Stat Comput
DOI 10.1007/s11222-007-9042-y

ML estimation for factor analysis: EM or non-EM?

J.-H. Zhao · Philip L.H. Yu · Qibao Jiang

Received: 24 June 2007 / Accepted: 5 November 2007

Abstract To obtain maximum likelihood (ML) estimation in *factor analysis* (FA), we propose in this paper a novel and fast conditional maximization (CM) algorithm, which has quadratic and monotone convergence, consisting of a sequence of CM log-likelihood (CML) steps. The main contribution of this algorithm is that the closed form expression for the parameter to be updated in each step can be obtained explicitly, without resorting to any numerical optimization methods. In addition, a new ECME algorithm similar to Liu's (Biometrika 81, 633–648, 1994) one is obtained as a by-product, which turns out to be very close to the simple iteration algorithm proposed by Lawley (Proc. R. Soc. Edinb. 60, 64–82, 1940) but our algorithm is guaranteed to increase log-likelihood at every iteration and hence to converge. Both algorithms inherit the simplicity and stability of EM but their convergence behaviors are much different as revealed in our extensive simulations: (1) In most situations, ECME and EM perform similarly; (2) CM outperforms EM and ECME substantially in all situations, no matter assessed by the CPU time or the number of iterations. Especially for the case close to the well known *Heywood* case, it accelerates EM by factors of around 100 or more. Also, CM is much more insensitive to the choice of starting values than EM and ECME.

Keywords CM · ECME · EM · Factor analysis · Maximum likelihood estimation

1 Introduction

Factor analysis (FA) is a powerful multivariate analysis technique that identifies the common characteristics among a set of variables and has been widely used in many disciplines such as botany, biology, social sciences, economics, and engineering.

The ML method for fitting FA has been very popular for many decades. This method was originally discussed in Lawley (1940), in which a simple iteration algorithm with the advantage of tempting simplicity was suggested. However, this method has several practical problems. First, convergence of this algorithm can not be guaranteed (Lawley and Maxwell 1971, p. 30) though in practice this algorithm usually converges (Jöreskog 1967). Second, in some cases the estimation of factor loadings fails because the computation generates imaginary numbers (Jöreskog 1967). Third, convergence of this algorithm is at best linear (Jennrich and Bobinson 1969).

In order to tackle the first two problems, the expectation-maximization (EM) algorithm (Dempster et al. 1977) with the advantage of simplicity and stability has been suggested (Rubin and Thayer 1982) for fitting FA. Although the attractive property of EM algorithm, its convergence may be impractically slow. To accelerate the above EM algorithm,

J.-H. Zhao (✉) · P.L.H. Yu
Department of Statistics and Actuarial Science, The University of Hong Kong, Fokfulam, Hong Kong
e-mail: jhzhao.ynu@gmail.com

P.L.H. Yu
e-mail: plhyu@hku.hk

J.-H. Zhao
Department of Statistics, Yunnan University, Kunming 650091, China

Q. Jiang
Department of Mathematics, Southeast University,
Nanjing 210096, China
e-mail: qibao@seu.edu.cn

Liu and Rubin (1998) proposed an ECME (Liu 1994) algorithm (denoted as ECME1 hereafter) which can benefit from a conditional likelihood maximization step via Newton-Raphson algorithm. However, a well known disadvantage of Newton-Raphson is that its convergence can not be guaranteed and thus it requires an additional check to see whether the log-likelihood in each iteration is increased.

In attempting to tackle all three problems, another group of researchers suggested using numerical optimization methods to maximize the likelihood function. For instance, Jöreskog (1967) considered a quasi Newton-Raphson algorithm (Fletcher and Powell 1963), which has been found empirically to converge faster than EM and become the standard algorithm for fitting FA so far. To achieve further acceleration, Jennrich and Bobinson (1969) recommended a Newton-Raphson algorithm. However, like the above ECME, this method suffers from the problem of Newton-Raphson mentioned above.

In this paper, we shall propose two new algorithms. The first algorithm is an ECME (denoted as ECME2 hereafter) and is very similar to the simple iteration algorithm but it can obviate the first two problems mentioned above. It consists of two steps: step 1 conditionally maximizes log-likelihood (CML) over factor loadings given uniqueness; step 2 conditionally maximizes expected complete log-likelihood, i.e., so-called Q function, (CMQ) over uniqueness given factor loadings. The second algorithm is a CM algorithm (Meng and Rubin 1993), in which, the first step is the same as the first one in ECME2. Then, instead of using numerical optimization method such as a (quasi) Newton-Raphson algorithm to update uniqueness given factor loadings, a sequence of CML steps are performed and in each step only an element of uniqueness is updated with the other elements and factor loadings fixed. The advantage of doing so is that a close-form expression for each element of uniqueness can be obtained explicitly.

The remainder of the paper is organized as follows. Section 2 gives a review of FA model and three algorithms: (Lawley 1940)'s simple iteration algorithm, an EM algorithm and an ECME1 algorithm. Sections 3 and 4 propose our ECME2 and CM algorithm, respectively. Section 5 conducts a simulation study to compare EM, ECME2 and CM. We end the paper with a conclusion in Sect. 6.

2 FA model and three estimation algorithms

2.1 FA model

Suppose that a d-dimensional data vector $\mathbf{x}_j$ in an i.i.d sample $\mathcal{X}_n = \{\mathbf{x}_j\}_{j=1}^n$ follows a q-factor model:

$$\begin{cases} \mathbf{x}_j = \mathbf{A}\mathbf{y}_j + \boldsymbol{\mu} + \boldsymbol{\epsilon}_j, \\ \mathbf{y}_j \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \quad \boldsymbol{\epsilon}_j \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Psi}), \end{cases} \tag{1}$$

where $\boldsymbol{\mu}$ is a d-dimensional mean vector, $\mathbf{A}$ is a $d \times q$ factor loadings matrix, $\mathbf{y}_j$ is a q-dimensional latent vectors, representing those factors common to all components of $\mathbf{x}_j$, and $\boldsymbol{\Psi} = \text{diag}\{\psi_1, \psi_2, \ldots, \psi_d\}$ is a positive diagonal matrix. We use $\mathbf{I}$ to denote an unit matrix whose dimension should be apparent from the context.

Under the model (1), $\mathbf{x}_j \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma} \triangleq \boldsymbol{\Psi} + \mathbf{A}\mathbf{A}'$. Let

$$\bar{\mathbf{x}} = \frac{1}{n}\sum\nolimits_{j=1}^n \mathbf{x}_j, \qquad \mathbf{S} = \frac{1}{n}\sum\nolimits_{j=1}^n (\mathbf{x}_j - \bar{\mathbf{x}})(\mathbf{x}_j - \bar{\mathbf{x}})' \tag{2}$$

be the sample mean vector and sample covariance matrix of $\mathbf{x}$. Then the log-likelihood is

$$\begin{aligned} &l(\boldsymbol{\mu}, \mathbf{A}, \boldsymbol{\Psi} | \mathcal{X}_n) \\ &\quad = -\frac{n}{2}\big\{\ln|\boldsymbol{\Sigma}| + \text{tr}(\boldsymbol{\Sigma}^{-1}\mathbf{S}) + (\bar{\mathbf{x}} - \boldsymbol{\mu})'\boldsymbol{\Sigma}^{-1}(\bar{\mathbf{x}} - \boldsymbol{\mu})\big\}. \end{aligned}$$

Thus the global maximal likelihood (ML) estimator of $\boldsymbol{\mu}$ is trivially the sample mean $\bar{\mathbf{x}}$, and hence $\boldsymbol{\theta} = (\mathbf{A}, \boldsymbol{\Psi})$ can be estimated by maximizing

$$l(\boldsymbol{\theta}) = -\frac{n}{2}\big\{\ln|\boldsymbol{\Sigma}| + \text{tr}(\boldsymbol{\Sigma}^{-1}\mathbf{S})\big\}. \tag{3}$$

This amounts to solving the following simultaneous equations (Jöreskog 1967):

$$\partial l(\boldsymbol{\theta})/\partial \mathbf{A} = -\frac{n}{2}\big[\boldsymbol{\Sigma}^{-1}\mathbf{A} - \boldsymbol{\Sigma}^{-1}\mathbf{S}\boldsymbol{\Sigma}^{-1}\mathbf{A}\big] = 0, \tag{4}$$

$$\partial l(\boldsymbol{\theta})/\partial \boldsymbol{\Psi} = -\frac{n}{2}\text{diag}(\boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1}\mathbf{S}\boldsymbol{\Sigma}^{-1}) = 0. \tag{5}$$

Both equations can be further simplified into the following equivalent form:

$$\mathbf{A} = \mathbf{S}\boldsymbol{\Sigma}^{-1}\mathbf{A}, \tag{6}$$

$$\boldsymbol{\Psi} = \text{diag}\big\{\mathbf{S} - \mathbf{A}\mathbf{A}'\big\}. \tag{7}$$

Equation (6) is obviously equivalent to (4). The equivalence between (5) and (7) is a result of (6) and the definition of the matrix $\boldsymbol{\Sigma}$. This fact will be proved in Appendix 7.1.

Since it is difficult to solve (6) and (7) explicitly, it is necessary to use iterative procedures to maximize $l(\boldsymbol{\theta})$. In the remainder of this section, we review three previously known algorithms. Technical details are included here for later use.

2.2 Lawley (1940)'s simple iteration algorithm

By definition of $\boldsymbol{\Sigma}$, we have the following identity (see also Lawley and Maxwell 1971, p. 27):

$$\boldsymbol{\Sigma}^{-1}\mathbf{A} = \boldsymbol{\Psi}^{-1}\mathbf{A}\big(\mathbf{I} + \mathbf{A}'\boldsymbol{\Psi}^{-1}\mathbf{A}\big)^{-1}. \tag{8}$$

Using this identity, (6) can be written as

$$\mathbf{A}\big(\mathbf{I} + \mathbf{A}'\boldsymbol{\Psi}^{-1}\mathbf{A}\big) = \mathbf{S}\boldsymbol{\Psi}^{-1}\mathbf{A}. \tag{9}$$

Let $\widetilde{\mathbf{A}} \triangleq \boldsymbol{\Psi}^{-1/2}\mathbf{A}$, $\widetilde{\mathbf{S}} \triangleq \boldsymbol{\Psi}^{-1/2}\mathbf{S}\boldsymbol{\Psi}^{-1/2}$. Then (9) can be written in the following equivalent form

$$\widetilde{\mathbf{A}}\big(\mathbf{I}+\widetilde{\mathbf{A}}'\widetilde{\mathbf{A}}\big)=\widetilde{\mathbf{S}}\widetilde{\mathbf{A}}. \tag{10}$$

Let $(\lambda_i, \mathbf{u}_i)$ be the eigenvalue-eigenvector pairs of $\widetilde{\mathbf{S}}$ with $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$ and $\boldsymbol{\Lambda}_q = \operatorname{diag}(\lambda_1, \lambda_2, \ldots, \lambda_q)$, $\mathbf{U}_q = (\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_q)$. If $\boldsymbol{\Lambda}_q > \mathbf{I}$, then

$$\widetilde{\mathbf{A}} = \mathbf{U}_q(\boldsymbol{\Lambda}_q - \mathbf{I})^{1/2} \tag{11}$$

will be a solution of (10), see also Jöreskog (1967 (16)).

Now Lawley (1940)'s algorithm recursively computes $\{\mathbf{A}^{(t)}, \boldsymbol{\Psi}^{(t)}\}_{t=0}^{\infty}$ by using the following two steps:

- Step 1. Given $\boldsymbol{\Psi}^{(t)}$, set $\mathbf{A}^{(t+1)} = [\boldsymbol{\Psi}^{(t)}]^{1/2} \cdot \mathbf{U}_q^{(t)}(\boldsymbol{\Lambda}_q^{(t)} - \mathbf{I})^{1/2}$, where, as in last paragraph, $\boldsymbol{\Lambda}_q^{(t)} = \operatorname{diag}(\lambda_1^{(t)}, \lambda_2^{(t)}, \ldots, \lambda_q^{(t)})$, $\mathbf{U}_q^{(t)} = (\mathbf{u}_1^{(t)}, \mathbf{u}_2^{(t)}, \ldots, \mathbf{u}_q^{(t)})$, and $(\lambda_i^{(t)}, \mathbf{u}_i^{(t)})$, $i = 1, \ldots, d$, are eigenvalue-eigenvector pairs of the matrix $[\boldsymbol{\Psi}^{(t)}]^{-1/2}\mathbf{S}[\boldsymbol{\Psi}^{(t)}]^{-1/2}$ with $\lambda_1^{(t)} \geq \lambda_2^{(t)} \geq \cdots \geq \lambda_d^{(t)}$.
- Step 2. Given $\mathbf{A}^{(t+1)}$, update $\boldsymbol{\Psi}$ using (7), i.e., set $\boldsymbol{\Psi}^{(t+1)} = \operatorname{diag}(\mathbf{S} - \mathbf{A}^{(t+1)}[\mathbf{A}^{(t+1)}]')$.

It is commented in Lawley and Maxwell (1971, p. 30) that there is no guarantee that the iteration of the above two steps can converge (i.e., the first problem). In addition, the inequality $\boldsymbol{\Lambda}_q > \mathbf{I}$ may not hold in practice, especially when q is large (Jöreskog 1967) and thus the second problem may occur.

2.3 EM type algorithms

If the parameters are known to be $(\mathbf{A}_0, \boldsymbol{\Psi}_0, \boldsymbol{\mu}_0)$, then we can compute

$$\mathbf{y}_j|\mathbf{x}_j \sim \mathcal{N}\big(\mathbf{A}_0'\boldsymbol{\Sigma}_0^{-1}(\mathbf{x}_j - \boldsymbol{\mu}_0), (\mathbf{I} + \mathbf{A}_0'\boldsymbol{\Psi}_0^{-1}\mathbf{A}_0)^{-1}\big),$$

where $\boldsymbol{\Sigma}_0 = \boldsymbol{\Psi}_0 + \mathbf{A}_0\mathbf{A}_0'$. Using this distribution, we can compute the expectation of any function $f(\mathbf{y}_j)$, which is denoted as $E(f(\mathbf{y}_j)|\mathbf{A}_0, \boldsymbol{\Psi}_0, \boldsymbol{\mu}_0)$. In particular, we have

$$E(\mathbf{y}_j|\mathbf{A}_0, \boldsymbol{\Psi}_0, \boldsymbol{\mu}_0) = \mathbf{A}'\boldsymbol{\Sigma}^{-1}(\mathbf{x}_j - \boldsymbol{\mu}_0), \tag{12}$$

$$\begin{aligned} &E(\mathbf{y}_j\mathbf{y}_j'|\mathbf{A}_0, \boldsymbol{\Psi}_0, \boldsymbol{\mu}_0) \\ &\quad = (\mathbf{I} + \mathbf{A}_0'\boldsymbol{\Psi}_0^{-1}\mathbf{A}_0)^{-1} \\ &\qquad + \mathbf{A}_0'\boldsymbol{\Sigma}_0^{-1}(\mathbf{x}_j - \boldsymbol{\mu}_0)(\mathbf{x}_j - \boldsymbol{\mu}_0)'\boldsymbol{\Sigma}_0^{-1}\mathbf{A}_0. \end{aligned} \tag{13}$$

As is well known (Dempster et al. 1977; Meng and van Dyk 1997), any EM type algorithm is implemented by choosing a particular data augmentation, and then the complete data likelihood is imputed (the so-called E-step) and maximized (i.e., M- or CM step). For our model, it is natural to consider the augmented data $(\mathcal{X}_n, \mathcal{Y}_n) = \{\mathbf{x}_j, \mathbf{y}_j\}_{j=1}^n$. The complete data log-likelihood is

$$\begin{aligned} l_c(\mathbf{A}, \boldsymbol{\Psi}, \boldsymbol{\mu}; \mathcal{Y}_n) &= -\frac{n}{2}\ln|\boldsymbol{\Psi}| \\ &\quad - \frac{1}{2}\sum_{j=1}^n\big\{\mathbf{y}_j'\mathbf{y}_j + (\mathbf{x}_j - \mathbf{A}\mathbf{y}_j - \boldsymbol{\mu})'\boldsymbol{\Psi}^{-1}(\mathbf{x}_j - \mathbf{A}\mathbf{y}_j - \boldsymbol{\mu})\big\}. \end{aligned}$$

Taking expectation, we obtain

$$E\{l_c(\mathbf{A}, \boldsymbol{\Psi}, \boldsymbol{\mu}; \mathcal{Y}_n)|\mathbf{A}_0, \boldsymbol{\Psi}_0, \boldsymbol{\mu}_0\} = Q(\mathbf{A}, \boldsymbol{\Psi}|\mathbf{A}_0, \boldsymbol{\Psi}_0) + R,$$

where

$$\begin{aligned} &Q(\mathbf{A}, \boldsymbol{\Psi}|\mathbf{A}_0, \boldsymbol{\Psi}_0) \\ &\quad = -\frac{n}{2}\ln|\boldsymbol{\Psi}| - \frac{n}{2}\operatorname{tr}\big\{\boldsymbol{\Psi}^{-1}\mathbf{S} - 2\boldsymbol{\Psi}^{-1}\mathbf{A}\mathbf{A}_0'\boldsymbol{\Sigma}_0^{-1}\mathbf{S} \\ &\qquad + (\mathbf{I} + \mathbf{A}'\boldsymbol{\Psi}^{-1}\mathbf{A})\big[(\mathbf{I} + \mathbf{A}_0'\boldsymbol{\Psi}_0^{-1}\mathbf{A}_0)^{-1} \\ &\qquad + \mathbf{A}_0'\boldsymbol{\Sigma}_0^{-1}\mathbf{S}\boldsymbol{\Sigma}_0^{-1}\mathbf{A}_0\big]\big\}, \\ &-2R/n = (\bar{\mathbf{x}} - \boldsymbol{\mu}_0)'\boldsymbol{\Sigma}_0^{-1}\mathbf{A}_0(\mathbf{I} + \mathbf{A}'\boldsymbol{\Psi}^{-1}\mathbf{A})\mathbf{A}_0'\boldsymbol{\Sigma}_0^{-1}(\bar{\mathbf{x}} - \boldsymbol{\mu}_0) \\ &\qquad\qquad + (\bar{\mathbf{x}} - \boldsymbol{\mu})'\boldsymbol{\Psi}^{-1}(\bar{\mathbf{x}} - \boldsymbol{\mu}) \\ &\qquad\qquad - 2(\bar{\mathbf{x}} - \boldsymbol{\mu})'\boldsymbol{\Psi}^{-1}\mathbf{A}\mathbf{A}_0'\boldsymbol{\Sigma}_0^{-1}(\bar{\mathbf{x}} - \boldsymbol{\mu}_0). \end{aligned}$$

Notice that if $\boldsymbol{\mu}_0 = \bar{\mathbf{x}}$, then the remainder R is maximized at $\boldsymbol{\mu} = \bar{\mathbf{x}}$. Since $\bar{\mathbf{x}}$ is the global maximal likelihood estimator of $\boldsymbol{\mu}$, we should set $\boldsymbol{\mu}^{(t)}$ to the fixed value $\bar{\mathbf{x}}$ in the following implementation of EM type algorithms.

Since the function $Q(\mathbf{A}, \boldsymbol{\Psi}|\mathbf{A}_0, \boldsymbol{\Psi}_0)$ is maximized at

$$\mathbf{A} = \mathbf{S}\boldsymbol{\Psi}_0^{-1}\mathbf{A}_0\big(\mathbf{I} + \mathbf{A}_0'\boldsymbol{\Sigma}_0^{-1}\mathbf{S}\boldsymbol{\Psi}_0^{-1}\mathbf{A}_0\big)^{-1}, \tag{14}$$

$$\boldsymbol{\Psi} = \operatorname{diag}\big(\mathbf{S} - \mathbf{S}\boldsymbol{\Sigma}_0^{-1}\mathbf{A}_0\mathbf{A}'\big), \tag{15}$$

we obtain Rubin and Thayer (1982)'s EM algorithm for FA:

EM: For given $\mathbf{A}^{(t)}, \boldsymbol{\Psi}^{(t)}$, compute

$$\begin{aligned} &\mathbf{A}^{(t+1)} \\ &\quad = \mathbf{S}[\boldsymbol{\Psi}^{(t)}]^{-1}\mathbf{A}^{(t)}\big(\mathbf{I} + [\mathbf{A}^{(t)}]'[\boldsymbol{\Sigma}^{(t)}]^{-1}\mathbf{S}[\boldsymbol{\Psi}^{(t)}]^{-1}\mathbf{A}^{(t)}\big)^{-1}, \\ &\boldsymbol{\Psi}^{(t+1)} \\ &\quad = \operatorname{diag}\big(\mathbf{S} - \mathbf{S}[\boldsymbol{\Sigma}^{(t)}]^{-1}\mathbf{A}^{(t)}[\mathbf{A}^{(t+1)}]'\big), \end{aligned}$$

where $\boldsymbol{\Sigma}^{(t)} = \boldsymbol{\Psi}^{(t)} + \mathbf{A}^{(t)}[\mathbf{A}^{(t)}]'$.

Using (8), the computation of $[\boldsymbol{\Sigma}^{(t)}]^{-1}$ in both formulas can be reduced. In Appendix 8.1, we describe an efficient implementation to be used in our simulation study in Sect. 5. Notice also that the function Q satisfies the continuous condition in Wu (1983, Theorem 2) and therefore all limiting points of the above EM algorithm are guaranteed to be stationary points of l.

As a modification on the above EM, Liu and Rubin (1998) proposed the following ECME type algorithm:

ECME1: For given $\mathbf{A}^{(t)}, \boldsymbol{\Psi}^{(t)}$,

- CM-step 1: set $\mathbf{A}^{(t+1)} = \mathbf{S}[\boldsymbol{\Psi}^{(t)}]^{-1}\mathbf{A}^{(t)}(\mathbf{I} + [\mathbf{A}^{(t)}]' \times [\boldsymbol{\Sigma}^{(t)}]^{-1}\mathbf{S}[\boldsymbol{\Psi}^{(t)}]^{-1}\mathbf{A}^{(t)})^{-1}$ which maximizes the function $Q(\mathbf{A}, \boldsymbol{\Psi}^{(t)}|\mathbf{A}^{(t)}, \boldsymbol{\Psi}^{(t)})$.
- CM-step 2: compute $\boldsymbol{\Psi}^{(t+1)} = \operatorname{Arg\,max}_{\boldsymbol{\Psi}} l(\mathbf{A}^{(t+1)}, \boldsymbol{\Psi})$ using Newton-Raphson algorithm. See (3) for the definition of the function $l(\cdot)$.

Here, we use $\operatorname{Arg\,max}_x f(x)$ to denote a point at which the function $f(x)$ is maximized. For CM-step 2, it is useful to introduce an exponential transformation for uniqueness to avoid constraint optimization. The corresponding gradient and Hessian expressions are available from Liu and Rubin (1998) and hence omitted here.

3 The ECME2 algorithm

Unlike ECME1, which performs CMQ over $\mathbf{A}$ followed by CML over $\boldsymbol{\Psi}$, ECME2 algorithm proposed in this section performs CML over $\mathbf{A}$ followed by CMQ over $\boldsymbol{\Psi}$. In detail, given an initial $\boldsymbol{\Psi}^{(0)}$, our ECME2 recursively computes the sequence $\{\mathbf{A}^{(t)}, \boldsymbol{\Psi}^{(t)}\}_{t=1}^{\infty}$ according to the following two steps:

- CM-step 1: Given $\boldsymbol{\Psi}^{(t)}$, compute $\mathbf{A}^{(t+1)} = \operatorname{Arg\,max}_{\mathbf{A}} l(\mathbf{A}, \boldsymbol{\Psi}^{(t)})$.
- CM-step 2: Given $(\mathbf{A}^{(t+1)}, \boldsymbol{\Psi}^{(t)})$, compute $\boldsymbol{\Psi}^{(t+1)} = \operatorname{Arg\,max}_{\boldsymbol{\Psi}} Q(\mathbf{A}^{(t+1)}, \boldsymbol{\Psi}|\mathbf{A}^{(t+1)}, \boldsymbol{\Psi}^{(t)})$.

Instead of using traditional search procedures, the optimization problem in CM-step 1 is solved explicitly in Sect. 3.1. The maximization in CM-step 2 can be easily solved by setting $\mathbf{A}_0 = \mathbf{A}^{(t+1)}$, $\boldsymbol{\Psi}_0 = \boldsymbol{\Psi}^{(t)}$ in (15). We show in Sect. 3.2 the resulting formula can be further simplified.

3.1 The maximization in the first CM-step

Let $\widetilde{\mathbf{A}} \triangleq [\boldsymbol{\Psi}^{(t)}]^{-1/2}\mathbf{A}$, $\widetilde{\mathbf{S}} \triangleq [\boldsymbol{\Psi}^{(t)}]^{-1/2}\mathbf{S}[\boldsymbol{\Psi}^{(t)}]^{-1/2}$. As we have shown in Sect. 2.2, any matrix $\mathbf{A}$ that maximizes the function $l(\mathbf{A}, \boldsymbol{\Psi}^{(t)})$ should satisfy

$$\widetilde{\mathbf{A}}\big(\mathbf{I} + \widetilde{\mathbf{A}}'\widetilde{\mathbf{A}}\big) = \widetilde{\mathbf{S}}\widetilde{\mathbf{A}}, \tag{16}$$

which is identically (10). Thus, to maximize $l(\mathbf{A}, \boldsymbol{\Psi}^{(t)})$, we first find all solutions to (16), and then choose the best one.

Let $\widetilde{\mathbf{A}} = \mathbf{U}_{q'}\mathbf{D}\mathbf{V}$ be the singular-value decomposition of $\widetilde{\mathbf{A}}$, where the $d \times q'$ matrix $\mathbf{U}_{q'}$ satisfies $\mathbf{U}_{q'}'\mathbf{U}_{q'} = \mathbf{I}$, $\mathbf{V}$ is a $q' \times q$ matrix satisfying $\mathbf{V}\mathbf{V}' = \mathbf{I}$, $\mathbf{D} = \operatorname{diag}(d_1, d_2, \ldots, d_{q'})$ is diagonal with $d_1 \geq d_2 \geq \cdots \geq d_{q'} > 0$, the integer $q' \leq q$ is of course the rank of $\widetilde{\mathbf{A}}$. With this decomposition, (16) is equivalent to

$$\mathbf{U}_{q'}(\mathbf{I} + \mathbf{D}^2) = \widetilde{\mathbf{S}}\mathbf{U}_{q'}. \tag{17}$$

This shows that the i-th column $\mathbf{u}_i$ of $\mathbf{U}_{q'}$ is an eigenvector of the matrix $\widetilde{\mathbf{S}}$ with $\lambda_i = 1 + d_i^2$ being the corresponding eigenvalue. Conversely, if $(\mathbf{u}_i, 1 + d_i^2)$, $i = 1, \ldots, q'$, are eigenvector-eigenvalue pairs of the matrix $\widetilde{\mathbf{S}}$, then $\widetilde{\mathbf{A}} = \mathbf{U}_{q'}\mathbf{D}\mathbf{V}$ solves (16), provided that the $q' \times q$ matrix $\mathbf{V}$ satisfies $\mathbf{V}\mathbf{V}' = \mathbf{I}$. Thus, to maximize $l(\mathbf{A}, \boldsymbol{\Psi}^{(t)})$, we need to consider the decomposition $\widetilde{\mathbf{S}} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}'$, where $\mathbf{U} = (\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_d)$ is a $d \times d$ orthogonal matrix, $\boldsymbol{\Lambda} = \operatorname{diag}(\lambda_1, \lambda_2, \ldots, \lambda_d)$ is diagonal. Since

$$\begin{aligned}&\mathbf{I} + \widetilde{\mathbf{A}}\widetilde{\mathbf{A}}' \\ &\quad = \mathbf{I} + \mathbf{U}_{q'}\mathbf{D}^2\mathbf{U}_{q'}' = \mathbf{U} \cdot \operatorname{diag}(\lambda_1, \ldots, \lambda_{q'}, 1, \ldots, 1) \cdot \mathbf{U}',\end{aligned}$$

we have

$$\begin{aligned}&-\frac{2}{n} \cdot l(\mathbf{A}, \boldsymbol{\Psi}^{(t)}) \\ &\quad = \ln|\boldsymbol{\Sigma}| + \operatorname{tr}(\boldsymbol{\Sigma}^{-1}\mathbf{S}) \\ &\quad = \ln|\boldsymbol{\Psi}^{(t)}| + \ln|\mathbf{I} + \widetilde{\mathbf{A}}\widetilde{\mathbf{A}}'| + \operatorname{tr}\big\{(\mathbf{I} + \widetilde{\mathbf{A}}\widetilde{\mathbf{A}}')^{-1}\widetilde{\mathbf{S}}\big\} \\ &\quad = \ln|\boldsymbol{\Psi}^{(t)}| + \sum_{i=1}^{q'}(\ln\lambda_i - \lambda_i + 1) + \sum_{k=1}^{d}\lambda_k.\end{aligned}$$

The sum $\ln|\boldsymbol{\Psi}^{(t)}| + \sum_{k=1}^{d}\lambda_k$ is obviously a constant in our problem. So it suffices to minimize the term $\sum_{i=1}^{q'}(\ln\lambda_i - \lambda_i + 1)$ by sorting the eigenvalues of $\widetilde{\mathbf{S}}$ in a suitable order and then determining the best q'. Since the function $f(\lambda) = \ln\lambda - \lambda + 1$ is negative and strictly decreasing in the interval $(1, \infty)$, we may assume $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$. The optimal q' is defined as follows: If $\lambda_q > 1$, then we set $q' = q$; if $\lambda_q \leq 1$, q' is the unique integer satisfying $\lambda_{q'} > 1 \geq \lambda_{q'+1}$. Now the function $l(\mathbf{A}, \boldsymbol{\Psi}^{(t)})$ is maximized at

$$\mathbf{A}^{(t+1)} = [\boldsymbol{\Psi}^{(t)}]^{1/2}\mathbf{U}_{q'}\big(\boldsymbol{\Lambda}_{q'} - \mathbf{I}\big)^{1/2}\mathbf{V}, \tag{18}$$

where $\boldsymbol{\Lambda}_{q'} = \operatorname{diag}(\lambda_1, \lambda_2, \ldots, \lambda_{q'})$. The matrix $\mathbf{V}$ can be arbitrarily chosen except for the requirement $\mathbf{V}\mathbf{V}' = \mathbf{I}$, and it has no effect on the value of $l(\mathbf{A}^{(t+1)}, \boldsymbol{\Psi}^{(t)})$. Clearly, as $\boldsymbol{\Lambda}_{q'} > \mathbf{I}$, it is impossible for $\mathbf{A}^{(t+1)}$ in (18) to contain any imaginary number and the second problem mentioned in Sect. 1 disappears.

3.2 The maximization in the second CM-step

Since $l(\mathbf{A}, \boldsymbol{\Psi}^{(t)})$ is maximized at $\mathbf{A}^{(t+1)}$, $\mathbf{A}^{(t+1)}$ in (18) must satisfy (6), i.e.,

$$\mathbf{A}^{(t+1)} = \mathbf{S}(\boldsymbol{\Psi}^{(t)} + \mathbf{A}^{(t+1)}[\mathbf{A}^{(t+1)}]')^{-1}\mathbf{A}^{(t+1)}. \tag{19}$$

By direct computation, or using the identities (19) and (8), one can easily show that (14) is satisfied for $\boldsymbol{\Psi}_0 = \boldsymbol{\Psi}^{(t)}$, $\mathbf{A} = \mathbf{A}_0 = \mathbf{A}^{(t+1)}$. According to Sect. 2.3, the same setting in (15) will give us

$$\begin{aligned}&\boldsymbol{\Psi}^{(t+1)}\\&\quad=\operatorname*{Arg\,max}_{\boldsymbol{\Psi}} Q(\mathbf{A}^{(t+1)},\boldsymbol{\Psi}|\mathbf{A}^{(t+1)},\boldsymbol{\Psi}^{(t)})\\&\quad=\operatorname{diag}\big(\mathbf{S}-\mathbf{S}(\boldsymbol{\Psi}^{(t)}+\mathbf{A}^{(t+1)}[\mathbf{A}^{(t+1)}]')^{-1}\mathbf{A}^{(t+1)}[\mathbf{A}^{(t+1)}]'\big).\end{aligned}$$

Using the identity (19), we obtain

$$\boldsymbol{\Psi}^{(t+1)}=\operatorname{diag}\big(\mathbf{S}-\mathbf{A}^{(t+1)}[\mathbf{A}^{(t+1)}]'\big). \tag{20}$$

Observing the similarity between (7) and (20), one may deduce from the equivalence between (5) and (7) that $\boldsymbol{\Psi}^{(t+1)}$ is also a solution of (5). But this is not correct, because different $\boldsymbol{\Psi}$ matrices are used in (19) and (20).

3.3 Practical consideration

Instead of computing $\mathbf{A}^{(t+1)}$ by (18) explicitly, we can simply compute

$$\widetilde{\mathbf{A}}^{(t+1)}=\mathbf{U}_{q'}(\boldsymbol{\Lambda}_{q'}-\mathbf{I})^{1/2}\mathbf{V}, \tag{21}$$

since we can use the notation of $\widetilde{\mathbf{S}}$ and $\widetilde{\mathbf{A}}^{(t+1)}$ to write (20) as

$$\boldsymbol{\Psi}^{(t+1)}=\operatorname{diag}\big(\widetilde{\mathbf{S}}-\widetilde{\mathbf{A}}^{(t+1)}[\widetilde{\mathbf{A}}^{(t+1)}]'\big)\boldsymbol{\Psi}^{(t)}. \tag{22}$$

In practice, for numerical stability (as computation of $\widetilde{\mathbf{S}}$ involves $[\boldsymbol{\Psi}^{(t)}]^{-1}$), we follow Jöreskog (1967) to restrict $\boldsymbol{\Psi}\geq\eta>0$, which can be easily incorporated into ECME2. At iteration $t+1$, we enforce ${\psi_i}^{(t+1)}=\eta$ if ${\psi_i}^{(t+1)}<\eta$ and update ψ_i otherwise, under which l is guaranteed to increase or at least not to decrease (one can easily verify that $Q(\psi_i)$ is unimodal in the interval $\psi_i>0$).

It can be verified that ECME2 satisfies the *space filling* condition of ECME (Liu 1994) and therefore the limiting point of ECME2 is guaranteed to be a stationary point of l. As a result, the first problem mentioned in Sect. 1 also disappears.

3.4 ECME2 vs. simple iteration algorithm

By inspection of (11) and (7) in simple iteration algorithm and (18) and (20) in ECME2, ECME2 looks very similar to Lawley (1940)'s simple iteration algorithm:

(1) Equation (7) is the same as (20)
(2) Equation (11) can be viewed as a special case of (18) when $q'=q$ and $\mathbf{V}$ is the first q' rows of $\mathbf{I}$.

However, ECME2 frees from an ad-hoc assumption: $\boldsymbol{\Lambda}_q>\mathbf{I}$ (detailed in Sect. 2.2) and more importantly, we actually find a way to prove monotone convergence of the simple iteration algorithm, which, to our knowledge, has not been proved so far.

ECME2 replaces the CMQ step for $\mathbf{A}$ in EM by a CML step, where closed form expression is available. Intuitively, ECME2 should be faster than EM, at least assessed by the number of iterations. Further, it is natural to ask whether it is possible to replace the CMQ step for $\boldsymbol{\Psi}$ in EM by a CML step, which also has closed form expression. Such resulting algorithm would be a CM algorithm (Meng and Rubin 1993), convergence of which is quadratic (Liu 1994). Unfortunately, it seems impossible to solve $\boldsymbol{\Psi}$ from (5) with $\mathbf{A}$ fixed. As a compromise, numerical optimization methods such as Newton-Raphson can be considered to perform this CML step. In fact, such a step is the same as the CM-step 2 in ECME1. In Sect. 4 we develop a more tempting CM algorithm since it has (1) quadratic convergence; (2) monotone convergence. Rather than following tradition to solve $\boldsymbol{\Psi}$ numerically, we focus on analytically solving ψ_i, the i-th element of $\boldsymbol{\Psi}$, keeping $\mathbf{A}$ and the other elements of $\boldsymbol{\Psi}$ fixed and we can obtain the closed form expression of ψ_i. Combining the CML step to update $\mathbf{A}$, we hence obtain a CM algorithm.

4 The CM algorithm

Let $\boldsymbol{\Psi}^{(t)}=\operatorname{diag}(\psi_1^{(t)},\psi_2^{(t)},\ldots,\psi_d^{(t)})$, $\boldsymbol{\Psi}_i^{(t)}\triangleq\operatorname{diag}(\psi_1^{(t+1)},\ldots,\psi_{i-1}^{(t+1)},\psi_i,\psi_{i+1}^{(t)},\ldots,\psi_d^{(t)})$. Given an initial $\boldsymbol{\Psi}^{(0)}$, our CM algorithm recursively does the following two steps for $t\geq 0$:

- CM-step 1: Given $\boldsymbol{\Psi}^{(t)}$, compute $\mathbf{A}^{(t+1)}=\operatorname{Arg\,max}_{\mathbf{A}} l(\mathbf{A},\boldsymbol{\Psi}^{(t)})$, $t\geq 0$;
- CM-step 2: Compute $\psi_i^{(t+1)}=\operatorname{Arg\,max}_{\psi_i} l(\mathbf{A}^{(t+1)},\boldsymbol{\Psi}_i^{(t)})$ for $i=1,2,\ldots,d$.

The first CM-step has been solved in Sect. 3.1. So only the second CM-step is considered below. Since $l(\mathbf{A}^{(t+1)},\boldsymbol{\Psi}_i^{(t)})$ is a function of ψ_i, it is denoted as $\bar{l}(\psi_i)$ for simplicity.

4.1 The maximization in the second CM-step

Let $\boldsymbol{\Sigma}_i=\boldsymbol{\Psi}_i^{(t)}+\mathbf{A}^{(t+1)}[\mathbf{A}^{(t+1)}]'$. To maximize $\bar{l}(\psi_i)$, we need to solve the equation

$$-\frac{2}{n}\cdot\frac{d\bar{l}(\psi_i)}{d\psi_i}=\big(\boldsymbol{\Sigma}_i^{-1}-\boldsymbol{\Sigma}_i^{-1}\mathbf{S}\boldsymbol{\Sigma}_i^{-1}\big)_{ii}=0, \tag{23}$$

where $(\cdot)_{ii}$ denotes the (i,i)-th element of the matrix in the parenthesis. Using the notation

$$\widetilde{\mathbf{S}}\triangleq[\boldsymbol{\Psi}^{(t)}]^{-1/2}\mathbf{S}[\boldsymbol{\Psi}^{(t)}]^{-1/2},$$
$$\widetilde{\boldsymbol{\Sigma}}_i\triangleq[\boldsymbol{\Psi}^{(t)}]^{-1/2}\boldsymbol{\Sigma}_i[\boldsymbol{\Psi}^{(t)}]^{-1/2},$$

we write (23) in the following equivalent form

$$\big(\widetilde{\boldsymbol{\Sigma}}_i^{-1}-\widetilde{\boldsymbol{\Sigma}}_i^{-1}\widetilde{\mathbf{S}}\widetilde{\boldsymbol{\Sigma}}_i^{-1}\big)_{ii}=0. \tag{24}$$

To solve this equation, the most difficult part is the term $\widetilde{\boldsymbol{\Sigma}}_i^{-1}$, which contains the unique indeterminate ψ_i. To manipulate this matrix, it is helpful to introduce the following notations:

$$\begin{aligned}
&\widetilde{\mathbf{A}} \triangleq [\boldsymbol{\Psi}^{(t)}]^{-1/2}\mathbf{A}^{(t+1)},\\
&\widetilde{\boldsymbol{\Psi}}_i \triangleq \boldsymbol{\Psi}_i^{(t)}[\boldsymbol{\Psi}^{(t)}]^{-1} \qquad (25)\\
&\quad = \mathbf{I} + \mathrm{diag}(\omega_1^{(t+1)}, \ldots, \omega_{i-1}^{(t+1)}, \omega_i, 0, \ldots, 0),
\end{aligned}$$

where

$$\begin{aligned}
&\omega_i \triangleq -1 + \psi_i/\psi_i^{(t)};\\
&\omega_k^{(t+1)} \triangleq -1 + \psi_k^{(t+1)}/\psi_k^{(t)}, \quad k = 1, \ldots, i-1.
\end{aligned}$$

Let $\mathbf{e}_i$ be the i-th column of the $d \times d$ identity matrix. Then

$$\widetilde{\boldsymbol{\Sigma}}_i = \widetilde{\boldsymbol{\Psi}}_i + \widetilde{\mathbf{A}}\widetilde{\mathbf{A}}' = \omega_i\, \mathbf{e}_i\mathbf{e}_i' + \mathbf{B}_i, \qquad (26)$$

where, by (25),

$$\mathbf{B}_i = \sum\nolimits_{k=1}^{i-1} \omega_k^{(t+1)}\, \mathbf{e}_k\mathbf{e}_k' + \mathbf{I} + \widetilde{\mathbf{A}}\widetilde{\mathbf{A}}'. \qquad (27)$$

Now the following Proposition 1 provides a way to invert the matrix $\widetilde{\boldsymbol{\Sigma}}_i$. Here and in the sequel we write $\mathbf{C} > 0$ for the fact that the matrix $\mathbf{C}$ is symmetric and positive definite.

Proposition 1 *Suppose* $\mathbf{C} > 0$, *ω is a real number such that* $1 + \omega\mathbf{e}_i'\mathbf{C}^{-1}\mathbf{e}_i \neq 0$. *Then*

$$\left(\omega\mathbf{e}_i\mathbf{e}_i' + \mathbf{C}\right)^{-1} = \mathbf{C}^{-1} - \frac{\omega\mathbf{C}^{-1}\mathbf{e}_i\mathbf{e}_i'\mathbf{C}^{-1}}{1 + \omega\mathbf{e}_i'\mathbf{C}^{-1}\mathbf{e}_i}. \qquad (28)$$

Suppose $\mathbf{B}_i$ and ω_i satisfies *Condition I:* (a) $\mathbf{B}_i > 0$; (b) $1 + \omega_i\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i > 0$. Then, by setting $\omega = \omega_i$, $\mathbf{C} = \mathbf{B}_i$ in (28), we obtain

$$\widetilde{\boldsymbol{\Sigma}}_i^{-1} = \mathbf{B}_i^{-1} - \omega_i\mathbf{B}_i^{-1}\mathbf{e}_i\mathbf{e}_i'\mathbf{B}_i^{-1}/(1 + \omega_i\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i). \qquad (29)$$

Using this equation, $(\widetilde{\boldsymbol{\Sigma}}_i^{-1} - \widetilde{\boldsymbol{\Sigma}}_i^{-1}\widetilde{\mathbf{S}}\widetilde{\boldsymbol{\Sigma}}_i^{-1})_{ii}$ can be expressed explicitly as a function of ω_i, i.e.,

$$\begin{aligned}
&(\widetilde{\boldsymbol{\Sigma}}_i^{-1} - \widetilde{\boldsymbol{\Sigma}}_i^{-1}\widetilde{\mathbf{S}}\widetilde{\boldsymbol{\Sigma}}_i^{-1})_{ii}\\
&\quad = \mathbf{e}_i'(\widetilde{\boldsymbol{\Sigma}}_i^{-1} - \widetilde{\boldsymbol{\Sigma}}_i^{-1}\widetilde{\mathbf{S}}\widetilde{\boldsymbol{\Sigma}}_i^{-1})\mathbf{e}_i\\
&\quad = \frac{(\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i)^2\omega_i + \mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i - \mathbf{e}_i'\mathbf{B}_i^{-1}\widetilde{\mathbf{S}}\mathbf{B}_i^{-1}\mathbf{e}_i}{(1 + \omega_i\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i)^2}.
\end{aligned}$$

It can be shown in Sect. 4.2 that (24) has the following unique solution

$$\omega_i^{(t+1)} = (\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i)^{-2}(\mathbf{e}_i'\mathbf{B}_i^{-1}\widetilde{\mathbf{S}}\mathbf{B}_i^{-1}\mathbf{e}_i - \mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i) \qquad (30)$$

in the interval $(-[\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i]^{-1}, \infty)$. Correspondingly,

$$\psi_i^{(t+1)} = (\omega_i^{(t+1)} + 1)\psi_i^{(t)} \qquad (31)$$

is a solution of (23).

4.2 When will *Condition I* be satisfied

Pick a very small $\eta > 0$. Consider the following modified version of the second CM-step:

- CM-step 2′: For $i = 1, 2, \ldots, d$, compute $\omega_i^{(t+1)}$ according to (30), then set $\psi_i^{(t+1)} = \eta$ if $\omega_i^{(t+1)} \leq -1$, and $\psi_i^{(t+1)} = (\omega_i^{(t+1)} + 1)\psi_i^{(t)}$ otherwise.

In this way, the $\boldsymbol{\Psi}$ matrix is always positive definite. Consequently, $\omega_k^{(t+1)} > -1$ for $k = 1, \ldots, i-1$. Thus

$$\mathbf{B}_i = \mathrm{diag}(\omega_1^{(t+1)} + 1, \ldots, \omega_{i-1}^{(t+1)} + 1, 1, \ldots, 1) + \widetilde{\mathbf{A}}\widetilde{\mathbf{A}}' > 0,$$

i.e., *condition* (a) always holds and $\omega_i^{(t+1)}$ in (30) can always be computed.

Since *condition* (a) holds, $\psi_i^{(t+1)}$ is a solution of (23) if *condition* (b) is also satisfied. This is the case if either $\omega_i^{(t+1)} > -1$, or $\mathbf{S}$ is positive definite. The case $\omega_i^{(t+1)} > -1$ is treated in Proposition 2. For the case $\mathbf{S} > 0$, we first note that $\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i > 0$, $\mathbf{e}_i'\mathbf{B}_i^{-1}\widetilde{\mathbf{S}}\mathbf{B}_i^{-1}\mathbf{e}_i > 0$. This implies that

$$\omega_i^{(t+1)} = \frac{\mathbf{e}_i'\mathbf{B}_i^{-1}\widetilde{\mathbf{S}}\mathbf{B}_i^{-1}\mathbf{e}_i}{(\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i)^2} - \frac{1}{\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i} > -\frac{1}{\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i},$$

so that $1 + \omega_i^{(t+1)}\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i > 0$. The assumption $\mathbf{S} > 0$ is very weak and satisfied normally. As a corollary, (23) has a unique solution $\psi_i^{(t+1)}$ in the interval $\psi_i > \psi_i^{(t)}(\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i - 1)/(\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i)$ if $\mathbf{S} > 0$ or $\omega_i^{(t+1)} > -1$ ($\mathbf{S}$ may be semi-positive definite).

Proposition 2 *Suppose* $\mathbf{B}_i > 0$. *Then* $1 + \omega_i\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i > 0$ *for* $\omega_i > -1$.

Now let us check whether $\bar{l}(\psi_i)$ is maximized at $\psi_i^{(t+1)}$. As we have computed in Sect. 4.1,

$$\begin{aligned}
-\frac{2}{n}\frac{d\bar{l}(\psi_i)}{d\psi_i} &= (\widetilde{\boldsymbol{\Sigma}}_i^{-1} - \widetilde{\boldsymbol{\Sigma}}_i^{-1}\widetilde{\mathbf{S}}\widetilde{\boldsymbol{\Sigma}}_i^{-1})_{ii} \cdot \psi_i^{(t)}\\
&= \frac{(\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i)^2\omega_i + \mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i - \mathbf{e}_i'\mathbf{B}_i^{-1}\widetilde{\mathbf{S}}\mathbf{B}_i^{-1}\mathbf{e}_i}{(1 + \omega_i\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i)^2} \cdot \psi_i^{(t)}
\end{aligned}$$

for any $\omega_i > -1/(\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i)$ or, equivalently, $\psi_i > \psi_i^{(t)}(\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i - 1)/(\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i)$. Since $\psi_i^{(t)}$, $\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i$ and $\mathbf{e}_i'\mathbf{B}_i^{-1} \times \widetilde{\mathbf{S}}\mathbf{B}_i^{-1}\mathbf{e}_i$ are all positive, we conclude that $\bar{l}'(\psi_i) > 0$ if $\psi_i < \psi_i^{(t+1)}$, and $\bar{l}'(\psi_i) < 0$ if $\psi_i > \psi_i^{(t+1)}$. Thus, the function $\bar{l}(\psi_i)$ is unimodal in the interval $\psi_i > \psi_i^{(t)}$

$(\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i - 1)/(\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i)$ and reaches its global maximum at the point $\psi_i^{(t+1)}$. In particular, if $\omega_i^{(t+1)} > -1$, then $\psi_i^{(t+1)} = \operatorname{Arg\,max}_{\psi_i \in (0,\infty)} \bar{l}(\psi_i)$.

On the other hand, if $\omega_i^{(t+1)} \le -1$, setting $\psi_i^{(t+1)} = \eta$ in CM-step 2′ is guaranteed to reach the maximum of $\bar{l}(\psi_i)$ in the interval $\psi \ge \eta$ since $\bar{l}(\psi_i)$ is unimodal. In practice, even if $\omega_i^{(t+1)} > -1$ but $0 \le \psi_i^{(t+1)} < \eta$, the unimodal property of $\bar{l}(\psi_i)$ allows us to a further modification of the second CM-step with the advantage of numerical stability (as computation of $\widetilde{\mathbf{S}}$ involves $[\boldsymbol{\Psi}^{(t)}]^{-1}$):

- CM-step 2*: For $i = 1, 2, \ldots, d$, compute $\omega_i^{(t+1)}$ according to (30), then set $\psi_i^{(t+1)} = \max[\eta, (\omega_i^{(t+1)} + 1)\psi_i^{(t)}]$.

Since $\psi_i^{(t)} \ge \eta$ is automatically true, we have $\bar{l}(\psi_i^{(t)}) \le \bar{l}(\eta)$ for the case $(\omega_i^{(t+1)} + 1)\psi_i^{(t)} \le \eta$. Thus this modified CM satisfies the usual monotonic convergence property, i.e., $\bar{l}(\psi_i^{(t+1)}) \ge \bar{l}(\psi_i^{(t)})$. By the general convergence properties of (E)CM in Meng and Rubin (1993), it is guaranteed to converge to a stationary point of l.

4.3 Recursive computation of the matrix $\mathbf{B}_i^{-1}$

Let $\mathbf{B}_1 = \mathbf{I} + \widetilde{\mathbf{A}}^{(t+1)}[\widetilde{\mathbf{A}}^{(t+1)}]'$. Then we have the following recursive relation:

$$\mathbf{B}_{i+1} = \omega_i^{(t+1)}\mathbf{e}_i\mathbf{e}_i' + \mathbf{B}_i, \quad i = 1, 2, \ldots, d-1. \tag{32}$$

If $\psi_i^{(t+1)}$ is restricted to be η, then $\omega_i^{(t+1)} = \eta/\psi_i^{(t)} - 1$. Using Proposition 1, we obtain

$$\mathbf{B}_{i+1}^{-1} = \mathbf{B}_i^{-1} - \omega_i^{(t+1)}\mathbf{B}_i^{-1}\mathbf{e}_i\mathbf{e}_i'\mathbf{B}_i^{-1}/(1 + \omega_i^{(t+1)}\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i). \tag{33}$$

Thus $\mathbf{B}_i^{-1}$ can be recursively computed. The first one is computed by using (18):

$$\mathbf{B}_1^{-1} = \mathbf{U}_{q'}(\boldsymbol{\Lambda}_{q'}^{-1} - \mathbf{I})\mathbf{U}_{q'}' + \mathbf{I}. \tag{34}$$

Note that (21) and (34) simply require at most the first q eigenvalues and eigenvectors of $\widetilde{\mathbf{S}}$.

4.4 On the nature of stationary points

We are interested in the nature of stationary points: $\boldsymbol{\theta}^* = (\mathbf{A}^*, \boldsymbol{\Psi}^*)$ obtained by EM, ECME2 and CM for fitting FA. Although it is unclear whether $\boldsymbol{\theta}^*$ by CM or ECME2 ensure local maxima or not, we argue that CM and ECME2 ensure conditional global maxima and EM does not, i.e., given $\boldsymbol{\Psi}^*$, $\mathbf{A}^*$ by CM and ECME2 is a global maximal point of l while $\mathbf{A}^*$ by EM is not the case. Since CM and ECME2 share (18), which ensure conditional global maxima of l given $\boldsymbol{\Psi}^{(t)}$ at each iteration $t+1$ and therefore it is also true for $\mathbf{A}^*$ and $\boldsymbol{\Psi}^*$. For EM, let t go to infinity in (14) and (15) and we have

$$\mathbf{A}^* = \boldsymbol{\Psi}^*\boldsymbol{\Sigma}^{*-1}\mathbf{S}\boldsymbol{\Psi}^{*-1}\mathbf{A}^*, \tag{35}$$

which can be written as

$$\widetilde{\mathbf{A}}^*(\mathbf{I} + [\widetilde{\mathbf{A}}^*]'\widetilde{\mathbf{A}}^*) = \widetilde{\mathbf{S}}^*\widetilde{\mathbf{A}}^*. \tag{36}$$

Equation (36) is similar to (16). This implies that $\mathbf{A}^*$ by EM only guarantees to be a solution of (16) but might not be a global maxima of l given $\boldsymbol{\Psi}^*$. Similar to the proof in Tipping and Bishop (1999, Appendix A) it can be shown that $\mathbf{A}^*$ by EM could be a saddle point of l given $\boldsymbol{\Psi}^*$.

In Appendix 8.1, we analyze the computational complexity of EM, ECME2 and CM. In Appendix 8.2, we show that the log likelihood of ECME2 and CM can be efficiently evaluated if we use log likelihood as a convergence criterion.

5 Simulations

5.1 Simulation data

As verified in Petersen et al. (2005), EM algorithms tend to be inefficient in low-noise linear models (including FA). Motivated by this, artificial data with three kinds of noise types: low, high and ordinary (not high and not low) are generated to show how EM and the proposed ECME2 and CM performs in these different data settings. On the other hand, it is well known that the rate of convergence of EM is determined by the ratio of *missing information* to *complete information* (Dempster et al. 1977). For FA, the missing information is closely related to subspace dimension q. Therefore, we examine the performance of EM, ECME2 and CM for fitting FA with different q on the same data set, which consists of $N = 1000$ data points in $\mathbb{R}^{10}$.

All computations are carried out by MATLAB 7.1 in win32 environment of desktop PC with CPU: AMD 2.21 GHz and RAM: 2 GB. In all simulations, we use the following:

- *Initialization:* Perform PCA for parameter initialization $\boldsymbol{\theta}^{(0)} = (\mathbf{A}^{(0)}, \boldsymbol{\Psi}^{(0)})$ required by EM or $\boldsymbol{\Psi}^{(0)}$ by ECME2 and CM, which is often used as a start for fitting FA. Although only $\boldsymbol{\Psi}^{(0)}$ is required by ECME2 and CM, (18) in the first iteration will make $\mathbf{A}^{(1)} = \mathbf{A}^{(0)}$ (by setting $\mathbf{V}$ equal to the first q' rows of $\mathbf{I}$). Therefore, all algorithms actually start by PCA.
- *Convergence criterion:* Stop algorithms if $l^{(t+1)} - l^{(t)} <$ tol or $t > K_{\max}$ with tol $= 10^{-6}$ and the maximal number of iterations: $K_{\max} = 5000$.
- *Constraint:* $\eta = 10^{-6}$.

5.1.1 Simulation 1: ordinary noise

The data sets in this simulation are generated from the model (1) with:

$$\boldsymbol{\Psi} = \mathrm{diag}(1\ \ 2\ \ 3\ \ 4\ \ 5\ \ 6\ \ 7\ \ 8\ \ 9\ \ 10);$$

$$\boldsymbol{\mu} = (3\ \ 3\ \ 3\ \ 3\ \ 7\ \ 7\ \ 7\ \ 7\ \ 7\ \ 7);$$

$$\mathbf{A}' = \begin{pmatrix} 1.3 & 1 & 1.5 & 2.3 & 1.8 & 1.2 & 1.5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1.8 & 2.2 & 1 & 1.8 & 1.2 & 1.5 \\ 3.5 & 2 & 2.5 & 1.5 & 2 & 3 & 2.5 & 1.8 & 1.4 & 1.3 \\ 4 & 2.2 & 1.3 & 2.4 & 0 & 0 & 0 & 2 & 3.1 & 2.7 \end{pmatrix}.$$

Figure 1(a), (b), (c) shows the typical evolvement of log-likelihood l of FA model with $q = 1$, $q = 2$ and $q = 3$, respectively. It can be observed that CM stops earliest and ECME2 stops earlier than EM, particularly, when $q = 1$.

5.1.2 Simulation 2: high noise

This simulation is motivated by an exercise in Hastie et al. (2001, p. 507), which examines whether FA can look for maximal correlation among three variables x_1, x_2, x_3 in the case that x_1 and x_2 are correlated and independent with x_3 and have much lower variance than x_3. Here, for simplicity, we simply use the same value of parameters: $\boldsymbol{\Psi}$, $\boldsymbol{\mu}$ and $\mathbf{A}$ as in Simulation 1 in this simulation except that $\psi_7 = 100$ and $\psi_9 = 200$ to simulate high noise. Figure 1(d), (e) and (f) shows the typical evolvement of log-likelihood l of FA model with $q = 1$, $q = 2$ and $q = 3$, respectively. It can be observed that (1) the winner is still CM, which has overwhelming speediness over ECME2 and EM, and ECME2 beats EM only when $q = 1$; (2) For EM and ECME2 in Fig. 1(e) and (f), there is significant increase in the final log

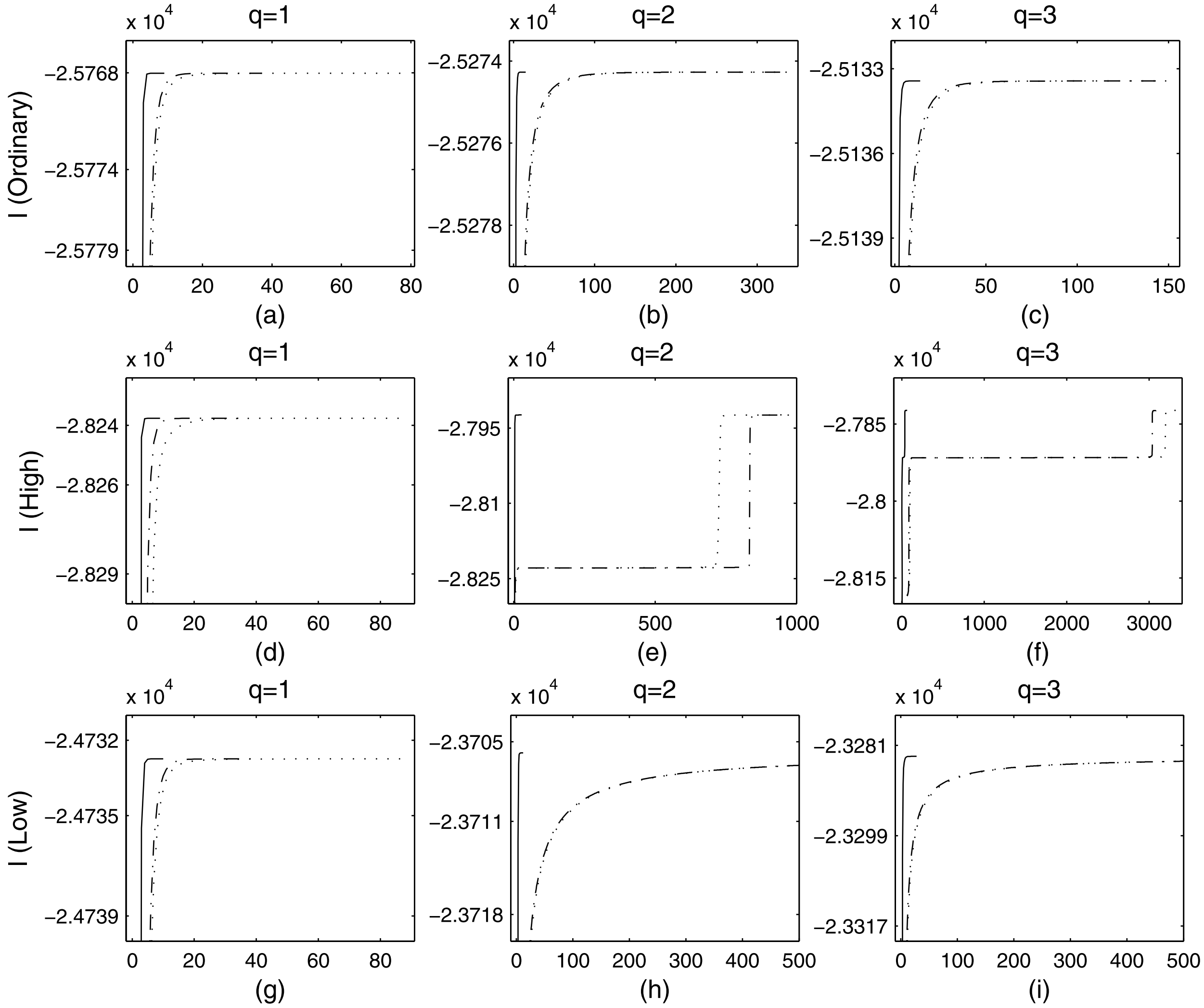

Fig. 1 A zoomed-in plot of the typical evolvement of log-likelihood l for EM (*dotted line*), ECME2 (*dashed line*) and CM (*solid line*) fitted to data sets with different noise types: Ordinary noise (*row 1*), high noise (*row 2*) and low noise (*row 3*) and for different q: $q = 1$ (*column 1*), $q = 2$ (*column 2*) and $q = 3$ (*column 3*). Only at most the former 500 iterations are shown for the case in low noise

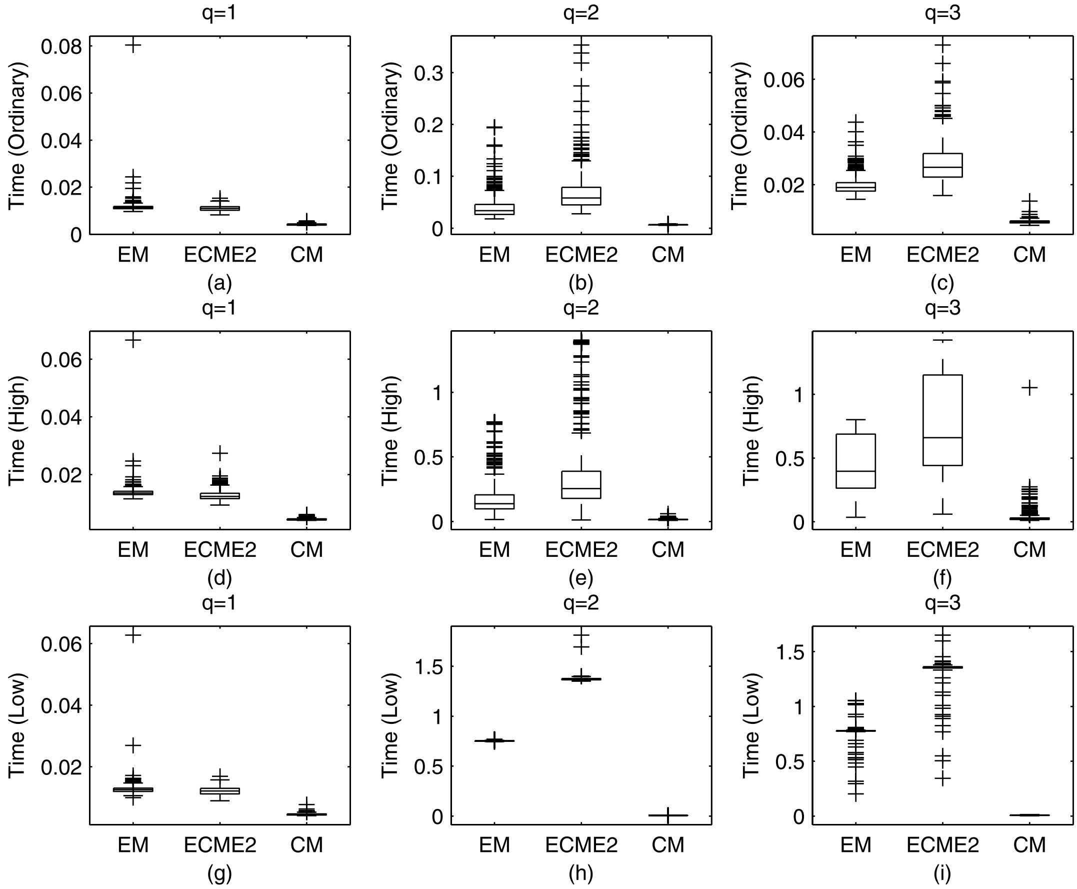

Fig. 2 The box plot of required time for convergence by EM, ECME2, and CM under different situations

likelihood after a large number of almost ‘converging’ iterations (a zoomed-in version of Fig. 1(e) is shown in Fig. 4(a)). Early stop, e.g. at 1000 or 2000 iterations in Fig. 1(f), may result in poorer estimations.

5.1.3 Simulation 3: Low noise

The same parameters: $\mathbf{\Psi}$, $\boldsymbol{\mu}$ and $\mathbf{A}$ as used in Simulation 1 are used in this simulation except that $\psi_7 = 10^{-4}$ and $\psi_9 = 10^{-4}$ to simulate low noise. Figure 1(g), (h) and (i) shows the typical evolvement of log-likelihood l of FA model with $q = 1$, $q = 2$ and $q = 3$, respectively. For clarity, only at most the former 500 iterations are shown since CM stops much earlier than EM and ECME2, usually using 10–30 iterations while both EM and ECME2 usually require the maximal number of iterations: $K_{\max} = 5000$. For EM and ECME2 in this case, it seems plausible to early stop since significant increase in the final log-likelihood does not appear. However, the problem is that we do not know in advance whether the final likelihood will change significantly or not. This case is closely related to the well known *Heywood Cases* (Bartholomew 1987, p. 69), which is often observed during the fitting of FA model for real data.

5.2 Performance analysis

We investigate the performance of EM and our proposed algorithms: ECME2 and CM, by repeating Simulation 1, Simulation 2 and Simulation 3 500 times, respectively. Let T and K denote the required CPU time and number of iterations when the *convergence criterion* is satisfied. To compare any two algorithms A and B, we define *speedup* of A over B by calculating the ratio: the median of 500 T’s or 500 K’s by B over that by A. Figure 2 and Fig. 3 show box plots of 500 T’s and 500 K’s by all algorithms in different cases. The performance of all algorithms is summarized as follows:

(1) Final likelihoods: In all 4500 simulations, when the *convergence criterion* is satisfied, the final log-likelihoods by CM are strictly larger than those by EM and ECME2, although there are usually no major difference.

(2) CM vs. EM and ECME2

   (a) Speedup: It is clear from Fig. 2 and Fig. 3 that CM outperforms EM and ECME2 no matter assessed by T or K in all situations. Importantly, from Fig. 2(h), (i) and Fig. 3(h), (i), CM is free of the inefficiency

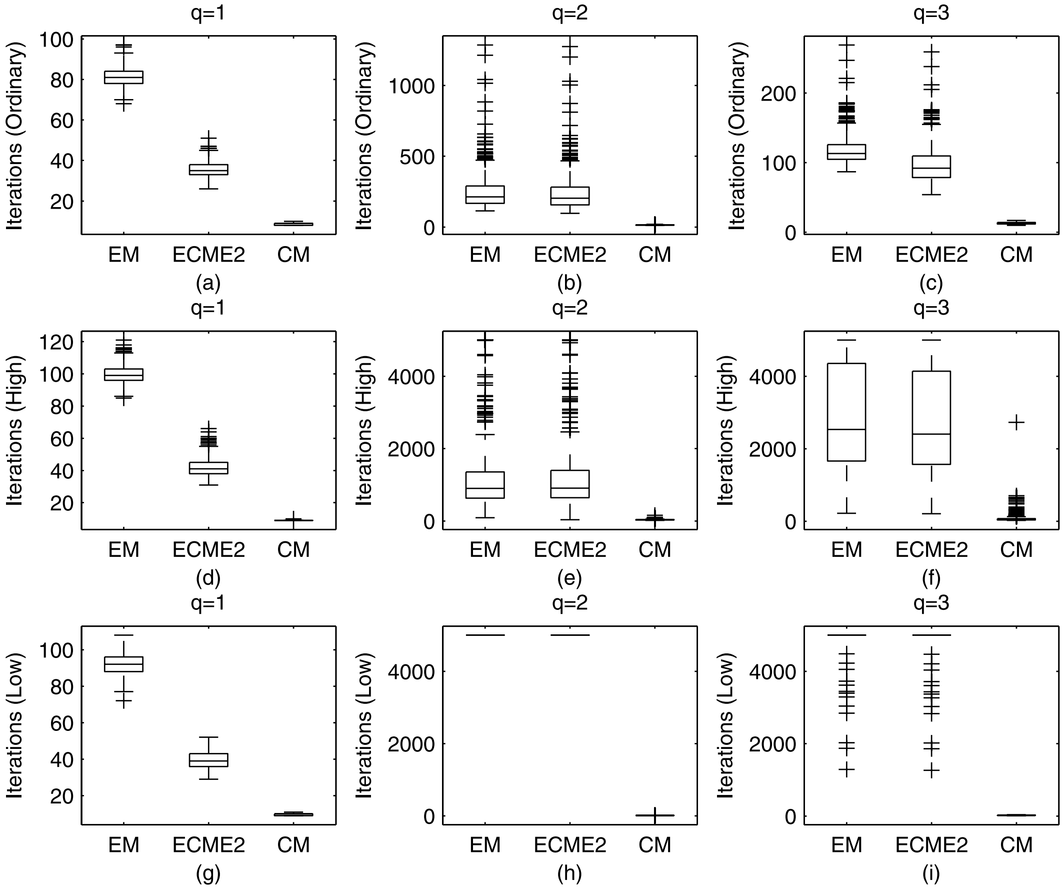

Fig. 3 The box plot of required number of iterations for convergence by EM, ECME2, and CM under different situations

Table 1 Speedup by CM over EM in CPU time (T) and number of iterations (K)

Noise\\q	1		2		3	
	T	K	T	K	T	K
Ordinary	2.7	9.0	5.1	15.2	3.2	8.7
High	3.1	11.0	9.5	25.7	18.7	47.8
Low	2.7	9.2	131.4	384.6	99.1	277.8

in low-noise limit suffered by EM and ECME2. Table 1 lists the speedup by CM over EM in T and K.

(b) Variation in number of iterations: It is easily observed from Fig. 3 that the box length of CM is far smaller than that of EM and ECME2 almost in all situations except the cases: Low noise, $q = 2$ and $q = 3$ represented in Fig. 3(h) and (i), in which EM and ECME2 are forced to stop early since they have arrived the maximal number of iterations $K_{\max} = 5000$. This observation implies that the variation of 500 K's by CM is far less than that by EM and ECME2 or in other words, it is more stable for CM than EM and ECME2 in terms of the number of iterations.

(3) ECME2 vs. EM: It can be observed from Fig. 2(a), (d) and (g) that ECME2 requires slightly less T for convergence than EM when $q = 1$ since it requires far fewer K (Fig. 3(a), (d) and (g)). However, such advantage disappears when $q = 2$ and $q = 3$ since it requires more computation than EM at each iteration (detailed in Appendix 8.1) and usually requires only slightly fewer K for convergence.

(4) 'Outliers' in CM: It can be observed from Fig. 2 that CM in Fig. 2(f) has relatively more 'outliers' (which requires much larger K than that is usually required) than CM in other figures. With respect to this, there are two cases: (1) CM can locate the global optimal solution in the sense that the estimate of uniqueness obtained by CM: $\hat{\boldsymbol{\Psi}}_{CM}$ is close to true one but the log-likelihood convergence curve is very 'flat' and thus CM tends to require more iterations; (2) It appears that CM is trapped into another local maxima in the sense that at least one element of $\hat{\boldsymbol{\Psi}}_{CM}$ is far from the true one, typically, $[\hat{\sigma}_9^2]_{CM} = \eta$, even if we use a stricter convergence criterion. We have randomly selected several

Table 2 Speedup by CM over QN in CPU time (T) and number of iterations (K)

Noise\\q	1		2		3	
	T	K	T	K	T	K
Ordinary	23.3	5.2	15.4	3.4	17.9	3.8
High	41.9	10.3	12.5	2.7	7.8	1.5
Low	23.4	5.3	22.2	3.8	16.5	2.4

Table 3 Speedup by CM over ECME1 in CPU time (T) and number of iterations (K)

Noise\\q	1		2		3	
	T	K	T	K	T	K
Ordinary	6.5	8.9	8.6	9.9	7.6	8.2
High	7.9	10.9	5.2	5.3	5.2	4.8
Low	6.9	9.1	8.1	8.5	18.2	17.8

replications of this case and found that it is possible that CM can obtain higher likelihood and require far fewer K if we choose another start point rather than PCA.

We also examine the performance of a quasi-Newton (QN) algorithm applied to fitting FA. Our implementation is based on Matlab function 'fmincon' with the same convergence criterion, constraint and PCA start as described above. We in general follow the SAS procedure provided in Chen (2003) to perform Jöreskog's method but there is a difference: Chen's procedure is based on Davidon, Fletcher and Powell (DFP) method while our use of 'fmincon' is based on Broyden, Fletcher, Goldfarb, and Shanno (BFGS) method. Therefore, although it has been shown by extensive numerical experiments in the literature that BFGS method is superior than DFP method, our implementation is not exactly the same as Jöreskog's method and we simply give a rough comparison.

Table 2 lists the obtained speedup by CM over QN. Almost 88 percent of all 4500 simulations, CM obtains strictly higher final log-likelihood than QN and all final log-likelihoods by CM and QN satisfy: $l_{CM} > l_{QN} - 10^{-4}$, which implies that l_{CM} is close to l_{QN} even if $l_{QN} > l_{CM}$. However, it is not the case for $l_{CM} > l_{QN}$ since there are still about 7.7 percent of all 4500 simulations which satisfy $l_{CM} - 100 > l_{QN}$. The obvious lower log-likelihood are mainly in the case: High noise, $q = 2$ and $q = 3$. It can be observed from Table 2 that CM outperforms QN in all situations, especially, in CPU time. Although there are some cases that the number of iterations is close to CM (e.g. High noise, $q = 3$), computation of QN in each iteration is much heavier than CM and thus QN requires much more CPU time. By comparing Table 1 with Table 2, in simple case like Ordinary noise, EM is superior than QN, while in hard case like Low noise with $q = 2$ and $q = 3$, QN is superior than EM.

Finally, we investigate how ECME1 performs in these different situations. Since the CM-step 2 in ECME1, using Newton-Raphson to update $\mathbf{\Psi}$, can not guarantee monotone convergence, a *mixture* of EM and ECME1 is considered here. Note that EM differs from ECME1 only in the step to update $\mathbf{\Psi}$. The *mixture* here means that (1) Perform 20 iterations of EM followed by ECME1; (2) If at iteration $t + 1$, the step to update $\mathbf{\Psi}$ in ECME1 does not increase likelihood, we use the corresponding step in EM to replace it. The reason of doing so is that (1) is expected to provide a good starting value for Newton-Raphson and (2) can guarantee the desired monotone convergence. In each iteration of ECME1, we simply do one iteration of Newton-Raphson. More iterations could be used but from our experience this usually increases the CPU time. On the other hand, we have found that the algorithm has numerical problem, especially, in the Low noise case if the *constraint* is not used. Therefore, we follow the treatment in Jöreskog (1967) to impose the *constraint* into the *mixture*: if at some iteration t, some elements of $\mathbf{\Psi}^{(t)}$ are less then η, we restrict those elements to be η and only the other elements of $\mathbf{\Psi}^{(t)}$ are updated in all remaining iterations. On the other hand, we stop the algorithm if the *convergence criterion* is satisfied or the maximal absolute element in the gradient vector of $l(\mathbf{\Psi})$ is less than 5×10^{-5}.

Table 3 lists the obtained speedup by CM over ECME1. All final likelihoods by CM are strictly higher than those by ECME1. From Table 3, CM outperforms ECME1 in all situations. By comparing Table 1 with Table 3, in some cases (e.g. High and Low noise with $q = 2, 3$), the *mixture* can accelerate the EM alone but it is not always the case.

5.3 On different starting values

In previous experiments, all algorithms start by PCA. We are interesting in studying the sensitivity of EM, ECME2 and CM towards the choice of staring values. To investigate this, we use two types of starting values: (1) ad hoc $\mathbf{\Psi}^{(0)}$ and (2) ad hoc $\mathbf{\Psi}^{(0)}$ and $\mathbf{A}^{(0)}$, since CM and ECME2 only require $\mathbf{\Psi}^{(0)}$ and EM requires $\mathbf{\Psi}^{(0)}$ and $\mathbf{A}^{(0)}$.

5.3.1 Type 1: ad hoc $\mathbf{\Psi}^{(0)}$

We take $\mathbf{\Psi}^{(0)} = 0.1\,\mathbf{I}$ (for EM, $\mathbf{A}^{(0)}$ is still decided by PCA). The data in Fig. 1(e) (High noise) is used. Figure 4(b) shows a close-up of the obtained convergence curve of l. For comparison, here we also provide a copy of Fig. 1(e), i.e., Fig. 4(a). The required number of iterations K for convergence by EM, ECME2 and CM are 975, 1076 and 28 in Fig. 4(a), and 11444, 12071 and 28 in Fig. 4(b).

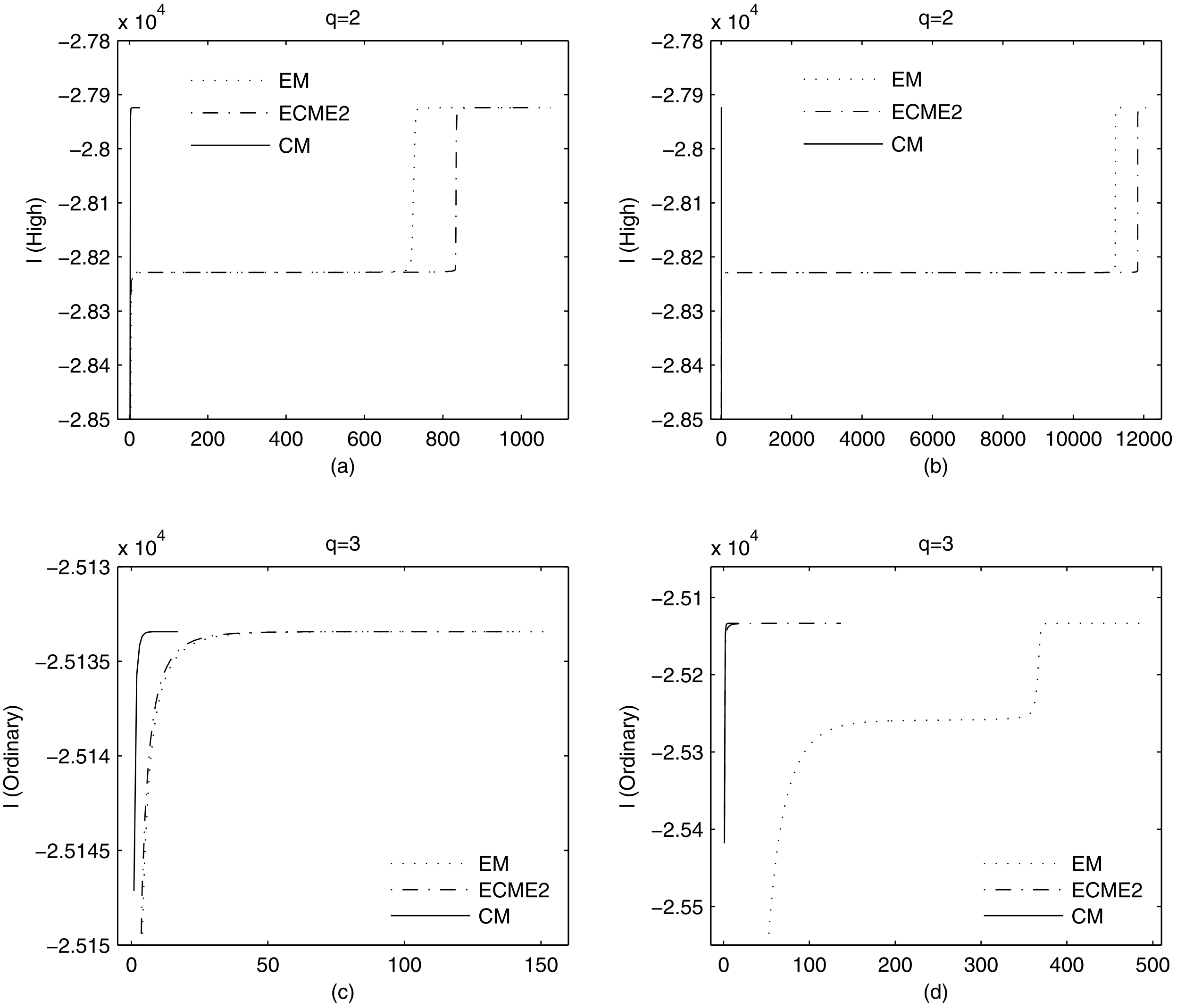

Fig. 4 A zoomed-in plot of the evolution of log-likelihood l. *Row 1*: High noise, $q = 2$; *row 2*: ordinary noise, $q = 3$; *column 1*: using PCA start; *column 2*: using an ad hoc starting value

5.3.2 *Type 2: ad hoc* $\mathbf{\Psi}^{(0)}$ *and* $\mathbf{A}^{(0)}$

We use $\mathbf{\Psi}^{(0)} = 0.1\,\mathbf{I}$. For EM, $\mathbf{A}^{(0)} = \mathbf{I}_q$, where $\mathbf{I}_q$ is the first q columns of $\mathbf{I}$. The data in Fig. 1(c) is used. Figure 4(c) and (d) show the convergence of l using PCA start and the ad hoc starting value, respectively. The required K by EM, ECME2 and CM are 154, 143 and 14 in Fig. 4(c) and 495, 139 and 17 in Fig. 4(d).

Although the result reported here is simply from single run, we have verified empirically the same story exists for the other replications in corresponding noise case. To sum up, (1) Usually, EM is sensitive to initialization in the sense that different initializations may require different K, ECME2 in high noise case does so as well, and PCA is a good candidate for initialization; (2) CM is much more insensitive to initialization than EM and ECME2.

6 Conclusion and future work

We propose a novel CM algorithm for fitting FA, which, like EM, is easy to implement and stable to converge. CM is an old member of EM, (or more general AECM (Meng and van Dyk 1997)) algorithms family but it is a new member in the fitting of ML FA area. Our empirical results in Sect. 5 show that CM plays a satisfactory role in this area since it outperforms EM (Rubin and Thayer 1982), ECME1 (Liu and Rubin 1998), ECME2 and QN (Jöreskog 1967) no matter assessed by CPU time or number of iterations. In addition, we show how our proposed ECME2 can be related to the simple iteration algorithm (Lawley 1940) and ECME2 frees from several problems suffered by the simple iteration algorithm.

An extension of CM algorithm for fitting the mixture of factor analyzers is currently being investigated. Recently, Yu et al. (2005) propose applying Monte Carlo EM algorithm to fit FA for ranking data. However, their method is basically based on the EM described in this paper. We are hence working on extension of CM to see whether it can achieve acceleration.

Matlab implementation of CM and ECME2 algorithm is available from the first author via email.

Acknowledgements The authors would like to thank the editors and one anonymous referee for their valuable comments and suggestions. The first author's work is partly supported by YNU Science fund no. 2004Q029C. The second author's work is supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. HKU 7176/02H).

Appendix 1: Proofs

7.1 The equivalence between (5) and (7)

The definition of $\boldsymbol{\Sigma}$ gives us the following identities

$$\boldsymbol{\Sigma}^{-1}\boldsymbol{\Psi} = \mathbf{I} - \boldsymbol{\Sigma}^{-1}\mathbf{A}\mathbf{A}', \qquad \boldsymbol{\Psi}\boldsymbol{\Sigma}^{-1} = \mathbf{I} - \mathbf{A}\mathbf{A}'\boldsymbol{\Sigma}^{-1}. \tag{37}$$

By (5), we obtain

$$\operatorname{diag}(\boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1}\mathbf{S}\boldsymbol{\Sigma}^{-1}) = 0.$$

Since $\boldsymbol{\Psi}$ is diagonal, this equation gives us

$$\begin{aligned}
0 &= \operatorname{diag}(\boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1}\mathbf{S}\boldsymbol{\Sigma}^{-1}) \cdot \boldsymbol{\Psi} \\
&= \operatorname{diag}(\boldsymbol{\Sigma}^{-1}\boldsymbol{\Psi} - \boldsymbol{\Sigma}^{-1}\mathbf{S}\boldsymbol{\Sigma}^{-1}\boldsymbol{\Psi}) \\
&= \operatorname{diag}(\mathbf{I} - \boldsymbol{\Sigma}^{-1}\mathbf{A}\mathbf{A}' - \boldsymbol{\Sigma}^{-1}\mathbf{S} + \boldsymbol{\Sigma}^{-1}\mathbf{S}\boldsymbol{\Sigma}^{-1}\mathbf{A}\mathbf{A}') \\
&= \operatorname{diag}(\mathbf{I} - \boldsymbol{\Sigma}^{-1}\mathbf{S}),
\end{aligned}$$

where the last two equalities follow from the first identity in (37) and (6), respectively. Now (7) is obtained by the following simple calculations:

$$\boldsymbol{\Psi} = \operatorname{diag}(\boldsymbol{\Psi}\boldsymbol{\Sigma}^{-1}\mathbf{S}) = \operatorname{diag}(\mathbf{S} - \mathbf{A}\mathbf{A}'\boldsymbol{\Sigma}^{-1}\mathbf{S}) = \operatorname{diag}(\mathbf{S} - \mathbf{A}\mathbf{A}'),$$

where, to obtain the last two equalities, the second identity in (37) and (6) are used respectively. Since the above derivation can be reversed, we obtain the desired equivalence.

7.2 Proof of Proposition 1

For any d-dimensional column vectors $\mathbf{u}, \mathbf{v}$ such that $1 + \mathbf{u}'\mathbf{C}^{-1}\mathbf{v} \neq 0$, we have

$$(\mathbf{u}\mathbf{v}' + \mathbf{C})[\mathbf{C}^{-1} - \mathbf{C}^{-1}\mathbf{u}\mathbf{v}'\mathbf{C}^{-1}/(1 + \mathbf{u}'\mathbf{C}^{-1}\mathbf{v})] = \mathbf{I}.$$

This shows that

$$(\mathbf{u}\mathbf{v}' + \mathbf{C})^{-1} = \mathbf{C}^{-1} - \frac{\mathbf{C}^{-1}\mathbf{u}\mathbf{v}'\mathbf{C}^{-1}}{1 + \mathbf{u}'\mathbf{C}^{-1}\mathbf{v}}. \tag{38}$$

Now (28) is obtained from (38) by setting $\mathbf{u} = \omega\mathbf{e}_i$, $\mathbf{v} = \mathbf{e}_i$. Actually, both (38) and (28) can be viewed as special cases of a generalized Woodbury's formula (Lange 1999, p. 90).

7.3 Proof of Proposition 2

For $\omega_i \geq 0$, the proposition is obviously true. For $-1 < \omega_i < 0$, one can easily verify that

$$\begin{aligned}
&-\omega_i^{-1} \cdot \mathbf{B}_i - \mathbf{e}_i\mathbf{e}_i' \\
&\quad = \operatorname{diag}\bigg(\frac{1 + \omega_1^{(t+1)}}{-\omega_i}, \ldots, \frac{1 + \omega_{i-1}^{(t+1)}}{-\omega_i}, \frac{1 + \omega_i}{-\omega_i}, \\
&\qquad -\frac{1}{\omega_i}, \ldots, -\frac{1}{\omega_i}\bigg) + \widetilde{\mathbf{A}}\widetilde{\mathbf{A}}' > 0.
\end{aligned}$$

By the following lemma, we have $\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i < -\frac{1}{\omega_i}$. Hence $1 + \omega_i\mathbf{e}_i'\mathbf{B}_i^{-1}\mathbf{e}_i > 0$.

Lemma 1 *Suppose* $\mathbf{C} > 0$, $\mathbf{v}$ *is a* d*-dimensional non-zero column vector,* α *is a real number. Then*

$$\alpha\mathbf{C} - \mathbf{v}\mathbf{v}' > 0 \Longleftrightarrow \mathbf{v}'\mathbf{C}^{-1}\mathbf{v} < \alpha. \tag{39}$$

Proof If $\alpha\mathbf{C} - \mathbf{v}\mathbf{v}'$ is positive definite, then

$$\mathbf{v}'\mathbf{C}^{-1} \cdot (\alpha\mathbf{C} - \mathbf{v}\mathbf{v}') \cdot \mathbf{C}^{-1}\mathbf{v} = \mathbf{v}'\mathbf{C}^{-1}\mathbf{v}(\alpha - \mathbf{v}'\mathbf{C}^{-1}\mathbf{v})$$

is a positive number. This is the case if and only if $\mathbf{v}'\mathbf{C}^{-1}\mathbf{v} < \alpha$, because $\mathbf{v}'\mathbf{C}^{-1}\mathbf{v} > 0$ by assumption.

Let $\mathbf{w}$ be an arbitrary d-dimensional non-zero column vector. Since

$$\|\mathbf{w}'\mathbf{C}^{1/2} + x\mathbf{v}'\mathbf{C}^{-1/2}\|^2 = \mathbf{w}'\mathbf{C}\mathbf{w} + 2x\mathbf{w}'\mathbf{v} + x^2\mathbf{v}'\mathbf{C}^{-1}\mathbf{v}$$

is non-negative for any real number x, we must have

$$(\mathbf{w}'\mathbf{v})^2 \leq (\mathbf{w}'\mathbf{C}\mathbf{w})(\mathbf{v}'\mathbf{C}^{-1}\mathbf{v}).$$

Now suppose $\mathbf{v}'\mathbf{C}^{-1}\mathbf{v} < \alpha$, then, by the above inequality, we have

$$\begin{aligned}
&\mathbf{w}'(\alpha\mathbf{C} - \mathbf{v}\mathbf{v}')\mathbf{w} \\
&\quad = \alpha\mathbf{w}'\mathbf{C}\mathbf{w} - (\mathbf{w}'\mathbf{v})^2 \geq \mathbf{w}'\mathbf{C}\mathbf{w}(\alpha - \mathbf{v}'\mathbf{C}^{-1}\mathbf{v}) > 0.
\end{aligned}$$

Since $\mathbf{w}$ is arbitrary, we conclude that $\alpha\mathbf{C} - \mathbf{v}\mathbf{v}' > 0$. □

Appendix 2: Some notes

8.1 Computational complexity analysis

It is helpful to inspect the computation cost in each iteration of different algorithms. Since $\mathbf{S}$, computation of which is $O(Nd^2)$, does not change during iterations of all algorithms, we assume that $\mathbf{S}$ has been calculated before running all algorithms. It should be pointed out that computation of $\mathbf{S}$ can be avoided by EM as detailed below. The computation cost is summarized in Table 4.

Looking at Table 4, (1) the computation burden of EM in each iteration is lightest, especially, for the data in 'large d, small q' setting, which implies that for such data EM may be fairly competitive. (2) Compared with EM, the burden of CM is heavier, particularly, in the step to update $\boldsymbol{\Psi}$. Nevertheless, methods such as Jöreskog (1967)'s method and ECME1 which perform CML over $\boldsymbol{\Psi}$ also have similar complexity.

- EM: Using (8), (14) and (15) can be rewritten as

$$\mathbf{A}^{(t+1)} = \mathbf{G}^{(t)}\big(\mathbf{I} + [\mathbf{H}^{(t)}]'\mathbf{F}^{(t)}\big)^{-1}, \tag{40}$$

$$\boldsymbol{\Psi}^{(t+1)} = \mathrm{diag}\big(\mathbf{S} - \mathbf{H}^{(t)}[\mathbf{A}^{(t+1)}]'\big), \tag{41}$$

where $\mathbf{F}^{(t)} = \boldsymbol{\Psi}^{(t)^{-1}}\mathbf{A}^{(t)}, \mathbf{G}^{(t)} = \mathbf{S}\mathbf{F}^{(t)}, \mathbf{H}^{(t)} = \mathbf{G}^{(t)}(\mathbf{I} + [\mathbf{A}^{(t)}]'\mathbf{F}^{(t)})^{-1}$.

To make the computation more efficient, (40) and (41) can be implemented as follows:

$$\boldsymbol{\Psi}^{(t)^{-1}} \to \mathbf{F}^{(t)} \to \mathbf{G}^{(t)} \to \mathbf{H}^{(t)} \to \mathbf{A}^{(t+1)} \to \boldsymbol{\Psi}^{(t+1)}.$$

By doing so, the cost of (40) is now dominated by $O(d^2q)$. As $\mathbf{H}^{(t)}$ has been calculated in (40), the main cost of (41), $\mathrm{diag}(\mathbf{H}^{(t)}[\mathbf{A}^{(t+1)}]')$, is simply $O(dq)$ using the fact that only the diagonal elements are required.

To avoid direct calculation of $\mathbf{S}$, a trick mentioned in Tipping and Bishop (1999) can be used. Since $\mathbf{S}$ is only involved in computation of $\mathbf{G}^{(t)}$ and $\mathrm{diag}(\mathbf{S})$ in (41), we can calculate $\sum_{n=1}^{N}(\mathbf{x}_n - \bar{\mathbf{x}})[(\mathbf{x}_n - \bar{\mathbf{x}})'\mathbf{F}^{(t)}]$ instead of $\sum_{n=1}^{N}[(\mathbf{x}_n - \bar{\mathbf{x}})(\mathbf{x}_n - \bar{\mathbf{x}})']\mathbf{F}^{(t)}$, and the cost of $\mathbf{G}^{(t)}$ can be reduced to $O(Ndq)$; Calculation of $\mathrm{diag}(\mathbf{S})$ only requires elementwise multiplication of $(\mathbf{x}_n - \bar{\mathbf{x}})$ and itself, which is $O(Nd)$. In summary, the saving in computing $\mathbf{S}$, which is $O(Nd^2)$, requires trading computation $O(Nd(q+1)K)$, where K is the number of iterations for convergence.
- ECME2: It is required in (21) to compute $\widetilde{\mathbf{S}}$ and its singular value decomposition, which is dominated by $O(d^3)$ (if Matlab function 'eigs' is used, the cost can be significantly reduced when d is large). Since $\widetilde{\mathbf{S}}$ is given in former step, cost of (22) is mainly caused by $\mathrm{diag}\{\widetilde{\mathbf{A}}^{(t+1)}[\widetilde{\mathbf{A}}^{(t+1)}]'\}$, which is simply $O(dq)$.

Table 4 Computation cost of different algorithms in each iteration

Algorithm	Update of $\mathbf{A}$	Update of $\boldsymbol{\Psi}$
EM	$O(d^2q)$	$O(dq)$
ECME2	$O(d^3)$	$O(dq)$
CM	$O(d^3)$	$O(d^3)$

- CM: The first step is the same as that in ECME2. The cost of (34) is $O(d^2q)$. In each step of the following d steps, the main computation is in (33) and (30), which is $O(d^2)$ and hence the total cost of d steps is $O(d^3)$.

8.2 Convergence criterion in ECME2 and CM

If we use log-likelihood l as a convergence criterion in ECME2 and CM, then at iteration $t+1$, we need to evaluate $l(\mathbf{A}^{(t)}, \boldsymbol{\Psi}^{(t)})$ (see (3)), which requires the computation of the determinant and inverse of a $d \times d$ matrix $\boldsymbol{\Sigma}$ though the computation can be reduced by using the relationship (e.g. see Lawley and Maxwell 1971, Chap. 4):

$$\boldsymbol{\Sigma}^{-1} = \boldsymbol{\Psi}^{-1} - \boldsymbol{\Psi}^{-1}\mathbf{A}\big(\mathbf{I} + \mathbf{A}'\boldsymbol{\Psi}^{-1}\mathbf{A}\big)^{-1}\mathbf{A}'\boldsymbol{\Psi}^{-1},$$

$$|\boldsymbol{\Sigma}| = |\boldsymbol{\Psi}||\mathbf{I} + \mathbf{A}'\boldsymbol{\Psi}^{-1}\mathbf{A}|.$$

In order to further reduce computation, for both algorithms, we tend to evaluate $l(\mathbf{A}^{(t+1)}, \boldsymbol{\Psi}^{(t)})$ after the CM-step 1 (instead of $l(\mathbf{A}^{(t)}, \boldsymbol{\Psi}^{(t)})$ before the CM-step 1). Then,

$$l(\mathbf{A}^{(t+1)}, \boldsymbol{\Psi}^{(t)}) = -\frac{N}{2}\Bigg\{d\ln 2\pi + \ln|\boldsymbol{\Psi}^{(t)}| + \mathrm{tr}(\widetilde{\mathbf{S}}) + \sum_{i=1}^{q'}[\ln(\widetilde{\lambda}_i) - \widetilde{\lambda}_i] + q'\Bigg\}. \tag{42}$$

Clearly, the computation of (42) is very cheap since it can make use of the existing eigen-decomposition of covariance matrix $\widetilde{\mathbf{S}}$ and the simple computation of $\ln|\boldsymbol{\Psi}^{(t)}|$.

References

Bartholomew, D.J.: Latent Variable Models and Factor Analysis, 1st edn. Oxford University Press, New York (1987)

Chen, R.: An SAS/IML procedure for maximum likelihood factor analysis. Behav. Res. Methods Instrum. Comput. **35**(3), 310–317 (2003)

Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data using the EM algorithm (with discussion). J. R. Stat. Soc. B **39**(1), 1–38 (1977)

Fletcher, R., Powell, M.J.D.: A rapidly convergent descent method for minimization. Comput. J. **2**, 163–168 (1963)

Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. Springer, New York (2001)

Jennrich, R.I., Bobinson, S.M.: A Newton-Raphson algorithm for maximum likelihood factor analysis. Psychometrika **34**(1), 111–123 (1969)

Jöreskog, K.G.: Some contributions to maximum likelihood factor analysis. Psychometrika **32**(4), 433–482 (1967)

Lange, K.: Numerical Analysis for Statisticians. Springer, New York (1999)

Lawley, D.N.: The estimation of factor loadings by the method of maximum likelihood. Proc. R. Soc. Edinb. A **60**, 64–82 (1940)

Lawley, D.N., Maxwell, A.E.: Factor Analysis as a Statistical Method, 2nd edn. Butterworths, London (1971)

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

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

Meng, X.-L., Rubin, D.B.: Maximum likelihood estimation via the ECM algorithm: a general framework. Biometrika **80**(2), 267–278 (1993)

Meng, X.L., van Dyk, D.A.: The EM algorithm—an old folk-song sung to a fast new tune. J. R. Stat. Soc. B **59**(3), 511–567 (1997)

Petersen, K.B., Winther, O., Hansen, L.K.: On the slow convergence of EM and VBEM in low-noise linear models. Neural Comput. **17**(9), 1921–1926 (2005)

Rubin, D.B., Thayer, T.T.: EM algorithms for ML factor analysis. Psychometrika **47**, 69–76 (1982)

Tipping, M.E., Bishop, C.M.: Probabilistic principal component analysis. J. R. Stat. Soc. Ser. B **61**, 611–622 (1999)

Wu, C.F.J.: On the convergence properties of the EM algorithm. Ann. Stat. **11**(1), 95–103 (1983)

Yu, P.L.H., Lam, K.F., Lo, S.M.: Factor analysis for ranked data with application to a job selection attitude survey. J. R. Stat. Soc. Ser. A **168**(3), 583–597 (2005)