Computation of the EM Iteration for Multivariate Normal Data with Missing Values *

Chris Fraley

MathSoft, Inc.
Data Analysis Products Division
1700 Westlake Avenue North, Suite 500
Seattle, WA 98109 USA

May 25, 1998

Contents

List of Figures

Keywords: EM algorithm, missing data, Cholesky factorization

*Funded by National Institutes of Health SBIR Grant 5R44CA65147-03

1 Introduction

We address the problem of estimating the distributional parameters for multivariate normal (MVN) data that contains missing values via the well-known EM (Expectation-Maximization) iteration (Dempster, Laird, and Rubin [3]). The data Y is represented as a matrix of n rows and p columns, whose rows correspond to the individual observations and whose columns correspond to the variables in the model. The rows of Y are assumed to be independent and identically distributed (iid) according to a multivariate normal distribution with mean vector μ and covariance matrix Σ. The objective is to estimate the parameters μ and Σ of this distribution, assuming no prior restrictions other than positive definiteness of Σ.

The (MVN) density of a single observation of Y is

$$(|2\pi\Sigma|)^{-1/2} \exp\left\{-\frac{1}{2}(y_k-\mu)^T\Sigma^{-1}(y_k-\mu)\right\}$$

where y_k is the kth row of Y represented as a column vector, so that an expression for the likelihood for Y is then

$$L(\mu,\Sigma \mid Y) \propto |\Sigma|^{-n/2} \exp\left\{-\frac{1}{2}\sum_{k=1}^{n}(y_k-\mu)^T\Sigma^{-1}(y_k-\mu)\right\}. \tag{1}$$

When all of the data in Y is observed, the values of μ and Σ maximizing (1) are

$$\hat{\mu} = \frac{1}{n}\sum_{k=1}^{n} y_k, \tag{2}$$

the vector of individual column means, and

$$\hat{\Sigma} = \frac{1}{n}Y^TY - \hat{\mu}\hat{\mu}^T. \tag{3}$$

However, when Y contains missing data, further assumptions have to be made about the distribution of the missing data in order to obtain a solution. Following Little and Rubin [10] and Schafer [12], let Y_O be the observed portion of Y, and Y_M be the missing portion. If the mechanism for creating the missing data is ignorable, that is, if the probability that a particular variable in an observation is missing may depend on Y_O but not on Y_M, then the relevant density can be obtained by integrating the missing data Y_M out of the complete data loglikelihood. This reasoning holds for more general distributions, although the multivariate normal model is the only one of interest here. The parameters can be estimated via the EM algorithm ([3] — see McLachlan and Krishnan [11] for a recent treatment of EM and its applications).

The EM iteration alternates between two steps, an 'E-step', in which the conditional expectation of the complete data loglikelihood given the observed data and the current parameter estimates is computed, and an 'M-step' in which parameters are determined that maximize the expected loglikelihood from the E-step. Under fairly mild regularity assumptions, the iteration converges to a local maximum of the complete data loglikelihood ([**?**], Boyles [1], Wu [14], [11], [12]).

Arguments for and against the use of EM have been presented elsewhere (e. g. Little and Rubin [10], Schafer [12]). Its main drawback is that the rate of covergence is linear and

can be slow. However, in the multivariate normal case direct optimization of the complete data loglikelihood via superlinearly convergent methods is impractical except when the number of variables p is small because there are $p + (p(p+1))/2$ parameters to be estimated, and the required derivatives are not sparse. Although the Gaussian model is not always a good representation of the data, it is nevertheless useful as a point of departure in iterative simultation for data whose distributions are not directly accessible [12].

The E-step involves linear regressions and formation of the covariance matrix that are typically computed via sweep operations. This allows relevant quantities to be updated rather than recomputed between successive missing data patterns, as well as efficient use of memory. A disadvantage is that it involves formation of sums of crossproducts of data values, so that scaling may be required to keep quantities involved in the computations from growing too large. Moreover, the sweep method allows growth in numerical errors which can increase the number of iterations required for convergence and/or reduce the accuracy of the resulting estimates of the optimal parameters. The purpose of this paper is to give a scheme based on parameterization in terms of the Cholesky factor of Σ that is more stable and accurate.

This paper is organized as follows. Section 2 gives the E-steps and M-steps of the iteration as they are conventionally formulated. Computation of the E-step via sweep operations is then described in section 3, followed by details of a Cholesky-based method for computing the EM iteration in section 4.

2 EM for Multivariate Normal Missing Data

This section summarizes information that is covered in more detail in [10] and [12]. In the EM iteration for multivariate normal datawith missing values, the E-step involves computing the expectation values of the sufficient statistics $\sum_{k=1}^{n} y_k$ and $\sum_{k=1}^{n} y_{ki} y_{kj}$, for $i, j = 1, 2, \ldots, p$ given the current values of the parameters μ and Σ. The M-step is then straighforward — these expectation values are just substituted into the maximum likelihood expressions (2) and (3) for complete data:

$$\mu \leftarrow n^{-1} \mathcal{E}(\sum_{k=1}^{n} y_k); \qquad \Sigma_{ij} \leftarrow n^{-1} \mathcal{E}(\sum_{k=1}^{n} y_{ki} y_{kj}) - \mu_i \mu_j. \tag{4}$$

It is understood that all expectations are taken with respect to the observed data and the current estimates of the parameters μ and Σ.

Since $\mathcal{E}(\sum_{k=1}^{n} y_k) = \sum_{k=1}^{n} \mathcal{E}(y_k)$, expectations of the individual missing values are computed in the E-step. An observation y_k containing missing data can be partitioned into observed and missing portions; that is

$$P y_k = \begin{pmatrix} y_k^O \\ y_k^M \end{pmatrix},$$

where P is permutation. The current parameter estimates can be correspondingly partitioned:

$$P\mu = \begin{pmatrix} \mu^O \\ \mu^M \end{pmatrix} \quad \text{and} \quad P\Sigma P^T = \begin{pmatrix} \Sigma_{OO} & \Sigma_{MO}^T \\ \Sigma_{MO} & \Sigma_{MM} \end{pmatrix}. \tag{5}$$

Because of the iid assumption, each observation has a multivariate normal distribution with unknown mean μ and covariance Σ (for which there are estimates). Because of the ignorability assumption, the conditional distribution of the missing observations is also normal, so that

$$\mathcal{E}(y_k^M | y_k^O, \mu, \Sigma) = \mu^M + \Sigma_{MO}\Sigma_{OO}^{-1}(y_k^O - \mu^O).$$

This is the mean value of a linear regression with the observed values of y_k as predictor variables, given the current μ and Σ. Hence

$$\begin{array}{rcl} \mathcal{E}(y_k^O) & = & y_k^O; \\ \mathcal{E}(y_k^M) & = & \mu^M + \Sigma_{MO}\Sigma_{OO}^{-1}\left(y_k^O - \mu^O\right). \end{array} \tag{6}$$

For the crossproduct terms,

$$\begin{array}{rcl} \mathcal{E}(y_{k_i}y_{k_j}) & = & \mathcal{E}(y_{k_i})\mathcal{E}(y_{k_j}) + Cov(y_{k_i}y_{k_j}) \\ & = & \begin{cases} y_{k_i}y_{k_j}, & y_{k_i},\, y_{k_j} \text{ observed}; \\ y_{k_i}\mathcal{E}(y_{k_j}), & y_{k_i} \text{ observed},\, y_{k_j} \text{ missing}; \\ \mathcal{E}(y_{k_i})\mathcal{E}(y_{k_j}) + \varsigma_{ij}^k, & y_{k_i},\, y_{k_j} \text{ missing}. \end{cases} \end{array} \tag{7}$$

The term ς_{ij}^k is the element corresponding to y_{k_i} and y_{k_j} in

$$\Sigma_{MM} - \Sigma_{MO}\Sigma_{OO}^{-1}\Sigma_{MO}^T, \tag{8}$$

the covariance matrix of the missing elements given the observed data and the current values of μ and Σ. This value is the same for all rows having a given missing data pattern.

Given the above formulas for the expectation values of the data, a procedure for computing the estimated parameters is given in Figure 2. Note that only Σ_{OO}, rather than Σ, need be nonsingular for all data patterns.

If the missing data patterns are organized in such a way that close patterns are processed in succession, then it is more efficient to update the quantities needed to form $\Sigma_{MM} - \Sigma_{MO}\Sigma_{OO}^{-1}\Sigma_{MO}^T$ and y_k^M than compute them completely from scratch. Finding the pattern that produces the most efficient computation in every case is impractical since it would involve the solution of a combinatorial problem. A good heuristic is to organize columns in order of increasing number of missing observations, and then order the rows in increasing numeric order treating the missing data patterns as bitwise representations of integers (missing obervations encoded as 0). Patterns of missingness in which observed data always precedes missing data are called *monotone* data patterns, and parameter estimates can be obtained directly rather than iteratively for data that falls into this category. The heuristic mentioned above for ordering will expose monotone patterns.

Because sums of products of the data elements are being formed to get the elements of Σ, quantities could become rather large course of the computation, leading to numerical instability. The data can be centered and scaled relative to the observed values, although there may be some loss of accuracy in recovering the original parameters. Moreover, estimates obtained from centered and scaled data will not necessarily correspond to larger values of the observed data loglikelihood than estimates from unscaled data (see section [**?**]).

Form s_o, the sum of the observed values in each variable of Y, and
W_o, the matrix of partial sums of products involving pairs of observed values.

Assume estimates μ, Σ of the mean and covariance are given.

`repeat`

$s \leftarrow s_o;\ W \leftarrow W_o$

`for` each missing data pattern $\mathcal{P}$

`for` each observation k conforming to $\mathcal{P}$

$y_k^M \leftarrow \mu^M + \Sigma_{MO}\Sigma_{OO}^{-1}\,(y_k^O - \mu^O)$

Add values of y_k^M to the appropriate components of s.

Form all products involving y_k^M and add them into W.
Add the corresponding entry of $\Sigma_{MM} - \Sigma_{MO}\Sigma_{OO}^{-1}\Sigma_{MO}^T$ to each product involving two missing terms of the current observation.

`end for`

`end for`

$\mu \leftarrow \dfrac{s}{n};\ \Sigma \leftarrow \dfrac{W}{n} - \mu\mu^T$

`until` termination criteria are satisfied

Figure 1: Conventional procedure for estimation of μ and Σ via EM.

3 Computation via Sweep Operations

Little and Rubin [10] and Schafer [12] give computational methods for the EM iteration for MVN missing data following the scheme of Figure 2 in terms of sweep operations. This allows advantage to be taken of symmetry as well as updating between related missing data patterns. Sweep operations are widely used for organizing computations for linear regression via the normal equations (see, e. g. Thisted [13]), and expectation values of the missing elements in the E-step of the EM iteration are obtained via linear regression (see section 2).

Sweeping a (symmetric) positive-definite matrix relative to the first variable has the following effect:

$$\begin{pmatrix} \alpha & b \\ b^T & C \end{pmatrix} \overset{sweep}{\underset{\{1\}}{\longrightarrow}} \begin{pmatrix} -1/\alpha & b^T/\alpha \\ b\ /\alpha & C - (bb^T)/\alpha \end{pmatrix}$$

The matrix can be swept relative to any variable, although it is easier to visualize by permuting the relevant rows and columns into the leading positions, while suppressing notation for the permutations. For the EM iteration, sweeping the current Σ relative to the observed variables for a given missing data pattern results in the quantities needed to compute the expectation values in the E-step (see section 2). As an example, for a missing data pattern in which the first k variables are observed and the remaining variables are missing, the current Σ would be swept relative to those variables to give:

$$\begin{pmatrix} \Sigma_{OO} & \Sigma_{MO}^T \\ \Sigma_{MO} & \Sigma_{MM} \end{pmatrix} \overset{sweep}{\underset{\{1,\ldots,k\}}{\longrightarrow}} \begin{pmatrix} -\Sigma_{OO}^{-1} & \Sigma_{OO}^{-1}\Sigma_{MO}^T \\ \Sigma_{MO}\Sigma_{OO}^{-1} & \Sigma_{MM} - \Sigma_{MO}\Sigma_{OO}^{-1}\Sigma_{MO}^T \end{pmatrix};\qquad \Sigma_{OO}\ k \times k.$$

A sweep operation cannot be represented by a single matrix operator; it is rather a composite of matrix operations. Sweep operations are reversible, so that in the EM iteration one can proceed from one missing data pattern to the next by doing a reverse sweep operation for each variable that is observed in the current pattern but missing in the next, and a sweep operation for each variable that is missing in the current pattern but observed in the next. All sweep and reverse sweep operations are commutative, so that they can be performed in any order. Since the matrices involved are symmetric, operations need only be carried out on either an upper or lower triangle.

Sweep operations are usually presented in terms of the individual arithmetic operations required to transform one tableau to another. However, provided the relevant rows and columns are gathered into contiguous blocks, sweeps can be carried out via matrix operations, allowing more opportunity for compiler and run-time optimization [9], [5], [4]. The same holds true for the matrix multiplication in computing the missing data estimates (6) and for extracting elements in (8) when forming crossproducts involving two missing entries. Depending on the missing data pattern and the computing environment, permuting to achieve the ordering in (5) could be more efficient despite the need for data movement.

4 Computation Based on the Cholesky Factor of Σ

4.1 Expectations of missing values

Analogous to the partitioning of Σ in (5), the columns in the Cholesky factor R of Σ can be partitioned so that observed variables precede the missing ones

$$\begin{pmatrix} R_{OO} & R_{OM} \\ 0 & R_{MM} \end{pmatrix}^T \begin{pmatrix} R_{OO} & R_{OM} \\ 0 & R_{MM} \end{pmatrix} = \begin{pmatrix} R_{OO}^T R_{OO} & R_{OO}^T R_{OM} \\ R_{OM}^T R_{OO} & R_{OM}^T R_{OM} + R_{MM}^T R_{MM} \end{pmatrix} = \begin{pmatrix} \Sigma_{OO} & \Sigma_{MO}^T \\ \Sigma_{MO} & \Sigma_{MM} \end{pmatrix}.$$

The expression for the expectation values of the missing data in (6) has the equivalent formulation

$$\mathcal{E}(y_k^M) \;=\; \mu^M + R_{OM}^T R_{OO}^{-T} (y_k^O - \mu^O) \tag{9}$$

since

$$\Sigma_{MO}\Sigma_{OO}^{-1} = R_{OM}^T R_{OO}(R_{OO}^{-1} R_{OO}^{-T}) = R_{OM}^T R_{OO}^{-T}.$$

If the number of missing observations in this particular pattern is greater than the number of observations having the pattern, then $z = R_{OO}^{-T}(y_k^O - \mu^O)$ should be computed for each missing observation by solving $R_{OO} z = (y_k^O - \mu^O)$, followed by formation of $R_{OM}^T z$. Otherwise it is more efficient to form $Z = R_{OO}^{-1} R_{OM}$ by solving $R_{OO} Z = R_{OM}$ (Z overwrites R_{OM}), then form $Z^T (y_k^O - \mu^O)$ for each missing observation. Note that only the nonsingularity of R_{OO} for each missing data pattern is required in order to obtain new parameter estimates. An advantage of using (9) is that the upper bound on the size of numerical errors is approximately the square root of that for the normal equations (6) (Golub and Van Loan [8]). Moreover, solution of linear equations with triangular coefficient matrices can be accomplished very efficiently ([8], Dongarra et al. [5]).

In order to take full advantage of the efficiency of the triangular solves in (9), it is necessary to order the rows of Y so that those with the same data pattern occur consecutively,

and to permute the columns of the Cholesky factor between missing data patterns. The following illustrates the permutation procedure for a five-dimensional case, in which observations having missing data in (say) column 5 are followed by ones in which columns 2 and 4 are missing. The first step is a permutation step, while the remaining steps restore the matrix to triangular form.

$$\begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & 0 & 0 & \times \end{pmatrix} \overset{permute}{\longrightarrow} \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & \times & \times & 0 & \times \\ 0 & 0 & \times & 0 & \times \\ 0 & 0 & \times & 0 & 0 \end{pmatrix} \overset{Givens}{\longrightarrow} \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \tilde{\times} & \tilde{\times} & \tilde{\times} & \tilde{\times} \\ 0 & \tilde{0} & \tilde{\times} & \tilde{\times} & \tilde{\times} \\ 0 & 0 & \times & 0 & \times \\ 0 & 0 & \times & 0 & 0 \end{pmatrix}$$

$$\overset{Givens}{\longrightarrow} \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \tilde{\times} & \tilde{\times} & \tilde{\times} \\ 0 & 0 & \tilde{0} & \tilde{\times} & \tilde{\times} \\ 0 & 0 & \times & 0 & 0 \end{pmatrix} \overset{Givens}{\longrightarrow} \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \tilde{\times} & \tilde{\times} & \tilde{\times} \\ 0 & 0 & 0 & \times & \times \\ 0 & 0 & \tilde{0} & \tilde{\times} & \tilde{\times} \end{pmatrix} \overset{Givens}{\longrightarrow} \begin{pmatrix} \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times \\ 0 & 0 & \times & \times & \times \\ 0 & 0 & 0 & \tilde{\times} & \tilde{\times} \\ 0 & 0 & 0 & \tilde{0} & \tilde{\times} \end{pmatrix}$$

The symbol $\overset{Givens}{\longrightarrow}$ stands for application of a *Givens rotation*, an elementary orthogonal transformation that allows selective and numerically stable introduction of zero elements in a matrix [8]. The marked entries are values changed in the last transformation. The basic idea is to permute the columns in such a way that missing columns follow the observed ones, but otherwise the order of columns is preserved from the previous configuration. This ensures that the sparser columns tend to be the leading columns, thus minimizing the number of operations required to restore upper-triangular form. In the example above more operations would have been required had column 5 preceded column 3 in the permutation.

The efficiency of the update sequence is dependent on the overall missing data pattern. For a monotone data pattern (observed values always precede missing values) in which the rows are ordered by increasing number of missing values, no column permutation is necessary. However in this case the parameter estimates need not be computed iteratively [10], [12].The number of operations required for the permutations will be minimized if the the rows and columns of Y are ordered so that the overall missing data pattern is as close to monotone as possible (see sections 2 and 4.3).

4.2 Forming the Cholesky factor of the estimate for Σ

Instead of obtaining an estimate for Y^TY in the E-step, the Cholesky factor of the estimate of Σ resulting from the M-step is formed as the rows are processed.

From the expressions for the elements of Σ is the M-step (4), it follows that

$$\hat{\Sigma} = \frac{\mathcal{E}(Y^TY)}{n} - \hat{\mu}\hat{\mu}^T. \tag{10}$$

The matrix $\mathcal{E}(Y^TY)$ can be expressed as the sum of two parts:

$$\mathcal{E}(Y^TY) = \mathcal{E}(Y)^T\mathcal{E}(Y) + S; \qquad S \equiv \sum_{k=1}^{n} Cov(y_{k_i}y_{k_j}).$$

Since $Cov(y_{k_i} y_{k_j})$ is the same for all rows k having a given missing data pattern, an alternative expression for $\mathcal{E}(Y^T Y)$ is

$$\mathcal{E}(Y^T Y) = \mathcal{E}(Y)^T \mathcal{E}(Y) + \sum_{i=1}^{m} n_i S_i, \tag{11}$$

where m is the number of missing data patterns, n_i is the number of observations having the ith missing data pattern, and S_i the covariance matrix associated with that pattern (its nonzero portion is a permutation of (8)). Moreover, each S_i is positive semi-definite, with non-zero elements corresponding to the matrix $\Sigma_{MM} - \Sigma_{MO}\Sigma_{OO}^{-1}\Sigma_{MO}^T$ in (8) for the ith data pattern. Since

$$\Sigma_{MM} - \Sigma_{MO}\Sigma_{OO}^{-1}\Sigma_{MO}^T = \left(R_{OM}^T R_{OM} + R_{MM}^T R_{MM}\right) - \left(R_{OM}^T R_{OO}^{-T}\right) R_{OO}^T R_{OM} = R_{MM}^T R_{MM},$$

R_{MM} is the Cholesky factor of $\Sigma_{MM} - \Sigma_{MO}\Sigma_{OO}^{-1}\Sigma_{MO}^T$. It follows that $S_i = A_i^T A_i$, where the associated R_{MM} is a submatrix of a permutation of the columns of A_i (all other elements of A_i vanish). Using this representation of S_i in (11) gives the following expression for the updated estimate of Σ (10) :

$$\hat{\Sigma} = \frac{\mathcal{E}(Y)^T \mathcal{E}(Y)}{n} - \hat{\mu}\hat{\mu}^T + \sum_{i=1}^{m} \frac{n_i}{n} R_i^T R_i.$$

Now

$$\frac{\mathcal{E}(Y)^T \mathcal{E}(Y)}{n} - \hat{\mu}\hat{\mu}^T = \frac{1}{n}\sum_{k=1}^{n} (\mathcal{E}(y_k) - \hat{\mu})(\mathcal{E}(y_k) - \hat{\mu})^T = \frac{1}{n}\tilde{Y}^T\tilde{Y},$$

where $\tilde{Y}$ is the matrix $\mathcal{E}(Y)$ after subtracting the estimated column means. The matrix $\tilde{Y}^T\tilde{Y}$ has an alternative decomposition in terms of rank-1 matrices that allows row-wise accumulation:

$$\tilde{Y}^T\tilde{Y} = \sum_{k=1}^{n} v_k\, v_k^T; \qquad v_k \equiv \sqrt{\frac{1}{k(k-1)}}\, s_{k-1} - \sqrt{\frac{k-1}{k}}\,\mathcal{E}(y_k) \quad s_0 \equiv 0, \quad s_j \equiv \sum_{i=1}^{j} \mathcal{E}(y_j).$$

Note that each v_k depends only on rows $j \le k$. Assuming that rows are ordered according to missing data patterns, we may write

$$\tilde{Y}^T\tilde{Y} = \tilde{Y}_0^T\tilde{Y}_0 + \sum_{i=1}^{m} \sum_{k=k_{\min}^{(i)}}^{k_{\max}^{(i)}} v_k\, v_k^T \equiv \sum_{i=0}^{m} \tilde{Y}_i^T\tilde{Y}_i,$$

where $\tilde{Y}_0^T\tilde{Y}_0$ involves only complete rows, and $k_{\min}^{(i)}, k_{\max}^{(i)}$ are the largest and smallest row indexes associated with the ith nontrivial missing data pattern. It follows that

$$n\hat{\Sigma} = \tilde{Y}_0^T\tilde{Y}_0 + \sum_{i=1}^{m} \left\{\tilde{Y}_i^T\tilde{Y}_i + n_i A_i^T A_i\right\} = \begin{pmatrix} \tilde{Y}_0 \\ \tilde{Y}_1 \\ \sqrt{n_1}A_1 \\ \vdots \\ \tilde{Y}_m \\ \sqrt{n_m}A_m \end{pmatrix}^T \begin{pmatrix} \tilde{Y}_0 \\ \tilde{Y}_1 \\ \sqrt{n_1}A_1 \\ \vdots \\ \tilde{Y}_m \\ \sqrt{n_m}A_m \end{pmatrix},$$

a formulation that is compatible with row-wise formation of the Cholesky factor. The Cholesky factor of $\tilde{Y}_0^T \tilde{Y}_0$ is formed at the outset, before the iteration is started. As the iteration, proceeds this Cholesky factor is updated one row at a time. For each missing data pattern $\mathcal{P}_i$, expectation values of the missing data are computed using (9). When the Cholesky factor has been updated for each row having that pattern, the non-zero rows of A_i are incorporated with the appropriate weight ($\sqrt{n_i}$). The number of these rows is equal to the number of missing elements (m_i) in the pattern.

The row update is accomplished as follows: if R is the Cholesky factor X, then the Cholesky factor $\check{R}$ of $\begin{pmatrix} X \\ x^T \end{pmatrix}^T \begin{pmatrix} X \\ x^T \end{pmatrix} = X^T X + xx^T$ can be formed using Givens rotations. The procedure is illustated below for a 5-dimensional example:

$$\begin{pmatrix} X \\ x^T \end{pmatrix} = \begin{pmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times \\ \times & \times & \times & \times \end{pmatrix} \overset{Givens}{\longrightarrow} \begin{pmatrix} \tilde{\times} & \tilde{\times} & \tilde{\times} & \tilde{\times} \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times \\ \tilde{0} & \tilde{\times} & \tilde{\times} & \tilde{\times} \end{pmatrix} \overset{Givens}{\longrightarrow} \begin{pmatrix} \times & \times & \times & \times \\ 0 & \tilde{\times} & \tilde{\times} & \tilde{\times} \\ 0 & 0 & \times & \times \\ 0 & 0 & 0 & \times \\ 0 & \tilde{0} & \tilde{\times} & \tilde{\times} \end{pmatrix}$$

$$\overset{Givens}{\longrightarrow} \begin{pmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \tilde{\times} & \tilde{\times} \\ 0 & 0 & 0 & \times \\ 0 & 0 & \tilde{0} & \tilde{\times} \end{pmatrix} \overset{Givens}{\longrightarrow} \begin{pmatrix} \times & \times & \times & \times \\ 0 & \times & \times & \times \\ 0 & 0 & \times & \times \\ 0 & 0 & 0 & \tilde{\times} \\ 0 & 0 & 0 & \tilde{0} \end{pmatrix} = \begin{pmatrix} \check{R} \\ 0 \end{pmatrix};$$

$$X^T X + xx^T = \check{R}^T \check{R}.$$

The marked entries are values changed in the last transformation. The time efficiency for the Cholesky update is $\mathcal{O}(p^2)$, in contrast to $\mathcal{O}(p^3)$ for forming a new Cholesky factor from the updated $p \times p$ matrix $X^T X + xx^T$. For details of the Cholesky update via Givens rotations, see [8]. The non-zero rows of A_i will often contain leading zeros, so that it will require fewer rotations to incorporate them than otherwise.

4.3 Retaining invariant Givens rotations

A synthesis of the Cholesky-based method for the EM iteration described in sections 4.1 and 4.2 is given in Figure 4.3.

We have already mentioned that it is desirable for the rows and columns of the data Y to be ordered so that its overall missing data pattern is as close to monotone as possible for efficient restoration of triangular form between missing data patterns (section 4.1) and described a heuristic for achieving this order (section 2).

Further advantage can be taken of the monotone structure of the data with Cholesky method. For a given data set, assume that the rows of its missing data pattern are ordered according to decreasing number of leading observed elements (complete rows can be ignored, since they are processed in advance). If the observations are processed in this order, then the Givens rotations needed to process the entries corresponding to the leading observed elements in the restoration to triangular form are known and can be applied before the iteration begins. The rotations can be saved in a preprocessing step as a vector that is

accessed sequentially during the iteration and applied starting with the first missing element in each row. Ordinarily two values are used to specify a Givens rotation, but these can be encoded by a single value if necessary in order to save space [8]. The requirement of additional storage of one or two values for each leading observed element in a row is offset by the savings for computing and applying the Givens rotations to the leading observed elements in each iteration.

References

[1] R. A. Boyles. On the convergence of the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 45:47–50, 1983.

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

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

[4] J. J. Dongarra, J. Du Croz, I. S. Duff, and S. Hammarling. A set of level 3 basic linear algebra subprograms. *ACM Transactions on Mathematical Software*, 16:1–28, 1990.

[5] J. J. Dongarra, J. Du Croz, S. Hammarling, and R. J. Hanson. An extended set of FORTRAN basic linear algebra subprograms. *ACM Transactions on Mathematical Software*, 14:1–32, 1988.

[6] C. Fraley. Algorithms for model-based Gaussian hierarchical clustering. *SIAM Journal on Scientific Computing*, 1998. (to appear).

[7] C. Fraley. Multivariate normal parameter estimation from monotone missing data patterns. Technical report, MathSoft, Inc., Data Analysis Products Division, 1998. (in preparation).

[8] G. H. Golub and C. F. Van Loan. *Matrix Computations.* Johns Hopkins, 3rd edition, 1996.

[9] C. L. Lawson, R. J. Hanson, D. Kincaid, and F. T. Krogh. Basic linear algebra subprograms for FORTRAN usage. *ACM Transactions on Mathematical Software*, 5:308–323, 1979.

[10] R. J. A. Little and D. B. Rubin. *Statistical Analysis with Missing Data.* Wiley, 1987.

[11] G. J. McLachlan and T. Krishnan. *The EM Algorithm and Extensions.* Wiley, 1997.

[12] J. L. Schafer. *Analysis of Incomplete Multivariate Data by Simulation.* Chapman and Hall, 1997.

[13] R. A. Thisted. *Elements of Statistical Computing*. Chapman and Hall, 1988.

[14] C. F. J. Wu. On convergence properties of the EM algorithm for Gaussian mixtures. *The Annals of Statistics*, 11:95–103, 1983.

Order the rows of Y according to missing data pattern.

Compute s_c and R_c, the column sums and Cholesky factor of the sample cross-product matrix for the complete observations.

Assume estimates μ, R of the mean and the Cholesky factor of Σ are given.

`repeat`

$k \leftarrow$ number of complete observations; $s \leftarrow s_c$; $U \leftarrow R_c$

`for` each nontrivial missing data pattern $\mathcal{P}_i, i = 1, 2, \ldots, m$

Let n_i be the number of observations conforming to $\mathcal{P}_i$, and
let m_i be the number of missing observations in that pattern.

Permute columns of R so that observed variables precede missing ones (P);
Restore to upper triangular via using orthogonal transformations (Q):

$$QRP = \begin{pmatrix} R_{OO} & R_{OM} \\ 0 & R_{MM} \end{pmatrix}.$$

Condition estimate of R_{OO}; take appropriate action if nearly singular.

`if` $n_i \leq m_i$ `then`

`for` each observation conforming to $\mathcal{P}_i$

Solve $R_{OO}^T z = (y_{k+1}^O - \mu^O)$ for z; $y_{k+1}^M \leftarrow \mu^M + R_{OM}^T z$.

Rank-one update of U with $\sqrt{\frac{1}{k(k+1)}}s - \sqrt{\frac{k}{(k+1)}}y_{k+1}$; $s \leftarrow s + y_{k+1}$; $k \leftarrow k+1$

`end for`

`else`

Solve $R_{OO}Z = R_{OM}$ for Z (Z overwrites R_{OM})

`for` each observation conforming to $\mathcal{P}_i$

$y_{k+1}^M \leftarrow \mu^M + Z^T(y_{k+1}^O - \mu^O)$.

Rank-one update of U with $\sqrt{\frac{1}{k(k+1)}}s - \sqrt{\frac{k}{(k+1)}}y_{k+1}$; $s \leftarrow s + y_{k+1}$; $k \leftarrow k+1$

`end for`

`end if`

`for` each missing variable in $\mathcal{P}_i$

Let w be a p-vector which has the corresponding row of R_{MM} in the respective positions of missing variables and is otherwise zero.

Rank-one update of U with $\sqrt{n_i}\ w$.

`end for`

`end for`

$\mu \leftarrow s/n$; $R \leftarrow U/\sqrt{n}$

`until` termination criteria are satisfied

Figure 2: EM iteration based on the Cholesky factor of the estimate of Σ. Can be enhanced by ordering data patterns to be as close to monotone as possible, as well as applying Givens rotations corresponding to the leading observed elements and saving them for later use.