Sparse Permutation Invariant Covariance Estimation

Adam J. Rothman

University of Michigan, Ann Arbor, USA.

Peter J. Bickel

University of California, Berkeley, USA.

Elizaveta Levina

University of Michigan, Ann Arbor, USA.

Ji Zhu

University of Michigan, Ann Arbor, USA.

Summary. The paper proposes a method for constructing a sparse estimator for the inverse covariance (concentration) matrix in high-dimensional settings. The estimator uses a penalized normal likelihood approach and forces sparsity by using a lasso-type penalty. We establish a rate of convergence in the Frobenius norm as both data dimension p and sample size n are allowed to grow, and show that the rate depends explicitly on how sparse the true concentration matrix is. We also show that a correlation-based version of the method exhibits better rates in the operator norm. The estimator is required to be positive definite, but we avoid having to use semi-definite programming by re-parameterizing the objective function in terms of the Cholesky factor of the concentration matrix, and derive an iterative optimization algorithm which reduces to solving a linear system at each iteration. Unlike other covariance estimation methods based on the Cholesky factor, our estimator is invariant to variable permutations. The method is compared to other estimators on simulated data and on a real data example of tumor tissue classification using gene expression data.

E-mail: ajrothma@umich.edu
E-mail: bickel@stat.berkeley.edu
E-mail: elevina@umich.edu
E-mail: jizhu@umich.edu

Address for correspondence: Elizaveta Levina, Department of Statistics, 439 West Hall, 1085 S. University, Ann Arbor, MI 48109-1107.

Keywords: Covariance matrix; High dimension low sample size; large p small n; Lasso; Sparsity; Cholesky decomposition

1. Introduction

Estimation of large covariance matrices, particularly in situations where the data dimension p is comparable to or larger than the sample size n, has attracted a lot of attention recently. The abundance of high-dimensional data is one reason for the interest in the problem: gene arrays, fMRI, various kinds of spectroscopy, climate studies, and many other applications often generate very high dimensions and moderate sample sizes. Another reason is the ubiquity of the covariance matrix in data analysis tools. Principal component analysis (PCA), linear and quadratic discriminant analysis (LDA and QDA), inference about the means of the components, and analysis of independence and conditional independence in graphical models all require an estimate of the covariance matrix or its inverse, also known as the precision or concentration matrix. Finally, recent advances in random matrix theory – see Johnstone (2001) for a review, and also Paul (2007) – allowed in-depth theoretical studies of the traditional estimator, the sample (empirical) covariance matrix, and showed that without regularization the sample covariance performs poorly in high dimensions. These results helped stimulate research on alternative estimators in high dimensions.

Many alternatives to the sample covariance matrix have been proposed. A large class of methods covers the situation where variables have a natural ordering, e.g., longitudinal data, time series, spatial data, or spectroscopy. The implicit regularizing assumption underlying these methods is that variables far apart in the ordering have small correlations (or partial correlations, if the object of regularization is the concentration matrix). Methods for regularizing covariance by banding or tapering have been proposed by Bickel and Levina (2004) and Furrer and Bengtsson (2007). Bickel and Levina (2006) showed consistency of banded estimators in the operator norm under mild conditions as long as $(\log p)/n \to 0$, for both banding the covariance matrix and the Cholesky factor of the inverse discussed below.

When the inverse of the covariance matrix is the primary goal and the variables are ordered, regularization is usually introduced via the modified Cholesky decomposition,

$$\Sigma^{-1} = L^T D^{-1} L.$$

Here L is a lower triangular matrix with $l_{jj} = 1$ and $l_{jj'} = -\phi_{jj'}$, where $\phi_{jj'}$, $j' < j$ is the

coefficient of $X_{j'}$ in the population regression of X_j on $X_1, \ldots, X_{j-1}$, and D is a diagonal matrix with residual variances of these regressions on the diagonal. Several approaches to regularizing the Cholesky factor L have been proposed, mostly based on its regression interpretation. A k-banded estimator of L can be obtained by regressing each variable only on its closest k predecessors; Wu and Pourahmadi (2003) proposed this estimator and chose k via an AIC penalty. Bickel and Levina (2006) showed that banding the Cholesky factor produces a consistent estimator in the operator norm under weak conditions on the covariance matrix, and proposed a cross-validation scheme for picking k. Huang et al. (2006) proposed adding either an l_2 (ridge) or an l_1 (lasso) penalty on the elements of L to the normal likelihood. The lasso penalty creates zeros in L in arbitrary locations, which is more flexible than banding, but (unlike in the case of banding) the resulting estimate of the inverse may not have any zeros at all. Levina et al. (2007) proposed adaptive banding, which, by using a nested lasso penalty, allows a different k for each regression, and hence is more flexible than banding while also retaining some sparsity in the inverse. Bayesian approaches to the problem introduce zeros via priors, either in the Cholesky factor (Smith and Kohn, 2002) or in the inverse itself (Wong et al., 2003).

There are, however, many applications where an ordering of the variables is not available: genetics, for example, or social and economic studies. Methods that are invariant to variable permutations (like the covariance matrix itself) are necessary in such applications. Regularizing large covariance matrices by Steinian shrinkage of eigenvalues has been proposed early on (Haff, 1980; Dey and Srinivasan, 1985). More recently, Ledoit and Wolf (2003) proposed a way to compute an optimal linear combination of the sample covariance with the identity matrix, which also results in shrinkage of eigenvalues. Shrinkage estimators are invariant to variable permutations but they do not affect the eigenvectors of the covariance, only the eigenvalues, and it has been shown that the sample eigenvectors are also not consistent when p is large (Johnstone and Lu, 2004). Shrinking eigenvalues also does not create sparsity in any sense. Sometimes alternative estimators are available in the context of a specific application – e.g., for a factor analysis model Fan et al. (2006) develop regularized estimators for both the covariance and its inverse.

Our focus here will be on sparse estimators of the concentration matrix. Sparse concentration matrices are widely studied in the graphical models literature, since zero partial correlations imply a graph structure. The classical graphical models approach, however, is

different from covariance estimation, since it normally focuses on just finding the zeros. For example, Drton and Perlman (2007) develop a multiple testing procedure for simultaneously testing hypotheses of zeros in the concentration matrix. There are also more algorithmic approaches to finding zeros in the concentration matrix, such as running a lasso regression of each variable on all the other variables (Meinshausen and Bühlmann, 2006), or the PC-algorithm (Kalisch and Bühlmann, 2007). Both have been shown to be consistent in high-dimensional settings, but none of these methods supply an estimator of the covariance matrix. In principle, once the zeros are found, a constrained maximum likelihood estimator of the covariance can be computed (Chaudhuri et al., 2007), but it is not clear what the properties of such a two-step procedure would be.

Two recent papers, d'Aspremont et al. (2007) and Yuan and Lin (2007), take a penalized likelihood approach by applying an l_1 penalty to the entries of the concentration matrix. This results in a permutation-invariant loss function that tends to produce a sparse estimate of the inverse, but in order to obtain a *bona fide* positive-definite estimator, d'Aspremont et al. (2007) and Yuan and Lin (2007) have to use semi-definite programming algorithms (Nesterov's method and the max-det algorithm, respectively). Both of these are computationally intensive and do not scale well with dimension, particularly the max-det algorithm, which uses interior point convex optimization. Yuan and Lin (2007) also provide an asymptotic analysis of this estimator, but only in the fixed p, large n case, which leaves the question of high-dimensional behavior unanswered.

This paper makes two main contributions. First, we analyze the estimator resulting from penalizing the normal likelihood with the l_1 penalty on the entries of the concentration matrix (we will refer to this estimator as SPICE – Sparse Permutation Invariant Covariance Estimator). We give an explicit convergence rate in the Frobenius norm allowing both p and n to grow and show that the rate depends on how sparse the true concentration matrix is. While the rate is not quite as good as that of banding (Bickel and Levina, 2006), this method does not require an ordering of the variables or assumptions on correlation decay, and still provides an improvement on the sample covariance matrix. A variation of the method which works with the correlation matrix shows somewhat better behavior in the operator norm, but is consistent with the first method in the Frobenius norm. The second contribution of the paper is an optimization algorithm that avoids semi-definite programming and automatically returns a positive-definite estimator. The main idea of the

algorithm is to parametrize the concentration matrix using the Cholesky decomposition, but, unlike other estimation methods that rely on the Cholesky decomposition, our algorithm is invariant under variable permutations. The algorithm reduces to solving (many) systems of linear equations, and thus scales better with dimension than semi-definite programming algorithms.

The rest of the paper is organized as follows: Section 2 summarizes the SPICE approach in general, and presents consistency results. The Cholesky-based computational algorithm, along with a discussion of optimization issues, is presented in Section 3. Section 4 presents numerical results for SPICE and a number of other methods, for simulated data and a real example on classification of colon tumors using gene expression data. Section 5 concludes with discussion.

2. Analysis of the SPICE method

We assume throughout that we observe $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n$, i.i.d. p-variate normal random variables with mean $\mathbf{0}$ and covariance matrix Σ_0, and write $\boldsymbol{X}_i = (X_{i1}, \ldots, X_{ip})^T$. Let $\Sigma_0 = [\sigma_{0ij}]$, and $\Omega_0 = \Sigma_0^{-1}$ be the inverse of the true covariance matrix. For any matrix $M = [m_{ij}]$, we write $|M|$ for the determinant of M, $\mathrm{tr}(M)$ for the trace of M, and $\varphi_{\max}(M)$ and $\varphi_{\min}(M)$ for the largest and smallest eigenvalues, respectively. We write $M^+ = \mathrm{diag}(M)$ for a diagonal matrix with the same diagonal as M, and $M^- = M - M^+$. In the asymptotic analysis, we will use the Frobenius matrix norm $\|M\|_F^2 = \sum_{i,j} m_{ij}^2$, and the operator norm (also known as matrix 2-norm), $\|M\|^2 = \varphi_{\max}(MM^T)$. We will also write $|\cdot|_1$ for the l_1 norm of a vector or matrix vectorized, i.e., for a matrix $|M|_1 = \sum_{i,j} |m_{ij}|$.

It is easy to see that under the normal assumption the negative log-likelihood, up to a constant, can be written in terms of the concentration matrix as

$$\ell(\boldsymbol{X}_1, \ldots, \boldsymbol{X}_n; \Omega) = \mathrm{tr}(\Omega\hat{\Sigma}) - \log|\Omega|,$$

where

$$\hat{\Sigma} = \frac{1}{n}\sum_{i=1}^{n} \big(\boldsymbol{X}_i - \bar{\boldsymbol{X}}\big)\big(\boldsymbol{X}_i - \bar{\boldsymbol{X}}\big)^T$$

is the sample covariance matrix.

We define the SPICE estimator $\hat{\Omega}_\lambda$ of the inverse covariance matrix as the minimizer of

the penalized negative log-likelihood,

$$\hat{\Omega}_\lambda = \arg\min_{\Omega \succ 0} \left\{ \mathrm{tr}(\Omega\hat{\Sigma}) - \log|\Omega| + \lambda|\Omega^-|_1 \right\} \tag{1}$$

where λ is a non-negative tuning parameter, and the minimization is taken over symmetric positive definite matrices.

The SPICE estimator is identical to the lasso-type estimator proposed by Yuan and Lin (2007), and very similar to the estimator of d'Aspremont et al. (2007) (they used $|\Omega|_1$ rather than $|\Omega^-|_1$ in the penalty). The loss function is invariant to permutations of variables and should encourage sparsity in $\hat{\Omega}$ due to the l_1 penalty applied to its off diagonal elements.

We make the following assumptions about the true model:

A1: Let the set $S = \{(i,j) : \Omega_{0ij} \neq 0,\ i \neq j\}$. Then $\mathrm{card}(S) \leq s$.

A2: $\varphi_{\min}(\Sigma_0) \geq \underline{k} > 0$, or equivalently $\varphi_{\max}(\Omega_0) \leq 1/\underline{k}$.

A3: $\varphi_{\max}(\Sigma_0) \leq \overline{k}$.

Note that assumption A2 guarantees that Ω_0 exists. Assumption A1 is more of a definition, since it does not stipulate anything about s ($s = p(p-1)/2$ would give a full matrix).

THEOREM 1. *Let $\hat{\Omega}_\lambda$ be the minimizer defined by* (1). *Under A1, A2, A3, if $\lambda \asymp \sqrt{\frac{\log p}{n}}$,*

$$\|\hat{\Omega}_\lambda - \Omega_0\|_F = O_P\left(\sqrt{\frac{(p+s)\log p}{n}}\right) . \tag{2}$$

The theorem can be restated, more suggestively, as

$$\frac{\|\hat{\Omega}_\lambda - \Omega_0\|_F^2}{p} = O_P\left(\left(1 + \frac{s}{p}\right)\frac{\log p}{n}\right) . \tag{3}$$

The reason for the second formulation (3) is the relation of the Frobenius norm to the operator norm, $\|M\|_F^2/p \leq \|M\|^2 \leq \|M\|_F^2$.

In the proof, we will need a lemma of Bickel and Levina (2006) (Lemma 3) which is based on a large deviation result of Saulis and Statulevičius (1991). We state the result here for completeness.

LEMMA 1. *Let Z_i be i.i.d. $\mathcal{N}(\mathbf{0}, \Sigma_p)$ and $\varphi_{\max}(\Sigma_p) \leq \overline{k} < \infty$. Then, if $\Sigma_p = [\sigma_{ab}]$,*

$$P\big[|\sum_{i=1}^n (Z_{ij}Z_{ik} - \sigma_{jk})| \geq n\nu\big] \leq c_1 \exp(-c_2 n\nu^2) \quad \textit{for } |\nu| \leq \delta \tag{4}$$

where c_1, c_2 and δ depend on $\overline{k}$ only.

Proof of Theorem 1. Let

$$\begin{aligned} Q(\Omega) &= \mathrm{tr}(\Omega\hat{\Sigma}) - \log|\Omega| + \lambda|\Omega^-| - \mathrm{tr}(\Omega_0\hat{\Sigma}) + \log|\Omega_0| - \lambda|\Omega_0^-|_1 \\ &= \mathrm{tr}\big[(\Omega-\Omega_0)(\hat{\Sigma}-\Sigma_0)\big] - (\log|\Omega| - \log|\Omega_0|) + \mathrm{tr}\big[(\Omega-\Omega_0)\Sigma_0\big] + \lambda(|\Omega^-|_1 - |\Omega_0^-|_1) \quad (5) \end{aligned}$$

Our estimate $\hat{\Omega}$ minimizes $Q(\Omega)$, or equivalently $\hat{\Delta} = \hat{\Omega} - \Omega_0$ minimizes $G(\Delta) \equiv Q(\Omega_0 + \Delta)$. Note that we suppress the dependence on λ in $\hat{\Omega}$ and $\hat{\Delta}$.

Now, using the integral form of the Taylor expansion gives

$$\log|\Omega_0+\Delta| - \log|\Omega_0| = \sum_{i,j}\Delta_{ij}\sigma_{0ij} - \frac{1}{2}\tilde{\Delta}^T\Big[\int_0^1 (\Omega_0+v\Delta)^{-1}\otimes(\Omega_0+v\Delta)^{-1}dv\Big]\tilde{\Delta} \quad (6)$$

where $\otimes$ is the Kronecker product (if $A = [a_{ij}]_{p_1\times q_1}$, $B = [b_{kl}]_{p_2\times q_2}$, then $A\otimes B = [a_{ij}b_{kl}]_{p_1p_2\times q_1q_2}$), and $\tilde{\Delta}$ is Δ vectorized to match the dimensions of the Kronecker product.

By symmetry of Δ and Σ_0,

$$\sum_{i,j}\Delta_{ij}\sigma_{0ij} = \mathrm{tr}(\Omega-\Omega_0)\Sigma_0 \ . \quad (7)$$

Therefore, we may write (5) as,

$$\begin{aligned} G(\Delta) =& \mathrm{tr}\big(\Delta(\hat{\Sigma}-\Sigma_0)\big) + \frac{1}{2}\tilde{\Delta}^T\Big[\int_0^1 (\Omega_0+v\Delta)^{-1}\otimes(\Omega_0+v\Delta)^{-1}dv\Big]\tilde{\Delta} \\ &+ \lambda(|\Omega_0^- + \Delta^-|_1 - |\Omega_0^-|_1) \quad (8) \end{aligned}$$

For an index set A and a matrix $M = [m_{ij}]$, write $M_A \equiv [m_{ij}I((i,j)\in A)]$, where $I(\cdot)$ is an indicator function. Recall $S = \{(i,j):\ \Omega_{0ij}\neq 0,\ i\neq j\}$ and let $\overline{S}$ be its complement. Note that $|\Omega_0^- + \Delta^-|_1 = |\Omega_{0S}^- + \Delta_S^-|_1 + |\Delta_{\overline{S}}^-|_1$, and $|\Omega_0^-|_1 = |\Omega_{0S}^-|_1$. Then the triangular inequality implies

$$\lambda\big(|\Omega_0^- + \Delta^-|_1 - |\Omega_0^-|_1\big) \geq \lambda\big(|\Delta_{\overline{S}}^-|_1 - |\Delta_S^-|_1\big) \ . \quad (9)$$

Now, using symmetry again as in (7), we write

$$|\mathrm{tr}\big(\Delta(\hat{\Sigma}-\Sigma_0)\big)| \leq \Big|\sum_{i\neq j}(\hat{\sigma}_{ij}-\sigma_{0ij})\Delta_{ij}\Big| + \Big|\sum_i(\hat{\sigma}_{ii}-\sigma_{0ii})\Delta_{ii}\Big| = \mathrm{I} + \mathrm{II}. \quad (10)$$

To bound term I, note that the union sum inequality and Lemma 1 imply that, with probability tending to 1,

$$\max_{i\neq j}|\hat{\sigma}_{ij}-\sigma_{0ij}| \leq C_1\sqrt{\frac{\log p}{n}}$$

and hence term I is bounded by

$$\mathrm{I} \leq C_1 \sqrt{\frac{\log p}{n}} |\Delta^-|_1. \tag{11}$$

The second bound comes from the Cauchy-Schwartz inequality and Lemma 1:

$$\begin{aligned} \mathrm{II} &\leq \left[\sum_{i=1}^{p} \left(\hat{\sigma}_{ii} - \sigma_{0ii} \right)^2 \right]^{1/2} \|\Delta^+\|_F \leq \sqrt{p} \max_{1 \leq i \leq p} |\hat{\sigma}_{ii} - \sigma_{0ii}| \, \|\Delta^+\|_F \\ &\leq C_2 \sqrt{\frac{p \log p}{n}} \|\Delta^+\|_F, \end{aligned} \tag{12}$$

also with probability tending to 1.

Consider now the set

$$\Theta_n(M, \varepsilon) = \{\Delta : \|\Delta^-\|_F = M r_n, \|\Delta^+\|_F = M t_n\},$$

where

$$r_n = \sqrt{\frac{s \log p}{n}} \to 0, \quad t_n = \sqrt{\frac{p \log p}{n}} \to 0.$$

Our main argument is the following. Note that G is convex, and

$$G(\hat{\Delta}) \leq G(0) = 0 \ .$$

Then, if we can show that

$$\inf\{G(\Delta) : \ \Delta \in \Theta_n(M, \varepsilon)\} > 0 \ ,$$

$\hat{\Delta}$ must be inside the sphere defined by Θ_n, and hence

$$\|\hat{\Delta}\|_F \leq M(r_n + t_n) \ . \tag{13}$$

Now, take

$$\lambda = \frac{C_1}{\varepsilon} \sqrt{\frac{\log p}{n}} \ . \tag{14}$$

By (8),

$$\begin{aligned} G(\Delta) &\geq \underline{k}^2 \|\Delta\|_F^2 - C_1 \sqrt{\frac{\log p}{n}} |\Delta^-|_1 - C_2 \sqrt{\frac{p \log p}{n}} \|\Delta^+\|_F + \lambda \left(|\Delta_{\overline{S}}^-|_1 - |\Delta_S^-|_1 \right) \\ &= \underline{k}^2 \|\Delta\|_F^2 - C_1 \sqrt{\frac{\log p}{n}} \left(1 - \frac{1}{\varepsilon} \right) |\Delta_{\overline{S}}^-|_1 - C_1 \sqrt{\frac{\log p}{n}} \left(1 + \frac{1}{\varepsilon} \right) |\Delta_S^-|_1 \\ &\quad - C_2 \sqrt{\frac{p \log p}{n}} \|\Delta^+\|_F \end{aligned} \tag{15}$$

The first term comes from a bound on the integral which we will argue separately below. The second term is always positive, and hence we may omit it for the lower bound. Now, note that

$$|\Delta_S^-|_1 \leq \sqrt{s}\|\Delta_S^-\|_F \leq \sqrt{s}\|\Delta^-\|_F.$$

Thus we have

$$\begin{aligned} G(\Delta) &\geq \|\Delta^-\|_F^2 \left[\underline{k}^2 - C_1\sqrt{\frac{s\log p}{n}}\left(1+\frac{1}{\varepsilon}\right)\|\Delta^-\|_F^{-1}\right] \\ &\quad + \|\Delta^+\|_F^2 \left[\underline{k}^2 - C_2\sqrt{\frac{p\log p}{n}}\|\Delta^+\|_F^{-1}\right] \\ &= \|\Delta^-\|_F^2 \left[\underline{k}^2 - \frac{C_1(1+\varepsilon)}{\varepsilon M}\right] + \|\Delta^+\|_F^2 \left[\underline{k}^2 - \frac{C_2}{M}\right] > 0 \end{aligned} \tag{16}$$

for M sufficiently large.

It only remains to check the bound on the integral term in (8). Recall that $\varphi_{\min}(M) = \min_{\|x\|=1} x^T M x$. After factoring out the norm of $\tilde{\Delta}$, we have, for $\Delta \in \Theta_n$,

$$\begin{aligned} &\varphi_{\min}\Big(\int_0^1 (\Omega_0 + v\Delta)^{-1} \otimes (\Omega_0 + v\Delta)^{-1} dv\Big) \\ &\quad \geq \int_0^1 \varphi_{\min}^2(\Omega_0 + v\Delta)^{-1} dv \geq \min_{0\leq v\leq 1} \varphi_{\min}^2(\Omega_0 + v\Delta)^{-1} \\ &\quad \geq \min\left\{\varphi_{\min}^2(\Omega_0+\Delta)^{-1} : \ \|\Delta\|_F \leq M(r_n + t_n)\right\}. \end{aligned}$$

The first inequality uses the fact that the eigenvalues of Kronecker products of symmetric matrices are the products of the eigenvalues of their factors. Now

$$\varphi_{\min}^2(\Omega_0+\Delta)^{-1} = \varphi_{\max}^{-2}(\Omega_0+\Delta) \geq (\|\Omega_0\| + \|\Delta\|)^{-2} \geq \underline{k}^2 \tag{17}$$

with probability tending to 1, since $\|\Delta\| \leq \|\Delta\|_F = o(1)$. This establishes the theorem. □

An inspection of the proof shows that the worst part of the rate, $\sqrt{p\log p/n}$, comes from estimating the diagonal. This suggests that if we were to use the correlation matrix rather than the covariance matrix, we should be able to get the rate of $\sqrt{s\log p/n}$. Indeed, let $\Sigma_0 = W\Gamma W$, where Γ is the true correlation matrix, and W is the diagonal matrix of true standard deviations. Let $\hat{W}$ and $\hat{\Gamma}$ be the sample estimates of W and Γ, i.e., $\hat{W}^2 = \hat{\Sigma}^+$, $\hat{\Gamma} = \hat{W}^{-1}\hat{\Sigma}\hat{W}^{-1}$. Let $K = \Gamma^{-1}$. Define a SPICE estimate of K by

$$\hat{K}_\lambda = \arg\min_{\Omega\succ 0}\left\{\mathrm{tr}(\Omega\hat{\Gamma}) - \log|\Omega| + \lambda|\Omega^-|_1\right\} \tag{18}$$

Then we immediately obtain

COROLLARY 1. *Under assumptions of Theorem 1,*

$$\|\hat{K}_\lambda - K\|_F = O_P\left(\sqrt{\frac{s\log p}{n}}\right) .$$

Corollary 1 does not, however, establish whether there is an advantage to switching to the correlation matrix for estimating Ω_0 itself rather than K. It turns out that in the Frobenius norm it does not make a difference, but it does allow us to get a SPICE result in the operator norm (matrix 2-norm). As discussed previously by Bickel and Levina (2006), El Karoui (2007) and others, the operator norm is more appropriate than the Frobenius norm for spectral analysis, e.g., PCA. It also allows for a direct comparison with banding rates obtained in Bickel and Levina (2006). Define a correlation-based estimator of the concentration matrix by

$$\tilde{\Omega}_\lambda = \hat{W}^{-1}\hat{K}_\lambda\hat{W}^{-1}. \tag{19}$$

Then we have the following result in operator norm.

THEOREM 2. *Under assumptions of Theorem 1,*

$$\|\tilde{\Omega}_\lambda - \Omega_0\| = O_P\left(\sqrt{\frac{(s+1)\log p}{n}}\right) .$$

Proof of Theorem 2. Write

$$\begin{aligned}\|\tilde{\Omega}_\lambda - \Omega_0\| &= \|\hat{W}^{-1}\hat{K}_\lambda\hat{W}^{-1} - W^{-1}KW^{-1}\| \\ &\le \|\hat{W}^{-1} - W^{-1}\|\,\|\hat{K}_\lambda - K\|\,\|\hat{W}^{-1} - W^{-1}\| \\ &+ \|\hat{W}^{-1} - W^{-1}\|(\|\hat{K}_\lambda\|\,\|W^{-1}\| + \|\hat{W}^{-1}\|\,\|K\|) \\ &+ \|\hat{K}_\lambda - K\|\,\|\hat{W}^{-1}\|\,\|W^{-1}\|\end{aligned}$$

where we are using the submultiplicative norm property $\|AB\| \le \|A\|\,\|B\|$ (see, e.g., Golub and Van Loan (1989)). Now, $\|W^{-1}\|$ and $\|K\|$ are $O(1)$ by assumptions A2 and A3. Lemma 1 implies that

$$\|\hat{W}^2 - W^2\| = O_P\left(\sqrt{\frac{\log p}{n}}\right), \tag{20}$$

and since $\|\hat{W}^{-1} - W^{-1}\| \overset{P}{\asymp} \|\hat{W}^2 - W^2\|$ (where by $A \overset{P}{\asymp} B$ we mean $A = O_P(B)$ and $B = O_P(A)$), we have the rate of $\sqrt{\log p/n}$ for $\|\hat{W}^{-1} - W^{-1}\|$. This together with Corollary 1 in turn implies that $\|\hat{W}^{-1}\|$ and $\|\hat{K}_\lambda\|$ are $O_P(1)$, and the theorem follows. □

Note that in the Frobenius norm, we only have $\|\hat{W}^2 - W^2\| = O_P(\sqrt{p\log p/n})$, and thus the Frobenius rate of $\tilde{\Omega}_\lambda$ is the same as that of $\hat{\Omega}_\lambda$.

3. The Cholesky-based SPICE algorithm

In this section, we develop an iterative algorithm for the computation of $\hat{\Omega}_\lambda$ in (1). Recall that the objective function f is given by:

$$f(\Omega) = \mathrm{tr}(\Omega\hat{\Sigma}) - \log|\Omega| + \lambda|\Omega^-|_1 \tag{21}$$

The objective function is convex in the elements of Ω and the algorithm finds $\hat{\Omega}$ by iteratively approximating the root point of $\nabla f(\Omega)$. Our strategy is to re-parameterize the objective (21) using the Cholesky decomposition of Ω. Rather than using the modified Cholesky decomposition with its regression interpretation, as has been standard in the literature, we simply write

$$\Omega = T^T T,$$

where $T = [t_{ij}]$ is a lower triangular matrix. We can still use the regression interpretation if needed, by writing

$$\begin{aligned} t_{jj'} &= -\frac{\phi_{jj'}}{\sqrt{d_{jj}}}, \quad j' < j \\ t_{jj} &= \frac{1}{\sqrt{d_{jj}}}, \end{aligned} \tag{22}$$

where $\phi_{jj'}$ is the coefficient of $X_{j'}$ in the regression of X_j on $X_1, \ldots, X_{j-1}$, and d_{jj} is the corresponding residual variance.

If we re-parameterize f in terms of T, the estimator will automatically be positive definite. To minimize the objective with respect to T, we use a quadratic approximation to f, take derivatives, and then iteratively solve a sequence of linear systems of equations involving groups of parameters in the Cholesky factor. Here we outline the main steps of the algorithm, and leave the full derivation for the Appendix.

In a slight abuse of notation, we write X for the $n \times p$ data matrix where each column has already been centered by its sample mean. The three terms in (21) can be expressed as

a function of T as follows:

$$\begin{aligned} \text{tr}(\Omega\hat{\Sigma}) &= \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{p}\left(\sum_{k=1}^{j} t_{jk}X_{ik}\right)^2 && (23) \\ \log|\Omega| &= 2\sum_{j=1}^{p}\log t_{jj} && (24) \\ |\Omega^-|_1 &= 2\sum_{j'>j}\left|\sum_{k=j'}^{p} t_{kj'}t_{kj}\right| && (25) \end{aligned}$$

The second step is to use quadratic approximations for $|u|$ and $\log(u)$, shown in (26) and (27), respectively. Since the algorithm is iterative, $u^{(k)}$ denotes the value of u from the previous iteration, and $u^{(k+1)}$ is the value at current iteration.

$$\begin{aligned} |u^{(k+1)}| &\approx \frac{(u^{(k+1)})^2}{2|u^{(k)}|} + \frac{|u^{(k)}|}{2} && (26) \\ \log u^{(k+1)} &\approx 2\frac{u^{(k+1)}}{u^{(k)}} - \frac{1}{2}\frac{(u^{(k+1)})^2}{(u^{(k)})^2} - \frac{3}{2} + \log(u^{(k)}) && (27) \end{aligned}$$

The quadratic approximation to f allows us to easily take derivatives of f with respect to current values of the parameters in T as we plug in values from the previous iteration as constants. Still, for the derivatives to be linear in the parameters, we have to further separate parameters in T into groups defined by the *columns* of T, with parameters grouped into vectors $\theta_c = (t_{cc}, t_{c+1,c}, \ldots, t_{pc})^\top$. The motivation for this grouping is to eliminate product terms in (25) between parameters in the same group, yielding linear equations after taking partial derivatives with respect to θ_c.

The algorithm requires an initial value $\hat{T}^{(0)}$, which corresponds to $\hat{\Omega}^{(0)}$. If the sample covariance $\hat{\Sigma}$ is non-degenerate, which is generally the case for $p < n$, one could simply set $\hat{\Omega}^{(0)} = \hat{\Sigma}^{-1}$. More generally, we found the following simple strategy to work well: approximate $\phi_{jj'}$ in (22) by regressing X_j on $X_{j'}$ *alone*, for $j' = 1, \ldots, j-1$, and then compute $\hat{T}^{(0)}$ using (22).

The Algorithm:

Step 0. Initialize $\hat{T} = \hat{T}^{(0)}$ and $\hat{\Omega}^{(0)} = (\hat{T}^{(0)})^T\hat{T}^{(0)}$.

Step 1. For each group $c = 1, \ldots, p$, solve $\nabla_{\theta_c} f(T) = 0$ to find $\hat{\theta}_c$ and update column c of $\hat{T}$.

Step 2. Repeat Step 1 until convergence of $\hat{T}$ and set $T^{(k+1)} = \hat{T}$.

Step 3. Set $\hat{\Omega}^{(k+1)} = (T^{(k+1)})^T T^{(k+1)}$ and repeat Steps 1-3 until convergence of $\hat{\Omega}$.

At first glance, Step 2 appears redundant. However, our approximations use values of $\hat{\Omega}^{(k)}$, and we found it faster to sweep through the columns of $\hat{T}$ till convergence without continuously updating the product $\hat{T}^T\hat{T}$. One could also say that Step 2 is needed because we only take partial derivatives ∇f with respect to one group of parameters at a time, holding all other parameters fixed; and Step 3 is needed because of the quadratic approximations for $|u|$ and $\log u$.

Note that the quadratic approximation (Step 3) is a standard technique in optimization and it has been in the statistics literature to handle lasso-type penalties, for example, Fan and Li (2001) and Huang et al. (2006). The iterative strategy in Step 2 is similar to the "shooting" method as in Fu (1998) and Friedman et al. (2007). The only difference is that the standard "shooting" method updates one parameter at a time, while we divide the parameters into disjoint "blocks" and update a block of parameters simultaneously. Since at each iteration the value of the objective function decreases, convergence is guaranteed.

Essentially, each iteration of the algorithm is equivalent to minimizing a normal likelihood penalized with a weighted ridge penalty. The penalty weights are inversely proportional to the magnitude of the off-diagonal elements of $\hat{\Omega}^{(k)}$. For sufficiently large λ, off-diagonal elements of $\hat{\Omega}^{(k)}$ will approach zero as k increases. Computationally, we avoid infinite weights by setting off-diagonal elements of $\hat{\Omega}^{(k)}$ to a small pre-set tolerance value ($\varepsilon = 10^{-10}$) if their magnitude falls below this value. After convergence, we replace these thresholded elements of $\hat{\Omega}^{(k)}$ with zeros.

In practice, we found that working with the correlation matrix as described in Theorem 2 is slightly better than working with the covariance matrix, although the differences are fairly small. Still, in all the numerical results we standardize the variables first and then rescale our estimate by the sample standard deviations of the variables.

Finally, like any other penalty-based approach, SPICE requires selecting the tuning parameter λ. In simulations, we generate a separate validation dataset, and select λ by maximizing the normal likelihood on the validation data with $\hat{\Omega}_\lambda$ estimated from the training data. Alternatively, one can use 5-fold cross-validation, which we do for the real data analysis. There is some theoretical basis for selecting the tuning parameter in this way – see Bickel and Levina (2007).

4. Numerical Results

In this section, we compare the performance of SPICE to other estimators using simulated and real data. Other permutation-invariant estimators we consider are the sample covariance itself and the shrinkage estimator of Ledoit and Wolf (2003). Neither of these methods introduces any sparsity in the estimator, so while they may be able to produce reasonable estimates under an overall estimation loss, there is no hope of recovering a sparse structure. We also include the Lasso regularization of the Cholesky factor proposed by Huang et al. (2006), which we will refer to simply as Lasso. They also use a normal likelihood loss, and the only difference from SPICE is that their penalty is on the entries of the modified Cholesky factor L rather than the concentration matrix itself, $\lambda \sum_{j>j'} |\phi_{jj'}|$. While this method is not strictly invariant to permutations, it does allow zeros in arbitrary locations in L, so it can potentially produce reasonable estimates under variable permutations.

4.1. Simulations

In simulations, we concentrate on comparing performance on a sparse concentration matrix. We construct two variable orderings under the same model: one where the Cholesky factor is also very sparse, and one where the Cholesky factor has no zeros at all, even though the concentration matrix itself is sparse. The first model is defined through the elements of its modified Cholesky factors L and D:

(a) Ω_1: $\phi_{j1} = 0.8$; $\phi_{jj'} = 0, j' > 1$; $d_j = 0.01$. This corresponds to a process generated by $X_1 = \varepsilon_1$, $X_j = 0.8X_1 + \varepsilon_j$ for $j = 2, \ldots, p$, with ε_j independent $N(0, d_j)$.

(b) $\Omega_2 = P^T \Omega_1 P$, where P is a permutation matrix reversing the order of the variables from $X_1, X_2, \ldots, X_p$ to $X_p, X_{p-1}, \ldots, X_1$. Under this model, $\phi_{jj'} \neq 0$ for all j, j'.

Both Ω_1 and Ω_2 are sparse (see Figure 1 (a),(d)) but only Ω_1 has a sparse Cholesky factor. For both covariance models, we generated $n = 100$ multivariate normal training observations and a separate set of 100 validation observations. We considered two different values of p, 30 and 100. The estimators were computed on the training data, with tuning parameters for SPICE and Lasso selected by minimizing the normal likelihood on the validation data. Using these values of the tuning parameters, we computed the estimated concentration matrix on the training data and compared it to the population concentration matrix.

We evaluate the concentration matrix estimation performance using five different measures. The first three are matrix norm losses, computed as

$$\Delta(\hat{\Omega}, \Omega) = \frac{\|\hat{\Omega} - \Omega\|}{\|\Omega\|}$$

where the norm $\|\cdot\|$ is either the operator norm, the Frobenius norm, or the matrix 1-norm ($\|M\|_1 = \max_j \sum_i |m_{ij}|$). We also compute the Kullback-Leibler loss as defined by Yuan and Lin (2007),

$$\Delta_{KL}(\hat{\Omega}, \Omega) = \text{tr}\left(\Sigma\hat{\Omega}\right) - \log\left|\Sigma\hat{\Omega}\right| - p \tag{28}$$

and the quadratic loss for the concentration matrix,

$$\Delta_Q(\Omega, \hat{\Omega}) = \text{tr}\left(\Sigma\hat{\Omega} - I\right)^2. \tag{29}$$

Note that all losses are based on $\hat{\Omega}$ and do not require inversion to compute $\hat{\Sigma}$, which is appropriate for a method estimating Ω. The Kullback-Leibler loss was used by Yuan and Lin (2007) and Levina et al. (2007) to assess performance of methods estimating Ω. Also note that the Kullback-Leibler loss and the quadratic loss of the concentration matrix are obtained from the standard entropy and quadratic losses of the covariance matrix (Lin and Perlman, 1985; Wu and Pourahmadi, 2003; Huang et al., 2006) by reversing the roles of Σ and Ω.

Results for the two models are summarized in Table 1, which gives the average losses and the corresponding standard errors over 50 replications. For sample covariance, Ledoit-Wolf's estimator, and SPICE, $\hat{\Omega}_1 = \hat{\Omega}_2$, since these estimators are invariant to permutations. For Lasso, which depends on the ordering, we report results for both $\hat{\Omega}_1$ and $\hat{\Omega}_2$. As expected, the Lasso method performs better for Ω_1 than Ω_2 since Ω_2 has a non-sparse Cholesky factor. SPICE outperforms all its competitors by a large margin under Kullback-Leibler and quadratic losses. Under matrix norm losses, all estimators except Lasso on Ω_2 are fairly close, with Ledoit-Wolf being best in matrix 1-norm and operator norm. Under Frobenius loss, SPICE is best for $p = 100$ and slightly worse when $p = 30$.

Out of all the estimators we consider, only SPICE and potentially Lasso have the ability to recognize sparsity in the inverse directly. To assess this, we compared percentages of true zeros estimated as zeros for SPICE and Lasso (Table 2), which shows that SPICE identifies a considerably larger percentage of true zeros than the Lasso, even for Ω_1. To show how these vary for each element of the concentration matrix, we show heatmaps of the

Table 1. Simulations: performance on Ω_1 and Ω_2.

p	Sample	Ledoit-Wolf	SPICE	Lasso $\hat{\Omega}_1$	Lasso $\hat{\Omega}_2$
	Kullback-Leibler loss				
30	8.38(0.14)	5.34(0.07)	1.92(0.03)	2.99(0.05)	3.80(0.07)
100	NA	116.64(1.84)	7.02(0.08)	19.13(0.31)	25.56(0.37)
	Quadratic loss				
30	35.28(0.96)	16.01(0.35)	3.38(0.08)	7.69(0.22)	12.77(0.38)
100	NA	933.29(26.19)	13.15(0.44)	144.23(9.58)	270.24(12.75)
	Matrix 1-norm				
30	0.74(0.04)	0.66(0.004)	0.73(0.006)	0.65(0.006)	0.72(0.04)
100	NA	0.87(0.001)	0.93(0.001)	0.91(0.003)	3.08(0.24)
	Operator norm				
30	0.62(0.04)	0.66(0.004)	0.73(0.006)	0.67(0.006)	0.60(0.04)
100	NA	0.87(0.001)	0.93(0.001)	0.92(0.003)	2.75(0.24)
	Frobenius norm				
30	0.66(0.04)	0.67(0.004)	0.71(0.006)	0.66(0.004)	0.60(0.04)
100	NA	0.97(0.001)	0.92(0.001)	0.92(0.003)	2.72(0.24)

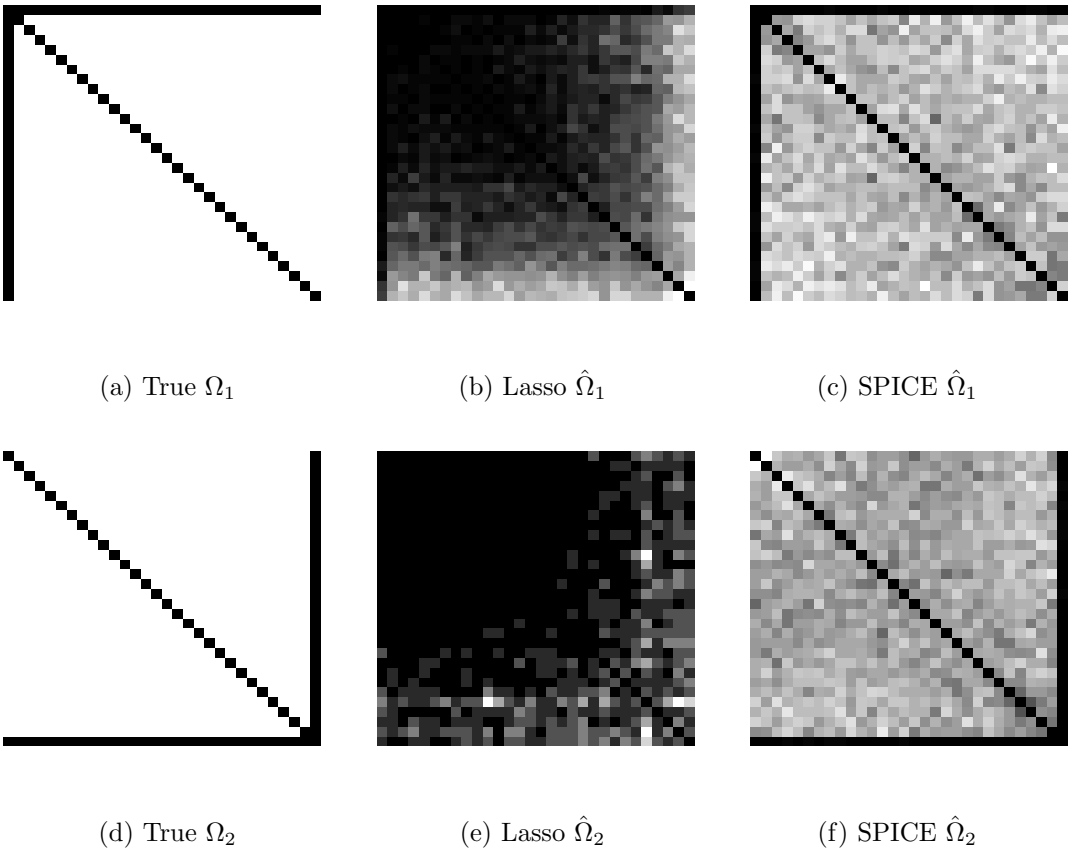
Table 2. Percentage of correctly estimated zeros in the concentration matrix (average and SE over 50 replications).

p	SPICE	Lasso $\hat{\Omega}_1$	Lasso $\hat{\Omega}_2$
30	48.0 (0.4)	21.0(0.7)	0.86(0.2)
100	64.7 (0.2)	16.8(1.2)	38.0(1.3)

number of zeros identified out of the 50 replications in Figure 1. This model is not easy for either method, but SPICE still has a substantial advantage over Lasso. Finally, following Yuan and Lin (2007) we show the average number of false positive and false negative edges identified in the corresponding graph over the 50 replications (Table 3). False positives are a much bigger problem for both methods than false negatives, but still, SPICE does better than Lasso on both false positives and false negatives.

Table 3. The number of false negative and false positive edges identified by each method (average and SE over 50 replications). The true number of edges is 29 for $p = 30$ and 99 for $p = 100$. The true number of non-edges is 406 for $p = 30$ and 4851 for $p = 100$.

	False Negatives			False Positives		
p	SPICE	Lasso $\hat{\Omega}_1$	Lasso $\hat{\Omega}_2$	SPICE	Lasso $\hat{\Omega}_1$	Lasso $\hat{\Omega}_2$
30	0.06 (0.04)	0.66(0.11)	0.90(0.13)	212.8(1.8)	320.8(3.0)	402.5(0.68)
100	5.9(0.41)	37.4(1.5)	26.2(0.51)	1714.8(8.8)	3005.4(64)	4034.4(56)



(a) True Ω_1 (b) Lasso $\hat{\Omega}_1$ (c) SPICE $\hat{\Omega}_1$

(d) True Ω_2 (e) Lasso $\hat{\Omega}_2$ (f) SPICE $\hat{\Omega}_2$

Fig. 1. Heatmaps of zeros identified in Ω_1 and Ω_2 out of 50 replications. White color is 50/50 zeros identified, black is 0/50.

4.2. Colon tumor classification example

In this section, we compare performance of the various estimators for LDA classification of tumors using gene expression data from Alon et al. (1999). In this experiment, colon adenocarcinoma tissue samples were collected, 40 of which were tumor tissues and 22 non-tumor tissues. Tissue samples were analyzed using an Affymetrix oligonucleotide array. The data were processed, filtered, and reduced to a subset of 2,000 gene expression values with the largest minimal intensity over the 62 tissue samples. Additional information about the dataset and pre-processing can be found in Alon et al. (1999).

To assess the performance at different dimensions, we reduce the full dataset of 2,000 gene expression values by selecting p most significant genes as measured by the two-sample t-statistic, for $p = 50, 100, 200$. Then we use linear discriminant analysis (LDA) to classify these tissues as either tumorous or non-tumorous. We classify each test observation $\boldsymbol{x}$ to either class $k = 0$ or $k = 1$ using the LDA rule

$$\delta_k(\boldsymbol{x}) = \arg\max_k \left\{ \boldsymbol{x}^T \hat{\Omega} \hat{\boldsymbol{\mu}}_k - \frac{1}{2} \hat{\boldsymbol{\mu}}_k^T \hat{\Omega} \hat{\boldsymbol{\mu}}_k + \log \hat{\pi}_k \right\}, \tag{30}$$

where $\hat{\pi}_k$ is the proportion of class k observations in the training data, $\hat{\boldsymbol{\mu}}_k$ is the sample mean for class k on the training data, and $\hat{\Omega}$ is an estimator of the inverse of the common covariance matrix on the training data computed by one of the methods under consideration. Detailed information on LDA can be found in Mardia et al. (1979).

To create training and test sets, we randomly split the data into a training set of size 42 and a testing set of size 20; following the approach used by Wang et al. (2007), we require the training set to have 27 tumor samples and 15 non-tumor samples. We repeat the split at random 100 times and measure the average classification error. The average errors with standard errors over the 100 splits are presented in Table 4.2. We omit the sample covariance because it is not invertible with such a small sample size, and include the naive Bayes classifier instead (where $\hat{\Sigma}$ is estimated by a diagonal matrix with sample variances on the diagonal). Naive Bayes has been shown to perform better than the sample covariance in high-dimensional settings (Bickel and Levina, 2004).

For an application such as classification, there are several possibilities for selecting the tuning parameter. Since we have no separate validation data available, we perform 5-fold cross-validation on the training data. One possibility (columns A in Table 4.2) is to continue using normal likelihood as a criterion for cross-validation, like we did in simulations.

Table 4. Averages and SEs of classification errors in % over 100 splits. Tuning parameter for Lasso and SPICE chosen by (A): 5-fold CV on the training data maximizing the likelihood; (B): 5-fold CV on the training data minimizing the classification error; (C): minimizing the classification error on the test data.

			Lasso			SPICE		
p	N. Bayes	L-W	A	B	C	A	B	C
50	15.8(0.8)	15.2(0.6)	15.3(0.7)	34.3(2.0)	12.0(0.6)	12.1(0.7)	14.7(0.7)	9.0(0.6)
100	20.0(0.8)	16.3(0.7)	19.5(0.8)	38.2(1.4)	16.4(0.7)	18.7(0.8)	16.9(0.9)	9.1(0.5)
200	23.1(1.0)	17.7(0.6)	23.2(1.0)	39.1(1.5)	18.2(0.7)	18.3(0.7)	18.0(0.7)	10.2(0.5)

Another possibility (columns B in Table 4.2) is to use classification error as the cross-validation criterion, since that is the ultimate performance measure in this case. Table 4.2 shows that for SPICE, both methods of tuning work similarly, whereas for Lasso tuning using the classification error, somewhat surprisingly, performs very poorly. For reference, we also include the best error rate achievable on the test data for a given estimator, which is obtained by selecting the tuning parameter to minimize the classification error on the test data (columns C in Table 4.2). Again, for each type of tuning SPICE does better than Lasso.

5. Discussion

We have analyzed a penalized likelihood approach to estimating a sparse concentration matrix via a lasso-type penalty, and showed that its rate of convergence depends explicitly on how sparse the true matrix is. This is analogous to results for banding (Bickel and Levina, 2006), where the rate of convergence depends on how quickly the off-diagonal elements of the true covariance decay, and for thresholding (Bickel and Levina, 2007; El Karoui, 2007), where the rate also depends on how sparse the true covariance is by various definitions of sparsity. We conjecture that other structures can be similarly dealt with, and other types of penalties may show similar behavior when applied to the "right" type of structure – for example, a ridge, bridge, or other more complex penalty may work well for a model that is not truly sparse but has many small entries. Investigation of these other structures is a subject for future work.

While we assumed normality, it can be replaced by a tail condition, analogously to Bickel

and Levina (2006). The use of normal likelihood is, of course, less justifiable if we do not assume normality, but it was found empirically that it still works reasonably well as a loss function even if the true distribution is not normal (Levina et al., 2007).

The Cholesky decomposition of covariance was only considered appropriate when variables are ordered, and we have shown it to be a useful tool for enforcing positive definiteness of the estimator even when variables have no natural ordering. Our optimization algorithm reduced to solving a linear system at each iteration, and we are exploring the possibility that other loss functions and penalties could also be re-parameterized and optimized similarly, possibly using different fast modern optimization algorithms.

Acknowledgments

We thank Sourav Chatterjee and Noureddine El Karoui (UC Berkeley) for helpful discussions. P. J. Bickel's research is partially supported by a grant from the NSF (DMS-0605236). E. Levina's research is partially supported by grants from the NSF (DMS-0505424) and the NSA (MSPF-04Y-120). J. Zhu's research is partially supported by grants from the NSF (DMS-0505432 and DMS-0705532).

References

Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D., and Levine, A. J. (1999). Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc Natl Acad Sci USA*, 96(12):6745–6750.

Bickel, P. J. and Levina, E. (2004). Some theory for Fisher's linear discriminant function, "naive Bayes", and some alternatives when there are many more variables than observations. *Bernoulli*, 10(6):989–1010.

Bickel, P. J. and Levina, E. (2006). Regularized estimation of large covariance matrices. *Ann. Statist.* To appear.

Bickel, P. J. and Levina, E. (2007). Covariance regularization by thresholding. Manuscript.

Chaudhuri, S., Drton, M., and Richardson, T. S. (2007). Estimation of a covariance matrix with zeros. *Biometrika*, 94(1):199–216.

d'Aspremont, A., Banerjee, O., and El Ghaoui, L. (2007). First-order methods for sparse covariance selection. *SIAM Journal on Matrix Analysis and its Applications*. To appear.

Dey, D. K. and Srinivasan, C. (1985). Estimation of a covariance matrix under Stein's loss. *Ann. Statist.*, 13(4):1581–1591.

Drton, M. and Perlman, M. D. (2007). A SINful approach to Gaussian graphical model selection. *J. Statist. Plann. Inference*. In press.

El Karoui, N. (2007). Operator norm consistent estimation of large dimensional sparse covariance matrices. Technical Report 734, UC Berkeley, Department of Statistics.

Fan, J., Fan, Y., and Lv, J. (2006). High dimensional covariance matrix estimation using a factor model. Technical report, Princeton University. Manuscript.

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *J. Amer. Statist. Assoc.*, 96(456):1348–1360.

Friedman, J., Hastie, T., and Tibshirani, R. (2007). Pathwise coordinate optimization. Technical report, Stanford University, Department of Statistics.

Fu, W. (1998). Penalized regressions: the bridge versus the lasso. *Journal of Computational and Graphical Statistics*, 7(3):397–416.

Furrer, R. and Bengtsson, T. (2007). Estimation of high-dimensional prior and posterior covariance matrices in Kalman filter variants. *Journal of Multivariate Analysis*, 98(2):227–255.

Golub, G. H. and Van Loan, C. F. (1989). *Matrix Computations*. The John Hopkins University Press, Baltimore, Maryland, 2nd edition.

Haff, L. R. (1980). Empirical Bayes estimation of the multivariate normal covariance matrix. *Ann. Statist.*, 8(3):586–597.

Huang, J., Liu, N., Pourahmadi, M., and Liu, L. (2006). Covariance matrix selection and estimation via penalised normal likelihood. *Biometrika*, 93(1):85–98.

Johnstone, I. M. (2001). On the distribution of the largest eigenvalue in principal components analysis. *Ann. Statist.*, 29(2):295–327.

Johnstone, I. M. and Lu, A. Y. (2004). Sparse principal components analysis. *J. Amer. Statist. Assoc.* Tentatively accepted.

Kalisch, M. and Bühlmann, P. (2007). Estimating high-dimensional directed acyclic graphs with the PC-algorithm. *J. Mach. Learn. Res.*, 8:613–636.

Ledoit, O. and Wolf, M. (2003). A well-conditioned estimator for large-dimensional covariance matrices. *Journal of Multivariate Analysis*, 88:365–411.

Levina, E., Rothman, A. J., and Zhu, J. (2007). Sparse estimation of large covariance matrices via a nested Lasso penalty. *Annals of Applied Statistics*. To appear.

Lin, S. P. and Perlman, M. D. (1985). A Monte Carlo comparison of four estimators for a covariance matrix. In Krishnaiah, P. R., editor, *Multivariate Analysis*, volume 6, pages 411–429. Elsevier Science Publishers.

Mardia, K. V., Kent, J. T., and Bibby, J. M. (1979). *Multivariate Analysis*. Academic Press, New York.

Meinshausen, N. and Bühlmann, P. (2006). High dimensional graphs and variable selection with the Lasso. *Annals of Statistics*, 34(3):1436–1462.

Paul, D. (2007). Asymptotics of the leading sample eigenvalues for a spiked covariance model. *Stat. Sinica*. To appear.

Saulis, L. and Statulevičius, V. A. (1991). *Limit Theorems for Large Deviations*. Kluwer Academic Publishers, Dordrecht.

Smith, M. and Kohn, R. (2002). Parsimonious covariance matrix estimation for longitudinal data. *J. Amer. Statist. Assoc.*, 97(460):1141–1153.

Wang, L., Zhu, J., and Zou, H. (2007). Hybrid huberized support vector machines for microarray classification. In *ICML '07: Proceedings of the 24th International Conference on Machine Learning*, pages 983–990, New York, NY, USA. ACM Press.

Wong, F., Carter, C., and Kohn, R. (2003). Efficient estimation of covariance selection models. *Biometrika*, 90:809–830.

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

Yuan, M. and Lin, Y. (2007). Model selection and estimation in the Gaussian graphical model. *Biometrika*, 94(1):19–35.

Appendix: Derivation of the Algorithm

In this section we give a full derivation of the linear systems involved in the optimization algorithm. Recall that we have re-parametrized the objective function (21) using (23)–(25). We divide T into p parameter groups corresponding to columns, compute partial derivatives with respect to $\theta_c = \{t_{lc}\}_{l=c}^{p}$ while holding all other parameters fixed, and solve a linear system corresponding to setting these partial derivatives to 0. Thus for each column c we solve $A^c\theta_c = b^c$, and our goal here is to derive explicit expressions for A^c and b^c.

For simplicity, we separate the likelihood and the penalty parts by writing $f(T) = \ell(T) + P(T)$. For the likelihood part, taking the partial derivative with respect to t_{lc}, $1 \le c \le p,\ c \le l \le p$ and applying the quadratic approximation (27) gives

$$\begin{aligned}
\frac{\partial}{\partial t_{\ell c}}\ell(T) &= -2\underbrace{\frac{\partial}{\partial t_{\ell c}}\sum_{j=1}^{p}\log t_{jj}}_{=0 \text{ if } j\neq c} + \frac{1}{n}\sum_{i=1}^{n}\underbrace{\frac{\partial}{\partial t_{\ell c}}\sum_{j=1}^{p}\left(\sum_{k=1}^{j} t_{jk}X_{ik}\right)^2}_{=0 \text{ if } j\neq l} \\
&= -2\mathbf{I}\{l=c\}\left[\frac{2}{t^0_{cc}} - \frac{t_{cc}}{(t^0_{cc})^2}\right] + 2\sum_{k=1}^{l} t_{lk}\hat{\sigma}_{kc} \\
&= t_{lc}\left[2\hat{\sigma}_{cc} + \mathbf{I}\{l=c\}\frac{2}{(t^0_{cc})^2}\right] + 2\sum_{k=1,\ k\neq c}^{l} t_{lk}\hat{\sigma}_{kc} - \mathbf{I}\{l=c\}\frac{4}{t^0_{cc}}, \qquad (31)
\end{aligned}$$

where t^0_{cc} denotes the value of t_{cc} from the previous iteration, and the last line is simply collecting the terms for setting up the linear system.

For the penalty part, write $\Omega = [\omega_{ij}]$. We use the quadratic approximation (26), which gives

$$\frac{\partial}{\partial t_{\ell c}}P(T) = \frac{\partial}{\partial t_{\ell c}}2\lambda\sum_{j'>j}\left|\sum_{k=j'}^{p} t_{kj'}t_{kj}\right| \approx \frac{\partial}{\partial t_{\ell c}}\sum_{j'>j}\frac{\lambda}{|\omega^0_{j'j}|}\omega^2_{j'j} = \sum_{k=1,k\neq c}^{l}\frac{\lambda}{|\omega^0_{ck}|}\frac{\partial}{\partial t_{\ell c}}\omega^2_{ck}\ , \quad (32)$$

since the only nonzero terms in (32) are those for which $j' \le l$ and either $j' = c$ or $j = c$. For $1 \le k \le l$ such that $k \neq c$, we have $\frac{\partial}{\partial t_{\ell c}}\omega^2_{ck} = 2\omega_{ck}t_{lk}$, and collecting terms together we

get

$$\frac{\partial}{\partial t_{\ell c}} P(T) = \sum_{q=c}^{p} t_{qc} \left[2\lambda \sum_{k=1, k\neq c}^{l} \frac{t_{lk}\, t_{qk}}{|\omega^0_{ck}|} \right] . \tag{33}$$

Combining together (31) and (33), we have the system of linear systems for parameters in column c, $A^c\theta_c = b^c$. The system has $p - c + 1$ equations with as many unknowns, and the elements of the matrix A^c and the vector b^c are given, for $c \leq l \leq p$ and $c \leq q \leq p$, by

$$A^c_{lq} = \sum_{k=1, k\neq c}^{l} \frac{2\lambda t_{lk}\, t_{qk}}{|\omega^0_{ck}|} + \mathbf{I}\{q = l\} \left[2\hat{\sigma}_{cc} + \mathbf{I}\{l = c\} \frac{2}{(t^0_{cc})^2} \right]$$

$$b^c_l = - \sum_{k=1,\ k\neq c}^{l} 2 t_{lk} \hat{\sigma}_{kc} + \mathbf{I}\{l = c\} \frac{4}{t^0_{cc}}$$