ML Estimation of Mean and Covariance Structures with Missing Data Using Complete Data Routines

[1]

Mortaza Jamshidian
University of Central Florida

Peter M. Bentler
University of California, Los Angeles

August 18, 1997
In press:
Journal of Educational and Behavioral Statistics

[1]This work has been partly supported by National Institute on Drug Abuse Grant DA01070. This paper was presented at the 2nd International Conference on Social Science Information Technology, Amsterdam, The Netherlands, December, 1994. The authors would like to thank the associate editor and the referees for their helpful comments.

Abstract

We consider maximum likelihood (ML) estimation of mean and covariance structure models when data are missing. Expectation maximization (EM), generalized expectation maximization (GEM), Fletcher-Powell, and Fisher-scoring algorithms are described for parameter estimation. It is shown how the machinery within a software that handles the complete data problem can be utilized to implement each algorithm. A numerical differentiation method for obtaining the observed information matrix and the standard errors is given. This method too uses the complete data program machinery. The likelihood ratio test is discussed for testing hypotheses. Three examples are used to compare the cost of the four algorithms mentioned above, as well as to illustrate the standard error estimation and the test of hypothesis considered. The sensitivity of the ML estimates as well as the mean imputed and listwise deletion estimates to missing data mechanisms is investigated using three artificial data sets that are missing completely at random (MCAR), missing at random (MAR), and neither MCAR nor MAR.

Key Words: Factor analysis, Incomplete data, Listwise Deletion, Mean imputation, Missing data mechanism, Observed information, Test of hypothesis.

1 Introduction

In mean and covariance structure analysis, an important application of multivariate statistics, a simple random sample from a multivariate normal population with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$ is drawn, and a hypothesized parameterization (structure) of the mean $\boldsymbol{\mu} = \boldsymbol{\mu}(\boldsymbol{\theta})$ and the covariance matrix $\boldsymbol{\Sigma} = \boldsymbol{\Sigma}(\boldsymbol{\theta})$ is evaluated. Based on unstructured maximum likelihood (ML) estimators of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, asymptotically efficient estimators of the parameter vector $\boldsymbol{\theta}$, the covariance matrix of the estimator, and goodness-of-fit χ^2 tests of the null hypothesis have been developed. A summary of this statistical theory can be found, for example, in Satorra (1992) and Browne and Arminger (1995). Effective computational procedures for implementing this theory exist in various standard computer programs such as EQS (Bentler, 1995), LISREL (Jöreskog & Sörbom, 1988), MECOSA (Schepers & Arminger, 1992), and SEPATH (Steiger, 1994), and have recently been discussed by Arminger (1994), Browne and Du Toit (1992), and Cudeck, Klebe, and Henly (1993).

This statistical theory, and its computational implementation, is based on the assumption that there is no missing data. Unfortunately, this is an empirically unlikely, if not actually untenable, assumption. In this paper we review the foundations of the theory in the presence of missing data. For the missing data mechanism we assume *ignorable nonresponse*, as defined by Rubin (1987, Chapter 2). This assumption is satisfied if data are missing completely at random (MCAR) or are missing at random (MAR) (see, Little & Rubin 1987, Chapter 5). Briefly, data are said to be MCAR if their missingness is independent of the missing values themselves or the observed values of the other variables. Data are said to be MAR if the missing data do not depend on the missing values themselves, but may depend on the observed values of other variables. We refer to missing data mechanisms

that are neither MCAR nor MAR as *not missing at random* (NMAR).

As we shall see, various approaches to handling missing data in this context already have been developed. These approaches require specialized and often complex computer routines for their implementation, which may account for the absence of theoretically adequate methods for handling missing data in extant structural modeling programs. In this paper we consider four algorithms that can be implemented in standard programs such as the ones mentioned above with little difficulty. The key, as we show, is the method by which the modules in a program that handles complete data problems, henceforth referred to as a *complete data program*, can be utilized to fit models to incomplete data.

Heuristic methods to dealing with missing data certainly exist. The most common of these use an estimate of the unstructured mean and covariance as empirical data in a complete data program to obtain an estimate of $\boldsymbol{\theta}$. Three common examples of such methods are the mean imputation (MI) method, the listwise deletion (LD) method, and the maximum likelihood imputation (MLI) method. In each case the unstructured mean and covariance is obtained as follows: The MI method replaces missing values of each variable by the mean of the observed values from that variable and uses the mean and the covariance of the completed data set as empirical data. The LD method discards all the incomplete cases and uses the mean and covariance based on completely observed cases. The MLI method uses maximum likelihood estimates of the unstructured mean and covariance that are obtained by iterative imputation of the missing data (see e.g., Little & Rubin, 1987, Chapter 8). Finkbeiner (1979) and Brown (1983) surveyed these and other methods in the context of exploratory factor analysis. As they pointed out, some of these methods result in bias and significant loss of efficiency when the amount of missing data is substantial. Of the three methods mentioned above, the MLI method was favored by Brown. Arminger and Sobel (1990)

pointed out two main shortcomings of the MLI method: First, it is difficult to obtain standard errors of estimates from this method because it is hard to account for the variability of the unstructured mean and covariance estimates used to obtain estimates of $\boldsymbol{\theta}$. Second, this procedure's estimates are not as efficient as ML estimates that we will discuss shortly.

As opposed to the heuristic methods mentioned above, a model based approach may be considered. More specifically, suppose $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n$ are iid variables, completely or partially observed, from the p-variate normal distribution $\mathcal{N}_p(\boldsymbol{\mu}(\boldsymbol{\theta}), \boldsymbol{\Sigma}(\boldsymbol{\theta}))$. Let $\boldsymbol{y}_i$ denote the observed (non-missing) part of $\boldsymbol{x}_i$. Then assuming ignorable nonresponse, $\boldsymbol{y}_i$ has a marginal normal distribution $\mathcal{N}_{p_i}(\boldsymbol{\mu}_i(\boldsymbol{\theta}), \boldsymbol{\Sigma}_i(\boldsymbol{\theta}))$, where p_i is the number of elements of $\boldsymbol{y}_i$, and $\boldsymbol{\mu}_i(\boldsymbol{\theta})$ and $\boldsymbol{\Sigma}_i(\boldsymbol{\theta})$ are appropriate subvector and submatrix of $\boldsymbol{\mu}(\boldsymbol{\theta})$ and $\boldsymbol{\Sigma}(\boldsymbol{\theta})$. Then for a given value of $\boldsymbol{Y} = (\boldsymbol{y}_1, \cdots, \boldsymbol{y}_n)$ an estimate of $\boldsymbol{\theta}$ is obtained by maximizing the *the observed data log-likelihood*

$$\mathcal{L}_y(\boldsymbol{\theta}|\boldsymbol{Y}) = -\frac{N}{2}\log(2\pi) - \frac{1}{2}\sum_{i=1}^{n}\left\{\log|\boldsymbol{\Sigma}_i(\boldsymbol{\theta})| + \text{trace}\left[\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{\theta})C_i(\boldsymbol{\theta})\right]\right\}, \quad (1)$$

where, $C_i(\boldsymbol{\theta}) = [\boldsymbol{y}_i - \boldsymbol{\mu}_i(\boldsymbol{\theta})]\,[\boldsymbol{y}_i - \boldsymbol{\mu}_i(\boldsymbol{\theta})]^T$ and $N = \sum_{i=1}^{n} p_i$. We denote the value that maximizes (1) by $\widehat{\boldsymbol{\theta}}$ and hereafter we refer to it as ML.

Finkbeiner (1979) proposed using $\widehat{\boldsymbol{\theta}}$ in the context of exploratory factor analysis when data are incomplete. Using a Monte Carlo study, he compared $\widehat{\boldsymbol{\theta}}$ to several heuristic estimates of $\boldsymbol{\theta}$, a few of which were mentioned above, and concluded that $\widehat{\boldsymbol{\theta}}$ was superior. Muthén, Kaplan, and Hollis (1987) also studied the ML estimates of $\boldsymbol{\theta}$ and concluded that these estimates were "superior (to a number of methods that they tried) even in situations that (ML) did not fulfill the prerequisite for it to be maximum likelihood." Muthén et.al. (1987) also discussed an extension of the maximum likelihood method that modeled a missing data mechanism. As mentioned above, here we consider only missing data mechanisms that satisfy the ignorable nonresponse assumption.

Finkbeiner (1979) proposed a Fletcher-Powell (FP) algorithm to obtain $\widehat{\boldsymbol{\theta}}$ for the factor analysis model. In his algorithm, the Fisher information matrix is computed at the initial point and it is updated by the Fletcher-Powell formulas (see e.g., Luenberger, 1984). Finkbeiner gave the necessary formulas for implementing his algorithm for the exploratory factor analysis model. Lee (1986) considered the covariance structure $\boldsymbol{\Sigma}(\boldsymbol{\theta})$ with $\boldsymbol{\mu}(\boldsymbol{\theta}) = \mathbf{0}$. To estimate the parameters $\boldsymbol{\theta}$, he proposed the generalized least squares and ML methods. For the ML method, he suggested using the Fisher-Scoring (FS) algorithm which, as he pointed out, is an iteratively reweighted Gauss-Newton algorithm. He developed the relevant formulas for the confirmatory factor analysis model. When data are missing, his assumption of zero means causes his estimates not to be fully efficient unless the population mean is known to be zero. His algorithm, however, can be extended to the case of nonzero means.

To date, the methods, just discussed have not been implemented on any of the standard software packages such as EQS or LISREL [1]. This may be because these packages generally handle models with mean and covariance structures that are more complex than that of the factor analysis model. Extending the formulas to accommodate these more complex models is cumbersome if one is to use the direct approach of implementing algorithms used by Finkbeiner (1979) and Lee (1986). For example, computing the score function, required in both the FP and the FS algorithms, by direct differentiation of the observed log-likelihood can be complicated for the general model. The formulas will depend on the structures and they require special code. In this paper we show how these algorithms can be implemented using existing modules in a complete data program.

A class of methods utilizes the complete data programs to obtain $\widehat{\boldsymbol{\theta}}$. We

[1]Since the submission of this paper, Finkbeiner's method has been implemented in AMOS

refer to these as *complete data based methods*. A complete data program maximizes the *the complete data log-likelihood*

$$\mathcal{L}_x(\boldsymbol{\theta}|\bar{\boldsymbol{x}}, \boldsymbol{S}) = -(n/2)\Big\{p\log(2\pi) + \log|\boldsymbol{\Sigma}(\boldsymbol{\theta})| + \text{trace}\left[\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta})\left(\boldsymbol{S} - \boldsymbol{\mu}(\boldsymbol{\theta})\bar{\boldsymbol{x}}^T - \bar{\boldsymbol{x}}\boldsymbol{\mu}(\boldsymbol{\theta})^T + \boldsymbol{\mu}(\boldsymbol{\theta})\boldsymbol{\mu}(\boldsymbol{\theta})^T\right)\right]\Big\}, \quad (2)$$

for given values of $\boldsymbol{S} = (1/n)\ \sum_{i=1}^{n} \boldsymbol{x}_i\boldsymbol{x}_i^T$ and $\bar{\boldsymbol{x}} = (1/n)\sum_{i=1}^{n}\boldsymbol{x}_i$, assuming that $\boldsymbol{x}_i$'s have no missing values. Allison (1987), Muthén et. al. (1987), and Arminger and Sobel (1990) described methods that use the multiple group option of existing complete data programs (e.g., EQS and LISREL). The idea is to treat every set of observations with the same missing data pattern as a group and then impose equality restrictions on the parameters across groups. However, as these authors have noted, their approach requires the matrix of second order sample moments for each group (in this context for each pattern of missing data) to be positive definite, so the number of observed cases for each pattern has to be at least as large as the number of variables observed for that pattern. This assumption is practically restrictive, and requires throwing out data for infrequent patterns. Bentler (1990) suggested the improvement of collecting all data that would be discarded into an additional group for which heuristic methods could be used to produce a sample mean and covariance matrix. Although Bentler's approach avoids discarding data, it is not fully efficient. Jamshidian (1997) gave an extension of the Expectation-Maximization (EM) algorithm of Rubin and Thayer (1982) to obtain $\widehat{\boldsymbol{\theta}}$ for the confirmatory factor analysis (CFA) model when data are incomplete. Generalization of his algorithm, however, to more complex mean and covariance structure models is not trivial. In addition Jamshidian's (1997) algorithm is a complete-data-based method, however, it does not use (2) as the complete data log-likelihood.

To overcome the shortcomings of the complete data methods just discussed, here we propose an EM algorithm whose implementation for a gen-

eral mean and covariance structure model is simple. It utilizes the modules already available in a standard complete data program. Our main goal is to facilitate extension of a complete data program to handling an incomplete data problem for a general mean and covariance structure model. In Section 2 we describe four algorithms for parameter estimation. In section 3 we discuss methods of obtaining standard errors. Section 4 discusses test of hypothesis. Finally Section 5 contains examples to evaluate the procedures discussed in sections 1–4. Moreover, an example is used to discuss sensitivity of the very commonly used MI and LD estimates as well as the ML estimates to the three missing data mechanisms of MCAR, MAR, and NMAR. Finally, in Section 6 we give a summary and discussion.

2 Algorithms for Parameter Estimation

In this section we describe algorithms for computing $\widehat{\boldsymbol{\theta}}$. In Section 2.1 we propose an EM algorithm and a closely related generalized EM (GEM) algorithm for obtaining $\widehat{\boldsymbol{\theta}}$ (Dempster, Laird, & Rubin, 1977). In Section 2.2 we describe an acceleration of our EM and GEM algorithms. Finally in Sections 2.3 and 2.4 we describe the FS and the FP algorithms. The latter two algorithms discussed are trivial extensions of those given by Lee (1986) and Finkbeiner (1979) to a general mean and covariance structure. Our contribution in this context is mainly to show how the components of each of these algorithms can be computed using the available modules in a complete data program.

To be more specific and for our future reference, we list three modules that are generally available in a complete data program.

Module (a) A module that computes the gradient (score) of $\mathcal{L}_x$, at a point $\boldsymbol{\theta}$, for given values of $\bar{\boldsymbol{x}}$ and $\boldsymbol{S}$. We denote this gradient by

$$\mathbf{g}_x(\boldsymbol{\theta}|\bar{\boldsymbol{x}}, \boldsymbol{S}) = \frac{\partial}{\partial \boldsymbol{\theta}} \mathcal{L}_x(\boldsymbol{\theta}|\bar{\boldsymbol{x}}, \boldsymbol{S}).$$

Module (b) A module that computes the Fisher information matrix

$$\mathcal{I}_x(\boldsymbol{\theta}) = E\left(\frac{\partial^2 \mathcal{L}_x}{\partial \boldsymbol{\theta}\partial \boldsymbol{\theta}}\right).$$

Module (c) A module, or collection of modules that maximize $\mathcal{L}_{\ x}(\boldsymbol{\theta}|\bar{\boldsymbol{x}}, \boldsymbol{S})$ with respect to $\boldsymbol{\theta}$ for given values of $\bar{\boldsymbol{x}}$ and $\boldsymbol{S}$.

The ith element of $\mathbf{g}_{\ x}$ is given by

$$\frac{\partial \mathcal{L}_x}{\partial \boldsymbol{\theta}_i} = \frac{n}{2}\text{trace}\left\{\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta})\left[\boldsymbol{S} - \boldsymbol{\Sigma}(\boldsymbol{\theta}) + (\boldsymbol{\mu}(\boldsymbol{\theta}) - 2\,\bar{\boldsymbol{x}})\boldsymbol{\mu}(\boldsymbol{\theta})^T\right]\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta})\frac{\partial \boldsymbol{\Sigma}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_i} - \boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta})\left[2(\boldsymbol{\mu}(\boldsymbol{\theta}) - \bar{\boldsymbol{x}})(\frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_i})^T\right]\right\},$$

and the (i,j)th element of $\mathcal{I}_{\ x}$ is given by

$$(\mathcal{I}_x)_{ij} = -\frac{n}{2}\text{trace}\left[\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta})\left(\frac{\partial \boldsymbol{\Sigma}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_i}\right)\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta})\left(\frac{\partial \boldsymbol{\Sigma}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_j}\right) + 2\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta})\left(\frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_i}\right)\left(\frac{\partial \boldsymbol{\mu}(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_j}\right)^T\right].$$

Modules (a), (b), and (c) are, for example, available in EQS and LISREL for the Bentler-Weeks (1980) and the LISREL models, respectively. We use (c) for our EM and GEM algorithms, and use (a) and (b) for the FP and the FS algorithms.

2.1 The EM and GEM algorithms

The EM algorithm of Dempster et. al. (1977) is a popular algorithm for ML estimation. It cleverly exploits the relation between the complete and incomplete data. The choice of complete data defines the algorithm. We choose $\boldsymbol{x}_1, \cdots, \boldsymbol{x}_n$ as the complete data for our algorithm. This choice is a natural one for maximizing (1), but surprisingly has not been proposed previously. It, for example, differs from that of Jamshidian (1997). The

EM algorithm is comprised of two steps; an expectation step (E-step), and a maximization step (M-step). At a point $\boldsymbol{\theta}$, the E-step consists of computing

$$Q(\boldsymbol{\theta}', \boldsymbol{\theta}) = E^* \left[\mathcal{L}_x(\boldsymbol{\theta}' | \bar{\boldsymbol{x}}, \boldsymbol{S}) \right], \tag{3}$$

where $E^*(\cdot) = E(\cdot | \boldsymbol{Y}, \boldsymbol{\theta})$. The M-step consists of maximizing $Q(\boldsymbol{\theta}\ \ ', \boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}'$ to obtain a new point, say $\widetilde{\boldsymbol{\theta}}$. The iteration process continually replaces $\boldsymbol{\theta}$ by $\widetilde{\boldsymbol{\theta}}$ and repeats the E and M steps until the sequence of values of $\boldsymbol{\theta}$ hopefully converges to $\widehat{\boldsymbol{\theta}}$. In our setting (3) can be written as

$$\begin{aligned} Q(\boldsymbol{\theta}', \boldsymbol{\theta}) = (-n/2) \Big\{ & p \log(2\pi) + \log |\boldsymbol{\Sigma}(\boldsymbol{\theta}')| + \\ & \text{trace} \left[\boldsymbol{\Sigma}^{-1}(\boldsymbol{\theta}') \left(\boldsymbol{S}^* - \boldsymbol{\mu}(\boldsymbol{\theta}')(\bar{\boldsymbol{x}}^*)^T - \bar{\boldsymbol{x}}^* \boldsymbol{\mu}(\boldsymbol{\theta}')^T + \boldsymbol{\mu}(\boldsymbol{\theta}') \boldsymbol{\mu}(\boldsymbol{\theta}')^T \right) \right] \Big\}, \end{aligned} \tag{4}$$

where

$$\boldsymbol{S}^* = (1/n) \sum_{i=1}^{n} E^* \left(\boldsymbol{x}_i \boldsymbol{x}_i^T \right) \tag{5}$$

and

$$\bar{\boldsymbol{x}}^* = (1/n) \sum_{i=1}^{n} E^* \left(\boldsymbol{x}_i \right). \tag{6}$$

To give explicit formulas for computing the expectations in (5) and (6) we simplify our notation by temporarily dropping the i indices, and thus we denote a typical case by $\boldsymbol{x}$ instead of $\boldsymbol{x}_{\ i}$. We use the imprecise but convenient notation $\boldsymbol{x}^{\ T} = (\boldsymbol{y}_o^T, \boldsymbol{y}_m^T)$, where $\boldsymbol{y}_{\ o}$ represents the observed part of $\boldsymbol{x}$ (previously denoted by $\boldsymbol{y}_{\ i}$ for case i) and $\boldsymbol{y}_{\ m}$ is the missing part. Then based on the observed and missing values we partition $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ as

$$\boldsymbol{\mu} = \begin{pmatrix} \boldsymbol{\mu}_o \\ \boldsymbol{\mu}_m \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} \boldsymbol{\Sigma}_{oo} & \boldsymbol{\Sigma}_{om} \\ \boldsymbol{\Sigma}_{mo} & \boldsymbol{\Sigma}_{mm} \end{pmatrix},$$

where here we also drop the arguments of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ whenever they are evaluated at $\boldsymbol{\theta}$. Now

$$E^*(\boldsymbol{x}) = \begin{pmatrix} \boldsymbol{y}_o \\ \boldsymbol{y}_m^* \end{pmatrix}, \tag{7}$$

with $\boldsymbol{y}_m^* = \boldsymbol{\mu}_m + \boldsymbol{\Sigma}_{mo}\boldsymbol{\Sigma}_{oo}^{-1}(\boldsymbol{y}_0 - \boldsymbol{\mu}_o)$, and

$$E^*(\boldsymbol{x}\boldsymbol{x}^T) = \begin{pmatrix} \boldsymbol{y}_o\boldsymbol{y}_o^T & \boldsymbol{y}_o(\boldsymbol{y}_m^*)^T \\ \boldsymbol{y}_m^*\boldsymbol{y}_o^T & E^*(\boldsymbol{y}_m\boldsymbol{y}_m^T) \end{pmatrix}, \tag{8}$$

with

$$E^*(\boldsymbol{y}_m\boldsymbol{y}_m^T) = \boldsymbol{\Sigma}_{mm} - \boldsymbol{\Sigma}_{mo}\boldsymbol{\Sigma}_{oo}^{-1}\boldsymbol{\Sigma}_{om} + \boldsymbol{y}_m^*(\boldsymbol{y}_m^*)^T.$$

Formulas (7) and (8) can be used to compute each term in the summations (5) and (6) respectively. In practice the pattern of missing data varies from case to case, and therefore the vector $\boldsymbol{\mu}$ and the matrix $\boldsymbol{\Sigma}$ are partitioned according to each pattern. As an example, if $p = 4$ and for a case say only variables 2 and 4 are observed, then $\boldsymbol{y}_o$ will be a 2×1 vector of the observed values, $\boldsymbol{\mu}_o$ is the subvector of $\boldsymbol{\mu}$ with its elements being the second and fourth element of $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}_{oo}$ is a 2×2 submatrix of $\boldsymbol{\Sigma}$ obtained by deleting the row and columns 1 and 3, $\boldsymbol{\Sigma}_{om}$ is the submatrix of $\boldsymbol{\Sigma}$ obtained by deleting rows 1 and 3 and columns 2 and 4 from $\boldsymbol{\Sigma}$. $\boldsymbol{\Sigma}_{om}$ and $\boldsymbol{\Sigma}_{mm}$ are similarly defined.

To recap, given a starting value $\boldsymbol{\theta}$, the EM algorithm proceeds as follows:

Step 1. Compute $\boldsymbol{S}^*$ and $\bar{\boldsymbol{x}}^*$ defined in (5) and (6). This mainly involves some simple matrix operations that do not depend on the structures of $\boldsymbol{\mu}(\boldsymbol{\theta})$ and $\boldsymbol{\Sigma}(\boldsymbol{\theta})$.

Step 2. Maximize $Q(\boldsymbol{\theta}', \boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}'$. Denote the maximum point by $\widetilde{\boldsymbol{\theta}}$. Note that

$$Q(\boldsymbol{\theta}', \boldsymbol{\theta}) = \mathcal{L}_x(\boldsymbol{\theta}' | \bar{\boldsymbol{x}}^*, \boldsymbol{S}^*).$$

Therefore this step can be carried out by a complete data program [see module (c)].

Step 3. If convergence is not achieved, replace $\boldsymbol{\theta}$ by $\widetilde{\boldsymbol{\theta}}$ and go to Step 1, otherwise stop.

A disadvantage of the EM algorithm for our problem here is that its Step 2 is generally iterative. GEM, proposed by Dempster et. al. (1977) is a

modification of EM that allows us to avoid iterations in Step 2. Instead of requiring the maximum of $Q(\boldsymbol{\theta}', \boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}'$, GEM only requires a point $\widetilde{\boldsymbol{\theta}}$ in Step 2 such that

$$Q(\widetilde{\boldsymbol{\theta}}, \boldsymbol{\theta}) > Q(\boldsymbol{\theta}, \boldsymbol{\theta}). \tag{9}$$

Dempster et. al. (1977) showed that the GEM algorithm, like the EM algorithm, is globally convergent. Theoretically any method can be adopted in the GEM algorithm to obtain $\widetilde{\boldsymbol{\theta}}$. A good choice, however, results in faster convergence. We propose using one step of the Fisher-Scoring algorithm with step-halving (see e.g., Lee & Jennrich, 1979). This gives a point $\widetilde{\boldsymbol{\theta}}$ that satisfies (9). Our choice of the Fisher-scoring step is also motivated by the fact that existing programs and recent theoretical discussions (see e.g., Cudeck et. al., 1993 and Browne & Du Toit, 1992) use and recommend starting their iterative process in the direction of the Fisher-scoring step.

2.2 The QN1 Algorithm

It is well-known that the EM and GEM algorithms converge slowly when applied to some problems. A number of methods have been proposed to accelerate the EM algorithm. Jamshidian and Jennrich (1997a) give a short review of these methods and propose "a pure accelerator", which they call QN1. It is called a pure accelerator since it only uses the EM steps for acceleration, and it is called QN1 since it is the first of the two acceleration methods based on the quasi-Newton algorithm that they proposed. Practically any accelerator can be used here. We chose QN1 since it is both simple to implement and effective in accelerating the EM algorithm.

To describe the QN1 algorithm, let $\widetilde{\mathbf{g}}(\boldsymbol{\theta})$ denote the EM step at $\boldsymbol{\theta}$. That is $\widetilde{\mathbf{g}}(\boldsymbol{\theta}) = \widetilde{\boldsymbol{\theta}} - \boldsymbol{\theta}$, where $\widetilde{\boldsymbol{\theta}}$ is obtained by using one cycle of the EM algorithm described in Section 2.1, starting from $\boldsymbol{\theta}$. Then the QN1 algorithm proceeds as follows:

Starting with $\boldsymbol{\theta}$ and $A = -I$, the negative of the identity matrix,

Step 1. Compute $\tilde{\mathbf{g}} = \tilde{\mathbf{g}}(\boldsymbol{\theta})$, $\Delta\boldsymbol{\theta} = -A\,\tilde{\mathbf{g}}$, and $\Delta\,\tilde{\mathbf{g}} = \tilde{\mathbf{g}}(\boldsymbol{\theta} + \Delta\boldsymbol{\theta}) - \tilde{\mathbf{g}}$.

Step 2. Using $\Delta\boldsymbol{\theta}$ and $\Delta\,\tilde{\mathbf{g}}$, replace A by $A + \Delta A$, where

$$\Delta A = \frac{(\Delta\boldsymbol{\theta} - A\Delta\,\tilde{\mathbf{g}})\Delta\boldsymbol{\theta}^{\,T} A}{\Delta\boldsymbol{\theta}^{\,T} A \Delta\tilde{\mathbf{g}}}.$$

Step 3. If convergence is not achieved, replace $\boldsymbol{\theta}$ by $\boldsymbol{\theta} + \Delta\boldsymbol{\theta}$, $\tilde{\mathbf{g}}$ by $\tilde{\mathbf{g}} + \Delta\tilde{\mathbf{g}}$, and go to Step 1, otherwise stop.

The QN1 algorithm was proposed by Jamshidian and Jennrich (1997a) to accelerate the EM algorithm. In examples of Section 5 we have used QN1 to accelerate the GEM algorithm. This is done by substituting the EM step $\tilde{\mathbf{g}}$ by the GEM step $\widetilde{\boldsymbol{\theta}} - \boldsymbol{\theta}$. Our experience with the examples of Section 5 shows that this acceleration is as effective for the GEM algorithm as it is for the EM algorithm.

2.3 The Fisher-Scoring Algorithm

In this section we describe the FS algorithm proposed by Lee (1986). We extend Lee's work by showing how the FS algorithm can be implemented using complete data routines. Starting at an initial point $\boldsymbol{\theta}$, the FS algorithm proceeds as follows:

Step 1. Compute $\mathcal{I}_{\,y}(\boldsymbol{\theta}) = E(\partial^2\mathcal{L}_y/\partial\boldsymbol{\theta}\partial\boldsymbol{\theta})$, the Fisher-information matrix at $\boldsymbol{\theta}$, and $\mathbf{g}_y(\boldsymbol{\theta})$, the gradient of $\mathcal{L}_{\,y}$ at $\boldsymbol{\theta}$.

Step 2. Set $\Delta\boldsymbol{\theta} = -\mathcal{I}_{\,y}^{\,-1}(\boldsymbol{\theta})\mathbf{g}_y(\boldsymbol{\theta})$.

Step 3. Set $\alpha = 1$, $\widetilde{\boldsymbol{\theta}} = \boldsymbol{\theta} + \alpha\Delta\boldsymbol{\theta}$. If $\mathcal{L}_{\,y}(\widetilde{\boldsymbol{\theta}}) > \mathcal{L}_y(\boldsymbol{\theta})$ go to Step 4, otherwise continue halving α until $\mathcal{L}_{\,y}(\widetilde{\boldsymbol{\theta}}) > \mathcal{L}_y(\boldsymbol{\theta})$.

Step 4. If convergence is not achieved, replace $\boldsymbol{\theta}$ by $\widetilde{\boldsymbol{\theta}}$ and go to Step 1, otherwise stop.

Note that each iteration requires computation of the gradient and the Fisher-information matrix. We show how these quantities can be obtained using modules in a complete data program.

Using a result due to Fisher (1925) it can be shown that

$$\mathbf{g}_y(\boldsymbol{\theta}) = E^*\left[\frac{\partial}{\partial \boldsymbol{\theta}}\mathcal{L}_x(\boldsymbol{\theta}|\bar{\boldsymbol{x}}, \boldsymbol{S})\right] = \mathbf{g}_x(\boldsymbol{\theta}|\bar{\boldsymbol{x}}^*, \boldsymbol{S}^*). \tag{10}$$

This result relates the gradients of the complete and observed data log-likelihoods. More specifically it reveals that $\mathbf{g}_y(\boldsymbol{\theta})$ can simply be obtained by inputting $\bar{\boldsymbol{x}}^*$ and $\boldsymbol{S}^*$ in place of $\bar{\boldsymbol{x}}$ and $\boldsymbol{S}$ in the gradient module of a complete data program [see module (a)].

The (k,l)th element of the Fisher-Information matrix $\mathcal{I}_y(\boldsymbol{\theta})$ is given by

$$[\mathcal{I}_y(\boldsymbol{\theta})]_{kl} = (-1/2)\sum_{i=1}^{n} \text{trace}\Big\{\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{\theta})\left(\frac{\partial \boldsymbol{\Sigma}_i(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_k}\right)\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{\theta})\left(\frac{\partial \boldsymbol{\Sigma}_i(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_l}\right) + 2\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{\theta})\left(\frac{\partial \boldsymbol{\mu}_i(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_k}\right)\left(\frac{\partial \boldsymbol{\mu}_i(\boldsymbol{\theta})}{\partial \boldsymbol{\theta}_l}\right)^T\Big\}, \tag{11}$$

where $\boldsymbol{\theta}_k$ and $\boldsymbol{\theta}_l$ are the kth and lth elements of $\boldsymbol{\theta}$. $\mathcal{I}_y$ does not depend on the observed values themselves, but it does depend on the pattern of missing data through $\boldsymbol{\Sigma}_i$ and $\boldsymbol{\mu}_i$. A routine that computes the Fisher-information matrix for a complete data problem [see module (b)] can be used once for every existing pattern of missing data to obtain each summand in (11). Then $\mathcal{I}_y$ is obtained as a weighted average of the Fisher-information matrices obtained for each pattern with the weights being the number of cases observed for that pattern.

2.4 The Fletcher-Powell Algorithm

Finkbeiner (1979) used the FP algorithm to obtain $\widehat{\boldsymbol{\theta}}$ in the context of exploratory factor analysis. We give this algorithm for the case of general mean and covariance structures.

Starting at a point $\boldsymbol{\theta}$, obtain $B = -\mathcal{I}_y^{-1}(\boldsymbol{\theta})$ and proceed as follows:

Step 1. Set $\mathbf{d} = B\mathbf{g}_y(\boldsymbol{\theta})$.

Step 2. Maximize $\mathcal{L}_y(\boldsymbol{\theta} + \alpha\mathbf{d})$ with respect to α to obtain $\widetilde{\boldsymbol{\theta}}$ and set $\Delta\boldsymbol{\theta} = \alpha\mathbf{d}$.

Step 3. Compute $\mathbf{g}_y(\widetilde{\boldsymbol{\theta}})$ and set $\Delta\mathbf{g} = \mathbf{g}_y(\widetilde{\boldsymbol{\theta}}) - \mathbf{g}_y(\boldsymbol{\theta})$.

Step 4. Replace B by $B + \Delta B$, where

$$\Delta B = \frac{\Delta\boldsymbol{\theta}\Delta\boldsymbol{\theta}^T}{\Delta\boldsymbol{\theta}^T\Delta\mathbf{g}} - \frac{(\boldsymbol{\theta}^T B\Delta\mathbf{g})^2}{\Delta\mathbf{g}^T B\Delta\mathbf{g}}$$

Step 5. If convergence is not achieved replace $\boldsymbol{\theta}$ by $\widetilde{\boldsymbol{\theta}}$ and go to Step 1, otherwise Stop.

As described in Section 2.3, $\mathbf{g}_y(\boldsymbol{\theta})$ and $\mathcal{I}_y(\boldsymbol{\theta})$ can be computed using complete data routines.

3 Computing Standard Errors

Presently popular software packages in mean and covariance structure analysis produce standard errors for the parameter estimates based on the inverse of the Fisher information matrix. This can also be done for the missing data problem by using the Fisher information as defined in (12). If we follow this approach, as we may, there is nothing further on this topic that needs discussion. There are other alternatives, however, that may be computationally more feasible while being theoretically at least as defensible.

Let $H(\widehat{\boldsymbol{\theta}})$ denote the negative of the Hessian of $\mathcal{L}_y(\boldsymbol{\theta})$, known as the observed information matrix. To obtain standard errors, Efron and Hinkley (1978) suggested using $V(\widehat{\boldsymbol{\theta}}) = H^{-1}(\widehat{\boldsymbol{\theta}})$. This quantity has not received much attention in the area of mean and covariance structure analysis probably because analytic computation of $H(\widehat{\boldsymbol{\theta}})$ is generally complex for many problems

in this area. Dolan and Molenaar (1991) used these standard errors as "exact SEs" against which they evaluated alternatives such as those based on the Fisher information matrix in the context of covariance structure analysis without missing data.

There have been a few proposals for computing $H(\boldsymbol{\theta})$, specially in the context of missing data and the EM algorithm. Jamshidian and Jennrich (1997b) give a review and propose a method that in their experience worked well. Their method uses the first order Richardson extrapolation of the center difference to differentiate the score vector. More specifically, they propose approximating the jth column of $H(\ \widehat{\boldsymbol{\theta}})$ by

$$G_j(\boldsymbol{\theta}) = -\frac{\mathbf{g}_y(\boldsymbol{\theta} - 2h_j\mathbf{e}_j) - 8\mathbf{g}_y(\boldsymbol{\theta} - h_j\mathbf{e}_j) + 8\mathbf{g}_y(\boldsymbol{\theta} + h_j\mathbf{e}_j) - \mathbf{g}_y(\boldsymbol{\theta} + 2h_j\mathbf{e}_j)}{12h_j}, \tag{12}$$

where $\mathbf{e}_j$ is the unit vector with all of its elements equal to zero except for its jth element which is equal to 1, and $h_{\ j} = 10^{-4}\max(1, |\boldsymbol{\theta}_j|)$ with $\boldsymbol{\theta}_j$ denoting the jth element of $\boldsymbol{\theta}$. Thus

$$\widetilde{H}(\widehat{\boldsymbol{\theta}}) = [G_1(\widehat{\boldsymbol{\theta}}), \cdots, G_q(\widehat{\boldsymbol{\theta}})] \tag{13}$$

is used to approximate $H(\ \widehat{\boldsymbol{\theta}})$, where q is the number of parameters in $\boldsymbol{\theta}$. Note that $\widetilde{H}(\widehat{\boldsymbol{\theta}})$ may not be symmetric. Jamshidian and Jennrich (1997b) propose using the degree of asymmetry in $\widetilde{H}(\widehat{\boldsymbol{\theta}})$ to predict the accuracy of $\widetilde{H}(\widehat{\boldsymbol{\theta}})$ as an estimate of $H(\boldsymbol{\theta})$. We will assess the accuracy of $\widetilde{H}(\widehat{\boldsymbol{\theta}})$ for our examples of Section 5 and see how well the asymmetry can predict its numerical errors.

4 Test of Hypothesis

In a mean and covariance structure analysis it is of interest to test the null hypothesis

$$H_0: \quad \boldsymbol{\Sigma} = \boldsymbol{\Sigma}(\boldsymbol{\theta}), \ \ \boldsymbol{\mu} = \boldsymbol{\mu}(\boldsymbol{\theta}). \tag{14}$$

for a given mean and covariance structure $\boldsymbol{\mu}(\boldsymbol{\theta})$ and $\boldsymbol{\Sigma}(\boldsymbol{\theta})$, versus the alternative that the mean vector and the covariance matrix have no structure. For this, we propose using the likelihood ratio test

$$\lambda = \frac{max_{\Omega_0} \mathcal{L}_y(\boldsymbol{\theta})}{max_{\Omega} \mathcal{L}_y(\boldsymbol{\theta})}. \tag{15}$$

where Ω represents the space of p by p symmetric matrices and p dimensional vectors over which $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu}$ vary respectively, and Ω_0 is the subspace of Ω restricted by the null hypothesis (14). It is well-known that $-2\log\lambda$, under the normality assumption, is asymptotically distributed as χ^2 with degrees of freedom equal to the difference between the dimensions of Ω and Ω_0. The χ^2 test is of course well known. However, it seems not to have been discussed in the context of mean and covariance structures with missing data.

Note that $-2\log\lambda = -2[\mathcal{L}_y(\widehat{\boldsymbol{\theta}}) - \mathcal{L}_y(\bar{\boldsymbol{\theta}})]$, where $\widehat{\boldsymbol{\theta}}$ is the ML estimate obtained under the structure imposed by (14), and $\bar{\boldsymbol{\theta}}$ is the maximum of $\mathcal{L}_y$ under the assumption that there is no structure on $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$. Both $\widehat{\boldsymbol{\theta}}$ and $\bar{\boldsymbol{\theta}}$ can be obtained using any of the algorithms described in Section 2.

In cases where the hypothesis (14) is not rejected, one may wish to consider testing a more restricted hypothesis. More generally, one may wish to test the null hypothesis

$$H_0: \quad \boldsymbol{\Sigma} = \boldsymbol{\Sigma}(\boldsymbol{\theta}), \;\; \boldsymbol{\mu} = \boldsymbol{\mu}(\boldsymbol{\theta}), \qquad \boldsymbol{\theta} \in \Omega_0$$

versus the alternative hypothesis

$$H_A: \quad \boldsymbol{\Sigma} = \boldsymbol{\Sigma}(\boldsymbol{\theta}), \;\; \boldsymbol{\mu} = \boldsymbol{\mu}(\boldsymbol{\theta}), \qquad \boldsymbol{\theta} \in \Omega_A$$

where Ω_0 is a subspace of Ω_A. For this, the ratio

$$\lambda = \frac{max_{\Omega_0} \text{Ł}_y(\boldsymbol{\theta})}{max_{\Omega_A} \text{Ł}_y(\boldsymbol{\theta})}, \tag{16}$$

may be considered. Again, $-2\log\lambda$ is distributed asymptotically as χ^2 with the degrees of freedom being the difference between the dimensions of Ω_0 and Ω_A.

5 Examples

In this section we use CFA examples to discuss some of the methods mentioned above. In Section 5.1 we compare the cost of the algorithms of Section 2. In Section 5.2 we investigate the effect of missing data mechanism on the ML, MI, and LD estimates using one example. Sections 5.3 and 5.4 apply and evaluate the standard error and test of hypothesis procedures of Sections 3 and 4 to our examples.

In the CFA model

$$\boldsymbol{\Sigma}(\boldsymbol{\theta}) = \Lambda\Phi\Lambda^T + \Psi,$$

where Λ is a p by k matrix of factor loadings, Φ is the matrix of factor covariances, and Ψ is a diagonal matrix containing the unique error variances. As is usual in CFA, we allow some elements in (Λ, Φ, Ψ) to be fixed to constant values. In all of our examples we assume no structure on $\boldsymbol{\mu}(\boldsymbol{\theta})$, but we do not assume it to be zero. Thus, for our examples, elements in $\boldsymbol{\theta}$ consist of the p parameters in $\boldsymbol{\mu}$ and the free parameters in Λ, Φ, and Ψ.

Our first example is similar to one of the examples used by Finkbeiner (1979). We generated 64 data points from a multivariate normal distribution with mean zero and covariance obtained from the population parameters

$$\Lambda^T = \begin{pmatrix} -.35 & .90 & .30 & -.60 & .00 & .00 \\ .00 & .00 & .20 & .60 & -.90 & -.35 \end{pmatrix},$$

$$\text{diag}(\Psi) = (\, .8775 \quad .1900 \quad .8700 \quad .2800 \quad .1900 \quad .8775 \,),$$

and $\Phi = I$. The sample size of 64, as Finkbeiner notes, is very near the borderline between large and small samples. We then created missing data using the pattern named "$m = 1$" by Finkbeiner (1979). In this pattern, there are 13 complete data cases, and 51 cases with missing variables ranging from 1 to 3. The model used, restricts the zero elements in population Λ to zero and fixes $\Phi = I$.

For our second example we generated 200 data points from a multivariate normal distribution with mean zero and covariance matrix equal to that given on page 98 of Jöreskog and Sörbom (1988). This covariance matrix is based on the nine psychological variables chosen by Jöreskog and Sörbom (1988) from the Holzinger and Swineford (1939) study. We created missing data by deleting 20% of the generated data points at random. This resulted in 75 different missing data patterns with the number of missing variables for each pattern ranging from 1 to 5. There were 30 complete cases. We fit the CFA model described by Jöreskog and Sörbom (1988, page 104) to the resulting data set. The model fit to this data was chosen by Jöreskog and Sörbom on the basis of post-hoc model modification. Hence it may or may not be the "true" model for this covariance matrix.

Our third example shows the performance of the algorithms on a problem with a large number of parameters (174 parameters). Two hundred cases were generated from a multivariate normal distribution with mean 0 and covariance matrix obtained using the following population parameters: $\Phi = I$, the 7 by 7 identity matrix, Ψ a diagonal matrix with its diagonal elements chosen uniformly from the interval $(0, 1)$, and $\Lambda^T = (\, I \quad \tilde{\Lambda}^T \,)$ with $\tilde{\Lambda}$ being a 17 by 7 matrix of uniform numbers from the interval $(0, 1)$. To have some missing data, we deleted a 50 by 12 matrix from the top right and a 50 by 12 matrix from the bottom left of the 200 by 24 data matrix. The model used to fit to the data fixed the upper 7 by 7 matrix in Λ as well as Φ to the identity. This example is similar to Example 6 of Jamshidian and Jennrich (1994).

5.1 Cost of algorithms

In this section we compare the cost of the GEM, QN1, FS, and FP algorithms described in Section 2. We measure cost by the number of floating point operations (FLOPs) required for convergence. The MATLAB program that we

used to code our algorithm counts the required FLOPs. Our comparison does not include the EM algorithm because for our examples the EM algorithm consistently had a higher cost than GEM. The cost per iteration for the EM algorithm is more than that of GEM in its Step 2. Generally, however, EM requires fewer iterations to converge than GEM, but the difference in the number of iterations often is not large enough to offset the extra cost per iteration of the EM algorithm. Also close to the solution, $\widetilde{\boldsymbol{\theta}}$ in the Step 2 GEM is very close to that of EM, making EM and GEM behave similarly close to the solution.

For starting values, we used MI estimates described in Section 1. As in Jamshidian and Jennrich (1994), we take a few GEM steps before starting the accelerator QN1. More specifically, we continue taking GEM steps until the relative gradient (Khalfan, Byrd, & Schnabel, 1993)

$$rg = \max_i \left[\left| \left[\mathbf{g}_y(\boldsymbol{\theta}) \right]_i \right| \frac{\max\{|[\boldsymbol{\theta} + \Delta\boldsymbol{\theta}]_i|, 1\}}{\max\{|\mathcal{L}_y(\boldsymbol{\theta} + \Delta\boldsymbol{\theta})|, 1\}} \right],$$

is less than or equal to 10^{-2}. Analogously, to start the FP algorithm we take Fisher-scoring steps until $rg \leq 10^{-2}$. These initial steps improve on the MI initial values at which QN1 and FP start. The FP algorithm requires a line search algorithm in its Step 2. For this we used that used by Jamshidian and Jennrich (1997a). We stop our algorithms if $rg \leq 10^{-6}$. This convergence criterion was recommended by Khalfan et al. (1993). It results in about six decimal places of accuracy in the value of the log-likelihood and about three to four digits of accuracy in the parameter estimates.

The results of our comparisons for Examples 1–3 are shown in Table 1. For Example 1, FS required half as many iterations as GEM, but both algorithms required about the same number of FLOPs. This is because of the higher cost per iteration of FS on this example. QN1 is faster than GEM by a factor of about 1.6. FP is the fastest algorithm for this example, both in terms of the number of iterations and the number of FLOPs. Compared

Table 1: Cost comparison of four algorithms on Examples 1, 2, and 3

	Example 1		Example 2		Example 3	
Method	Iter.	FLOPS/10^5	Iter.	FLOPS/10^6	Iter.	FLOPS/10^8
GEM	25	5.4	78	11.7	78	5.7
QN1	14	3.3	16	2.5	28	2.1
FS	12	5.5	100	40.6	42	4.5
FP	8	2.8	10	2.9	26	1.0

to QN1, however, it is only faster by a factor of 1.2.

For Example 2, QN1 was the fastest algorithm, accelerating GEM by a factor of 4.7. While FP required a smaller number of iterations than QN1 for this example, FP had a higher cost than QN1 on this example. This higher cost is due to the fact that FP's line searches at each iteration required two function-gradient evaluations causing its cost per iteration to be almost twice as that of QN1. The FS algorithm did not perform well on this example. We stopped FS after 100 iterations. At this point $rg = 10^{-4}$ and the cost of FS was about 16 times as that of QN1.

For Example 3, GEM was the slowest algorithm closely followed by FS. QN1 accelerated GEM by a factor of about 2.7. FP was the fastest algorithm, beating QN1 by a factor of about 2.1. The number of iterations for QN1 and FP are close, but as opposed to the previous example the cost per iteration of FP is less than that of QN1 here. This is because for this example, in its Step 2, QN1 requires inversion of a 174 by 174 matrix at every iteration. Such an operation is not required by FP, at least after the few initial Fisher-scoring steps that are taken.

To conclude this section, FP and QN1 are close competitors on our examples. FS does not perform very well overall. GEM is the simplest algorithm

to implement, however, since QN1 effectively accelerates GEM and is a simple modification of GEM, it should be considered. An advantage of GEM over QN1 is its global convergence property.

5.2 The effect of the missing data mechanism

As noted in the Introduction theoretically the ML estimator $\widehat{\boldsymbol{\theta}}$ requires the missing data mechanism to be either MCAR or MAR. In this section we use one example to discuss sensitivity of the ML estimates to the missing data mechanism. We also compare the ML estimates to the often used MI and LD estimates. The example here is mainly to give insight. Based on our experiments with a number of examples, the results discussed here seem to be very typical. A simulation study of these results would indeed be valuable.

Our example in this section starts with a complete data set having 500 cases generated artificially from a multivariate normal population with $\boldsymbol{\mu} = 0$ and covariance matrix obtained from the population parameters

$$\Lambda^T = \begin{pmatrix} .81 & .66 & .77 & 0 & 0 & 0 & 0 & 0 & .54 \\ 0 & 0 & 0 & .86 & .97 & .78 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & .74 & 1 & .5 \end{pmatrix}, \ \Phi = \begin{pmatrix} 1 & .55 & .40 \\ .55 & 1 & .25 \\ .40 & .25 & 1 \end{pmatrix},$$

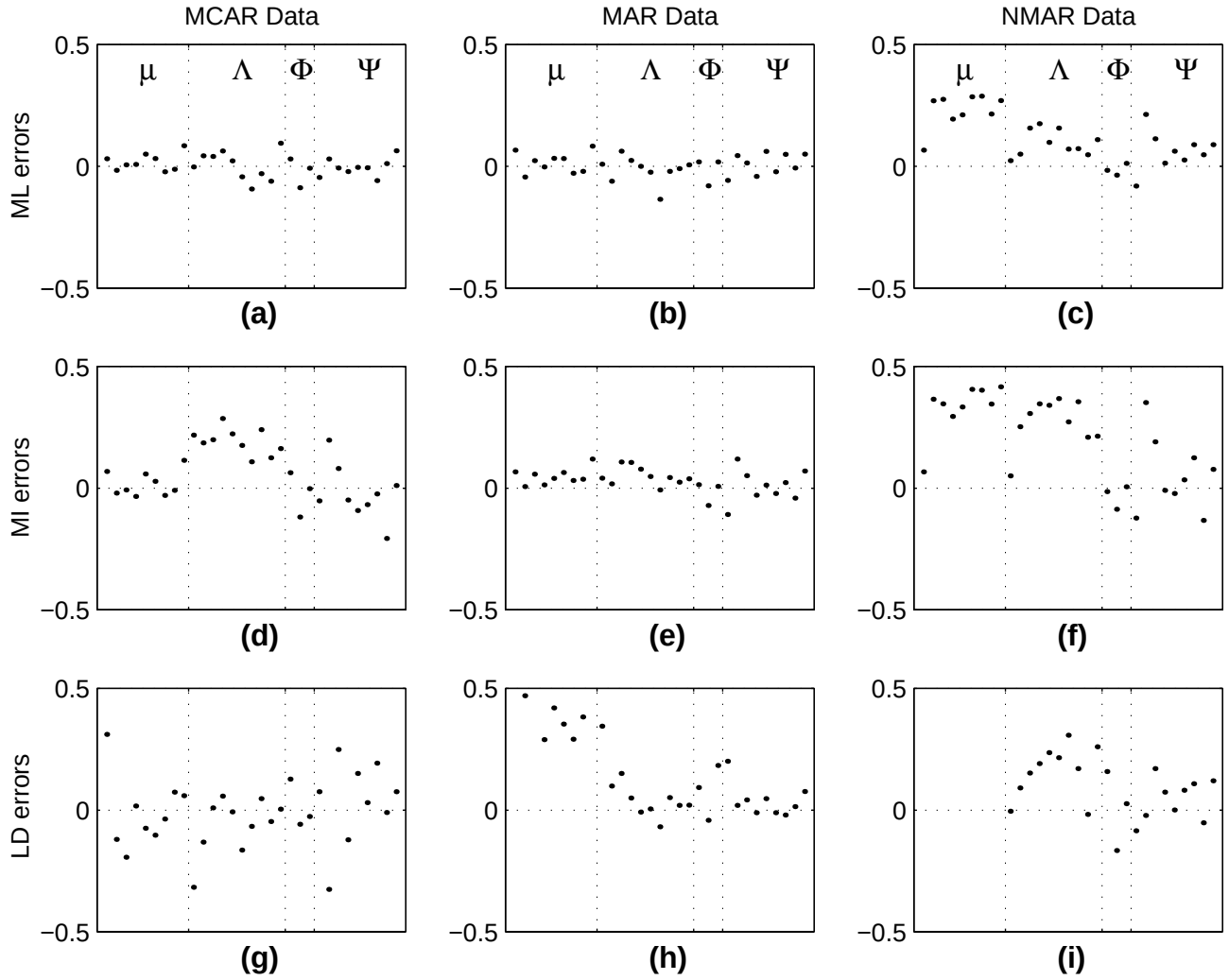
and $\text{diag}(\Psi) = (.54,\ .91,\ .61,\ .23,\ .24,\ .32,\ .53,\ .12,\ .41)$. These population values are the ML estimates obtained in the Example 2 above. The sample size of 500 was used to have small sampling variability to better illustrate the effect of missingness. Using the 500 by 9 data matrix, we created three data sets with missing data by the following three procedures: (i) Delete each data point with a probability of .25, using a uniform random number generator. This gives an MCAR data set. (ii) Standardize observed values of the variable 1 to have mean 0 and variance 1. Denote the *ith* case of the standardized variable by v_{i1}. For $i = 1, \cdots, 500$ and $j = 2, \cdots, 9$, calculate $u_{ij} = v_{i1} + z_{ij}$, where z_{ij}'s are independent observations from the standard normal distribution. Delete the (i, j)th observation if $u_{ij} > .8$. This results

in an MAR data set. Note that no missing values are created in variable 1, and the missingness on variables 2–9 depend on the values observed for variable 1. (iii) Let $u_{ij} = v_{ij} + z_{ij}$, where z_{ij} is as defined above and v_{ij} is the i-th case of the variable j standardized to have mean 0 and variance 1. For $i = 1, \cdots, 500$ and $j = 2, \cdots, 9$ delete the (i, j)th observation if $u_{ij} > .8$. This gives a data set that is NMAR. Note that again there are no missing values in variable 1, and the missingness in variables 2–9 depends on the missing values themselves. These three procedures resulted in approximately 25% missing data in each case. The MCAR data have 187 patterns of missing with 44 complete cases. The MAR data have 154 patterns of missing and 144 complete cases, and the NMAR data have 162 patterns of missing with 66 complete cases.

Figure 1 contains 9 plots comparing the three methods ML, MI, and LD on the three data sets just described. The vertical axis of each plot is the difference between the values of the population parameters and their corresponding estimates obtained from each of the three methods mentioned. The horizontal axis of each plot shows the parameter numbers. Parameters 1–9 are the mean parameters $\boldsymbol{\mu}$, parameters 10–19 are the free (nonzero) parameters in Λ, parameters 20–22 are parameters in Φ, and parameters 23–31 are parameters in Ψ.

The MCAR data: The plots in the first column of Figure 1 show the errors in the ML, MI, and LD estimates for the MCAR data set. The ML errors, shown in Figure 1(a), are very close to zero. In fact Figure 1(a) is very similar to the plot obtained when fitting the model to the 500 by 9 complete data. (We have not included the complete data plot here.) Overall, the MI errors for estimates of $\boldsymbol{\mu}$, Φ, and Ψ are only a bit larger than their corresponding ML errors. What stands out, however, is that the MI estimates of Λ are biased in this case. The LD estimates do not show any bias, but their errors seem to be larger than both the ML and the MI. The main reason for the larger

Figure 1: Comparison of ML, MI, and LD on three data sets



LD errors is that the LD estimates are based on a sample of size $n = 44$. Unbiasedness of the LD may be explained by the fact that the sample used here is a random subsample of the sample of size $n = 500$.

The MAR data: As expected, the ML errors in this case, shown in Figure 1(b), are small and no bias is apparent. The MI errors in Figure 1(e) also have small errors. Note, however, that the MI estimates of both $\boldsymbol{\mu}$ and Λ show a slight bias. The LD errors, shown in Figure 1(h), are larger as compared to errors in Figures 1(b) and 1(e). Again this is mainly due to a smaller sample used by LD. The bias in estimates of $\boldsymbol{\mu}$ stands out clearly. In fact three of the parameters in μ had errors larger than .5 and are not shown in Figure 1(h).

The NMAR data: It is of interest to investigate ML estimates for data that are neither MCAR nor MAR. Figure 1(c) shows ML errors for our NMAR data. These errors are larger than the ML errors of our MCAR and MAR data. Also all the parameter estimates show bias. For this data, the MI estimates have even larger errors than the ML estimates with bias apparent in the estimates of Λ, $\boldsymbol{\mu}$, and Φ. Interestingly, the ML and MI errors of the first parameter in $\boldsymbol{\mu}$ is small. Recall that the first variable in the NMAR data does not have any missing values. Finally, as shown in Figure 1(i), the LD estimates of the NMAR data are very poor. The estimates of $\boldsymbol{\mu}$ were biased with a minimum error of .7. Again estimates of Λ and to some extent Ψ also show bias.

To summarize, as expected ML performed well under the MCAR and MAR missing data mechanisms, and overall the ML estimates were superior to the MI and LD estimates. For the NMAR data, all three methods produced biased estimates with LD having the largest bias. The MI estimates were biased in all cases, and the LD estimates were biased for the MAR and the NMAR data sets.

5.3 Standard errors

In this section we assess the accuracy of the $\widetilde{H}(\widehat{\boldsymbol{\theta}})$ for the Examples of Section 5.1. To do this comparison, we use the *Maximum Relative Error* (MRE)

$$\text{MRE} = \max_{\ell} \frac{\ell^T(H^{-1}(\widehat{\boldsymbol{\theta}}) - \widetilde{H}^{-1}(\widehat{\boldsymbol{\theta}}))\ell}{\ell^T H^{-1}(\widehat{\boldsymbol{\theta}})\ell} \tag{17}$$

introduced by Jamshidian and Jennrich (1997b). This is the upper bound in the relative error when estimating the variance of an arbitrary linear combination $\ell^T\widehat{\boldsymbol{\theta}}$. For example, when $\ell = \mathbf{e}_j$ MRE indicates the maximum relative error in estimating the standard error of the jth element of $\widehat{\boldsymbol{\theta}}$. For our examples, we obtained the values of $H(\widehat{\boldsymbol{\theta}})$ by analytical differentiation of $\mathcal{L}_y(\boldsymbol{\theta})$.

The MRE's for Examples 1–3 were 3×10^{-9}, 2×10^{-11}, and 1×10^{-10}, respectively. These values indicate that $\widetilde{H}(\widehat{\boldsymbol{\theta}})$ is practically identical to $H(\widehat{\boldsymbol{\theta}})$. This is impressive since the analytic derivation and coding of the the derivatives required for $H(\widehat{\boldsymbol{\theta}})$ is much more complex than that required for $\widetilde{H}(\widehat{\boldsymbol{\theta}})$. In fact the difference in complexity increases as the structures imposed on the mean and covariance increases.

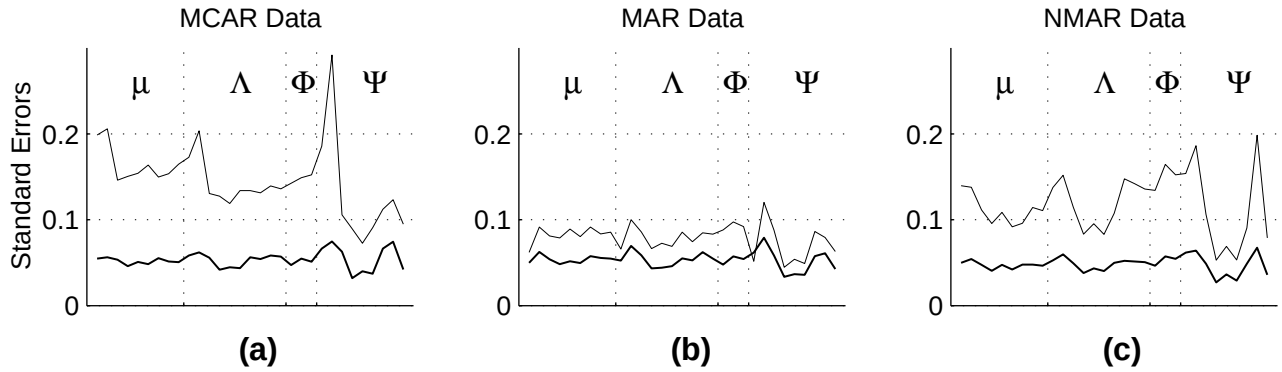
As noted in Section 3, the asymmetry in $\widetilde{H}(\widehat{\boldsymbol{\theta}})$ can be used to measure numerical errors. Jamshidian and Jennrich (1997b) suggest using

$$\widehat{\text{MRE}} = \max_{\ell} \frac{\ell^T(\widetilde{H}^{-1}(\widehat{\boldsymbol{\theta}}) - (\widetilde{H}^{-1}(\widehat{\boldsymbol{\theta}}))^T)\ell}{\ell^T\widetilde{H}^{-1}(\widehat{\boldsymbol{\theta}})\ell} \tag{18}$$

to estimate the MRE's. For Examples 1–3 the values of $\widehat{\text{MRE}}$ were 1×10^{-9}, 6×10^{-11}, and 2×10^{-9}, respectively. Comparison of these values to the MRE values shown above indicates that $\widehat{\text{MRE}}$ is a very good estimate of MRE.

Finally, we evaluate standard errors of ML and LD for the MACR, MAR, and NMAR data sets of Section 5.2. Note that standard errors of the MI estimates are not easy to obtain since it is hard to account for the variability of the imputed mean values. Figures 2(a)-(c) show the standard errors of ML (heavier line) and LD (lighter line) estimates of the MCAR, MAR, and the

Figure 2: The standard errors for ML and LD



NMCAR data. In all three figures the standard errors correctly reflect the smaller errors in the ML estimates as compared to LD estimates. The ML standard errors for both MCAR and MAR are very close. The LD standard errors of MAR are smaller than those of MCAR. This is because the MAR data set has 144 complete cases whereas our MCAR data set has only 44 complete cases. The ML standard errors of the NMAR data are as small as the ML standard errors of the MCAR and MAR data. Comparing the variability of the ML errors in Figures 1(a)–1(c) to their corresponding variability measured by the standard errors shown in Figures 2(a)–2(c), it seems that the standard errors underestimate the variability of the ML estimates for the NMAR data.

To summarize this section, $\widetilde{H}(\widehat{\boldsymbol{\theta}})$ approximates $H(\boldsymbol{\theta})$ very well. These standard errors reflect variability of the ML and LD estimates for the MCAR and MAR data set well. They, however, underestimate the variability of ML and LD estimates for our NMAR data.

5.4 Test of hypothesis

We have used the likelihood ratio test (16) to test the structures that we assumed for our Examples 1 to 3 of Section 5.1. Table 5 gives the values

Table 2: The restricted and unrestricted values of the log-likelihood, the value of $-2\log\lambda$, degrees of freedom, and p-values for examples 1 to 4 of Section 3.

Example	1	2	3
$\max_{\Omega_0} \mathcal{L}_y(\boldsymbol{\theta})$	-436.93	-1880.10	-4701.82
$\max_{\Omega} \mathcal{L}_y(\boldsymbol{\theta})$	-432.55	-1840.68	-4622.10
$-2\log\lambda$	8.76	78.83	159.44
d.f.	7	23	150
p-values	0.2703	0.0000	0.2835

$\mathcal{L}(\widehat{\boldsymbol{\theta}})$, $\mathcal{L}(\bar{\boldsymbol{\theta}})$, $-2\log\lambda$, the degrees of freedom, and the associated p-value for our examples. Recall that the data for the Examples 1 and 3 were generated using the same structure that we used in our model. Therefore we expect not to reject the null hypothesis (15) for these examples. Indeed the p-values are in accordance with our expectation.

The p-value for Example 2 does not support the hypothesized model in Example 2. To investigate whether this was caused by the missing data or that in fact the model does not fit the covariance matrix used to generate the data, we tested the model using 10 complete data sets generated from the covariance matrix of Example 2. Using the significance level of .01, the model was rejected in 9 out of 10 cases. So the results for both the missing data case and the complete data case are consistent. The model proposed by Jöreskog and Sörbom (1988) based on post-hoc model modification may not be a good one for the given covariance matrix.

6 Summary and discussion

One of our aims in this paper was to show how complete data programs and the machinery within them can be used to obtain maximum likelihood estimates of the parameters in a mean and and covariance structure model when data are incomplete. As this idea was presented for a number of algorithms in Section 2, it can be used with most gradient optimization algorithms to maximize the observed likelihood function when data are missing. The algorithms discussed can also be extended to estimate parameters of mean and covariance structure models with grouped incomplete data. The key is to obtain $\boldsymbol{S}^*$ and $\bar{\boldsymbol{x}}^*$ for each group separately using Equations (5) and (6). These quantities then can be utilized in formulation of a GEM algorithm or in obtaining the gradient, using the Fisher's identity (10), to formulate an FS or FP algorithm much the same way as we did for the single group case in Section 2.

The method of Jamshidian and Jennrich (1997b) to computing the observed information for standard errors, described in Section 3, is simple and produces very accurate results. Two points of caution, however, about the observed information standard errors are as follows: (i) if data are NMAR, these standard errors seem to underestimate the variability of ML estimates. (ii) These standard errors are not valid for the MI method, as they do not take into account the variability of the imputed means.

Our brief investigation of bias in Section 5.2 showed that ML estimates may be biased for data that are NMAR. MI estimates are generally biased and LD estimates are not biased for the MCAR data, but are biased for MAR and NMAR missing data mechanisms.

With regard to test of hypothesis we discussed the likelihood ratio test in Section 4 and gave examples in Section 2.4. Satorra (1989) has reviewed a number of other statistics for the complete data problem. Computationally

these statistics can be obtained using the methods discussed here. For example, if $\tilde{\boldsymbol{\theta}}$ is an estimate of the parameters obtained under a set of restrictions $\mathbf{a}(\boldsymbol{\theta}) = 0$ then the statistic associated with the score test in our context is

$$S = n\mathbf{g}_y^T(\tilde{\boldsymbol{\theta}})H^{-1}(\tilde{\boldsymbol{\theta}})\mathbf{g}_y(\tilde{\boldsymbol{\theta}}).$$

$\mathbf{g}_y(\tilde{\boldsymbol{\theta}})$ can be obtained from a complete data program using Fisher's identity (10), and $H(\tilde{\boldsymbol{\theta}})$ can be obtained using the Jamshidian and Jennrich (1997b) method. What is of interest and requires further research is to determine the asymptotic distribution of S and possibly the other statistics discussed in Sattora (1989) in presence of missing data.

Finally note that in addition to the assumptions of MCAR and MAR, the assumption of multivariate normality of the data made here is needed for the ML estimates to be asymptotically most efficient. The effect of departures from normality on the ML estimates discussed may be another topic of research. We think, however, that this effect should not be any different than that for the complete data case.

Authors

Mortaza Jamshidian is an Assistant Professor, Department of Statistics, University of Central Florida, Orlando, Florida 32816-2370.

Peter M. Bentler is Professor, Department of Psychology and Center for Statistics, University of California, Los Angeles, 4627 Franz Hall, Los Angeles, CA 90024-1563.

References

[1] Allison, P. D. (1987). Estimation of linear models with incomplete data. In C. Clogg (Ed.), *Sociological methodology,* (pp. 71–103). San Francisco: Jossey-Bass.

[2] Arminger, G. (1994). Specification and estimation of non-standard mean and covariance structure models with Mecosa. In F. Faulbaum (Ed.) *Softstat '93: Advances in statistical software 4* (pp. 13–22). Stuttgart: Gustav Fischer Verlag.

[3] Arminger, G., & Sobel, M. E. (1990). Pseudo-maximum likelihood estimation of mean and covariance structure with missing data. *Journal of the American Statistical Association,* **85**, 195–203.

[4] Bentler, P. M. (1990). EQS structural models with missing data. *BMDP Communications*, **22**, 9–10.

[5] Bentler, P. M. (1995). *EQS structural equations program manual.* Encino, CA: Multivariate software.

[6] Bentler, P. M., & Weeks, D. G. (1980). Linear structural equations with latent variables. *Psychometrika*, **45**, 289–308.

[7] Brown, C. H. (1983). Asymptotic comparison of missing data procedures for estimating factor loadings. *Psychometrika,* **48**, 269–291.

[8] Browne, M. W., & Arminger, G. (1995). Specification and estimation of mean and covariance structure models. In G. Arminger, C. C. Clogg, & M. E. Sobel (Eds.), Handbook of statistical modeling for the social and behavioral sciences (pp. 185–249). New York: Plenum.

[9] Browne, M. W., & Du Toit, S. H. C. (1992). Automated fitting of nonstandard models. *Multivariate Behavioral Research*, **27**, 269–300.

[10] Cudeck, R., Klebe, K. J., & Henly, S. J. (1993). A simple Gauss-Newton procedure for covariance structure analysis with high level computer languages. *Psychometrika,* **44**, 99-113.

[11] Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the EM algorithm (with Discussion). *Journal of the Royal Statistical Society,* **B39**, 1–38.

[12] Dolan, C. V., & Molenaar, P. C. M. (1991). A comparison of four methods of calculating standard errors of maximum likelihood estimates in the analysis of covariance structure. *British Journal of Mathematical and Statistical Psychology,* **44**, 359–368.

[13] Efron, B., & Hinkley, D. V. (1978). The observed versus the expected information. *Biometrika,* **65**, 457–487.

[14] Finkbeiner, C. (1979). Estimation for the multiple factor model when data are missing. *Psychometrika,* **44**, 409–420.

[15] Fisher, R. A. (1925). Theory of statistical estimation. *Proceedings of Cambridge Philosophical Society*, **22**, 700–725.

[16] Holzinger, K. J., & Swineford, F. (1939). A study in factor analysis: The stability of a bi-factor solution. *Supplementary educational monograph*, No. 48. Chicago: University of Chicago.

[17] Jamshidian, M. (1997). An EM algorithm for ML factor analysis with missing data, In *Latent Variable Modeling and Applications to Causality*, (Ed. Berkane, M.), New York: Springer Verlag, 247–258.

[18] Jamshidian, M., & Jennrich, R. I. (1994). Conjugate gradient methods in confirmatory factor analysis. *Computational Statistics and Data Analysis*, **17**, 247–263.

[19] Jamshidian, M., & Jennrich, R. I. (1997a). Acceleration of the EM algorithm by using quasi-Newton methods. *Journal of the Royal Statistical Society*, Series B, **59**, 569–587.

[20] Jamshidian, M., & Jennrich, R. I. (1997b). Standard errors for EM estimation. *Proceedings of the 2nd IASC World Conference, Pasedena*, (to appear).

[21] Jöreskog, K. G., & Sörbom, D. (1988). *LISREL 7, A Guide to the Program and Applications.* SPSS, Chicago.

[22] Khalfan, H. F., Byrd, R. H., & Schnabel, R. B. (1993). A theoretical and experimental study of the symmetric rank-one update. *SIAM Journal on Optimization*, **3**, 1–24.

[23] Lee, S. Y. (1986). Estimation for structural equation models with missing data. *Psychometrika*, **51**, 93–99.

[24] Lee, S. Y., & Jennrich, R. I. (1979). A study of algorithms for covariance structure analysis with specific comparison using factor analysis. *Psychometrika,* **44**, 99–113.

[25] Little, R. J. A., & Rubin, D. B. (1987). *Statistical analysis with missing data.* New York: Wiley.

[26] Luenberger, D. G. (1984). *Linear and nonlinear programming.* Reading, MA: Addison-Wesley.

[27] Muthén, B., Kaplan, D., & Hollis, M. (1987). On structural equation modeling with data that are not missing completely at random. *Psychometrika,* **52**, 431–462.

[28] Rubin, D. B. (1987). *Multiple imputation for nonresponse in surveys.* New York: Wiley.

[29] Rubin, D. B., & Thayer, D. T. (1982). EM algorithms for factor analysis. *Psychometrika*, **47**, 69–76.

[30] Satorra, A. (1989). Alternative test criteria in covariance structure analysis: A unified approach. *Psychometrika*, **54**, 131–151.

[31] Satorra, A. (1992). Asymptotic robust inferences in the analysis of mean and covariance structures. In P. V. Marsden (Ed.) *Sociological methodology* (pp. 249–278). Oxford: Blackwell.

[32] Schepers, A., & Arminger, G. (1992). *MECOSA: Mean and covariance structure analysis*. Frauenfeld, Switzerland: SLI-AG.

[33] Steiger, J. H. (1994). SEPATH - A statistica for windows structural equation modeling program. In F. Faulbaum (Ed.) *Softstat '93: Advances in statistical software 4* (pp. 99–105). Stuttgart: Gustav Fischer Verlag.