Statistics and Its Interface Volume 9 (2016) 461–468

High-dimensional covariance estimation under the presence of outliers

Hsin-Cheng Huang* and Thomas C. M. Lee†,‡

This paper considers the problem of robust covariance estimation in the so-called "large p small n" setting. Its first contribution is the proposal of a novel (non-robust) high-dimensional covariance estimation method that is based on eigenvalue regularization. The method is called *Cover*, short for COVariance Eigenvalue-Regularized estimation. It is fast to execute and enjoys excellent theoretical properties for the case when p is fixed. As a second contribution, this paper modifies *Cover* by incorporating Huber's loss function into the estimation procedure. By design, the resulting method is robust to outliers and is called *RCover*. The empirical performances of *Cover* and *RCover* are tested and compared with existing methods via a sequence of numerical experiments. It is shown that, with the presence of outliers, *RCover* almost always outperforms other methods tested.

Keywords and phrases: Difference convex programming, Eigenvalue regularization, ES-Algorithm, Huber function.

1. INTRODUCTION

The estimation of covariance matrices is a fundamental problem in many multivariate methods. Examples include discriminant data analysis, longitudinal data analysis, time series analysis and spatial data analysis, just to name a few. In addition, with the availability of high volume data in various application areas such as gene arrays, brain imaging and climate problems, estimating covariance matrices in the high-dimensional context has attracted a lot of recent attention from different researchers.

In such high-dimensional settings, various regularized estimators are proposed under the assumption that the true covariance matrix is sparse. Some of these estimators depend on modified Cholesky decompositions (e.g., Rothman et al., 2010; Wu and Pourahmadi, 2003), some use penalized likelihoods (e.g., Bickel and Levina, 2008*b*; Friedman et al., 2008; Furrer and Bengtsson, 2007; Huang et al., 2006; Lam and Fan, 2009; Levina et al., 2008), and some are based on thresholding (e.g., Bickel and Levina, 2008*a*; Cai and Liu, 2011; Rothman et al., 2009; Fan et al., 2013) and penalized criteria (e.g., Rothman, 2012; Xue et al., 2012; Liu et al., 2014). For a more comprehensive review, see Fan et al. (2013) or Liu et al. (2014). While many of these estimators have been shown to enjoy excellent rates of convergence, so far little work has been done to the case when the data may be contaminated by outliers. A notable exception is the work of Chen et al. (2011), where the Gaussian assumption is relaxed and the data are modeled using a member from the class of elliptical distributions. In view of this, a main goal of this paper is to develop a robust method for high-dimensional covariance estimation with outliers. It achieves this goal by first proposing a fast (non-robust) method *Cover*, short for COVariance Eigenvalue–Regularized estimation, to perform the estimation when there is no outlier. Then it applies Huber's methodology to *Cover* to obtain a robust version of the method, termed *RCover*.

The rest of this paper is organized as follows. In Section 2 the new (non-robust) method *Cover* for covariance estimation is presented. Note that *Cover* is based on eigenvalue regularization and is very fast to compute. Then Section 3 modifies this method to handle outliers. As mentioned before, the resulting robust method is called *RCover*. Practical performances of both proposed methods are evaluated via simulation experiments in Section 4 and a real data application in Section 5. Lastly, concluding remarks are given in Section 6.

2. COVARIANCE ESTIMATION WITHOUT OUTLIERS: COVER

Consider a random sample, $\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n$, generated from $N(\boldsymbol{0}, \boldsymbol{\Sigma})$, where $\boldsymbol{\Sigma}$ is an unknown $p \times p$ positive-definite matrix. Define $\boldsymbol{Y} \equiv (\boldsymbol{Y}_1, \ldots, \boldsymbol{Y}_n)'$. Let $\boldsymbol{S} \equiv \boldsymbol{Y}'\boldsymbol{Y}/n$ denote the sample covariance matrix. In this section we first present our method for estimating $\boldsymbol{\Sigma}$ when there is no outlier. The case with outliers will be delayed to Section 3. We allow the possibility of $p \geq n$. In sequel $\|\cdot\|_1$ denotes the L_1 norm, while $\|\cdot\|_F$ denotes the Frobenius norm.

2.1 Eigenvalue regularization

Our methodology is based on penalizing the eigenvalues of the sample covariance $\boldsymbol{S}$. Suppose that $\boldsymbol{Y}$ is of rank K. Let $\tilde{\boldsymbol{U}}\tilde{\boldsymbol{D}}\tilde{\boldsymbol{V}}'$ be the singular value decomposition (SVD) of $\boldsymbol{Y}$, where $\tilde{\boldsymbol{U}}$ is an $n \times K$ matrix, $\tilde{\boldsymbol{V}}$ is a $p \times K$ matrix, and

*Supported in part by the National Science Council of Taiwan under grant NSC 100-2628-M-001-004-MY3.
†Corresponding author.
‡Supported in part by the National Science Foundation under grants 1512945 and 1513484.

$\tilde{\boldsymbol{D}} = \operatorname{diag}(\tilde{d}_1, \ldots, \tilde{d}_K)$ is a $K \times K$ matrix with $\tilde{d}_1 \geq \cdots \geq \tilde{d}_K$. Then $\boldsymbol{S} = \boldsymbol{Y}'\boldsymbol{Y}/n = \tilde{\boldsymbol{V}}\tilde{\boldsymbol{D}}^2\tilde{\boldsymbol{V}}'/n$, and consequently the eigenvalues of $\boldsymbol{S}$ are given by

$$\tilde{\lambda}_k = \begin{cases} \tilde{d}_k^2/n, & k = 1, \ldots, K, \\ 0, & k = K+1, \ldots, p. \end{cases}$$

To define our estimate for $\boldsymbol{\Sigma}$ we need to minimize the following nonconvex cost function with respect to $\boldsymbol{U}$, $\boldsymbol{V}$ and $\boldsymbol{D}$:

$$(1)\quad \left\|\tilde{\boldsymbol{U}}\tilde{\boldsymbol{D}}^2\tilde{\boldsymbol{V}}' - \boldsymbol{U}\boldsymbol{D}^2\boldsymbol{V}'\right\|_F^2 + \tau_1 \sum_{k=1}^{K-1} \min\left(\frac{|d_k^2 - d_{k+1}^2|}{\tau_2}, 1\right).$$

Here $\boldsymbol{U}$ is an $n \times K$ matrix, $\boldsymbol{V}$ is a $p \times K$ matrix, and $\boldsymbol{D} = \operatorname{diag}(d_1, \ldots, d_K)$ with $d_1 \geq \cdots \geq d_K \geq 0$. Also, $\tau_1 \geq 0$ is a tuning parameter controlling the degree of clustering for the d_k's, and $\tau_2 > 0$ is a thresholding parameter beyond which the difference between two consecutive d_k's will not be penalized further. Lastly, we require $\boldsymbol{U}'\boldsymbol{U} = \boldsymbol{V}'\boldsymbol{V} = \boldsymbol{I}_K$. Section 2.2 below develops a practical algorithm for minimizing (1). In the rest of this section we denote the joint minimizers of (1) as $\hat{\boldsymbol{U}}$, $\hat{\boldsymbol{V}}$ and $\hat{\boldsymbol{D}} = \operatorname{diag}(\hat{d}_1, \ldots, \hat{d}_K)$. Notice that the second term in (1), which plays the role of a penalty, encourages the small d_k's to have the same value.

Our (non-robust) estimate of $\boldsymbol{\Sigma}$ is defined as follows. Let J be such that $\hat{d}_1 \geq \cdots \geq \hat{d}_{J-1} > \hat{d}_J = \cdots = \hat{d}_K$. We first estimate the eigenvalues λ_k's as

$$\hat{\lambda}_k = \begin{cases} \hat{d}_k^2/n, & k = 1, \ldots, J-1, \\ (\hat{d}_J^2 + \cdots + \hat{d}_K^2)/\{n(K-J+1)\}, & k = J, \ldots, p, \end{cases}$$

making $\sum_{k=1}^p \hat{d}_k^2/n = \sum_{k=1}^p \hat{\lambda}_k$. Let $\hat{\boldsymbol{Q}}$ be an orthogonal matrix whose first K columns are the same as $\hat{\boldsymbol{V}}$, and define $\hat{\boldsymbol{\Lambda}} = \operatorname{diag}(\hat{\lambda}_1, \ldots, \hat{\lambda}_p)$. Then the proposed estimate of $\boldsymbol{\Sigma}$ is given by

$$(2)\qquad \hat{\boldsymbol{\Sigma}} = \hat{\boldsymbol{Q}}\hat{\boldsymbol{\Lambda}}\hat{\boldsymbol{Q}}'.$$

Since the second term of (1) penalizes the d_k's, one can see that the above estimate $\hat{\boldsymbol{\Sigma}}$ is an eigenvalue-regularized estimate.

The motivation of using (1) is as follows. From Theorem 1 (see below) one can see that (1) encourages each of the estimated eigenvalues to be pulled towards one of $J \ll p$ possible distinct values (J and these distinct values are unknown and will be estimated). This process can be seen as a multi-level shrinkage estimation, with each estimate being shrunk towards a value out of a set of J possible values, as opposed to being shrunk to zero which most shrinkage methods do. Since shrinkage is, when applied correctly, known to provide an excellent bias-variance trade-off, it is reasonable to expect that (1) will lead to improved estimates for $\boldsymbol{\Sigma}$. This multi-level shrinkage method has been applied successfully for regression coefficient regularization in Shen and Huang (2010) and Shen et al. (2012). Another advantage of using (1) is that, it leads to a very fast algorithm, as to be described next.

The tuning parameters τ_1 and τ_2 can be selected using M-fold cross-validation (CV). Specifically, the index set $\{1, \ldots, n\}$ is first partitioned into M parts, $A_1, \ldots, A_M$, of roughly the same size. For each $m = 1, \ldots, M$, let $\tilde{\boldsymbol{\Sigma}}^{(-m)}$ be a generic estimate of $\boldsymbol{\Sigma}$ based on the data $\{\boldsymbol{Y}_i : i \notin A_m\}$. Then the tuning parameters can be selected by minimizing either

$$(3)\qquad \sum_{m=1}^M \left\| \frac{1}{|A_m|} \sum_{i \in A_m} \boldsymbol{Y}_i'\boldsymbol{Y}_i - \tilde{\boldsymbol{\Sigma}}^{(-m)} \right\|_F^2,$$

in terms of the Frobenius loss, or

$$\sum_{m=1}^M \left\{ \sum_{i \in A_m} \left(\log\left|\tilde{\boldsymbol{\Sigma}}^{(-m)}\right| + \boldsymbol{Y}_i'\left(\tilde{\boldsymbol{\Sigma}}^{(-m)}\right)^{-1}\boldsymbol{Y}_i \right) \right\},$$

in terms of the Kullback-Liebler loss.

2.2 A fast algorithm for minimizing (1)

The joint minimizers of (1), denoted as $\hat{\boldsymbol{U}}$, $\hat{\boldsymbol{V}}$ and $\hat{\boldsymbol{D}} = \operatorname{diag}(\hat{d}_1, \ldots, \hat{d}_K)$, generally have no closed form expressions. This subsection develops a fast algorithm for computing these minimizers. Without loss of generality, we assume $n \geq p$. Otherwise, we can consider the SVD of $\boldsymbol{Y}'$ and obtain an equivalent problem of (1).

First we show that $\hat{\boldsymbol{U}} = \tilde{\boldsymbol{U}}$ and $\hat{\boldsymbol{V}} = \tilde{\boldsymbol{V}}$, where $\tilde{\boldsymbol{U}}\tilde{\boldsymbol{D}}\tilde{\boldsymbol{V}}'$ is the SVD of $\boldsymbol{Y}$. By direct calculations

$$\begin{aligned} &\left\|\tilde{\boldsymbol{U}}\tilde{\boldsymbol{D}}^2\tilde{\boldsymbol{V}}' - \boldsymbol{U}\boldsymbol{D}^2\boldsymbol{V}'\right\|_F^2 \\ &\quad = \operatorname{tr}\left\{\left(\tilde{\boldsymbol{U}}\tilde{\boldsymbol{D}}^2\tilde{\boldsymbol{V}}' - \boldsymbol{U}\boldsymbol{D}^2\boldsymbol{V}'\right)'\left(\tilde{\boldsymbol{U}}\tilde{\boldsymbol{D}}^2\tilde{\boldsymbol{V}}' - \boldsymbol{U}\boldsymbol{D}^2\boldsymbol{V}'\right)\right\} \\ &\quad = \operatorname{tr}\left(\tilde{\boldsymbol{D}}^4\right) + \operatorname{tr}\left(\boldsymbol{D}^4\right) - 2\operatorname{tr}\left\{\left(\tilde{\boldsymbol{U}}\tilde{\boldsymbol{D}}^2\tilde{\boldsymbol{V}}'\right)'\left(\boldsymbol{U}\boldsymbol{D}^2\boldsymbol{V}'\right)\right\} \\ &\quad \geq \operatorname{tr}\left(\tilde{\boldsymbol{D}}^4\right) + \operatorname{tr}\left(\boldsymbol{D}^4\right) - 2\operatorname{tr}\left(\tilde{\boldsymbol{D}}^2\boldsymbol{D}^2\right), \end{aligned}$$

where the last inequality follows from von Neumann's trace inequality (von Neumann, 1937). So $\boldsymbol{U} = \tilde{\boldsymbol{U}}$ and $\boldsymbol{V} = \tilde{\boldsymbol{V}}$ jointly minimize $\left\|\tilde{\boldsymbol{U}}\tilde{\boldsymbol{D}}^2\tilde{\boldsymbol{V}}' - \boldsymbol{U}\boldsymbol{D}^2\boldsymbol{V}'\right\|_F^2$. Now we remain to derive an algorithm for computing $\hat{\boldsymbol{D}}$.

Let $\boldsymbol{\delta}^* = (\delta_1^*, \ldots, \delta_K^*)' \equiv \boldsymbol{W}\boldsymbol{\delta}$, where $\boldsymbol{\delta} = (d_1^2, \ldots, d_K^2)'$ and

$$(4)\qquad \boldsymbol{W} \equiv \begin{pmatrix} 1 & -1 & 0 & \cdots & 0 \\ 0 & 1 & -1 & \cdots & \vdots \\ 0 & 0 & 1 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & -1 \\ 0 & \cdots & 0 & 0 & 1 \end{pmatrix}_{K \times K}.$$

Setting $\boldsymbol{U} = \tilde{\boldsymbol{U}}$ and $\boldsymbol{V} = \tilde{\boldsymbol{V}}$, we have

$$\left\|\tilde{\boldsymbol{U}}\tilde{\boldsymbol{D}}^2\tilde{\boldsymbol{V}}' - \tilde{\boldsymbol{U}}\boldsymbol{D}^2\tilde{\boldsymbol{V}}'\right\|_F^2 = \left\|\tilde{\boldsymbol{U}}\tilde{\boldsymbol{D}}^2\tilde{\boldsymbol{V}}' - \tilde{\boldsymbol{U}}\operatorname{diag}(\boldsymbol{\delta})\tilde{\boldsymbol{V}}'\right\|_F^2$$

$$= \big\|\tilde{\boldsymbol{D}}^2 - \mathrm{diag}(\boldsymbol{\delta})\big\|_F^2$$
$$= \|\tilde{\boldsymbol{\delta}} - \boldsymbol{W}^{-1}\boldsymbol{\delta}^*\|^2.$$

where $\tilde{\boldsymbol{\delta}} = (\tilde{d}_1^2, \ldots, \tilde{d}_K^2)'$. Then (1) can be rewritten as:

$$\Gamma(\boldsymbol{\delta}^*) = \|\tilde{\boldsymbol{\delta}} - \boldsymbol{W}^{-1}\boldsymbol{\delta}^*\|^2 + \frac{\tau_1}{\tau_2}\sum_{k=1}^{K-1}\min(|\delta_k^*|, \tau_2), \tag{5}$$

and our goal now is to minimize (5) with respect to $\boldsymbol{\delta}^*$. This is a nonconvex optimization which can be solved using difference convex (DC) programming (An and Tao, 1997) via a sequence of convex approximations. Basically, the idea is to decompose the nonconvex cost function (5) into a difference of two convex functions $\Gamma(\boldsymbol{\delta}^*) = \Gamma_1(\boldsymbol{\delta}^*) - \Gamma_2(\boldsymbol{\delta}^*)$, where

$$\Gamma_1(\boldsymbol{\delta}^*) = \|\tilde{\boldsymbol{\delta}} - \boldsymbol{W}^{-1}\boldsymbol{\delta}^*\|_F^2 + \frac{\tau_1}{\tau_2}\sum_{k=1}^{K-1}|\delta_k^*| \quad \text{and} \tag{6}$$
$$\Gamma_2(\boldsymbol{\delta}^*) = \frac{\tau_1}{\tau_2}\sum_{k=1}^{K-1}\max(|\delta_k^*| - \tau_2, 0).$$

Starting with an initial estimate, $\hat{\boldsymbol{\delta}}^{*(0)}$, one successively obtains an improved estimate $\hat{\boldsymbol{\delta}}^{*(m)}$ of (5) with $\min(|\delta_k^*| - \tau_2, 0)$ in (6) replaced by its affine minorization $(|\delta_k^*| - \tau_2)I\big(|\hat{\delta}_k^{*(m-1)}| > \tau_2\big)$, resulting in an upper convex approximation:

(7)
$$\Gamma^{(m)}(\boldsymbol{\delta}^*) = \Gamma_1(\boldsymbol{\delta}^*) - \frac{\tau_1}{\tau_2}\sum_{k=1}^{K-1}(|\delta_k^*| - \tau_2)I\big(|\hat{\delta}_k^{*(m-1)}| \ge \tau_2\big)$$
$$= \|\tilde{\boldsymbol{\delta}} - \boldsymbol{W}^{-1}\boldsymbol{\delta}^*\|^2 + \frac{\tau_1}{\tau_2}\sum_{k=1}^{K-1}|\delta_k^*|I\big(|\hat{\delta}_k^{*(m-1)}| < \tau_2\big)$$
$$+ \tau_1\sum_{k=1}^{K-1}I\big(|\hat{\delta}_k^{*(m-1)}| \ge \tau_2\big),$$

for $m \in \mathbb{N} \equiv \{1, 2, \ldots\}$, where the last term of (7) does not depend on $\boldsymbol{\delta}^*$. This DC algorithm has an attractive property that $\Gamma\big(\hat{\boldsymbol{\delta}}^{*(m+1)}\big) \le \Gamma\big(\hat{\boldsymbol{\delta}}^{*(m)}\big)$ for $m \in \{0, 1, \ldots\}$. In fact, it converges in a finite number of steps, say M steps, where $\hat{\boldsymbol{\delta}}^{*(M)} = \hat{\boldsymbol{\delta}}^{*(M+1)}$, which happens when

$$\sum_{k=1}^{K-1}I\big(|\hat{\delta}_k^{*(M-1)}| \ge \tau_2\big)I\big(|\hat{\delta}_k^{*(M)}| \ge \tau_2\big) = K - 1. \tag{8}$$

We denote the converged solution of $\boldsymbol{\delta}^*$ by $\hat{\boldsymbol{\delta}}^* = \hat{\boldsymbol{\delta}}^{*(M)}$. The final solution of $\boldsymbol{\delta}$ is then given by $\hat{\boldsymbol{\delta}} = \boldsymbol{W}^{-1}\hat{\boldsymbol{\delta}}^*$, from which $\hat{\boldsymbol{D}}$ can be easily constructed.

To solve (7), we apply the coordinate descent method. Let $\boldsymbol{\omega}_k$ be the kth column vector of $\boldsymbol{W}^{-1}$. The updating formulae for $\hat{\boldsymbol{\delta}}^{*(m)} = \big(\hat{\delta}_1^{*(m)}, \ldots, \hat{\delta}_K^{*(m)}\big)'$ are given by the following, which have simple closed form expressions:

$$\hat{\delta}_k^{*(m)} = \frac{1}{2\|\boldsymbol{\omega}_k\|^2}\max\Big\{2(\tilde{\boldsymbol{\delta}} - \tilde{\boldsymbol{\delta}}_{-k}^{(m-1)})'\boldsymbol{\omega}_k - \frac{\tau_1}{\tau_2}I\big(|\hat{\delta}_k^{*(m-1)}| < \tau_2\big), 0\Big\}; \quad k = 1, \ldots, K-1, \tag{9}$$
$$\hat{\delta}_K^{*(m)} = \frac{1}{\|\boldsymbol{\omega}_K\|^2}\max\{(\tilde{\boldsymbol{\delta}} - \tilde{\boldsymbol{\delta}}_{-K}^{(m-1)})'\boldsymbol{\omega}_K, 0\},$$

where $\tilde{\boldsymbol{\delta}}_{-k}^{(m-1)} = \sum_{j:j\ne k}\hat{\delta}_j^{*(m-1)}\boldsymbol{\omega}_j$ for $k = 1, \ldots, K$.

The major steps are summarized in Algorithm 1. We note that this algorithm is extremely fast, as it only requires to perform one SVD (in Step 1), and the iterations in Step 4 have closed form expressions. Note also that this algorithm is guaranteed to converge, because the DC algorithm converges in a finite number of steps with $\hat{\boldsymbol{\delta}}^{*(M)} = \hat{\boldsymbol{\delta}}^{*(M+1)}$ (i.e., (8) is satisfied), and the coordinate descent iterations of (9) is known to converge to the Lasso solution $\hat{\boldsymbol{\delta}}^{*(m)}$ of (6). Nevertheless, there is no guarantee that the converged value is a global minimum; i.e., this algorithm may converge to a local minimizer of (5).

Algorithm 1 The *Cover* Algorithm

Description: Given data $\boldsymbol{Y}$, compute the *Cover* estimate $\hat{\boldsymbol{\Sigma}}$ for the covariance matrix $\boldsymbol{\Sigma}$ of $\boldsymbol{Y}$.

1: Compute the SVD $\tilde{\boldsymbol{U}}\tilde{\boldsymbol{D}}\tilde{\boldsymbol{V}}'$ of $\boldsymbol{Y}$.
2: Set $\hat{\boldsymbol{U}} = \tilde{\boldsymbol{U}}$ and $\hat{\boldsymbol{V}} = \tilde{\boldsymbol{V}}$.
3: Obtain an initial estimate $\hat{\boldsymbol{\delta}}^{*(0)}$ for $\hat{\boldsymbol{\delta}}^*$ and set $m = 1$; see REMARK below.
4: Iterate (9) for a given m until convergence. Denote the converged solution as $\hat{\boldsymbol{\delta}}^{*(m)}$.
5: If $\hat{\boldsymbol{\delta}}^{*(m)} \ne \hat{\boldsymbol{\delta}}^{*(m-1)}$, go to Step 4 with m replaced by $m+1$, otherwise denote the converged solution as $\hat{\boldsymbol{\delta}}^*$.
6: Calculate $\hat{\boldsymbol{D}}$ as $\hat{\boldsymbol{\delta}} = \boldsymbol{W}^{-1}\hat{\boldsymbol{\delta}}^*$, with $\boldsymbol{W}$ defined in (4).
7: With $\hat{\boldsymbol{U}}$, $\hat{\boldsymbol{V}}$ and $\hat{\boldsymbol{D}}$, compute the *Cover* estimate for $\boldsymbol{\Sigma}$ using (2).

REMARK: in Step 3 a good initial estimate $\hat{\boldsymbol{\delta}}^{*(0)}$ can be chosen to be the estimate obtained from the nearby tuning parameter values, and $\hat{\boldsymbol{\delta}}^* = (\boldsymbol{0}, \mathrm{tr}(\boldsymbol{S})/p)'$ when both τ_1/τ_2 and τ_2 are large enough.

2.3 Theoretical properties

Suppose that p is fixed and $\boldsymbol{\Sigma}$ have J distinct eigenvalues $\zeta_1 > \cdots > \zeta_J$ with multiplicities $m_1, \ldots, m_J$. That is, m_1 eigenvalues of $\boldsymbol{\Sigma}$ share the same value ζ_1, m_2 eigenvalues of $\boldsymbol{\Sigma}$ share the same value ζ_2, and so on. Then by Anderson (1963) or Muirhead (1982, Theorem 9.3.1), the maximum likelihood (ML) estimate of ζ_j is

$$\hat{\zeta}_j^{(\mathrm{ml})} = \frac{1}{m_j}\sum_{k\in\Omega_j}\tilde{\lambda}_k; \quad j = 1, \ldots, J, \tag{10}$$

where $\Omega_j = \{k \in \mathbb{N} : m_1 + \cdots + m_{j-1} < k \le m_1 + \cdots + m_j\}$. Let $\lambda_1 \ge \cdots \ge \lambda_p$ be the eigenvalues of $\boldsymbol{\Sigma}$ and write $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_p)'$. If we know which eigenvalues are equal in advance, then the ML estimate of $\boldsymbol{\lambda}$ under the above assumption is

$$\hat{\boldsymbol{\lambda}}^{(\mathrm{ml})} = (\underbrace{\hat{\zeta}_1^{(\mathrm{ml})}, \ldots, \hat{\zeta}_1^{(\mathrm{ml})}}_{m_1}, \underbrace{\hat{\zeta}_2^{(\mathrm{ml})}, \ldots, \hat{\zeta}_2^{(\mathrm{ml})}}_{m_2}, \ldots, \underbrace{\hat{\zeta}_J^{(\mathrm{ml})}, \ldots, \hat{\zeta}_J^{(\mathrm{ml})}}_{m_J})$$

and the ML estimate of $n\boldsymbol{W}\boldsymbol{\lambda}$ is $\hat{\boldsymbol{\delta}}^{*(\mathrm{ml})} = n\boldsymbol{W}\hat{\boldsymbol{\lambda}}^{(\mathrm{ml})}$. The following theorem shows that our estimate $\hat{\boldsymbol{\lambda}} = (\hat{\lambda}_1, \ldots, \hat{\lambda}_p)'$ achieves the oracle property as if the clusters of eigenvalues are known in advance.

Theorem 1. *Consider the cost function (1). Assume that p is fixed and there are J unknown distinct eigenvalues with unknown multiplicities, $m_1, \ldots, m_J$. In addition, assume that with probability tending to 1, there is only one local minimizer of (5) as $n \to \infty$. Let $\gamma = \min\{(\zeta_2 - \zeta_1), \ldots, (\zeta_J - \zeta_{J-1})\}$, and further suppose that $\gamma > \tau_2/n > 0$ and $\tau_1/\tau_2 > 0$. Then*

$$P\big(\hat{\boldsymbol{\lambda}} = \hat{\boldsymbol{\lambda}}^{(\mathrm{ml})}\big) \to 1 \quad as \quad n \to \infty.$$

Proof. Let $\boldsymbol{\omega}_k$ be the k-th column of $\boldsymbol{W}^{-1}$, for $k = 1, \ldots, p$. Then $\boldsymbol{\delta}^*$ is a local minimizer of (5) if

$$\boldsymbol{\omega}_k'\big(\tilde{\boldsymbol{\delta}} - \boldsymbol{W}^{-1}\boldsymbol{\delta}^*\big) + \frac{\tau_1}{\tau_2} s_k = 0; \quad k = 1, \ldots, p, \tag{11}$$

where $s_k \in [-1, 1]$ if $|\delta_k^*| = 0$, $s_k = \mathrm{sign}(\delta_k^*)$ if $0 < |\delta_k^*| < \tau_2$, and $s_k = 0$ if $|\delta_k^*| > \tau_2$. Clearly, the proposed estimate $\hat{\boldsymbol{\delta}}^*$ obtained iteratively from (7) satisfies (11) after convergence. It remains to show that $\hat{\boldsymbol{\lambda}}^{*(\mathrm{ml})}$ satisfies (11). From (10), we obtain that $\hat{\boldsymbol{\delta}}^{*(\mathrm{ml})} = \big(\hat{\delta}_1^{*(\mathrm{ml})}, \ldots, \hat{\delta}_p^{*(\mathrm{ml})}\big)' = n\boldsymbol{W}\hat{\boldsymbol{\lambda}}^{(\mathrm{ml})}$ minimizes $\|\tilde{\boldsymbol{\delta}} - \boldsymbol{W}^{-1}\boldsymbol{\delta}^*\|^2$ over all $\boldsymbol{\delta}^* \in \mathcal{D}$, where

$$\mathcal{D} = \big\{(\underbrace{0, \ldots, 0}_{m_1 - 1}, a_1, \underbrace{0, \ldots, 0}_{m_2 - 1}, a_2, \ldots, \underbrace{0, \ldots, 0}_{m_{J-1} - 1}, a_{J-1}, 0, \ldots, 0, a_J)' : a_1, \ldots, a_J \in \mathbb{R}\big\}.$$

Hence $\boldsymbol{\omega}_k'\big(\tilde{\boldsymbol{\delta}} - \boldsymbol{W}^{-1}\hat{\boldsymbol{\delta}}^{*(\mathrm{ml})}\big) = 0$ for those k such that $\big|\hat{\delta}_k^{*(\mathrm{ml})}\big| > 0$. In addition, since $(\tilde{\lambda}_1, \ldots, \tilde{\lambda}_p)' \to \boldsymbol{\lambda}$ as $n \to \infty$, we have $\boldsymbol{\omega}_k'\big(\tilde{\boldsymbol{\delta}} - \boldsymbol{W}^{-1}\hat{\boldsymbol{\delta}}^{*(\mathrm{ml})}\big) \to 0$ for $k = 1, \ldots, p$, and $P\big(\big|\hat{\delta}_k^{*(\mathrm{ml})}\big| > \tau_2\big) \to 1$ for those k such that $|\delta_k^*| > 0$. Thus $\hat{\boldsymbol{\delta}}^{*(\mathrm{ml})}$ satisfies (11). This completes the proof. □

3. COVARIANCE ESTIMATION WITH OUTLIERS: RCOVER

This section extends the above methodology to situations where outliers are present. The corresponding estimate for $\boldsymbol{\Sigma}$ is still given by the expression (2), except now that $\hat{\boldsymbol{U}}$, $\hat{\boldsymbol{V}}$ and $\hat{\boldsymbol{D}}$ are defined differently.

3.1 A robust criterion

To be precise, $\hat{\boldsymbol{U}}$, $\hat{\boldsymbol{V}}$ and $\hat{\boldsymbol{D}}$ are defined as the joint minimizers of the following cost function:

$$\rho\big(\tilde{\boldsymbol{U}}\tilde{\boldsymbol{D}}^2\tilde{\boldsymbol{V}}' - \boldsymbol{U}\boldsymbol{D}^2\boldsymbol{V}'\big) + \tau_1 \sum_{k=1}^{K-1} \min\left(\frac{|d_k^2 - d_{k+1}^2|}{\tau_2}, 1\right), \tag{12}$$

where the L_2 based norm $\|\cdot\|_F^2$ in (1) is replaced by

$$\rho(\boldsymbol{M}) = \sum_{i=1}^{n}\sum_{k=1}^{p} \big\{m_{ik}^2 I(|m_{ik}| \le c_k) + c_k(2|m_{ik}| - c_k) I(|m_{ik}| > c_k)\big\}.$$

Here m_{ik} is the (i,k)th entry of the matrix $\boldsymbol{M}$, and $c_k > 0$ for $k = 1, \ldots, p$ are pre-chosen cut-off constants. The function $\rho(\cdot)$ can be seen as a matrix version of the Huber function, which is widely used to downweigh the effects of outliers (e.g., Huber, 1981). Notice that when $c_1 = \cdots = c_p = +\infty$, the cost function (12) reduces to (1).

With a slight abuse of notation, we denote by $\hat{\boldsymbol{U}}$, $\hat{\boldsymbol{V}}$ and $\hat{\boldsymbol{D}} = \mathrm{diag}(\hat{d}_1, \ldots, \hat{d}_K)$ as, respectively, the robust estimates of $\boldsymbol{U}$, $\boldsymbol{V}$ and $\boldsymbol{D}$ obtained by minimizing (12). With these the robust estimate for $\boldsymbol{\Sigma}$ can be calculated using (2).

3.2 A fast algorithm for minimizing (12)

The nonlinear nature of $\rho(\cdot)$ makes the minimization of (12) a non-trivial task. In the context of robust nonparametric regression, Oh et al. (2007) develop a so-called ES-algorithm to handle similar minimization problems. Following the idea of this ES-algorithm, we propose the following algorithm for our robust covariance estimation problem.

Suppose we are given initial estimates $\hat{\boldsymbol{U}}^{(0)}$, $\hat{\boldsymbol{V}}^{(0)}$ and $\hat{\boldsymbol{D}}^{(0)}$ of $\boldsymbol{U}$, $\boldsymbol{V}$ and $\boldsymbol{D}$, respectively. Then for $m = 0, 1, \ldots$, iterate the following steps until convergence.

1. Calculate $\hat{\boldsymbol{Y}}^{(m)} = \hat{\boldsymbol{U}}^{(m)}\hat{\boldsymbol{D}}^{(m)}\hat{\boldsymbol{V}}'^{(m)}$.
2. Evaluate $\tilde{\boldsymbol{Y}}^{(m)} = \hat{\boldsymbol{Y}}^{(m)} + \frac{1}{2}\eta\{\boldsymbol{Y} - \hat{\boldsymbol{Y}}^{(m)}\}$, where $\eta(\cdot)$ is the (elementwise) derivative of $\rho(\cdot)$.
3. Compute the SVD of $\tilde{\boldsymbol{Y}}^{(m)} = \tilde{\boldsymbol{U}}^{(m)}\tilde{\boldsymbol{D}}^{(m)}\tilde{\boldsymbol{V}}'^{(m)}$.
4. Obtain the next iterative estimates $\hat{\boldsymbol{U}}^{(m+1)}$, $\hat{\boldsymbol{V}}^{(m+1)}$ and $\hat{\boldsymbol{D}}^{(m+1)}$ by minimizing

$$\|\tilde{\boldsymbol{U}}^{(m)}\big(\tilde{\boldsymbol{D}}^{(m)}\big)^2\tilde{\boldsymbol{V}}'^{(m)} - \boldsymbol{U}\boldsymbol{D}^2\boldsymbol{V}'\|_F^2 + \tau_1 \sum_{k=1}^{K-1} \min\left(\frac{|d_k^2 - d_{k+1}^2|}{\tau_2}, 1\right),$$

which can be done efficiently by Algorithm 1.

This *RCover* algorithm essentially replaces the nonlinear minimization induced by $\rho(\cdot)$ in (12) by a sequence of L_2-type minimizations listed in Step 4 above. As the iteration continues the effect of outliers is gradually downweighed through the application of $\eta(\cdot)$ in Step 2.

The tuning parameters τ_1 and τ_2 can be selected by M-fold CV as similar to (3). Rather than applying (3), we consider a robust version given by

$$\sum_{m=1}^{M} \rho^*\left(\frac{1}{|A_m|}\sum_{i \in A_m} \boldsymbol{Y}_i'\boldsymbol{Y}_i - \hat{\boldsymbol{\Sigma}}_{\tau_1,\tau_2}^{(-m)}\right), \tag{13}$$

where $\hat{\boldsymbol{\Sigma}}_{\tau_1,\tau_2}^{(-m)}$ is the *RCover* estimate based on the data $\{\boldsymbol{Y}_i : i \notin A_m\}$,

Table 1. Simulation results using the spectral loss in Example 1 based on 200 replications. Numbers in parentheses are standard errors

n	r	*ML*		*GLasso*		*PDSCE*		*Cover*		*RCover*	
50	0%	5.35	(0.00)	2.60	(0.41)	3.87	(0.00)	**1.99**	(0.00)	**1.99**	(0.00)
50	5%	58.43	(0.03)	31.24	(4.73)	52.16	(0.04)	1.67	(0.00)	**1.63**	(0.00)
50	10%	70.31	(0.03)	28.13	(5.02)	63.65	(0.04)	3.22	(0.00)	**1.44**	(0.00)
200	0%	2.27	(0.00)	**1.12**	(0.12)	1.19	(0.00)	1.45	(0.00)	1.48	(0.00)
200	5%	20.45	(0.01)	7.08	(1.04)	16.72	(0.01)	2.04	(0.00)	**1.55**	(0.00)
200	10%	25.23	(0.01)	8.34	(0.99)	20.16	(0.01)	3.16	(0.00)	**2.32**	(0.00)
500	0%	1.35	(0.00)	**0.71**	(0.05)	0.76	(0.00)	1.05	(0.00)	1.06	(0.00)
500	5%	10.91	(0.00)	4.06	(0.36)	7.97	(0.00)	2.71	(0.00)	**2.58**	(0.00)
500	10%	14.27	(0.00)	5.49	(0.34)	10.30	(0.00)	3.63	(0.00)	**3.26**	(0.00)

$$\rho^*(\boldsymbol{M}) = \sum_{j=1}^{p}\sum_{k=1}^{p}\big\{m_{jk}^2 I(|m_{jk}| \le c_{jk}) + c_{jk}(2|m_{jk}| - c_{jk})I(|m_{jk}| > c_{jk})\big\},$$

m_{jk} is the (j,k)th entry of the matrix $\boldsymbol{M}$, and $c_{jk} > 0$ for $1 \le j,k \le p$ are pre-chosen cut-off constants.

4. SIMULATION EXPERIMENTS

Different numerical experiments were conducted to evaluate the practical performances of the proposed methods. Let σ_{ij} be the (i,j)th entry of $\boldsymbol{\Sigma}$. We generate data $\boldsymbol{Y}_1,\ldots,\boldsymbol{Y}_n$ from $N(\boldsymbol{0},\boldsymbol{\Sigma})$, and consider five examples for $\boldsymbol{\Sigma}$:

1. $\sigma_{ij} = 0.5^{|i-j|}$; $i,j = 1,\ldots,p$, where $\boldsymbol{\Sigma}^{-1}$ is sparse and the eigenvalues of $\boldsymbol{\Sigma}$ are not clustered at some values.
2. $\theta_{ij} = 0.5^{|i-j|}$; $i,j = 1,\ldots,p$, where $\boldsymbol{\Sigma}$ is sparse and the eigenvalues of $\boldsymbol{\Sigma}$ are not clustered at some values.
3. $\boldsymbol{\Sigma} = \boldsymbol{I}_p$.
4. $\boldsymbol{\Sigma} = \boldsymbol{\Lambda} = \mathrm{diag}(\underbrace{9,\ldots,9}_{4},\underbrace{5,\ldots,5}_{4},\underbrace{3,\ldots,3}_{4},\underbrace{1,\ldots,1}_{p-12})$.
5. $\boldsymbol{\Sigma} = \boldsymbol{V}\boldsymbol{\Lambda}\boldsymbol{V}'$, where $\boldsymbol{\Lambda}$ is given above, and $\boldsymbol{V}$ is a randomly generated diagonal matrix.

For each example, we replace $100r\%$ of $\boldsymbol{Y}_1,\ldots,\boldsymbol{Y}_n$ by outliers generated from $N(\boldsymbol{0},\boldsymbol{\Sigma}+v^2\boldsymbol{I})$, where $v^2 = 25\,\mathrm{tr}(\boldsymbol{\Sigma})/p$. We consider $p = 100$, and nine different combinations of $n \in \{50,200,500\}$ and $r \in \{0,0.05,0.1\}$. The following five methods are considered:

1. *ML*: the maximum likelihood,
2. *GLasso*: the graphical Lasso of Friedman et al. (2008),
3. *PDSCE*: the positive definite sparse covariance estimators of Rothman (2012),
4. *Cover*: the COVariance Eigenvalue–Regularized estimate proposed in Section 2, and
5. *RCover*: the robust version of *Cover* developed in Section 3.

Three loss functions are used to evaluate the quality of any covariance estimate. They are the spectral loss, defined as the square root of the maximum eigenvalue of $(\tilde{\boldsymbol{\Sigma}}-\boldsymbol{\Sigma})'(\tilde{\boldsymbol{\Sigma}}-\boldsymbol{\Sigma})$, the squared Frobenius loss $\|\tilde{\boldsymbol{\Sigma}}-\boldsymbol{\Sigma}\|_F^2$, and the matrix ℓ_1 loss $\max_{j=1,\ldots,p}\sum_{i=1}^{p}|\tilde{\sigma}_{ij}-\sigma_{ij}|$, where $\tilde{\boldsymbol{\Sigma}} = (\tilde{\sigma}_{ij})$ is a generic estimate of $\boldsymbol{\Sigma}$.

In each simulated example, we consider 7 tuning parameters of $\tau_2 \in \{1,2,4,8,16,32,\infty\}$ in combination with 200 tuning parameters of τ_1/τ_2 equally spaced in the log scale for the proposed methods (*Cover* and *RCover*). Similarly, we consider 200 tuning parameters equally spaced in the log scale for *GLasso*. Both *Glasso* and *PDSCE* are implemented using the R package "glasso" and "PDSCE", respectively, where the tuning parameter of *PDSCE* is selected according to the package's default setup. The tuning parameters of *GLasso* and *Cover* are selected by 5-fold CV of (3). For *RCover*, we select $c_j = 1.345\,\hat{\sigma}_{jj}$, where $\hat{\sigma}_{jj} = \mathrm{median}\{|Y_{ij} - \mathrm{med}\{Y_{1j},\ldots,Y_{nj}\}| : i = 1,\ldots,n\}$, and Y_{ij} is the jth element of $\boldsymbol{Y}_i$, and the tuning parameters of *RCover* are selected by 5-fold CV of (13) with $c_{jk} = \hat{\sigma}_{jj}\hat{\sigma}_{kk}$.

As the method of Chen et al. (2011) pre-sets the trace of their covariance estimate to be p, it performs very well for Example 3 when the trace is exactly p, but poorly for Example 4 when the trace of the covariance matrix is away from p. Therefore this method is not included in the simulation as it would make the comparison less meaningful.

The results for the spectral loss based on 200 simulation replicates are shown in Tables 1 to 5. The lowest value for each combination of experimental settings is boldfaced. From these tables, one can see that, when there is no outlier, *GLasso*, *PDSCE* or *Cover* produces the best results depending on the situations, and *RCover* is not far behind. However, when outliers are present, the performances of *GLasso* and *PDSCE* deteriorate substantially, while *RCover* shows resistance to the outlier effect and gives overall the best results. Results for the squared Frobenius loss and the matrix ℓ_1 loss are similar and hence are omitted.

5. AN APPLICATION TO SPEECH SIGNAL CLASSIFICATION

We apply the proposed *Cover* method to a Parkinson's disease dataset to discriminate between healthy people and

Table 2. Similar to Table 1 but for Example 2

n	r	*ML*		*GLasso*		*PDSCE*		*Cover*		*RCover*	
50	0%	8.22	(0.00)	3.19	(0.48)	5.99	(0.00)	**1.36**	(0.00)	1.37	(0.00)
50	5%	96.77	(0.05)	50.38	(7.97)	85.94	(0.07)	2.98	(0.00)	**1.57**	(0.00)
50	10%	116.43	(0.05)	45.72	(7.87)	105.38	(0.07)	5.47	(0.00)	**2.25**	(0.00)
200	0%	3.39	(0.00)	1.53	(0.17)	1.71	(0.00)	1.56	(0.00)	**1.35**	(0.00)
200	5%	33.95	(0.01)	11.10	(1.64)	27.80	(0.01)	3.40	(0.00)	**2.31**	(0.00)
200	10%	41.95	(0.01)	13.54	(1.53)	33.48	(0.01)	5.45	(0.00)	**3.76**	(0.00)
500	0%	1.99	(0.00)	1.12	(0.10)	**0.72**	(0.00)	1.36	(0.00)	1.32	(0.00)
500	5%	18.18	(0.01)	6.21	(0.53)	13.21	(0.00)	3.91	(0.00)	**3.26**	(0.00)
500	10%	23.60	(0.01)	9.01	(0.53)	16.98	(0.00)	5.79	(0.00)	**5.38**	(0.00)

Table 3. Similar to Table 1 but for Example 3

n	r	*ML*		*GLasso*		*PDSCE*		*Cover*		*RCover*	
50	0%	4.56	(0.00)	1.10	(0.16)	3.34	(0.00)	**0.02**	(0.00)	0.06	(0.00)
50	5%	58.36	(0.03)	29.83	(4.69)	52.22	(0.04)	1.00	(0.00)	**0.36**	(0.00)
50	10%	69.85	(0.03)	27.15	(4.68)	63.05	(0.04)	2.56	(0.00)	**0.65**	(0.00)
200	0%	1.83	(0.00)	0.50	(0.05)	0.54	(0.00)	0.02	(0.00)	**0.01**	(0.00)
200	5%	20.49	(0.01)	6.36	(0.92)	16.73	(0.01)	1.25	(0.00)	**0.49**	(0.00)
200	10%	25.17	(0.01)	8.13	(0.97)	20.03	(0.01)	2.49	(0.00)	**1.42**	(0.00)
500	0%	1.05	(0.00)	0.37	(0.03)	0.26	(0.00)	**0.02**	(0.00)	**0.02**	(0.00)
500	5%	10.96	(0.00)	3.44	(0.29)	7.94	(0.00)	1.44	(0.00)	**1.13**	(0.00)
500	10%	14.23	(0.00)	5.21	(0.36)	10.21	(0.00)	2.67	(0.00)	**2.43**	(0.00)

Table 4. Similar to Table 1 but for Example 4

n	r	*ML*		*GLasso*		*PDSCE*		*Cover*		*RCover*	
50	0%	8.33	(0.01)	4.99	(1.33)	**2.63**	(0.00)	6.48	(0.00)	6.56	(0.00)
50	5%	84.06	(0.05)	44.33	(7.05)	74.62	(0.06)	15.13	(0.00)	**6.36**	(0.00)
50	10%	100.68	(0.05)	39.67	(6.78)	90.31	(0.05)	**4.03**	(0.00)	5.60	(0.00)
200	0%	3.75	(0.00)	2.15	(0.60)	**1.21**	(0.00)	2.93	(0.00)	5.09	(0.01)
200	5%	29.54	(0.01)	9.65	(1.43)	23.67	(0.01)	4.85	(0.00)	**5.22**	(0.00)
200	10%	36.25	(0.01)	11.92	(1.42)	28.77	(0.01)	4.05	(0.00)	**3.99**	(0.00)
500	0%	2.23	(0.00)	1.15	(0.30)	**0.75**	(0.00)	2.01	(0.00)	1.90	(0.00)
500	5%	15.78	(0.00)	5.11	(0.47)	10.96	(0.01)	**3.91**	(0.00)	3.93	(0.00)
500	10%	20.51	(0.00)	7.58	(0.52)	14.10	(0.01)	**5.15**	(0.00)	5.26	(0.00)

Table 5. Similar to Table 1 but for Example 5

n	r	*ML*		*GLasso*		*PDSCE*		*Cover*		*RCover*	
50	0%	8.30	(0.01)	6.24	(0.63)	5.70	(0.00)	6.49	(0.00)	**4.82**	(0.00)
50	5%	84.90	(0.05)	44.51	(6.99)	76.74	(0.06)	**5.10**	(0.00)	6.35	(0.00)
50	10%	101.35	(0.04)	40.22	(6.92)	92.12	(0.05)	**4.06**	(0.00)	5.69	(0.00)
200	0%	3.69	(0.00)	5.13	(0.64)	6.57	(0.00)	**3.01**	(0.00)	3.44	(0.00)
200	5%	29.54	(0.01)	9.96	(1.45)	23.73	(0.01)	**4.83**	(0.00)	5.23	(0.00)
200	10%	36.35	(0.01)	11.95	(1.35)	28.96	(0.01)	4.05	(0.00)	**3.90**	(0.01)
500	0%	2.24	(0.00)	4.10	(0.33)	6.66	(0.00)	1.98	(0.00)	**1.87**	(0.00)
500	5%	15.86	(0.00)	5.24	(0.52)	10.98	(0.01)	**3.90**	(0.00)	**3.90**	(0.00)
500	10%	20.65	(0.01)	7.70	(0.52)	14.10	(0.01)	**5.18**	(0.00)	5.49	(0.01)

those with the disease. The dataset collected in a case-control study is available from the UC Irvine Machine Learning Repository. There are 195 speech signals recorded from 31 individuals, among which 147 signals are from people with Parkinson's disease (i.e., the case group), and the remaining 48 signals are from healthy people (i.e., the control group). There are 22 variables extracted from each signal. While some of the 195 speech signals are originated from the same individuals, they are treated as independent in our analysis.

Following Rothman (2012), we randomly partitioned this dataset into a training set of size 65 with 49 cases and a testing set of size 130 with 98 cases. We then estimated the covariance matrices corresponding to case and control based on the training data, and evaluated the performance of the covariance matrix estimates in terms of misclassification rate using quadratic discriminant analysis. The quadratic discriminant rule is given by

$$\arg\max_{j\in\{0,1\}} \Big\{ -\frac{1}{2}\log|\hat{\boldsymbol{\Sigma}}_j| - \frac{1}{2}\big(\boldsymbol{Y}_i^{(\text{test})} - \hat{\boldsymbol{\mu}}_j\big)' \times \hat{\boldsymbol{\Sigma}}_j^{-1}\big(\boldsymbol{Y}_i^{(\text{test})} - \hat{\boldsymbol{\mu}}_j\big) + \log(\hat{\pi}_j)\Big\}; \quad i = 1, \ldots, 130,$$

where $j = 0, 1$ refer to the control group and the case group respectively, $\log(\hat{\pi}_0) = 16/65$, $\log(\hat{\pi}_1) = 49/65$, and $\hat{\boldsymbol{\mu}}_j$ and $\hat{\boldsymbol{\Sigma}}_j$ are, respectively, the sample mean and the covariance matrix estimate based on the training data for group j. Lastly, $\boldsymbol{Y}_i^{(\text{test})}$ is the i-th observation in the testing set.

The tuning parameters of τ_1 and τ_2 were selected using 5-fold likelihood cross-validation among 7 tuning parameters of $\tau_2 \in \{1, 2, 4, 8, 16, 32, \infty\}$ and 200 tuning parameters of τ_1/τ_2 equally spaced in the log scale. The resulting misclassification rate based on 500 random partitions into training and testing sets is 0.202, with standard error 0.002. This is significantly smaller than 0.218 obtained from the sparse covariance matrix method of Rothman (2012), showing the effectiveness of the proposed methodology.

6. CONCLUDING REMARKS

In this paper two covariance estimation methods, *Cover* and *RCover*, are developed. The former is an eigenvalue-regularized method for which the corresponding estimator is defined as the minimizer of a penalized least-squares criterion. This *Cover* estimator is extremely fast to compute, possesses good theoretical support, and performs well in simulations. However, as with many other least-squares based estimators, it could produce poor estimates when outliers are present. To address this issue, *RCover* modifies *Cover*'s penalized least-squares criterion with Huber's loss function, and invokes the idea of the ES-algorithm (Oh et al., 2007) to develop a practical algorithm to solve the corresponding optimization problem. The *RCover* estimator performs very well in simulations, especially when the data are contaminated by outliers.

ACKNOWLEDGMENTS

The authors thank I-Ping Tu (Academia Sinica, Taiwan) for a discussion on eigenvalue clustering and regularization, which led to the development of *Cover*. The authors are also grateful to the referees and the Co-Guest Editors for their constructive comments and help which led to a much improved version of the paper.

Received 27 December 2014

REFERENCES

An, L. T. H. and Tao, P. D. (1997), 'Solving a class of linearly constrained indefinite quadratic problems by dc algorithms', *Journal of Global Optimization* **11**, 253–285. MR1469128

Anderson, T. W. (1963), 'Asymptotic theory for principal component analysis', *The Annals of Mathematical Statistics* **34**, 122–148. MR0145620

Bickel, P. J. and Levina, E. (2008*a*), 'Covariance regularization by thresholding', *The Annals of Statistics* **36**, 2577–2604. MR2485008

Bickel, P. J. and Levina, E. (2008*b*), 'Regularized estimation of large covariance matrices', *The Annals of Statistics* **36**, 199–227. MR2387969

Cai, T. and Liu, W. (2011), 'Adaptive thresholding for sparse covariance matrix estimation', *Journal of the American Statistical Association* **106**, 672–684. MR2847949

Chen, Y., Wiesel, A. and Hero, A. O. (2011), 'Robust shrinkage estimation of high-dimensional covariance matrices', *IEEE Transactions on Signal Processing* **59**, 4097–4107. MR2865971

Fan, J., Liao, Y. and Mincheva, M. (2013), 'Large covariance estimation by thresholding principal orthogonal complements (with discussion)', *Journal of the Royal Statistical Society Series B* **75**, 603–680. MR3091653

Friedman, J., Hastie, T. and Tibshirani, R. (2008), 'Sparse inverse covariance estimation with the graphical lasso', *Biostatistics* **9**, 432–441.

Furrer, R. and Bengtsson, T. (2007), 'Estimation of high-dimensional prior and posterior covariance matrices in Kalman filter variants', *Journal of Multivariate Analysis* **98**, 227–255. MR2301751

Huang, J. Z., Liu, N., Pourahmadi, M. and Liu, L. (2006), 'Covariance matrix selection and estimation via penalised normal likelihood', *Biometrika* **93**, 85–98. MR2277742

Huber, P. J. (1981), *Robust Statistics*, John Wiley & Sons, New York. MR0606374

Lam, C. and Fan, J. (2009), 'Sparsistency and rates of convergence in large covariance matrix estimation', *The Annals of Statistics* **37**, 42–54. MR2572459

Levina, E., Rothman, A. J. and Zhu, J. (2008), 'Sparse estimation of large covariance matrices via a nested lasso penalty', *The Annals of Applied Statistics* **2**, 245–263. MR2415602

Liu, H., Wang, L. and Zhao, T. (2014), 'Sparse covariance matrix estimation with eigenvalue constraints', *Journal of Computational and Graphical Statistics* **23**, 439–459. MR3215819

Muirhead, R. J. (1982), *Aspects of Multivariate Statistical Theory*, John Wiley & Sons, New York. MR0652932

Oh, H.-S., Nychka, D. and Lee, T. C. M. (2007), 'The role of pseudo data for robust smoothing with application to wavelet regression', *Biometrika* **94**, 893–904. MR2416798

Rothman, A. J. (2012), 'Positive definite estimators of large covariance matrices', *Biometrika* **99**, 733–740. MR2966781

Rothman, A. J., Levina, E. and Zhu, J. (2009), 'Generalized thresholding of large covariance matrices', *Journal of the American Statistical Association* **104**, 177–186. MR2504372

Rothman, A. J., Levina, E. and Zhu, J. (2010), 'A new approach to Cholesky-based covariance regularization in high dimensions', *Biometrika* **97**, 539–550. MR2672482

Shen, X. and Huang, H.-C. (2010), 'Grouping pursuit through a regularization solution surface', *Journal of the American Statistical Association* **105**, 727–739. MR2724856

Shen, X., Huang, H.-C. and Pan, W. (2012), 'Simultaneous supervised clustering and feature selection over a graph', *Biometrika* **99**, 899–914. MR2999167

von Neumann, J. (1937), 'Some matrix inequalities and metrization of metric-space', *Tomsk University Review* **1**, 286–300. Reprinted in A. H. Taub (Ed.) (1962). *John von Neumann: Collected Works* **4**, Pergamon, New York.

Wu, W. B. and Pourahmadi, M. (2003), 'Nonparametric estimation of large covariance matrices of longitudinal data', *Biometrika* **90**, 831–844. MR2024760

Xue, L., Ma, S. and Zou, H. (2012), 'Positive-definite l1-penalized estimation of large covariance matrices', *Journal of the American Statistical Association* **107**, 1480–1491. MR3036409

Hsin-Cheng Huang
Institute of Statistical Science
Academia Sinica
Taipei 115
Taiwan
E-mail address: hchuang@stat.sinica.edu.tw

Thomas C. M. Lee
Department of Statistics
University of California at Davis
CA 95758
USA
E-mail address: tcmlee@ucdavis.edu