Signal Processing 165 (2019) 278–291

Contents lists available at ScienceDirect

Signal Processing

journal homepage: www.elsevier.com/locate/sigpro

Regularized robust estimation of mean and covariance matrix for incomplete data

Junyan Liu*, Daniel P. Palomar

Hong Kong University of Science and Technology, Hong Kong, China

ARTICLE INFO

Article history:
Received 27 December 2018
Revised 26 June 2019
Accepted 8 July 2019
Available online 9 July 2019

Keywords:
Monotone missing-data pattern
Robust estimation
Regularization
Minorization–maximization

ABSTRACT

This paper considers the robust estimation of the mean and covariance matrix for incomplete multivariate observations with the monotone missing-data pattern. First, we develop two efficient numerical algorithms for the existing robust estimator for the monotone incomplete data, i.e., the maximum likelihood (ML) estimator assuming the samples are from a Student's t-distribution. The proposed algorithms can be more than one order of magnitude faster than the existing algorithms. Then, to deal with the unreliability and the inapplicability of the Student's t ML estimator when the number of samples is relatively small compared to the dimension of parameters, we propose a regularized robust estimator, which is defined as the maximizer of a penalized log-likelihood. The penalty term is constructed with a prior target as its global maximizer, towards which the estimator will shrink the mean and covariance matrix. In addition, two numerical algorithms are derived for the regularized estimator. Numerical simulations show the fast convergence rates of the proposed algorithms and the good estimation accuracy of the proposed regularized estimator.

1. Introduction

Estimating the mean and covariance matrix is a fundamental problem in statistical signal processing related fields. A wide range of applications, such as noise attenuation in image processing [1], adaptive beamforming in communications [2], and portfolio construction in finance [3], all depend on the accurate estimation of the mean and covariance matrix. A common approach to estimate these quantities is to use a sample average. However, in many practical applications, the samples may be incomplete, i.e., containing missing values. For example, in astronomical, meteorological, or satellite-based applications, weather or other conditions may disturb sample taking schemes, which will lead to missing data problems [4]. In wireless communications, sensor failure or noise can result in the loss of data [5,6]. In financial markets, missing data occurs when the stocks of interest have various lengths of available historical data. Under such missing data scenarios, the traditional sample average method to estimate the mean and covariance matrix is no longer applicable, and efficient approaches are needed.

One simple way to estimate the mean and covariance matrix for incomplete data is via maximum likelihood (ML) estimation assuming that the samples are independent and identically drawn from a Gaussian distribution [7,8]. Since, for complete data, it is known that the sample mean and sample covariance matrix coincide with the Gaussian maximum likelihood estimates, this extension to incomplete data is natural. However, in practice, the distribution of the data in many applications has a heavier tail than the Gaussian distribution, either due to the intrinsic property of the application, e.g., in financial engineering [9], or the existence of outliers, e.g., samples corrupted by impulsive noise [10]. For these cases, the Gaussian ML estimator will give completely unreliable estimates, and one may seek instead a robust ML estimator assuming that the underlying distribution is some heavy-tailed elliptical distribution, such as the Student's t-distribution. Little proposed in [11] to use the Student's t-distribution with a known degree of freedom ν as the underlying distribution to estimate the mean and covariance matrix for data with missing values. Later, a more general case of the Student's t-distribution with unknown ν was considered in [12,13]. The Student's t ML estimator can provide reliable estimates of mean and covariance matrix even if erroneous observations occur or the samples follow a heavy-tailed distribution. To get the ML estimates, the expectation-maximization (EM) algorithm was employed. Although the EM algorithm is a popular tool, it has been criticized for its slow convergence speed [14,15]. Thus, various extensions of the EM algorithm, e.g., the expectation/conditional maximization either (ECME) algorithm [16,17], and the parameter-expanded EM (PX-EM) approach [15], have been proposed to accelerate its convergence.

* Corresponding author.
E-mail addresses: jliubl@connect.ust.hk (J. Liu), palomar@ust.hk (D.P. Palomar).

https://doi.org/10.1016/j.sigpro.2019.07.009
0165-1684/

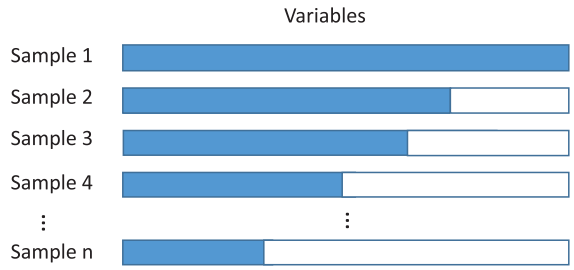

Fig. 1. Incomplete data set with the monotone missing-data pattern. The blue-filled rectangles are the observed values, and the blank rectangles are missing values. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Another issue that often occurs in contemporary applications is the shortage of samples compared to the dimension of the parameters being estimated, like may happen in bioinformatics, financial engineering, and wireless communications [18–21]. When the number of samples is relatively small compared to the dimension of the parameters to be estimated, the information provided by the samples is insufficient for an accurate statistical inference. In addition, it is obvious that the sample covariance will be singular when $n < p$. The above robust estimator has the same drawback. Although there have been significant efforts to develop methodologies for mean and covariance estimation from incomplete data in the small sample size regime [22–24], they are all based on the traditional Gaussian distribution assumptions, and less attention has been placed on robust estimation.

This paper focuses on the robust estimation of mean and covariance matrix for incomplete data with the monotone missing-data pattern (see Fig. 1 for an illustration of the pattern). According to which values are observed, and which values are missing, the incomplete data sets can be classified into different missing-data patterns. The monotone missing-data pattern is a kind of pattern that appears frequently in real-world applications, where observations are missing after some point possibly different for each variable. For example, in longitudinal studies, the dropout of some subjects before the end of a study will lead to the monotone missing-data pattern [13], while in wireless communications, the failure of some of the sensors before measurements are completed can also result in this kind of incomplete data [6]. In financial markets, assets become part of the market at different times through initial public offerings (IPOs), which leads to a reversed version of this pattern as well [25,26]. Our main contributions in this paper are:

1. We investigate the existing robust estimator of mean and covariance matrix for monotone incomplete data sets, and develop two more efficient algorithms, which can be more than one order of magnitude faster than the existing PX-ECME algorithm in [13], based on the general minorization-maximization (MM) framework.
2. In order to deal with the "large p small n" situation, for which the existing robust estimator will provide unreliable estimates or may even not exist, we propose a regularized robust estimator of mean and covariance matrix, defined as the maximizer of a penalized likelihood function. The proposed estimator shrinks the mean and covariance matrix towards a prior target, and can provide reliable estimates even when the sample size is small relative to the problem dimension.
3. We design two efficient algorithms for this regularized robust estimator. Simulations on both synthetic data and real data show that the proposed regularized estimator can provide more accurate estimates than the existing robust estimators in the small sample size regime.

The remainder of this paper is organized as follows. Section 2 explains the concept of the monotone missing-data pattern and reviews the existing robust estimator. Section 5.1 introduces the proposed regularized robust estimator and gives the problem formulation. Section 4 is devoted to the introduction of the MM framework and the derivation of the proposed algorithms for the existing robust estimator. Section 5 develops two algorithms for proposed regularized robust estimator. In Section 6, simulations results are provided. Section 7 concludes the paper.

Notation: $\mathbb{R}^p$ stands for the p-dimensional real-valued vector space. $\mathbb{S}^p_{++}$ stands for the set of symmetric positive definite $p \times p$ matrices, which is a closed cone in $\mathbb{R}^p$. The superscripts $(\cdot)^{-1}$ and $(\cdot)^T$ denote the matrix inverse and transpose operator, respectively. $\mathsf{E}(\cdot)$ and $\mathsf{Cov}(\cdot)$ denote the mathematical expectation and covariance operator, respectively. $\det(\cdot)$ and $\mathsf{Tr}(\cdot)$ denote the matrix determinant and trace, respectively. $\mathbf{\Sigma} \succ \mathbf{0}$ means that $\mathbf{\Sigma}$ is a symmetric positive definite matrix.

2. Robust estimation for monotone incomplete data

In this section, we introduce the monotone missing-data pattern, and review the existing robust estimation approach for mean and covariance from monotone incomplete data.

2.1. Monotone missing – data pattern

Suppose we collect n independent samples of a p-dimensional random vector $\mathbf{y}$ in an $n \times p$ matrix $\mathbf{Y} = \{\mathbf{y}_i = (y_{i,1}, y_{i,2}, \ldots, y_{i,p}); i = 1, 2, \ldots, n\}$. We say the data set follows the monotone missing-data pattern when the incomplete samples can be sorted into K different groups such that, in the first group containing samples $i = 1, \ldots, n_1$, only the first p_1 components are observed; more generally, in the kth group containing samples $i = n_{k-1} + 1, \ldots, n_k$, only the first p_k components are observed [13]:

$$
\begin{aligned}
& y^{(1)}_{i,1}, y^{(1)}_{i,2}, \ldots, y^{(1)}_{i,p_k}, \ldots, y^{(1)}_{i,p_1} && \text{for } i = 1, \ldots, n_1; \\
& \ldots \\
& y^{(k)}_{i,1}, y^{(k)}_{i,2}, \ldots, y^{(k)}_{i,p_k} && \text{for } i = n_{k-1} + 1, \ldots, n_k; \\
& \ldots \\
& y^{(K)}_{i,1}, \ldots, y^{(K)}_{i,p_K} && \text{for } i = n_{K-1} + 1, \ldots, n_K,
\end{aligned}
$$

where $p = p_1 > \cdots > p_K$ and $0 = n_0 < n_1 < \cdots < n_K = n$. The superscript represents the group that incomplete sample belongs to. We denote by $\mathbf{y}_{i,\text{obs}}$ and $\mathbf{y}_{i,\text{mis}}$ the observed values and missing values of the sample $\mathbf{y}_i$ such that $\mathbf{y}_i = (\mathbf{y}_{i,\text{obs}}, \mathbf{y}_{i,\text{mis}})$. The missing-data mechanism is assumed to be ignorable, i.e., the missingness does not depend on values of the missing data [13]. The examples given in the introduction can all be considered to have this missing-data mechanism. Given the incomplete monotone data set $\mathbf{Y}$, we are interested in the robust estimation of mean and covariance matrix for the random variable $\mathbf{y}$. The estimation accuracy increases with the increase of number of observed values $(\sum_{k=1}^{K}(n_k - n_{k-1})p_k)$.

2.2. Robust estimation of mean and covariance matrix

The Student's t distribution based ML estimator is a widely used robust estimator for mean and covariance matrix [12,27]. Suppose the random variable $\mathbf{y}$ follows a Student's t-distribution: $\mathbf{y} \sim t_p(\boldsymbol{\mu}, \mathbf{\Sigma}, \nu)$, where $\boldsymbol{\mu} \in \mathbb{R}^p$ is the location parameter, $\mathbf{\Sigma} \in \mathbb{S}^p_{++}$ is the shape matrix, and $\nu > 0$ is the number of degrees of freedom.

Then, for $\nu > 2$, the mean and covariance matrix of $\mathbf{y}$ are $\boldsymbol{\mu}$ and $\mathbf{R} = \frac{\nu}{\nu-2}\boldsymbol{\Sigma}$, respectively[1].

When there are no missing values, i.e., we have a complete data set $\{\mathbf{y}_i\}$, the ML estimation problem for $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$, and ν, is

$$\underset{\boldsymbol{\mu},\boldsymbol{\Sigma}\succ\mathbf{0},\nu\geq\nu^-}{\text{maximize}} \; l(\{\mathbf{y}_i\}|\boldsymbol{\mu},\boldsymbol{\Sigma},\nu) \tag{1}$$

where

$$\begin{aligned} l(\{\mathbf{y}_i\}|\boldsymbol{\mu},\boldsymbol{\Sigma},\nu) = &-\frac{\nu+p}{2}\sum_{i=1}^{n}\log\left(\nu+(\mathbf{y}_i-\boldsymbol{\mu})^T\boldsymbol{\Sigma}^{-1}(\mathbf{y}_i-\boldsymbol{\mu})\right) \\ &-\frac{n}{2}\log\left(\det\left(\boldsymbol{\Sigma}\right)\right)+\frac{n\nu}{2}\log\left(\nu\right) \\ &+n\log\left(\Gamma\left(\frac{\nu+p}{2}\right)\right) \\ &-n\log\left(\Gamma\left(\frac{\nu}{2}\right)\right)-\frac{np}{2}\log\left(\pi\right). \end{aligned} \tag{2}$$

with $\Gamma(\cdot)$ being the gamma function. After obtaining the ML estimates $\hat{\boldsymbol{\mu}}$, $\hat{\boldsymbol{\Sigma}}$ and $\hat{\nu}$, the estimates for the mean and covariance matrix are $\hat{\boldsymbol{\mu}}$ and $\hat{\mathbf{R}} = \frac{\hat{\nu}}{\hat{\nu}-2}\hat{\boldsymbol{\Sigma}}$, respectively.

Setting the gradient of $l(\{\mathbf{y}_i\}|\boldsymbol{\mu},\boldsymbol{\Sigma},\nu)$ to zero, we get

$$\frac{\hat{\nu}+p}{n}\sum_{i=1}^{n}\frac{\mathbf{y}_i-\hat{\boldsymbol{\mu}}}{\hat{\nu}+\left(\mathbf{y}_i-\hat{\boldsymbol{\mu}}\right)^T\hat{\boldsymbol{\Sigma}}^{-1}\left(\mathbf{y}_i-\hat{\boldsymbol{\mu}}\right)}=0, \tag{3}$$

$$\frac{\hat{\nu}+p}{n}\sum_{i=1}^{n}\frac{\left(\mathbf{y}_i-\hat{\boldsymbol{\mu}}\right)\left(\mathbf{y}_i-\hat{\boldsymbol{\mu}}\right)^T}{\hat{\nu}+\left(\mathbf{y}_i-\hat{\boldsymbol{\mu}}\right)^T\hat{\boldsymbol{\Sigma}}^{-1}\left(\mathbf{y}_i-\hat{\boldsymbol{\mu}}\right)}=\hat{\boldsymbol{\Sigma}}. \tag{4}$$

The estimates $\hat{\boldsymbol{\mu}}$ and $\hat{\boldsymbol{\Sigma}}$ can be interpreted as the weighted sample averages. The weights are inversely proportional to $(\mathbf{y}_i-\hat{\boldsymbol{\mu}})^T\hat{\boldsymbol{\Sigma}}^{-1}(\mathbf{y}_i-\hat{\boldsymbol{\mu}})$, which means they decrease as the samples get far away from the center. This property indicates this estimator is more robust to outliers compared with the (unweighted) sample average, which is equivalent to the ML estimate assuming a Gaussian underlying distribution. The Student's t ML estimator in (3) and (4) belongs to the class of M-estimators, whose robustness properties have been studied in [27].

When the data set is incomplete and follows the monotone missing-data pattern, the ML estimation problem is much more complicated. Let us denote by $\mathbf{y}_{(p_k)}$ the vector of the first p_k components of $\mathbf{y}$, and by $\boldsymbol{\mu}_{(p_k)}$ the vector of the first p_k components of $\boldsymbol{\mu}$, and by $\boldsymbol{\Sigma}_{(p_k)}$ the upper-left $p_k \times p_k$ submatrix of $\boldsymbol{\Sigma}$. The marginal distribution of $\mathbf{y}_{(p_k)}$ is

$$\mathbf{y}_{(p_k)} \sim t_{p_k}\left(\boldsymbol{\mu}_{(p_k)},\boldsymbol{\Sigma}_{(p_k)},\nu\right). \tag{5}$$

Therefore, given the monotone data set $\mathbf{Y}$, the ML estimation problem for $\boldsymbol{\mu}$, $\boldsymbol{\Sigma}$ and ν can be formulated as follows:

$$\underset{\boldsymbol{\mu},\boldsymbol{\Sigma}\succ\mathbf{0},\nu\geq\nu^-}{\text{maximize}} \; l(\mathbf{Y}|\boldsymbol{\mu},\boldsymbol{\Sigma},\nu)=\sum_{k=1}^{K}h_k(\boldsymbol{\mu},\boldsymbol{\Sigma},\nu), \tag{6}$$

where

$$\begin{aligned} h_k(\boldsymbol{\mu},\boldsymbol{\Sigma},\nu) &= \sum_{i=n_{k-1}+1}^{n_k}\log p\left(\mathbf{y}_{i,(p_k)}|\boldsymbol{\mu}_{(p_k)},\boldsymbol{\Sigma}_{(p_k)},\nu\right) \\ &= -\frac{\nu+p_k}{2}\sum_{i=n_{k-1}+1}^{n_k}\log\Big(\nu+\left(\mathbf{y}_{i,(p_k)}-\boldsymbol{\mu}_{(p_k)}\right)^T \\ &\quad\times\left(\boldsymbol{\Sigma}_{(p_k)}\right)^{-1}\left(\mathbf{y}_{i,(p_k)}-\boldsymbol{\mu}_{(p_k)}\right)\Big) \\ &\quad-\frac{n_k-n_{k-1}}{2}\log\left(\det\left(\boldsymbol{\Sigma}_{(p_k)}\right)\right) \\ &\quad+(n_k-n_{k-1})\left\{\frac{\nu}{2}\log\left(\nu\right)+\log\left(\Gamma\left(\frac{\nu+p_k}{2}\right)\right)\right. \\ &\quad\left.-\log\left(\Gamma\left(\frac{\nu}{2}\right)\right)-\frac{p_k}{2}\log\left(\pi\right)\right\} \end{aligned} \tag{7}$$

with $\mathbf{y}_{i,(p_k)}$ denoting the first p_k components of $\mathbf{y}_i$.

2.3. Existing algorithm

The above ML estimation problem (6) is too complicated to be solved directly. Interestingly, the Student's t-distribution can be written as a mixture of Gaussian distributions:

$$\text{Model O1}: \quad \begin{aligned} &\mathbf{y}_i|\boldsymbol{\mu},\boldsymbol{\Sigma},\tau_i \sim \mathcal{N}_p(\boldsymbol{\mu},\boldsymbol{\Sigma}/\tau_i), \\ &\tau_i \sim Gamma(\nu/2,\nu/2). \end{aligned} \tag{8}$$

Therefore, in [17], the authors proposed to regard both $\{\tau_i\}$ and $\{\mathbf{y}_{i,mis}\}$ as unobserved latent data, and use the ECME algorithm to solve the robust estimation problem iteratively. The ECME algorithm is a variant of the EM algorithm. At iteration $t+1$, it updates the parameters according to

$$\left(\boldsymbol{\mu}^{(t+1)},\boldsymbol{\Sigma}^{(t+1)}\right)=\underset{(\boldsymbol{\mu},\boldsymbol{\Sigma})}{\arg\max}\,Q\left(\boldsymbol{\mu},\boldsymbol{\Sigma},\nu^{(t)}|\boldsymbol{\mu}^{(t)},\boldsymbol{\Sigma}^{(t)},\nu^{(t)}\right), \tag{9}$$

and

$$\nu^{(t+1)}=\underset{\nu\geq\nu^-}{\arg\max}\,l\left(\mathbf{Y}|\boldsymbol{\mu}^{(t+1)},\boldsymbol{\Sigma}^{(t+1)},\nu\right), \tag{10}$$

where $Q(\boldsymbol{\mu},\boldsymbol{\Sigma},\nu^{(t)}|\boldsymbol{\mu}^{(t)},\boldsymbol{\Sigma}^{(t)},\nu^{(t)})$ is the expectation of the complete data log-likelihood with respect to the posterior distribution $p(\{\tau_i\},\{\mathbf{y}_{i,\text{mis}}\}|\boldsymbol{\mu}^{(t)},\boldsymbol{\Sigma}^{(t)},\nu^{(t)},\{\mathbf{y}_{i,\text{obs}}\})$:

$$\begin{aligned} Q\left(\boldsymbol{\mu},\boldsymbol{\Sigma},\nu^{(t)}|\boldsymbol{\mu}^{(t)},\boldsymbol{\Sigma}^{(t)},\nu^{(t)}\right) &= \mathsf{E}_{p\left(\{\tau_i\},\{\mathbf{y}_{i,\text{mis}}\}|\boldsymbol{\mu}^{(t)},\boldsymbol{\Sigma}^{(t)},\nu^{(t)},\{\mathbf{y}_{i,\text{obs}}\}\right)}\left(l\left(\{\mathbf{y}_{i,\text{obs}}\},\{\mathbf{y}_{i,\text{mis}}\},\{\tau_i\}|\boldsymbol{\mu},\boldsymbol{\Sigma},\nu^{(t)}\right)\right) \\ &= \int l\left(\{\mathbf{y}_{i,\text{obs}}\},\{\mathbf{y}_{i,\text{mis}}\},\{\tau_i\}|\boldsymbol{\mu},\boldsymbol{\Sigma},\nu^{(t)}\right)p\left(\{\tau_i\},\{\mathbf{y}_{i,\text{mis}}\}|\boldsymbol{\mu}^{(t)},\boldsymbol{\Sigma}^{(t)},\nu^{(t)},\{\mathbf{y}_{i,\text{obs}}\}\right)\mathrm{d}\tau_1\ldots\mathrm{d}\tau_n\mathrm{d}\mathbf{y}_{1,\text{mis}}\ldots\mathrm{d}\mathbf{y}_{n,\text{mis}} \\ &= \sum_{i=1}^{n}\left\{-\frac{1}{2}\mathrm{Tr}\left(\mathsf{E}\left(\tau_i\mathbf{y}_i\mathbf{y}_i^T\right)\boldsymbol{\Sigma}^{-1}\right)+\boldsymbol{\mu}^T\boldsymbol{\Sigma}^{-1}\mathsf{E}(\tau_i\mathbf{y}_i)-\frac{1}{2}\boldsymbol{\mu}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}\mathsf{E}(\tau_i)\right\}-\frac{n}{2}\log\left(\det\left(\boldsymbol{\Sigma}\right)\right)+const. \end{aligned} \tag{11}$$

The expectation operations in (11) are all with respect to the posterior distribution $p(\{\tau_i\},\{\mathbf{y}_{i,\text{mis}}\}|\boldsymbol{\mu}^{(t)},\boldsymbol{\Sigma}^{(t)},\nu^{(t)},\{\mathbf{y}_{i,\text{obs}}\})$. For simplicity of notations, we omit the subscripts of these expectations. The update $(\boldsymbol{\mu}^{(t+1)},\boldsymbol{\Sigma}^{(t+1)})$ can be expressed as

$$\boldsymbol{\mu}^{(t+1)}=\frac{\sum_{i=1}^{n}\mathsf{E}(\tau_i\mathbf{y}_i)}{\sum_{i=1}^{n}\mathsf{E}(\tau_i)} \tag{12}$$

and

$$\begin{aligned} \boldsymbol{\Sigma}^{(t+1)} = \frac{1}{n}\sum_{i=1}^{n}\Big\{&\mathsf{E}\left(\tau_i\mathbf{y}_i\mathbf{y}_i^T\right)-2\mathsf{E}(\tau_i\mathbf{y}_i)\left(\boldsymbol{\mu}^{(t+1)}\right)^T \\ &+\mathsf{E}(\tau_i)\boldsymbol{\mu}^{(t+1)}\left(\boldsymbol{\mu}^{(t+1)}\right)^T\Big\}. \end{aligned} \tag{13}$$

Since the convergence speed of the EM type algorithms is typically very slow, the authors of [15] introduced the parameter

[1] In practice, when we use Student's t ML estimator to estimate mean and covariance, we do not want ν to be close to 2, since the covariance will blow up. Thus, in this paper we let $\nu \geq \nu^-$, where $\nu^- = 2.01$.

expansion-EM (PX-EM) method to accelerate EM-based algorithms, which leads to the benchmark, the PX-ECME algorithm in [13]. Interested reader may refer to Sections 8.5.3 and 12.3 of [13] for details.

3. Regularized robust estimation for monotone incomplete data

When the number of samples is sufficient, the above robust estimator can provide accurate estimates. However, in some practical applications, the number of samples is relatively small compared to the number of the parameters. In such cases, the above robust estimator will give unreliable estimates, and the algorithm designed for the estimator may even fail to converge. Motivated by the idea in [20,21], we propose to regularize the above estimator by shrinking the estimator to a prior target $(\mathbf{t}, \mathbf{T})$. This method not only provides a way to incorporate some prior information into the estimator, but also helps stabilize the estimator in small sample size regime [21].

The proposed regularized robust estimation problem can be expressed as follows:

$$\underset{\boldsymbol{\mu},\, \boldsymbol{\Sigma}\succ\mathbf{0},\, \nu\geq\nu^-}{\text{maximize}} \; l(\mathbf{Y}|\boldsymbol{\mu},\, \boldsymbol{\Sigma},\, \nu) - \alpha u(\boldsymbol{\mu},\, \boldsymbol{\Sigma},\, \nu), \tag{14}$$

where α is a nonnegative parameter, and $u(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu)$ is a penalty function, which increases when the mean and covariance $\boldsymbol{\mu}$ and $\frac{\nu}{\nu-2}\boldsymbol{\Sigma}$ deviate from the prior mean and covariance $(\mathbf{t}, \mathbf{T})$. Let us denote the objective function by $l^{\text{shrink}}(\mathbf{Y}|\boldsymbol{\mu},\, \boldsymbol{\Sigma},\, \nu)$. Here we adopt a widely used penalty, the Kullback-Leibler (KL) divergence between the normal distributions $\mathcal{N}(\mathbf{t}, \mathbf{T})$ and $\mathcal{N}(\boldsymbol{\mu}, \frac{\nu}{\nu-2}\boldsymbol{\Sigma})$[21,28]:

$$\begin{aligned} u(\boldsymbol{\mu},\, \boldsymbol{\Sigma},\, \nu) &= D_{KL}\left(\mathcal{N}(\mathbf{t}, \mathbf{T})||\mathcal{N}\left(\boldsymbol{\mu}, \frac{\nu}{\nu-2}\boldsymbol{\Sigma}\right)\right) \\ &= \frac{1}{2}(\mathbf{t}-\boldsymbol{\mu})^T\left(\frac{\nu}{\nu-2}\boldsymbol{\Sigma}\right)^{-1}(\mathbf{t}-\boldsymbol{\mu}) \\ &\quad + \frac{1}{2}\mathrm{Tr}\left(\left(\frac{\nu}{\nu-2}\boldsymbol{\Sigma}\right)^{-1}\mathbf{T}\right) \\ &\quad - \frac{p}{2} - \frac{1}{2}\log\det(\mathbf{T}) + \frac{1}{2}\log\det\left(\frac{\nu}{\nu-2}\boldsymbol{\Sigma}\right). \end{aligned} \tag{15}$$

The following proposition indicates that $u(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu)$ is a proper penalty function.

Proposition 1. *For any $\boldsymbol{\mu} \in \mathbb{R}^p$, $\boldsymbol{\Sigma} \in \mathbb{S}^p_{++}$, and $\nu \geq \nu^-$, it follows that $u(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu) \geq 0$ with the equality achieved if and only if $\boldsymbol{\mu} = \mathbf{t}$ and $\frac{\nu}{\nu-2}\boldsymbol{\Sigma} = \mathbf{T}$.*

Proof. The KL divergence is always nonnegative, and is equal to zero if and only if the two distributions are identical [29]. □

4. Algorithms for robust estimation

In this section, we develop two algorithms faster than the benchmark PX-ECME algorithm for the robust estimation problem (6) based on the minorization-maximization (MM) framework. Let us first give a brief introduction to the MM framework.

4.1. Minorization–Maximization framework

Consider the following optimization problem:

$$\begin{aligned} &\underset{\mathbf{x}}{\text{maximize}} \; f(\mathbf{x}) \\ &\text{subject to } \mathbf{x} \in \mathcal{X}, \end{aligned} \tag{16}$$

where $\mathcal{X}$ is a closed convex set in $\mathbb{R}^p$, and f: $\mathcal{X} \to \mathbb{R}$ is a continuous function. When the problem is too complicated to be solved directly, the MM framework circumvents such a difficulty by solving a sequence of simpler optimization problems [30,31].

At iteration $t+1$, the MM framework first find a minorizing function $g(\mathbf{x}|\mathbf{x}^{(t)})$ for $f(\mathbf{x})$, which satisfies the following conditions:

$$\begin{aligned} &g\left(\mathbf{x}^{(t)}|\mathbf{x}^{(t)}\right) = f\left(\mathbf{x}^{(t)}\right), \\ &g\left(\mathbf{x}|\mathbf{x}^{(t)}\right) \leq f(\mathbf{x}),\ \forall \mathbf{x} \in \mathcal{X}, \\ &g'\left(\mathbf{x}^{(t)};\, \mathbf{d}|\mathbf{x}^{(t)}\right) = f'\left(\mathbf{x}^{(t)};\, \mathbf{d}\right),\ \forall \mathbf{x}^{(t)} + \mathbf{d} \in \mathcal{X}. \end{aligned} \tag{17}$$

with $f'(\mathbf{x}^{(t)};\, \mathbf{d})$ and $g'(\mathbf{x}^{(t)};\, \mathbf{d}|\mathbf{x}^{(t)})$ standing for the directional derivative. In other words, the minorizing function $g(\mathbf{x}|\mathbf{x}^{(t)})$ is a global lower bound for $f(\mathbf{x})$ and coincides with $f(\mathbf{x})$ at $\mathbf{x}^{(t)}$. Then the MM framework updates $\mathbf{x}$ as

$$\mathbf{x}^{(t+1)} = \underset{\mathbf{x}\in\mathcal{X}}{\arg\max}\; g\left(\mathbf{x}|\mathbf{x}^{(t)}\right). \tag{18}$$

We can easily see $f(\mathbf{x}^{(t+1)}) \geq f(\mathbf{x}^{(t)})$. It is proved in [31] that any limit point of the sequence $\{\mathbf{x}^{(t)}\}$ is a stationary point of the original problem (16).

The idea of minorization and maximization can also be applied blockwise, i.e., we can divide the variables into different blocks, and conduct the minorization and maximization for each block of variables, with other blocks of variables fixed in every iteration.

The key to the success of MM lies in constructing the minorizing function. On the one hand, to achieve a fast convergence speed, a minorizing function that follows the shape of the objective function is desirable. On the other hand, it should be simple to maximize so that the computational cost per iteration is low. It is not easy to find a good trade-off between these two opposite targets [21].

Interestingly, the EM algorithm is actually a special case of the MM, and the ECME algorithm is a special case of the block MM [30]. More specifically, denote the observed data by $\mathbf{X}$, the unobserved latent variable by $\mathbf{Z}$, the parameter by $\boldsymbol{\theta}$, and the expectation of the complete data log-likelihood by $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$. The minorizing function for the observed data log-likelihood $l(\mathbf{X}|\boldsymbol{\theta})$ in the EM algorithm is

$$g\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}\right) = Q\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}\right) - Q\left(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}\right) + l\left(\mathbf{X}|\boldsymbol{\theta}^{(t)}\right), \tag{19}$$

which is the sum of $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ and a constant. In addition, the following proposition indicates that, when applying the EM algorithm, if we can consider fewer variables as latent data, the resulting minorizing function will be a tighter approximation of the original log-likelihood.

Proposition 2. *For any two different unobserved latent variables $\mathbf{Z}_1$ and $\mathbf{Z}_2$, the minorizing functions satisfy*

$$g_{\mathbf{Z}_1}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}\right) \geq g_{\mathbf{Z}_1\mathbf{Z}_2}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}\right), \tag{20}$$

where $g_{\mathbf{Z}_1}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ is the resulting minorizing function with $\mathbf{Z}_1$ considered as latent variable, and $g_{\mathbf{Z}_1\mathbf{Z}_2}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ is the resulting minorizing function with both $\mathbf{Z}_1$ and $\mathbf{Z}_2$ considered as latent variables.

Proof. See Appendix A.1. □

4.2. Robust estimation via block MM

The objective function of the robust estimation problem (6) is very complicated. It is difficult to apply MM for all variables jointly. Therefore, we partition the three variables into two blocks: $(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ as one block, and ν as another block. Then we apply block MM to solve this problem iteratively. At iteration $t+1$, we first conduct the minorization and maximization for $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, with ν fixed as $\nu^{(t)}$, and then optimize ν, with $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ fixed as $\boldsymbol{\mu}^{(t+1)}$ and $\boldsymbol{\Sigma}^{(t+1)}$.

The optimization of ν is easy to solve. If we fix the values of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ as $\boldsymbol{\mu}^{(t+1)}$ and $\boldsymbol{\Sigma}^{(t+1)}$, the objective function is

$$\begin{aligned} l\left(\mathbf{Y}|\boldsymbol{\mu}^{(t+1)},\ \boldsymbol{\Sigma}^{(t+1)},\ \nu\right) &= \sum_{k=1}^{K}\sum_{i=n_{k-1}+1}^{n_k} -\frac{\nu+p_k}{2}\log\left(\nu+\delta_i^{(t+1)}\right) \\ &+ \sum_{k=1}^{K}(n_k-n_{k-1})\log\left(\Gamma\left(\frac{\nu+p_k}{2}\right)\right) \\ &+ \frac{n\nu}{2}\log(\nu) - n\log\left(\Gamma\left(\frac{\nu}{2}\right)\right) + const., \end{aligned} \tag{21}$$

where

$$\delta_i^{(t+1)} = \left(\mathbf{y}_{i,(p_k)} - \boldsymbol{\mu}_{(p_k)}^{(t+1)}\right)^T \left(\boldsymbol{\Sigma}_{(p_k)}^{(t+1)}\right)^{-1} \left(\mathbf{y}_{i,(p_k)} - \boldsymbol{\mu}_{(p_k)}^{(t+1)}\right) \tag{22}$$

with $k = \arg_k(n_{k-1} < i \le n_k)$. It is a function of a scalar variable, and the maximizer $\nu^{(t+1)}$ can be found by a one-dimensional search.

In the following, we concentrate on the minorization and maximization for $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, which will take some effort. With the ν fixed as $\nu^{(t)}$, the objective function is:

$$\begin{aligned} l\left(\mathbf{Y}|\boldsymbol{\mu},\ \boldsymbol{\Sigma},\ \nu^{(t)}\right) = \sum_{k=1}^{K}\Bigg\{ &-\frac{\nu^{(t)}+p_k}{2}\sum_{i=n_{k-1}+1}^{n_k}\log\Bigg(\nu^{(t)} + \\ &\left(\mathbf{y}_{i,(p_k)} - \boldsymbol{\mu}_{(p_k)}\right)^T \left(\boldsymbol{\Sigma}_{(p_k)}\right)^{-1}\left(\mathbf{y}_{i,(p_k)} - \boldsymbol{\mu}_{(p_k)}\right)\Bigg) \\ &- \frac{n_k - n_{k-1}}{2}\log\det\left(\boldsymbol{\Sigma}_{(p_k)}\right)\Bigg\} + const. \end{aligned} \tag{23}$$

By the concavity of the $\log(\cdot)$, we have

$$\log(x) \le \frac{1}{x_0}(x - x_0) + \log(x_0). \tag{24}$$

Therefore, at point $(\boldsymbol{\mu}^{(t)},\ \boldsymbol{\Sigma}^{(t)},\ \nu^{(t)})$, we have

$$\begin{aligned} &\log\left(\nu^{(t)} + \left(\mathbf{y}_{i,(p_k)} - \boldsymbol{\mu}_{(p_k)}\right)^T \left(\boldsymbol{\Sigma}_{(p_k)}\right)^{-1}\left(\mathbf{y}_{i,(p_k)} - \boldsymbol{\mu}_{(p_k)}\right)\right) \\ &\le \frac{\nu^{(t)} + \left(\mathbf{y}_{i,(p_k)} - \boldsymbol{\mu}_{(p_k)}\right)^T \left(\boldsymbol{\Sigma}_{(p_k)}\right)^{-1}\left(\mathbf{y}_{i,(p_k)} - \boldsymbol{\mu}_{(p_k)}\right)}{\nu^{(t)} + \left(\mathbf{y}_{i,(p_k)} - \boldsymbol{\mu}_{(p_k)}^{(t)}\right)^T \left(\boldsymbol{\Sigma}_{(p_k)}^{(t)}\right)^{-1}\left(\mathbf{y}_{i,(p_k)} - \boldsymbol{\mu}_{(p_k)}^{(t)}\right)} + const., \end{aligned} \tag{25}$$

and then $l(\mathbf{Y}|\boldsymbol{\mu},\ \boldsymbol{\Sigma},\ \nu^{(t)})$ is minorized by

$$\begin{aligned} g\left(\boldsymbol{\mu},\ \boldsymbol{\Sigma}|\boldsymbol{\mu}^{(t)},\ \boldsymbol{\Sigma}^{(t)},\ \nu^{(t)}\right) = \sum_{k=1}^{K}\Bigg\{ &\sum_{i=n_{k-1}+1}^{n_k} -\frac{1}{2}\omega_i^{(t)}\left(\mathbf{y}_{i,(p_k)} - \boldsymbol{\mu}_{(p_k)}\right)^T \\ &\left(\boldsymbol{\Sigma}_{(p_k)}\right)^{-1}\left(\mathbf{y}_{i,(p_k)} - \boldsymbol{\mu}_{(p_k)}\right) - \frac{n_k - n_{k-1}}{2}\log\det\left(\boldsymbol{\Sigma}_{(p_k)}\right)\Bigg\} + const., \end{aligned} \tag{26}$$

where

$$\omega_i^{(t)} = \frac{\nu^{(t)} + p_k}{\nu^{(t)} + \left(\mathbf{y}_{i,(p_k)} - \boldsymbol{\mu}_{(p_k)}^{(t)}\right)^T \left(\boldsymbol{\Sigma}_{(p_k)}^{(t)}\right)^{-1}\left(\mathbf{y}_{i,(p_k)} - \boldsymbol{\mu}_{(p_k)}^{(t)}\right)} \tag{27}$$

with $k = \arg_k(n_{k-1} < i \le n_k)$.

The minorizing function (26) is too complicated to be maximized directly. But after applying the following Lemma 1 [32], we successfully reparameterize $g(\boldsymbol{\mu},\ \boldsymbol{\Sigma}|\boldsymbol{\mu}^{(t)},\ \boldsymbol{\Sigma}^{(t)},\ \nu^{(t)})$ into a simpler form in Proposition 3, whose maximizer is easier to obtain.

Lemma 1. *Given the positive definite matrix $\boldsymbol{\Sigma}$ and the upper triangular Cholesky decomposition of its inverse*

$$\boldsymbol{\Sigma}^{-1} = \mathbf{H}\mathbf{H}^T, \tag{28}$$

where $\mathbf{H}$ is an upper triangular matrix with positive diagonal entries, the upper triangular Cholesky decomposition for $(\boldsymbol{\Sigma}_{(j)})^{-1}$ is

$$\left(\boldsymbol{\Sigma}_{(j)}\right)^{-1} = \mathbf{H}_{(j)}\left(\mathbf{H}_{(j)}\right)^T, \tag{29}$$

where $\boldsymbol{\Sigma}_{(j)}$ is the upper left $j\times j$ submatrix of $\boldsymbol{\Sigma}$, and $\mathbf{H}_{(j)}$ is the upper left $j\times j$ submatrix of $\mathbf{H}$.

Proposition 3. *If we replace $\boldsymbol{\Sigma}$ in (26) with $(\mathbf{H}\mathbf{H}^T)^{-1}$, the function $g\left(\boldsymbol{\mu},\ \boldsymbol{\Sigma}|\boldsymbol{\mu}^{(t)},\boldsymbol{\Sigma}^{(t)},\ \nu^{(t)}\right)$ can be reparameterized as*

$$\begin{aligned} &g\left(\boldsymbol{\mu},\ \mathbf{H}|\boldsymbol{\mu}^{(t)},\ \boldsymbol{\Sigma}^{(t)},\ \nu^{(t)}\right) \\ &= -\frac{1}{2}\left(\boldsymbol{\mu} - \boldsymbol{\theta}^{(t)}(\mathbf{H})\right)^T \mathbf{B}^{(t)}(\mathbf{H})\left(\boldsymbol{\mu} - \boldsymbol{\theta}^{(t)}(\mathbf{H})\right) \\ &- \frac{1}{2}\sum_{j=1}^{p}\mathbf{h}_j^T\mathbf{S}_j^{(t)}\mathbf{h}_j + \sum_{j=1}^{p}\log\left(h_{j,j}\right)n_{k(j)} + const., \end{aligned} \tag{30}$$

where, for $j = 1, 2, \ldots, p$, $\mathbf{h}_j$ is the vector of the first j components of the jth column of $\mathbf{H}$,

$$k(j) = \arg_k(p_{k-1} < j \le p_k), \tag{31}$$

the weighted covariance of $\mathbf{y}_{i,(j)}$ (the first j components) in the first $k(j)$ groups

$$\mathbf{S}_j^{(t)} = \sum_{i=1}^{n_{k(j)}}\omega_i^{(t)}\left(\mathbf{y}_{i,(j)} - \bar{\mathbf{y}}_j^{(t)}\right)\left(\mathbf{y}_{i,(j)} - \bar{\mathbf{y}}_j^{(t)}\right)^T, \tag{32}$$

$$\boldsymbol{\theta}^{(t)}(\mathbf{H}) = \mathbf{H}^{-T}\left(\mathbf{h}_1^T\bar{\mathbf{y}}_1^{(t)},\ \mathbf{h}_2^T\bar{\mathbf{y}}_2^{(t)},\ \ldots,\ \mathbf{h}_p^T\bar{\mathbf{y}}_p^{(t)}\right)^T, \tag{33}$$

and

$$\mathbf{B}^{(t)}(\mathbf{H}) = \mathbf{H}\mathrm{Diag}(\Omega_1^{(t)},\ \Omega_2^{(t)},\ \ldots,\ \Omega_p^{(t)})\mathbf{H}^T, \tag{34}$$

with *the weighted sample mean*

$$\bar{\mathbf{y}}_j^{(t)} = \frac{\sum_{i=1}^{n_{k(j)}}\omega_i^{(t)}\mathbf{y}_{i,(j)}}{\sum_{i=1}^{n_{k(j)}}\omega_i^{(t)}} \tag{35}$$

and

$$\Omega_j^{(t)} = \sum_{i=1}^{n_{k(j)}}\omega_i^{(t)}. \tag{36}$$

Proof. See Appendix A.2. □

The reparameterized function $g(\boldsymbol{\mu},\ \mathbf{H}|\boldsymbol{\mu}^{(t)},\ \boldsymbol{\Sigma}^{(t)},\ \nu^{(t)})$ still looks nontrivial. However, by applying some tricks, we obtain its closed-form maximizer, and thus, achieve the closed-form maximizer for $g(\boldsymbol{\mu},\ \boldsymbol{\Sigma}|\boldsymbol{\mu}^{(t)},\ \boldsymbol{\Sigma}^{(t)},\ \nu^{(t)})$ successfully. The closed-form maximizer is given next in Proposition 4.

Proposition 4. *The minorizing function (26) is maximized by*

$$\boldsymbol{\mu}^{(t+1)} = \left(\mathbf{H}^{(t+1)}\right)^{-T}\left[\left(\mathbf{h}_1^{(t+1)}\right)^T\bar{\mathbf{y}}_1^{(t)},\ \ldots,\ \left(\mathbf{h}_p^{(t+1)}\right)^T\bar{\mathbf{y}}_p^{(t)}\right]^T \tag{37}$$

and

$$\boldsymbol{\Sigma}^{(t+1)} = \left(\mathbf{H}^{(t+1)}\left(\mathbf{H}^{(t+1)}\right)^T\right)^{-1}, \tag{38}$$

where, for $j = 1, 2, \ldots, p$,

$$\mathbf{h}_j^{(t+1)} = \left(\mathbf{L}_j^{(t)}\right)^{-T}\left(0,\ \ldots,\ 0,\ n_{k(j)}^{\frac{1}{2}}\right)^T, \tag{39}$$

with $\mathbf{L}_j^{(t)}(\mathbf{L}_j^{(t)})^T$ being the lower triangular Cholesky decomposition of $\mathbf{S}_j^{(t)}$; i.e.,

$$\mathbf{S}_j^{(t)} = \mathbf{L}_j^{(t)}\left(\mathbf{L}_j^{(t)}\right)^T. \tag{40}$$

Proof. See Appendix A.3. □

Finally, the robust estimation algorithm developed based on the block MM framework is summarized in Algorithm 1.

Algorithm 1 Fast robust estimation via block MM (FREBMM).

1) Initialize $\mathbf{\Sigma}^{(0)}$ as an arbitrarypositive definite matrix, $\boldsymbol{\mu}^{(0)}$ asan arbitrary vector, and $\nu^{(0)}$ as an arbitrary number$\left(\nu^{(0)} \geq \nu^{-}\right)$.
2) Iterate

$$\boldsymbol{\mu}^{(t+1)} = \left(\mathbf{H}^{(t+1)}\right)^{-T}\left[\left(\mathbf{h}_1^{(t+1)}\right)^T\bar{\mathbf{y}}_1^t, \ldots, \left(\mathbf{h}_p^{(t+1)}\right)^T\bar{\mathbf{y}}_p^t\right]^T, \tag{41}$$

$$\mathbf{\Sigma}^{(t+1)} = \left(\mathbf{H}^{(t+1)}\left(\mathbf{H}^{(t+1)}\right)^T\right)^{-1}, \tag{42}$$

and

$$\nu^{(t+1)} = \arg\max_{\nu\geq\nu^{-}} l\left(\mathbf{Y}|\boldsymbol{\mu}^{(t+1)}, \mathbf{\Sigma}^{(t+1)}, \nu\right), \tag{43}$$

where $\mathbf{H}^{(t+1)}$ is an upper triangular matrixwith the first j components of its j-th column $\mathbf{h}_j^{(t+1)}$given by (39), and $\bar{\mathbf{y}}_j^{(t)}$is the weighted sample mean given by (35).

4.3. Minorizing functions comparison and acceleration

Actually, Algorithm 1 can also be regarded as an implementation of the ECME algorithm. But different from the previous ECME algorithm based on Model O1, which regards both the weights $\{\tau_i\}$ of the mixture and missing data $\{\mathbf{y}_{i,\text{mis}}\}$ as latent data, Algorithm 1 is based on the following complete data model:

$$\text{Model O2}: \quad \begin{array}{l}\mathbf{y}_{i,\text{obs}}|\boldsymbol{\mu}, \mathbf{\Sigma}, \tau_i \sim \mathcal{N}_{p_{k(i)}}\left(\boldsymbol{\mu}_{(p_{k(i)})}, \mathbf{\Sigma}_{(p_{k(i)})}/\tau_i\right),\\ \tau_i \sim Gamma(\nu/2, \nu/2).\end{array} \tag{44}$$

which only considers $\{\tau_i\}$ as latent data. The resulting expectation of the complete data log-likelihood is

$$\begin{aligned} &Q_2\left(\boldsymbol{\mu}, \mathbf{\Sigma}, \nu^{(t)}|\boldsymbol{\mu}^{(t)}, \mathbf{\Sigma}^{(t)}, \nu^{(t)}\right)\\ &\quad = \mathrm{E}_{p\left(\{\tau_i\}.|\boldsymbol{\mu}^{(t)}, \mathbf{\Sigma}^{(t)}, \nu^{(t)}, \{\mathbf{y}_{i,\text{obs}}\}\right)}\left(l\left(\{\mathbf{y}_{i,\text{obs}}\}, \{\tau_i\}|\boldsymbol{\mu}, \mathbf{\Sigma}, \nu^{(t)}\right)\right)\\ &\quad = \int l\left(\{\mathbf{y}_{i,\text{obs}}\}, \{\tau_i\}|\boldsymbol{\mu}, \mathbf{\Sigma}, \nu^{(t)}\right)\\ &\qquad p\left(\{\tau_i\}|\boldsymbol{\mu}^{(t)}, \mathbf{\Sigma}^{(t)}, \nu^{(t)}, \{\mathbf{y}_{i,\text{obs}}\}\right)\mathrm{d}\tau_1 \ldots \mathrm{d}\tau_n. \end{aligned} \tag{45}$$

The posterior distribution of τ_i is

$$\tau_i|\boldsymbol{\mu}^{(t)}, \mathbf{\Sigma}^{(t)}, \nu^{(t)}, \{\mathbf{y}_{i,\text{obs}}\} \sim Gamma(a, b) \tag{46}$$

with

$$a = \frac{\nu^{(t)} + p_{k(i)}}{2} \tag{47}$$

and

$$b = \frac{\nu^{(t)} + \left(\mathbf{y}_{i,(p_{k(i)})} - \boldsymbol{\mu}_{(p_{k(i)})}^{(t)}\right)^T\left(\mathbf{\Sigma}_{(p_{k(i)})}^{(t)}\right)^{-1}\left(\mathbf{y}_{i,(p_{k(i)})} - \boldsymbol{\mu}_{(p_{k(i)})}^{(t)}\right)}{2}. \tag{48}$$

It is easy to prove that it is equivalent to the minorizing function (26).

According to Proposition 2, our minorizing function follows the shape of the original function better than that of ECME(O1) due to the fewer latent variables. To illustrate this, in Fig. 2, we show an example of the minorizing functions of the FREBMM and ECME algorithms along the line μ_1, given a randomly generated monotone incomplete data set with $p = 100$, $n = 500$, and $k = 5$. The minorizing function of the FREBMM algorithm is obviously a tighter approximation of the objective function. Therefore, we can expect that the proposed algorithm has a faster convergence rate than the ECME(O1) algorithm.

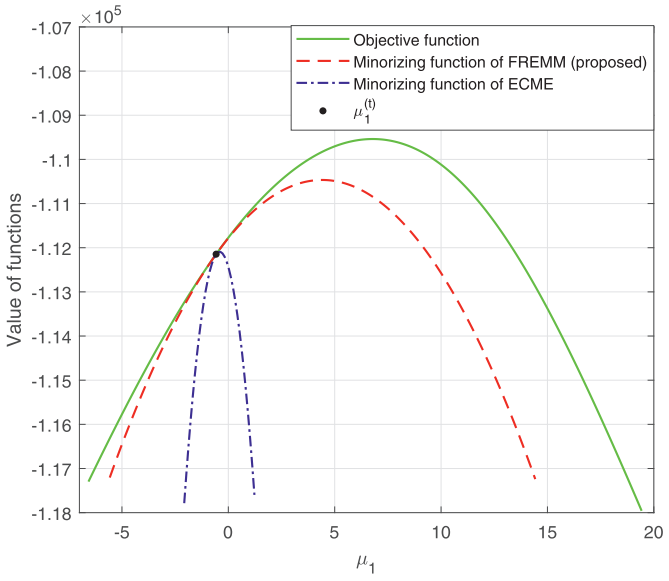

Fig. 2. Minorizing functions comparison.

In addition, similar to the case of the PX-ECME(O1) [13,15], the proposed FREBMM algorithm can also be further accelerated using the PX-EM method by embedding Model O2 within the following expanded model:

$$\text{Model X2}: \quad \begin{array}{l}\mathbf{y}_{i,\text{obs}}|\boldsymbol{\mu}^*, \mathbf{\Sigma}^*, \tau_i \sim \mathcal{N}_{p_{k(i)}}\left(\boldsymbol{\mu}^*_{(p_{k(i)})}, \mathbf{\Sigma}^*_{(p_{k(i)})}/\tau_i\right),\\ \tau_i \sim \beta Gamma(\nu/2, \nu/2).\end{array} \tag{49}$$

where $\boldsymbol{\mu}^* \in \mathbb{R}^p$, $\mathbf{\Sigma}^* \in \mathbb{S}^p_{++}$, and $\beta > 0$. The resulting accelerated scheme is given in Algorithm 2. The only difference is in (51).

Algorithm 2 PX-FREBMM.

1) Initialize $\mathbf{\Sigma}^{(0)}$ as an arbitrarypositive definite matrix, $\boldsymbol{\mu}^{(0)}$ asan arbitrary vector, and $\nu^{(0)}$ as an arbitrary number$\left(\nu^{(0)} \geq \nu^{-}\right)$.
2) Iterate

$$\boldsymbol{\mu}^{(t+1)} = \left(\mathbf{H}^{(t+1)}\right)^{-T}\left[\left(\mathbf{h}_1^{(t+1)}\right)^T\bar{\mathbf{y}}_1^{(t)}, \ldots, \left(\mathbf{h}_p^{(t+1)}\right)^T\bar{\mathbf{y}}_p^{(t)}\right]^T, \tag{50}$$

$$\mathbf{\Sigma}^{(t+1)} = \frac{n}{\sum_{i=1}^n \omega_i^{(t)}}\left(\mathbf{H}^{(t+1)}\left(\mathbf{H}^{(t+1)}\right)^T\right)^{-1}, \tag{51}$$

and

$$\nu^{(t+1)} = \arg\max_{\nu\geq\nu^{-}} l\left(\mathbf{Y}|\boldsymbol{\mu}^{(t+1)}, \mathbf{\Sigma}^{(t+1)}, \nu\right), \tag{52}$$

where $\mathbf{H}^{(t+1)}$ is an upper triangular matrixwith the first j components of its jth column $\mathbf{h}_j^{(t+1)}$given by (39), and $\bar{\mathbf{y}}_j^{(t)}$is the weighted sample mean given by (35).

Remark 1. In this paper, we assume that the samples are drawn independently from a Student's t-distribution. But actually the same idea and tricks can be applied to the ML estimation of the parameters of other Gaussian mixture distributions from monotone incomplete data. Let $\mathbf{y}_i|\boldsymbol{\mu}, \mathbf{\Sigma}, \tau_i \sim_{i.i.d.} \mathcal{N}_p(\boldsymbol{\mu}, \mathbf{\Sigma}/\tau_i)$, where $\{\tau_i\}$ are unobserved i.i.d. positive scalar random variables with known

probability functions. To obtain the ML estimates for $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, we can regard only $\{\tau_i\}$ as latent data and derive the closed-form maximizer for the expected complete data log-likelihood function using the tricks in Propositions 3 and 4.

4.4. Computational cost

Now we compare the computational complexity of the proposed FREBMM and PX-FREBMM algorithms with that of the existing ECME and PX-ECME algorithms. The per-iteration computational cost comes from two sources: the computation of $\boldsymbol{\mu}^{(t+1)}$ and $\boldsymbol{\Sigma}^{(t+1)}$, and the one-dimensional search for $\nu^{(t+1)}$. Since the update method for $\nu^{(t+1)}$ is the same in all the four algorithms, here we only consider the computational cost for $\boldsymbol{\mu}^{(t+1)}$ and $\boldsymbol{\Sigma}^{(t+1)}$. For the proposed FREBMM and PX-FREBMM algorithms, we need to first compute $\{\mathbf{S}_j^{(t)}\}$, and then do the Cholesky decomposition for each $\mathbf{S}_j^{(t)}$. The cost for computing $\{\mathbf{S}_j^{(t)}\}$ is $\mathcal{O}(\sum_{i=1}^K p_k^3(n_k - n_{k-1}))$, and the computational cost for all the Cholesky decomposition's is $\mathcal{O}(p^4)$. The total computational cost is $\mathcal{O}(p^4 + \sum_{i=1}^K p_k^3(n_k - n_{k-1}))$. For the ECME and PX-EM algorithms, the dominating cost is the computation of many expectations. The resulting per-iteration computational cost for is $\mathcal{O}(np^2 + \sum_{i=1}^K (p_k^3 + pp_k)(n_k - n_{k-1}) + \sum_{i=1}^K p^2 p_k)$.

5. Algorithms for regularized robust estimation

In this section, we derive two algorithms for the proposed regularized robust estimator (14) based on the block MM framework.

5.1. Regularized robust estimation via block MM

Recall the objective function of the regularized robust estimation problem (14) is

$$l^{\text{shrink}}(\mathbf{Y}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu) = l(\mathbf{Y}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu) - \alpha u(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu). \tag{53}$$

Similarly, we exploit the block MM framework to solve it. At iteration $t+1$, we first update $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, with ν fixed as $\nu^{(t)}$, and then update ν, with $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ fixed as $\boldsymbol{\mu}^{(t+1)}$ and $\boldsymbol{\Sigma}^{(t+1)}$. The optimization of ν can be solved by one-dimensional search. For the optimization of $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, according to (9) and (19), we have

$$\begin{aligned} l\left(\mathbf{Y}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu^{(t)}\right) &\geq Q\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu^{(t)}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)}\right) \\ &+ l\left(\mathbf{Y}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)}\right) - Q\left(\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)}\right), \end{aligned} \tag{54}$$

therefore, at $(\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)})$, $l^{\text{shrink}}(\mathbf{Y}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu^{(t)})$ is minorized by

$$\begin{aligned} &g_1^{\text{shrink}}\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)}\right) \\ &= Q\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu^{(t)}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)}\right) - \alpha u\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu^{(t)}\right) + const. \\ &= \sum_{i=1}^n \left\{ -\frac{1}{2}\text{Tr}\left(\mathsf{E}\left(\tau_i \mathbf{y}_i \mathbf{y}_i^T\right)\boldsymbol{\Sigma}^{-1}\right) + \boldsymbol{\mu}^T\boldsymbol{\Sigma}^{-1}\mathsf{E}(\tau_i\mathbf{y}_i) - \frac{1}{2}\boldsymbol{\mu}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{\mu}\mathsf{E}(\tau_i) \right\} \\ &\quad -\frac{n}{2}\log\left(\det\left(\boldsymbol{\Sigma}\right)\right) - \frac{\alpha}{2}(\mathbf{t}-\boldsymbol{\mu})^T\left(\frac{\nu^{(t)}}{\nu^{(t)}-2}\boldsymbol{\Sigma}\right)^{-1}(\mathbf{t}-\boldsymbol{\mu}) \\ &\quad -\frac{\alpha}{2}\text{Tr}\left(\left(\frac{\nu^{(t)}}{\nu^{(t)}-2}\boldsymbol{\Sigma}\right)^{-1}\mathbf{T}\right) - \frac{\alpha}{2}\log\det\left(\frac{\nu^{(t)}}{\nu^{(t)}-2}\boldsymbol{\Sigma}\right) + const. \end{aligned} \tag{55}$$

Setting the gradient of $g_1^{\text{shrink}}(\boldsymbol{\mu}, \boldsymbol{\Sigma}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)})$ to zero gives the closed-form update (56) and (57) in Algorithm 3.

Algorithm 3 Regularized robust estimation via block MM (RREBMM).

1) Initialize $\boldsymbol{\Sigma}^{(0)}$ as an arbitrarypositive definite matrix, $\boldsymbol{\mu}^{(0)}$ asan arbitrary vector, and $\nu^{(0)}$ as an arbitrary number$\left(\nu^{(0)} \geq \nu^-\right)$.
2) Iterate

$$\boldsymbol{\mu}^{(t+1)} = \frac{\sum_{i=1}^n \mathsf{E}(\tau_i)}{\sum_{i=1}^n \mathsf{E}(\tau_i) + \alpha\frac{\nu^{(t)}-2}{\nu^{(t)}}} \frac{\sum_{i=1}^n \mathsf{E}(\tau_i\mathbf{y}_i)}{\sum_{i=1}^n \mathsf{E}(\tau_i)} + \frac{\alpha\frac{\nu^{(t)}-2}{\nu^{(t)}}}{\sum_{i=1}^n \mathsf{E}(\tau_i) + \alpha\frac{\nu^{(t)}-2}{\nu^{(t)}}}\mathbf{t}, \tag{56}$$

$$\begin{aligned} \boldsymbol{\Sigma}^{(t+1)} =& \frac{n}{n+\alpha}\frac{1}{n}\sum_{i=1}^n \left\{ \mathsf{E}\left(\tau_i\mathbf{y}_i\mathbf{y}_i^T\right) - 2\mathsf{E}(\tau_i\mathbf{y}_i)\left(\boldsymbol{\mu}^{(t+1)}\right)^T \right. \\ &\left. + \mathsf{E}(\tau_i)\boldsymbol{\mu}^{(t+1)}\left(\boldsymbol{\mu}^{(t+1)}\right)^T \right\} \\ &+ \frac{\alpha}{n+\alpha}\left\{ \frac{\nu^{(t)}-2}{\nu^{(t)}}\mathbf{T} + \frac{\nu^{(t)}-2}{\nu^{(t)}}\left(\mathbf{t}-\boldsymbol{\mu}^{(t+1)}\right)\left(\mathbf{t}-\boldsymbol{\mu}^{(t+1)}\right)^T \right\}, \end{aligned} \tag{57}$$

and

$$\nu^{(t+1)} = \arg\max_{\nu\geq\nu^-} l^{\text{shrink}}\left(\mathbf{Y}|\boldsymbol{\mu}^{(t+1)}, \boldsymbol{\Sigma}^{(t+1)}, \nu\right), \tag{58}$$

where the expectations can be computed based on the method in [17].

We can see the update is a linear combination of the estimates from the samples and target. When $\alpha = 0$, the shrinkage estimator reduces to the previous nonshrinkage estimator (see Section 2.3), and when $\alpha \to +\infty$, the shrinkage estimator reduces to the trivial case yielding the prior target. The term $\frac{\alpha}{n+\alpha}(\frac{\nu^{(t)}-2}{\nu^{(t)}}\mathbf{T} + \frac{\nu^{(t)}-2}{\nu^{(t)}}(\mathbf{t}-\boldsymbol{\mu}^{(t+1)})(\mathbf{t}-\boldsymbol{\mu}^{(t+1)})^T)$ helps to make $\boldsymbol{\Sigma}^{(t+1)}$ well conditioned, and thus, allows continuation of the iterative process.

Even better, from (26), at $(\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)})$, $l^{\text{shrink}}(\mathbf{Y}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu^{(t)})$ is also minorized by

$$\begin{aligned} &g_2^{\text{shrink}}\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)}\right) = g\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)}\right) \\ &\quad -\alpha u\left(\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu^{(t)}\right) + const. \\ &= \sum_{k=1}^K \left\{ \sum_{i=n_{k-1}+1}^{n_k} -\frac{1}{2}\omega_i^{(t)}\left(\mathbf{y}_{i,(p_k)} - \boldsymbol{\mu}_{(p_k)}\right)^T\left(\boldsymbol{\Sigma}_{(p_k)}\right)^{-1}\left(\mathbf{y}_{i,(p_k)} - \boldsymbol{\mu}_{(p_k)}\right) \right. \\ &\quad \left. -\frac{n_k - n_{k-1}}{2}\log\det\left(\boldsymbol{\Sigma}_{(p_k)}\right) \right\} - \frac{\alpha}{2}(\mathbf{t}-\boldsymbol{\mu})^T\left(\frac{\nu^{(t)}}{\nu^{(t)}-2}\boldsymbol{\Sigma}\right)^{-1}(\mathbf{t}-\boldsymbol{\mu}) \\ &\quad -\frac{\alpha}{2}\text{Tr}\left(\left(\frac{\nu^{(t)}}{\nu^{(t)}-2}\boldsymbol{\Sigma}\right)^{-1}\mathbf{T}\right) - \frac{\alpha}{2}\log\det\left(\frac{\nu^{(t)}}{\nu^{(t)}-2}\boldsymbol{\Sigma}\right) + const. \end{aligned} \tag{59}$$

Compared with the minorizing function g_1^{shrink}, g_2^{shrink} is a tighter approximation of $l^{\text{shrink}}(\mathbf{Y}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu^{(t)})$, since $g(\boldsymbol{\mu}, \boldsymbol{\Sigma}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)})$ is a tighter approximation of $l(\mathbf{Y}|\boldsymbol{\mu}, \boldsymbol{\Sigma}, \nu^{(t)})$ than the minorizing function in the ECME(O1). On the other hand, $g_2^{\text{shrink}}(\boldsymbol{\mu}, \boldsymbol{\Sigma}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)})$ is so complicated that we cannot maximize it directly. Luckily, after reparameterization using Lemma 1, we can derive a closed-form maximizer. The maximizer is given in Proposition 5, and the resulting algorithm is summarized in Algorithm 4. Similar to Algorithm 1, Algorithms 3 and 4 can also be considered as penalized ECME algorithms.

Algorithm 4 Fast regularized robust estimation via block MM (FR-REBMM).

1) Initialize $\boldsymbol{\Sigma}^{(0)}$ as an arbitrarypositive definite matrix, $\boldsymbol{\mu}^{(0)}$ asan arbitrary vector, and $\nu^{(0)}$ as an arbitrary number$\left(\nu^{(0)} \geq \nu^{-}\right)$.
2) Iterate

$$\boldsymbol{\mu}^{(t+1)} = \left(\mathbf{H}^{(t+1)}\right)^{-T}\left[\left(\mathbf{h}_1^{(t+1)}\right)^T\tilde{\mathbf{y}}_1^{(t)}, \ldots, \left(\mathbf{h}_p^{(t+1)}\right)^T\tilde{\mathbf{y}}_p^{(t)}\right]^T, \tag{60}$$

$$\boldsymbol{\Sigma}^{(t+1)} = \left(\mathbf{H}^{(t+1)}\left(\mathbf{H}^{(t+1)}\right)^T\right)^{-1}, \tag{61}$$

and

$$\nu^{(t+1)} = \arg\max_{\nu \geq \nu^{-}} l^{\text{shrink}}\left(\mathbf{Y}|\boldsymbol{\mu}^{(t+1)}, \boldsymbol{\Sigma}^{(t+1)}, \nu\right), \tag{62}$$

where $\mathbf{H}^{(t+1)}$ is an upper triangular matrixwith the first j components of its jth column $\mathbf{h}_j^{(t+1)}$given by (66), $\tilde{\mathbf{y}}_j^{(t)}$given by (35).

Proposition 5. *The minorizing function (26) is maximized by*

$$\boldsymbol{\mu}^{(t+1)} = \left(\mathbf{H}^{(t+1)}\right)^{-T}\left[\left(\mathbf{h}_1^{(t+1)}\right)^T\tilde{\mathbf{y}}_1^{(t)}, \ldots, \left(\mathbf{h}_p^{(t+1)}\right)^T\tilde{\mathbf{y}}_p^{(t)}\right]^T \tag{63}$$

and

$$\boldsymbol{\Sigma}^{(t+1)} = \left(\mathbf{H}^{(t+1)}\left(\mathbf{H}^{(t+1)}\right)^T\right)^{-1}, \tag{64}$$

where, for $j = 1, 2, \ldots, p$,

$$\tilde{\mathbf{y}}_j^{(t)} = \frac{\Omega_j^{(t)}\bar{\mathbf{y}}_j^{(t)} + \alpha\frac{\nu^{(t)}-2}{\nu^{(t)}}\mathbf{t}_{(j)}}{\Omega_j^{(t)} + \alpha\frac{\nu^{(t)}-2}{\nu^{(t)}}}, \tag{65}$$

and

$$\mathbf{h}_j^{(t+1)} = \left(\tilde{\mathbf{L}}_j^{(t)}\right)^{-T}\left(0, \ldots, 0, \sqrt{n_{k(j)} + \alpha}\right)^T, \tag{66}$$

with $\tilde{\mathbf{L}}_j^{(t)}(\tilde{\mathbf{L}}_j^{(t)})^T$ *being the lower triangular Cholesky decomposition for*

$$\tilde{\mathbf{S}}_j^{(t)} = \frac{\alpha\frac{\nu^{(t)}-2}{\nu^{(t)}}\Omega_j^{(t)}\left(\bar{\mathbf{y}}_j^{(t)} - \mathbf{t}_{(j)}\right)\left(\bar{\mathbf{y}}_j^{(t)} - \mathbf{t}_{(j)}\right)^T}{\Omega_j^{(t)} + \alpha\frac{\nu^{(t)}-2}{\nu^{(t)}}} + \mathbf{S}_j^{(t)} + \alpha\frac{\nu^{(t)}-2}{\nu^{(t)}}\mathbf{T}_{(j)}. \tag{67}$$

Proof. See Appendix A.4. □

5.2. Computational cost

The introduction of the regularization term does not lead to much additional computational cost. For the RREBMM algorithm, the per-iteration computational cost for $\boldsymbol{\mu}^{(t+1)}$ and $\boldsymbol{\Sigma}^{(t+1)}$ is $\mathcal{O}(np^2 + \sum_{i=1}^{K}(p_k^3 + pp_k)(n_k - n_{k-1}) + \sum_{i=1}^{K} p^2 p_k)$, the same with the ECME(O1). For FRREBMM algorithm, the per-iteration computational cost is $\mathcal{O}(p^4 + \sum_{i=1}^{K} p_k^3(n_k - n_{k-1}))$, the same with the FREBMM algorithm.

6. Simulations

To show the performance of the proposed regularized robust estimator and the proposed algorithms, we present some numerical experimental results in this section. All experiments were conducted on a PC with a 3.20 GHz i5-4570 CPU and 8 GB RAM. The estimation performance is quantified by the normalized mean square errors (NMSEs) defined as $\text{NMSE}_{\boldsymbol{\mu}} = \frac{\text{E}\|\hat{\boldsymbol{\mu}} - \boldsymbol{\mu}_{\text{true}}\|^2}{\|\boldsymbol{\mu}_{\text{true}}\|^2}$ and $\text{NMSE}_{\mathbf{R}} = \frac{\text{E}\|\hat{\mathbf{R}} - \mathbf{R}_{\text{true}}\|_F^2}{\|\mathbf{R}_{\text{true}}\|_F^2}$, where $\hat{\boldsymbol{\mu}}$ and $\hat{\mathbf{R}} = \frac{\hat{\nu}}{\hat{\nu}-2}\hat{\boldsymbol{\Sigma}}$ are the estimates for mean and covariance matrix. In all the synthetic data simulations, the samples are drawn from the heavy-tailed distribution $t_p(\boldsymbol{\mu}_{\text{true}}, \boldsymbol{\Sigma}_{\text{true}}, \nu_{\text{true}})$, where $\boldsymbol{\mu}_{\text{true}} = \mathbf{1}$, $\boldsymbol{\Sigma}_{\text{true}}$ is a Toeplitz covariance matrix of the form $(\boldsymbol{\Sigma}_{\text{true}})_{ij} = 0.8^{|i-j|}$, and $\nu_{\text{true}} = 3$. The true covariance is $\mathbf{R}_{\text{true}} = \frac{\nu_{\text{true}}}{\nu_{\text{true}}-2}\boldsymbol{\Sigma}_{\text{true}}$.

6.1. Comparison of the algorithms for robust estimation

In this part, we compare the proposed FREBMM algorithm and its variant PX-FREBMM (i.e., Algorithms 1 and 2) with the existing ECME and PX-ECME algorithms in [13,16] for the robust estimation of the mean and covariance matrix from the monotone incomplete data. The stopping criteria for all the algorithms are $\frac{\|\boldsymbol{\Sigma}^{(t+1)} - \boldsymbol{\Sigma}^{(t)}\|_F}{\|\boldsymbol{\Sigma}^{(t)}\|_F} < 10^{-4}$, $\frac{\|\boldsymbol{\mu}^{(t+1)} - \boldsymbol{\mu}^{(t)}\|_2}{\|\boldsymbol{\mu}^{(t)}\|_2} < 10^{-4}$, and $\frac{|\nu^{(t+1)} - \nu^{(t)}|}{|\nu^{(t)}|} < 10^{-4}$. First, we test the performance of the four algorithms on a monotone incomplete data set with $p = 100$, $n = 500$ and $k = 5$ for 100 random initial points. Table 1 displays the comparison in terms of numbers of iterations required to converge, the CPU time cost, and the estimation errors. The proposed FREBMM and PX-FREBMM algorithms achieve the same estimation accuracy with the ECME and PX-ECME algorithms using much fewer iterations and less time. Fig. 3 depicts the evolution curve of the objective value versus the number of iterations for a random initial point. The convergence rates of the proposed FREBMM and PX-FREBMM algorithms dominate the benchmarks.

As discussed in Section 4.3, the reason for the faster convergence of the proposed algorithms is that the minorizing function is tighter, since the proposed algorithms do not consider the missing values $\{\mathbf{y}_{i,\text{mis}}\}$ as latent variables, and have fewer latent variables than the benchmarks. Since the number of missing values is decided by the missing rates ($\varphi = 1 - \frac{\sum_{k=1}^{K}(n_k - n_{k-1})p_k}{np}$) and data set size ($p$, n), we next test the computational complexity of the four algorithms for data sets with different missing rates and sizes. For a given set of p, n, and missing rate φ, we randomly generate 100 monotone incomplete data sets from $t_p(\boldsymbol{\mu}_{\text{true}}, \boldsymbol{\Sigma}_{\text{true}}, \nu_{\text{true}})$ and test the four algorithms on these data sets. Note that number of groups K is randomly generated for each data set. Fig. 4 shows the average running time versus different missing rates with the data size fixed as $p = 100$ and $n = 500$. Fig. 5 shows the average running time versus different data sizes with the missing rate fixed as

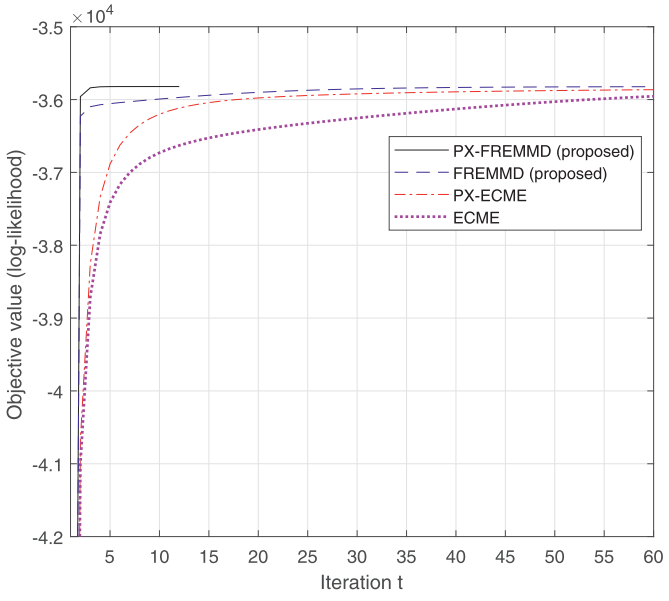

Fig. 3. Objective value versus iteration.

Table 1
Performance comparison of different algorithms.

	ECME	PX-ECME	FREBMM (proposed)	PX-FREBMM (proposed)
Average number of iterations	285	279	140	12
Average CPU time in sec	119.90	117.00	56.42	4.71
NMSE of $\hat{\boldsymbol{\mu}}$	0.0023	0.0023	0.0023	0.0023
NMSE of $\hat{\mathbf{R}}$	0.1263	0.1263	0.1263	0.1263

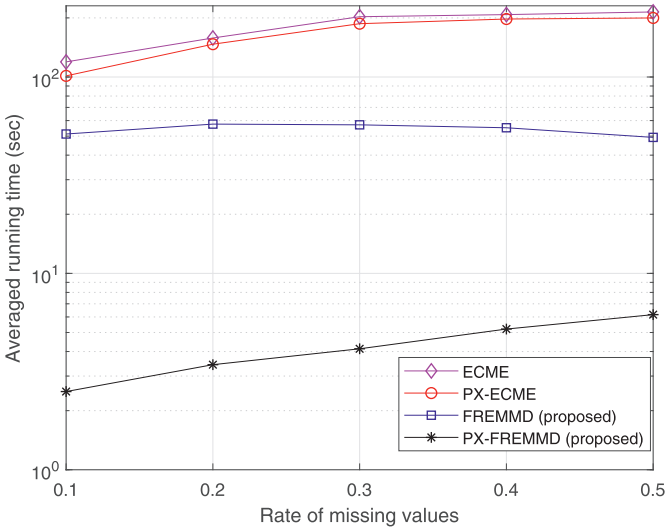

Fig. 4. Average running time versus missing rates φ.

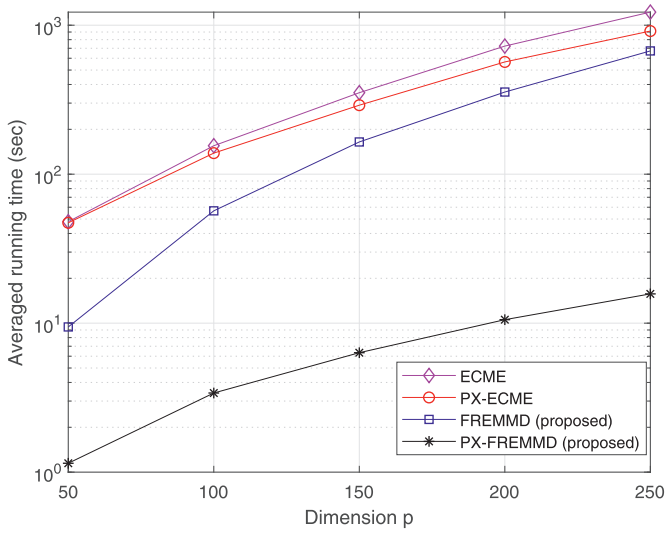

Fig. 5. Average running time versus dimension p.

$\varphi = 20\%$ (for convenience, we let $n = 5p$). We can see that the proposed PX-FREBMM algorithm is more than one order of magnitude faster than the benchmarks for all the settings.

6.2. Regularized robust estimation

In this part, we show the performance of the proposed shrinkage robust estimator in the small sample size regime. We consider two classes of estimators: the estimators for the incomplete data with the monotone missing-data pattern, which include the Student's t ML estimator, the shrinkage Gaussian estimator in [23] and the proposed shrinkage Student's t ML estimator, and the estimator based on only complete samples, the shrinkage Student's t ML estimator in [21]. The shrinkage target for the covariance matrix is set to be the identity matrix motivated by Ledoit and Wolf [18], and the target for the mean is set to be the sample median. Note that the shrinkage target we use here does not depend on any prior knowledge about the true parameter. As for the tuning parameter α of the proposed shrinkage estimator, we let $\rho(\alpha) = \frac{n}{n+\alpha}$, and search for $\alpha^\star$ that yields the shrinkage estimator with the smallest NMSE with ρ in $\{0.1, 0.2, \ldots, 1\}$. This is to eliminate the effect of parameter tuning. Since, in [23], the authors only developed the shrinkage Gaussian estimator for the monotone data with two groups, we first test the performance of the estimators on monotone incomplete data sets with two groups. We consider 100-dimensional monotone incomplete data sets, where there are n_1 complete samples in the first group and 50 samples in the second group with only first 75 components observed. For a given n_1, we randomly generate 100 monotone incomplete data sets from $t_p(\boldsymbol{\mu}_{\text{true}}, \boldsymbol{\Sigma}_{\text{true}}, \nu_{\text{true}})$. Figs. 6 and 7 show the average estimation errors. The existing shrinkage Gaussian estimator is too inefficient for heavy-tailed data sets, and the proposed shrinkage Student's t ML estimator outperforms other estimators, since it considers the heavy-tail, is well stabilized by the shrinkage, and makes use of whole data set. Fig. 8 gives an example for the convergence of the proposed algorithms RREBMM and FRREBMM. Both algorithms converge, and the FRREBMM algorithm is faster than the RREBMM algorithm, since its minorizing function is tighter.

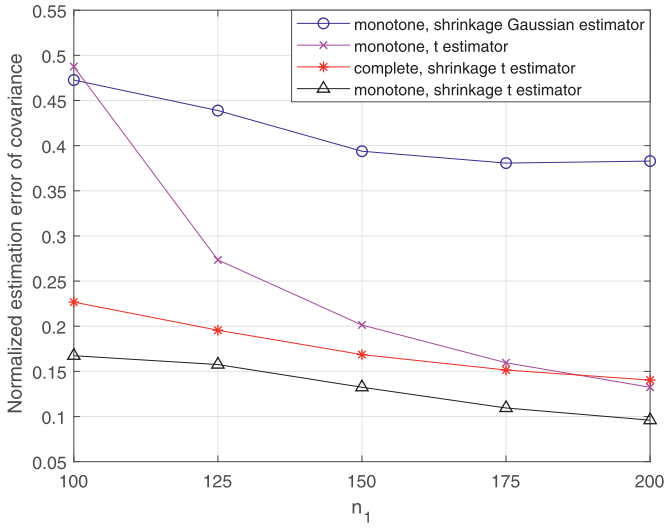

Fig. 6. Estimation errors of the covariance $\hat{\mathbf{R}}$.

Then we test the performance of the proposed estimator on monotone incomplete data sets with different numbers of groups. We set $p = 100$. For an incomplete data set with K groups, there are 150 complete samples in the first group and 50 samples in any other kth group with the first $110 - 10k$ components observed. Similarly, a number of 100 incomplete data sets are generated for each setting. Figs. 9 and 10 show the average estimation errors for data sets with different K's. The proposed shrinkage estimator always provide more reliable estimates than other two estimators.

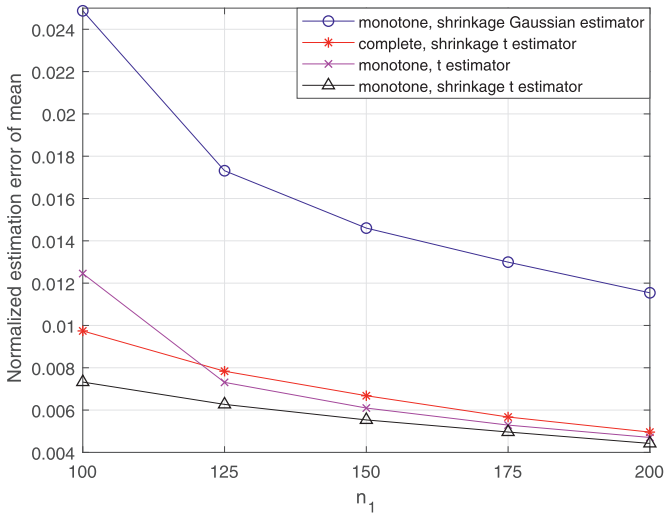

Fig. 7. Estimation errors of $\hat{\boldsymbol{\mu}}$.

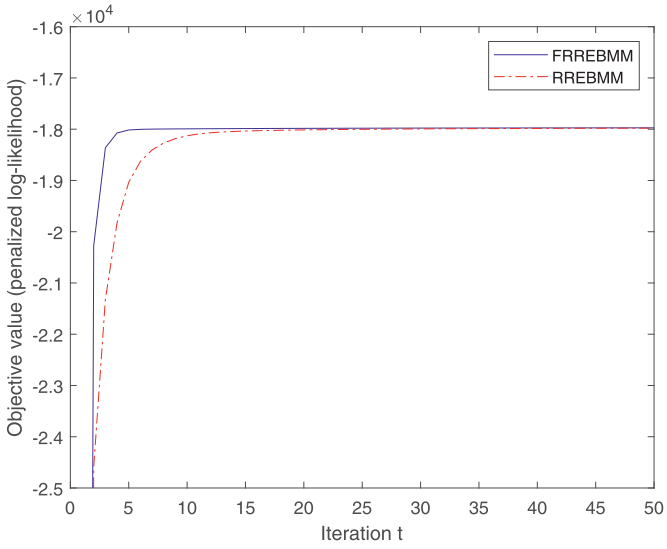

Fig. 8. Objective value versus iteration.

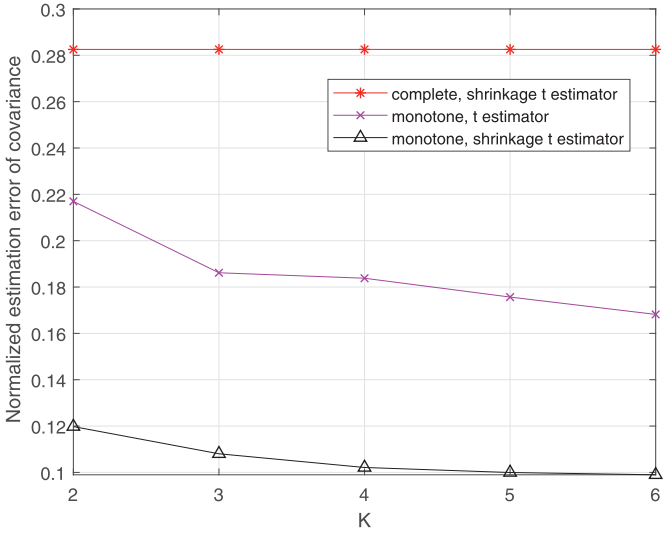

Fig. 9. Estimation errors of the covariance $\hat{\mathbf{R}}$.

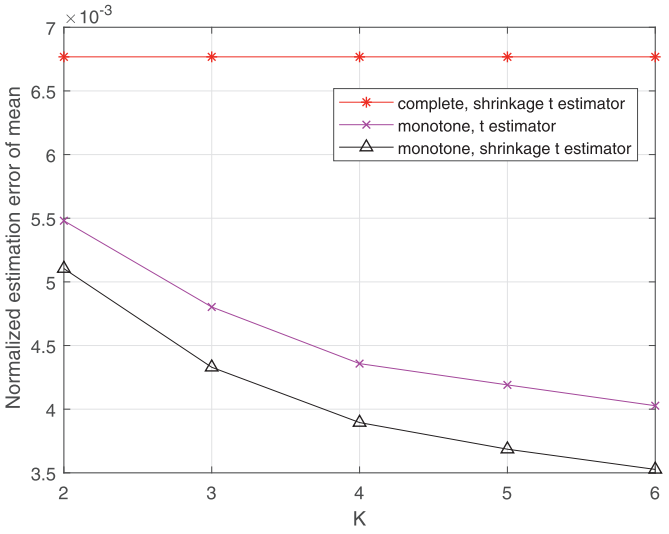

Fig. 10. Estimation errors of $\hat{\boldsymbol{\mu}}$.

Now we show the sensitivity of the proposed shrinkage estimator to the shrinkage target $\mathbf{t}$ and $\mathbf{T}$. We set $K = 2$, $p = 100$, $p_2 = 75$, $n_1 = 100$, and $n_2 = 150$. We first fix $\mathbf{t}$ as the sample median, and analyze the sensitivity to $\mathbf{T}$. Table 2 lists the average estimation errors with $\mathbf{T}$ being a Toeplitz matrix with $\mathbf{T}_{ij} = 3\beta^{|i-j|}$, $\beta \in \{0.1, 0.2, \ldots, 0.7\}$ (the true covariance matrix $(\mathbf{T}_{\text{true}})_{ij} = 3 * 0.8^{|i-j|}$). Then we fix $\mathbf{T}$ as the identity matrix, and analyze the sensitivity to $\mathbf{t}$. Table 3 lists the average estimation errors with $\mathbf{t} = t\mathbf{1}$, $t \in \{0.1, 0.2, \ldots, 0.9\}$. The tables indicate that the estimation accuracy increases as the prior target gets close to the true value. Even if the prior target is far from the true value, e.g., $t = 0.1$ and $\beta = 0.1$, the estimation error is still no worse than the existing nonshrinkage Student's t ML estimator. The reason is that when $t = 0.1$, the regularization parameter $\alpha^\star$ is small, and the estimates are dominated by the information from samples, and when $\beta = 0.1$, $\mathbf{T}$ is close to the indentity matrix, and this still helps in improving the accuracy by shrinking the eigenvalues of $\hat{\mathbf{R}}$ towards to the center in the small sample regime. To summarize, one can expect that a more informative prior ($\mathbf{t}$, $\mathbf{T}$) close to the true value can lead to more accurate estimation. Even the prior ($\mathbf{t}$, $\mathbf{T}$) is wrong, it still performs no worse than the nonshrinkage Student's t ML estimator given that $\alpha^\star$ is well selected.

In the last simulation, we apply the proposed shrinkage robust estimator for the monotone missing-data pattern to estimate the covariance matrix for stocks with available histories of various lengths, and compare it with other different covariance estimators on a real financial market data set. We consider 48 constituent stocks of the Hang Seng Index, and download their dividend-adjusted monthly close prices from Jan. 1998 to Nov. 2017 from the Bloomberg. Since different stocks went public via IPOs at different times, the number of historical monthly log returns for each stock varies from 86 to 239. These log returns can be considered as following the monotone missing-data pattern (see Fig. 11 for an illustration) [25]. An important reason for estimating the covariance matrix of stocks is to provide inputs into portfolio construction. As conventionally done in the financial literature [21,33], we compare the performance of the covariance matrix estimators in the setup of the minimum variance portfolio constructed using the estimates based on the historical log returns. The mathematical formulation of the minimum variance portfolio construction problem is

$$\begin{aligned} &\underset{\mathbf{w}}{\text{minimize}} \ \mathbf{w}^T\mathbf{R}\mathbf{w} \\ &\text{subject to } \mathbf{1}^T\mathbf{w} = 1, \end{aligned} \tag{68}$$

Table 2
Average estimation errors of the proposed shrinkage estimator for different **T**.

β	0.1	0.2	0.3	0.4	0.5	0.6	0.7	Nonshrinkage
NMSE of $\hat{\mathbf{R}}$	0.2186	0.1914	0.1680	0.1448	0.1183	0.0853	0.0444	0.5239

Table 3
Average estimation errors of the proposed shrinkage estimator for different **t**.

t	0.1	0.2	0.3	0.4	0.5
NMSE of $\hat{\boldsymbol{\mu}}$	0.0077	0.0076	0.0075	0.0074	0.0073
t	0.6	0.7	0.8	0.9	nonshrinkage
NMSE of $\hat{\boldsymbol{\mu}}$	0.0070	0.0067	0.0061	0.0042	0.0102

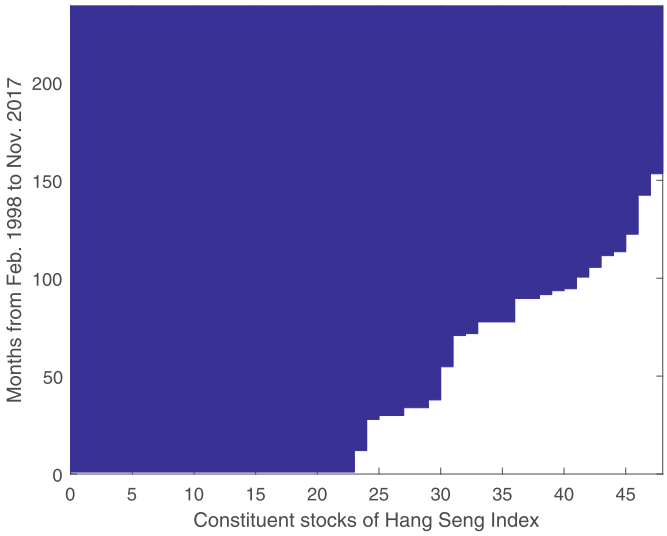

Fig. 11. Monotone missing-data pattern in the log returns of Heng Seng Index constituent stocks. The blue parts are the observed values. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

where **R** is the estimated covariance matrix of the stocks and **w** denotes the weights to be allocated on different stocks [21].

Let us regard Feb. 1998 as the first month. At a particular month n, for nonshrinkage estimators, we use the previous $n-1$ monthly returns to estimate the covariance matrix **R**, and find the optimal allocation weights $\mathbf{w}^\star$ based on the estimate. For shrinkage estimators, we first divide the previous $n-1$ monthly returns into two parts with the first $n-1-n_{\text{val}}$ monthly returns as the training data to estimate the covariance **R** and find the optimal allocation weights **w** for different regularization parameters α, and the remaining n_{val} monthly returns as the validation data for selecting the optimal $\alpha^\star$ that yields the smallest $\text{Var}(\{\mathbf{w}^T\mathbf{r}_t\}_{t=n-n_{\text{val}},\ldots,n-1})$. Then we use the overall $n-1$ monthly returns to estimate the covariance matrix using $\alpha^\star$, and compute the corresponding optimal allocation weights $\mathbf{w}^\star$.

After obtaining the allocation weights $\mathbf{w}^\star$ for all the estimators, we construct the portfolio using $\mathbf{w}^\star$, and compute the portfolio variance in the next n_{test} months. This estimation and test procedure is repeated from $n = 187$ to 227. In the simulation, we set $\rho(\alpha) = \frac{n}{n+\alpha}$, and search for $\alpha^\star$ with $\rho(\alpha)$ in $\{0.1, 0.2, \ldots, 1\}$. The shrinkage target for mean and covariance are sample median and identity matrix, respectively. The parameters are set to be $n_{\text{val}} = 12$ and $n_{\text{test}} = 12$.

Two classes of estimators are considered: the estimators for the incomplete data with the monotone missing-data pattern, which include the Gaussian ML estimator, the Student's t ML estimator, and the proposed shrinkage Student's t ML estimator, and the estimators based on only complete samples, which include sample average, the Student's t ML estimator, and shrinkage Student's t estimator. Fig. 12 compares the risk (variance) of the minimum variance portfolio constructed based on different covariance estimators. The shrinkage estimators yield lower risk than the nonshrinkage estimators, and the proposed shrinkage Student's t estimator for the monotone missing-data pattern performs better than the existing shrinkage estimator Student's t based only on complete samples, since it makes good use of all the data.

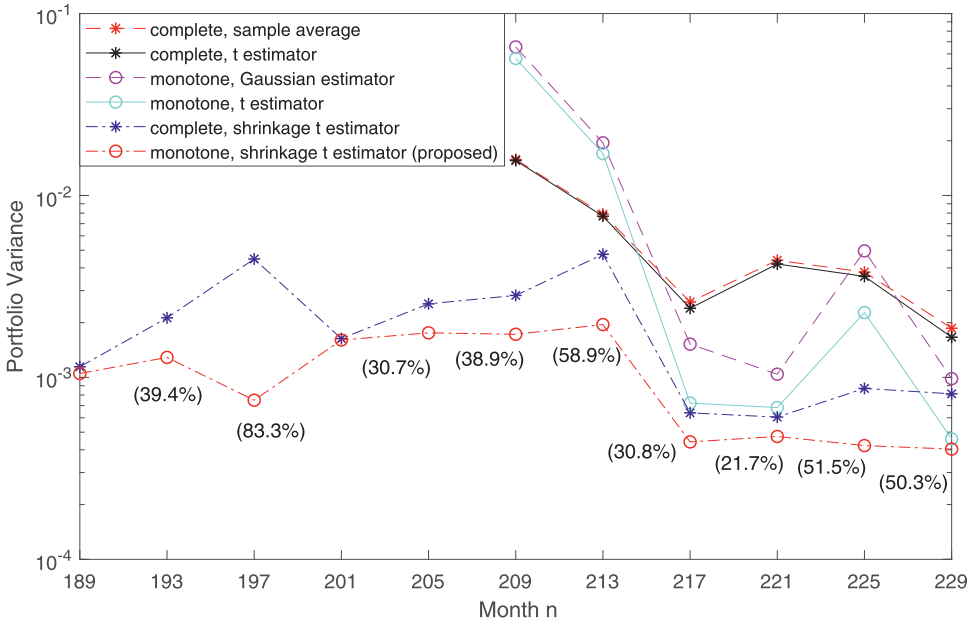

Fig. 12. Risk (variance) comparison of portfolio constructed based on different covariance estimators (the numbers in the parentheses are the percentage decreases of portfolio variances obtained by monotone shrinkage t estimator compared with that of complete shrinkage t estimator).

7. Conclusions

In this paper, we have considered the robust estimation of the mean and covariance matrix for incomplete data with the monotone missing-data pattern. The contribution of his paper is twofold. First, we have derived two algorithms based on the MM framework for the existing Student's t ML estimator. The minorizing function of the proposed algorithms achieves a much tighter approximation of the objective function than that of the existing algorithms, therefore, the proposed algorithms enjoy faster convergence rates. Secondly, we have proposed a regularized estimator by adding a penalty term to the original Student's t log-likelihood function. And two optimization algorithms have been designed for it based on the MM framework. The proposed regularized estimator can work considerably better in small sample size regime.

Although this paper focuses on the incomplete data with the monotone missing-data pattern, the above proposed regularized estimator and algorithms can be extended to incomplete data with any arbitrary missing-data pattern. Similarly, we can use the ML estimator assuming the samples follow a Student's t-distribution, and regularize the estimator by shrinking the estimator to a prior target.

Declaration of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported by the Hong Kong RGC 16208917 research grant.

Appendix A. Proof

A.1. Proof for Proposition 2

Let us denote by $Q_{\boldsymbol{Z}_1}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ the expectation of complete data log-likelihood with $\boldsymbol{Z}_1$ considered as the latent variable, and by $Q_{\boldsymbol{Z}_1\boldsymbol{Z}_2}(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)})$ the expectation of complete data log-likelihood with both $\boldsymbol{Z}_1$ and $\boldsymbol{Z}_2$ considered as the latent variables. According the definition, we have

$$
\begin{aligned}
Q_{\boldsymbol{Z}_1}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}\right) &= \int \log\left(p\left(\mathbf{Y},\boldsymbol{Z}_1|\boldsymbol{\theta}\right)\right)p\left(\boldsymbol{Z}_1|\mathbf{Y},\boldsymbol{\theta}^{(t)}\right)\mathrm{d}\boldsymbol{Z}_1 = \int \log\left(\int p\left(\boldsymbol{Z}_2|\mathbf{Y},\boldsymbol{Z}_1,\boldsymbol{\theta}^{(t)}\right)\frac{p\left(\mathbf{Y},\boldsymbol{Z}_1,\boldsymbol{Z}_2|\boldsymbol{\theta}\right)}{p\left(\boldsymbol{Z}_2|\mathbf{Y},\boldsymbol{Z}_1,\boldsymbol{\theta}^{(t)}\right)}\mathrm{d}\boldsymbol{Z}_2\right)p\left(\boldsymbol{Z}_1|\mathbf{Y},\boldsymbol{\theta}^{(t)}\right)\mathrm{d}\boldsymbol{Z}_1\\
&\geq \int\left(\int p\left(\boldsymbol{Z}_2|\mathbf{Y},\boldsymbol{Z}_1,\boldsymbol{\theta}^{(t)}\right)\log\left(\frac{p\left(\mathbf{Y},\boldsymbol{Z}_1,\boldsymbol{Z}_2|\boldsymbol{\theta}\right)}{p\left(\boldsymbol{Z}_2|\mathbf{Y},\boldsymbol{Z}_1,\boldsymbol{\theta}^{(t)}\right)}\right)\mathrm{d}\boldsymbol{Z}_2\right)p\left(\boldsymbol{Z}_1|\mathbf{Y},\boldsymbol{\theta}^{(t)}\right)\mathrm{d}\boldsymbol{Z}_1\\
&= \iint \log\left(\frac{p\left(\mathbf{Y},\boldsymbol{Z}_1,\boldsymbol{Z}_2|\boldsymbol{\theta}\right)}{p}\left(\boldsymbol{Z}_2|\mathbf{Y},\boldsymbol{Z}_1,\boldsymbol{\theta}^{(t)}\right)\right)p\left(\boldsymbol{Z}_2|\mathbf{Y},\boldsymbol{Z}_1,\boldsymbol{\theta}^{(t)}\right)p\left(\boldsymbol{Z}_1|\mathbf{Y},\boldsymbol{\theta}^{(t)}\right)\mathrm{d}\boldsymbol{Z}_2\mathrm{d}\boldsymbol{Z}_1\\
&= \iint \log\left(\frac{p\left(\mathbf{Y},\boldsymbol{Z}_1,\boldsymbol{Z}_2|\boldsymbol{\theta}\right)p\left(\mathbf{Y},\boldsymbol{Z}_1|\boldsymbol{\theta}^{(t)}\right)}{p\left(\mathbf{Y},\boldsymbol{Z}_1,\boldsymbol{Z}_2|\boldsymbol{\theta}^{(t)}\right)}\right)p\left(\boldsymbol{Z}_1,\boldsymbol{Z}_2|\mathbf{Y},\boldsymbol{\theta}^{(t)}\right)\mathrm{d}\boldsymbol{Z}_2\mathrm{d}\boldsymbol{Z}_1\\
&= \iint \log\left(p\left(\mathbf{Y},\boldsymbol{Z}_1,\boldsymbol{Z}_2|\boldsymbol{\theta}\right)\right)p\left(\boldsymbol{Z}_1,\boldsymbol{Z}_2|\mathbf{Y},\boldsymbol{\theta}^{(t)}\right)\mathrm{d}\boldsymbol{Z}_2\mathrm{d}\boldsymbol{Z}_1 - \iint \log\left(p\left(\mathbf{Y},\boldsymbol{Z}_1,\boldsymbol{Z}_2|\boldsymbol{\theta}^{(t)}\right)\right)p\left(\boldsymbol{Z}_1,\boldsymbol{Z}_2|\mathbf{Y},\boldsymbol{\theta}^{(t)}\right)\mathrm{d}\boldsymbol{Z}_2\mathrm{d}\boldsymbol{Z}_1\\
&\quad + \iint \log\left(p\left(\mathbf{Y},\boldsymbol{Z}_1|\boldsymbol{\theta}^{(t)}\right)\right)p\left(\boldsymbol{Z}_1,\boldsymbol{Z}_2|\mathbf{Y},\boldsymbol{\theta}^{(t)}\right)\mathrm{d}\boldsymbol{Z}_2\mathrm{d}\boldsymbol{Z}_1 = Q_{\boldsymbol{Z}_1\boldsymbol{Z}_2}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}\right) - Q_{\boldsymbol{Z}_1\boldsymbol{Z}_2}\left(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}\right) + Q_{\boldsymbol{Z}_1}\left(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}\right).
\end{aligned}
\tag{A.1}
$$

where the inequality is from the Jensen's inequality. Therefore,

$$
\begin{aligned}
&Q_{\boldsymbol{Z}_1}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}\right) - Q_{\boldsymbol{Z}_1}\left(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}\right) + l\left(\mathbf{Y}|\boldsymbol{\theta}^{(t)}\right)\\
&\quad \geq Q_{\boldsymbol{Z}_1\boldsymbol{Z}_2}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}\right) - Q_{\boldsymbol{Z}_1\boldsymbol{Z}_2}\left(\boldsymbol{\theta}^{(t)}|\boldsymbol{\theta}^{(t)}\right) + l\left(\mathbf{Y}|\boldsymbol{\theta}^{(t)}\right),
\end{aligned}
\tag{A.2}
$$

i.e.,

$$
g_{\boldsymbol{Z}_1}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}\right) \geq g_{\boldsymbol{Z}_1\boldsymbol{Z}_2}\left(\boldsymbol{\theta}|\boldsymbol{\theta}^{(t)}\right). \tag{A.3}
$$

A.2. Proof for Proposition 2

Let us define, for $j = 1, 2, \ldots, p$, the weighted covariance of $\mathbf{y}_{i,(j)}$ (first j components) in the $k(j)$th group around the mean $\boldsymbol{\mu}_{(j)}$,

$$
\mathbf{C}_j^{(t)}(\boldsymbol{\mu}) = \sum_{i=n_{k(j)-1}+1}^{n_{k(j)}} \omega_i^{(t)}\left(\mathbf{y}_{i,(j)} - \boldsymbol{\mu}_{(j)}\right)\left(\mathbf{y}_{i,(j)} - \boldsymbol{\mu}_{(j)}\right)^T, \tag{A.4}
$$

and the weighted covariance of $\mathbf{y}_{i,(j)}$ in the first $k(j)$ groups,

$$
\mathbf{R}_j^{(t)}(\boldsymbol{\mu}) = \sum_{i=1}^{n_{k(j)}} \omega_i^{(t)}\left(\mathbf{y}_{i,(j)} - \boldsymbol{\mu}_{(j)}\right)\left(\mathbf{y}_{i,(j)} - \boldsymbol{\mu}_{(j)}\right)^T. \tag{A.5}
$$

Note that

$$
\mathbf{R}_j^{(t)}(\boldsymbol{\mu}) = \sum_{k=1}^{k(j)} \left[\mathbf{C}_{p_k}^{(t)}(\boldsymbol{\mu})\right]_{(j)}, \tag{A.6}
$$

where $[\mathbf{C}_{p_k}^{(t)}(\boldsymbol{\mu})]_{(j)}$ is the upper left $j\times j$ submatrix of $\mathbf{C}_j^{(t)}(\boldsymbol{\mu})$, and

$$
\mathbf{R}_j^{(t)}(\boldsymbol{\mu}) = \mathbf{S}_j^{(t)} + \Omega_j^{(t)}\left(\boldsymbol{\mu}_{(j)} - \bar{\mathbf{y}}_j^{(t)}\right)\left(\boldsymbol{\mu}_{(j)} - \bar{\mathbf{y}}_j^{(t)}\right)^T. \tag{A.7}
$$

Substituting (29) and (A.4) into $g(\boldsymbol{\mu}, \boldsymbol{\Sigma}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)})$ gives

$$
\begin{aligned}
&g\left(\boldsymbol{\mu},\mathbf{H}|\boldsymbol{\mu}^{(t)},\boldsymbol{\Sigma}^{(t)},\nu^{(t)}\right)\\
&= \sum_{k=1}^{K}\left\{-\frac{1}{2}\mathrm{Tr}\left(\mathbf{H}_{(p_k)}^T\mathbf{C}_{p_k}^{(t)}(\boldsymbol{\mu})\mathbf{H}_{(p_k)}\right) + \frac{n_k - n_{k-1}}{2}\log\det\left(\mathbf{H}_{(p_k)}\mathbf{H}_{(p_k)}^T\right)\right\}\\
&= -\frac{1}{2}\sum_{k=1}^{K}\sum_{j=1}^{p_k}\mathbf{h}_j^T\left[\mathbf{C}_{p_k}^{(t)}(\boldsymbol{\mu})\right]_{(j)}\mathbf{h}_j + \sum_{k=1}^{K}(n_k - n_{k-1})\log\det\left(\mathbf{H}_{(p_k)}\right)\\
&= -\frac{1}{2}\sum_{j=1}^{p}\sum_{k=1}^{k(j)}\mathbf{h}_j^T\left[\mathbf{C}_{p_k}^{(t)}(\boldsymbol{\mu})\right]_{(j)}\mathbf{h}_j + \sum_{k=1}^{K}\left\{(n_k - n_{k-1})\sum_{j=1}^{p_k}\log\left(h_{j,j}\right)\right\}
\end{aligned}
$$

$$= -\frac{1}{2}\sum_{j=1}^{p}\mathbf{h}_j^T\mathbf{R}_j^{(t)}(\boldsymbol{\mu})\mathbf{h}_j + \sum_{j=1}^{p}\sum_{k=1}^{k(j)}\log\left(h_{j,j}\right)(n_k - n_{k-1})$$

$$= -\frac{1}{2}\sum_{j=1}^{p}\Omega_j^{(t)}\mathbf{h}_j^T\left(\boldsymbol{\mu}_{(j)} - \bar{\mathbf{y}}_j^{(t)}\right)\left(\boldsymbol{\mu}_{(j)} - \bar{\mathbf{y}}_j^{(t)}\right)^T\mathbf{h}_j - \frac{1}{2}\sum_{j=1}^{p}\mathbf{h}_j^T\mathbf{S}_j^{(t)}\mathbf{h}_j + \sum_{j=1}^{p}\log\left(h_{j,j}\right)n_{k(j)}$$

$$= -\frac{1}{2}\left(\boldsymbol{\mu} - \boldsymbol{\theta}^{(t)}(\mathbf{H})\right)^T\mathbf{B}^{(t)}(\mathbf{H})\left(\boldsymbol{\mu} - \boldsymbol{\theta}^{(t)}(\mathbf{H})\right) - \frac{1}{2}\sum_{j=1}^{p}\mathbf{h}_j^T\mathbf{S}_j^{(t)}\mathbf{h}_j + \sum_{j=1}^{p}\log\left(h_{j,j}\right)n_{k(j)}, \tag{A.8}$$

where the constant in $g(\boldsymbol{\mu}, \boldsymbol{\Sigma}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)})$ is ignored.

A.3. Proof for Proposition 3

Suppose the $u(\boldsymbol{\mu}, \mathbf{H}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)})$ is maximized by $\boldsymbol{\mu}^{(t+1)}$ and $\mathbf{H}^{(t+1)}$. From (30), it is obvious that

$$\boldsymbol{\mu}^{(t+1)} = \boldsymbol{\theta}^{(t)}\left(\mathbf{H}^{(t+1)}\right) = \left(\mathbf{H}^{(t+1)}\right)^{-T}\left[\left(\mathbf{h}_1^{(t+1)}\right)^T\bar{\mathbf{y}}_1^{(t)}, \ldots, \left(\mathbf{h}_p^{(t+1)}\right)^T\bar{\mathbf{y}}_p^{(t)}\right]^T. \tag{A.9}$$

The derivative of $u(\boldsymbol{\mu}, \mathbf{H}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)})$ with respect to $\mathbf{h}_j$ at $\boldsymbol{\mu}^{(t+1)}$ and $\mathbf{H}^{(t+1)}$ should be equal to $\mathbf{0}$:

$$\begin{aligned}
&\frac{\partial u\left(\boldsymbol{\mu}, \mathbf{H}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)}\right)}{\partial \mathbf{h}_j}\Big|_{\boldsymbol{\mu}^{(t+1)}, \mathbf{H}^{(t+1)}} \\
&= \left(\boldsymbol{\mu}^{(t+1)} - \boldsymbol{\theta}^{(t)}\left(\mathbf{H}^{(t+1)}\right)\right)^T\mathbf{B}^{(t)}\left(\mathbf{H}^{(t+1)}\right)\frac{\partial\boldsymbol{\theta}^{(t)}(\mathbf{H})}{\partial\mathbf{h}_j}\Big|_{\mathbf{H}^{(t+1)}} \\
&\quad - \frac{1}{2}\left(\mathbf{1}_{j\times 1}\otimes\left(\boldsymbol{\mu}^{(t+1)} - \boldsymbol{\theta}^{(t)}\left(\mathbf{H}^{(t+1)}\right)\right)\right)^T \\
&\quad \times Diag\left(\left[\frac{\partial\mathbf{B}^{(t)}(\mathbf{H})}{\partial h_{j,1}}\Big|_{\mathbf{H}^{(t+1)}}, \ldots, \frac{\partial\mathbf{B}^{(t)}(\mathbf{H})}{\partial h_{j,j}}\Big|_{\mathbf{H}^{(t+1)}}\right]\right) \\
&\quad \times\left(\mathbf{I}_{j\times j}\otimes\left(\boldsymbol{\mu}^{(t+1)} - \boldsymbol{\theta}^{(t)}\left(\mathbf{H}^{(t+1)}\right)\right)\right) \\
&\quad - \left(\mathbf{h}_j^{(t+1)}\right)^T\mathbf{S}_j^{(t)} + \left(0, \ldots, 0, 1/h_{j,j}^{(t+1)}\right)n_{k(j)} \\
&= -\left(\mathbf{h}_j^{(t+1)}\right)^T\mathbf{S}_j^{(t)} + \left(0, \ldots, 0, 1/h_{j,j}^{(t+1)}\right)n_{k(j)} \\
&= \mathbf{0},
\end{aligned} \tag{A.10}$$

which is equivalent to

$$\left(\mathbf{L}_j^{(t)}\right)^T\mathbf{h}_j^{(t+1)} = \left(\mathbf{L}_j^{(t)}\right)^{-1}\left(0, \ldots, 0, 1/h_{j,j}^{(t+1)}\right)^T n_{k(j)}. \tag{A.11}$$

Denote $\mathbf{c}_j^{(t+1)} = \left(\mathbf{L}_j^{(t)}\right)^T\mathbf{h}_j^{(t+1)}$. The equation (A.11) can be rewritten as

$$\mathbf{c}_j^{(t+1)} = \left(0, \ldots, 0, 1/c_{j,j}^{(t+1)}\right)^T n_{k(j)}. \tag{A.12}$$

Therefore,

$$\mathbf{c}_j^{(t+1)} = \left(0, \ldots, 0, n_{k(j)}^{\frac{1}{2}}\right)^T, \tag{A.13}$$

and thus, $\mathbf{h}_j^{(t+1)} = (\mathbf{L}_j^{(t)})^{-T}(0, \ldots, 0, n_{k(j)}^{\frac{1}{2}})^T$. Then the minorizing function (26) is maximized by (37) and (38).

A.4. Proof for Proposition 4

Substituting (29) into $g_2^{\text{shrink}}(\boldsymbol{\mu}, \boldsymbol{\Sigma}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)})$ gives

$$\begin{aligned}
&g_2^{\text{shrink}}\left(\boldsymbol{\mu}, \mathbf{H}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)}\right) \\
&= -\frac{1}{2}\sum_{j=1}^{p}\mathbf{h}_j^T\left\{\Omega_j^{(t)}\left(\boldsymbol{\mu}_{(j)} - \bar{\mathbf{y}}_j^{(t)}\right)\left(\boldsymbol{\mu}_{(j)} - \bar{\mathbf{y}}_j^{(t)}\right)^T + \mathbf{S}_j^{(t)}\right\}\mathbf{h}_j \\
&\quad + \sum_{j=1}^{p}n_{k(j)}\log\left(h_{j,j}\right) - \frac{1}{2}\sum_{j=1}^{p}\mathbf{h}_j^T \\
&\quad \times\left\{\alpha\frac{\nu^{(t)} - 2}{\nu^{(t)}}\left(\boldsymbol{\mu}_{(j)} - \mathbf{t}_{(j)}\right)\left(\boldsymbol{\mu}_{(j)} - \mathbf{t}_{(j)}\right)^T + \alpha\frac{\nu^{(t)} - 2}{\nu^{(t)}}\mathbf{T}_{(j)}\right\}\mathbf{h}_j \\
&\quad + \alpha\sum_{j=1}^{p}\log\left(h_{j,j}\right) \\
&= -\frac{1}{2}\sum_{j=1}^{p}\left(\Omega_j^{(t)} + \alpha\frac{\nu^{(t)} - 2}{\nu^{(t)}}\right)\mathbf{h}_j^T\left(\boldsymbol{\mu}_{(j)} - \tilde{\mathbf{y}}_j^{(t)}\right)\left(\boldsymbol{\mu}_{(j)} - \tilde{\mathbf{y}}_j^{(t)}\right)^T\mathbf{h}_j \\
&\quad - \frac{1}{2}\sum_{j=1}^{p}\mathbf{h}_j^T\tilde{\mathbf{S}}_j^{(t)}\mathbf{h}_j + \sum_{j=1}^{p}\left(n_{k(j)} + \alpha\right)\log\left(h_{j,j}\right) \\
&= -\frac{1}{2}\left(\boldsymbol{\mu} - \tilde{\boldsymbol{\theta}}^{(t)}(\mathbf{H})\right)^T\tilde{\mathbf{B}}^{(t)}(\mathbf{H})\left(\boldsymbol{\mu} - \tilde{\boldsymbol{\theta}}^{(t)}(\mathbf{H})\right) - \frac{1}{2}\sum_{j=1}^{p}\mathbf{h}_j^T\tilde{\mathbf{S}}_j^{(t)}\mathbf{h}_j \\
&\quad + \sum_{j=1}^{p}\left(n_{k(j)} + \alpha\right)\log\left(h_{j,j}\right)
\end{aligned} \tag{A.14}$$

where

$$\tilde{\boldsymbol{\theta}}^{(t)}(\mathbf{H}) = \mathbf{H}^{-T}\left(\mathbf{h}_1^T\tilde{\mathbf{y}}_1^{(t)}, \ldots, \mathbf{h}_p^T\tilde{\mathbf{y}}_p^{(t)}\right)^T, \tag{A.15}$$

$$\tilde{\mathbf{B}}^{(t)}(\mathbf{H}) = \mathbf{H}Diag(\Omega_1^{(t)} + \alpha\frac{\nu^{(t)} - 2}{\nu^{(t)}}, \ldots, \Omega_p^{(t)} + \alpha\frac{\nu^{(t)} - 2}{\nu^{(t)}})\mathbf{H}^T. \tag{A.16}$$

Following the same way in the proof for Proposition 4, we can get the maximizer of $g_2^{\text{shrink}}(\boldsymbol{\mu}, \mathbf{H}|\boldsymbol{\mu}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \nu^{(t)})$:

$$\boldsymbol{\mu}^{(t+1)} = \left(\mathbf{H}^{(t+1)}\right)^{-T}\left[\left(\mathbf{h}_1^{(t+1)}\right)^T\tilde{\mathbf{y}}_1^{(t)}, \ldots, \left(\mathbf{h}_p^{(t+1)}\right)^T\tilde{\mathbf{y}}_p^{(t)}\right]^T, \tag{A.17}$$

$$\mathbf{h}_j^{(t+1)} = \left(\tilde{\mathbf{L}}_j^{(t)}\right)^{-T}\left(0, \ldots, 0, \sqrt{n_{k(j)} + \alpha}\right)^T. \tag{A.18}$$

References

[1] V. Koivunen, N. Himayat, S.A. Kassam, Nonlinear filtering techniques for multivariate images design and robustness characterization, Signal Process. 57 (1) (1997) 81–91.

[2] R. Abrahamsson, Y. Selen, P. Stoica, Enhanced covariance matrix estimators in adaptive beamforming, in: Proceedings of the IEEE International Conference on Acoustic Speech and Signal Processing (ICASSP), 2, 2007, pp. 969–972. Hawaii.

[3] F. Rubio, X. Mestre, D.P. Palomar, Performance analysis and optimal selection of large minimum variance portfolios under estimation risk, IEEE J. Sel. Topics Signal Process. 6 (4) (2012) 337–350.

[4] Y. Wang, J. Li, P. Stoica, Spectral Analysis of Signals, The Missing Data Case, Morgan & Claypool, San Rafal, CA, 2005.

[5] S. Vigneshwaran, N. Sundararajan, P. Saratchandran, Direction of arrival (DOA) estimation under array sensor failures using a minimal resource allocation neural network, IEEE Trans. Antennas Propag. 55 (2) (2007) 334–343.

[6] E.G. Larsson, P. Stoica, High-resolution direction finding: the missing data case, IEEE Trans. Signal Process. 49 (5) (2001) 950–958.

[7] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, J. R. Stat. Soc. Series B Stat. Methodol. 39 (1) (1977) 1–38.

[8] C. Liu, Efficient ML estimation of the multivariate normal distribution from incomplete data, J. Multivar. Anal. 69 (2) (1999) 206–217.

[9] U.A. Müller, M.M. Dacorogna, O.V. Pictet, Heavy tails in high-frequency financial data, in: R. Adler, R. Feldman, M. Taqqu (Eds.), A Practical Guide to Heavy Tails: Statistics Techniques and Applications, 4th ed., Birkhäuser, Boston, MA, 1998, pp. 55–78.
[10] A.M. Zoubir, V. Koivunen, Y. Chakhchoukh, M. Muma, Robust estimation in signal processing: a tutorial-style treatment of fundamental concepts, IEEE Signal Process. Mag. 29 (4) (2012) 61–80.
[11] R.J. Little, Robust estimation of the mean and covariance matrix from data with missing values, Appl. Stat. 37 (1) (1988) 23–38.
[12] K.L. Lange, R.J. Little, J.M. Taylor, Robust statistical modeling using the t distribution, J. Am. Stat. Assoc. 84 (408) (1989) 881–896.
[13] R.J. Little, D.B. Rubin, Statistical Analysis with Missing Data, 2nd ed., John Wiley & Sons, Hoboken, N.J., 2002.
[14] R. Varadhan, C. Roland, Simple and globally convergent methods for accelerating the convergence of any EM algorithm, Scand. J. Stat. 35 (2) (2008) 335–353.
[15] C. Liu, D.B. Rubin, Y.N. Wu, Parameter expansion to accelerate EM: the PX-EM algorithm, Biometrika 85 (4) (1998) 755–770.
[16] C. Liu, D.B. Rubin, The ECME algorithm: a simple extension of EM and ECM with faster monotone convergence, Biometrika 81 (4) (1994) 633–648.
[17] C. Liu, D.B. Rubin, ML Estimation of the t distribution using EM and its extensions, ECM and ECME, Stat. Sin. 5 (1) (1995) 19–39. Hoboken, NJ.
[18] O. Ledoit, M. Wolf, A well-conditioned estimator for large-dimensional covariance matrices, J. Multivar. Anal. 88 (2) (2004) 365–411.
[19] Y. Chen, A. Wiesel, A.O. Hero, Robust shrinkage estimation of high-dimensional covariance matrices, IEEE Trans. Signal Process. 59 (9) (2011) 4097–4107.
[20] A. Wiesel, Unified framework to regularized covariance estimation in scaled Gaussian models, IEEE Trans. Signal Process. 60 (1) (2012) 29–38.
[21] Y. Sun, P. Babu, D.P. Palomar, Regularized robust estimation of mean and covariance matrix under heavy-tailed distributions, IEEE Trans. Signal Process. 63 (12) (2015) 3096–3109.
[22] N. Städler, D.J. Stekhoven, P. Bühlmann, Pattern alternating maximization algorithm for missing data in high-dimensional problems, J. Mach. Learn. Res. 15 (1) (2014) 1903–1928.
[23] M. Hyodo, N. Shutoh, T. Seo, T. Pavlenko, Estimation of the covariance matrix with two-step monotone missing data, Commun. Stat. Theory Methods 45 (7) (2016) 1910–1922.
[24] M. Zamanighomi, Z. Wang, G.B. Giannakis, Estimating high-dimensional covariance matrices with misses for Kronecker product expansion models, in: Proceedings of the IEEE International Conference on Acoustic Speech and Signal Processing (ICASSP), IEEE, Shanghai,China, 2016, pp. 2667–2671.
[25] R.F. Stambaugh, Analyzing investments whose histories differ in length, J. Financ. Econ. 45 (3) (1997) 285–331.
[26] R.B. Gramacy, J.H. Lee, R. Silva, On estimating covariances between many assets with histories of highly variable length, 2007. arXiv: 0710.5837, [stat.ME].
[27] R.A. Maronna, Robust M-estimators of multivariate location and scatter, Ann. Stat. 4 (1) (1976) 51–67.
[28] J.R. Hershey, P.A. Olsen, Approximating the kullback leibler divergence between gaussian mixture models, in: Proceedings of the IEEE International Conference on Acoustic Speech and Signal Processing (ICASSP), 4, 2007. Hawaii.
[29] T.M. Cover, J.A. Thomas, Elements of Information Theory, John Wiley & Sons, New York, 1991.
[30] Y. Sun, P. Babu, D.P. Palomar, Majorization-minimization algorithms in signal processing, communications, and machine learning, IEEE Trans. Signal Process. 65 (3) (2017) 794–816.
[31] M. Razaviyayn, M. Hong, Z.-Q. Luo, A unified convergence analysis of block successive minimization methods for nonsmooth optimization, SIAM J. Optim. 23 (2) (2013) 1126–1153.
[32] C. Liu, Bartlett'S decomposition of the posterior distribution of the covariance for normal monotone ignorable missing data, J. Multivar. Anal. 46 (2) (1993) 198–206.
[33] L.K. Chan, J. Karceski, J. Lakonishok, On portfolio optimization: forecasting covariances and choosing the risk model, Rev. Financ. Stud. 12 (5) (1999) 937–974.