Biometrika (2010), **0**, 0, *pp*. 1–24

 Advance Access publication on DAY MONTH YEAR

Printed in Great Britain

Sparse Estimation of a Covariance Matrix

BY JACOB BIEN AND ROBERT TIBSHIRANI

Departments of Statistics and Health, Research & Policy, Stanford University, Sequoia Hall, 390 Serra Mall, Stanford University, Stanford, California 94305-4065, U.S.A.

jbien@stanford.edu tibs@stanford.edu

SUMMARY

We consider a method for estimating a covariance matrix on the basis of a sample of vectors drawn from a multivariate normal distribution. In particular, we penalize the likelihood with a lasso penalty on the entries of the covariance matrix. This penalty plays two important roles: it reduces the effective number of parameters, which is important even when the dimension of the vectors is smaller than the sample size since the number of parameters grows quadratically in the number of variables, and it produces an estimate which is sparse. In contrast to sparse inverse covariance estimation, our method's close relative, the sparsity attained here is in the covariance matrix itself rather than in the inverse matrix. Zeros in the covariance matrix correspond to marginal independencies; thus, our method performs model selection while providing a positive definite estimate of the covariance. The proposed penalized maximum likelihood problem is not convex, so we use a majorize-minimize approach in which we iteratively solve convex approximations to the original non-convex problem. We discuss tuning parameter selection and demonstrate on a flow-cytometry dataset how our method produces an interpretable graphical display of the relationship between variables. We perform simulations that suggest that simple

elementwise thresholding of the empirical covariance matrix is competitive with our method for identifying the sparsity structure. Additionally, we show how our method can be used to solve a previously studied special case in which a desired sparsity pattern is prespecified.

Some key words: Concave-convex procedure; Covariance graph; Covariance matrix; Generalized gradient descent; Lasso; Majorization-minimization; Regularization; Sparsity.

1. INTRODUCTION

Estimation of a covariance matrix on the basis of a sample of vectors drawn from a multivariate Gaussian distribution is among the most fundamental problems in statistics. However, with the increasing abundance of high-dimensional datasets, the fact that the number of parameters to estimate grows with the square of the dimension suggests that it is important to have robust alternatives to the standard sample covariance matrix estimator. In the words of Dempster (1972),

> "The computational ease with which this abundance of parameters can be estimated should not be allowed to obscure the probable unwisdom of such estimation from limited data."

Following this note of caution, many authors have developed estimators which mitigate the situation by reducing the effective number of parameters through imposing sparsity in the *inverse* covariance matrix. Dempster (1972) suggests setting elements of the inverse covariance matrix to zero. Meinshausen & Bühlmann (2006) propose using a series of lasso regressions to identify the zeros of the inverse covariance matrix. More recently, Yuan & Lin (2007), Banerjee et al. (2008), and Friedman et al. (2007) frame this as a sparse estimation problem, performing penalized maximum likelihood with a lasso penalty on the inverse covariance matrix; this is known as the *graphical lasso*. Zeros in the inverse covariance matrix are of interest because they correspond to conditional independencies between variables.

In this paper, we consider the problem of estimating a sparse covariance matrix. Zeros in a covariance matrix correspond to marginal independencies between variables. A *Markov network* is a graphical model that represents variables as nodes and conditional dependencies between variables as edges; a *covariance graph* is the corresponding graphical model for marginal independencies. Thus, sparse estimation of the covariance matrix corresponds to estimating a covariance graph as having a small number of edges. While less well-known than Markov networks, covariance graphs have also been met with considerable interest (Drton & Richardson, 2008). For example, Chaudhuri et al. (2007) consider the problem of estimating a covariance matrix given a prespecified zero-pattern; Khare and Rajaratnam, in an unpublished 2009 technical report available at http://statistics.stanford.edu/∼ckirby/techreports/GEN/2009/2009-01.pdf, formulate a prior for Bayesian inference given a covariance graph structure; Butte et al. (2000) introduce the related notion of a *relevance network*, in which genes with pairwise correlation exceeding a threshold are connected by an edge; also, Rothman et al. (2009) consider applying shrinkage operators to the sample covariance matrix to get a sparse estimate. Most recently, Rothman et al. (2010) propose a lasso-regression based method for estimating a sparse covariance matrix in the setting where the variables have a natural ordering.

The purpose of this present work is to develop a method which, in contrast to pre-existing methods, estimates both the non-zero covariances and the graph structure, i.e., the locations of the zeros, simultaneously. In particular, our method is permutation invariant in that it does not assume an ordering to the variables (Rothman et al., 2008). In other words, our method does for covariance matrices what the graphical lasso does for inverse covariance matrices. Indeed, as with the graphical lasso, we propose maximizing a penalized likelihood.

2. THE OPTIMIZATION PROBLEM

Suppose that we observe a sample of n multivariate normal random vectors, $X_1, \ldots, X_n \sim N_p(0, \Sigma)$. The log-likelihood is

$$\ell(\Sigma) = -\frac{np}{2}\log 2\pi - \frac{n}{2}\log\det\Sigma - \frac{n}{2}\mathrm{tr}(\Sigma^{-1}S),$$

where we define $S = n^{-1}\sum_{i=1}^{n} X_i X_i^T$. The lasso (Tibshirani, 1996) is a well-studied regularizer which has the desirable property of encouraging many parameters to be exactly zero. In this paper, we suggest adding to the likelihood a lasso penalty on $P * \Sigma$, where P is an arbitrary matrix with non-negative elements and $*$ denotes elementwise multiplication. Thus, we propose the estimator that solves

$$\mathrm{Minimize}_{\Sigma \succ 0} \left\{ \log\det\Sigma + \mathrm{tr}(\Sigma^{-1}S) + \lambda\|P * \Sigma\|_1 \right\}, \tag{1}$$

where for a matrix A, we define $\|A\|_1 = \|\mathrm{vec}A\|_1 = \sum_{ij}|A_{ij}|$. Two common choices for P would be the matrix of all ones or this matrix with zeros on the diagonal to avoid shrinking diagonal elements of Σ. Lam & Fan (2009) study the theoretical properties of a class of problems including this estimator but do not discuss how to solve the optimization problem. Additionally, while writing a draft of this paper, we learned of independent and concurrent work by Khare and Rajaratnam, presented at the 2010 Joint Statistical Meetings, in which they propose solving (1) with this latter choice for P. Another choice is to take $P_{ij} = 1\{i \neq j\}/|S_{ij}|$, which is the covariance analogue of the adaptive lasso penalty (Zou, 2006). In Section 6, we will discuss another choice of P that provides an alternative method for solving the prespecified zeros problem considered by Chaudhuri et al. (2007).

In words, (1) seeks a matrix Σ under which the observed data would have been likely *and* for which many variables are marginally independent. The graphical lasso problem is identical to (1) except that the penalty takes the form $\|\Sigma^{-1}\|_1$ and the optimization variable is Σ^{-1}.

Solving (1) is a formidable challenge since the objective function is non-convex and therefore may have many local minima. A key observation in this work is that the optimization problem, although non-convex, possesses special structure that suggests a method for performing the optimization. In particular, the objective function decomposes into the sum of a convex and a concave function. Numerous papers in fields spanning machine learning and statistics have made use of this structure to develop specialized algorithms: difference of convex programming focuses on general techniques to solving such problems both exactly and approximately (Horst & Thoai, 1999; An & Tao, 2005); the concave-convex procedure (Yuille & Rangarajan, 2003) has been used in various machine learning applications and studied theoretically (Yuille & Rangarajan, 2003; Argyriou et al., 2006; Sriperumbudur & Lanckriet, 2009); majorization-minimization algorithms have been applied in statistics to solve problems such as least-squares multidimensional scaling, which can be written as the sum of a convex and concave part (de Leeuw & Mair, 2009); most recently, Zhang (2010) approaches regularized regression with non-convex penalties from a similar perspective.

3. ALGORITHM FOR PERFORMING THE OPTIMIZATION

3·1. *A majorization-minimization approach*

While (1) is not convex, we show in Appendix 1 that the objective is the sum of a convex and concave function. In particular, $\mathrm{tr}(\Sigma^{-1}S) + \lambda\|P * \Sigma\|_1$ is convex in Σ while $\log\det\Sigma$ is concave. This observation suggests a majorize-minimize scheme to approximately solving (1).

Majorize-minimize algorithms work by iteratively minimizing a sequence of majorizing functions (e.g., chapter 6 of Lange 2004; Hunter & Li 2005). The function $f(x)$ is said to be majorized by $g(x \mid x_0)$, if $f(x) \leq g(x \mid x_0)$ for all x and $f(x_0) = g(x_0 \mid x_0)$. To minimize f, the algorithm starts at a point $x^{(0)}$ and then repeats until convergence, $x^{(t)} = \mathrm{argmin}_x g(x \mid x^{(t-1)})$.

This is advantageous when the function $g(\cdot \mid x_0)$ is easier to minimize than $f(\cdot)$. These updates have the favorable property of being non-increasing, i.e., $f(x^{(t)}) \leq f(x^{(t-1)})$.

A common majorizer for the sum of a convex and a concave function is to replace the latter part with its tangent. This method has been referred to in various literatures as the concave-convex procedure, the difference of convex functions algorithm, and multi-stage convex relaxations. Since $\log\det\Sigma$ is concave, it is majorized by its tangent plane: $\log\det\Sigma \leq \log\det\Sigma_0 + \mathrm{tr}\{\Sigma_0^{-1}(\Sigma - \Sigma_0)\}$. Therefore, the objective function of (1),

$$f(\Sigma) = \log\det\Sigma + \mathrm{tr}(\Sigma^{-1}S) + \lambda\|P * \Sigma\|_1,$$

is majorized by $g(\Sigma \mid \Sigma_0) = \log\det\Sigma_0 + \mathrm{tr}(\Sigma_0^{-1}\Sigma) - p + \mathrm{tr}(\Sigma^{-1}S) + \lambda\|P * \Sigma\|_1$. This suggests the following majorize-minimize iteration to solve (1):

$$\hat{\Sigma}^{(t)} = \mathrm{argmin}_{\Sigma \succ 0}\left[\mathrm{tr}\{(\hat{\Sigma}^{(t-1)})^{-1}\Sigma\} + \mathrm{tr}(\Sigma^{-1}S) + \lambda\|P * \Sigma\|_1\right]. \tag{2}$$

To initialize the above algorithm, we may take $\hat{\Sigma}^{(0)} = S$ or $\hat{\Sigma}^{(0)} = \mathrm{diag}(S_{11}, \ldots, S_{pp})$. We have thus replaced a difficult non-convex problem by a sequence of easier convex problems, each of which is a semidefinite program. The value of this reduction is that we can now appeal to algorithms for convex optimization. A similar stategy was used by Fazel et al. (2003), who pose a non-convex $\log\det$-minimization problem. While we cannot expect (2) to yield a global minimum of our non-convex problem, An & Tao (2005) show that limit points of such an algorithm are critical points of the objective (1).

In the next section, we propose an efficient method to perform the convex minimization in (2). It should be noted that if $S \succ 0$, then by Proposition 1 of Appendix 2, we may tighten the constraint $\Sigma \succ 0$ of (2) to $\Sigma \succeq \delta I_p$ for some $\delta > 0$, which we can compute and depends on the smallest eigenvalue of S. We will use this fact to prove a rate of convergence of the algorithm presented in the next section.

3·2. *Solving* (2) *using generalized gradient descent*

Problem (2) is convex and therefore any local minimum is guaranteed to be the global minimum. We employ a generalized gradient descent algorithm, which is the natural extension of gradient descent to non-differentiable objectives (e.g., Beck & Teboulle 2009). Given a differentiable convex problem $\min_{x\in\mathcal{C}} L(x)$, the standard projected gradient step is $x = P_{\mathcal{C}}\{x - t\nabla L(x)\}$ and can be viewed as solving the problem $x = \mathrm{argmin}_{z\in\mathcal{C}}(2t)^{-1}\|z - \{x - t\nabla L(x)\}\|^2$. To solve $\min_{x\in\mathcal{C}} L(x) + p(x)$ where p is a non-differentiable function, generalized gradient descent instead solves $x = \mathrm{argmin}_{z\in\mathcal{C}}(2t)^{-1}\|z - \{x - t\nabla L(x)\}\|^2 + p(z)$.

In our case, we want to solve

$$\mathrm{Minimize}_{\Sigma\succeq\delta I_p}\left\{\mathrm{tr}(\Sigma_0^{-1}\Sigma) + \mathrm{tr}(\Sigma^{-1}S) + \lambda\|P * \Sigma\|_1\right\},$$

where for notational simplicity we let $\Sigma_0 = \hat{\Sigma}^{(t-1)}$ be the solution from the previous iteration of (2). Since the matrix derivative of $L(\Sigma) = \mathrm{tr}(\Sigma_0^{-1}\Sigma) + \mathrm{tr}(\Sigma^{-1}S)$ is $dL(\Sigma)/d\Sigma = \Sigma_0^{-1} - \Sigma^{-1}S\Sigma^{-1}$, the generalized gradient steps are given by

$$\Sigma = \mathrm{argmin}_{\Omega\succeq\delta I_p}\left\{(2t)^{-1}\|\Omega - \Sigma + t(\Sigma_0^{-1} - \Sigma^{-1}S\Sigma^{-1})\|_F^2 + \lambda\|P * \Omega\|_1\right\}. \qquad (3)$$

Without the constraint $\Omega \succeq \delta I_p$, this reduces to the simple update

$$\Sigma \leftarrow \mathcal{S}\left\{\Sigma - t(\Sigma_0^{-1} - \Sigma^{-1}S\Sigma^{-1}),\ \lambda tP\right\},$$

where $\mathcal{S}$ is the elementwise soft-thresholding operator defined by $\mathcal{S}(A, B)_{ij} = \mathrm{sign}(A_{ij})(A_{ij} - B_{ij})_+$. Clearly, if the unconstrained solution to (3) happens to have minimum eigenvalue greater than or equal to δ, then the above expression is the correct generalized gradient step. In practice, we find that this is often the case, meaning we may solve (3) quite efficiently; however, when we find that the minimum eigenvalue of the soft-thresholded matrix is below δ, we perform the optimization using the *alternating direction method of multipliers* (e.g., Boyd et al. 2011), which is given in Appendix 3.

Generalized gradient descent is guaranteed to get within ϵ of the optimal value in $O(\epsilon^{-1})$ steps as long as $dL(\Sigma)/d\Sigma$ is Lipschitz continuous (Beck & Teboulle, 2009). While this condition is not true of our objective on $\Sigma \succ 0$, we show in Appendix 2 that we can change the constraint to $\Sigma \succeq \delta I_p$ for some $\delta > 0$ without changing the solution. On this set, $dL(\Sigma)/d\Sigma$ is Lipschitz, with constant $2\|S\|_2\delta^{-3}$, thus establishing that generalized gradient descent will converge with the stated rate.

In summary, Algorithm 1 presents our algorithm for solving (1). It has two loops: an outer loop in which the majorize-minimize algorithm approximates the non-convex problem iteratively by a series of convex relaxations; and an inner loop in which generalized gradient descent is used to solve each convex relaxation. The first iteration is usually simple soft-thresholding of S, unless the result has an eigenvalue less than δ. Generalized gradient descent belongs to a larger class of

Algorithm 1 Basic Algorithm for solving (1)

1: $\Sigma \leftarrow S$
2: **repeat**
3: $\quad\Sigma_0 \leftarrow \Sigma$
4: $\quad$**repeat**
5: $\quad\quad\Sigma \leftarrow \mathcal{S}\left\{\Sigma - t(\Sigma_0^{-1} - \Sigma^{-1}S\Sigma^{-1}),\ \lambda tP\right\}$ where $\mathcal{S}$ denotes elementwise soft-thresholding. If $\Sigma \not\succeq \delta I_p$, then instead perform alternating direction method of multipliers given in Appendix 3.
6: $\quad$**until** convergence
7: **until** convergence

first-order methods, which do not require computing the Hessian. Nesterov (2005) shows that a simple modification of gradient descent can dramatically improve the rate of convergence so that a value within ϵ of optimal is attained within only $O(\epsilon^{-1/2})$ steps (e.g., Beck & Teboulle 2009).

Due to space restrictions, we do not include this latter algorithm, which is a straightforward modification of Algorithm 1. Running our algorithm on a sequence of problems in which $\Sigma = I_p$ and with λ chosen to ensure an approximately constant proportion of non-zeros across differently sized problems, we estimate that the run time scales approximately like p^3. We will be releasing an `R` package which implements this approach to the ℓ_1-penalized covariance problem.

For a different perspective of our minimize-majorize algorithm, we rewrite (1) as

$$\text{Minimize}_{\Sigma \succ 0, \Theta \succ 0} \left\{ \text{tr}(\Sigma^{-1} S) + \lambda \|P * \Sigma\|_1 + \text{tr}(\Sigma\Theta) - \log\det\Theta \right\}. \tag{4}$$

This is a biconvex optimization problem in that the objective is convex in either variable holding the other fixed; however, it is not *jointly* convex because of the $\text{tr}(\Sigma\Theta)$ term. The standard alternate minimization technique to this biconvex problem reduces to the algorithm of (2). To see this, note that minimizing over Θ while holding Σ fixed gives $\widehat{\Theta} = \Sigma^{-1}$.

3·3. *A note on the $p > n$ case*

When $p > n$, S cannot be full rank and thus there exists $v \neq 0$ such that $Sv = 0$. Let $V = [v : V_\perp]$ be an orthogonal matrix. Denoting the original problem's objective as $f(\Sigma) = \log\det\Sigma + \text{tr}(\Sigma^{-1}S) \; + \; \lambda\|P * \Sigma\|_1$, we see that

$$f(\alpha vv^T + V_\perp V_\perp^T) = \log\alpha + \text{tr}(V_\perp^T S V_\perp) + \lambda\|P * (\alpha vv^T + V_\perp V_\perp^T)\|_1 \to -\infty, \quad \alpha \to 0.$$

Conversely, if $S \succ 0$, then, writing the eigenvalue decomposition of $\Sigma = \sum_{i=1}^p \lambda_i u_i u_i^T$ with $\lambda_1 \geq \cdots \geq \lambda_p > 0$, we have

$$f(\Sigma) \geq \log\det\Sigma + \text{tr}(\Sigma^{-1}S) = \text{constant} + \log\lambda_p + u_p^T S u_p/\lambda_p \to \infty$$

as $\lambda_p \to 0$ since $u_p^T S u_p > 0$.

Thus, if $S \succ 0$, the problems $\inf_{\Sigma \succeq 0} f(\Sigma)$ and $\inf_{\Sigma \succ 0} f(\Sigma)$ are equivalent, while if S is not full rank then the solution will be degenerate. We therefore set $S = S + \epsilon I_p$ for some $\epsilon > 0$ when

S is not full rank. In this case, the observed data lies in a lower dimensional subspace of R^p, and adding ϵI_p to S is equivalent to augmenting the dataset with points that do not lie perfectly in the span of the observed data.

3·4. *Using the sample correlation matrix instead of the sample covariance matrix*

Let $D = \mathrm{diag}(S_{11}, \ldots, S_{pp})$ so that $R = D^{-1/2} S D^{-1/2}$ is the sample correlation matrix. Rothman et al. (2008) suggest that, in the case of estimating the concentration matrix, it can be advantageous to use R instead of S. In this section, we consider solving

$$\widehat{\Theta}(R, P) = \mathrm{argmin}_{\Theta \succ 0} \left\{ \log\det\Theta + \mathrm{tr}(\Theta^{-1} R) + \lambda \|P * \Theta\|_1 \right\}, \tag{5}$$

and then taking $\widetilde{\Sigma} = D^{1/2} \widehat{\Theta}(R, P) D^{1/2}$ as an estimate for the covariance matrix. Expressing the objective function in (5) in terms of $\Sigma = D^{1/2} \Theta D^{1/2}$ gives, after some manipulation,

$$-\sum_{i=1}^{p} \log(S_{ii}) + \log\det\Sigma + \mathrm{tr}(\Sigma^{-1} S) + \lambda \|(D^{-1/2} P D^{-1/2}) * \Sigma\|_1.$$

Thus, the estimator $\widetilde{\Sigma}$ based on the sample correlation matrix is equivalent to solving (1) with a rescaled penalty matrix: $P_{ij} \leftarrow P_{ij}/(S_{ii} S_{jj})^{1/2}$. This gives insight into (5): it applies a stronger penalty to variables with smaller variances. For n large, $S_{ii} \approx \Sigma_{ii}$, and so we can think of this modification as applying the lasso penalty on the correlation scale, i.e., $\|P * \Omega\|_1$ where $\Omega_{ij} = \Sigma_{ij}(\Sigma_{ii}\Sigma_{jj})^{-1/2}$, rather than on the covariance scale. An anonymous referee points out that this estimator has the desirable property of being invariant to both scaling of variables and to permutation of variable labels.

4. CROSS-VALIDATION FOR TUNING PARAMETER SELECTION

In applying this method, one will usually need to select an appropriate value of λ. Let $\widehat{\Sigma}_\lambda(S)$ denote the estimate of Σ we get by applying our algorithm with tuning parameter λ to $S = n^{-1} \sum_{i=1}^{n} X_i X_i^T$ where $X_1, \ldots, X_n$ are n independent $N_p(0, \Sigma)$ random vec-

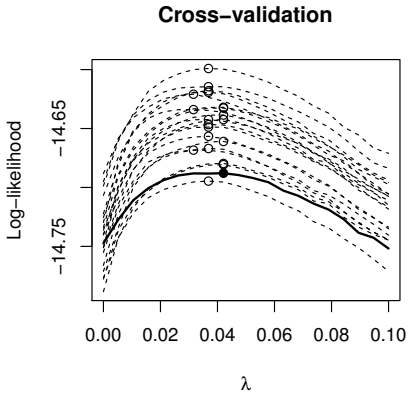

Fig. 1. Tuning parameter selection via cross-validation: Each dashed line is a realization of $\hat{\alpha}_{CV}(\lambda)$ and the solid line is $\alpha(\lambda)$. Each open circle shows a realization of $\hat{\lambda}_{CV}$; the solid circle shows $\mathrm{argmax}_\lambda \alpha(\lambda)$.

tors. We would like to choose a value of λ that makes $\alpha(\lambda) = \ell\{\widehat{\Sigma}_\lambda(S); \Sigma\}$ large, where $\ell(\Sigma_1; \Sigma_2) = -\log\det\Sigma_1 - \mathrm{tr}(\Sigma_2\Sigma_1^{-1})$. If we had an independent validation set, we could simply use $\hat{\alpha}(\lambda) = \ell\{\widehat{\Sigma}_\lambda(S); S_{\text{valid}}\}$, which is an unbiased estimator of $\alpha(\lambda)$; however, typically this will not be the case, and so we use a cross-validation approach instead: For $\mathcal{A} \subseteq \{1, \ldots, n\}$, let $S_{\mathcal{A}} = |\mathcal{A}|^{-1} \sum_{i\in\mathcal{A}} x_i x_i^T$ and let $\mathcal{A}^c$ denote the complement of $\mathcal{A}$. Partitioning $\{1, \ldots, n\}$ into k subsets, $\mathcal{A}_1, \ldots, \mathcal{A}_k$, we then compute $\hat{\alpha}_{CV}(\lambda) = k^{-1} \sum_{i=1}^k \ell\{\widehat{\Sigma}_\lambda(S_{\mathcal{A}_i^c}); S_{\mathcal{A}_i}\}$.

To select a value of λ that will generalize well, we choose $\hat{\lambda}_{CV} = \mathrm{argmax}_\lambda \hat{\alpha}_{CV}(\lambda)$. Figure 1 shows 20 realizations of cross-validation for tuning parameter selection. While $\hat{\alpha}_{CV}(\lambda)$ appears to be biased upward for $\alpha(\lambda)$, we see that the value of λ that maximizes $\alpha(\lambda)$ is still well-estimated by cross-validation, especially considering the flatness of $\alpha(\lambda)$ around the maximum.

5. EMPIRICAL STUDY

5·1. *Simulation*

To evaluate the performance of our covariance estimator, which we will refer to as the ℓ_1-*penalized covariance* method, we generate $X_1, \ldots, X_n \sim N_p(0, \Sigma)$, where Σ is a sparse symmetric positive semidefinite matrix. We take $n = 200$ and $p = 100$ and consider three types of covariance graphs, corresponding to different sparsity patterns, considered for example in a 2010 unpublished technical report by Friedman, Hastie, and Tibshirani, available at http://www-stat.stanford.edu/∼tibs/ftp/ggraph.pdf:

I. CLIQUES MODEL: We take $\Sigma = \mathrm{diag}(\Sigma_1, \ldots, \Sigma_5)$, where $\Sigma_1, \ldots, \Sigma_5$ are dense matrices. This corresponds to a covariance graph with five disconnected cliques of size 20.

II. HUBS MODEL: Again $\Sigma = \mathrm{diag}(\Sigma_1, \ldots, \Sigma_5)$, however each submatrix Σ_k is zero except for the last row/column. This corresponds to a graph with five connected components each of which has all nodes connected to one particular node.

III. RANDOM MODEL: We assign $\Sigma_{ij} = \Sigma_{ji}$ to be non-zero with probability 0·02, independently of other elements.

IV. FIRST-ORDER MOVING AVERAGE MODEL: We take $\Sigma_{i,i-1} = \Sigma_{i-1,i}$ to be non-zero for $i = 2, \ldots, p$.

In the first three cases, we generate the non-zero elements as ± 1 with random signs. In the moving average model, we take all non-zero values to be 0·4. For all the models, to ensure that $S \succ 0$ when $n > p$, we then add to the diagonal of Σ a constant so that the resulting matrix has condition number equal to p as in Rothman et al. (2008). Fixing Σ, we then generate ten samples of size n.

We compare three approaches for estimating Σ on the basis of S:

(a) *Simple soft-thresholding* takes $\widehat{\Sigma}_{ij} = \mathcal{S}(S_{ij}, c)$ for $i \neq j$ and $\widehat{\Sigma}_{ii} = S_{ii}$. This method is a special case of Rothman et al. (2009)'s generalized thresholding proposal and does not necessarily lead to a positive definite matrix.

(b) The ℓ_1*-penalized covariance method* with $P_{ij} = 1\{i \neq j\}$ uses Algorithm 1 where an equal penalty is applied to each off-diagonal element.

(c) The ℓ_1*-penalized covariance method* with $P_{ij} = |S_{ij}|^{-1}1\{i \neq j\}$ uses Algorithm 1 with an adaptive lasso penalty on off-diagonal elements. This choice of weights penalizes less strongly those elements that have large values of $|S_{ij}|$. In the regression setting, this modification has been shown to have better selection properties (Zou, 2006).

We evaluate each method on the basis of its ability to correctly identify which elements of Σ are zero and on its closeness to Σ based on both the root-mean-square error, $\|\widehat{\Sigma} - \Sigma\|_F/p$, and entropy loss, $-\log\det(\widehat{\Sigma}\Sigma^{-1}) + \mathrm{tr}(\widehat{\Sigma}\Sigma^{-1}) - p$. The latter is a natural measure for comparing covariance matrices and has been used in this context by Huang et al. (2006).

The first four rows of Fig. 2 show how the methods perform under the models for Σ described above. We vary c and λ to produce a wide range of sparsity levels. From the receiver operating characteristic curves, we find that simple soft-thesholding identifies the correct zeros with comparable accuracy to the ℓ_1-penalized covariance approaches (b) and (c). Relatedly, Friedman, Hastie, and Tibshirani, in their 2010 technical report, observe with surprise the effectiveness of soft-thesholding of the empirical correlation matrix for identifying the zeros in the inverse covariance matrix. In terms of root-mean-square error, all three methods perform similarly in the cliques model (I) and random model (III). In both these situations, method (b) dominates in the denser realm while method (a) does best in the sparser realm. In the moving average model (IV), both soft-thresholding (a) and the adaptive ℓ_1-penalized covariance method (c) do better in the sparser realm, with the latter attaining the lowest error. For the hubs model (II), ℓ_1-penalized

covariance (b) attains the best root-mean-square error across all sparsity levels. In terms of entropy loss there is a pronounced difference between the ℓ_1-penalized covariance methods and soft-thresholding. In particular, we find that the former methods get much closer to the truth in this sense than soft-thresholding in all four cases. This behavior reflects the difference in nature between minimizing a penalized Frobenius distance, as is done with soft-thresholding, and minimizing a penalized negative-log-likelihood, as in (1). The rightmost plot shows that for the moving average model (IV) soft-thresholding produces covariance estimates that are not positive semidefinite for some sparsity levels. When the estimate is not positive definite, we do not plot the entropy loss. By contrast, the ℓ_1-penalized covariance method is guaranteed to produce a positive definite estimate regardless of the choice of P. The bottom row of Fig. 2 shows the performance of the ℓ_1-penalized covariance method when S is not full-rank. In particular, we take $n = 50$ and $p = 100$. The receiver-operating characteristic curves for all three methods decline greatly in this case, reflecting the difficulty of estimation when $p > n$. Despite trying a range of values of λ, we find that the ℓ_1-penalized covariance method does not produce a uniform range of sparsity levels, but rather jumps from being about 33% zero to 99% zero. As with model (IV), we find that soft-thresholding leads to estimates that are not positive semidefinite, in this case for a wide range of sparsity levels.

5·2. *Cell signalling dataset*

We apply our ℓ_1-penalized covariance method to a dataset that has previously been used in the sparse graphical model literature (Friedman et al., 2007). The data consists of flow cytometry measurements of the concentrations of $p = 11$ proteins in $n = 7466$ cells (Sachs et al., 2005). Figure 3 compares the covariance graphs learned by the ℓ_1-penalized covariance method to the Markov network learned by the graphical lasso (Friedman et al., 2007). The two types of graph have different interpretations: if the estimated covariance graph has a missing edge between

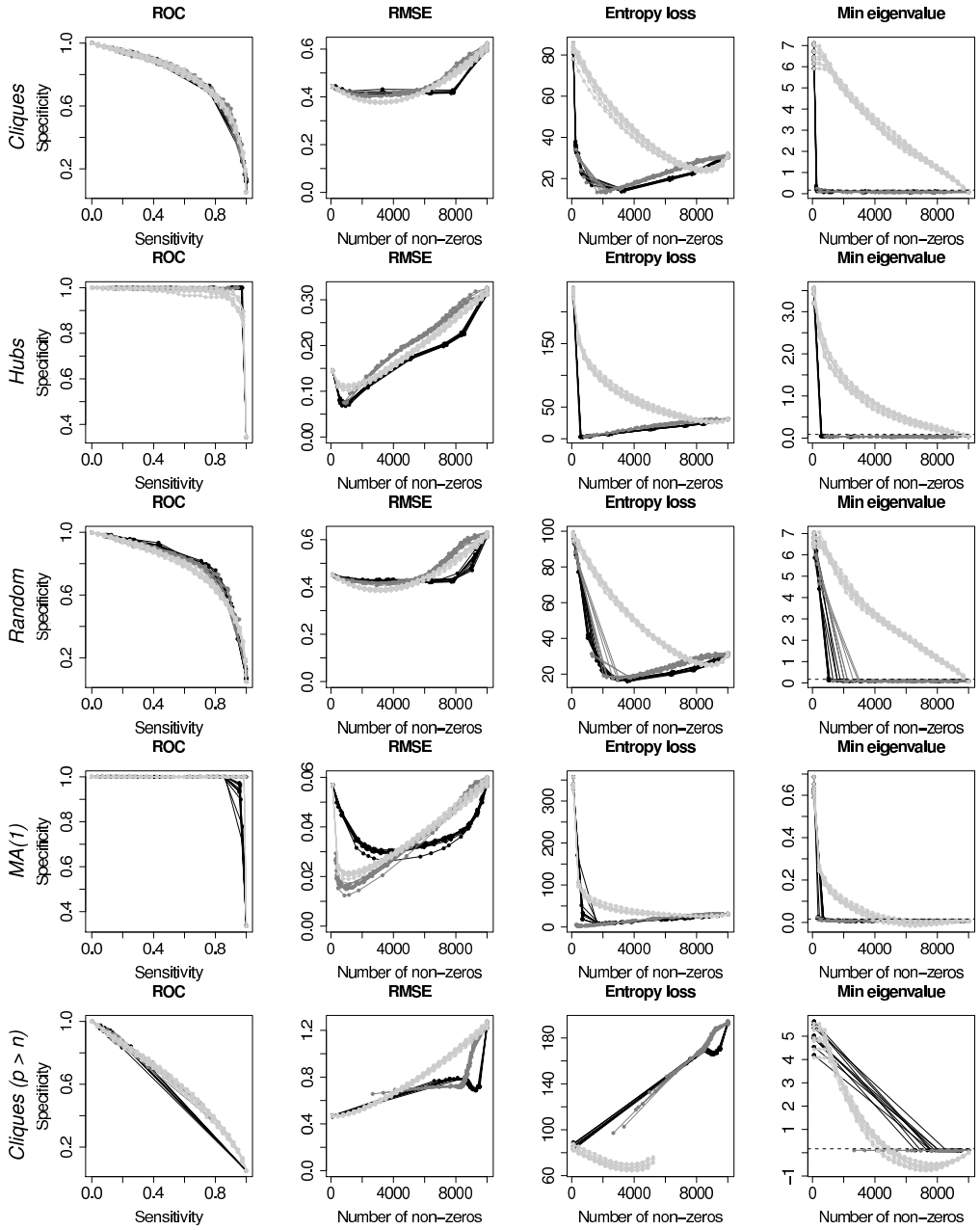

Fig. 2. Simulation study: Black and dark-grey curves are the ℓ_1-penalized methods with equal penalty on off-diagonals and with an adaptive lasso penalty, respectively, and the light-grey curves are soft-thresholding of the non-diagonal elements of S. From top to bottom, the rows show the (I) cliques, (II) hubs, (III) random, and (IV) first-order moving average, (V) cliques with $p > n$ models for Σ. From left to right, the columns show the receiver-operating characteristic curves, root-mean-square errors, entropy loss, and minimum eigenvalue of the estimates. The horizontal dashed line shows the minimum eigenvalue

two proteins, then we are stating that the concentration of one protein gives no information about the concentration of another. On the other hand, a missing edge in the Markov network means that, conditional on all other proteins' concentrations, the concentration of one protein gives no information about the concentration of another. Both of these statements assume that the data are multivariate Gaussian. The right panel of Fig. 3 shows the extent to which similar protein pairs are identified by the two methods for a series of sparsity levels. We compare the observed proportion of co-occurring edges to a null distribution in which two graphs are selected independently from the uniform distribution of graphs having a certain number of edges. The dashed and dotted lines show the mean and 0·025- and 0·975-quantiles of the null distribution, respectively, which for k-edge graphs is a $\text{Hypergeometric}\{p(p-1)/2, k, k\}/k$ distribution. We find that the presence of edges in the two types of graphs is anti-correlated relative to the null, emphasizing the difference between covariance and Markov graphical models. It is therefore important that a biologist understand the difference between these two measures of association since the edges estimated to be present will often be quite different.

6. EXTENSIONS AND OTHER CONVEX PENALTIES

Chaudhuri et al. (2007) propose a method for performing maximum likelihood over a fixed covariance graph, i.e., subject to a prespecified, fixed set of zeros, $\Omega = \{(i,j) : \Sigma_{ij} = 0\}$. This problem can be expressed in our form by taking P defined by $P_{ij} = 1$ if $(i,j) \in \Omega$ and $P_{ij} = 0$ otherwise, and λ sufficiently large. In this case, (1) is maximum likelihood subject to the desired sparsity pattern. The method presented in this paper therefore gives an alternative method for approximately solving this fixed-zero problem. In practice, we find that this method achieves very similar values of the likelihood as the method of Chaudhuri et al. (2007), which is implemented in the `R` package `ggm`.

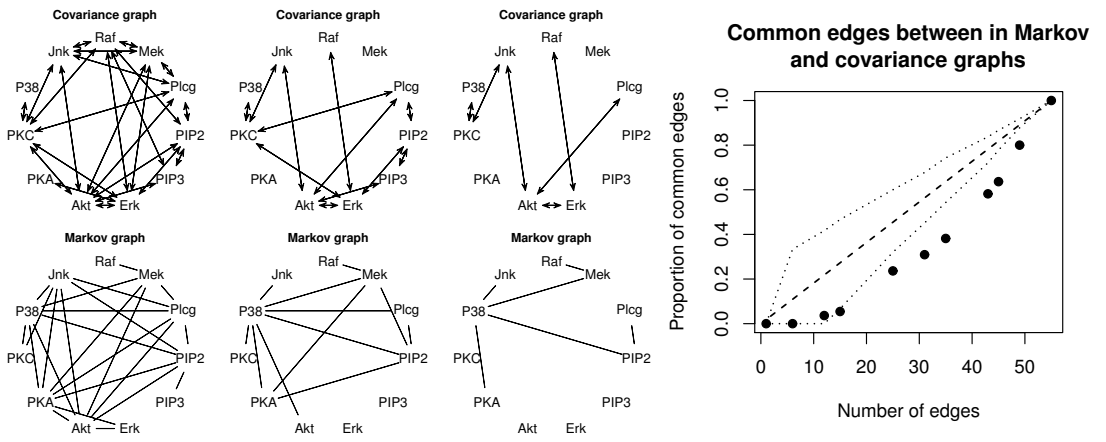

Fig. 3. Cell signalling dataset. (Left) Comparison of our algorithm's solution to the sparse covariance maximum likelihood problem (1) to the graphical lasso's solution to the sparse inverse covariance maximum likelihood problem. Here we adopt the convention of using bi-directed edges for covariance graphs (e.g., Chaudhuri et al. 2007). Different values of the regularization parameter were chosen to give same sparsity levels. (Right) Each black circle shows the proportion of edges shared by the covariance graph from our algorithm to the Markov graph from the graphical lasso at a given sparsity level. The dashed and dotted lines show the mean and 0·025- and 0·975-quantiles of the null distribution, respectively.

In deriving the majorize-minimize algorithm of (2), we only used that $\|P * \Sigma\|_1$ is convex. Thus, the approach in (2) extends straightforwardly to any convex penalty. For example, in some situations we may desire certain groups of edges to be simultaneously missing from the covariance graph. Given a collection of such sets $\mathcal{G}_1, \ldots, \mathcal{G}_K \subset \{1, \ldots, p\}^2$, we may apply a group lasso penalty:

$$\text{Minimize}_{\Sigma \succ 0} \left\{ \log\det\Sigma + \text{tr}(\Sigma^{-1}S) + \lambda \sum_{k=1}^{K} |\mathcal{G}_k|^{1/2} \|\text{vec}(\Sigma)_{\mathcal{G}_k}\|_2 \right\}, \tag{6}$$

where $\text{vec}(\Sigma)_{\mathcal{G}_k}$ denotes the vector formed by the elements of Σ in $\mathcal{G}_k$. For example in some instances such as in time series data, the variables have a natural ordering and we may desire a banded sparsity pattern (Rothman et al., 2010). In such a case, one could take $\mathcal{G}_k = \{(i,j) : |i-j| = k\}$ for $k = 1, \ldots, p-1$. Estimating the kth band as zero would correspond to a model in which a variable is marginally independent of the variable k time units earlier.

As another example, we could take $\mathcal{G}_k = \{(k,i) : i \neq k\} \cup \{(i,k) : i \neq k\}$ for $k = 1, \ldots, p$. This encourages a node-sparse graph considered by Friedman, Hastie, and Tibshirani, in their 2010 technical report, in the case of the inverse covariance matrix. Estimating $\Sigma_{ij} = 0$ for all $(i,j) \in \mathcal{G}_k$ corresponds to the model in which variable k is independent of all others. It should be noted however that a variable's being marginally independent of all others is equivalent to its being conditionally independent of all others. Therefore, if node-sparsity in the covariance graph is the only goal, i.e., no other penalties on Σ are present, a better procedure would be to apply this group lasso penalty to the inverse covariance, thereby admitting a convex problem.

We conclude with an extension that may be worth pursuing. A difficulty with (1) is that it is not convex and therefore any algorithm that attempts to solve it may converge to a suboptimal local minimum. Exercise 7·4 of Boyd & Vandenberghe (2004), on page 394, remarks that the log-likelihood $\ell(\Sigma)$ is concave on the convex set $\mathcal{C}_0 = \{\Sigma : 0 \prec \Sigma \preceq 2S\}$. This fact can be verified by noting that over this region the positive curvature of $\text{tr}(\Sigma^{-1}S)$ exceeds the negative curvature of $\log\det\Sigma$. This suggests a related estimator that is the result of a convex optimization problem: Let $\widehat{\Sigma}_c$ denote a solution to

$$\text{Minimize}_{0 \prec \Sigma \preceq 2S} \left\{ \log\det\Sigma + \text{tr}(\Sigma^{-1}S) + \lambda \|P * \Sigma\|_1 \right\}. \tag{7}$$

While of course we cannot in general expect $\widehat{\Sigma}_c$ to be a solution to (1), adding this constraint may not be unreasonable. In particular, if $n, p \to \infty$ with $p/n \to y \in (0,1)$, then by a result of Silverstein (1985), $\lambda_{\min}(\Sigma_0^{-1/2} S \Sigma_0^{-1/2}) \to (1 - y^{1/2})^2$ almost surely, where $S \sim \text{Wishart}(\Sigma_0,\ n)$.

It follows that the constraint $\Sigma_0 \preceq 2S$ will hold almost surely in this limit if $(1 - y^{1/2})^2 > 0{\cdot}5$, i.e., $y < 0{\cdot}085$. Thus, in the regime that n is large and p does not exceed $0{\cdot}085n$, the constraint set of (7) contains the true covariance matrix with high probability.

ACKNOWLEDGEMENT

We thank Ryan Tibshirani and Jonathan Taylor for useful discussions and two anonymous reviewers and an editor for helpful comments. Jacob Bien was supported by the Urbanek Family Stanford Graduate Fellowship and the Lieberman Fellowship; Robert Tibshirani was partially supported by the National Science Foundation and the National Institutes of Health.

SUPPLEMENTARY MATERIAL

Included on *Biometrika*'s website, Supplementary Material shows a simulation evaluating the performance of our estimator as n increases.

APPENDIX 1

Convex plus concave

Examining the objective of problem (1) term by term, we observe that $\log\det\Sigma$ is concave while $\mathrm{tr}(\Sigma^{-1}S)$ and $\lambda\|\Sigma\|_1$ are convex in Σ. The second derivative of $\log\det\Sigma$ is $-\Sigma^{-2}$, which is negative definite, from which it follows that $\log\det\Sigma$ is concave. As shown in example 3·4 of Boyd & Vandenberghe (2004), on page 76, $X_i^T\Sigma^{-1}X_i$ is jointly convex in X_i and Σ. Since $\mathrm{tr}(\Sigma^{-1}S) = (1/n)\sum_{i=1}^n X_i^T\Sigma^{-1}X_i$, it follows that $\mathrm{tr}(\Sigma^{-1}S)$ is the sum of convex functions and therefore is itself convex.

APPENDIX 2

Justifying the Lipschitz claim

Let $L(\Sigma) = \mathrm{tr}(\Sigma_0^{-1}\Sigma) + \mathrm{tr}(\Sigma^{-1}S)$ denote the differentiable part of the majorizing function of (1). We wish to prove that $dL(\Sigma)/d\Sigma = \Sigma_0^{-1} - \Sigma^{-1}S\Sigma^{-1}$ is Lipschitz continuous over the region of the optimization problem. Since this is not the case for $\lambda_{\min}(\Sigma) \to 0$, we begin by showing that the constraint region can be restricted to $\Sigma \succeq \delta I_p$.

PROPOSITION 1. *Let $\widetilde{\Sigma}$ be an arbitrary positive definite matrix, e.g., $\widetilde{\Sigma} = S$. Problem* (1) *is equivalent to*

$$\mathrm{Minimize}_{\Sigma \succeq \delta I_p} \left\{ \log\det\Sigma + \mathrm{tr}(\Sigma^{-1}S) + \lambda \|P * \Sigma\|_1 \right\} \tag{A1}$$

for some $\delta > 0$ that depends on $\lambda_{\min}(S)$ and $f(\widetilde{\Sigma})$.

Proof. Let $g(\Sigma) = \log\det\Sigma + \mathrm{tr}(\Sigma^{-1}S)$ denote the differentiable part of the objective function $f(\Sigma) = g(\Sigma) + \lambda\|P * \Sigma\|_1$, and let $\Sigma = \sum_{i=1}^p \lambda_i u_i u_i^T$ be the eigendecomposition of Σ with $\lambda_1 \geq \cdots \geq \lambda_p$.

Given a point $\widetilde{\Sigma}$ with $f(\widetilde{\Sigma}) < \infty$, we can write (1) equivalently as

$$\mathrm{Minimize}\ f(\Sigma) \text{ subject to } \Sigma \succ 0, f(\Sigma) \leq f(\widetilde{\Sigma}).$$

We show in what follows that the constraint $f(\Sigma) \leq f(\widetilde{\Sigma})$ implies $\Sigma \succeq \delta I_p$ for some $\delta > 0$.

Now, $g(\Sigma) = \sum_{i=1}^p \log\lambda_i + u_i^T S u_i/\lambda_i = \sum_{i=1}^p h(\lambda_i; u_i^T S u_i)$, where $h(x;a) = \log x + a/x$. For $a > 0$, the function h has a single stationary point at a, where it attains a minimum value of $\log a + 1$, has $\lim_{x\to 0^+} h(x;a) = +\infty$ and $\lim_{x\to\infty} h(x;a) = +\infty$, and is convex for $x \leq 2a$. Also, $h(x;a)$ is increasing in a for all $x > 0$. From these properties and the fact that $\lambda_{\min}(S) = \min_{\|u\|^2=1} u^T S u$, it follows that

$$\begin{aligned} g(\Sigma) &\geq \sum_{i=1}^p h\{\lambda_i; \lambda_{\min}(S)\} \geq h\{\lambda_p; \lambda_{\min}(S)\} + \sum_{i=1}^{p-1} h\{\lambda_{\min}(S); \lambda_{\min}(S)\} \\ &= h\{\lambda_p; \lambda_{\min}(S)\} + (p-1)\{\log\lambda_{\min}(S) + 1\}. \end{aligned}$$

Thus, $f(\Sigma) \leq f(\widetilde{\Sigma})$ implies $g(\Sigma) \leq f(\widetilde{\Sigma})$ and so

$$h\{\lambda_p; \lambda_{\min}(S)\} + (p-1)\{\log \lambda_{\min}(S) + 1\} \leq f(\widetilde{\Sigma}).$$

This constrains λ_p to lie in an interval $[\delta_-, \delta_+] = \{\lambda : h\{\lambda; \lambda_{\min}(S)\} \leq c\}$, where $c = f(\widetilde{\Sigma}) - (p-1)\{\log \lambda_{\min}(S) + 1\}$ and $\delta_-, \delta_+ > 0$. We compute δ_- using Newton's method. To see that $\delta_- > 0$, note that h is continuous and monotone decreasing on $(0, a)$ and $\lim_{x \to 0^+} h(x; a) = +\infty$.

As $\lambda_{\min}(S)$ increases, $[\delta_-, \delta_+]$ becomes narrower and more shifted to the right. The interval also narrows as $f(\widetilde{\Sigma})$ decreases.

For example, we may take $\widetilde{\Sigma} = \mathrm{diag}(S_{11}, \ldots, S_{pp})$ and $P = 11^T - I_p$, which yields

$$h\{\lambda_p, \lambda_{\min}(S)\} \leq \sum_{i=1}^{p} \log\{S_{ii}/\lambda_{\min}(S)\} + \log \lambda_{\min}(S) + 1.$$

We next show that $dL(\Sigma)/d\Sigma = \Sigma_0^{-1} - \Sigma^{-1} S \Sigma^{-1}$ is Lipschitz continuous on $\Sigma \succ \delta I_p$ by bounding its first derivative. Using the product rule for matrix derivatives, we have

$$\begin{aligned}
\frac{d}{d\Sigma}(\Sigma_0^{-1} - \Sigma^{-1} S \Sigma^{-1}) &= -(\Sigma^{-1} S \otimes I_p)(-\Sigma^{-1} \otimes \Sigma^{-1}) - (I_p \otimes \Sigma^{-1})\{(I_p \otimes S)(-\Sigma^{-1} \otimes \Sigma^{-1})\} \\
&= (\Sigma^{-1} S \Sigma^{-1}) \otimes \Sigma^{-1} + \Sigma^{-1} \otimes (\Sigma^{-1} S \Sigma^{-1}).
\end{aligned}$$

We bound the spectral norm of this matrix:

$$\begin{aligned}
\left\| \frac{d}{d\Sigma} \frac{dL}{d\Sigma} \right\|_2 &\leq \|(\Sigma^{-1} S \Sigma^{-1}) \otimes \Sigma^{-1}\|_2 + \|\Sigma^{-1} \otimes \Sigma^{-1} S \Sigma^{-1}\|_2 \\
&\leq 2\|\Sigma^{-1} S \Sigma^{-1}\|_2 \|\Sigma^{-1}\|_2 \\
&\leq 2\|S\|_2 \|\Sigma^{-1}\|_2^3.
\end{aligned}$$

The first inequality follows from the triangle inequality; the second uses the fact that the eigenvalues of $A \otimes B$ are the pairwise products of the eigenvalues of A and B; the third uses the sub-multiplicativity of the spectral norm. Finally, $\Sigma \succeq \delta I_p$ implies that $\Sigma^{-1} \preceq \delta^{-1} I_p$ from which it follows that

$$\left\| \frac{d}{d\Sigma} \frac{dL}{d\Sigma} \right\|_2 \leq 2\|S\|_2 \delta^{-3}.$$

APPENDIX 3

Alternating direction method of multipliers for solving (3)

To solve (3), we repeat until convergence:

1. Diagonalize $\{\Sigma - t(\Sigma_0^{-1} - \Sigma^{-1}S\Sigma^{-1}) + \rho\Theta^k - Y^k\}/(1+\rho) = UDU^T$;
2. $\Sigma^{k+1} \leftarrow UD_\delta U^T$ where $D_\delta = \text{diag}\{\max(D_{ii}, \delta)\}$;
3. $\Theta^{k+1} \leftarrow \mathcal{S}\{\Sigma^{k+1} + Y^k/\rho,\ (\lambda/\rho)P\}$, i.e., soft-threshold elementwise;
4. $Y^{k+1} \leftarrow Y^k + \rho(\Sigma^{k+1} - \Theta^{k+1})$.

REFERENCES

AN, L. & TAO, P. (2005). The dc (difference of convex functions) programming and dca revisited with dc models of real world nonconvex optimization problems. *Annals of Operations Research* **133**, 23–46.

ARGYRIOU, A., HAUSER, R., MICCHELLI, C. & PONTIL, M. (2006). A dc-programming algorithm for kernel selection. In *Proceedings of the 23rd international conference on Machine learning*. ACM.

BANERJEE, O., EL GHAOUI, L. E. & D'ASPREMONT, A. (2008). Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *Journal of Machine Learning Research* **9**, 485–516.

BECK, A. & TEBOULLE, M. (2009). A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences* **2**, 183–202.

BOYD, S., PARIKH, N., CHU, E., PELEATO, B. & ECKSTEIN, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning* **3**, 1–124.

BOYD, S. & VANDENBERGHE, L. (2004). *Convex Optimization*. Cambridge University Press.

BUTTE, A. J., TAMAYO, P., SLONIM, D., GOLUB, T. R. & KOHANE, I. S. (2000). Discovering functional relationships between RNA expression and chemotherapeutic susceptibility using relevance networks. *Proceedings of the National Academy of Sciences of the United States of America* **97**, 12182–12186.

CHAUDHURI, S., DRTON, M. & RICHARDSON, T. S. (2007). Estimation of a covariance matrix with zeros. *Biometrika* **94**, 199–216.

DE LEEUW, J. & MAIR, P. (2009). Multidimensional scaling using majorization: SMACOF in R. *Journal of Statistical Software* **31**, 1–30.

DEMPSTER, A. P. (1972). Covariance selection. *Biometrics* **28**, 157–175.

DRTON, M. & RICHARDSON, T. S. (2008). Graphical methods for efficient likelihood inference in gaussian covariance models. *J. Mach. Learn. Res.* **9**, 893–914.

FAZEL, M., HINDI, H. & BOYD, S. (2003). Log-det heuristic for matrix rank minimization with applications to hankel and euclidean distance matrices. In *American Control Conference, 2003. Proceedings of the 2003*, vol. 3. IEEE.

FRIEDMAN, J., HASTIE, T. & TIBSHIRANI, R. (2007). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics* **9**, 432–441.

HORST, R. & THOAI, N. V. (1999). Dc programming: Overview. *Journal of Optimization Theory and Applications* **103**, 1–43.

HUANG, J., LIU, N., POURAHMADI, M. & LIU, L. (2006). Covariance matrix selection and estimation via penalised normal likelihood. *Biometrika* **93**, 85.

HUNTER, D. R. & LI, R. (2005). Variable selection using MM algorithms. *Ann Stat* **33**, 1617–1642.

LAM, C. & FAN, J. (2009). Sparsistency and rates of convergence in large covariance matrix estimation. *Annals of Statistics* **37**, 4254–4278.

LANGE, K. (2004). *Optimization*. New York: Springer-Verlag.

MEINSHAUSEN, N. & BÜHLMANN, P. (2006). High dimensional graphs and variable selection with the lasso. *Annals of Statistics* **34**, 1436–1462.

NESTEROV, Y. (2005). Smooth minimization of non-smooth functions. *Mathematical Programming* **103**, 127–152.

ROTHMAN, A., LEVINA, E. & ZHU, J. (2008). Sparse permutation invariant covariance estimation. *Electronic Journal of Statistics* **2**, 494–515.

ROTHMAN, A., LEVINA, E. & ZHU, J. (2010). A new approach to Cholesky-based covariance regularization in high dimensions. *Biometrika* **97**, 539.

ROTHMAN, A. J., LEVINA, E. & ZHU, J. (2009). Generalized thresholding of large covariance matrices. *Journal of the American Statistical Association* **104**, 177–186.

SACHS, K., PEREZ, O., PE'ER, D., LAUFFENBURGER, D. & NOLAN, G. (2005). Causal protein-signaling networks derived from multiparameter single-cell data. *Science* **308**, 523–529.

SILVERSTEIN, J. (1985). The smallest eigenvalue of a large dimensional wishart matrix. *The Annals of Probability* **13**, 1364–1368.

SRIPERUMBUDUR, B. & LANCKRIET, G. (2009). On the convergence of the concave-convex procedure. In *Advances in Neural Information Processing Systems 22*, Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams & A. Culotta, eds. pp. 1759–1767.

TIBSHIRANI, R. (1996). Regression shrinkage and selection via the lasso. *J. Royal. Statist. Soc. B.* **58**, 267–288.

YUAN, M. & LIN, Y. (2007). Model selection and estimation in the Gaussian graphical model. *Biometrika* **94**, 19–35.

YUILLE, A. L. & RANGARAJAN, A. (2003). The concave-convex procedure. *Neural Computation* **15**, 915–936.

ZHANG, T. (2010). Analysis of multi-stage convex relaxation for sparse regularization. *Journal of Machine Learning Research* **11**, 1081–1107.

ZOU, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association* **101**, 1418–1429.

[*Received MONTH YEAR. Revised MONTH YEAR*]