Chapter 5

EM

Geoffrey J. McLachlan and Shu-Kay Ng

Contents

Abstract 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. In particular, the EM algorithm simplifies considerably the problem of fitting finite mixture models by ML, where mixture models are used to model heterogeneity in cluster analysis and pattern recognition contexts. The EM algorithm has a number of appealing properties, including its numerical stability, simplicity of implementation, and reliable global convergence. There are also extensions of the EM algorithm to tackle complex problems in various data mining applications. It is, however, highly desirable if its simplicity and stability can be preserved.

5.1 Introduction

The expectation-maximization (EM) algorithm has been of considerable interest in recent years in the development of algorithms in various application areas such as data mining, machine learning, and pattern recognition [20, 27, 28]. The seminal paper of Dempster et al. [8] on the EM algorithm greatly stimulated interest in the use of finite mixture distributions to model heterogeneous data. This is because the fitting of

© 2009 by Taylor & Francis Group, LLC

mixture models by maximum likelihood (ML) is a classic example of a problem that is simplified considerably by the EM's conceptual unification of ML estimation from data that can be viewed as being incomplete [20]. 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 [6, 13, 31]. Finite mixture distributions provide a flexible and mathematical-based approach to the modeling and clustering of data observed on random phenomena. We focus here on the use of the EM algorithm for the fitting of finite mixture models via the ML approach.

With the mixture model-based approach to clustering, the observed p-dimensional data $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n$ are assumed to have come from a mixture of an initially specified number g of component densities in some unknown proportions $\pi_1, \ldots, \pi_g$, which sum to 1. The mixture density of $\boldsymbol{y}_j$ is expressed as

$$f(\boldsymbol{y}_j; \boldsymbol{\Psi}) = \sum_{i=1}^{g} \pi_i f_i(\boldsymbol{y}_j; \boldsymbol{\theta}_i) \qquad (j = 1, \ldots, n) \tag{5.1}$$

where the component density $f_i(\boldsymbol{y}_j; \boldsymbol{\theta}_i)$ is specified up to a vector $\boldsymbol{\theta}_i$ of unknown parameters ($i = 1, \ldots, g$). The vector of all the unknown parameters is given by

$$\boldsymbol{\Psi} = \left(\pi_1, \ldots, \pi_{g-1}, \boldsymbol{\theta}_1^T, \ldots, \boldsymbol{\theta}_g^T\right)^T$$

where the superscript T denotes vector transpose. The parameter vector $\boldsymbol{\Psi}$ can be estimated by ML. The objective is to maximize the likelihood $L(\boldsymbol{\Psi})$, or equivalently, the log likelihood $\log L(\boldsymbol{\Psi})$, as a function of $\boldsymbol{\Psi}$, over the parameter space. That is, the ML estimate of $\boldsymbol{\Psi}$, $\hat{\boldsymbol{\Psi}}$, is given by an appropriate root of the log likelihood equation,

$$\partial \log L(\boldsymbol{\Psi}) / \partial \boldsymbol{\Psi} = \boldsymbol{0} \tag{5.2}$$

where

$$\log L(\boldsymbol{\Psi}) = \sum_{j=1}^{n} \log f(\boldsymbol{y}_j; \boldsymbol{\Psi})$$

is the log likelihood function for $\boldsymbol{\Psi}$ formed under the assumption of independent data $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n$. The aim of ML estimation [13] is to determine an estimate $\hat{\boldsymbol{\Psi}}$ for each n, so that it defines a sequence of roots of Equation (5.2) that is consistent and asymptotically efficient. Such a sequence is known to exist under suitable regularity conditions [7]. With probability tending to one, these roots correspond to local maxima in the interior of the parameter space. For estimation models in general, the likelihood usually has a global maximum in the interior of the parameter space. Then typically a sequence of roots of Equation (5.2) with the desired asymptotic properties is provided by taking $\hat{\boldsymbol{\Psi}}$ for each n to be the root that globally maximizes $L(\boldsymbol{\Psi})$; in this case, $\hat{\boldsymbol{\Psi}}$ is the MLE [18]. We shall henceforth refer to $\hat{\boldsymbol{\Psi}}$ as the MLE, even in situations where it may not globally maximize the likelihood. Indeed, in the example on mixture models to be presented in Section 5.4.1, the likelihood is unbounded. However, there may still exist under the usual regularity conditions a sequence of roots of Equation (5.2) with the properties of consistency, efficiency, and asymptotic normality [16].

© 2009 by Taylor & Francis Group, LLC

5.2 Algorithm Description

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 [18]. Within the incomplete-data framework of the EM algorithm, we let $\boldsymbol{y} = (\boldsymbol{y}_1^T, \ldots, \boldsymbol{y}_n^T)^T$ denote the vector containing the observed data and we let $\boldsymbol{z}$ denote the vector containing the incomplete data. The complete-data vector is declared to be

$$\boldsymbol{x} = (\boldsymbol{y}^T, \boldsymbol{z}^T)^T$$

The EM algorithm approaches the problem of solving the "incomplete-data" log likelihood Equation (5.2) indirectly by proceeding iteratively in terms of the "complete-data" log likelihood, $\log L_c(\boldsymbol{\Psi})$. As it depends explicitly on the unobservable data $\boldsymbol{z}$, the E-step is performed on which $\log L_c(\boldsymbol{\Psi})$ is replaced by the so-called Q-function, which is its conditional expectation given $\boldsymbol{y}$, using the current fit for $\boldsymbol{\Psi}$. More specifically, on the $(k+1)$th iteration of the EM algorithm, the E-step computes

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

where $E_{\boldsymbol{\Psi}^{(k)}}$ denotes expectation using the parameter vector $\boldsymbol{\Psi}^{(k)}$. The M-step updates the estimate of $\boldsymbol{\Psi}$ by that value $\boldsymbol{\Psi}^{(k+1)}$ of $\boldsymbol{\Psi}$ that maximizes the Q-function, $Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)})$, with respect to $\boldsymbol{\Psi}$ over the parameter space [18]. The E- and M-steps are alternated repeatedly until the changes in the log likelihood values are less than some specified threshold. As mentioned in Section 5.1, the EM algorithm is numerically stable with each EM iteration increasing the likelihood value as

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

It can be shown that both the E- and M-steps will have particularly simple forms when the complete-data probability density function is from an exponential family [18]. 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, a generalized EM (GEM) algorithm [8] may be adopted for which the M-step requires $\boldsymbol{\Psi}^{(k+1)}$ to be chosen such that $\boldsymbol{\Psi}^{(k+1)}$ increases the Q-function $Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)})$ over its value at $\boldsymbol{\Psi} = \boldsymbol{\Psi}^{(k)}$. That is,

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

holds; see [18].

Some of the drawbacks of the EM algorithm are (a) it does not automatically produce an estimate of the covariance matrix of the parameter estimates. This disadvantage, however, can easily be removed by using appropriate methodology associated with the EM algorithm [18]; (b) it is sometimes very slow to converge; and (c) in some problems, the E- or M-steps may be analytically intractable. We shall briefly address the last two issues in Section 5.5.

© 2009 by Taylor & Francis Group, LLC

5.3 Software Implementation

The EMMIX program: McLachlan et al. [22] have developed the program EMMIX as a general tool to fit mixtures of multivariate normal or t-distributed components by ML via the EM algorithm to continuous multivariate data. It also includes many other features that were found to be of use when fitting mixture models. These include the provision of starting values for the application of the EM algorithm, the provision of standard errors for the fitted parameters in the mixture model via various methods, and the determination of the number of components; see below.

Starting values for EM algorithm: With applications where the log 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. In the context of finite mixture models, an initial parameter value can be obtained using the k-means clustering algorithm, hierarchical clustering methods, or random partitions of the data [20]. With the EMMIX program, there is an additional option for random starts whereby the user can first subsample the data before using a random start based on the subsample each time. This is to limit the effect of the central limit theorem, which would have the randomly selected starts being similar for each component in large samples [20].

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 ML estimates; see [11, 15, 18]. Alternatively, standard error estimation may be obtained with the EMMIX program using the bootstrap resampling approach implemented parametrically or nonparametrically [18, 20].

Number of components: We can make a choice as to an appropriate value of the number of components (clusters) g by consideration of the likelihood function. In the absence of any prior information as to the number of clusters present in the data, we can monitor the increase in log likelihood function as the value of g increases. At any stage, the choice of $g = g_0$ versus $g = g_0 + 1$ can be made by either performing the likelihood ratio test or using some information-based criterion, such as the Bayesian Information Criterion (BIC). Unfortunately, regularity conditions do not hold for the likelihood ratio test statistic λ to have its usual null distribution of chi-squared with degrees of freedom equal to the difference d in the number of parameters for $g = g_0+1$ and $g = g_0$ components in the mixture model. The EMMIX program provides a bootstrap resampling approach to assess the null distribution (and hence the p-value) of the statistic $(-2\log\lambda)$. Alternatively, one can apply BIC, although regularity conditions do not hold for its validity here. The use of BIC leads to the selection of $g = g_0+1$ over $g = g_0$ if $-2\log\lambda$ is greater than $d\log(n)$.

© 2009 by Taylor & Francis Group, LLC

Other mixture software: There are some other EM-based software for mixture modeling via ML. For example, Fraley and Raftery [9] have developed the MCLUST program for hierarchical clustering on the basis of mixtures of normal components under various parameterizations of the component-covariance matrices. It is interfaced to the S-PLUS commercial software and has the option to include an additional component in the model for background (Poisson) noise. The reader is referred to the appendix in McLachlan and Peel [20] for the availability of software for the fitting of mixture models.

5.4 Illustrative Examples

We give in this section two examples to demonstrate how the EM algorithm can be conveniently applied to find the ML estimates in some commonly occurring situations in data mining. Both examples concern the application of the EM algorithm for the ML estimation of finite mixture models, which is widely adopted to model heterogeneous data [20]. They illustrate how an incomplete-data formulation is used to derive the EM algorithm for computing ML estimates.

5.4.1 Example 5.1: Multivariate Normal Mixtures

This example concerns the application of the EM algorithm for the ML estimation of finite mixture models with multivariate normal components [20]. With reference to Equation (5.1), the mixture density of $\boldsymbol{y}_j$ is given by

$$f(\boldsymbol{y}_j; \boldsymbol{\Psi}) = \sum_{i=1}^{g} \pi_i \phi(\boldsymbol{y}_j; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \quad (j = 1, \ldots, n) \tag{5.3}$$

where $\phi(\boldsymbol{y}_j; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ denotes the p-dimensional multivariate normal distribution with mean $\boldsymbol{\mu}_i$ and covariance matrix $\boldsymbol{\Sigma}_i$. Here the vector $\boldsymbol{\Psi}$ of unknown parameters consists of the mixing proportions $\pi_1, \ldots, \pi_{g-1}$, the elements of the component means $\boldsymbol{\mu}_i$, and the distinct elements of the component-covariance matrices $\boldsymbol{\Sigma}_i$. The log likelihood for $\boldsymbol{\Psi}$ is then given by

$$\log L(\boldsymbol{\Psi}) = \sum_{j=1}^{n} \log \left\{ \sum_{i=1}^{g} \pi_i \phi(\boldsymbol{y}_j; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i) \right\}$$

Solutions of the log likelihood equation corresponding to local maxima can be found iteratively by application of the EM algorithm.

Within the EM framework, each $\boldsymbol{y}_j$ is conceptualized to have arisen from one of the g components of the mixture model [Equation (5.3)]. We let $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n$ denote the unobservable component-indicator vectors, where the ith element z_{ij} of $\boldsymbol{z}_j$ is taken to be one or zero according as the jth observation $\boldsymbol{y}_j$ does or does not come

© 2009 by Taylor & Francis Group, LLC

from the ith component. The observed-data vector $\boldsymbol{y}$ is viewed as being incomplete, as the associated component-indicator vectors, $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n$, are not available. The complete-data vector is therefore $\boldsymbol{x} = (\boldsymbol{y}^T, \boldsymbol{z}^T)^T$, where $\boldsymbol{z} = (\boldsymbol{z}_1^T, \ldots, \boldsymbol{z}_n^T)^T$. The complete-data log likelihood for $\boldsymbol{\Psi}$ is given by

$$\log L_c(\boldsymbol{\Psi}) = \sum_{i=1}^{g} \sum_{j=1}^{n} z_{ij} \{\log \pi_i + \log \phi(\boldsymbol{y}_j; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)\} \tag{5.4}$$

The EM algorithm is applied to this problem by treating the z_{ij} in Equation (5.4) as missing data. On the $(k+1)$th iteration, the E-step computes the Q-function, $Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)})$, which is the conditional expectation of the complete-data log likelihood given $\boldsymbol{y}$ and the current estimates $\boldsymbol{\Psi}^{(k)}$. As the complete-data log likelihood [Equation (5.4)] is linear in the missing data z_{ij}, we simply have to calculate the current conditional expectation of Z_{ij} given the observed data $\boldsymbol{y}$, where Z_{ij} is the random variable corresponding to z_{ij}. That is,

$$\begin{aligned} E_{\boldsymbol{\Psi}^{(k)}}(Z_{ij}|\boldsymbol{y}) &= \mathrm{pr}_{\boldsymbol{\Psi}^{(k)}}\{Z_{ij} = 1|\boldsymbol{y}\} \\ &= \tau_i(\boldsymbol{y}_j; \boldsymbol{\Psi}^{(k)}) \\ &= \pi_i^{(k)} \phi\big(\boldsymbol{y}_j; \boldsymbol{\mu}_i^{(k)}, \boldsymbol{\Sigma}_i^{(k)}\big) \Big/ \sum_{h=1}^{g} \pi_h^{(k)} \phi\big(\boldsymbol{y}_j; \boldsymbol{\mu}_h^{(k)}, \boldsymbol{\Sigma}_h^{(k)}\big) \end{aligned} \tag{5.5}$$

for $i = 1, \ldots, g$; $j = 1, \ldots, n$. The quantity $\tau_i(\boldsymbol{y}_j; \boldsymbol{\Psi}^{(k)})$ is the posterior probability that the jth observation $\boldsymbol{y}_j$ belongs to the ith component of the mixture. From Equations (5.4) and (5.5), it follows that

$$Q(\boldsymbol{\Psi}; \boldsymbol{\Psi}^{(k)}) = \sum_{i=1}^{g} \sum_{j=1}^{n} \tau_i(\boldsymbol{y}_j; \boldsymbol{\Psi}^{(k)}) \{\log \pi_i + \log \phi(\boldsymbol{y}_j; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)\} \tag{5.6}$$

For mixtures with normal component densities, it is computationally advantageous to work in terms of the sufficient statistics [26] given by

$$\begin{aligned} T_{i1}^{(k)} &= \sum_{j=1}^{n} \tau_i(\boldsymbol{y}_j; \boldsymbol{\Psi}^{(k)}) \\ \boldsymbol{T}_{i2}^{(k)} &= \sum_{j=1}^{n} \tau_i(\boldsymbol{y}_j; \boldsymbol{\Psi}^{(k)}) \boldsymbol{y}_j \\ \boldsymbol{T}_{i3}^{(k)} &= \sum_{j=1}^{n} \tau_i(\boldsymbol{y}_j; \boldsymbol{\Psi}^{(k)}) \boldsymbol{y}_j \boldsymbol{y}_j^T \end{aligned} \tag{5.7}$$

For normal components, the M-step exists in closed form and is simplified on the basis of the sufficient statistics in Equation (5.7) as

$$\begin{aligned} \pi_i^{(k+1)} &= T_{i1}^{(k)}/n \\ \boldsymbol{\mu}_i^{(k+1)} &= \boldsymbol{T}_{i2}^{(k)}/T_{i1}^{(k)} \\ \boldsymbol{\Sigma}_i^{(k+1)} &= \big\{\boldsymbol{T}_{i3}^{(k)} - T_{i1}^{(k)^{-1}} \boldsymbol{T}_{i2}^{(k)} \boldsymbol{T}_{i2}^{(k)^T}\big\}/T_{i1}^{(k)} \end{aligned} \tag{5.8}$$

© 2009 by Taylor & Francis Group, LLC

see [20, 26]. In the case of unrestricted component-covariance matrices $\boldsymbol{\Sigma}_i$, $L(\boldsymbol{\Psi})$ is unbounded, as each data point gives rise to a singularity on the edge of the parameter space [16, 20]. Consideration has to be given to the problem of relatively large (spurious) local maxima that occur as a consequence of a fitted component having a very small (but nonzero) generalized variance (the determinant of the covariance matrix). Such a component corresponds to a cluster containing a few data points either relatively close together or almost lying in a lower dimensional subspace in the case of multivariate data.

In practice, the component-covariance matrices $\boldsymbol{\Sigma}_i$ can be restricted to being the same, $\boldsymbol{\Sigma}_i = \boldsymbol{\Sigma}$ $(i = 1, \ldots, g)$, where $\boldsymbol{\Sigma}$ is unspecified. In this case of homoscedastic normal components, the updated estimate of the common component-covariance matrix $\boldsymbol{\Sigma}$ is given by

$$\boldsymbol{\Sigma}^{(k+1)} = \sum_{i=1}^{g} T_{i1}^{(k)} \boldsymbol{\Sigma}_i^{(k+1)} / n \tag{5.9}$$

where $\boldsymbol{\Sigma}_i^{(k+1)}$ is given by Equation (5.8), and the updates of π_i and $\boldsymbol{\mu}_i$ are as above in the heteroscedastic case [Equation (5.8)].

The well-known set of *Iris* data is available at the UCI Repository of machine learning databases [1]. The data consist of measurements of the length and width of both sepals and petals of 50 plants for each of the three types of *Iris* species *setosa*, *versicolor*, and *virginica*. Here, we cluster these four-dimensional data, ignoring the known classification of the data, by fitting a mixture of $g = 3$ normal components with heteroscedastic diagonal component-covariance matrices using the EMMIX program [22]. The vector of unknown parameters $\boldsymbol{\Psi}$ now consists of the mixing proportions π_1, π_2, the elements of the component means $\boldsymbol{\mu}_i$, and the diagonal elements of the component-covariance matrices $\boldsymbol{\Sigma}_i$ $(i = 1, 2, 3)$. An initial value $\boldsymbol{\Psi}^{(0)}$ is chosen to be

$$\begin{aligned}
&\pi_1^{(0)} = 0.31,\ \pi_2^{(0)} = 0.33,\ \pi_3^{(0)} = 0.36 \\
&\boldsymbol{\mu}_1^{(0)} = (5.0, 3.4, 1.5, 0.2)^T,\ \boldsymbol{\mu}_2^{(0)} = (5.8, 2.7, 4.2, 1.3)^T \\
&\qquad \boldsymbol{\mu}_3^{(0)} = (6.6, 3.0, 5.5, 2.0)^T \\
&\boldsymbol{\Sigma}_1^{(0)} = \mathrm{diag}(0.1, 0.1, 0.03, 0.01) \quad \boldsymbol{\Sigma}_2^{(0)} = \mathrm{diag}(0.2, 0.1, 0.2, 0.03) \\
&\qquad \boldsymbol{\Sigma}_3^{(0)} = \mathrm{diag}(0.3, 0.1, 0.3, 0.1)
\end{aligned}$$

which is obtained through the use of k-means clustering method. With the EMMIX program, the default stopping criterion is that the change in the log likelihood from the current iteration and the log likelihood from 10 iterations previously differs by less than 0.000001 of the current log likelihood [22]. The results of the EM algorithm are presented in Table 5.1. The MLE of $\boldsymbol{\Psi}$ can be taken to be the value of $\boldsymbol{\Psi}^{(k)}$ on iteration $k = 29$. Alternatively, the EMMIX program offers automatic starting values for the application of the EM algorithm. As an example, an initial value $\boldsymbol{\Psi}^{(0)}$ is determined from 10 random starts (using 70% subsampling of the data), 10 k-means starts, and 6 hierarchical methods; see Section 5.3 and [22]. The final estimates of $\boldsymbol{\Psi}$ are the same as those given in Table 5.1.

© 2009 by Taylor & Francis Group, LLC

TABLE 5.1 Results of the EM Algorithm for Example 5.1

Iteration	$\pi_i^{(k)}$	$\boldsymbol{\mu}_i^{(k)T}$	Diagonal Elements of $\boldsymbol{\Sigma}_i^{(k)}$	Log Likelihood
0	0.310	(5.00, 3.40, 1.50, 0.20)	(0.100, 0.100, 0.030, 0.010)	−317.98421
	0.330	(5.80, 2.70, 4.20, 1.30)	(0.200, 0.100, 0.200, 0.030)	
	0.360	(6.60, 3.00, 5.50, 2.00)	(0.300, 0.100, 0.300, 0.100)	
1	0.333	(5.01, 3.43, 1.46, 0.25)	(0.122, 0.141, 0.030, 0.011)	−306.90935
	0.299	(5.82, 2.70, 4.20, 1.30)	(0.225, 0.089, 0.212, 0.034)	
	0.368	(6.62, 3.01, 5.48, 1.98)	(0.322, 0.083, 0.325, 0.088)	
2	0.333	(5.01, 3.43, 1.46, 0.25)	(0.122, 0.141, 0.030, 0.011)	−306.87370
	0.300	(5.83, 2.70, 4.21, 1.30)	(0.226, 0.087, 0.218, 0.034)	
	0.367	(6.62, 3.01, 5.47, 1.98)	(0.323, 0.083, 0.328, 0.087)	
10	0.333	(5.01, 3.43, 1.46, 0.25)	(0.122, 0.141, 0.030, 0.011)	−306.86234
	0.303	(5.83, 2.70, 4.22, 1.30)	(0.227, 0.087, 0.224, 0.035)	
	0.364	(6.62, 3.02, 5.48, 1.99)	(0.324, 0.083, 0.328, 0.086)	
20	0.333	(5.01, 3.43, 1.46, 0.25)	(0.122, 0.141, 0.030, 0.011)	−306.86075
	0.304	(5.83, 2.70, 4.22, 1.30)	(0.228, 0.087, 0.225, 0.035)	
	0.363	(6.62, 3.02, 5.48, 1.99)	(0.324, 0.083, 0.327, 0.086)	
29	0.333	(5.01, 3.43, 1.46, 0.25)	(0.122, 0.141, 0.030, 0.011)	−306.86052
	0.305	(5.83, 2.70, 4.22, 1.30)	(0.229, 0.087, 0.225, 0.035)	
	0.362	(6.62, 3.02, 5.48, 1.99)	(0.324, 0.083, 0.327, 0.085)	

5.4.2 Example 5.2: Mixtures of Factor Analyzers

McLachlan and Peel [21] adopt a mixture of factor analyzers model to cluster the so-called wine data set, which is available at the UCI Repository of machine learning databases [1]. These data give the results of a chemical analysis of wines grown in the same region in Italy, but derived from three different cultivars. The analysis determined the quantities of $p = 13$ consituents found in each of $n = 178$ wines. To cluster this data set, a three-component normal mixture model can be adopted. However, as $p = 13$ in this problem, the (unrestricted) covariance matrix $\boldsymbol{\Sigma}_i$ has 91 parameters for each i ($i = 1, 2, 3$), which means that the total number of parameters is very large relative to the sample size of $n = 178$. A mixture of factor analyzers can be used to reduce the number of parameters to be fitted. In a mixture of factor analyzers, each observation $\boldsymbol{Y}_j$ is modeled as

$$\boldsymbol{Y}_j = \boldsymbol{\mu}_i + \boldsymbol{B}_i \boldsymbol{U}_{ij} + \boldsymbol{\epsilon}_{ij}$$

with probability π_i ($i = 1, \ldots, g$) for $j = 1, \ldots, n$, where $\boldsymbol{U}_{ij}$ is a q-dimensional ($q < p$) vector of latent or unobservable variables called *factors* and $\boldsymbol{B}_i$ is a $p \times q$ matrix of factor loadings (parameters). The factors $\boldsymbol{U}_{i1}, \ldots, \boldsymbol{U}_{in}$ are distributed independently $N(\boldsymbol{0}, \boldsymbol{I}_q)$, independently of the $\boldsymbol{\epsilon}_{ij}$, which are distributed independently

© 2009 by Taylor & Francis Group, LLC

$N(\mathbf{0}, \boldsymbol{D}_i)$, where $\boldsymbol{I}_q$ is the $q \times q$ identity matrix and $\boldsymbol{D}_i$ is a $p \times p$ diagonal matrix $(i = 1, \ldots, g)$. That is,

$$f(\boldsymbol{y}_j; \boldsymbol{\Psi}) = \sum_{i=1}^{g} \pi_i \phi(\boldsymbol{y}_j; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$$

where

$$\boldsymbol{\Sigma}_i = \boldsymbol{B}_i \boldsymbol{B}_i^T + \boldsymbol{D}_i \qquad (i = 1, \ldots, g)$$

The vector of unknown parameters $\boldsymbol{\Psi}$ now consists of the elements of the $\boldsymbol{\mu}_i$, the $\boldsymbol{B}_i$, and the $\boldsymbol{D}_i$, along with the mixing proportions π_i $(i = 1, \ldots, g-1)$.

The alternating expectation conditional-maximization (AECM) algorithm [24] can be used to fit the mixture of factor analyzers model by ML; see Section 5.5. The unknown parameters are partitioned as $(\boldsymbol{\Psi}_1^T, \boldsymbol{\Psi}_2^T)^T$, where $\boldsymbol{\Psi}_1$ contains the π_i $(i = 1, \ldots, g-1)$ and the elements of $\boldsymbol{\mu}_i$ $(i = 1, \ldots, g)$. The subvector $\boldsymbol{\Psi}_2$ contains the elements of $\boldsymbol{B}_i$ and $\boldsymbol{D}_i$ $(i = 1, \ldots, g)$. The AECM algorithm is an extension of the expectation-conditional maximization (ECM) algorithm [23], where the specification of the complete-data is allowed to be different on each conditional maximization (CM) step. In this application, one iteration consists of two cycles corresponding to the partition of $\boldsymbol{\Psi}$ into $\boldsymbol{\Psi}_1$ and $\boldsymbol{\Psi}_2$, and there is one E-step and one CM-step for each cycle. For the first cycle of the AECM algorithm, we specify the missing data to be just the component-indicator vectors, $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n$; see Equation (5.4). The E-step on the first cycle on the $(k+1)$th iteration is essentially the same as given in Equations (5.5) and (5.6). The first CM-step computes the updated estimate $\boldsymbol{\Psi}_1^{(k+1)}$ as

$$\pi_i^{(k+1)} = \sum_{j=1}^{n} \tau_{ij}^{(k)} / n$$

and

$$\boldsymbol{\mu}_i^{(k+1)} = \sum_{j=1}^{n} \tau_{ij}^{(k)} \boldsymbol{y}_j / \sum_{j=1}^{n} \tau_{ij}^{(k)}$$

for $i = 1, \ldots, g$. For the second cycle for the updating of $\boldsymbol{\Psi}_2$, we specify the missing data to be the factors $\boldsymbol{U}_{i1}, \ldots, \boldsymbol{U}_{in}$, as well as the component-indicator vectors, $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n$. On setting $\boldsymbol{\Psi}^{(k+1/2)}$ equal to $(\boldsymbol{\Psi}_1^{(k+1)^T}, \boldsymbol{\Psi}_2^{(k)^T})^T$, the E-step on the second cycle calculates the conditional expectations as

$$E_{\boldsymbol{\Psi}^{(k+1/2)}}\{Z_{ij}(\boldsymbol{U}_{ij} - \boldsymbol{\mu}_i) | \boldsymbol{y}_j\} = \tau_{ij}^{(k+1/2)} \boldsymbol{\gamma}_i^{(k)^T} (\boldsymbol{y}_j - \boldsymbol{\mu}_i)$$

and

$$\begin{aligned} &E_{\boldsymbol{\Psi}^{(k+1/2)}}\{Z_{ij}(\boldsymbol{U}_{ij} - \boldsymbol{\mu}_i)(\boldsymbol{U}_{ij} - \boldsymbol{\mu}_i)^T | \boldsymbol{y}_j\} \\ &= \tau_{ij}^{(k+1/2)} \{\boldsymbol{\gamma}_i^{(k)^T} (\boldsymbol{y}_j - \boldsymbol{\mu}_i)(\boldsymbol{y}_j - \boldsymbol{\mu}_i)^T \boldsymbol{\gamma}_i^{(k)} + \boldsymbol{\Omega}_i^{(k)}\} \end{aligned}$$

where

$$\boldsymbol{\gamma}_i^{(k)} = \left(\boldsymbol{B}_i^{(k)} \boldsymbol{B}_i^{(k)^T} + \boldsymbol{D}_i^{(k)}\right)^{-1} \boldsymbol{B}_i^{(k)}$$

© 2009 by Taylor & Francis Group, LLC

and

$$\boldsymbol{\Omega}_i^{(k)} = \boldsymbol{I}_q - \boldsymbol{\gamma}_i^{(k)T} \boldsymbol{B}_i^{(k)}$$

for $i = 1, \ldots, g$. The E-step above uses the result that the conditional distribution of $\boldsymbol{U}_{ij}$ given $\boldsymbol{y}_j$ and $z_{ij} = 1$ is given by

$$\boldsymbol{U}_{ij} | \boldsymbol{y}_j, z_{ij} = 1 \sim N\big(\boldsymbol{\gamma}_i^T (\boldsymbol{y}_j - \boldsymbol{\mu}_i),\ \boldsymbol{\Omega}_i\big)$$

for $i = 1, \ldots, g;\ j = 1, \ldots, n$. The CM-step on the second cycle provides the updated estimate $\boldsymbol{\Psi}_2^{(k+1)}$ as

$$\boldsymbol{B}_i^{(k+1)} = \boldsymbol{V}_i^{(k+1/2)} \boldsymbol{\gamma}_i^{(k)} \big(\boldsymbol{\gamma}_i^{(k)T} \boldsymbol{V}_i^{(k+1/2)} \boldsymbol{\gamma}_i^{(k)} + \boldsymbol{\Omega}_i^{(k)}\big)^{-1}$$

and

$$\boldsymbol{D}_i^{(k+1)} = \text{diag}\big\{\boldsymbol{V}_i^{(k+1/2)} - \boldsymbol{B}_i^{(k+1)} \boldsymbol{H}_i^{(k+1/2)} \boldsymbol{B}_i^{(k+1)T}\big\}$$

where

$$\boldsymbol{V}_i^{(k+1/2)} = \frac{\sum_{j=1}^n \tau_{ij}^{(k+1/2)} (\boldsymbol{y}_j - \boldsymbol{\mu}_i^{(k+1)})(\boldsymbol{y}_j - \boldsymbol{\mu}_i^{(k+1)})^T}{\sum_{j=1}^n \tau_{ij}^{(k+1/2)}}$$

and

$$\boldsymbol{H}_i^{(k+1/2)} = \boldsymbol{\gamma}_i^{(k)T} \boldsymbol{V}_i^{(k+1/2)} \boldsymbol{\gamma}_i^{(k)} + \boldsymbol{\Omega}_i^{(k)}$$

As an illustration, a mixture of factor analyzers model with different values of q is fitted to the wine data set, ignoring the known classification of the data. To determine the initial estimate of $\boldsymbol{\Psi}$, the EMMIX program is used to fit the normal mixture model with unrestricted component-covariance matrices using ten random starting values (with 70% subsampling of the data). The estimates of π_i and $\boldsymbol{\mu}_i$ so obtained are used as the initial values for π_i and $\boldsymbol{\mu}_i$ in the AECM algorithm. The estimate of $\boldsymbol{\Sigma}_i$ so obtained (denoted as $\boldsymbol{\Sigma}_i^{(0)}$) is used to determine the initial estimate of $\boldsymbol{D}_i$, where $\boldsymbol{D}_i^{(0)}$ is taken to be the diagonal matrix formed from the diagonal elements of $\boldsymbol{\Sigma}_i^{(0)}$. An initial estimate of $\boldsymbol{B}_i$ can be obtained using the method described in [20]. The results of the AECM algorithm from $q = 1$ to $q = 8$ are presented in Table 5.2. We have also reported the value of minus twice the likelihood ratio test statistic λ (i.e., twice the increase in the log likelihood), as we proceed from fitting a mixture of q factor analyzers to one with $q + 1$ component factors. For a given level of the number of components g, regularity conditions hold for the asymptotic null distribution of $-2 \log \lambda$ to be chi-squared with d degrees of freedom, where d is the difference between the number of parameters under the null and alternative hypotheses for the value of q. It can be seen from Table 5.2 that the apparent error rate of the outright clustering is smallest for $q = 2$ and 3. However, this error rate is unknown in a clustering context and so cannot be used as a guide to the choice of q. Concerning the use of the likelihood ratio test to decide on the number of factors q, the test of $q = q_0 = 6$ versus $q = q_0 + 1 = 7$ is not significant ($P = 0.28$), on taking $-2 \log \lambda$ to be chi-squared with $d = g(p - q_0) = 21$ degrees of freedom under the null hypothesis that $q = q_0 = 6$.

© 2009 by Taylor & Francis Group, LLC

TABLE 5.2 Results of the AECM Algorithm for Example 5.2

q	Log Likelihood	Error (%Error)	$-2 \log \lambda$
1	−3102.254	2 (1.12)	—
2	−2995.334	1 (0.56)	213.8
3	−2913.122	1 (0.56)	164.4
4	−2871.655	3 (1.69)	82.93
5	−2831.860	4 (2.25)	79.59
6	−2811.290	4 (2.25)	41.14
7	−2799.204	4 (2.25)	24.17
8	−2788.542	4 (2.25)	21.32

5.5 Advanced Topics

In this section, we consider some extensions of the EM algorithm to handle problems with more difficult E-step and/or M-step computations, and to tackle problems of slow convergence. Moreover, we present a brief account of the applications of the EM algorithm in the context of Hidden Markov Models (HMMs), which provide a convenient way of formulating an extension of a mixture model to allow for dependent data.

In some applications of the EM algorithm such as with generalized linear mixed models, the E-step is complex and does not admit a close-form solution to the Q-function. In this case, the E-step may be executed by a Monte Carlo (MC) process. At the $(k+1)$th iteration, the E-step involves

- simulation of M independent sets of realizations of the missing data $\boldsymbol{Z}$ from the conditional distribution $g(\boldsymbol{z}|\boldsymbol{y}; \boldsymbol{\Psi}^{(k)})$
- approximation of the Q-function by

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

  where $\boldsymbol{z}^{(m_k)}$ is the mth set of missing values based on $\boldsymbol{\Psi}^{(k)}$

In the M-step, the Q-function is maximized over $\boldsymbol{\Psi}$ to obtain $\boldsymbol{\Psi}^{(k+1)}$. This variant is known as the *Monte Carlo EM (MCEM)* algorithm [33]. As an 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 [4]. The problems of specifying M and monitoring convergence are of central importance in the routine use of the algorithm; see [4, 18, 33].

With the EM algorithm, the M-step involves only complete-data ML estimation, which is often computationally simple. However, in some applications, such as that in mixtures of factor analyzers (Section 5.4.2), the M-step is rather complicated.

© 2009 by Taylor & Francis Group, LLC

The ECM algorithm [23] is a natural extension of the EM algorithm in situations where the maximization process on the M-step is relatively simple when conditional on some function of the parameters under estimation. 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 CM steps. In particular, the ECM algorithm preserves the appealing convergence properties of the EM algorithm [18, 23]. The AECM algorithm [24] mentioned in Section 5.4.2 allows the specification of the complete-data to vary where necessary over the CM-steps within and between iterations. This flexible data augmentation and model reduction scheme is eminently suitable for applications like mixtures of factor analyzers where the parameters are large in number.

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. An incremental version of the EM algorithm was proposed by Neal and Hinton [25] to improve the rate of convergence of the EM algorithm. This incremental EM (IEM) algorithm proceeds by dividing the data into B blocks and implementing the (partial) E-step for only a block of data at a time before performing an M-step. That is, a "scan" of the IEM algorithm consists of B partial E-steps and B full M-steps [26]. It can be shown from Exercises 6 and 7 in Section 5.6 that the IEM algorithm in general converges with fewer scans and hence faster than the EM algorithm. The IEM algorithm also increases the likelihood at each scan; see the discussion in [27].

In the mixture framework with observations $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n$, the unobservable component-indicator vector $\boldsymbol{z} = (\boldsymbol{z}_1^T, \ldots, \boldsymbol{z}_n^T)^T$ can be termed as the "hidden variable." In speech recognition applications, the $\boldsymbol{z}_j$ may be unknown serially dependent prototypical spectra on which the observed speech signals $\boldsymbol{y}_j$ depend ($j = 1, \ldots, n$). Hence the sequence or set of hidden values $\boldsymbol{z}_j$ cannot be regarded as independent. In the automatic speech recognition applications or natural language processing (NLP) tasks, a stationary Markovian model over a finite state space is generally formulated for the distribution of the hidden variable $\boldsymbol{Z}$ [18]. As a consequence of the dependent structure of $\boldsymbol{Z}$, the density of $\boldsymbol{Y}_j$ will not have its simple representation [Equation (5.1)] of a mixture density as in the independence case. However, $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n$ are assumed conditionally independent given $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n$; that is

$$f(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n | \boldsymbol{z}_1, \ldots, \boldsymbol{z}_n; \boldsymbol{\theta}) = \prod_{j=1}^{n} f(\boldsymbol{y}_j | \boldsymbol{z}_j; \boldsymbol{\theta})$$

where $\boldsymbol{\theta}$ denotes the vector containing the unknown parameters in these conditional distributions that are known a priori to be distinct. The application of the EM algorithm to this problem is known as the *Baum–Welch algorithm* in the HMM literature. Baum and his collaborators formulated this algorithm before the appearance of the EM algorithm in Dempster et al. [8] and established the convergence properties for this algorithm; see [2] and the references therein. The E-step can be implemented exactly, but it does require a forward and backward recursion through the data [18]. The M-step

© 2009 by Taylor & Francis Group, LLC

can be implemented in closed form, using formulas which are a combination of the MLEs for the multinomial parameters and Markov chain transition probabilities; see [14, 30].

5.6 Exercises

Ten exercises are given in this section. They arise in various scientific fields in the contexts of data mining and pattern recognition, in which the EM algorithm or its variants have been applied. The exercises include problems where the incompleteness of the data is perhaps not as natural or evident as in the two illustrative examples in Section 5.4.

1. Böhning et al. [3] consider a cohort study on the health status of 602 preschool children from 1982 to 1985 in northest Thailand [32]. The frequencies of illness spells (fever, cough, or both) during the study period are presented in Table 5.3. A three-component mixture of Poisson distributions is fitted to the data. The log likelihood function is given by

$$\log L(\boldsymbol{\Psi}) = \sum_{j=1}^{n} \log \left\{ \sum_{i=1}^{3} \pi_i f(y_j, \theta_i) \right\}$$

where $\boldsymbol{\Psi} = (\pi_1, \pi_2, \theta_1, \theta_2, \theta_3)^T$ and

$$f(y_j, \theta_i) = \exp(-\theta_i)\theta_i^{y_j}/y_j! \qquad (i = 1, 2, 3)$$

With reference to Section 5.4.1, let

$$\tau_i(y_j; \boldsymbol{\Psi}^{(k)}) = \pi_i^{(k)} f\left(y_j, \theta_i^{(k)}\right) \Big/ \sum_{h=1}^{3} \pi_h^{(k)} f\left(y_j, \theta_h^{(k)}\right) \qquad (i = 1, 2, 3)$$

denote the posterior probability that y_j belongs to the ith component. Show that the M-step updates the estimates as

$$\pi_i^{(k+1)} = \sum_{j=1}^{n} \tau_i(y_j; \boldsymbol{\Psi}^{(k)})/n \qquad (i = 1, 2)$$

$$\theta_i^{(k+1)} = \sum_{j=1}^{n} \tau_i(y_j; \boldsymbol{\Psi}^{(k)}) y_j / \left(n\pi_i^{(k+1)}\right) \qquad (i = 1, 2, 3)$$

Using the initial estimates $\pi_1 = 0.6, \pi_2 = 0.3, \theta_1 = 2, \theta_2 = 9$, and $\theta_3 = 17$, find the MLE of $\boldsymbol{\Psi}$.

© 2009 by Taylor & Francis Group, LLC

TABLE 5.3 Frequencies of Illness Spells for a Cohort Sample of Preschool Children in Northest Thailand

No. of Illnesses	Frequency	No. of Illnesses	Frequency	No. of Illnesses	Frequency
0	120	8	25	16	6
1	64	9	19	17	5
2	69	10	18	18	1
3	72	11	18	19	3
4	54	12	13	20	1
5	35	13	4	21	2
6	36	14	3	23	1
7	25	15	6	24	2

2. The fitting of mixtures of (multivariate) t distributions was proposed by McLachlan and Peel [19] to provide a more robust approach to the fitting of normal mixture models. A g-component mixture of t distributions is given by

$$f(\boldsymbol{y}_j; \boldsymbol{\Psi}) = \sum_{i=1}^{g} \pi_i f(\boldsymbol{y}_j; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i, \nu_i)$$

where the component density $f(\boldsymbol{y}_j; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i, \nu_i)$ has a multivariate t distribution with location $\boldsymbol{\mu}_i$, positive definite inner product matrix $\boldsymbol{\Sigma}_i$, and ν_i degrees of freedom ($i = 1, \ldots, g$); see [19, 29]. The vector of unknown parameters is

$$\boldsymbol{\Psi} = (\pi_1, \ldots, \pi_{g-1}, \boldsymbol{\theta}^T, \boldsymbol{\nu}^T)^T$$

where $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_g)^T$ are the degrees of freedom for the t distributions, and $\boldsymbol{\theta} = (\boldsymbol{\theta}_1^T, \ldots, \boldsymbol{\theta}_g^T)^T$, and where $\boldsymbol{\theta}_i$ contains the elements of $\boldsymbol{\mu}_i$ and the distinct elements of $\boldsymbol{\Sigma}_i$ ($i = 1, \ldots, g$). With reference to Section 5.4.1, the observed data augmented by the component-indicator vectors $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_n$ are viewed as still being incomplete. Additional missing data, $u_1, \ldots, u_n$, are introduced into the complete-data vector, that is,

$$\boldsymbol{x} = \left(\boldsymbol{y}^T, \boldsymbol{z}_1^T, \ldots, \boldsymbol{z}_n^T, u_1, \ldots, u_n\right)^T$$

where $u_1, \ldots, u_n$ are defined so that, given $z_{ij} = 1$,

$$\boldsymbol{Y}_j | u_j, z_{ij} = 1 \sim N(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i / u_j)$$

independently for $j = 1, \ldots, n$, and

$$U_j | z_{ij} = 1 \sim \text{gamma}\left(\tfrac{1}{2}\nu_i, \tfrac{1}{2}\nu_i\right)$$

Show that the complete-data log likelihood can be written in three terms as

$$\log L_c(\boldsymbol{\Psi}) = \log L_{1c}(\boldsymbol{\pi}) + \log L_{2c}(\boldsymbol{\nu}) + \log L_{3c}(\boldsymbol{\theta}) \tag{5.10}$$

© 2009 by Taylor & Francis Group, LLC

where

$$\log L_{1c}(\boldsymbol{\pi}) = \sum_{i=1}^{g}\sum_{j=1}^{n} z_{ij}\log\pi_i$$

$$\log L_{2c}(\boldsymbol{\nu}) = \sum_{i=1}^{g}\sum_{j=1}^{n} z_{ij}\left\{-\log\Gamma(\tfrac{1}{2}\nu_i) + \tfrac{1}{2}\nu_i\log(\tfrac{1}{2}\nu_i) + \tfrac{1}{2}\nu_i(\log u_j - u_j) - \log u_j\right\}$$

and

$$\log L_{3c}(\boldsymbol{\theta}) = \sum_{i=1}^{g}\sum_{j=1}^{n} z_{ij}\left\{-\tfrac{1}{2}p\log(2\pi) - \tfrac{1}{2}\log|\boldsymbol{\Sigma}_i| - \tfrac{1}{2}u_j\delta(\boldsymbol{y}_j, \boldsymbol{\mu}_i, ; \boldsymbol{\Sigma}_i)\right\}$$

where

$$\delta(\boldsymbol{y}_j, \boldsymbol{\mu}_i; \boldsymbol{\Sigma}_i) = (\boldsymbol{y}_j - \boldsymbol{\mu}_i)^T\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{y}_j - \boldsymbol{\mu}_i)$$

3. With reference to the above mixtures of t distributions, show that the E-step on the $(k+1)$th iteration of the EM algorithm involves the calculation of

$$E_{\boldsymbol{\Psi}^{(k)}}(Z_{ij}|\boldsymbol{y}) = \tau_{ij}^{(k)} = \frac{\pi_i^{(k)} f(\boldsymbol{y}_j; \boldsymbol{\mu}_i^{(k)}, \boldsymbol{\Sigma}_i^{(k)}, \nu_i^{(k)})}{f(\boldsymbol{y}_j; \boldsymbol{\Psi}^{(k)})} \tag{5.11}$$

$$E_{\boldsymbol{\Psi}^{(k)}}(U_j|\boldsymbol{y}, z_{ij} = 1) = u_{ij}^{(k)} = \frac{\nu_i^{(k)} + p}{\nu_i^{(k)} + \delta(\boldsymbol{y}_j, \boldsymbol{\mu}_i^{(k)}; \boldsymbol{\Sigma}_i^{(k)})} \tag{5.12}$$

and

$$E_{\boldsymbol{\Psi}^{(k)}}(\log U_j|\boldsymbol{y}, z_{ij} = 1) = \log u_{ij}^{(k)} + \left\{\psi\left(\frac{\nu_i^{(k)} + p}{2}\right) - \log\left(\frac{\nu_i^{(k)} + p}{2}\right)\right\} \tag{5.13}$$

for $i = 1, \ldots, g$; $j = 1, \ldots, n$. In Equation (5.13),

$$\psi(r) = \{\partial\Gamma(r)/\partial r\}/\,\Gamma(r)$$

is the Digamma function [29]. Hint for Equation (5.12): the gamma distribution is the conjugate prior distribution for U_j; Hint for Equation (5.13): if a random variable S has a gamma(α, β) distribution, then

$$E(\log S) = \psi(\alpha) - \log\beta.$$

Also, it follows from Equation (5.10) that $\boldsymbol{\pi}^{(k+1)}$, $\boldsymbol{\theta}^{(k+1)}$, and $\boldsymbol{\nu}^{(k+1)}$ can be computed on the M-step independently of each other. Show that the updating formulas for the first two are

$$\pi_i^{(k+1)} = \sum_{j=1}^{n}\tau_{ij}^{(k)}/n$$

$$\boldsymbol{\mu}_i^{(k+1)} = \sum_{j=1}^{n}\tau_{ij}^{(k)}u_{ij}^{(k)}\boldsymbol{y}_j \Big/ \sum_{j=1}^{n}\tau_{ij}^{(k)}u_{ij}^{(k)}$$

© 2009 by Taylor & Francis Group, LLC

and

$$\boldsymbol{\Sigma}_i^{(k+1)} = \frac{\sum_{j=1}^n \tau_{ij}^{(k)} u_{ij}^{(k)} (\boldsymbol{y}_j - \boldsymbol{\mu}_i^{(k+1)})(\boldsymbol{y}_j - \boldsymbol{\mu}_i^{(k+1)})^T}{\sum_{j=1}^n \tau_{ij}^{(k)}}$$

The updates $\nu_i^{(k+1)}$ for the degrees of freedom need to be computed iteratively. It follows from Equation (5.10) that $\nu_i^{(k+1)}$ is a solution of the equation

$$\left\{ -\psi\left(\tfrac{1}{2}\nu_i\right) + \log\left(\tfrac{1}{2}\nu_i\right) + 1 + \tfrac{1}{n_i^{(k)}} \sum_{j=1}^n \tau_{ij}^{(k)} \left(\log u_{ij}^{(k)} - u_{ij}^{(k)} \right) + \psi\left(\frac{\nu_i^{(k)} + p}{2} \right) - \log\left(\frac{\nu_i^{(k)} + p}{2} \right) \right\} = 0$$

where $n_i^{(k)} = \sum_{j=1}^n \tau_{ij}^{(k)}$ $(i = 1, \ldots, g)$.

4. The EMMIX program [22] has an option for the fitting of mixtures of multivariate t components. Now fit a mixture of two t components (with unrestricted scale matrices $\boldsymbol{\Sigma}_i$ and unequal degrees of freedom ν_i) to the *Leptograpsus* crab data set of Campbell and Mahon [5]. With the crab data, one species has been split into two new species, previously grouped by color form, orange and blue. Data are available on 50 specimens of each sex of each species. Attention here is focussed on the sample of $n = 100$ five-dimensional measurements on orange crabs (the two components correspond to the males and females). Run the EMMIX program with automatic starting values from 10 random starts (using 100% subsampling of the data), 10 k-means starts, and 6 hierarchical methods (with user-supplied initial values $\nu_1^{(0)} = \nu_2^{(0)} = 13.193$ which is obtained in the case of equal scale matrices and equal degrees of freedom). Verify estimates of $\boldsymbol{\nu}$ are $\hat{\nu}_1 = 12.2$ and $\hat{\nu}_2 = 300.0$ and the numbers assigned to each component are, respectively, 47 and 53 (misclassification rate = 3%).

5. For a mixture of g component distributions of generalized linear models (GLMs) in proportions $\pi_1, \ldots, \pi_g$, the density of the jth response variable Y_j is given by

$$f(y_j; \boldsymbol{\Psi}) = \sum_{i=1}^g \pi_i f(y_j; \theta_{ij}, \kappa_i)$$

where the log density for the ith component is given by

$$\log f(y_j; \theta_{ij}, \kappa_i) = \kappa_i^{-1}\{\theta_{ij} y_j - b(\theta_{ij})\} + c(y_j; \kappa_i) \quad (i = 1, \ldots, g)$$

where θ_{ij} is the natural or canonical parameter and κ_i is the dispersion parameter. For the ith component GLM, denote μ_{ij} the conditional mean of Y_j and $\eta_{ij} = h_i(\mu_{ij}) = \boldsymbol{\beta}_i^T \boldsymbol{x}_j$ the linear predictor, where $h_i(\cdot)$ is the link function and $\boldsymbol{x}_j$ is a vector of explanatory variables on the jth response y_j [20]. The vector of unknown parameters is $\boldsymbol{\Psi} = (\pi_1, \ldots, \pi_{g-1}, \kappa_1, \ldots, \kappa_g, \boldsymbol{\beta}_1^T, \ldots, \boldsymbol{\beta}_g^T)^T$. Let z_{ij} denote the component-indicator variables as defined in Section 5.4.1. The E-step is essentially the same as given in Equations (5.5) and (5.6), with the

© 2009 by Taylor & Francis Group, LLC

component densities $\phi(\boldsymbol{y}_j; \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ replaced by $f(y_j; \theta_{ij}, \kappa_i)$. On the M-step, the updating formula for $\pi_i^{(k+1)}$ $(i = 1, \ldots, g-1)$ is

$$\pi_i^{(k+1)} = \sum_{j=1}^{n} \tau_{ij}^{(k)} / n$$

where

$$\tau_{ij}^{(k)} = \pi_i^{(k)} f\left(y_j; \theta_{ij}^{(k)}, \kappa_i^{(k)}\right) \Big/ \sum_{h=1}^{g} \pi_h^{(k)} f\left(y_j; \theta_{hj}^{(k)}, \kappa_h^{(k)}\right)$$

The updates $\kappa_i^{(k+1)}$ and $\boldsymbol{\beta}_i^{(k+1)}$ need to be computed iteratively by solving

$$\begin{aligned} \sum_{j=1}^{n} \tau_{ij}^{(k)} \partial \log f(y_j; \theta_{ij}, \kappa_i)/\partial\kappa &= 0 \\ \sum_{j=1}^{n} \tau_{ij}^{(k)} \partial \log f(y_j; \theta_{ij}, \kappa_i)/\partial\boldsymbol{\beta}_i &= \mathbf{0} \end{aligned} \tag{5.14}$$

Consider a mixture of gamma distributions, where the gamma density function for the ith component is given by

$$f(y_j; \mu_{ij}, \alpha_i) = \frac{(\frac{\alpha_i}{\mu_{ij}})^{\alpha_i} y_j^{(\alpha_i - 1)} \exp(-\frac{\alpha_i}{\mu_{ij}} y_j)}{\Gamma(\alpha_i)}$$

where $\alpha_i > 0$ is the shape parameter, which does not depend on the explanatory variables. The linear predictor is modelled via a log-link as

$$\eta_{ij} = h_i(\mu_{ij}) = \log \mu_{ij} = \boldsymbol{\beta}_i^T \boldsymbol{x}_j$$

With reference to Equation (5.14), show that the M-step for a mixture of gamma distributions involves solving the nonlinear equations

$$\sum_{j=1}^{n} \tau_{ij}^{(k)} \{1 + \log \alpha_i - \log \mu_{ij} + \log y_j - y_j/\mu_{ij} - \psi(\alpha_i)\} = 0,$$

$$\sum_{j=1}^{n} \tau_{ij}^{(k)} (-1 + y_j/\mu_{ij}) \alpha_i \boldsymbol{x}_j = \mathbf{0}$$

where $\psi(r) = \{\partial\Gamma(r)/\partial r\}/\Gamma(r)$ is the digamma function.

6. With the IEM algorithm described in Section 5.5, let $\boldsymbol{\Psi}^{(k+b/B)}$ denote the value of $\boldsymbol{\Psi}$ after the bth iteration on the $(k+1)$th scan $(b = 1, \ldots, B)$. In the context of g-component normal mixture models (Section 5.4.1), the partial E-step on the $(b+1)$th iteration of the $(k+1)$th scan replaces z_{ij} by $\tau_i(\boldsymbol{y}_j; \boldsymbol{\Psi}^{(k+b/B)})$ for those $\boldsymbol{y}_j$ in the $(b+1)$th block $(b = 0, \ldots, B-1;\ i = 1, \ldots, g)$. With reference to Equation (5.7), let $\boldsymbol{T}_{iq,b+1}^{(k+b/B)}$ denote the conditional expectations of

© 2009 by Taylor & Francis Group, LLC

the sufficient statistics for the $(b+1)$th block ($b = 0, \ldots, B-1$; $q = 1, 2, 3$). For example,

$$\boldsymbol{T}_{i1,b+1}^{(k+b/B)} = \sum_{j \in S_b} \tau_i(\boldsymbol{y}_j; \boldsymbol{\Psi}^{(k+b/B)}) \qquad (i = 1, \ldots, g)$$

where S_b is a subset of $\{1, \ldots, n\}$ containing the subscripts of those $\boldsymbol{y}_j$ that belong to the $(b+1)$th block ($b = 0, \ldots, B-1$). From Equations (5.7) and (5.8), show that the M-step on the $(b+1)$th iteration of the $(k+1)$th scan of the IEM algorithm involves the update of the estimates of π_i, $\boldsymbol{\mu}_i$, and $\boldsymbol{\Sigma}_i$ as follows:

$$\begin{aligned}
\pi_i^{(k+(b+1)/B)} &= T_{i1}^{(k+b/B)}/n \\
\boldsymbol{\mu}_i^{(k+(b+1)/B)} &= \boldsymbol{T}_{i2}^{(k+b/B)}/T_{i1}^{(k+b/B)} \\
\boldsymbol{\Sigma}_i^{(k+(b+1)/B)} &= \left\{\boldsymbol{T}_{i3}^{(k+b/B)} - T_{i1}^{(k+b/B)^{-1}} \boldsymbol{T}_{i2}^{(k+b/B)} \boldsymbol{T}_{i2}^{(k+b/B)^T}\right\}/T_{i1}^{(k+b/B)}
\end{aligned}$$

for $i = 1, \ldots, g$, where

$$\boldsymbol{T}_{iq}^{(k+b/B)} = \boldsymbol{T}_{iq}^{(k+(b-1)/B)} - \boldsymbol{T}_{iq,b+1}^{(k-1+b/B)} + \boldsymbol{T}_{iq,b+1}^{(k+b/B)} \tag{5.15}$$

for $i = 1, \ldots, g$ and $q = 1, 2, 3$. It is noted that the first and second terms on the right-hand side of Equation (5.15) are already available from the previous iteration and the previous scan, respectively. In practice, the IEM algorithm is implemented by running the standard EM algorithm for the first few scans to avoid the "premature component starvation" problem [26]. In this case, we have

$$\boldsymbol{T}_{iq}^{(k)} = \sum_{b=1}^{B} \boldsymbol{T}_{iq,b}^{(k)} \qquad (i = 1, \ldots, g;\ q = 1, 2, 3)$$

7. With the IEM algorithm, Ng and McLachlan [26] provide a simple guide for choosing the number of blocks B for normal mixtures. In the case of component-covariance matrices specified to be diagonal (such as in Example 5.1), they suggest $B \approx n^{1/3}$. For the *Iris* data in Example 5.1, it implies that $B \approx (150)^{1/3}$. Run an IEM algorithm to the *Iris* data with $B = 5$ and the same initial values of $\boldsymbol{\Psi}$ as in Example 5.1. Verify that (a) the final estimates and the log likelihood value are approximately the same as those using the EM algorithm, and (b) the IEM algorithm converges with fewer scans than the EM algorithm and increases the likelihood at each scan; see the discussion in [27].

8. Ng and McLachlan [28] apply the ECM algorithm for training the mixture of experts (ME) networks [10, 12]. In ME networks, there are several modules, referred to as expert networks. These expert networks approximate the distribution of $\boldsymbol{y}_j$ within each region of the input space. The expert network maps its input $\boldsymbol{x}_j$ to an output $\boldsymbol{y}_j$, with conditional density $f_h(\boldsymbol{y}_j|\boldsymbol{x}_j; \boldsymbol{\theta}_h)$, where $\boldsymbol{\theta}_h$ is a vector of unknown parameters for the hth expert network ($h = 1, \ldots, M$). The gating network provides a set of scalar coefficients $\pi_h(\boldsymbol{x}_j; \boldsymbol{\alpha})$ that weight the

© 2009 by Taylor & Francis Group, LLC

contributions of the various experts, where $\boldsymbol{\alpha}$ is a vector of unknown parameters in the gating network. The final output of the ME network is a weighted sum of all the output vectors produced by the expert networks,

$$f(\boldsymbol{y}_j|\boldsymbol{x}_j;\boldsymbol{\Psi}) = \sum_{h=1}^{M} \pi_h(\boldsymbol{x}_j;\boldsymbol{\alpha}) f_h(\boldsymbol{y}_j|\boldsymbol{x}_j;\boldsymbol{\theta}_h)$$

Within the incomplete-data framework of the EM algorithm, we introduce the indicator variables Z_{hj}, where z_{hj} is 1 or 0 according to whether $\boldsymbol{y}_j$ belongs or does not belong to the hth expert. Show that the complete-data log likelihood for $\boldsymbol{\Psi}$ is given by

$$\log L_c(\boldsymbol{\Psi}) = \sum_{j=1}^{n}\sum_{h=1}^{M} z_{hj}\{\log \pi_h(\boldsymbol{x}_j;\boldsymbol{\alpha}) + \log f_h(\boldsymbol{y}_j|\boldsymbol{x}_j;\boldsymbol{\theta}_h)\}$$

and the Q-function can be decomposed into two terms with respect to $\boldsymbol{\alpha}$ and $\boldsymbol{\theta}_h$ $(h = 1, \ldots, M)$, respectively, as

$$Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)}) = Q_{\boldsymbol{\alpha}} + Q_{\boldsymbol{\theta}}$$

where

$$Q_{\boldsymbol{\alpha}} = \sum_{j=1}^{n}\sum_{h=1}^{M} \tau_{hj}^{(k)} \log \pi_h(\boldsymbol{x}_j;\boldsymbol{\alpha})$$

$$Q_{\boldsymbol{\theta}} = \sum_{j=1}^{n}\sum_{h=1}^{M} \tau_{hj}^{(k)} \log f_h(\boldsymbol{y}_j|\boldsymbol{x}_j;\boldsymbol{\theta}_h)$$

and where

$$\tau_{hj}^{(k)} = \pi_h(\boldsymbol{x}_j;\boldsymbol{\alpha}^{(k)}) f_h\left(\boldsymbol{y}_j|\boldsymbol{x}_j;\boldsymbol{\theta}_h^{(k)}\right) \Big/ \sum_{r=1}^{M} \pi_r(\boldsymbol{x}_j;\boldsymbol{\alpha}^{(k)}) f_r\left(\boldsymbol{y}_j|\boldsymbol{x}_j;\boldsymbol{\theta}_r^{(k)}\right)$$

9. With the ME networks above, the output of the gating network is usually modeled by the multinomial logit (or softmax) function as

$$\pi_h(\boldsymbol{x}_j;\boldsymbol{\alpha}) = \frac{\exp(\boldsymbol{v}_h^T\boldsymbol{x}_j)}{1+\sum_{r=1}^{M-1}\exp(\boldsymbol{v}_r^T\boldsymbol{x}_j)} \qquad (h = 1, \ldots, M-1)$$

and $\pi_M(\boldsymbol{x}_j;\boldsymbol{\alpha}) = 1/(1+\sum_{r=1}^{M-1}\exp(\boldsymbol{v}_r^T\boldsymbol{x}_j))$. Here $\boldsymbol{\alpha}$ contains the elements in $\boldsymbol{v}_h$ $(h = 1, \ldots, M-1)$. Show that the updated estimate of $\boldsymbol{\alpha}^{(k+1)}$ on the M-step is obtained by solving

$$\sum_{j=1}^{n}\left(\tau_{hj}^{(k)} - \frac{\exp(\boldsymbol{v}_h^T\boldsymbol{x}_j)}{1+\sum_{r=1}^{M-1}\exp(\boldsymbol{v}_r^T\boldsymbol{x}_j)}\right)\boldsymbol{x}_j = 0$$

© 2009 by Taylor & Francis Group, LLC

for $h = 1, \ldots, M-1$, which is a set of nonlinear equations. It is noted that the nonlinear equation for the hth expert depends not only on the parameter vector $\boldsymbol{v}_h$, but also on other parameter vectors in $\boldsymbol{\alpha}$. In other words, each parameter vector $\boldsymbol{v}_h$ cannot be updated independently. With the IRLS algorithm presented in [12], the independence assumption on these parameter vectors was used implicitly. Ng and McLachlan [28] propose an ECM algorithm for which the M-step is replaced by $(M-1)$ computationally simpler CM-steps for $\boldsymbol{v}_h$ $(h = 1, \ldots, M-1)$.

10. McLachlan and Chang [17] consider the mixture model-based approach to the cluster analysis of mixed data, where the observations consist of both continuous and categorical variables. Suppose that p_1 of the p feature variables in $\boldsymbol{Y}_j$ are categorical, where the qth categorical variable takes on m_q distinct values $(q = 1, \ldots, p_1)$. With the location model-based cluster approach [20], the p_1 categorical variables are uniquely transformed to a single multinomial random variable $\boldsymbol{U}$ with S cells, where $S = \prod_{q=1}^{p_1} m_q$ is the number of distinct patterns (locations) of the p_1 categorical variables. We let $(\boldsymbol{u}_j)_s$ be the label for the sth location of the jth entity $(s = 1, \ldots, S;\ j = 1, \ldots, n)$, where $(\boldsymbol{u}_j)_s = 1$ if the realizations of the p_1 categorical variables correspond to the sth pattern, and is zero otherwise. The location model assumes further that conditional on $(\boldsymbol{u}_j)_s = 1$, the conditional distribution of the $p - p_1$ continuous variables is normal with mean $\boldsymbol{\mu}_{is}$ and covariance matrix $\boldsymbol{\Sigma}_i$, which is the same for all S cells. Let p_{is} be the conditional probability that $(\boldsymbol{U}_j)_s = 1$ given its membership of the ith component of the mixture $(s = 1, \ldots, S;\ i = 1, \ldots, g)$. With reference to Section 5.4.1, show that on the $(k+1)$th iteration of the EM algorithm, the updated estimates are given by

$$\pi_i^{(k+1)} = \sum_{s=1}^{S} \sum_{j=1}^{n} \delta_{js} \tau_{ijs}^{(k)} \Big/ n$$

$$p_{is}^{(k+1)} = \sum_{j=1}^{n} \delta_{js} \tau_{ijs}^{(k)} \Big/ \sum_{r=1}^{S} \sum_{j=1}^{n} \delta_{jr} \tau_{ijr}^{(k)}$$

$$\boldsymbol{\mu}_{is}^{(k+1)} = \sum_{j=1}^{n} \delta_{js} \tau_{ijs}^{(k)} \boldsymbol{y}_j^* \Big/ \sum_{j=1}^{n} \delta_{js} \tau_{ijs}^{(k)}$$

and

$$\boldsymbol{\Sigma}_i^{(k+1)} = \sum_{s=1}^{S} \sum_{j=1}^{n} \delta_{js} \tau_{ijs}^{(k)} \big(\boldsymbol{y}_j^* - \boldsymbol{\mu}_{is}^{(k+1)}\big)\big(\boldsymbol{y}_j^* - \boldsymbol{\mu}_{is}^{(k+1)}\big)^T \Big/ \sum_{s=1}^{S} \sum_{j=1}^{n} \delta_{js} \tau_{ijs}^{(k)}$$

where δ_{js} is 1 or 0 according as to whether $(\boldsymbol{u}_j)_s$ equals 1 or 0, $\boldsymbol{y}_j^*$ contains the continuous variables in $\boldsymbol{y}_j$, and

$$\tau_{ijs}^{(k)} = \pi_i^{(k)} p_{is}^{(k)} \phi\big(\boldsymbol{y}_j^*; \boldsymbol{\mu}_{is}^{(k)}, \boldsymbol{\Sigma}_i^{(k)}\big) \Big/ \sum_{h=1}^{g} \pi_h^{(k)} p_{hs}^{(k)} \phi\big(\boldsymbol{y}_j^*; \boldsymbol{\mu}_{hs}^{(k)}, \boldsymbol{\Sigma}_h^{(k)}\big)$$

for $s = 1, \ldots, S;\ i = 1, \ldots, g$.

© 2009 by Taylor & Francis Group, LLC

References

[1] A. Asuncion and D.J. Newman. UCI Machine Learning Repository. University of California, School of Information and Computer Sciences, Irvine, 2007. http://www.ics.uci.edu/ mlearn/MLRepository.html.

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

[3] D. Böhning, P. Schlattmann, and B. Lindsay. Computer-assisted analysis of mixtures (C.A.MAN): Statistical algorithms. *Biometrics*, 48:283–303, 1992.

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

[5] N.A. Campbell and R.J. Mahon. A multivariate study of variation in two species of rock crab of genus *Leptograpsus*. *Australian Journal of Zoology*, 22:417–425, 1974.

[6] D.R. Cox and D. Hinkley. *Theoretical Statistics*. Chapman & Hall, London, 1974.

[7] H. Cramér. *Mathematical Methods of Statistics*. Princeton University Press, New Jersey, 1946.

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

[9] C. Fraley and A.E. Raftery. Mclust: Software for model-based cluster analysis. *Journal of Classification*, 16:297–306, 1999.

[10] R.A. Jacobs, M.I. Jordan, S.J. Nowlan, and G.E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3:79–87, 1991.

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

[12] M.I. Jordan and R.A. Jacobs. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6:181–214, 1994.

[13] E.L. Lehmann and G. Casella. *Theory of Point Estimation*. Springer-Verlag, New York, 2003.

[14] B.G. Leroux and M.L. Puterman. Maximum-penalized-likelihood estimation for independent and Markov-dependent mixture models. *Biometrics*, 48:545–558, 1992.

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

© 2009 by Taylor & Francis Group, LLC

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

[17] G.J. McLachlan and S.U. Chang. Mixture modelling for cluster analysis. *Statistical Methods in Medical Research*, 13:347–361, 2004.

[18] G.J. McLachlan and T. Krishnan. *The EM Algorithm and Extensions (2nd edition)*. Wiley, New Jersey, 2008.

[19] G.J. McLachlan and D. Peel. Robust cluster analysis via mixtures of multivariate t-distributions. In *Lecture Notes in Computer Science*, pages 658–666. Springer-Verlag, Berlin, 1998. Vol. 1451.

[20] G.J. McLachlan and D. Peel. *Finite Mixture Models*. Wiley, New York, 2000.

[21] G.J. McLachlan and D. Peel. Mixtures of factor analyzers. In P. Langley, editor, *Proceedings of the 17th International Conference on Machine Learning*, pages 599–606, San Francisco, 2000. Morgan Kaufmann.

[22] G.J. McLachlan, D. Peel, K.E. Basford, and P. Adams. The emmix software for the fitting of mixtures of normal and t-components. *Journal of Statistical Software*, 4:No. 2, 1999.

[23] X.-L. Meng and D. Rubin. Maximum likelihood estimation via the ECM algorithm: A general framework. *Biometrika*, 80:267–278, 1993.

[24] X.-L. Meng and D.A. van Dyk. The EM algorithm—an old folk song sung to a fast new tune. *Journal of the Royal Statistical Society B*, 59:511–567, 1997.

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

[26] S.K. Ng and G.J. McLachlan. 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, 2003.

[27] S.K. Ng and G.J. McLachlan. Speeding up the EM algorithm for mixture model-based segmentation of magnetic resonance images. *Pattern Recognition*, 37:1573–1589, 2004.

[28] S.K. Ng and G.J. McLachlan. Using the EM algorithm to train neural networks: Misconceptions and a new algorithm for multiclass classification. *IEEE Transactions on Neural Networks*, 15:738–749, 2004.

[29] D. Peel and G.J. McLachlan. Robust mixture modelling using the t distribution. *Statistics and Computing*, 10:335–344, 2000.

[30] L.R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77:257–286, 1989.

[31] C.R. Rao. *Linear Statistical Inference and Its Applications (2nd edition)*. Wiley, New York, 1973.

© 2009 by Taylor & Francis Group, LLC

[32] F.-P. Schelp, P. Vivatanasept, P. Sitaputra, S. Sormani, P. Pongpaew, N. Vudhivai, S. Egormaiphol, and D. Böhning. Relationship of the morbidity of under-fives to anthropometric measurements and community health intervention. *Tropical Medicine and Parasitology*, 41:121–126, 1990.

[33] G.C.G. Wei and M.A. Tanner. 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, 1990.

© 2009 by Taylor & Francis Group, LLC