix of the parameter estimates. However, this disadvantage can be easily removed by using appropriate methodology associated with the EM algorithm (McLachlan and Krishnan, 1997, Chapter 4).

2. It is sometimes very slow to converge.
3. In some problems, the E- or M-steps may be analytically intractable.

We shall briefly address these three issues in Sections 3.5 and 4.

3 Examples of the EM algorithm

3.1 Example 1: Normal Mixtures

One of the classical formulations of the two-group discriminant analysis or the statistical pattern recognition problem involves a mixture of two p-dimensional normal distributions with a common covariance matrix. The problem of two-group cluster analysis with multiple continuous observations has also been formulated in this way. Here, we have n independent observations $\boldsymbol{y}_1, \boldsymbol{y}_2, \ldots, \boldsymbol{y}_n$ from the mixture density

$$(1-\pi)\phi(\boldsymbol{y};\, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) + \pi\phi(\boldsymbol{y};\, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}),$$

where $\phi(\boldsymbol{y};\, \boldsymbol{\mu}_i, \boldsymbol{\Sigma})$ denotes the p-dimensional normal density function with mean vector $\boldsymbol{\mu}_i$ and common covariance matrix $\boldsymbol{\Sigma}$, $i = 1, 2$. The $(1-\pi)$ and π denote the proportions of the two clusters, respectively. The problem of estimating the parameters $\boldsymbol{\Psi} = (\pi, \boldsymbol{\mu}_1,\, \boldsymbol{\mu}_2,\, \boldsymbol{\Sigma})$ is an instance of the problem of resolution of mixtures or in pattern recognition parlance an "unsupervised learning problem". The MLE problem here is quite messy and classical statistical and pattern recognition literature has struggled with it for a long time.

Consider the corresponding "supervised learning problem", where observations on the random vector $\boldsymbol{X} = (Z, \boldsymbol{Y})$ are $\boldsymbol{x}_1 = (z_1, \boldsymbol{y}_1)$, $\boldsymbol{x}_2 = (z_2, \boldsymbol{y}_2), \ldots,$ $\boldsymbol{x}_n = (z_n, \boldsymbol{y}_n)$. Here z_j is an indicator variable which identifies the jth observation as coming from the first ($z = 0$) or the second ($z = 1$) component ($j = 1, \ldots, n$). The MLE problem is far simpler here with easy closed-form MLE. The classificatory variable z_j could be called the "missing variable" and data $\boldsymbol{z} = (z_1, z_2, \ldots, z_n)^\top$ the missing data. The unsupervised learning problem could be called the incomplete-data problem and the supervised learning problem the complete-data problem. A relatively simple iterative method for computing the MLE for the unsupervised problem could be given exploiting the simplicity of the MLE for the supervised problem. This is the essence of the EM algorithm.

The complete-data log likelihood function for $\boldsymbol{\Psi}$ is given by

$$\log L_c(\boldsymbol{\Psi}) = \sum_{j=1}^{n} (1-z_j) \log \phi(\boldsymbol{y};\, \boldsymbol{\mu}_1, \boldsymbol{\Sigma}) + z_j \log \phi(\boldsymbol{y};\, \boldsymbol{\mu}_2, \boldsymbol{\Sigma}). \qquad (10)$$

By differentiating (10) with respect to $\boldsymbol{\Psi}$, the MLEs of $\boldsymbol{\Psi}$ are obtained, as if $\boldsymbol{z}$ were actually observed:

$$\pi = \sum_{j=1}^{n} z_j/n, \tag{11}$$

$$\boldsymbol{\mu}_1 = \sum_{j=1}^{n}(1-z_j)\boldsymbol{y}_j/(n-\sum_{j=1}^{n} z_j), \quad \boldsymbol{\mu}_2 = \sum_{j=1}^{n} z_j\boldsymbol{y}_j/\sum_{j=1}^{n} z_j, \tag{12}$$

$$\boldsymbol{\Sigma} = \sum_{j=1}^{n}[(1-z_j)(\boldsymbol{y}_j-\boldsymbol{\mu}_1)(\boldsymbol{y}_j-\boldsymbol{\mu}_1)^\top + z_j(\boldsymbol{y}_j-\boldsymbol{\mu}_2)(\boldsymbol{y}_j-\boldsymbol{\mu}_2)^\top]/n, \tag{13}$$

Now the EM algorithm for this problem starts with some initial value $\boldsymbol{\Psi}^{(0)}$ for the parameters. As $\log L_c(\boldsymbol{\Psi})$ in (10) is a linear function of the unobservable data $\boldsymbol{z}$ for this problem, the calculation of $Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)})$ on the E-step is effected simply by replacing z_j by its current conditional expectation given the observed data $\boldsymbol{y}$, which is the usual posterior probability of the jth observation arising from component 2

$$\tau_j^{(k)} = E_{\boldsymbol{\Psi}^{(k)}}(Z_j|\boldsymbol{y}) = \frac{\pi^{(k)}\phi(\boldsymbol{y}_j; \boldsymbol{\mu}_2^{(k)}, \boldsymbol{\Sigma}^{(k)})}{(1-\pi^{(k)})\phi(\boldsymbol{y}_j; \boldsymbol{\mu}_1^{(k)}, \boldsymbol{\Sigma}^{(k)}) + \pi^{(k)}\phi(\boldsymbol{y}_j; \boldsymbol{\mu}_2^{(k)}, \boldsymbol{\Sigma}^{(k)})}.$$

The M-step then consists of substituting these $\tau_j^{(k)}$ values for z_j in equations (11) to (13). The E- and M-steps are then iterated until convergence. Unlike in the MLE for the supervised problem, in the M-step of the unsupervised problem, the posterior probabilities τ_j, which are between 0 and 1, are used. The mean vectors $\boldsymbol{\mu}_i$ $(i = 1, 2)$ and the covariance matrix $\boldsymbol{\Sigma}$ are computed using the $\tau_j^{(k)}$ as weights in weighted averages.

It is easy to extend the above method to a mixture of $g > 2$ multinormal mixtures or even to a mixture of $g > 2$ distributions from other (identifiable) families. For a detailed discussion of the applications of the EM algorithm in the resolution of finite mixtures and other issues of finite mixtures, see McLachlan and Peel (2000).

3.2 Example 2: Censored Failure-Time Data

In survival or reliability analyses, the focus is the distribution of time T to the occurrence of some event that represents failure (for computational methods in survival analysis see also Chapter III.12). In many situations, there will be individuals who do not fail at the end of the study, or individuals who withdraw from the study before it ends. Such observations are censored, as we know only that their failure times are greater than particular values. We let $\boldsymbol{y} = (c_1, \delta_1, \ldots, c_n, \delta_n)^\top$ denote the observed failure-time data, where $\delta_j = 0$ or 1 according as the jth observation T_j is censored or uncensored at c_j $(j = 1, \ldots, n)$. That is, if T_j is uncensored, $t_j = c_j$, whereas if $t_j > c_j$, it is censored at c_j.

In the particular case where the p.d.f. for T is exponential with mean μ, we have

$$f(t;\,\mu) = \mu^{-1}\exp(-t/\mu)I_{(0,\infty)}(t) \qquad (\mu > 0), \tag{14}$$

where the indicator function $I_{(0,\infty)}(t) = 1$ for $t > 0$ and is zero elsewhere. The unknown parameter vector $\boldsymbol{\Psi}$ is now a scalar, being equal to μ. Denote by s the number of uncensored observations. By re-ordering the data so that the uncensored observations precede censored observations. It can be shown that the log likelihood function for μ is given by

$$\log L(\mu) = -s\log\mu - \sum_{j=1}^{n} c_j/\mu. \tag{15}$$

By equating the derivative of (15) to zero, the MLE of μ is

$$\hat{\mu} = \sum_{j=1}^{n} c_j/s. \tag{16}$$

Thus there is no need for the iterative computation of $\hat{\mu}$. But in this simple case, it is instructive to demonstrate how the EM algorithm would work and how its implementation could be simplified as the complete-data log likelihood belongs to the regular exponential family (see Section 2.1).

The complete-data vector $\boldsymbol{x}$ can be declared to be $\boldsymbol{x} = (t_1, \ldots, t_s, \boldsymbol{z}^\top)^\top$, where $\boldsymbol{z} = (t_{s+1}, \ldots, t_n)^\top$ contains the unobservable realizations of the $n - s$ censored random variables. The complete-data log likelihood is given by

$$\log L_c(\mu) = -n\log\mu - \sum_{j=1}^{n} t_j/\mu. \tag{17}$$

As $\log L_c(\mu)$ is a linear function of the unobservable data $\boldsymbol{z}$, the E-step is effected simply by replacing $\boldsymbol{z}$ by its current conditional expectation given $\boldsymbol{y}$. By the lack of memory of the exponential distribution, the conditional distribution of $T_j - c_j$ given that $T_j > c_j$ is still exponential with mean μ. So, we have

$$E_{\mu^{(k)}}(T_j|\boldsymbol{y}) = E_{\mu^{(k)}}(T_j|T_j > c_j) = c_j + \mu^{(k)} \tag{18}$$

for $j = s+1, \ldots, n$. Accordingly, the Q-function is given by

$$Q(\mu;\,\mu^{(k)}) = -n\log\mu - \mu^{-1}\left\{\sum_{j=1}^{n} c_j + (n-s)\mu^{(k)}\right\}.$$

In the M-step, we have

$$\mu^{(k+1)} = \left\{\sum_{j=1}^{n} c_j + (n-s)\mu^{(k)}\right\} \Big/ n. \tag{19}$$

On putting $\mu^{(k+1)} = \mu^{(k)} = \mu^*$ in (19) and solving for μ^*, we have for $s < n$ that $\mu^* = \hat{\mu}$. That is, the EM sequence $\{\mu^{(k)}\}$ has the MLE $\hat{\mu}$ as its unique limit point, as $k \to \infty$; see McLachlan and Krishnan (1997, Section 1.5.2).

From (17), it can be seen that $\log L_c(\mu)$ has the exponential family form (4) with canonical parameter μ^{-1} and sufficient statistic $\boldsymbol{t}(\boldsymbol{X}) = \sum_{j=1}^{n} T_j$. Hence, from (18), the E-step requires the calculation of $\boldsymbol{t}^{(k)} = \sum_{j=1}^{n} c_j + (n-s)\mu^{(k)}$. The M-step then yields $\mu^{(k+1)}$ as the value of μ that satisfies the equation

$$\boldsymbol{t}^{(k)} = E_{\mu}\{\boldsymbol{t}(\boldsymbol{X})\} = n\mu.$$

This latter equation can be seen to be equivalent to (19), as derived by direct differentiation of the Q-function.

3.3 Example 3: Nonapplicability of EM algorithm

Examples 1 and 2 may have given an impression that the E-step consists in replacing the missing data by their conditional expectations given the observed data at current parameter values. Although in many examples this may be the case as $\log L_c(\boldsymbol{\Psi})$ is a linear function of the missing data $\boldsymbol{z}$, it is not quite so in general. Rather, as should be clear from the general theory described in Section 2.1, the E-step consists in replacing $\log L_c(\boldsymbol{\Psi})$ by its conditional expectation given the observed data at current parameter values. Flury and Zoppé (2000) give the following interesting example to demonstrate the point that the E-step does not always consist in plugging in "estimates" for missing data. This is also an example where the E-step cannot be correctly executed at all since the expected value of the complete-data log likelihood does not exist, showing thereby that the EM algorithm is not applicable to this problem, at least for this formulation of the complete-data problem.

Let the lifetimes of electric light bulbs of a certain type have a uniform distribution in the interval $(0, \theta]$, $\theta > 0$ and unknown. A total of $n + m$ bulbs are tested in two independent experiments. The observed data consist of $\boldsymbol{y} = (y_1, \ldots, y_n)$ and $\boldsymbol{e} = (e_{n+1}, \ldots, e_{n+m})$, where $\boldsymbol{y}$ are exact lifetimes of a random sample of n bulbs and $\boldsymbol{e}$ are indicator observations on a random sample of m bulbs, taking value 1 if the bulb is still burning at a fixed time point $T > 0$ and 0 if it is expired. The missing data is $\boldsymbol{z} = (y_{n+1}, \ldots, y_{n+m})^{\top}$. Let s be the number of e_j's with value 1 and $y_{\max} = \max\{y_1, \ldots, y_n\}$.

In this example, the unknown parameter vector $\boldsymbol{\Psi}$ is a scalar, being equal to θ. Let us first work out the MLE of θ directly. The likelihood is

$$L(\theta) = \theta^{-n} I_{[y_{\max}, \infty)}(\theta) \times \left(\frac{T}{\max(T, \theta)}\right)^{m-s} \left(1 - \frac{T}{\max(T, \theta)}\right)^{s},$$

where I_A is the notation for the indicator function of set A. For $s = 0$, $L(\theta)$ is decreasing in θ for $\theta \geq y_{\max}$ and hence the MLE is $\hat{\theta} = y_{\max}$. For $s \geq 1$, we have $\max(T, \theta) = \theta$. Here the function $L_1(\theta) = (\theta)^{-(n+m)}(\theta - T)^s$ has a

unique maximum at $\tilde{\theta} = \frac{n+m}{n+m-s}T$ and is monotonically decreasing for $\theta > \tilde{\theta}$. Hence the MLE of θ is

$$\hat{\theta} = \begin{cases} \tilde{\theta} & \text{if } \tilde{\theta} > y_{\max} \text{ and } s \geq 1 \\ y_{\max} & \text{otherwise.} \end{cases}$$

Now let us try the EM algorithm for the case $s \geq 1$. The complete data can be formulated as $y_1, \ldots, y_n, y_{n+1}, \ldots, y_{n+m}$ and the complete-data MLE is

$$\max_{j=1,\ldots,n+m} y_j.$$

Since $s \geq 1$, we have $\theta \geq T$. Now if we take the approach of replacing the missing observations, then we compute

$$E_{\theta^{(k)}}(y_j^{(k+1)}|\boldsymbol{y}, \boldsymbol{e}) = E_{\theta^{(k)}}(y_j|e_j) = \begin{cases} \frac{1}{2}(T+\theta) \text{ if } e_j = 1 \\ \\ \frac{1}{2}T \qquad\quad \text{if } e_j = 0 \end{cases}$$

for $j = n+1, \ldots, n+m$. The M-step is

$$\theta^{(k+1)} = \max\{y_{\max}, \tfrac{1}{2}(T + \theta^{(k)})\}.$$

Combining the E- and M-steps, we can write the EM algorithm as a sequence of iterations of the equation

$$\theta^{(k+1)} = \boldsymbol{M}(\theta^k) \equiv \max\{y_{\max}, \tfrac{1}{2}(T + \theta^{(k)})\}.$$

It is easily seen that if we start with any $\theta^{(0)}$, this procedure will converge to $\hat{\theta} = \max\{y_{\max}, T\}$, by noting that $\hat{\theta} = \boldsymbol{M}(\hat{\theta})$.

The reason for the apparent EM algorithm not resulting in the MLE is that the E-step is wrong. In the E-step, we are supposed to find the conditional expectation of $\log L_c(\theta)$ given $\boldsymbol{y}, \boldsymbol{e}$ at current parameter values. Now given the data with $s \geq 1$, we have $\theta \geq T$ and hence the conditional distributions of y_j are uniform in $[T, \theta^{(k)}]$. Thus for $\theta < \theta^{(k)}$ the conditional density of missing y_j takes value 0 with positive probability and hence the conditional expected value of the complete-data log likelihood we are seeking does not exist.

3.4 Starting values for EM Algorithm

The EM algorithm will converge very slowly if a poor choice of initial value $\boldsymbol{\Psi}^{(0)}$ were used. Indeed, in some cases where the likelihood is unbounded on the edge of the parameter space, the sequence of estimates $\{\boldsymbol{\Psi}^{(k)}\}$ generated by the EM algorithm may diverge if $\boldsymbol{\Psi}^{(0)}$ is chosen too close to the boundary. Also, with applications where the likelihood equation has multiple roots corresponding to local maxima, the EM algorithm should be applied from a wide choice of starting values in any search for all local maxima. A variation of the EM algorithm (Wright and Kennedy, 2000) uses interval analysis methods

to locate multiple stationary points of a log likelihood within any designated region of the parameter space.

Here, we illustrate different ways of specification of initial value within mixture models framework. For independent data in the case of mixture models of g components, the effect of the E-step is to update the posterior probabilities of component membership. Hence the first E-step can be performed by specifying a value $\boldsymbol{\tau}_j^{(0)}$ for each j $(j = 1, \ldots, n)$, where $\boldsymbol{\tau}_j = (\tau_{1j}, \ldots, \tau_{gj})^\top$ is the vector containing the g posterior probabilities of component membership for $\boldsymbol{y}_j$. The latter is usually undertaken by setting $\boldsymbol{\tau}_j^{(0)} = \boldsymbol{z}_j^{(0)}$, where

$$\boldsymbol{z}^{(0)} = (\boldsymbol{z}_1^{(0)\top}, \ldots, \boldsymbol{z}_n^{(0)\top})^\top$$

defines an initial partition of the data into g components. For example, an *ad hoc* way of initially partitioning the data in the case of, say, a mixture of $g = 2$ normal components with the same covariance matrices (Example 1, Section 3.1) would be to plot the data for selections of two of the p variables, and then draw a line that divides the bivariate data into two groups that have a scatter that appears normal. For higher-dimensional data, an initial value $\boldsymbol{z}^{(0)}$ for $\boldsymbol{z}$ might be obtained through the use of some clustering algorithm, such as k-means or, say, an hierarchical procedure if n is not too large.

Another way of specifying an initial partition $\boldsymbol{z}^{(0)}$ of the data is to randomly divide the data into g groups corresponding to the g components of the mixture model. With random starts, the effect of the central limit theorem tends to have the component parameters initially being similar at least in large samples. One way to reduce this effect is to first select a small random subsample from the data, which is then randomly assigned to the g components. The first M-step is then performed on the basis of the subsample. The subsample has to be sufficiently large to ensure that the first M-step is able to produce a nondegenerate estimate of the parameter vector $\boldsymbol{\Psi}$ (McLachlan and Peel, 2000, Section 2.12). In the context of g normal components, a method of specifying a random start is to generate the means $\boldsymbol{\mu}_i^{(0)}$ independently as

$$\boldsymbol{\mu}_1^{(0)}, \ldots, \boldsymbol{\mu}_g^{(0)} \overset{i.i.d.}{\sim} N(\bar{\boldsymbol{y}}, \boldsymbol{V}),$$

where $\bar{\boldsymbol{y}}$ is the sample mean and $\boldsymbol{V} = \sum_{j=1}^{n}(\boldsymbol{y}_j - \bar{\boldsymbol{y}})(\boldsymbol{y}_j - \bar{\boldsymbol{y}})^\top/n$ is the sample covariance matrix of the observed data. With this method, there is more variation between the initial values $\boldsymbol{\mu}_i^{(0)}$ for the component means $\boldsymbol{\mu}_i$ than with a random partition of the data into g components. The component-covariance matrices $\boldsymbol{\Sigma}_i$ and the mixing proportions π_i can be specified as

$$\boldsymbol{\Sigma}_i^{(0)} = \boldsymbol{V} \text{ and } \pi_i^{(0)} = 1/g \quad (i = 1, \ldots, g).$$

Ueda and Nakano (1998) considered a deterministic annealing EM (DAEM) algorithm in order for the EM iterative process to be able to recover from a

poor choice of starting value. They proposed using the principle of maximum entropy and the statistical mechanics analogy, whereby a parameter, say θ, is introduced with $1/\theta$ corresponding to the "temperature" in an annealing sense. With their DAEM algorithm, the E-step is effected by averaging $\log L_c(\boldsymbol{\Psi})$ over the distribution taken to be proportional to that of the current estimate of the conditonal density of the complete data (given the observed data) raised to the power of θ; see for example McLachlan and Peel (2000, pp. 58–60).

3.5 Provision of Standard Errors

Several methods have been suggested in the EM literature for augmenting the EM computation with some computation for obtaining an estimate of the covariance matrix of the computed MLE. Many such methods attempt to exploit the computations in the EM steps. These methods are based on the observed information matrix $\boldsymbol{I}(\hat{\boldsymbol{\Psi}}; \boldsymbol{y})$, the expected information matrix $\mathcal{I}(\boldsymbol{\Psi})$ or on resampling methods. Baker (1992) reviews such methods and also develops a method for computing the observed information matrix in the case of categorical data. Jamshidian and Jennrich (2000) review more recent methods including the Supplemented EM (SEM) algorithm of Meng and Rubin (1991) and suggest some newer methods based on numerical differentiation.

Theorectically one may compute the asymptotic covariance matrix by inverting the observed or expected information matrix at the MLE. In practice, however, this may be tedious analytically or computationally, defeating one of the advantages of the EM approach. Louis (1982) extracts the observed information matrix in terms of the conditional moments of the gradient and curvature of the complete-data log likelihood function introduced within the EM framework. These conditional moments are generally easier to work out than the corresponding derivatives of the incomplete-data log likelihood function. An alternative approach is to numerically differentiate the likelihood function to obtain the Hessian. In a EM-aided differentiation approach, Meilijson (1989) suggests perturbation of the incomplete-data score vector to compute the observed information matrix. In the SEM algorithm (Meng and Rubin, 1991), numerical techniques are used to compute the derivative of the EM operator $\boldsymbol{M}$ to obtain the observed information matrix. The basic idea is to use the fact that the rate of convergence is governed by the fraction of the missing information to find the increased variability due to missing information to add to the assessed complete-data covariance matrix. More specifically, let $\boldsymbol{V}$ denote the asymptotic covariance matrix of the MLE $\hat{\boldsymbol{\Psi}}$. Meng and Rubin (1991) show that

$$\boldsymbol{I}^{-1}(\hat{\boldsymbol{\Psi}}; \boldsymbol{y}) = \mathcal{I}_c^{-1}(\hat{\boldsymbol{\Psi}}; \boldsymbol{y}) + \Delta\boldsymbol{V}, \tag{20}$$

where $\Delta\boldsymbol{V} = \{\boldsymbol{I}_d - \boldsymbol{J}(\hat{\boldsymbol{\Psi}})\}^{-1}\boldsymbol{J}(\hat{\boldsymbol{\Psi}})\mathcal{I}_c^{-1}(\hat{\boldsymbol{\Psi}}; \boldsymbol{y})$ and $\mathcal{I}_c(\hat{\boldsymbol{\Psi}}; \boldsymbol{y})$ is the conditional expected complete-data information matrix, and where $\boldsymbol{I}_d$ denotes the $d \times d$ identity matrix. Thus the diagonal elements of $\Delta\boldsymbol{V}$ give the increases in the

asymptotic variances of the components of $\hat{\boldsymbol{\Psi}}$ due to missing data. For a wide class of problems where the complete-data density is from the regular exponential family, the evaluation of $\mathcal{I}_c(\hat{\boldsymbol{\Psi}}; \boldsymbol{y})$ is readily facilitated by standard complete-data computations (McLachlan and Krishnan, 1997, Section 4.5). The calculation of $\boldsymbol{J}(\hat{\boldsymbol{\Psi}})$ can be readily obtained by using only EM code via numerically differentiation of $\boldsymbol{M}(\boldsymbol{\Psi})$. Let $\hat{\boldsymbol{\Psi}} = \boldsymbol{\Psi}^{(k+1)}$ where the sequence of EM iterates has been stopped according to a suitable stopping rule. Let M_i be the ith component of $\boldsymbol{M}(\boldsymbol{\Psi})$. Let $\boldsymbol{u}^{(j)}$ be a column d-vector with the jth coordinate 1 and others 0. With a possibly different EM sequence $\boldsymbol{\Psi}^{(k)}$, let r_{ij} be the (i,j)th element of $\boldsymbol{J}(\hat{\boldsymbol{\Psi}})$, we have

$$r_{ij}^{(k)} = \frac{M_i[\hat{\boldsymbol{\Psi}} + (\boldsymbol{\Psi}_j^{(k)} - \hat{\boldsymbol{\Psi}}_j \boldsymbol{u}^{(j)})] - \hat{\boldsymbol{\Psi}}_i}{\boldsymbol{\Psi}_j^{(k)} - \hat{\boldsymbol{\Psi}}_j}.$$

Use a suitable stopping rule like $|r_{ij}^{(k+1)} - r_{ij}^{(k)}| < \sqrt{\epsilon}$ to stop each of the sequences r_{ij} $(i, j = 1, 2, \ldots, d)$ and take $r_{ij}^{*} = r_{ij}^{(k+1)}$; see McLachlan and Krishnan (1997, Section 4.5).

It is important to emphasize that estimates of the covariance matrix of the MLE based on the expected or observed information matrices are guaranteed to be valid inferentially only asymptotically. In particular for mixture models, it is well known that the sample size n has to be very large before the asymptotic theory of maximum likelihood applies. A resampling approach, the bootstrap (Efron, 1979; Efron and Tibshirani, 1993), has been considered to tackle this problem. Basford et al. (1997) compared the bootstrap and information-based approaches for some normal mixture models and found that unless the sample size was very large, the standard errors obtained by an information-based approach were too unstable to be recommended.

The bootstrap is a powerful technique that permits the variability in a random quantity to be assessed using just the data at hand. Standard error estimation of $\hat{\boldsymbol{\Psi}}$ may be implemented according to the bootstrap as follows. Further discussion on bootstrap and resampling methods can be found in Chapters III.2 and III.3 of this handbook.

1. A new set of data, $\boldsymbol{y}^*$, called the bootstrap sample, is generated according to $\hat{F}$, an estimate of the distribution function of $\boldsymbol{Y}$ formed from the original observed data $\boldsymbol{y}$. That is, in the case where $\boldsymbol{y}$ contains the observed values of a random sample of size n, $\boldsymbol{y}^*$ consists of the observed values of the random sample
$$\boldsymbol{Y}_1^*, \ldots, \boldsymbol{Y}_n^* \overset{\text{i.i.d.}}{\sim} \hat{F},$$
where the estimate $\hat{F}$ (now denoting the distribution function of a single observation $\boldsymbol{Y}_j$) is held fixed at its observed value.
2. The EM algorithm is applied to the bootstrap observed data $\boldsymbol{y}^*$ to compute the MLE for this data set, $\hat{\boldsymbol{\Psi}}^*$.

3. The bootstrap covariance matrix of $\hat{\boldsymbol{\Psi}}^*$ is given by

$$\mathrm{Cov}^*(\hat{\boldsymbol{\Psi}}^*) = E^*[\{\hat{\boldsymbol{\Psi}}^* - E^*(\hat{\boldsymbol{\Psi}}^*)\}\{\hat{\boldsymbol{\Psi}}^* - E^*(\hat{\boldsymbol{\Psi}}^*)\}^\top], \tag{21}$$

where E^* denotes expectation over the bootstrap distribution specified by $\hat{F}$.

The bootstrap covariance matrix can be approximated by Monte Carlo methods. Steps 1 and 2 are repeated independently a number of times (say, B) to give B independent realizations of $\hat{\boldsymbol{\Psi}}^*$, denoted by $\hat{\boldsymbol{\Psi}}^*_1, \ldots, \hat{\boldsymbol{\Psi}}^*_B$. Then (21) can be approximated by the sample covariance matrix of these B bootstrap replications to give

$$\mathrm{Cov}^*(\hat{\boldsymbol{\Psi}}^*) \approx \sum_{b=1}^{B} (\hat{\boldsymbol{\Psi}}^*_b - \overline{\hat{\boldsymbol{\Psi}}}^*)(\hat{\boldsymbol{\Psi}}^*_b - \overline{\hat{\boldsymbol{\Psi}}}^*)^\top / (B-1), \tag{22}$$

where $\overline{\hat{\boldsymbol{\Psi}}}^* = \sum_{b=1}^{B} \hat{\boldsymbol{\Psi}}^*_b / B$. The standard error of the ith element of $\hat{\boldsymbol{\Psi}}$ can be estimated by the positive square root of the ith diagonal element of (22). It has been shown that 50 to 100 bootstrap replications are generally sufficient for standard error estimation (Efron and Tibshirani, 1993).

In Step 1 above, the nonparametric version of the bootstrap would take $\hat{F}$ to be the empirical distribution function formed from the observed data $\boldsymbol{y}$. Situations where we may wish to use the latter include problems where the observed data are censored or are missing in the conventional sense. In these cases the use of the nonparametric bootstrap avoids having to postulate a suitable model for the underlying mechanism that controls the censorship or the absence of the data. A generalization of the nonparametric version of the bootstrap, known as the weighted bootstrap, has been studied by Newton and Raftery (1994).

4 Variations on the EM Algorithm

In this section, further modifications and extensions to the EM algorithm are considered. In general, there are extensions of the EM algorithm

1. to produce standard errors of the MLE using the EM;
2. to surmount problems of difficult E-step and/or M-step computations;
3. to tackle problems of slow convergence;
4. in the direction of Bayesian or regularized or penalized ML estimations.

We have already discussed methods like the SEM algorithm for producing standard errors of EM-computed MLE in Section 3.5. The modification of the EM algorithm for Bayesian inference will be discussed in Section 5.1. In this section, we shall focus on the problems of complicated E- or M-steps and of slow convergence of the EM algorithm.

4.1 Complicated E-step

In some applications of the EM algorithm, the E-step is complex and does not admit a close-form solution to the Q-function. In this case, the E-step at the $(k+1)$th iteration may be executed by a Monte Carlo (MC) process:

1. Make M independent draws of the missing values $\boldsymbol{Z}$, $\boldsymbol{z}^{(1_k)}, \ldots, \boldsymbol{z}^{(M_k)}$, from the conditional distribution $k(\boldsymbol{z}|\boldsymbol{y}; \boldsymbol{\Psi}^{(k)})$.
2. Approximate the Q-function as

$$Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)}) \approx Q_M(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)}) = \frac{1}{M} \sum_{m=1}^{M} \log k(\boldsymbol{\Psi}|\boldsymbol{z}^{(m_k)}; \boldsymbol{y}).$$

In the M-step, the Q-function is maximized over $\boldsymbol{\Psi}$ to obtain $\boldsymbol{\Psi}^{(k+1)}$. The variant is known as the Monte Carlo EM (MCEM) algorithm (Wei and Tanner, 1990). As MC error is introduced at the E-step, the monotonicity property is lost. But in certain cases, the algorithm gets close to a maximizer with a high probability (Booth and Hobert, 1999). The problems of specifying M and monitoring convergence are of central importance in the routine use of the algorithm. Wei and Tanner (1990) recommend small values of M be used in initial stages and be increased as the algorithm moves closer to convergence. As to monitoring convergence, they recommend that the values of $\boldsymbol{\Psi}^{(k)}$ be plotted against k and when convergence is indicated by the stabilization of the process with random fluctuations about $\hat{\boldsymbol{\Psi}}$, the process may be terminated or continued with a larger value of M. Alternative schemes for specifying M and stopping rule are considered by Booth and Hobert (1999) and McCulloch (1997).

4.1.1 Example 4: Generalized Linear Mixed Models

Generalized linear mixed models (GLMM) are extensions of generalized linear models (GLM) (McCullagh and Nelder, 1989) that incorporate random effects in the linear predictor of the GLM (more material on the GLM can be found in Chapter III.7). We let $\boldsymbol{y} = (y_1, \ldots, y_n)^\top$ denote the observed data vector. Conditional on the unobservable random effects vector, $\boldsymbol{u} = (u_1, \ldots, u_q)^\top$, we assume that $\boldsymbol{y}$ arise from a GLM. The conditional mean $\mu_j = E(y_j|\boldsymbol{u})$ is related to the linear predictor $\eta_j = \boldsymbol{x}_j^\top \boldsymbol{\beta} + \boldsymbol{z}_j^\top \boldsymbol{u}$ by the link function $g(\mu_j) = \eta_j$ $(j = 1, \ldots, n)$, where $\boldsymbol{\beta}$ is a p-vector of fixed effects and $\boldsymbol{x}_j$ and $\boldsymbol{z}_j$ are, respectively, p-vector and q-vector of explanatory variables associated with the fixed and random effects. This formulation encompasses the modeling of data involving multiple sources of random error, such as repeated measures within subjects and clustered data collected from some experimental units (Breslow and Clayton, 1993).

We let the distribution for $\boldsymbol{u}$ be $g(\boldsymbol{u}; \boldsymbol{D})$ that depends on parameters $\boldsymbol{D}$. The observed data $\boldsymbol{y}$ are conditionally independent with density functions of the form

$$f(y_j|\boldsymbol{u};\,\boldsymbol{\beta},\kappa) = \exp[m_j\kappa^{-1}\{\theta_j y_j - b(\theta_j)\} + c(y_j;\,\kappa)], \tag{23}$$

where θ_j is the canonical parameter, κ is the dispersion parameter, and m_j is the known prior weight. The conditional mean and canonical parameters are related through the equation $\mu_j = b'(\theta_j)$, where the prime denotes differentiation with respect to θ_j. Let $\boldsymbol{\Psi}$ denotes the vector of unknown parameters within $\boldsymbol{\beta}$, κ, and $\boldsymbol{D}$. The likelihood function for $\boldsymbol{\Psi}$ is given by

$$L(\boldsymbol{\Psi}) = \int \prod_{j=1}^{n} f(y_j|\boldsymbol{u};\,\boldsymbol{\beta},\kappa) g(\boldsymbol{u};\boldsymbol{D}) d\boldsymbol{u}, \tag{24}$$

which cannot usually be evaluated in closed form and has an intractable integral whose dimension depends on the structure of the random effects.

Within the EM framework, the random effects are considered as missing data. The complete data is then $\boldsymbol{x} = (\boldsymbol{y}^\top, \boldsymbol{u}^\top)^\top$ and the complete-data log likelihood is given by

$$\log L_c(\boldsymbol{\Psi}) = \sum_{j=1}^{n} \log f(y_j|\boldsymbol{u};\,\boldsymbol{\beta},\kappa) + \log g(\boldsymbol{u};\boldsymbol{D}). \tag{25}$$

On the $(k+1)$th iteration of the EM algorithm, the E-step involves the computation of the Q-function, $Q(\boldsymbol{\Psi};\,\boldsymbol{\Psi}^{(k)}) = E_{\boldsymbol{\Psi}^{(k)}}\{\log L_c(\boldsymbol{\Psi})|\boldsymbol{y}\}$, where the expectation is with respect to the conditional distribution of $\boldsymbol{u}|\boldsymbol{y}$ with current parameter value $\boldsymbol{\Psi}^{(k)}$. As this conditional distribution involves the (marginal) likelihood function $L(\boldsymbol{\Psi})$ given in (24), an analytical evaluation of the Q-function for the model (23) will be impossible outside the normal theory mixed model (Booth and Hobert, 1999). The MCEM algorithm can be adopted to tackle this problem by replacing the expectation in the E-step with a MC approximation. Let $\boldsymbol{u}^{(1_k)}, \ldots, \boldsymbol{u}^{(M_k)}$ denote a random sample from $k(\boldsymbol{u}|\boldsymbol{y};\,\boldsymbol{\Psi}^{(k)})$ at the $(k+1)$th iteration. A MC approximation of the Q-function is given by

$$Q_M(\boldsymbol{\Psi};\,\boldsymbol{\Psi}^{(k)}) = \frac{1}{M}\sum_{m=1}^{M}\{\log f(\boldsymbol{y}|\boldsymbol{u}^{(m_k)};\,\boldsymbol{\beta},\kappa) + \log g(\boldsymbol{u}^{(m_k)};\boldsymbol{D})\}. \tag{26}$$

From (26), it can be seen that the first term of the approximated Q-function involves only parameters $\boldsymbol{\beta}$ and κ, while the second term involves only $\boldsymbol{D}$. Thus, the maximization in the MC M-step is usually relatively simple within the GLMM context (McCulloch, 1997).

Alternative simulation schemes for $\boldsymbol{u}$ can be used for (26). For example, Booth and Hobert (1999) proposed the rejection sampling and a multivariate t importance sampling approximations. McCulloch (1997) considered dependent MC samples using MC Newton-Raphson (MCNR) algorithm.

4.2 Complicated M-step

One of major reasons for the popularity of the EM algorithm is that the M-step involves only complete-data ML estimation, which is often computationally

simple. But if the complete-data ML estimation is rather complicated, then the EM algorithm is less attractive. In many cases, however, complete-data ML estimation is relatively simple if maximization process on the M-step is undertaken conditional on some functions of the parameters under estimation. To this end, Meng and Rubin (1993) introduce a class of GEM algorithms, which they call the Expectation–Conditional Maximization (ECM) algorithm.

4.2.1 ECM and Multicycle ECM Algorithms

The ECM algorithm takes advantage of the simplicity of complete-data conditional maximization by replacing a complicated M-step of the EM algorithm with several computationally simpler conditional maximization (CM) steps. Each of these CM-steps maximizes the Q-function found in the preceding E-step subject to constraints on $\boldsymbol{\Psi}$, where the collection of all constraints is such that the maximization is over the full parameter space of $\boldsymbol{\Psi}$.

A CM-step might be in closed form or it might itself require iteration, but because the CM maximizations are over smaller dimensional spaces, often they are simpler, faster, and more stable than the corresponding full maximizations called for on the M-step of the EM algorithm, especially when iteration is required. The ECM algorithm typically converges more slowly than the EM in terms of number of iterations, but can be faster in total computer time. More importantly, the ECM algorithm preserves the appealing convergence properties of the EM algorithm, such as its monotone convergence.

We suppose that the M-step is replaced by $S > 1$ steps and let $\boldsymbol{\Psi}^{(k+s/S)}$ denote the value of $\boldsymbol{\Psi}$ on the sth CM-step of the $(k+1)$th iteration. In many applications of the ECM algorithm, the S CM-steps correspond to the situation where the parameter vector $\boldsymbol{\Psi}$ is partitioned into S subvectors,

$$\boldsymbol{\Psi} = (\boldsymbol{\Psi}_1^\top, \ldots, \boldsymbol{\Psi}_S^\top)^\top.$$

The sth CM-step then requires the maximization of the Q-function with respect to the sth subvector $\boldsymbol{\Psi}_s$ with the other $(S-1)$ subvectors held fixed at their current values. The convergence properties and the rate of convergence of the ECM algorithm have been discussed in Meng (1994) and Meng and Rubin (1993); see also the discussion in Sexton and Swensen (2000). In particular, it can be shown that

$$Q(\boldsymbol{\Psi}^{(k+1)}; \boldsymbol{\Psi}^{(k)}) \geq Q(\boldsymbol{\Psi}^{(k+(S-1)/S)}; \boldsymbol{\Psi}^{(k)}) \geq \ldots, \geq Q(\boldsymbol{\Psi}^{(k)}; \Psi^{(k)}). \quad (27)$$

This shows that the ECM algorithm is a GEM algorithm and so possesses its desirable convergence properties. As noted in Section 2.3, the inequality (27) is a sufficient condition for

$$L(\boldsymbol{\Psi}^{(k+1)}) \geq L(\boldsymbol{\Psi}^{(k)}) \quad (28)$$

to hold.

In many cases, the computation of an E-step may be much cheaper than the computation of the CM-steps. Hence one might wish to perform one E-step before each CM-step. A cycle is defined to be one E-step followed by one CM-step. The corresponding algorithm is called the multicycle ECM (Meng and Rubin, 1993). A multicycle ECM may not necessarily be a GEM algorithm; that is, the inequality (27) may not be hold. However, it is not difficult to show that the multicycle ECM algorithm monotonically increases the likelihood function $L(\boldsymbol{\Psi})$ after each cycle, and hence, after each iteration. The convergence results of the ECM algorithm apply to a multicycle version of it. An obvious disadvantage of using a multicycle ECM algorithm is the extra computation at each iteration. Intuitively, as a tradeoff, one might expect it to result in larger increases in the log likelihood function per iteration since the Q-function is being updated more often (Meng, 1994; Meng and Rubin, 1993).

4.2.2 Example 5: Single-Factor Analysis Model

Factor analysis is commonly used for explaining data, in particular, correlations between variables in multivariate observations and for dimensionality reduction. In a typical factor analysis model, each observation $\boldsymbol{Y}_j$ is modeled as

$$\boldsymbol{Y}_j = \boldsymbol{\mu} + \boldsymbol{B}\boldsymbol{U}_j + \boldsymbol{e}_j \quad (j = 1, \ldots, n), \tag{29}$$

where $\boldsymbol{U}_j$ is a q-dimensional ($q < p$) vector of latent or unobservable variables called factors and $\boldsymbol{B}$ is a $p \times q$ matrix of factor loadings (parameters). The $\boldsymbol{U}_j$ are assumed to be i.i.d. as $N(\boldsymbol{O}, \boldsymbol{I}_q)$, independently of the errors $\boldsymbol{e}_j$, which are assumed to be i.i.d. as $N(\boldsymbol{O}, \boldsymbol{D})$, where $\boldsymbol{D} = \mathrm{diag}\,(\sigma_1^2, \ldots, \sigma_p^2)$, and where $\boldsymbol{I}_q$ denotes the $q \times q$ identity matrix. Thus, conditional on $\boldsymbol{U}_j = \boldsymbol{u}_j$, the Y_j are independently distributed as $N(\boldsymbol{\mu} + \boldsymbol{B}\,\boldsymbol{u}_j, \boldsymbol{D})$. Unconditionally, the $\boldsymbol{Y}_j$ are i.i.d. according to a normal distribution with mean $\boldsymbol{\mu}$ and covariance matrix

$$\boldsymbol{\Sigma} = \boldsymbol{B}\boldsymbol{B}^\top + \boldsymbol{D}. \tag{30}$$

If q is chosen sufficiently smaller than p, representation (30) imposes some constraints on $\boldsymbol{\Sigma}$ and thus reduces the number of free parameters to be estimated. Note that in the case of $q > 1$, there is an infinity of choices for $\boldsymbol{B}$, since (30) is still satisfied if $\boldsymbol{B}$ is replaced by $\boldsymbol{B}\boldsymbol{C}$, where $\boldsymbol{C}$ is any orthogonal matrix of order q. As $\frac{1}{2}q(q-1)$ constraints are needed for $\boldsymbol{B}$ to be uniquely defined, the number of free parameters is $pq + p - \frac{1}{2}q(q-1)$; see Lawley and Maxwell (1971, Chapter 1) and McLachlan et al. (2003).

The factor analysis model (29) can be fitted by the EM algorithm and its variants. The MLE of the mean $\boldsymbol{\mu}$ is obviously the sample mean $\boldsymbol{\mu}$ of the n observed values $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n$ corresponding to the random sample $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n$. Hence in the sequel, $\boldsymbol{\mu}$ can be set equal to $\boldsymbol{\mu}$ without loss of generality. The log likelihood for $\boldsymbol{\Psi}$ that can be formed from the observed data $\boldsymbol{y} = (\boldsymbol{y}_1^\top, \ldots, \boldsymbol{y}_n^\top)^\top$ is, apart from an additive constant,

$$\log L(\boldsymbol{\Psi}) = -\tfrac{1}{2}n\{\log \mid \boldsymbol{B}\boldsymbol{B}^\top + \boldsymbol{D} \mid + \sum_{j=1}^{m}(\boldsymbol{y}_j - \boldsymbol{\mu})^\top(\boldsymbol{B}\boldsymbol{B}^\top + \boldsymbol{D})^{-1}(\boldsymbol{y}_j - \boldsymbol{\mu})\}.$$

We follow Dempster et al. (1977) and formulate $\boldsymbol{x} = (\boldsymbol{y}^\top, \boldsymbol{u}_1^\top, \ldots, \boldsymbol{u}_n^\top)^\top$ as the complete-data vector, where $\boldsymbol{u}_j$ corresponds to $\boldsymbol{U}_j$. Thus, the complete-data log likelihood is, but for an additive constant,

$$\log L_{\rm c}(\boldsymbol{\Psi}) = -\tfrac{1}{2}n\log \mid \boldsymbol{D} \mid -\tfrac{1}{2}\sum_{j=1}^{n}\{(\boldsymbol{y}_j - \boldsymbol{\mu} - \boldsymbol{B}\boldsymbol{u}_j)^\top \boldsymbol{D}^{-1}(\boldsymbol{y}_j - \boldsymbol{\mu} - \boldsymbol{B}\boldsymbol{u}_j) + \boldsymbol{u}_j^\top \boldsymbol{u}_j\}.$$

The complete-data density belongs to the exponential family, and the complete-data sufficient statistics are $\boldsymbol{C}_{yy}, \boldsymbol{C}_{yu}$, and $\boldsymbol{C}_{uu}$, where

$$\boldsymbol{C}_{yy} = \sum_{j=1}^{n}(\boldsymbol{y}_j - \boldsymbol{\mu})(\boldsymbol{y}_j - \boldsymbol{\mu})^\top;\ \boldsymbol{C}_{yu} = \sum_{j=1}^{n}(\boldsymbol{y}_j - \boldsymbol{\mu})\boldsymbol{u}_j^\top;\ \boldsymbol{C}_{uu} = \sum_{j=1}^{n}\boldsymbol{u}_j\boldsymbol{u}_j^\top.$$

On the $(k+1)$th iteration of the EM algorithm, we have

E-Step: Compute the conditional expectation of the sufficient statistics given $\boldsymbol{y}$ and the current fit $\boldsymbol{\Psi}^{(k)}$ for $\boldsymbol{\Psi}$:

$$E_{\boldsymbol{\Psi}^{(k)}}(\boldsymbol{C}_{yy} \mid \boldsymbol{y}) = \boldsymbol{C}_{yy}, \quad E_{\boldsymbol{\Psi}^{(k)}}(\boldsymbol{C}_{yu} \mid \boldsymbol{y}) = \boldsymbol{C}_{yy}\boldsymbol{\gamma}^{(k)},$$

and

$$E_{\boldsymbol{\Psi}^{(k)}}(\boldsymbol{C}_{uu} \mid \boldsymbol{y}) = \boldsymbol{\gamma}^{(k)^\top}\boldsymbol{C}_{yy}\boldsymbol{\gamma}^{(k)} + n\boldsymbol{\omega}^{(k)},$$

where

$$\boldsymbol{\gamma}^{(k)} = \{\boldsymbol{B}^{(k)}\boldsymbol{B}^{(k)^\top} + \boldsymbol{D}^{(k)}\}^{-1}\boldsymbol{B}^{(k)} \text{ and } \boldsymbol{\omega}^{(k)} = \boldsymbol{I}_q - \boldsymbol{\gamma}^{(k)^\top}\boldsymbol{B}^{(k)}.$$

M-Step: Calculate $\boldsymbol{B}^{(k+1)} = \boldsymbol{C}_{yy}\boldsymbol{\gamma}^{(k)}(\boldsymbol{\gamma}^{(k)^\top}\boldsymbol{C}_{yy}\boldsymbol{\gamma}^{(k)} + n\boldsymbol{\omega}^{(k)})^{-1}$ and

$$\begin{aligned}\boldsymbol{D}^{(k+1)} &= \operatorname{diag}\{\boldsymbol{C}_{yy}/n - \boldsymbol{B}^{(k+1)}\boldsymbol{H}^{(k)}\boldsymbol{B}^{(k+1)^\top}\} \\ &= n^{-1}\operatorname{diag}\{\boldsymbol{C}_{yy} - \boldsymbol{C}_{yy}\boldsymbol{\gamma}^{(k)}\boldsymbol{B}^{(k+1)^\top}\},\end{aligned} \tag{31}$$

where

$$\boldsymbol{H}^{(k)} = E_{\boldsymbol{\Psi}^{(k)}}(\boldsymbol{C}_{uu} \mid \boldsymbol{y})/n = \boldsymbol{\gamma}^{(k)^\top}\boldsymbol{C}_{yy}\boldsymbol{\gamma}^{(k)}/n + \boldsymbol{\omega}^{(k)}. \tag{32}$$

It is noted that direct differentiation of $\log L(\boldsymbol{\Psi})$ shows that the ML estimate of the diagonal matrix $\boldsymbol{D}$ satisfies

$$\hat{\boldsymbol{D}} = \operatorname{diag}\{\boldsymbol{C}_{yy}/n - \hat{\boldsymbol{B}}\hat{\boldsymbol{B}}^\top\}. \tag{33}$$

As remarked by Lawley and Maxwell (1971, pp. 30), (33) looks temptingly simple to use to solve for $\hat{\boldsymbol{D}}$, but was not recommended due to convergence

problems. On comparing (33) with (31), it can be seen that with the calculation of the ML estimate of $\boldsymbol{D}$ directly from $\log L(\boldsymbol{\Psi})$, the unconditional expectation of $\boldsymbol{U}_j\boldsymbol{U}_j^\top$, which is the identity matrix, is used in place of the conditional expectation in (32) on the E-step. Although the EM algorithm is numerically stable, the rate of convergence is slow, which can be attributed to the typically large fraction of missing data. Liu and Rubin (1994, 1998) have considered the application of the ECME algorithm to this problem; see Section 4.3.1 for the description of the algorithm. The M-step is replaced by two CM-steps. On the first CM-step, $\boldsymbol{B}^{(k+1)}$ is calculated as on the M-step above, while on the second CM-step the diagonal matrix $\boldsymbol{D}^{(k+1)}$ is obtained by using an algorithm such as Newton-Raphson to maximize the actual log likelihood with $\boldsymbol{B}$ fixed at $\boldsymbol{B}^{(k+1)}$.

The single-factor analysis model provides only a global linear model for the representation of the data in a lower-dimensional subspace, the scope of its application is limited. A global nonlinear approach can be obtained by postulating a mixture of linear submodels for the distribution of the full observation vector $\boldsymbol{Y}_j$ given the (unobservable) factors $\boldsymbol{u}_j$ (McLachlan et al., 2003). This mixture of factor analyzers has been adopted both (a) for model-based density estimation from high-dimensional data, and hence for the clutering of such data, and (b) for local dimensionality reduction; see for example McLachlan and Peel (2000, Chapter 8). Some more material on dimension reduction methods can be found in Chapter III.6 of this handbook.

4.3 Speeding up Convergence

Several suggestions are available in the literature for speeding up convergence, some of a general kind and some problem-specific; see for example McLachlan and Krishnan (1997, Chapter 4). Most of them are based on standard numerical analytic methods and suggest a hybrid of EM with methods based on Aitken acceleration, over-relaxation, line searches, Newton methods, conjugate gradients, etc. Unfortunately, the general behaviour of these hybrids is not always clear and they may not yield monotonic increases in the log likelihood over iterations. There are also methods that approach the problem of speeding up convergence in terms of "efficient" data augmentation scheme (Meng and van Dyk, 1997). Since the convergence rate of the EM algorithm increases with the proportion of observed information in the prescribed EM framework (Section 2.4), the basic idea of the scheme is to search for an efficient way of augmenting the observed data. By efficient, they mean less augmentation of the observed data (greater speed of convergence) while maintaining the simplicity and stability of the EM algorithm. A common trade-off is that the resulting E- and/or M-steps may be made appreciably more difficult to implement. To this end, Meng and van Dyk (1997) introduce a working parameter in their specification of the complete data to index a class of possible schemes to facilitate the search.

4.3.1 ECME, AECM, and PX–EM Algorithms

Liu and Rubin (1994, 1998) present an extension of the ECM algorithm called the ECME (expectation–conditional maximization either) algorithm. Here the "either" refers to the fact that with this extension, each CM-step either maximizes the Q-function or the actual (incomplete-data) log likelihood function $\log L(\boldsymbol{\Psi})$, subject to the same constraints on $\boldsymbol{\Psi}$. The latter choice should lead to faster convergence as no augmentation is involved. Typically, the ECME algorithm is more tedious to code than the ECM algorithm, but the reward of faster convergence is often worthwhile especially because it allows convergence to be more easily assessed.

A further extension of the EM algorithm, called the Space-Alternating Generalized EM (SAGE), has been proposed by Fessler and Hero (1994), where they update sequentially small subsets of parameters using appropriately smaller complete data spaces. This approach is eminently suitable for situations like image reconstruction where the parameters are large in number. Meng and van Dyk (1997) combined the ECME and SAGE algorithms. The so-called Alternating ECM (AECM) algorithm allows the data augmentation scheme to vary where necessary over the CM-steps, within and between iterations. With this flexible data augmentation and model reduction schemes, the amount of data augmentation decreases and hence efficient computations are achieved.

In contrast to the AECM algorithm where the optimal value of the working parameter is determined before EM iterations, a variant is considered by Liu et al. (1998) which maximizes the complete-data log likelihood as a function of the working parameter within each EM iteration. The so-called parameter-expanded EM (PX–EM) algorithm has been used for fast stable computation of MLE in a wide range of models. This variant has been further developed, known as the one-step-late PX–EM algorithm, to compute MAP or maximum penalized likelihood (MPL) estimates (van Dyk and Tang, 2003). Analogous convergence results hold for the ECME, AECM, and PX–EM algorithms as for the EM and ECM algorithms. More importantly, these algorithms preserve the monotone convergence of the EM algorithm as stated in (28).

4.3.2 Extensions to the EM for Data Mining Applications

With the computer revolution, massively huge data sets of millions of multidimensional observations are now commonplace. There is an ever increasing demand on speeding up the convergence of the EM algorithm to large databases. But at the same time, it is highly desirable if its simplicity and stability can be preserved. In applications where the M-step is computationally simple, for example, in fitting mutivariate normal mixtures, the rate of convergence of the EM algorithm depends mainly on the computation time of an E-step as each data point is visited at each E-step. There have been some promising developments on modifications to the EM algorithm for the

ML fitting of mixture models to large databases that preserve the simplicity of implementation of the EM in its standard form.

Neal and Hinton (1998) proposed the incremental EM (IEM) algorithm to improve the convergence rate of the EM algorithm. With this algorithm, the available n observations are divided into B ($B \leq n$) blocks and the E-step is implemented for only a block of data at a time before performing a M-step. A "scan" of the IEM algorithm thus consists of B partial E-steps and B M-steps. The argument for improved rate of convergence is that the algorithm exploits new information more quickly rather than waiting for a complete scan of the data before parameters are updated by an M-step. Another method suggested by Neal and Hinton (1998) is the sparse EM (SPEM) algorithm. In fitting a mixture model to a data set by ML via the EM, the current estimates of some posterior probabilities $\tau_{ij}^{(k)}$ for a given data point $\boldsymbol{y}_j$ are often close to zero. For example, if $\tau_{ij}^{(k)} < 0.005$ for the first two components of a four-component mixture being fitted, then with the SPEM algorithm we would fix $\tau_{ij}^{(k)}$ (i=1,2) for membership of $\boldsymbol{y}_j$ with respect to the first two components at their current values and only update $\tau_{ij}^{(k)}$ (i=3,4) for the last two components. This sparse E-step will take time proportional to the number of components that needed to be updated. A sparse version of the IEM algorithm (SPIEM) can be formulated by combining the partial E-step and the sparse E-step. With these versions, the likelihood is still increased after each scan. Ng and McLachlan (2003a) study the relative performances of these algorithms with various number of blocks B for the fitting of normal mixtures. They propose to choose B to be that factor of n that is the closest to $B^* = \text{round}(n^{2/5})$ for unrestricted component-covariance matrices, where round(r) rounds r to the nearest integer.

Other approaches for speeding up the EM algorithm for mixtures have been considered in Bradley et al. (1998) and Moore (1999). The former developed a scalable version of the EM algorithm to handle very large databases with a limited memory buffer. It is based on identifying regions of the data that are compressible and regions that must be maintained in memory. Moore (1999) has made use of multiresolution kd-trees (mrkd-trees) to speed up the fitting process of the EM algorithm on normal mixtures. Here kd stands for k-dimensional where, in our notation, $k = p$, the dimension of an observation $\boldsymbol{y}_j$. His approach builds a multiresolution data structure to summarize the database at all resolutions of interest simultaneously. The mrkd-tree is a binary tree that recursively splits the whole set of data points into partitions. The contribution of all the data points in a tree node to the sufficient statistics is simplified by calculating at the mean of these data points to save time. Ng and McLachlan (2003b) combined the IEM algorithm with the mrkd-tree approach to further speed up the EM algorithm. They also studied the convergence properties of this modified version and the relative performance with some other variants of the EM algorithm for speeding up the convergence for the fitting of normal mixtures.

Neither the scalable EM algorithm nor the *mrk*d-tree approach guarantee the desirable reliable convergence properties of the EM algorithm. Moreover, the scalable EM algorithm becomes less efficient when the number of components g is large, and the *mrk*d-trees-based algorithms slow down as the dimension p increases; see for example Ng and McLachlan (2003b) and the references therein. Further discussion on data mining applications can be found in Chapter III.13 of this handbook.

5 Miscellaneous Topics on the EM Algorithm

5.1 EM Algorithm for MAP Estimation

Although we have focussed on the application of the EM algorithm for computing MLEs in a frequentist framework, it can be equally applied to find the mode of the posterior distribution in a Bayesian framework. This problem is analogous to MLE and hence the EM algorithm and its variants can be adapted to compute MAP estimates. The computation of the MAP estimate in a Bayesian framework via the EM algorithm corresponds to the consideration of some prior density for $\boldsymbol{\Psi}$. The E-step is effectively the same as for the computation of the MLE of $\boldsymbol{\Psi}$ in a frequentist framework, requiring the calculation of the Q-function. The M-step differs in that the objective function for the maximization process is equal to the Q-function, augmented by the log prior density. The combination of prior and sample information provides a posterior distribution of the parameter on which the estimation is based.

The advent of inexpensive high speed computers and the simultaneous rapid development in posterior simulation techniques such as Markov chain Monte Carlo (MCMC) methods (Gelfand and Smith, 1990) enable Bayesian estimation to be undertaken. In particular, posterior quantities of interest can be approximated through the use of MCMC methods such as the Gibbs sampler. Such methods allow the construction of an ergodic Markov chain with stationary distribution equal to the posterior distribution of the parameter of interest. A detailed description of the MCMC technology can be found in Chapter II.3.

Although the application of MCMC methods is now routine, there are some difficulties that have to be addressed with the Bayesian approach, particularly in the context of mixture models. One main hindrance is that improper priors yield improper posterior distributions. Another hindrance is that when the number of components g is unknown, the parameter space is simultaneously ill-defined and of infinite dimension. This prevents the use of classical testing procedures and priors (McLachlan and Peel, 2000, Chapter 4). A fully Bayesian approach with g taken to be an unknown parameter has been considered by Richardson and Green (1997). Their MCMC methods allow *jumps* to be made for variable dimension parameters and thus can handle g being unspecified. A further hindrance is the effect of label switching, which arises

when there is no real prior information that allows one to discriminate between the components of a mixture model belonging to the same parametric family. This effect is very important when the solution is being calculated iteratively and there is the possibility that the labels of the components may be switched on different iterations (McLachlan and Peel, 2000, Chapter 4).

5.2 Iterative Simulation Algorithms

In computing Bayesian solutions to incomplete-data problems, iterative simulation techniques have been adopted to find the MAP estimates or estimating the entire posterior density. These iterative simulation techniques are conceptually similar to the EM algorithm, simply replacing the E- and M-steps by draws from the current conditional distribution of the missing data and $\boldsymbol{\Psi}$, respectively. However, in some methods such as the MCEM algorithm described in Section 4.1, only the E-step is so implemented. Many of these methods can be interpreted as iterative simulation analogs of the various versions of the EM and its extensions. Some examples are Stochastic EM, Data Augmentation algorithm, and MCMC methods such as the Gibbs sampler (McLachlan and Krishnan, 1997, Chapter 6). Here, we give a very brief outline of the Gibbs sampler; see also Chapter II.3 of this handbook and the references therein.

The Gibbs sampler is extensively used in many Bayesian problems where the joint distribution is too complicated to handle, but the conditional distributions are often easy enough to draw from; see Casella and George (1992). On the Gibbs sampler, an approximate sample from $p(\boldsymbol{\Psi} \mid \boldsymbol{y})$ is obtained by simulating directly from the (full) conditional distribution of a subvector of $\boldsymbol{\Psi}$ given all the other parameters in $\boldsymbol{\Psi}$ and $\boldsymbol{y}$. We write $\boldsymbol{\Psi} = (\boldsymbol{\Psi}_1, \ldots, \boldsymbol{\Psi}_d)$ in component form, a d-dimensional Gibbs sampler makes a Markov transition from $\boldsymbol{\Psi}^{(k)}$ to $\boldsymbol{\Psi}^{(k+1)}$ via d successive simulations as follows:

(1) Draw $\Psi_1^{(k+1)}$ from $p(\Psi_1 \mid \boldsymbol{y}; \Psi_2^{(k)}, \ldots, \Psi_d^{(k)})$.
(2) Draw $\Psi_2^{(k+1)}$ from $p(\Psi_2 \mid \boldsymbol{y}; \Psi_1^{(k+1)}, \Psi_3^{(k)} \ldots, \Psi_d^{(k)})$.
$\vdots$ $\vdots$ $\vdots$
(d) Draw $\Psi_d^{(k+1)}$ from $p(\Psi_d \mid \boldsymbol{y}; \Psi_1^{(k+1)}, \ldots, \Psi_{d-1}^{(k+1)})$.

The vector sequence $\{\boldsymbol{\Psi}^{(k)}\}$ thus generated is known to be a realization of a homogeneous Markov Chain. Many interesting properties of such a Markov sequence have been established, including geometric convergence, as $k \to \infty$; to a unique stationary distribution that is the posterior density $p(\Psi_1^{(k)}, \ldots, \Psi_d^{(k)} \mid \boldsymbol{y})$ under certain conditions; see Roberts and Polson (1994). Among other sampling methods, there is the Metropolis-Hastings algorithm (Hastings, 1970), which, in contrast to the Gibbs sampler, accepts the candidate simulated component in $\boldsymbol{\Psi}$ with some defined probability (McLachlan and Peel, 2000, Chapter 4).

The Gibbs sampler and other such iterative simulation techniques being Bayesian in their point of view consider both parameters and missing values

as random variables and both are subjected to random draw operations. In the iterative algorithms under a frequentist framework, like the EM-type algorithms, parameters are subjected to a maximization operation and missing values are subjected to an averaging operation. Thus the various versions of the Gibbs sampler can be viewed as stochastic analogs of the EM, ECM, and ECME algorithms. Besides these connections, the EM-type algorithms also come in useful as starting points for iterative simulation algorithms where typically regions of high density are not known *a priori* (McLachlan and Krishnan, 1997, Section 6.7.3). The relationship between the EM algorithm and the Gibbs sampler and the connection between their convergence properties have been examined in Sahu and Roberts (1999).

5.3 Further Applications of the EM Algorithm

Since the publication of Dempster et al. (1977), the number, variety, and range of applications of the EM algorithm and its extensions have been tremendous. Applications in many different contexts can be found in monographs Little and Rubin (2002), McLachlan and Krishnan (1997), and McLachlan and Peel (2000). We conclude the chapter with a quick summary of some of the more interesting and topical applications of the EM algorithm.

5.3.1 Bioinformatics: Mixture of factor analyzers

The analysis of gene expression microarray data using clustering techniques has an important role to play in the discovery, validation, and understanding of various classes of cancer; see for example Alon et al. (1999) and van't Veer et al. (2002). Clustering algorithms can be applied to the problem of clustering genes and tumour tissues (McLachlan et al., 2002) and also in the discovery of motif patterns in DNA sequences (Bailey and Elkan, 1995); see also Chapter IV.3 for the description of biomolecular sequences and structures. The EM algorithm and its variants have been applied to tackle some of the problems arisen in such applications. For example, the clustering of tumour tissues on the basis of genes expression is a nonstandard cluster analysis problem since the dimension of each tissue sample is so much greater than the number of tissues. In McLachlan et al. (2002), mixture of factor analyzers is adopted to reduce effectively the dimension of the feature space of genes. The AECM algorithm (Meng and van Dyk, 1997) can be used to fit the mixture of factor analyzers by ML (McLachlan and Peel, 2000, Chapter 8).

5.3.2 Image analysis: Hidden Markov models

In image analysis, the observed data $\boldsymbol{y}_j$ refers to intensities measured on n pixels in a scene, the associated component indicator vectors $\boldsymbol{z}_j$ will not be independently distributed as the intensities between neighboring pixels are

spatially correlated. The set of hidden states $\boldsymbol{z}_j$ is viewed as missing data (McLachlan and Peel, 2000, Chapter 13; van Dyk and Meng, 2001) and a stationary Markovian model over a finite state space is generally formulated for the distribution of the hidden variable $\boldsymbol{Z}$. In one dimension, this Markovian model is a Markov chain, and in two and higher dimensions a Markov random field (MRF) (Besag, 1986).

The use of the EM algorithm in a hidden Markov chain, known in the Hidden Markov model literature as the Baum-Welch algorithm (Baum et al., 1970), has been formulated long before Dempster et al. (1977). Also, Robert et al. (1993) consider a stochastic Bayesian approach to parameter estimation for a hidden Markov chain. Lystig and Hughes (2002) provide a means of implementing a NR approach to obtain parameter estimates and an exact computation of the observed information matrix for hidden Markov models.

The EM algorithm for the hidden MRF is considerably more difficult; see McLachlan (1992, Chapter 13) and the references therein. Even in the exponential family case (see Section 2.1) the E- and M- steps are difficult to perform even by numerical methods, except in some very simple cases like a one-parameter case; in some cases they may be implemented by suitable Gibbs sampler algorithms. A variety of practical procedures has been considered in the literature. They are reviewed by Qian and Titterington (1992), who also suggest a Monte Carlo restoration-estimation algorithm. An approximation to the E-step, based on a fractional weight version of Besag's iterated conditional modes (ICM) algorithm (Besag, 1986), has been adopted for the segmentation of magnetic resonance images (McLachlan and Peel, 2000, Section 13.4). An alternative approach is a Bayesian one, where the likelihood can be regularized using a prior, resulting in a better-conditioned log likelihood. This can also be interpreted as a penalized likelihood approach. Random field models such as Gibbs priors are often used in this context to capture the local smooth structures of the images (Geman and Geman, 1984).

References

Alon, U., Barkai, N., Notterman, D.A. et al. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences USA*, 96:6745–6750.

Bailey, T.L. and Elkan, C. (1995). Unsupervised learning of multiple motifs in biopolymers using expectation maximization. *Machine Learning*, 21:51–80.

Baker, S.G. (1992). A simple method for computing the observed information matrix when using the EM algorithm with categorical data. *Journal of Computational and Graphical Statistics*, 1:63–76.

Basford, K.E., Greenway, D.R., McLachlan, G.J., and Peel, D. (1997). Standard errors of fitted means under normal mixture models. *Computational Statistics*, 12:1–17.

Baum, L.E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximisation technique occurring in the statistical analysis of probabilistic functions of Markov processes. *Annals of Mathematical Statistics*, 41:164–171.

Besag, J. (1986). On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society, Series B*, 48:259–302.

Booth, J.G. and Hobert, J.P. (1999). Maximizing generalized linear mixed model likelihoods with an automated Monte Carlo EM algorithm. *Journal of the Royal Statistical Society, Series B*, 61:265–285.

Bradley, P.S., Fayyad, U.M., and Reina, C.A. (1998). Scaling EM (expectation-maximization) clustering to large databases. Technical Report No. MSR-TR-98-35 (revised February, 1999), Microsoft Research, Seattle.

Breslow, N.E. and Clayton, D.G. (1993). Approximate inference in generalized linear mixed models. *Journal of the American Statistical Association*, 88:9–25.

Casella, G. and George, E.I. (1992). Explaining the Gibbs sampler. *American Statistician*, 46:167–174.

Cramér, H. (1946). *Mathematical Methods of Statistics*. Princeton University Press, Princeton, New Jersey.

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

Efron, B. (1979). Bootstrap methods: another look at the jackknife. *Annals of Statistics*, 7:1–26.

Efron, B. and Tibshirani, R. (1993). *An Introduction to the Bootstrap*. Chapman & Hall, London.

Fessler, J.A. and Hero, A.O. (1994). Space-alternating generalized expectation-maximization algorithm. *IEEE Transactions on Signal Processing*, 42:2664–2677.

Flury, B. and Zoppé, A. (2000). Exercises in EM. *American Statistician*, 54:207–209.

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

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

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

Jamshidian, M. and Jennrich, R.I. (2000). Standard errors for EM estimation. *Journal of the Royal Statistical Society, Series B*, 62:257–270.

Lawley, D.N. and Maxwell, A.E. (1971). *Factor Analysis as a Statistical Method*. Butterworths, London, second edition.

Little, R.J.A. and Rubin, D.B. (2002). *Statistical Analysis with Missing Data*. Wiley, New York, second edition.

Liu, C. and Rubin, D.B. (1994). The ECME algorithm: a simple extension of EM and ECM with faster monotone convergence. *Biometrika*, 81:633–648.

Liu, C. and Rubin, D.B. (1998). Maximum likelihood estimation of factor analysis using the ECME algorithm with complete and incomplete data. *Statistica Sinica*, 8:729–747.

Liu, C., Rubin, D.B., and Wu, Y.N. (1998). Parameter expansion to accelerate EM: the PX–EM algorithm. *Biometrika*, 85:755–770.

Louis, T.A. (1982). Finding the observed information matrix when using the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 44:226–233.

Lystig, T.C. and Hughes, J.P. (2002). Exact computation of the observed information matrix for hidden Markov models. *Journal of Computational and Graphical Statistics*, 11:678–689.

McCullagh, P.A. and Nelder, J. (1989). *Generalized Linear Models*. Chapman & Hall, London, second edition.

McCulloch, C.E. (1997). Maximum likelihood algorithms for generalized linear mixed models. *Journal of the American Statistical Association*, 92:162–170.

McLachlan, G.J. (1992). *Discriminant Analysis and Statistical Pattern Recognition*. Wiley, New York.

McLachlan, G.J. and Basford, K.E. (1988). *Mixture Models: Inference and Applications to Clustering*. Marcel Dekker, New York.

McLachlan, G.J., Bean, R.W., and Peel, D. (2002). A mixture model-based approach to the clustering of microarray expression data. *Bioinformatics*, 18:413–422.

McLachlan, G.J. and Krishnan, T. (1997). *The EM Algorithm and Extensions*. Wiley, New York.

McLachlan, G.J. and Peel, D. (2000). *Finite Mixture Models*. Wiley, New York.

McLachlan, G.J., Peel, D., and Bean, R.W. (2003). Modelling high-dimensional data by mixtures of factor analyzers. *Computational Statistics & Data Analysis*, 41:379–388.

Meilijson, I. (1989). A fast improvement of the EM algorithm in its own terms. *Journal of the Royal Statistical Society, Series B*, 51:127–138.

Meng, X.L. (1994). On the rate of convergence of the ECM algorithm. *Annals of Statistics*, 22:326–339.

Meng, X.L. and Rubin, D.B. (1991). Using EM to obtain asymptotic variance-covariance matrices: the SEM algorithm. *Journal of the American Statistical Association*, 86:899–909.

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 van Dyk, D. (1997). The EM algorithm – an old folk song sung to a fast new tune. *Journal of the Royal Statistical Society, Series B*, 59:511–567.

Moore, A.W. (1999). Very fast EM-based mixture model clustering using multiresolution kd-trees. In Kearns, M.S., Solla, S.A., and Cohn, D.A., editors, *Advances in Neural Information Processing Systems 11*, pages 543–549. MIT Press, MA.

Neal, R.M. and Hinton, G.E. (1998). A view of the EM algorithm that justifies incremental, sparse, and other variants. In Jordan, M.I., editor, *Learning in Graphical Models*, pages 355–368. Kluwer, Dordrecht.

Nettleton, D. (1999). Convergence properties of the EM algorithm in constrained parameter spaces. *Canadian Journal of Statistics*, 27:639–648.

Newton, M.A. and Raftery, A.E. (1994). Approximate Bayesian inference with the weighted likelihood bootstrap. *Journal of the Royal Statistical Society, Series B*, 56:3–48.

Ng, S.K. and McLachlan, G.J. (2003a). On the choice of the number of blocks with the incremental EM algorithm for the fitting of normal mixtures. *Statistics and Computing*, 13:45–55.

Ng, S.K. and McLachlan, G.J. (2003b). On some variants of the EM algorithm for the fitting of finite mixture models. *Austrian Journal of Statistics*, 32:143–161.

Qian, W. and Titterington, D.M. (1992). Stochastic relaxations and EM algorithms for Markov random fields. *Journal of Statistical Computation and Simulation*, 40:55–69.

Richardson, S. and Green, P.J. (1997). On Bayesian analysis of mixtures with an unknown number of components. *Journal of the Royal Statistical Society, Series B*, 59:731–792 (correction (1998), pp. 661).

Robert, C.P., Celeux, G., and Diebolt, J. (1993). Bayesian estimation of hidden Markov chains: A stochastic implementation. *Statistics & Probability Letters*, 16:77–83.

Roberts, G.O. and Polson, N.G. (1994). On the geometric convergence of the Gibbs sampler. *Journal of the Royal Statistical Society, Series B*, 56:377–384.

Sahu, S.K. and Roberts, G.O. (1999). On convergence of the EM algorithm and the Gibbs sampler. *Statistics and Computing*, 9:55–64.

Sexton, J. and Swensen, A.R. (2000). ECM algorithms that converge at the rate of EM. *Biometrika*, 87:651–662.

Ueda, N. and Nakano, R. (1998). Deterministic annealing EM algorithm. *Neural Networks*, 11:271–282.

van Dyk, D.A. and Meng, X.L. (2001). The art of data augmentation. *Journal of Computational and Graphical Statistics*, 10:1–111.

van Dyk, D.A. and Tang, R. (2003). The one-step-late PXEM algorithm. *Statistics and Computing*, 13:137–152.

van't Veer, L.J., Dai, H., van de Vijver, M.J. et al. (2002). Gene expression profiling predicts clinical outcome of breast cancer. *Nature*, 415:530–536.

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. *Journal of the American Statistical Association*, 85:699–704.

Wright, K. and Kennedy, W.J. (2000). An interval analysis approach to the EM algorithm. *Journal of Computational and Graphical Statistics*, 9:303–318.

Wu, C.F.J. (1983). On the convergence properties of the EM algorithm. *Annals of Statistics*, 11:95–103.

Index