Supplementary materials for this article are available online.
Please go to www.tandfonline.com/r/JCGS

Estimation of Symmetry-Constrained Gaussian Graphical Models: Application to Clustered Dense Networks

Xin GAO and Hélène MASSAM

We propose a model selection algorithm for high-dimensional clustered data. Our algorithm combines a classical penalized likelihood method with a composite likelihood approach in the framework of colored graphical Gaussian models. Our method is designed to identify high-dimensional dense networks with a large number of edges but sparse edge classes. Its empirical performance is demonstrated through simulation studies and a network analysis of a gene expression dataset.

Key Words: Concentration matrix; Gene networks; Model selection; Partial correlation matrix; Penalized estimation; Social networks.

1. INTRODUCTION

The analysis of complex high-dimensional data is one of the main problems of modern statistics. To identify the dependencies or independencies among continuous variables, the main parameter of interest is the covariance matrix between these variables. Conditional independence between variables can be represented by means of a graph where each vertex represents a variable and where the absence of an edge (i, j) implies the conditional independence of the variable X_i and the variable X_j given all the other variables. Gaussian models with conditional independences between selected pairs of variables represented by means of a graph are called graphical Gaussian models and have been one of the main tools of modern statistics for the analysis of high-dimensional data. Since, in the case of Gaussian data, conditional independence between variables translates into fixed zeros in the precision (inverse covariance) matrix, graphical Gaussian models allow for a substantial dimensionality reduction.

In real applications modeled with a graphical Gaussian model, there often exist additional symmetry constraints on the parameters. For example, genes belonging to the same functional or structural group may behave in a similar manner and thus share similar network properties (Toh and Horimoto 2002). Similarly, in the analysis of social networks

Xin Gao (E-mail: *xingao@mathstat.yorku.ca*) and Hélène Massam (E-mail: *massamh@yorku.ca*), Department of Mathematics and Statistics, York University, Toronto, ON, Canada.

Journal of Computational and Graphical Statistics, Volume 24, Number 4, Pages 909–929
DOI: 10.1080/10618600.2014.937811
Color versions of one or more of the figures in the article can be found online at *www.tandfonline.com/r/jcgs*.

data, a fundamental problem is to estimate the friendship patterns among individuals, where individuals belonging to the same geographical region or social group can be considered as nodes from the same cluster in the network (Ma, Gong, and Bohnert 2007). For such clustered networks, we may assume that the edges linking the same pair of clusters share similar properties and that the nodes belonging to the same cluster also have similar properties. These restrictions will result in restrictions on the model parameters and further dimensionality reduction.

To express these additional symmetry constraints, Højsgaard and Lauritzen (2008) introduced colored graphical Gaussian models. Two of the models they define are the so-called RCON and RCOR models which are graphical Gaussian models Markov with respect to an undirected graph, with additional symmetry constraints on, respectively, the entries of the concentration matrix $\Theta = \Sigma^{-1}$ and the partial correlation matrix $P = \text{diag}(\Theta)^{-1/2}\Theta\text{diag}(\Theta)^{-1/2}$. Likelihood estimation in both models can be obtained through Newton iteration or partial maximization (Højsgaard and Lauritzen 2007). However, at each iterative step, such an algorithm involves the inversion of the concentration matrix, which can be computationally costly for large matrices.

In this article, we work with colored graphical models but use a penalized likelihood approach which performs model selection and estimation simultaneously. In practice, either the clusters are known or we identify them through one of the many known clustering techniques (see Section 4). Once the clusters of variables are identified, the model selection is done within the class of colored graphical Gaussian models with given clusters. The penalized likelihood function will select edges within and between given clusters. We circumvent the problem of the inversion of large matrices for the computation of the maximum likelihood estimate by using a composite likelihood function rather than the likelihood function. Following Besag (1974) and Friedman, Hastie, and Tibshirani (2010), the factors of the composite likelihood are the likelihoods derived from the conditional distributions of the variable X_i given all the other variables. These conditional distributions are univariate Gaussian distributions, where the conditional mean and variance can be formulated as functions of the other variables and the original parameters. The idea of forming fully conditional composite likelihood can be viewed as performing all the linear regressions of one variable on all the other variables in the graphical model. The composition of all these conditional likelihoods will naturally yield a composite likelihood as a function of the entries of Θ and P, the parameters of the RCON and RCOR models, respectively.

In the literature, various penalty functions and algorithms have been proposed to estimate and select unconstrained Gaussian graphical models. Yuan and Lin (2007) proposed penalized likelihood methods for estimating the concentration matrix with the L_1 LASSO penalty. Banerjee, Ghaoui, and D'Aspremont (2007) proposed a block-wise updating algorithm for the estimation of the concentration matrix. Further along this line, Friedman, Hastie, and Tibshirani (2008) proposed the graphical LASSO algorithm through a coordinate-wise updating scheme. Fan, Feng, and Wu (2009) proposed to estimate the concentration matrix using the adaptive LASSO and the smoothly clipped absolute deviation (SCAD) penalty to reduce the bias problem. Zhang (2010) proposed the minimax concave penalty (MCP). The MCP penalty applied in the regression setting and the graphical model setting have been investigated by Breheny and Huang (2011) and Mazumder, Friedman, and Hastie (2011), respectively.

The remainder of this article is organized as follows. In Section 2, we recall some basic properties of colored graphical Gaussian models and composite likelihood. In Sections 3.1 and 3.2, we derive the penalized likelihood function for the RCON and RCOR models and present the coordinate descent algorithm with the LASSO, SCAD, and MCP penalties to perform the estimation. In Section 3.3, we investigate the asymptotic behavior of the penalized composite likelihood estimate and establish its ORACLE property. In Section 4, we discuss ways of determining the color classes if they are not known a priori. In Section 5, simulation studies are presented to demonstrate the empirical performance of the method. In Section 6, we apply our method to a clustered microarray dataset to model the network linking individual genes.

2. PRELIMINARIES

2.1 RCON AND RCOR GRAPHICAL GAUSSIAN MODELS

The reader is referred to Højsgaard and Lauritzen (2008) for a detailed description of colored graphical Gaussian models. We will recall here their main features. Let $\mathcal{G} = (V, E)$ be an undirected graph, where $V = \{1, 2, \ldots, p\}$ is the set of vertices and E the set of edges. If $X = (X_1, \ldots, X_p)^t$ follows the $N_p(0, \Sigma)$ distribution for some positive definite covariance matrix Σ, it is a well-known result that X_i is independent of X_j given all the other variables $X_{V\setminus\{i,j\}}$ if and only if θ_{ij}, the ijth entry of Θ, is 0 where $\Theta = \Sigma^{-1}$ (Lauritzen 1996) . For a given graph $\mathcal{G}$, we therefore consider the cone $P_{\mathcal{G}}$ of positive definite matrices with zero ijth entry whenever the edge (i, j) does not belong to $\mathcal{G}$. The graphical Gaussian model Markov with respect to $\mathcal{G}$ is the model

$$\{N_p(0, \Sigma), \quad \Theta = \Sigma^{-1} \in P_{\mathcal{G}}\}. \tag{1}$$

We note that, here and in the remainder of the article, we take μ to be equal to 0 without any loss of generality. If $\mu \neq 0$, we simply center our data. Given two variables X_i and X_j, let $\theta_{(ij)}$ be the 2×2 matrix $(\theta_{lk})_{l=i,j,\, k=i,j}$. The conditional covariance matrix of $(X_i, X_j)^t$ given $X_{V\setminus\{i,j\}}$ is

$$\Sigma_{(ij)\cdot V\setminus\{i,j\}} = \frac{1}{\det(\theta_{(ij)})} \begin{pmatrix} \theta_{jj} & -\theta_{ij} \\ -\theta_{ji} & \theta_{ii} \end{pmatrix},$$

and similarly, the conditional covariance σ^{ii} of X_i given all the other variables is

$$\sigma^{ii} = \Sigma_{ii\cdot V\setminus\{i\}} = [(\Sigma^{-1})_{ii}]^{-1} = \frac{1}{\theta_{ii}}. \tag{2}$$

The $(p-1)$-dimensional vector $\Sigma_{jj}\Sigma^{-1}_{j,V\setminus\{j\}}$ of regression coefficients β_{ij} of X_j on the other variables is equal to

$$(\beta_{ij},\ i \in V \setminus \{j\}) = -\theta_{V\setminus\{j\},j}\theta_{jj}^{-1}. \tag{3}$$

It follows that, for model (1), if $(i, j) \notin E$, $\beta_{ij} = 0$. Later in the article, we will use the convenient notation B_j for the p-dimensional vector

$$B_j = (\beta_{1j}, \ldots, \beta_{j-1,j}, 0, \beta_{j+1,j}, \ldots, \beta_{pj}) \tag{4}$$

of regression coefficients of X_j on X_i, $i = 1, \ldots, j-1, j+1, \ldots, p$ augmented by the entry 0 in the jth spot. Also for model (1), the conditional correlation coefficient of X_j and X_k given $X_{V\setminus\{j,k\}}$, that is, the partial correlation coefficient between X_j and X_k, is $\rho_{jk} = -\frac{\theta_{jk}}{\sqrt{\theta_{jj}\theta_{kk}}}$ and, therefore, the partial correlation matrix $P = (\rho_{ij})_{1 \le i,j \le p} = \operatorname{diag}(\Theta)^{-1/2}\Theta \operatorname{diag}(\Theta)^{-1/2}$ and the concentration matrix Θ have the same zeros.

Now, let $\mathcal{V} = \{V_1, \ldots, V_k\}$ form a partition of $V = \{1, \ldots, p\}$ and let $\mathcal{E} = \{E_1, \ldots, E_l\}$ form a partition of the edge set E. If all the vertices belonging to an element V_i of $\mathcal{V}$ have the same color, we say that $\mathcal{V} = \{V_1, \ldots, V_k\}$ is a coloring of V. Similarly if all the edges belonging to an element E_i of $\mathcal{E}$ have the same color, we say that $\mathcal{E}$ is a coloring of the edges of $\mathcal{G}$ and that $(\mathcal{V}, \mathcal{E})$ is a colored graph.

Consider model (1). If, for $\Theta = \Sigma^{-1} \in P_{\mathcal{G}}$, we impose the further restrictions that

1. if m is a vertex class in $\mathcal{V}$, then for all $i \in m$, θ_{ii} are equal;
2. and, if s is an edge class in $\mathcal{E}$, then for all $(i, j) \in s$, the entries θ_{ij} of the concentration matrix are equal,

then model (1) becomes a colored graphical Gaussian model called the RCON$(\mathcal{V}, \mathcal{E})$ model. We use the notation

$$\theta = (\theta_{E_1}, \ldots, \theta_{E_l}, \theta_{V_1}, \ldots, \theta_{V_k}) \tag{5}$$

for the vector of free parameters in Θ.

Let us now define an RCOR model. Given $\mathcal{V}$ and $\mathcal{E}$ as above, if a model of the type (1) satisfies the two additional properties that

1. if $m \in \mathcal{V}$, then for all $i \in m$, θ_{ii} are equal;
2. and, if $s \in \mathcal{E}$, then for all $(i, j) \in s$, the entries ρ_{ij} of the partial correlation matrix are equal,

then this model becomes the RCOR$(\mathcal{V}, \mathcal{E})$ model. We use the notation

$$\rho = (\rho_{E_1}, \ldots, \rho_{E_l}) \tag{6}$$

for the vector of free correlation parameters in P. We note that an RCON model forms a natural exponential family while an RCOR model forms a curved exponential family. In the models defined above, the color for the vertices and the edges are not related to each other. But we could impose further constraints linking those two colorings. For example, we could further assume that different edges linking two vertices belonging to the same two vertex color classes have to belong to the same edge color class. Such constraints only affect the partitioning of the vertices and edges, but do not affect the estimation.

2.2 Composite Likelihood

The estimation of Gaussian graphical model has been mainly based on the likelihood method. An alternative method of estimation based on composite likelihood (henceforth abbreviated CL) has drawn much attention in recent years (Cox and Reid 2004; Varin 2008).

It has been demonstrated to possess good theoretical properties, such as consistency for the parameter estimation, and can be used to establish hypothesis testing procedures. Let $Y = (Y_1, \ldots, Y_p)^T$ be a random vector in $\mathcal{R}^p$. Let $\{f(y;\theta),\ y \in \mathcal{Y}, \theta \in \mathcal{T}\}$ be a parametric model, with $\mathcal{Y} \subseteq \mathcal{R}^p$, $\mathcal{T} \subseteq \mathcal{R}^q$, $p \geq 1$, and $q \geq 1$. Let $\{\mathcal{A}_i,\ i = 1, \ldots, m\}$ be a set of events with associated likelihood functions $L_i(\theta; y) = f(y \in \mathcal{A}_i; \theta)$. Then, according to Varin (2008), see also Lindsay (1988), a composite likelihood (CL) is the weighted product of the likelihoods corresponding to each event,

$$L_c(\theta; y) = \prod_{i=1}^{m} f(y \in \mathcal{A}_i;\ \theta)^{w_i},$$

where $w_i, i = 1, \ldots, m$ are positive weights. As the composite score function $\partial \log L_c(\theta; y)/\partial\theta$ is a linear combination of several likelihood score functions, its expectation is equal to zero under the usual regularity conditions (Varin 2008). Even though the composite likelihood is not a real likelihood, the maximum composite likelihood estimate is still consistent for the true parameter. The asymptotic covariance matrix of the maximum composite likelihood estimator takes the form of the inverse of the Godambe information: $H(\theta)^T V(\theta)^{-1} H(\theta)$, where $H(\theta) = E\{-\sum_{i=1}^{m} \partial^2 \log f(y \in \mathcal{A}_i; \theta)/\partial\theta\partial\theta^T\}$ and $V(\theta) = \text{var}\{\sum_{i=1}^{m} \partial \log f(y \in \mathcal{A}_i; \theta)/\partial\theta\}$ are the sensitivity matrix and the variability matrix, respectively. Readers are referred to Cox and Reid (2004) and Varin (2008) for a more detailed discussion on the asymptotic behavior of the maximum composite likelihood estimator. Gaussian graphical models with added colored classes constraints on the parameters are not closed exponential families in the sense of Mardia et al. (2009). So the maximum composite likelihood estimates for the RCOR and RCON models suffer a slight loss of efficiency compared to the maximum likelihood estimates. We recall, however, that in terms of computational complexity, composite likelihood estimation does not require large matrix inversions and will make the computations quite fast.

3. COMPOSITE LIKELIHOOD ESTIMATION

3.1 THE RCON MODEL

Let $X = (X_1, \ldots, X_p)$ be a random vector with distribution following the RCON$(\mathcal{V}, \mathcal{E})$ model. Following (2), (3), and (4), we have that the conditional distribution of X_j given $X_{V\setminus\{j\}}$ can be written as

$$f(x_j; \Theta, x_{V\setminus\{j\}}) = \frac{\theta_{jj}^{1/2}}{\sqrt{2\pi}} \exp -\frac{1}{2}\theta_{jj} \left(x_j + \theta_{jj}^{-1} \left(\sum_{i=1, i\neq j}^{p} \theta_{ji} x_i \right) \right)^2. \tag{7}$$

Let $x^{(1)}, \ldots, x^{(n)}$ be a sample from this RCON model with $x^{(i)} = (x_{i1}, x_{i2}, \ldots, x_{ip})^T$ and let $\mathbf{X}$ be the $n \times p$ data matrix. The composite likelihood function obtained from the conditional distribution of X_j given $X_{V\setminus\{j\}}$, $j = 1, \ldots, p$ is

$$L_c(\Theta) = \prod_{i=1}^{n} \prod_{j=1}^{p} f(x_{ij}; \Theta, x^{(i)}_{V\setminus\{j\}}). \tag{8}$$

The composite log-likelihood can also be written in matrix form as

$$\ell_c(\Theta) = \frac{1}{2}\sum_{j=1}^{p}\left(n\log\theta_{jj} - \theta_{jj}||X_{(j)} - \mathbf{X}B_j||_2^2\right) \tag{9}$$

up to a constant, where $X_{(j)} = (x_{1j}, x_{2j}, \ldots, x_{nj})^T$ is the jth column of the $n \times p$ data matrix and B_j is a p-vector with elements β_{ij} except for a zero at the jth position (see (4)).

From (7) above, we see that each $f(x_{ij}; \Theta, x^{(i)}_{V\setminus\{j\}})$, $i = 1, \ldots, n$ depends only on the vector θ_j of free parameters in the jth row of Θ. Since $\ell_c(\Theta)$ is a function of just the vector θ of free parameters, we will simply write $\ell_c(\Theta) = \ell_c(\theta)$ and we, therefore, have that

$$\begin{aligned} E\left(\frac{-\partial^2\ell_c(\theta)}{\partial\theta^2}\right) &= \sum_{i=1}^{n}\sum_{j=1}^{p} E\left(E\left(\frac{-\partial^2 l\left(x_{ij}|x^{(i)}_{V\setminus\{j\}}\right)}{\partial\theta_j^2}|x^{(i)}_{V\setminus\{j\}}\right)\right) \\ &= \sum_{i=1}^{n}\sum_{j=1}^{p} E\left(\text{var}\left(\frac{\partial l\left(x_{ij}|x^{(i)}_{V\setminus\{j\}}\right)}{\partial\theta_j}|x^{(i)}_{V\setminus\{j\}}\right)\right) \end{aligned}$$

is a positive definite matrix. Thus $-\ell_c(\theta)$ is asymptotically convex around the true null value θ_0. We propose to estimate the sparse RCON model by solving the minimization problem:

$$\min_{\theta_{E_s}, 1\le s\le l, \theta_{V_m}, 1\le m\le k} Q(\theta) = -\ell_c(\theta) + n\sum_{s\in\mathcal{E}} p_\lambda(|\theta_{E_s}|),$$

where $p_\lambda(.)$ is a penalty function, λ is the penalty parameter and the penalty is on the off-diagonal parameters θ_{E_s}, $s = 1, \ldots, l$ (there is no sparsity for the reciprocal conditional variances θ_{ii}). The contribution of the composite likelihood is of the order of n, so the penalty term has to be of the order of n to compete. Such a penalization scheme will encourage the sparsity of edge classes but not the sparsity of the total number of edges.

There are several penalty functions available. The LASSO penalty (Tibshirani 1996), $p_\lambda(|\theta|) = \lambda|\theta|$, increases linearly with the size of its argument. It is convex and the numerical algorithm is stable. It is a classical tool and it is widely used in many applications. However, the LASSO estimates may suffer from bias for large parameters. Furthermore, the LASSO estimator may not be selection consistent unless a strong irrepresentable condition is satisfied. To avoid such problems, the smoothly clipped absolute deviation (SCAD, Fan and Li 2001) and the minimax concave penalty (MCP, Zhang 2010) have been proposed.

- The SCAD penalty function is symmetric about 0 and for any real $\theta > 0$ is equal to

$$p_\lambda(\theta) = \begin{cases} \lambda\theta, & \text{if } \theta \le \lambda; \\ \frac{1}{(a-1)}\left(a\lambda\theta - \frac{\theta^2}{2}\right) + C_1, & \text{if } \lambda < \theta \le a\lambda; \\ C_2, & \text{if } a\lambda < \theta\ , \end{cases} \tag{10}$$

  where a is some constant, usually set to 3.7, $C_1 = \frac{-1}{2(a-1)}\lambda^2$, and $C_2 = \frac{\lambda^2(a+1)}{2}$.

- The MCP penalty gradually relaxes the penalization rate until, when $\theta > \gamma\lambda$, the rate of penalization decreases to zero. The penalty function is symmetric about 0, and for

any real $\theta > 0$, it takes the form

$$p_{\lambda,\gamma}(\theta) = \begin{cases} \lambda\theta - \frac{\theta^2}{2\gamma}, & \text{if } \theta \le \lambda\gamma; \\ \frac{1}{2}\gamma\lambda^2, & \text{if } \theta > \lambda\gamma \end{cases} \tag{11}$$

for $\lambda \ge 0$, and $\gamma > 1$.

In the literature, it has been shown that both SCAD and MCP regression methods have the so-called oracle property, implying that the corresponding penalized estimator is consistent and as efficient as the maximum likelihood estimate obtained under the true subset model. To numerically minimize $Q(\theta)$, we can employ the coordinate descent algorithm, which proceeds by updating each parameter of the objective function one at a time (Tseng 2001; Friedman et al. 2007). The use of the LASSO, SCAD, and MCP penalty with coordinate descent algorithm and the convergence properties of the corresponding estimators have been extensively investigated in Breheny and Huang (2011) and Mazumder, Friedman, and Hastie (2011).

- We first consider the LASSO penalty. Differentiating $Q(\theta)$, we obtain the first derivative of the objective function with respect to the edge class parameter θ_{E_s} (see the online supplementary file for its derivation). This provides the update for θ_{E_s}:

$$\hat{\theta}_{E_s} = \frac{S\left(-\frac{1}{n}\sum_{j=1}^{p}\left(\sum_{k;(k,j)\in E_s} X_{(j)}^T X_{(k)} + \theta_{jj}^{-1}\sum_{k;(k,j)\in E_s}\sum_{l;(l,j)\in E_s^c} X_{(k)}^T X_{(l)}\theta_{lj}\right), \lambda\right)}{\frac{1}{n}\left(\sum_{j=1}^{p}\sum_{k;(k,j)\in E_s}\sum_{l;(l,j)\in E_s}\theta_{jj}^{-1} X_{(k)}^T X_{(l)}\right)},$$

  where $S(z,\lambda) = \text{sign}\,(z)(|z| - \lambda)_+$ is the soft-thresholding operator, $C = \frac{1}{n}X^T X$ denotes the sample covariance matrix, and $E_s^c = \{(k,j)|k \ne j \text{ and } (k,j) \notin E_s\}$. Given the color edge group E_s, define the edge adjacency matrix T^{E_s}, with $T_{kj}^{E_s} = 1$, if $(k,j) \in E_s$, and $T_{kj}^{E_s} = 0$ otherwise. The update for θ_{E_s} can be simplified as follows:

$$\hat{\theta}_{E_s} = \frac{S\left(-\left(\text{tr}(T^{E_s}C) + \text{tr}(T^{E_s}(T^{E_s^c} \odot B)C)\right), \lambda\right)}{\text{tr}(T^{E_s}(T^{E_s}\Sigma)C)}, \tag{12}$$

  where $\odot$ denotes the component-wise product, B is the $p \times p$ matrix with columns B_j, $j = 1, \ldots, p$, and Σ denotes the $p \times p$ diagonal matrix with entries θ_{jj}^{-1}.

- If the SCAD penalty is used, we use the method of one-step local linear approximation (Zou and Li 2008). The size of the penalty is equal to the first derivative of the penalty function evaluated at an initial consistent estimate $\theta_{E_s}^*$. We, therefore, modify the LASSO updating Equation (12) into the following SCAD updating equation:

$$\hat{\theta}_{E_s} = \frac{S\left(-\text{tr}(T^{E_s}C) + \text{tr}(T^{E_s}(T^{E_s^c} \odot B)C),\, p'_\lambda(|\theta_{E_s}^*|)\right)}{\text{tr}(T^{E_s}(T^{E_s}\Sigma)C)}, \tag{13}$$

  where p'_λ denotes the first derivative of the SCAD penalty function.

- If the MCP penalty is used, we can apply the univariate thresholding rule proposed in Breheny and Huang (2011) and Mazumder, Friedman, and Hastie (2011) and obtain

the updating equation:

$$\hat{\theta}_{E_s} = \begin{cases} S(u,\lambda)/\left(v - \frac{1}{\gamma}\right), & \text{if } |u| \le v\gamma\lambda, \\ u/v, \text{ if } |u| > v\gamma\lambda, \end{cases} \tag{14}$$

with $u = -\text{tr}(T^{E_s}C) + \text{tr}(T^{E_s}(T^{E_s^c} \odot B)C)$, and $v = \text{tr}(T^{E_s}(T^{E_s}\Sigma)C)$.

Now, let us differentiate $Q(\theta)$ with respect to θ_{V_m}, the common values of θ_{jj} for the vertex class V_m. Using (9) again and recalling that $-XB_j = \sum_{k\neq j}\theta_{kj}\theta_{jj}^{-1}X_{(k)}$, the likelihood equation for θ_{V_m} is

$$\frac{\partial Q(\theta)}{\partial \theta_{V_m}} = \frac{n}{2}\left[\left(\sum_{j\in V_m} C_{jj}\right) - |V_m|\theta_{V_m}^{-1} - \left(\sum_{j\in V_m} q_j\right)\theta_{V_m}^{-2}\right] = 0, \tag{15}$$

where $q_j = \sum_{k\neq j}\sum_{l\neq j}\theta_{kj}\theta_{lj}C_{kl} = \theta_{jj}^2 B_j^t X^t X B_j/n$ and $|V_m|$ is the cardinality of V_m. Therefore, the solution of this likelihood equation is

$$\widehat{\theta_{V_m}^{-1}} = \frac{-|V_m| + \sqrt{|V_m|^2 + 4(\sum_{j\in V_m} q_j)(\sum_{j\in V_m} C_{jj})}}{2\sum_{j\in V_m} q_j}.$$

Since, clearly from its expression above, q_j is positive, the quadratic Equation (15) has one unique positive solution as given above. Alternating the updating scheme throughout all the θ_{E_s}, and θ_{V_m} until convergence, we obtain the penalized sparse estimate of the concentration matrix under the RCON model.

3.2 THE RCOR MODEL

Consider an RCOR($\mathcal{V}, \mathcal{E}$) model with vertex coloring $\mathcal{V}$ and edge coloring $\mathcal{E}$. Recall that $P = (\rho_{ij})_{1\le i,\ j\le p}$ denotes the partial correlation matrix. Given an edge color class, for all edges $(i, j) \in E_s$, ρ_{ij} are all equal and denoted as ρ_{E_s}. Let $\rho = (\rho_{E_s},\ E_s \in \mathcal{E})$. Given a vertex color class, for all vertices $i \in V_m$, θ_{ii} are all equal and denoted as θ_{V_m}. Let Θ_D denote the diagonal matrix with entries θ_{jj}, $j = 1, \ldots, p$. We propose to estimate the sparse RCOR model by solving the minimization problem:

$$\min_{\rho_{E_s}, 1\le s\le l, \theta_{V_m}, 1\le m\le k} Q(\Theta_D, \rho) = -\ell_c(\rho, \Theta_D) + n\sum_s p_\lambda(|\rho_{E_s}|). \tag{16}$$

First we consider the LASSO penalty. Differentiating (16) with respect to ρ_{E_s}, we obtain the thresholded estimate of the partial correlation which takes the following form:

$$\hat{\rho}_{E_s} = \frac{S\left(\text{tr}\left(T^{E_s}\left(\Theta_D^{\frac{1}{2}}C\Theta_D^{\frac{1}{2}}\right)\right) - \text{tr}\left(T^{E_s}\left(T^{E_s^c}\odot\tilde{P}\right)\left(\Theta_D^{\frac{1}{2}}C\Theta_D^{\frac{1}{2}}\right)\right), \lambda\right)}{\text{tr}\left(T^{E_s}T^{E_s}\left(\Theta_D^{\frac{1}{2}}C\Theta_D^{\frac{1}{2}}\right)\right)}, \tag{17}$$

where $\tilde{P}$ denotes the matrix P with the 1 on the diagonal being replaced by 0. For the SCAD and MCP penalties, updating equations similar to (13) and (14) can be obtained in a parallel manner.

Differentiating (16) with respect to θ_{V_m}, we obtain

$$\frac{\partial Q(\Theta_D, \rho)}{\partial \theta_{V_m}} = \frac{n}{2}\{-|V_m|y^2 + by + a\},$$

where $y = \theta_{V_m}^{-1/2}$, $b = -2\text{tr}(T^{V_m} C \Sigma^{-1/2} T^{V_m^c} \tilde{P}) + \text{tr}(\tilde{P} T^{v_m^c} \Sigma^{-1/2} C T^{V_m} \tilde{P})$, and $a = \text{tr}(T^{V_m} C) - 2\text{tr}(T^{V_m} C T^{V_m} \tilde{P})) + \text{tr}(\tilde{P} T^{V_m} C T^{V_m} \tilde{P})$. We solve the likelihood equation $|V_m|y^2 - by - a = 0$. The solution

$$y = \frac{b + \sqrt{b^2 + 4a|V_m|}}{2|V_m|}$$

is the unique positive solution because $a = \text{tr}(C(T^{V_m} - \tilde{P} T^{V_m})^T (T^{V_m} - \tilde{P} T^{V_m})) > 0$.

3.3 ASYMPTOTIC PROPERTIES

Although penalized estimators based on SCAD or MCP penalty have been shown to possess the ORACLE property in the regression setting, this property has not been established for composite likelihood on colored graphical Gaussian models. We do so in Theorems 1 and 2. These two theorems are stated for the SCAD penalty and the asymptotic behavior of $\hat{\theta}_{E_s}$ in (13) for the RCON model. Similar statements can be proved for the MCP penalty and the asymptotic behavior of the estimates of the RCOR model. For notational convenience, let $z = \{E_s : \theta_{E_s} \neq 0\} \cup \mathcal{V}$ denote all the nonzero parameters representing nonzero edge classes and all vertex classes and $z^c = \{E_s : \theta_{E_s} = 0\}$ denote all the zero edge classes. We assume that both $H(\theta)$ and $V(\theta)$ are positive definite, where $H(\theta) = E(-\partial^2 \ell_c(\theta)/\partial\theta\partial\theta^T)$, and $V(\theta) = \text{var}(\partial \ell_c(\theta)/\partial\theta)$.

Theorem 1. Given the SCAD penalty function $p_\lambda(\theta)$, for a sequence of λ_n such that $\lambda_n \to 0$, and $\sqrt{n}\lambda_n \to \infty$ as $n \to \infty$, there exists a local maximizer $\hat{\theta}$ of $Q(\theta)$ with $||\hat{\theta} - \theta_0||_2 = O_p(n^{-\frac{1}{2}})$. Furthermore, we have

$$\lim_{n\to\infty} P(\hat{\theta}_{z^c} = 0) = 1.$$

The proofs of Theorems 1 and 2 are given in the supplementary file. Next, we establish the asymptotic distribution of the estimator $\hat{\theta}_z$, where θ_z denoting the subvector of nonzero parameters in θ. Let Θ_0 be the true value of the parameter. Define the matrix $\Sigma_1 = \text{diag}\{p''_{\lambda_n}(|\theta_{j0}|); j \in z\}$, and the vector $b_1 = (p'_{\lambda_n}(|\theta_{j_0}|)\text{sign}(\theta_{j0}); j \in z)$. Let H_{zz} and V_{zz} denote the submatrices of $H(\theta_0)$ and $V(\theta_0)$, respectively, corresponding to z.

Theorem 2. Given the SCAD penalty function $p_\lambda(\theta)$, for a sequence of λ_n such that $\lambda_n \to 0$ and $\sqrt{n}\lambda_n \to \infty$, as $n \to \infty$, the subvector of the root-n consistent estimator $\hat{\theta}_z$ has the following asymptotic distribution:

$$\sqrt{n} H_{zz}(\hat{\theta}_z - \theta_{z0}) \to N\{0, V_{zz}\}, \text{ as } n \to \infty.$$

Under the assumption of Theorem 2, the subvector of the root-n consistent estimator $\hat{\theta}_z$ has the following asymptotic covariance $H_{zz}^{-1} V_{zz} H_{zz}^{-1}$. The formulas for $\partial^2 \ell_c/\partial\theta\partial\theta^T$ can

be found in the proof of Lemma 1 of the online supplementary file. The corresponding estimate for the Hessian matrix is $\hat{H}_{zz} = \partial^2 \ell_c / \partial\theta_z \partial\theta_z^T|_{\hat{\theta}}$. To estimate V_{zz}, we use the sample covariance matrix of the composite score vector $\hat{V}_{zz} = \frac{1}{n}\sum_{i=1}^{n} S_{iz} S_{iz}^T|_{\hat{\theta}}$, where $S_i = \partial \ell_c(Y_i)/\partial\theta$ denotes the score vector obtained for the ith observation and S_{iz} denote the subvector of S_i corresponding to the subvector of θ_z. Combining the estimated H_{zz} and V_{zz}, we can compute the standard error estimates of the penalized composite likelihood estimates. In practice, the bootstrap method can also be used to provide the standard error estimation.

4. REMARKS ON THE DETERMINATION OF COLOR CLASSES

So far, we have assumed the color classes are known. If there are not known, there exist strategies to determine them. We list some of them here. First the fused LASSO penalty (Tibshirani 2005) can be used to collapse the estimates of θ_{ij} which are close to each other. For illustration purposes, let us focus on the determination of the color classes for edges. The approach can be readily extended to obtain color classes for vertices. We first obtain an initial LASSO estimate of all the edges and rank them from the smallest to the largest. We then consider the following objective function with penalties on the distances between parameters which are adjacent in the order previously obtained:

$$\min -\ell_c(\theta) + n\sum_{i<j} p_{\lambda_1}(|\theta_{ij}|) + n\sum_{i<j} p_{\lambda_2}(|\theta_{ij} - a(\theta_{ij})|) + n\sum_{i<j} p_{\lambda_2}(|\theta_{ij} - b(\theta_{ij})|), \tag{18}$$

where p_{λ_1} and p_{λ_2} are two penalty functions (L_1 or SCAD), and $a(\theta_{ij})$ and $b(\theta_{ij})$ are the two edge parameters which are ordered left and right next to θ_{ij}. The resulting estimates from the graphical fused LASSO are clumped together. As λ_2 increases, more and more edges are fused together and therefore form color classes. By tuning the size of λ_2, we can vary the number of color classes. To solve this minimization problem, we can adopt the fused LASSO signal approximator (FLSA) algorithm originally proposed for regression problems (Friedman et al. 2007) to the graphical model setting. The algorithm consists of iteratively applying the three following steps with λ_1 being fixed and λ_2 being incremented starting from zero with small δ values added at each cycle:

- Coordinate descent step: compute $\frac{\partial Q(\theta)}{\theta_{ij}}$, and use the subgradient at points where the objective function is not differentiable. By the Karush–Kuhn–Tucker (KKT) conditions, we solve for θ_{ij} while other parameters are fixed at the current update.
- Fusion step: examine adjacent θ_{ij}'s, $a(\theta_{ij})$ and $b(\theta_{ij})$ and force them to be fused into one parameter and see if the objective function can be decreased by doing so.
- Smooth step: add a small increment δ to λ_2 which thus becomes $\lambda_2 + \delta$ and repeat the coordinate descent and fusion steps.

As an alternative strategy for the determination of color classes, we can perform spectral clustering (Ng, Jordan, and Weiss 2001; Qin and Rohe 2013) on the nodes. The algorithm consists of the following steps:

- Obtain the standard LASSO estimate A of Θ, and form the diagonal matrix D with $D_{ii} = \sum_{l=1}^{n} A_{il} + \tau$, where τ is a positive number chosen to make sure that the

appropriate matrices are positive definite. Construct the normalized symmetric graph Laplacian which is $L = D^{-1/2}AD^{-1/2}$.

- Inspect the eigenvalues of L, take the k largest eigenvalues and form the $N \times k$ matrix X with columns the corresponding eigenvectors of L. We note that one may use the Laplacian defined as $I - L$ (Shi and Malik 2000) rather than L as we did here. The two versions of the Laplacian matrix have the same set of eigenvectors, but the eigenvalues change from λ_i to $1 - \lambda_i$. Therefore, if $I - L$ is used, one should choose the k smallest eigenvalues rather than the k largest. Form the matrix $\tilde{X}$ with entries $\tilde{X}_{ij} = X_{ij}/(\sum_{j'} X_{ij'}^2)^{\frac{1}{2}}$.
- Run k-means on the rows of $\tilde{X}$. The cluster of rows is equivalent to the cluster of nodes.

The two methods discussed above can both provide the color classes for our proposed colored graph algorithm.

5. SOME NUMERICAL EXAMPLES

5.1 SIMULATED EXAMPLES WITHOUT PENALIZATION

We examine the performance of the unpenalized composite likelihood estimator on large matrices. First we consider the RCON model. We simulate data under different scenarios with n varying from 250 to 1000 and p varying from 40, 60, to 100. We include 30 different edge classes and 20 different vertex classes. We simulate multivariate Gaussian random vectors with entries of the sparse precision matrix given by $\theta_{\mathcal{E}} = (0_{25}, 0.2591, 0.1628, -0.1934, 0.0980, 0.0518)$, and $\theta_{\mathcal{V}} = (1.3180, 1.8676, 1.788004, 1.7626, 1.6550, 1.1538, 1.3975, 1.7877, 1.7090, 1.6931, 1.46313, 1.5131, 1.7084, 1.7344, 1.1441, 1.8059, 1.7446, 1.8522, 1.3146, 1.1001)$, where 0_p denotes a zero vector of length p. The values for nonzero θ_{E_s} are uniformly sampled from -0.3 to 0.3. The values for θ_{V_m} are uniformly sampled from 1 to 2. Then these $\theta_{\mathcal{E}}$ and $\theta_{\mathcal{V}}$ are used by all the 100 simulated datasets. We consider two different scenarios for the assignment of edges to the edge classes. In the balanced network scenario, each edge is randomly assigned to the $s = 30$ edge classes with equal probability. So the number of edges for each edge class varies from dataset to dataset but is on average equal to $p(p-1)/2s$ edges. In a similar manner, for each simulated dataset, each node is randomly assigned to the $m = 20$ vertex (or node) classes with equal probability. So the number of nodes in each node class varies from simulation to simulation but each node class has an average number of p/m nodes. For comparison purposes, we also simulate an unbalanced network and investigate the performance of our method: the first three edge classes have only on average $p(p-1)/(4s)$ edges, the fourth class has an average of $13p(p-1)/(4s)$ number of edges while all the other classes have an average of $p(p-1)/s$ of edges. The results are given in Table 1 where we compare both the absolute errors and relative errors of the composite likelihood estimates with those of the naive estimates from 100 simulated datasets. The naive estimator estimates the edge class parameters and vertex class parameters by simply averaging all the values belonging to the same class in the inverse sample covariance matrix.

Table 1. Comparison of composite likelihood and moment estimates for simulated datasets and the RCON model (balanced and unbalanced network) with no penalization

		Comp		Naive		Comp		Naive	
n	p	ae $\hat{\theta}_{E_s}$	mre $\hat{\theta}_{E_s}$	ae $\tilde{\theta}_{E_s}$	mre $\tilde{\theta}_{E_s}$	ae $\hat{\theta}_{V_m}$	mre $\hat{\theta}_{V_m}$	ae $\tilde{\theta}_{V_m}$	mre $\tilde{\theta}_{V_m}$
250	40	0.0986	0.1312	0.1589	0.2622	0.4263	0.0481	1.5228	0.1987
		(0.0133)	(0.0444)	(0.0239)	(0.0722)	(0.0837)	(0.0089)	(0.1464)	(0.0199)
250	60	0.0626	0.0830	0.1619	0.3283	0.3224	0.0365	2.3640	0.3224
		(0.0092)	(0.0302)	(0.0211)	(0.0792)	(0.0552)	(0.0064)	(0.1526)	(0.0205)
250	100	0.0332	0.0470	0.2727	0.6672	0.2207	0.0246	4.8573	0.6766
		(0.0045)	(0.0175)	(0.0207)	(0.0965)	(0.0340)	(0.0038)	(0.2072)	(0.0273)
500	40	0.0714	0.0890	0.0940	0.1419	0.2883	0.0322	0.7289	0.0910
		(0.0095)	(0.0364)	(0.0130)	(0.0521)	(0.0450)	(0.0051)	(0.0968)	(0.0124)
500	60	0.0455	0.0549	0.0816	0.1567	0.2345	0.0261	1.0399	0.1394
		(0.0068)	(0.0237)	(0.0110)	(0.0438)	(0.0362)	(0.0042)	(0.0861)	(0.0124)
500	100	0.0233	0.0316	0.1049	0.2576	0.1532	0.0172	1.8300	0.2536
		(0.0039)	(0.0130)	(0.0089)	(0.0384)	(0.0310)	(0.0035)	(0.0742)	(0.0101)
1000	40	0.0498	0.0654	0.0603	0.0815	0.2071	0.0235	0.3855	0.0458
		(0.0067)	(0.0235)	(0.0077)	(0.0313)	(0.0379)	(0.0043)	(0.0582)	(0.0071)
1000	60	0.0319	0.0425	0.0475	0.0814	0.1626	0.0182	0.5059	0.0653
		(0.0042)	(0.0160)	(0.0063)	(0.0273)	(0.0278)	(0.0032)	(0.0499)	(0.0071)
1000	100	0.0166	0.0225	0.0492	0.1124	0.1096	0.0124	0.8164	0.1121
		(0.0022)	(0.0085)	(0.0053)	(0.0226)	(0.0163)	(0.0019)	(0.0421)	(0.0059)
250	40	0.1157	0.1336	0.1788	0.2545	0.4239	0.0474	1.5219	0.1973
		(0.0207)	(0.0549)	(0.0296)	(0.0865)	(0.0663)	(0.0075)	(0.1471)	(0.0200)
250	60	0.0712	0.0807	0.1713	0.3307	0.3301	0.0372	2.3616	0.3217
		(0.0118)	(0.0339)	(0.0242)	(0.0901)	(0.0583)	(0.0064)	(0.1532)	(0.0204)
250	100	0.0371	0.0434	0.2778	0.6771	0.2191	0.0248	4.8582	0.6760
		(0.0059)	(0.0156)	(0.0216)	(0.0885)	(0.0351)	(0.0041)	(0.2050)	(0.0284)
500	40	0.0773	0.0908	0.1005	0.1319	0.2983	0.0335	0.7282	0.0901
		(0.0120)	(0.0357)	(0.0152)	(0.0430)	(0.0476)	(0.0053)	(0.0973)	(0.0121)
500	60	0.0496	0.0543	0.0879	0.1574	0.2289	0.0260	1.0327	0.1389
		(0.0074)	(0.0214)	(0.0112)	(0.0414)	(0.0367)	(0.0042)	(0.0868)	(0.0121)
500	100	0.0261	0.0299	0.1080	0.2536	0.1526	0.0170	1.8286	0.2536
		(0.0043)	(0.0114)	(0.0089)	(0.0374)	(0.0288)	(0.0032)	(0.0738)	(0.0102)
1000	40	0.0575	0.0617	0.0689	0.0792	0.2071	0.0231	0.3841	0.0459
		(0.0107)	(0.0248)	(0.0121)	(0.0319)	(0.0361)	(0.0042)	(0.0520)	(0.0068)
1000	60	0.0365	0.0436	0.0519	0.0844	0.1632	0.0182	0.5053	0.0648
		(0.0063)	(0.0187)	(0.0085)	(0.0289)	(0.0243)	(0.0031)	(0.0484)	(0.0068)
1000	100	0.0184	0.0208	0.0517	0.1139	0.1101	0.0122	0.8162	0.1122
		(0.0029)	(0.0086)	(0.0049)	(0.0186)	(0.0191)	(0.0021)	(0.0431)	(0.0058)

NOTE: The "ae" stands for absolute error and "mre" stands for mean relative error. For example, the "ae" of $\hat{\theta}_{E_s}$ is defined as $||\hat{\theta}_{E_s} - \theta_{E_s,0}||_2$; and the "mre" of $\hat{\theta}_{E_s}$ is defined as mean of $(|(\hat{\theta}_{E_s} - \theta_{E_s,0})/\theta_{E_s,0}|)$. The "mre" are only calculated for the nonzero subset of parameters. The first half of table above the line is for balanced network and the second half of the table below the line is for unbalanced network.

The proposed composite likelihood estimates consistently enjoy much smaller errors than the naive method across all settings. As shown also in Table 1, the absolute sum of squared errors of the edge class parameters are slightly larger for the unbalanced network than for the balanced one. With regard to the mean of the relative errors for the nonzero edge class parameters, the two networks have a comparable performance.

Next, we investigate the empirical performance of the proposed composite likelihood estimator under the RCOR model. We simulate under different scenarios

with n varying from 250 to 1000 and p varying from 40, 60 to 100. We include 30 different edge classes and 20 different vertex classes. We simulate multivariate Gaussian random vectors with entries of the sparse partial correlation matrix given by $\rho_{\mathcal{E}} = (0_{26}, 0.1628, -0.1534, 0.0980, 0.0518)$ and with $\theta_{\mathcal{V}} = (3.0740, 3.6966, 3.7772, 3.5475, 3.2841, 3.4699, 3.7235, 3.5987, 3.3313, 3.8183, 3.9236, 3.9008, 3.9011, 3.0470, 3.0139, 3.2072, 3.8438, 3.4823, 3.9373, 3.0125)$. The values for nonzero θ_{E_s} are uniformly sampled from -0.3 to 0.3. The values for θ_{V_m} are uniformly sampled between 3 and 4. We choose the range of 3 to 4 for θ_{V_m}, because we find that for large p, such as $p = 100$, in order for the matrix to be positive definite, the θ_{V_m} have to be large. This set of θ is used by all the 100 simulated datasets. The assignment of edges to each edge color classes is done randomly with equal probability. We conduct a similar random assignment of vertices to each vertex color class. The simulation results are given in Table 2. We provide both the absolute errors and relative errors for the composite likelihood estimates and the naive estimates from the 100 simulated datasets. For both the estimated partial correlations and the conditional variances, the composite likelihood estimates yield consistently smaller errors compared to the naive estimates. This superior performance is consistent across all the different sample sizes and different dimensions of the matrices.

5.2 SIMULATED EXAMPLES WITH PENALIZATION

In our next calculations, we introduce penalization. We examine the empirical performance of the penalized composite likelihood estimator. We simulate the RCON model using the same settings as in Table 1. We consider different scenarios with $n = 250$ or $n = 500$, and $p = 40$, $p = 60$, and $p = 100$. We use the penalized composite likelihood estimator to estimate the sparse matrix. The tuning parameter is selected by the Bayesian information criterion (BIC), with $\text{BIC} = -2\ell(\hat{\theta}) + df \log n$, or composite likelihood Bayesian information criterion (COMP-BIC), with $\text{COMP-BIC} = -2\ell_c(\hat{\theta}) + df \log n$, where df denotes the total number of nonzero edges and $\hat{\theta}$ denotes the penalized composite likelihood estimate (Gao and Song 2010). Although traditionally, the degree of freedom should be the number of parameters, we use the number of nonzero edges here instead. The reason is that for large network with large p, the likelihood term grows with both n and p, for the penalty term to compete with the likelihood term, we use the term of $\log n$ multiplied by the number of nonzero edges, which grows with p as well. As shown in the simulation result below, the model selection performance with the number of nonzero edges used as the degree of freedom is very good. It yields small false positive rates and small false negative rates.

For each setting, 100 simulated datasets are generated and for each dataset we calculate the number of false negatives and false positives. The results are given in Table 3: we see that the proposed method has satisfactory model selection properties with very low false negative and false positive edges. For example, with $n = 500$ and $p = 60$, each simulated dataset has an average number of 1475 zero edges and 295 nonzero edges. The proposed method identifies an average of zero false negatives and 0.58 false positives. The size of the tuning parameters is also listed in Table 3. To compare the efficiency of our method using colored graphical Gaussian models with that of more classical methods, we also did a model search using the unconstrained LASSO and the unconstrained SCAD. We

Table 2. Comparison of composite likelihood and moment estimates for simulated datasets and RCOR model with no penalization

		Comp		Naive		Comp		Naive		Comp		Naive	
n	p	ae $\hat{\rho}_{E_s}$	mre $\hat{\rho}_{E_s}$	ae $\tilde{\rho}_{E_s}$	mre $\tilde{\rho}_{E_s}$	ae $1/\hat{\theta}_{V_m}$	mre $1/\hat{\theta}_{V_m}$	ae $1/\tilde{\theta}_{v_m}$	mre $1/\tilde{\theta}_{v_m}$	ae $\hat{\theta}_{V_m}$	mre $\hat{\theta}_{V_m}$	ae $\tilde{\theta}_{v_m}$	mre $\tilde{\theta}_{v_m}$
250	40	0.0618	0.0926	0.0730	0.1091	0.0807	0.0510	0.2194	0.1613	1.0144	0.0519	3.3943	0.1981
		(0.0084)	(0.0399)	(0.0101)	(0.0464)	(0.0133)	(0.0078)	(0.0170)	(0.0139)	(0.1697)	(0.0079)	(0.3119)	(0.0198)
250	60	0.0396	0.0622	0.0516	0.0776	0.0647	0.0406	0.3150	0.2411	0.8073	0.0410	5.2494	0.3225
		(0.0058)	(0.0229)	(0.0069)	(0.0310)	(0.0103)	(0.0068)	(0.0147)	(0.0117)	(0.1318)	(0.0071)	(0.3262)	(0.0203)
250	100	0.0193	0.0269	0.0336	0.0507	0.0473	0.0302	0.5172	0.4017	0.5896	0.0303	10.7957	0.6759
		(0.0036)	(0.0121)	(0.0045)	(0.0201)	(0.0081)	(0.0056)	(0.0129)	(0.0100)	(0.1041)	(0.0057)	(0.4515)	(0.0282)
500	40	0.0448	0.0712	0.0499	0.0821	0.0558	0.0354	0.1159	0.0810	0.6984	0.0357	1.6238	0.0904
		(0.0057)	(0.0296)	(0.0063)	(0.0366)	(0.0090)	(0.0063)	(0.0134)	(0.0106)	(0.1128)	(0.0063)	(0.2024)	(0.0127)
500	60	0.0274	0.0395	0.0326	0.0492	0.0459	0.0291	0.1611	0.1208	0.5740	0.0292	2.3089	0.1392
		(0.0038)	(0.0173)	(0.0046)	(0.0198)	(0.0080)	(0.0055)	(0.0113)	(0.0090)	(0.1085)	(0.0057)	(0.1929)	(0.0117)
500	100	0.0134	0.0208	0.0206	0.0333	0.0332	0.0207	0.2604	0.2012	0.4133	0.0208	4.0585	0.2532
		(0.0024)	(0.0099)	(0.0029)	(0.0139)	(0.0055)	(0.0035)	(0.0085)	(0.0067)	(0.0677)	(0.0035)	(0.1708)	(0.0105)
1000	40	0.0317	0.0487	0.0347	0.0539	0.0401	0.0253	0.0650	0.0432	0.5004	0.0254	0.8596	0.0459
		(0.0043)	(0.0185)	(0.0049)	(0.0212)	(0.0069)	(0.0043)	(0.0080)	(0.0059)	(0.0795)	(0.0043)	(0.1126)	(0.0066)
1000	60	0.0196	0.0277	0.0229	0.0337	0.0323	0.0205	0.0836	0.0605	0.4001	0.0206	1.1183	0.0651
		(0.0023)	(0.0131)	(0.0028)	(0.0145)	(0.0050)	(0.0033)	(0.0074)	(0.0060)	(0.0650)	(0.0033)	(0.1070)	(0.0068)
1000	100	0.0097	0.0129	0.0139	0.0208	0.0238	0.0150	0.1311	0.1003	0.2923	0.0150	1.8116	0.1120
		(0.0015)	(0.0060)	(0.0019)	(0.0092)	(0.0038)	(0.0025)	(0.0066)	(0.0049)	(0.0446)	(0.0025)	(0.0952)	(0.0061)

NOTE: Numbers without parentheses are errors averaged from the 100 simulated datasets; the numbers within parentheses are standard deviations obtained from the 100 simulated datasets. The "ae" stands for absolute error and "mre" stands for mean relative error. For example, the "ae" of $\hat{\rho}_{E_s}$ is defined as $||\hat{\rho}_{E_s} - \rho_{E_s,0}||_2$; and the "mre" of of $\hat{\rho}_{E_s}$ is defined as mean of $|(\hat{\rho}_{E_s} - \rho_{E_s,0})/\rho_{E_s,0}|$. The "mre" are only calculated for the nonzero subset of parameters.

Table 3. Comparison of model selection performance from different methods (symmetry-constrained model with SCAD penalty, unconstrained model with SCAD penalty and unconstrained model with LASSO penalty) and different information criterion (BIC and composite BIC) under RCON model

				Sym-scad BIC			Sym-scad COMP-BIC			Lasso BIC			fp Scad BIC		
n	p	#ZE	#NZE	fn	fp	λ	fn	fp	λ	fn	fp	λ	fn	fp	λ
250	40	652	128	27.5400	0.0000	1.2805	27.0600	0.0000	1.2615	102.7800	12.6000	0.3145	96.2200	20.7200	0.2865
		(8)	(8)	(10.5872)	(0.0000)	(0.3166)	(11.2518)	(0.0000)	(0.3090)	(8.7566)	(5.3880)	(0.0240)	(10.2726)	(8.4148)	(0.0243)
250	60	1475	295	2.3000	5.4400	1.4990	2.3000	5.4400	1.4990	256.2400	16.6300	0.3985	245.9300	25.0200	0.3720
		(12)	(12)	(11.4111)	(19.2518)	(0.2511)	(11.4111)	(19.2518)	(0.2511)	(14.6122)	(5.7956)	(0.0204)	(15.6135)	(8.9171)	(0.0246)
250	100	4127	823	0.0000	6.9700	2.7960	0.0000	6.9700	2.7960	759.0400	31.3600	0.6085	759.0400	31.3600	0.6085
		(17)	(17)	(0.0000)	(34.3283)	(0.5516)	(0.0000)	(34.3283)	(0.5516)	(20.9954)	(10.2509)	(0.0263)	(20.9954)	(10.2509)	(0.0263)
500	40	652	128	26.6200	0.0000	1.2610	19.2200	0.0000	1.0095	81.2800	8.5500	0.2035	79.2000	10.7600	0.1980
		(8)	(8)	(10.9856)	(0.0000)	(0.3739)	(13.9831)	(0.0000)	(0.3295)	(7.7799)	(3.4623)	(0.0086)	(7.6594)	(6.7106)	(0.0136)
500	60	1475	295	0.0000	0.5800	1.0885	0.0000	0.5800	1.0885	221.8500	6.4700	0.2570	212.8600	9.5900	0.2455
		(12)	(12)	(0.0000)	(5.8000)	(0.1963)	(0.0000)	(5.8000)	(0.1963)	(15.3954)	(3.0665)	(0.0151)	(17.8331)	(7.4156)	(0.0169)
500	100	4127	823	0.0000	0.0000	2.0600	0.0000	0.0000	2.0600	733.0100	3.0200	0.3590	714.1300	4.8600	0.3440
		(17)	(17)	(0.0000)	(0.0000)	(0.4020)	(0.0000)	(0.0000)	(0.4020)	(27.4857)	(2.0449)	(0.0174)	(37.1564)	(4.2068)	(0.0197)

NOTE: Numbers without parentheses are averages of each quantity and numbers within parentheses are standard deviations over 100 simulated datasets. #ZE denotes the number of zero edges and #NZE denotes the number of nonzero edges.

Table 4. Comparison of model selection performance from different methods (symmetry-constrained model with SCAD penalty, unconstrained model with SCAD penalty, and unconstrained model with LASSO penalty) and different information criterion (BIC and composite BIC) under RCOR model

				fp Sym-scad BIC			fp Sym-scad COMP-BIC			fp Lasso BIC			fp Scad BIC		
n	p	#ZE	#NZE	fn	fp	λ	fn	fp	λ	fn	fp	λ	fn	fp	λ
250	40	677	103	20.1200	3.7700	1.5675	20.1200	3.7700	1.5675	94.4600	23.9600	0.2810	91.2600	36.3400	0.2550
		(7)	(7)	(18.6462)	(11.4696)	(0.2763)	(18.6462)	(11.4696)	(0.2763)	(7.9346)	(9.3980)	(0.0268)	(7.2455)	(8.6856)	(0.0119)
250	60	1533	237	0.5900	82.0700	1.7290	0.5900	82.0700	1.7290	223.3900	36.7700	0.3500	217.2600	58.8700	0.3185
		(11)	(11)	(5.9000)	(71.2611)	(0.1543)	(5.9000)	(71.2611)	(0.1543)	(11.6921)	(9.0026)	(0.0141)	(11.8352)	(15.6145)	(0.0236)
250	100	4291	659	0.0000	76.3700	3.4620	0.0000	76.3700	3.4620	632.0700	82.5500	0.5245	631.7500	83.3300	0.5235
		(17)	(17)	(0.0000)	(114.4081)	(0.3811)	(0.0000)	(114.4081)	(0.3811)	(17.6585)	(21.2961)	(0.0279)	(17.5599)	(20.7019)	(0.0278)
500	40	677	103	14.7500	0.0000	1.3845	14.6200	0.0000	1.3760	85.9600	29.0400	0.1675	80.9600	40.4900	0.1520
		(7)	(7)	(16.4724)	(0.0000)	(0.3848)	(16.5375)	(0.0000)	(0.3837)	(8.6922)	(13.9341)	(0.0232)	(7.3456)	(7.5619)	(0.0000)
500	60	1533	237	0.0000	10.3800	1.5340	0.0000	10.3800	1.5340	212.3200	32.7200	0.2020	212.1000	33.3100	0.2015
		(11)	(11)	(0.0000)	(23.0957)	(0.2534)	(0.0000)	(23.0957)	(0.2534)	(10.7533)	(7.0354)	(0.0000)	(11.3427)	(9.3751)	(0.0050)
500	100	4291	659	0.0000	1.5200	2.7140	0.0000	1.5200	2.7140	612.6500	56.7100	0.2520	612.0500	57.6100	0.2515
		(17)	(17)	(0.0000)	(15.2000)	(0.5403)	(0.0000)	(15.2000)	(0.5403)	(17.7077)	(9.3758)	(0.0000)	(18.1922)	(12.5874)	(0.0050)

NOTE: Numbers without parentheses are averages of each quantity and numbers within parentheses are standard deviations over 100 simulated datasets. #ZE denotes the number of zero edges and #NZE denotes the number of nonzero edges.

used the GLASSO package (Friedman et al. 2008) to perform the unconstrained LASSO. For the unconstrained SCAD, we follow the standard procedure, that is, we first find the LASSO estimate. Then at this estimate, we linearize the SCAD penalty function, evaluate it at the LASSO estimate and use it as the penalty term in the GLASSO package. With $n = 500$ and $p = 60$, the LASSO has an average of 221.85 false negatives and 6.47 false positives and SCAD has an average of 212.86 false negatives and 9.59 false positives. Another interesting phenomenon is that with the same sample size, the symmetry-constrained approach has better performance as p increases, and in contrast, the LASSO and SCAD have decreased performance as p increases. This is because increasing p increases the number of parameters in the LASSO and SCAD, but does not affect the number of parameters in the symmetry-constrained approach. On the contrary, with the same sample size, and the same number of edge and vertex classes, increasing p actually provides more information about the edge class and vertex class parameters. We further examine the empirical performance of the penalized composite likelihood estimator for model selection with an RCOR model. We consider different scenarios with $n = 250$, $n = 500$, and $p = 40$, $p = 60$, and $p = 100$. We include 30 different edge classes and 20 different vertex classes. We simulate multivariate Gaussian random vectors with entries of the sparse partial correlation matrix given by $\rho_{\mathcal{E}} = (0_{26}, 0.0628, -0.0534, 0.0380, 0.0519)$ and with $\theta_{\mathcal{V}} = (1.3181, 1.8676, 1.7880, 1.7626, 1.6550, 1.1539, 1.3975, 1.7877, 1.7090, 1.6931, 1.4631, 1.5131, 1.7084, 1.7344, 1.1442, 1.8060, 1.7447, 1.8522, 1.3146, 1.1001)$. In Table 4, one can see that the proposed method has satisfactory model selection property with very low false negative and false positive results. With $n = 500$ and $p = 60$, our approach has an average of 0 false negative results and 10.38 false positive results. In comparison, the LASSO has an average of 212.32 false negatives and 32.72 false positives and SCAD has an average of 212.10 false negatives and 33.31 false positives. These results exemplify that if the data are generated from a clustered network, the symmetry-constrained approach, whether with an RCON or an RCOR model, fully uses the clustering structure in model selection and outperforms the unconstrained approach.

6. APPLICATION

We now apply our proposed method to a real biological dataset. The experiment was conducted to examine how GM-CSF modulates global changes in neutrophil gene expressions (Kobayashi et al. 2005). Time course summary PMNs were isolated from venous blood of healthy individuals. Human PMNs (107) were cultured with and without 100 ng/mL GM-CSF for up to 24 h. The experiment was performed in triplicate, using PMNs from three healthy individuals for each treatment. There are in total 12,625 genes monitored, each gene is measured nine times at time 0, and then measured six times at time 3, 6, 12, 18, 24. At each of these five points, three of the six measurements were obtained for the treatment group and the other three were obtained for the control group. We first proceed with standard gene expression analysis. For each gene, we perform an ANOVA test on the treatment effect while acknowledging the time effect. We rank the F statistic for each gene and select the top 1000 genes that have the most significant changes in expression between treatment and control group. Our goal is to study the network among these 1000 genes. We

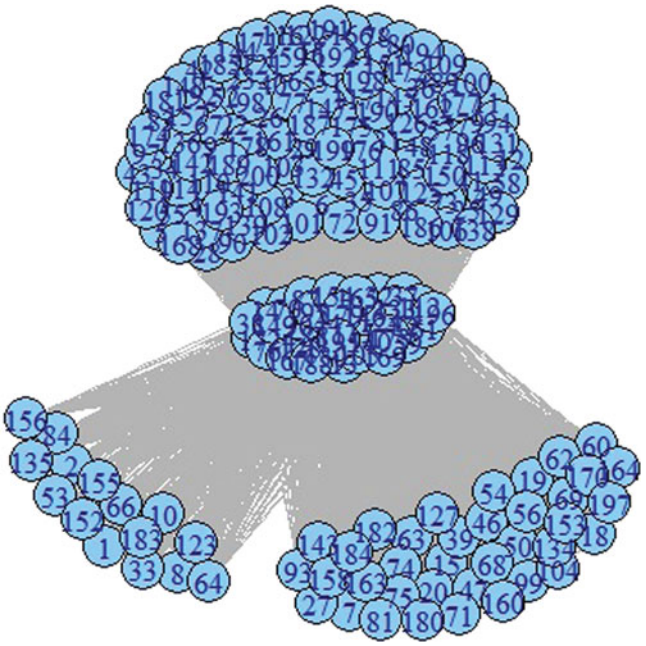

Figure 1. The clustered gene networks for the top ranked 1000 individual genes is estimated based on the symmetry-constrained SCAD. Only the subgraph for the first 200 genes is depicted.

cluster these genes using the spectral clustering method (Qin and Rohe 2013). The number of clusters is chosen by inspecting the biggest gap between the eigenvalues of the Laplacian matrix of the data matrix. There are five leading large eigenvalues followed by 995 small eigenvalues. Therefore, the number of clusters is chosen to be five. The genes clustered together can be viewed as a group of genes who share similar expression profiles. This imposes symmetry constraints to the networks modeling. We assume that edges connecting the same pair of clusters or edges linking genes from the same cluster belong to the same edge class and vertices belonging to the same cluster belong to the same vertex class. Therefore, there is a total of 15 edge classes and 5 vertex classes parameters to be estimated based on a 1000×1000 data matrix. We perform penalized symmetry-constrained SCAD estimation and the tuning parameter is selected using the BIC and the Comp-BIC criteria. Both BIC and COMP-BIC are minimized at the same subset model. The selected model has five nonzero edge classes which include 335,196 nonzero edges. In Figure 1, the gene network among individual genes is depicted. Some of the underlying clustering structure is evident from the plot. The model is sparse containing only a few nonzero edge classes. Nevertheless, the overall network is very dense with more than 335,000 nonzero edges. Due to space limitations, we only show the estimated graph of the first 200 genes. For comparison purposes, we also perform network estimation based on the plain unconstrained SCAD, and the selected subset model has 4446 edges. This is a much sparser network than the one obtained by the symmetry constrained approach. Due to space limitations, Figure 2(a) shows the sparse network obtained by SCAD for the first 200 genes only. There is a small group of genes with at least one edge in Figure 2(a). In Figure 2(b), we zoom into

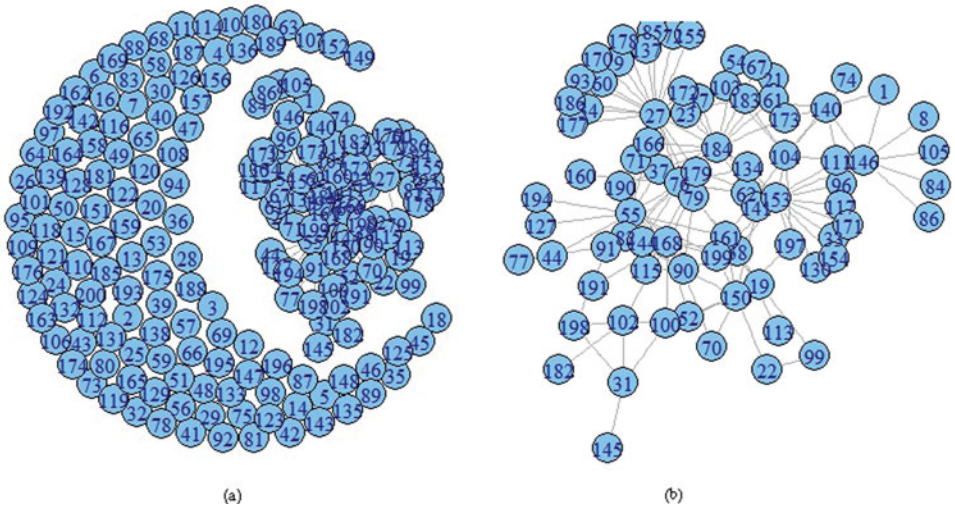

Figure 2. The gene networks for the top ranked 1000 individual genes is estimated based on the unconstrained SCAD. Only the subgraph for the first 200 genes is depicted.

this group and depict the network among all the genes that have at least one edge. The validity of the symmetry-constrained approach is dependent upon the underlying clustering structure. If such a clustering structure is based on some prior biological knowledge, the symmetry-constrained approach is the option of choice. For many biological systems, the sparsity assumption could be too constraining. Our proposed approach offers an alternative tool to model a potentially dense network.

7. CONCLUSION

For symmetry constrained RCON or RCOR graphical Gaussian models, the penalized composite likelihood based on conditional distributions offers a computationally convenient way to perform estimation and model selection while maintaining efficiency of the estimator. When the Gaussian graphical model is parameterized, in terms of edge and vertex classes, it is shown that the proposed penalized composite likelihood estimator will threshold the estimates for zero parameters to 0 with probability tending to 1 and the asymptotic distribution of the estimates for nonzero parameters follow the multivariate normal distribution corresponding to the estimation under the true submodel. In the literature, high-dimensional network modeling has been mainly restricted to sparse models. When the actual network is dense, symmetry constraints can be imposed onto the model to reflect the underlying symmetric structure and reduce the dimensionality of the model. It is a very useful dimension reduction strategy to model high-dimensional dense network with clusters.

SUPPLEMENTARY MATERIALS

Supplementary file: Proofs and technical derivations. (pdf file).

Code package: MATLAB and R codes for simulations. (zip file).

Data analysis: The web address for the gene expression dataset and the R codes to analyze the gene expression data. (zip file).

ACKNOWLEDGMENTS

The authors gratefully acknowledge the support of their respective NSERC Discovery grants. The authors express their gratitude to the editor, the associate editor, and all referees for their insightful comments and suggestions which led to a greatly improved article.

[Received February 2013. Revised June 2014.]

REFERENCES

Banerjee, O., Ghaoui, L. E., and D'Aspremont, A. (2007), "Model Selection Through Sparse Maximum Likelihood Estimation," *Journal of Machine Learning Research*, 9, 485–516. [910]

Besag, J. (1974), "Spatial Interaction and the Statistical Analysis of Lattice Systems," *Journal of the Royal Statistical Society*, Series B, 36, 192–236. [910]

Breheny, P., and Huang, J. (2011), "Coordinate Descent Algorithms for Nonconvex Penalized Regression, With Applications to Biological Feature Selection," *Annals of Applied Statistics*, 5, 232–253. [910,915]

Cox, D. R., and Reid, N. (2004), "A Note on Pseudolikelihood Constructed From Marginal Densities," *Biometrika*, 91, 729–737. [912,913]

Fan, J., and Li, R. (2001), "Variable Selection via Nonconcave Penalized Likelihood and Its Oracle Properties," *Journal of the American Statistical Association*, 96, 1348–1360. [914]

Fan, J., Feng, Y., and Wu, Y. (2009), "Network Exploration via the Adaptive LASSO and SCAD Penalties," *Annals of Applied Statistics*, 3, 521–541. [910]

Friedman, J., Hastie, T., Hofling, H., and Tibshirani, R. (2007), "Pathwise Coordinate Optimization," *Annals of Applied Statistics*, 1, 302–332. [915,918]

Friedman, J., Hastie, T., and Tibshirani, R. (2008), "Sparse Inverse Covariance Estimation With the Graphical LASSO," *Biostatistics*, 9, 432–441. [910,925]

——— (2010), "Applications of the LASSO and Grouped LASSO to the Estimation of Sparse Graphical Models," technical report, Stanford University, Department of Statistics. [910]

Gao, X., and Song, P. (2010), "Composite Likelihood Bayesian Information Criteria for Model Selection in High-Dimensional Data," *Journal of the American Statistical Association*, 105, 1531–1540. [921]

Højsgaard, S., and Lauritzen, S. L. (2007), "Inference in Graphical Gaussian Models With Edge and Vertex Symmetries With the gRc Package for R," *Journal of Statistical Software*, 23, 1–26. [910]

——— (2008), "Graphical Gaussian Models With Edge and Vertex Symmetries," *Journal of the Royal Statistical Society*, Series B, 70, 1005–1027. [910,911]

Kobayashi, S. D., Voyich, J. M., Whitney, A. R., and DeLeo, F. R. (2005), "Spontaneous Neutrophil Apoptosis and Regulation of Cell Survival by Granulocyte Macrophage-Colony Stimulating Factor," *Journal of Leukocite Biology*, 78, 1408–1418. [925]

Lauritzen, S. L. (1996), *Graphical Models*, Oxford: Clarendon Press. [911]

Lindsay, B. (1988), "Composite Likelihood Methods," *Statistical Inference from Stochastic Processes*, ed. Prabhu, N. U., Providence, RI: American Mathematical Society, pp. 221–239. [913]

Ma, S., Gong, Q., and Bohnert, H. J. (2007), "An Arabidopsis Gene Network Based on the Graphical Gaussian Model," *Genome Research*, 17, 1614–1625. [910]

Mardia, K. V., Kent, J., Hughes, G., and Taylor, C. C. (2009), "Maximum Likelihood Estimation Using Composite Likelihoods for Closed Exponential Families," *Biometrika*, 96, 975–982. [913]

Mazumder, R., Friedman, J. H., and Hastie, T. (2011), "SparseNet: Coordinate Descent With Nonconvex Penalties," *Journal of the American Statistical Association*, 106, 1125–1138. [910,915]

Ng, A., Jordan, M., and Weiss, Y. (2001), "On Spectral Clustering: Analysis and an Algorithm," *Advances in Neural Information Processing Systems*, Cambridge, MA: MIT Press, pp. 849–856. [918]

Qin, T., and Rohe, K. (2013), "Regularized Spectral Clustering Under the Degree-Corrected Stochastic Blockmodel," *arXiv*: 1309.4111. [918,926]

Shi, J., and Malik, J. (2000), "Normalized Cuts and Image Segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22, 888–905. [919]

Tibshirani, R. (1996), "Regression Shrinkage and Selection via the Lasso," *Journal of the Royal Statistical Society,* Series B, 58, 267–288. [914]

Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., and Knight, K. (2005), "Sparsity and Smoothness via the Fused LASSO," *Journal of the Royal Statistical Society*, Series B, 67, 91–108. [918]

Toh, H., and Horimoto, K. (2002), "Inference of a Genetic Network by a Combined Approach of Cluster Analysis and Graphical Gaussian Modeling," *Bioinformatics*, 18, 287–297. [909]

Tseng, P. (2001), "Convergence of a Block Coordinate Descent Method for Nondifferentiable Minimization," *Journal of Optimization Theory and Applications*, 109, 475–494. [915]

Varin, C. (2008), "On Composite Marginal Likelihoods," *AStA Advances in Statistical Analysis*, 92, 1–28. [912,913]

Yuan, M., and Lin, Y. (2007), "Model Selection and Estimation in the Gaussian Graphical Model," *Biometrika*, 94, 19–35. [910]

Zou, H., and Li, R. (2008), "One-Step Sparse Estimates in Nonconcave Penalized Likelihood Models" (with discussion), *The Annals of Statistics*, 36, 1509–1533. [915]

Zhang, C.-H. (2010), "Nearly Unbiased Variable Selection Under Minimax Concave Penalty," *Annals of Statistics*, 38, 635–1285. [910,914]