Published in final edited form as:

Stat Med. 2019 June 15; 38(13): 2364–2380. doi:10.1002/sim.8112.

Assisted graphical model for gene expression data analysis

Xinyan Fan¹, Kuangnan Fang^{1,2}, Shuangge Ma^{1,3}, Shuaichao Wang⁴, and Qingzhao Zhang^{1,2,5}

¹Department of Statistics, School of Economics, Xiamen University, Xiamen, China

²Fujian Key Laboratory of Statistical Sciences, Xiamen University, Xiamen, China

³Department of Biostatistics, Yale School of Public Health, New Haven, Connecticut

⁴School of Life Sciences and Biotechnology, Shanghai Jiao Tong University, Shanghai, China

⁵The Wang Yanan Institute for Studies in Economics, Xiamen University, Xiamen, China

Abstract

The analysis of gene expression data has been playing a pivotal role in recent biomedical research. For gene expression data, network analysis has been shown to be more informative and powerful than individual-gene and geneset-based analysis. Despite promising successes, with the high dimensionality of gene expression data and often low sample sizes, network construction with gene expression data is still often challenged. In recent studies, a prominent trend is to conduct multidimensional profiling, under which data are collected on gene expressions as well as their regulators (copy number variations, methylation, microRNAs, SNPs, etc). With the regulation relationship, regulators contain information on gene expressions and can potentially assist in estimating their characteristics. In this study, we develop an assisted graphical model (AGM) approach, which can effectively use information in regulators to improve the estimation of gene expression graphical structure. The proposed approach has an intuitive formulation and can adaptively accommodate different regulator scenarios. Its consistency properties are rigorously established. Extensive simulations and the analysis of a breast cancer gene expression data set demonstrate the practical effectiveness of the AGM.

Keywords

assisted analysis; gene expression; graphical model; multidimensional omics data

1 | INTRODUCTION

In omics studies, the important role of gene expression data cannot be overly stressed. Findings from gene expression data analysis have had a significant impact on basic, translational, and clinical sciences. In the analysis of gene expression data, network (graph)-based methods, which take a system perspective, have been shown to be more informative and more powerful than individual-gene- and geneset-based analysis. Extensive

methodological, computational, and theoretical research has been conducted on network-based methods. There are two main families of network construction approaches. The first family is unconditional, which models the connection between two genes independent of the other genes. A representative approach is the weighted gene co-expression network analysis (WGCNA).¹ The second family is conditional, which determines whether two genes are connected conditional on the other genes. Representative methods include neighborhood selection,² graphical Lasso,^{3,4} and the constrained ℓ_1 minimization approach.⁵ The conditional analysis can be more informative and, at the same time, more challenging.

A popular conditional construction approach proceeds as follows. For n iid observations, denote $\mathbf{y}_1, \dots, \mathbf{y}_n \in \mathbb{R}^p$ as their length- p gene expression measurements. Assume that $\mathbf{y}_i \sim \mathcal{N}(0, \Sigma)$ for all i , where the zero mean can be achieved by normalization and Σ is the positive definite $p \times p$ covariance matrix. It has been proved that two genes, conditional on the other $p - 2$ genes, are independent if and only if their corresponding element in $\Theta^0 = \Sigma^{-1}$ is zero. As such, determining the gene expression network structure can be effectively formulated as a problem of sparsely estimating Σ^{-1} . Multiple sparse estimation approaches have been developed. With its satisfactory theoretical and computational properties, the penalization technique has been extensively adopted. Assume that all gene expressions have been normalized to have mean zero. Consider $Y = (\mathbf{y}_1, \dots, \mathbf{y}_n)^\top$ and $S = \frac{1}{n-1} Y^\top Y$. The popular graphical Lasso approach^{3,4,6-8} considers the estimate (up to a constant)

$$\hat{\Theta} = \operatorname{argmax}_{\Theta} \left\{ \log(\det(\Theta)) - \operatorname{tr}(S\Theta) - \lambda \|\Theta^{-}\|_1 \right\}, \quad (1)$$

subject to the constraint that Θ is positive definite. In (1), $\lambda \geq 0$ is a tuning parameter, and $\|\Theta^{-}\|_1$ denotes the sum of the absolute values of the off-diagonal elements of Θ .

Gene expression studies have the “large p , small n ” characteristic. In network construction, the number of parameters to be estimated is $\mathcal{O}(p^2)$, which can be very large even with a moderate p . As such, although many promising successes have been achieved, the network construction result from the analysis of a single data set is still often unsatisfactory. To tackle this problem, one strategy that has been developed in the literature is to pool and jointly analyze multiple data sets.^{9,10} This has been referred to as “horizontal integration”. In recent omics studies, a prominent trend is to conduct multidimensional profiling, under which data are collected on gene expressions as well as their regulators (copy number variations, ie, CNVs, microRNAs, methylation, SNPs, and others) on the same subjects. With the regulation relationship, regulators contain information on gene expressions and can potentially *assist in estimating their properties*. In the contexts of regression analysis¹¹ and clustering,¹² assisted analysis methods have been developed and shown to outperform gene-expression-only analysis. However, such analysis has not been conducted in network construction, which, as shown in the literature, differs significantly from regression and clustering analysis and can be more challenging.

In this article, we consider data with measurements on gene expressions as well as their regulators. The objective is to more accurately estimate the network (graph) structure of

gene expressions with the assistance of information in regulators. For this purpose, a novel assisted analysis method is developed, which has an intuitive formulation and can adaptively accommodate different regulation scenarios. This study is related to but significantly advances from published studies in multiple aspects. Specifically, taking into account information in regulators may make the analysis more informative but at the same time more challenging than the analysis of gene expression data only. The analyzed data have characteristics significantly different from those in the horizontal integrative analysis and hence demand different techniques. It is noted that the proposed analysis also differs from the “vertical integration” analysis. In vertical integration, usually, the goal is to estimate a “mega” graph that is composed of both gene expressions and regulators.¹³ In contrast, our goal is to more accurately estimate the gene expression graph, which has been motivated by the “centrality” of gene expressions. Although assisted analysis has been conducted for regression and clustering, with the significant difference of network construction, new developments are needed.

The rest of this article is organized as follows. The assisted graphical model (AGM) approach is developed in Section 2, and its statistical properties and computational algorithm are established. Numerical studies, including simulation in Section 3 and the analysis of a breast cancer data set in Section 4, demonstrate its satisfactory practical performance. Concluding remarks are presented in Section 5, and additional technical and numerical details are presented in the Appendix.

2 | METHODS

For each subject, beyond the p gene expressions, assume that data are also available on q regulators. Gene expressions are regulated by multiple types of regulators, each of which is multiple-/high-dimensional. When multiple types of regulators are present, the work of Zhu et al¹⁴ and other published studies suggest stacking them together and creating a “mega” vector of regulators. Denote $X = (\mathbf{x}_1, \dots, \mathbf{x}_n)^T$ as the $n \times q$ data matrix of regulators with $\sum_{i=1}^n \mathbf{x}_i = \mathbf{0}$.

2.1 | Estimating the covariance matrix

In (1), a proper estimation of the covariance matrix is essential. In the first step of the proposed analysis, we first develop an alternative estimate of the covariance matrix with the assistance of information in X .

Consider the model

$$Y = XB + W, \quad (2)$$

where $B = (b_{ij})$ is the $q \times p$ matrix of unknown regression coefficients and represents the “transition” from regulators to gene expressions, and the $n \times p$ matrix W accommodates both “random errors” as well as regulation mechanisms not measured. When X includes all relevant regulators (that is, W consists of “noises” only), or when the regulation mechanisms

in X and W are independent, $E(\mathbf{x}\mathbf{w}^T) = \mathbf{0}$. As such, $\Sigma = B^T \Sigma_X B + \Sigma_W$. With this result, we propose an alternative estimate of the covariance matrix as

$$\bar{S} = B^T S_X B + S_W, \quad (3)$$

where S_X and S_W are the sample estimates of Σ_X and Σ_W . Specifically, S_X can be directly computed as the covariance matrix of X , and S_W can be computed using the estimated residuals from (2).

Remark 1. Modeling the gene expression-regulator relationship.—In (2), a linear regression model is adopted. The same model has been adopted in the works of Zhu et al,¹⁴ Shi et al,¹⁵ and many others and shown to be highly effective. In principle, it is possible to adopt more complicated models, for example, those with interactions and/or nonparametric effects. However, this may dramatically increase computational cost, lead to unreliable estimates (given the limited sample size), and is hence not pursued. It is also possible to derive the gene expression-regulator relationship from biological experiments and/or published literature. However, such information is still partial and hence not used here.

Remark 2. Estimating the regression coefficient matrix B .—In (3), B is unknown and needs to be replaced with an estimate. Consider

$$\hat{B} = \operatorname{argmin}_B \left\{ \frac{1}{2n} \|Y - XB\|_F^2 + \sum_{i=1}^q \sum_{j=1}^p P_1(b_{ij}; \mu, a) \right\}, \quad (4)$$

where $\|\cdot\|_F$ denotes the Frobenius norm, and $P_1(t; \mu, a) = \mu \int_0^t (1 - x/\mu a)_+ dx$ is the minimax concave penalty (MCP) function with tuning parameter μ and regularization parameter a .¹⁶ Here, penalized estimation is adopted to accommodate the high dimensionality and under the assumption that B is sparse (that is, each gene expression is only regulated by a small number of regulators, and each regulator regulates the expression levels of a small number of genes). The MCP can be replaced by other penalties. It is noted that similar penalization strategies have been adopted in multiple published studies.^{12,17}

Loosely speaking, the sparsity assumption and $E(\mathbf{x}\mathbf{w}^T) = \mathbf{0}$ impose a certain structure on the covariance matrix of Y . Similar strategies (improving estimation via imposing structures) have been adopted in the literature. A representative example is the factor model developed. It is noted that the biological and statistical assumptions of our approach and theirs are significantly different. Specifically, the covariance matrix Σ_W is assumed diagonal and sparse in the factor model studies.^{18,19} However, in this paper, the sparsity assumption is imposed on Θ rather than Σ_W . In addition, in the literature, approaches have also been developed for more accurate estimation of the covariance matrix or covariance function, for purposes such as the estimation of mean regression parameters or functions.²⁰ However, it is noted that the data settings and analysis goals are quite different from those in this article.

2.2 | AGM estimation

Motivated by the development in the previous section, we propose the following estimate as an alternative to (1):

$$\operatorname{argmax}_{\Theta} \left\{ \log(\det(\Theta)) - \operatorname{tr}[(1 - \alpha)S + \alpha\bar{S}]\Theta - \lambda \|\Theta^{-}\|_1 \right\}, \quad (5)$$

subject to the constraint that Θ is positive definite. $\alpha \in [0, 1]$ is a tuning parameter.

The proposed estimate shares the same strategy as in (1). The difference is that $(1 - \alpha)S + \alpha\bar{S}$, which balances between S and $\bar{S}$, takes the place of S . As described earlier, when $E(\mathbf{x}\mathbf{w}^T) = \mathbf{0}$, $\bar{S}$ is a more sensible estimate than S . However, this assumption not necessarily holds. For example, W may contain unmeasured regulation mechanisms that are correlated with those in X . In practical data analysis, the correlation between X and W is unknown. Thus, we take a weighted average, with the weight determined by data.

The objective function in (5) can be rewritten as

$$\left\{ [\log(\det(\Theta)) - \operatorname{tr}(S\Theta)] + \frac{\alpha}{1 - \alpha} [\log(\det(\Theta)) - \operatorname{tr}(\bar{S}\Theta)] - \frac{\lambda}{1 - \alpha} \|\Theta^{-}\|_1 \right\},$$

which has a more lucid interpretation. That is, the proposed objective function is a weighted average of graphical Lasso with S and $\bar{S}$, with a data-dependent weight. This strategy has a Bayesian “flavor” and shares some similarity with the prior Lasso approach.²¹ That is, in a sense, $\bar{S}$ is determined by the “prior knowledge” of the sparsity of B and uncorrelation of X and W . However, there are significant differences from the prior Lasso. Specifically, both S and $\bar{S}$ are computed from data as opposed to being extracted from other prior studies. In addition, network construction as considered here is quite different from regression analysis in the prior Lasso.

2.3 | Statistical properties

Denote the true covariance matrix and precision matrix of $\mathbf{y}$ as $\Sigma_{p \times p}$ and $\Theta_{p \times p}^0$, respectively. Denote the true value of B as $B_{q \times p}^0 = (b_{ij}^0)$. Assume $E(\mathbf{x}^T \mathbf{w}) = 0$. Denote θ_{ij}^0 as the (i, j) th component of Θ^0 . Define $\mathcal{A}_0 = \{(i, j) : \theta_{ij}^0 \neq 0, i = 1, \dots, p, j = 1, \dots, p\}$ and $s = |\mathcal{A}_0| - p$, which is the number of nonzero elements in the off-diagonal entries of Θ^0 . The following conditions are assumed.

Condition 1.—In (4), consider a generic penalty $P_1(t, \mu)$, where $\mu > 0$ is the tuning parameter. Note that, for MCP, we suppress its dependence on the regularization parameter. Then, $P_1'(t; \mu)$ is nonincreasing in $t \in (0, \infty)$ and $P_1'(0_+; \mu) = O(\sqrt{\log p/n})$.

Condition 2.—The true regression coefficient matrix $B_{q \times p}^0$ is sparse. There is a constant C that $\max_{j \in \{1, \dots, p\}} \sum_{i=1}^q |b_{ij}^0| \leq C$.

Condition 3.—Let $a_j = \max_{1 \leq i \leq q} \{P_1'(|b_{ij}^0|; \mu); b_{ij}^0 \neq 0\}$, then $a_j = O\left(n^{-\frac{1}{2}}\right)$ for $j = 1, \dots, p$.

Let $g_j = \max_{1 \leq i \leq q} \{P_1''(|b_{ij}^0|; \mu); b_{ij}^0 \neq 0\}$, then $g_j \rightarrow 0$ as $n \rightarrow \infty$ for $j = 1, \dots, p$. In addition, there are constants η and ξ such that $t_1, t_2 > \eta\mu$, $|P_1''(t_1; \mu) - P_1''(t_2; \mu)| \leq \xi|t_1 - t_2|$.

Condition 4.—There exist constants τ_1, τ_2 such that $0 < \tau_1 < \phi_{\min}(\Sigma) < \phi_{\max}(\Sigma) < \tau_2 < \infty$, where $\phi_{\min}$ and $\phi_{\max}$ denote the smallest and largest eigenvalues.

The aforementioned conditions are mild and comparable to those in the literature. Specifically, Condition 1 assumes the boundedness of the first-order derivative of the penalty as well as its convergence rate. Condition 2 is the matrix sparsity condition and has been motivated by the sparse regulation relationship. Condition 3 has also been assumed in the work of Fan and Peng.²² Following the work of Fan and Peng,²² if $q^4/n \rightarrow 0$ as $n \rightarrow \infty$, then $\|\hat{B}_{\cdot, j} - B_{\cdot, j}^0\|_2 = O(\sqrt{q/n})$, where $\hat{B}_{\cdot, j}$ and $B_{\cdot, j}^0$ are the j th columns of $\hat{B}$ and B^0 , and $\|\cdot\|_2$ is ℓ_2 norm of a vector. Condition 4 is standard.^{7,8} It guarantees that the inverse of Σ exists and is well conditioned. The following theorem establishes the consistency of $\hat{\Theta}$.

Theorem 1.—Let $\hat{\Theta}$ be the maximizer defined in (5). Under Conditions 1 to 4, if $\lambda = O(\sqrt{\log p/n})$, $(s + p)(\log(p))^k/n = O(1)$ for some $k > 1$ and $q^4/n \rightarrow 0$, then

$$\|\hat{\Theta} - \Theta^0\|_F = O_P\left(\sqrt{\frac{(s + p)\log p}{n}}\right).$$

This theorem explicitly establishes conditions on s , p , and q , under which $\hat{\Theta}$ is a consistent estimate. The number of nonzero elements $(s + p)$, dimensionality, and sample size affect the rate of convergence. The total bias is at the rate $O_P(\sqrt{p \log p/n})$ if $s = O(p)$. Since $\log p$ diverges slowly, p can be comparable to n .⁸ The condition $q^4/n \rightarrow 0$, following the work of Fan and Peng,²² can be relaxed. An inspection of the proof reveals that all we have to make sure is that the maximum absolute column sum of the estimate of B is bounded. In addition, under additional conditions, for example, $\Sigma \otimes \Sigma$ satisfying the irrerepresentable condition and others,²³ we can get $\|\hat{\Theta} - \Theta^0\|_\infty = O_P(\sqrt{\log p/n})$. For more details, we refer to the work of Ravikumar et al.²³ Proof is presented in the Appendix.

2.4 | Computation

Computation is accomplished in two steps. In the first step, we compute the estimate of B . With the MCP, this can be accomplished using the coordinate descent algorithm and existing software. In this paper, we use the R package “ncvreg” to estimate B .²⁴ Note that this step can be done in a highly parallel manner to reduce computer time. The regularization parameter a controls the degree of concavity and unbiasedness. Smaller values of a are better

at retaining the unbiasedness of MCP while making the penalty more concave, which may lead to difficulty in optimization. Therefore, it is advisable to choose a value that is “big enough”, but “not too big”. Breheny and Huang²⁴ suggested setting $a = 3$. We have experimented with different a values and reached the same conclusion. The tuning parameter μ is chosen using 5-fold cross validation. The second step is a graphical Lasso and can be accomplished with existing algorithms and software. The R package “*glasso*” is used to accomplish the second step.^{4,6} There are two tuning parameters involved, which can be chosen using cross validation. Note that, in the literature,^{10,25} there have been extensive discussions on tuning parameter/model selection in graph models, which are also applicable to our case. With the algorithms for both steps well developed, computation does not pose a challenge.

3 | SIMULATION

Simulation is conducted to assess the performance of the proposed method. As shown in Figure 1, we consider three popular network structures, namely, the Erdos-Renyi, scale-free, and nearest-neighbor. All three structures have been well examined in the literature.^{9,26} Briefly, we generate the Erdos-Renyi network with probability 0.05 for drawing an edge between two arbitrary vertices. The scale-free network is generated with two edges to add in each step. The nearest-neighbor network is generated by modifying the data generating mechanism described in the work of Guo et al.⁹ Specifically, we generate p points randomly on a unit square, calculate all $p(p-1)/2$ pairwise distances, and find the k nearest neighbors of each point. The nearest-neighbor network is obtained by linking any two points that are among the k -nearest neighbors of each other. Integer k controls the degree of sparsity of the network, and we set $k = 5$ in our simulation.

With a given network structure, we generate the corresponding covariance matrix as follows. The $p \times p$ matrix Θ is created. We set those elements not corresponding to edges as zero. For elements corresponding to edges, we generate values randomly from a uniform distribution with support $[-0.4, -0.1] \cup [0.1, 0.4]$. To ensure positive definiteness, we set $\theta_{ij} = \Sigma_i \cdot j \theta_{ij} + 0.1$. Finally, $\Sigma = \Theta^{-1}$.

As described in details later, we consider five examples, which serve different purposes. In Example 1, data are generated under the assumed models, the percentage of Σ that can be explained by $B^T \Sigma_X B$ is moderate, and each gene expression is regulated by at least one regulator. In Example 2, we examine how the percentage of Σ that can be explained by $B^T \Sigma_X B$ affects the performance of AGM. In Example 3, we allow a subset of gene expressions not regulated by regulators. Examples 2 and 3 are designed to examine different aspects of the regulation strengths on performance. In Example 4, we consider the situation where gene expressions also depend on regulators that are not collected in X . This example accommodates the practical scenario that profiling may not be “complete”. In Example 5, we illustrate how the nonlinear regulation relationships affect the performance of AGM. Examples IV and V are designed to examine performance under model misspecification and can serve as a test of sensitivity.

To gauge the performance of the proposed approach, we comprehensively consider multiple measures. We first consider the receiver operating characteristic (ROC) curve, which is generated by considering a sequence of tuning parameter λ 's values and evaluating true positive rate (TPR) and false positive rate (FPR) at each value. Second, we consider the precision-recall curve, which is the plot of positive predictive value (PPV) versus TPR and generated in a similar manner as the ROC. We also consider error, which is defined as $ER = \sum_{i,j} (\theta_{ij} - \hat{\theta}_{ij})^2$, with θ_{ij} s and $\hat{\theta}_{ij}$ s as the entries of Θ and $\hat{\Theta}$, respectively. In addition, we also consider the dKL of the estimated distribution from the true distribution, which is defined as $\frac{1}{2}[-\log(\det(\hat{\Theta} \Sigma)) + \text{tr}(\hat{\Theta} \Sigma)]$. For ER and dKL, following the same spirit as ROC and precision-recall curve, we vary the tuning parameter values (and hence NUM - the number of identified edges) and examine them as functions of NUM. Among the four measures, the first two are on selection accuracy, and the latter two are on estimation accuracy. In the second set of evaluation, we select tunings using the 5-fold cross validation approach and evaluate TPR, FPR, ER, and dKL values at the optimal tunings. To establish the value of assisted analysis, we compare AGM with its direct competitor, the standard graphical model (GM) approach. We note that there are many other ways for constructing gene expression networks. Comparing the AGM with approaches other than the GM may not be very sensible as they are built on different statistical grounds.

Example 1.

Set $n = 200$ and $p = 100$ and 200 . For B , we consider two structures: (i) block. Here, $B = I_{p/5 \times p/5} \otimes B_5^* \times 5$, where I is identity matrix and B^* has diagonal elements $B_{jj}^* = 2$ and off-diagonal elements $B_{ij}^* = 0.9^{|i-j|}$; (ii) banded, where $B_{ij} = 2\mathbf{1}(i-j=0) + 1.6\mathbf{1}(i=j-4) + \mathbf{1}(i=j-2) - 2\mathbf{1}(i=j+2) - 1.4\mathbf{1}(i=j+5)$. $\mathbf{1}$ is the indicator function. Generate $r_j \sim \nu(0.6, 0.8)$ for $j = 1, \dots, p$. Let $d_1, \dots, d_p$ be the eigenvalues of Σ corresponding to eigenvectors $\mathbf{u}_1, \dots, \mathbf{u}_p$. We generate $\mathbf{x} \sim \mathcal{N}(0, \Sigma_x)$ with $B^\top \Sigma_x B = \sum_{j=1}^p r_j d_j \mathbf{u}_j \mathbf{u}_j^\top$ and $\mathbf{w} \sim \mathcal{N}(0, \sum_{j=1}^p (1-r_j) d_j \mathbf{u}_j \mathbf{u}_j^\top)$ independently. Lastly, we generate $\mathbf{y} = B^\top \mathbf{x} + \mathbf{w}$.

For the two tuning parameters in (5), α is newly introduced. To better appreciate its impact, for the four combinations (two p values and two B structures), we first consider α fixed at 0.4 and 0.6 and present the ROC curves, precision-recall curves, ER, and dKL results in Figures 2, 3, 4, and 5. The observed patterns for the three-network structures are very similar. The AGM dominates GM in all four measures. Specifically, its ROC and precision-recall curves are above those of GM in the whole range. In terms of estimation, when NUM is small, the ER and dKL values of AGM are better than of those GM, although the differences are not as prominent as when NUM is large. This observation is reasonable. When NUM is small, only the strongest (easiest) signals are identified. For those “easy targets”, GM can perform reasonably well. When NUM gets bigger, ie, when it is needed to identify weaker signals, the benefit of additional information becomes prominent. We have also more closely examined the results with $\alpha = 0.4$ and 0.6 and found that $\alpha = 0.6$ outperforms. This is reasonable, as under a proper model specification, $\bar{S}$ can be more effective than S , and more information should be borrowed. The satisfactory performance of

AGM is further confirmed in Table 1 with the cross-validation selected tunings. In terms of selection, AGM has significantly higher TPR values, at the price of small increases in FPR values. It has better ER and dKL performance.

Example 2.

In Example 1, r_j controls the percentage of Σ that can be explained by $B^T \Sigma_x B$. In this example, we take a closer look at the role of r_j . Specifically, we consider three signal levels: (i) low with $r_j \sim \mathbf{v}$ (0.1, 0.2), (ii) moderate with $r_j \sim \mathbf{v}$ (0.5, 0.6), and (iii) high with $r_j \sim \mathbf{v}$ (0.8, 0.9). We fix $p = 100$, $n = 200$, and a block structure for B (the same as under Example 1). The generation of $\mathbf{x}$ and $\mathbf{w}$ then follows the same steps as under Example 1. Detailed results on AGM are presented in Figure B1 (Appendix). There are several interesting observations. For a fixed network structure and a fixed r_j level, when α increases, the area under the ROC curve (AUC) value increases, which suggests the benefit of borrowing information from Sand assisted analysis. It is interesting to observe that $r_j \sim \mathbf{v}$ (0.5, 0.6) has the best performance. It performs better than $r_j \sim \mathbf{v}$ (0.1, 0.2) because of containing more useful information. For $r_j \sim \mathbf{v}$ (0.8, 0.9), gene expressions and regulators contain almost identical information, which leads to less improvement. Another observation is that the cross-validation selected tunings have good performance but do not reach the best AUC values. That is, there is still room for improvement, and more research on tuning parameter selection is needed. With the cross-validation selected tunings, in Table 2, we compare AGM and GM in terms of TPR, FPR, ER, and dKL. The observations are similar to those for Example 1: AGM has slightly inferior FPR values but excels in the other three measures.

Example 3.

In this example, some gene expressions are not regulated by the regulators contained in X . This reflects the fact that the regulating mechanisms of gene expressions are not completely known. Set $p = 100$, $n = 200$, and the block structure for B as under Example 1. In addition, set $r_j \sim \mathbf{v}$ (0.4, 0.5). Let $\tilde{B}_A = B_A$ and $\tilde{B}_{A^c} = 0$, where $A = \{(i, j) : i = 1, \dots, p, j = 1, \dots, K\}$, and A^c is the complement of A . Consider three levels of K : 60, 75, and 90. The generation of $\mathbf{x}$ and $\mathbf{w}$ then follows the same steps as under Example 1. Detailed results on AGM are presented in Figure B2 (Appendix). The comparison results with GM are presented in Table B1 (Appendix). The competitive performance of AGM is again observed.

Example 4.

We first generate data in the same manner as under Example 1 with $p = 100$, $n = 200$, $r_j \sim \mathbf{v}$ (0.6, 0.8), and a block structure for B . Then, $\pi = 25, 15$, and 5 regulators are removed from analysis. That is, the analysis is conducted with $p - \pi$ regulators. The analysis results are shown in Figure B3 and Table B2 (Appendix), which suggest that AGM can outperform GM.

Example 5.

Set $p = 100$, $n = 200$, and the block structure for B as under Example 1. In addition, set $r_j \sim \mathbf{v}$ (0.6, 0.8). Let $f(\mathbf{x}) = (f(x_1), \dots, f(x_p))^T$. $f(x_j)$'s are nonlinear for $j = 1, \dots, p_f$ whereas $f(x_j) = x_j$ otherwise. We generate $f(\mathbf{x}) \sim \mathcal{N}(0, \Sigma_x)$ with $B^T \Sigma_x B = \sum_{j=1}^p r_j d_j \mathbf{u}_j \mathbf{u}_j^T$, and

generate $\mathbf{w}$ as under Example 1. At last, we generate $\mathbf{y} = B^T f(\mathbf{x}) + \mathbf{w}$. Consider three levels of p_β ie, 10, 50, and 90, and two nonlinear forms $x^2 - 2.5$ and $\ln(x)$. Detailed results on AGM are presented in Figure B4 (Appendix). The comparison results with GM are presented in Table B3 (Appendix). The main observation is that, even when some regulation relationships are nonlinear, AGM still has competitive performance.

4 | DATA ANALYSIS

The Cancer Genome Atlas (TCGA) is a collective effort organized by NCI. High-quality profiling has been conducted on multiple cancer types. Here, we analyze breast invasive carcinoma, which is a very common cancer type. Data are downloaded from TCGA Provisional using the CGDS-R package. We refer to the TCGA website and published studies for more information on TCGA and this data set.²⁷ Following published studies, we analyze the processed level 3 data. For gene expression, we download and analyze the robust Z-score, which is a lowess-normalized, log-transformed, and median-centered version of gene expression data that takes into account all of the gene-expression arrays under consideration. It determines whether a gene is up- or down-regulated relative to the reference population. For regulators, we focus on CNVs, whose regulation of gene expressions has been long established. In TCGA, the loss and gain levels of copy number changes have been identified using segmentation analysis and the GISTIC algorithm and expressed in the form of log2 ratio of a sample versus the reference intensity. Data on 17 214 gene expressions and 22 247 CNVs are available. Jointly analyzing all measurements is computationally infeasible. Thus, we conduct a supervised screening using the overall survival and select the top 150 “most interesting” gene expressions. We then select 150 CNVs with the highest correlations with those gene expressions.

The analysis results using AGM and GM are presented in Figure 6. In the first row, we show the AGM (left) and GM (middle) network structures with tuning parameters selected using cross validation. The AGM identifies 2813 edges, with the median degree 37.5. The GM identifies 2806 edges, with the median degree 38.0. The second row of Figure 6 shows the difference between AGM and GM network structures of small, moderate, and strong connections. Specifically, the left network describes the difference between AGM and GM with tuning parameters selected using cross validation, which suggests that the two approaches lead to different networks. We further apply the hard thresholding technique on the AGM and GM network structures, with threshold values 0.1 and 0.2, respectively, to retain moderate and strong connections. The middle and right networks describe the differences between the moderate and strong connections. While the two networks are quite similar with threshold 0.2, the AGM and GM networks are still considerably different even with moderate connections. This suggests that, as observed in simulation, AGM makes a bigger difference for relatively weaker signals.

With real data and big network structures, it is difficult to assess edge identification and estimation accuracy. We resort to a random sampling approach, which may provide some support to the validity of analysis. Specifically, we split data into a training and a testing set, with sizes 4:1. With the training set, both AGM and GM are applied. Then, we compute the negative log-likelihood statistic $\text{tr}(\hat{\Theta}S) - \log(\det(\hat{\Theta}))$ with the training set $\hat{\Theta}$ and testing set S

to evaluate prediction. This process is repeated 100 times, out of which AGM has a better prediction 94 times. In addition, with the 100 random samplings, we also compute the probability that each edge is identified. As in the literature, this may serve as an evaluation of stability. For the edges selected by AGM using the whole data set, the average probability is 0.72, compared to 0.70 for GM. Both prediction and stability evaluations suggest the improvement of AGM over GM.

5 | DISCUSSION

The construction of gene expression networks has important implications. Beyond having their own independent value, the networks have also served as the basis of many downstream analyses. The availability of multidimensional profiling data and central importance of gene expression data analysis make the assisted analysis warranted. This study has advanced from the existing GM and assisted analysis in regression and clustering by developing an assisted GM approach. The proposed approach has an intuitive formulation and interpretation. Statistical and numerical studies show that the proposed approach has satisfactory consistency properties and competitive practical performance. Overall, this study provides a useful new venue for gene expression network analysis.

The GM approach assumes joint normality, which is “inherited” by the AGM. With practical gene expression data, this assumption may be violated. One potential remedy is to first conduct transformation²⁸ to achieve normality. However, this may make the analysis results much less interpretable. To ensure interpretability, we note that quite a few GM studies have analyzed gene expression data without transformation.^{26,29} The goal of this study is to improve over the GM, as opposed to relaxing the assumptions of GM. If desirable, approaches such as transformation, which have been applied to the GM, can also be coupled with the AGM.

It should be noted that, although described for gene expressions and regulators, the proposed approach may have broader applications. It only demands data on two types of measurements, with the second type of data connected to the first type of data via a regression model. For example, it is also applicable to the analysis of “protein + gene expression” data, “financial returns + equity market risks” data, and “PM2.5 + climate” data. Other potential applications can also be found in social science, engineering, and other fields.

ACKNOWLEDGEMENTS

We thank the associate editor and reviewers for their careful review and insightful comments, which have led to a significant improvement of this article. This study was supported by the National Natural Science Foundation of China (71471152), the National Bureau of Statistics of China (2016LD01, 2015629), the Fundamental Research Funds for the Central Universities (20720171064, 20720171095, 20720181003), and the National Institutes of Health (CA216017).

Funding information

National Natural Science Foundation of China, Grant/Award Number: 71471152; National Bureau of Statistics of China, Grant/Award Number: 2016LD01 and 2015629; Fundamental Research Funds for the Central Universities, Grant/Award Number: 20720171064, 20720171095 and 20720181003; National Institutes of Health, Grant/Award Number: CA216017

APPENDIX A

PROOF OF THEOREM 1

Inspired by the work of Lam and Fan,⁸ the key is to show $\|\bar{S} - \Sigma\|_{\infty} = O_P(\sqrt{\log p/n})$. Since

$$\|\bar{S} - \Sigma\|_{\infty} = \|\bar{S} - S + S - \Sigma\|_{\infty} \leq \|\bar{S} - S\|_{\infty} + \|S - \Sigma\|_{\infty},$$

we only need to separately prove $\|\bar{S} - S\|_{\infty} = O_P(\sqrt{\log p/n})$ and $\|S - \Sigma\|_{\infty} = O_P(\sqrt{\log p/n})$.

$\|S - \Sigma\|_{\infty} = O_P(\sqrt{\log p/n})$ is established in Lemma A.3³⁰ and holds under the assumption for $\log p/n = o(1)$.

We now show that $\|\bar{S} - S\|_{\infty} = O_P(\sqrt{\log p/n})$. Assume that both gene expressions and regulators have been normalized to have zero means. Then,

$$S - \bar{S} = \frac{1}{n-1} (Y^T Y - \hat{B}^T X^T X \hat{B} - \hat{W}^T \hat{W}) = \frac{1}{n-1} (\hat{B}^T X^T \hat{W} + \hat{W}^T X \hat{B}). \quad (A1)$$

From the definition of $\hat{B}$, we have

$$-\frac{1}{n} X^T Y + \frac{1}{n} X^T X \hat{B} + P'_1(|\hat{B}|; \mu) \circ \text{sign}(\hat{B}) = 0,$$

where “ $\circ$ ” is the Hadamard product operator. Here, $P'_1(|\hat{B}|; \mu) = (P'_1(|\hat{b}_{ij}|; \mu))_{q \times p}$. $\text{sign}(\hat{B})_{ij} = \text{sign}(\hat{b}_{ij})$ if $\hat{b}_{ij} \neq 0$, and $\text{sign}(\hat{B})_{ij} \in [-1, 1]$ if $\hat{b}_{ij} = 0$.

Then, $\frac{1}{n} X^T \hat{W} = P'_1(|\hat{B}|; \mu) \circ \text{sign}(\hat{B})$, and $\frac{1}{n} \|\hat{B}^T X^T \hat{W}\|_{\infty} = \|\hat{B}^T P'_1(|\hat{B}|; \mu) \circ \text{sign}(\hat{B})\|_{\infty}$.

Under Condition 1, $\|P'_1(|\hat{B}|; \mu)\|_{\infty} = O_P(\sqrt{\log p/n})$. Therefore,

$$\|\hat{B}^T P'_1(|\hat{B}|; \mu) \circ \text{sign}(\hat{B})\|_{\infty} \leq O_P\left(\sqrt{\frac{\log p}{n}}\right) \|\hat{B}^T \circ \text{sign}(\hat{B}^T)\|_{\infty},$$

where $\mathbf{1}$ is a $q \times p$ matrix with all entries equal to 1.

Under Condition 2, $\max_j \sum_{i=1}^q |b_{ij}^0| \leq C$, then $\|B^{0^T} \circ \text{sign}(B^{0^T})\|_{\infty} = O(1)$. Under

Condition 3, $\hat{B}$ converges to B^0 , so $\|\hat{B}^T \circ \text{sign}(\hat{B}^T)\|_{\infty} = O_P(1)$. From (A1),

$$\begin{aligned} \|S - \bar{S}\|_{\infty} &\leq \frac{1}{n-1} (\|\hat{B}^T X^T \hat{W}\|_{\infty} + \|\hat{W}^T X \hat{B}\|_{\infty}) \\ &= \frac{n-2}{n-1} \|\hat{B}^T X^T \hat{W}\|_{\infty} = O_P\left(\sqrt{\frac{\log p}{n}}\right). \end{aligned}$$

Then, Theorem 1 follows the proof of Theorem 1 in the work of Lam and Fan.⁸

APPENDIX B

ADDITIONAL FIGURES AND TABLES

FIGURE B1.

Simulation II: Performance of assisted graphical model. Right, middle, and left panels: Erdos-Renyi, scale-free, and nearest-neighbor networks. Black, red, and blue curves correspond to $r_j \sim \mathcal{U}(0.1, 0.2)$, $r_j \sim \mathcal{U}(0.5, 0.6)$, and $r_j \sim \mathcal{U}(0.8, 0.9)$. The solid points correspond to the cross validation selected tunings. AUC, area under the ROC curve; ROC, receiver operating characteristic [Colour figure can be viewed at wileyonlinelibrary.com]

FIGURE B2.

Simulation III: Performance of assisted graphical model. Right, middle, and left panels: Erdos-Renyi, scale-free, and nearest-neighbor networks. Black, red, and blue curves correspond to $K = 60, 75, 90$. The solid points correspond to the cross validation selected tunings. AUC, area under the ROC curve; ROC, receiver operating characteristic [Colour figure can be viewed at wileyonlinelibrary.com]

FIGURE B3.

Simulation IV: Performance of assisted graphical model. Right, middle, and left panels:

Erdos-Renyi, scale-free, and nearest-neighbor networks. Black, red, and blue curves

correspond to $\pi = 25, 15, 5$. The solid points correspond to the cross validation selected

tunings. AUC, area under the ROC curve; ROC, receiver operating characteristic [Colour

figure can be viewed at wileyonlinelibrary.com]

FIGURE B4.

Simulation V: Performance of assisted graphical model. First/second row: nonlinear forms $x^2 - 2.5$ and $\ln(x)$.

Right/middle/left column: Erdos-Renyi, scale-free, and nearest-neighbor networks. Black, red, and blue curves correspond to $p_f = 10, 50, 90$. The solid points correspond to the cross validation selected tunings. AUC, area under the ROC curve; ROC, receiver operating characteristic [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE B1

Simulation III: summary statistics on the models selected using cross validation

<i>K</i>		Erdos-Renyi Network				Scale-Free Network				Nearest-Neighbor Network			
		TPR	FPR	ER	dKL	TPR	FPR	ER	dKL	TPR	FPR	ER	dKL
60	GM	0.68	0.14	21.40	106.11	0.60	0.13	15.70	105.83	0.76	0.12	15.82	105.40
		(0.03)	(0.02)	(0.76)	(0.20)	(0.03)	(0.01)	(0.49)	(0.26)	(0.02)	(0.01)	(0.78)	(0.27)
	AGM	0.78	0.17	18.05	105.23	0.65	0.14	14.32	105.17	0.83	0.13	13.23	104.51
		(0.02)	(0.01)	(0.70)	(0.20)	(0.02)	(0.01)	(0.44)	(0.22)	(0.02)	(0.01)	(0.60)	(0.20)
75	GM	0.67	0.14	21.59	106.13	0.59	0.12	15.75	105.85	0.77	0.12	15.51	105.33
		(0.03)	(0.01)	(0.79)	(0.21)	(0.03)	(0.01)	(0.35)	(0.25)	(0.02)	(0.01)	(0.68)	(0.21)
	AGM	0.78	0.16	18.03	105.11	0.66	0.14	14.11	105.02	0.84	0.14	12.72	104.36
		(0.02)	(0.01)	(0.63)	(0.19)	(0.02)	(0.01)	(0.35)	(0.23)	(0.02)	(0.01)	(0.55)	(0.18)
90	GM	0.68	0.14	21.34	106.07	0.59	0.12	15.80	105.86	0.76	0.12	15.78	105.41
		(0.03)	(0.01)	(0.79)	(0.22)	(0.03)	(0.01)	(0.39)	(0.22)	(0.02)	(0.01)	(0.50)	(0.22)
	AGM	0.78	0.16	17.98	105.00	0.67	0.15	13.99	104.91	0.85	0.14	12.85	104.38
		(0.02)	(0.01)	(0.81)	(0.20)	(0.02)	(0.01)	(0.35)	(0.19)	(0.02)	(0.01)	(0.47)	(0.19)

Abbreviations: AGM, assisted graphical model; FPR, false positive rate; GM, graphical model; TPR, true positive rate.

TABLE B2

Simulation IV: summary statistics on the models selected using cross validation

	π	Erdos-Renyi Network				Scale-Free Network				Nearest-Neighbor Network			
		TPR	FPR	ER	dKL	TPR	FPR	ER	dKL	TPR	FPR	ER	dKL
GM	-	0.68	0.14	21.39	106.08	0.59	0.12	15.83	105.92	0.77	0.12	15.64	105.36
		(0.03)	(0.01)	(0.90)	(0.25)	(0.02)	(0.01)	(0.41)	(0.24)	(0.02)	(0.01)	(0.65)	(0.25)
AGM	25	0.73	0.16	20.08	105.75	0.64	0.14	14.67	105.47	0.79	0.13	15.28	105.24
		(0.03)	(0.02)	(0.85)	(0.25)	(0.02)	(0.01)	(0.43)	(0.22)	(0.02)	(0.01)	(0.58)	(0.23)
	15	0.75	0.16	19.75	105.66	0.65	0.14	14.50	105.36	0.80	0.13	14.84	105.09
		(0.03)	(0.02)	(0.83)	(0.24)	(0.03)	(0.01)	(0.44)	(0.20)	(0.02)	(0.01)	(0.57)	(0.21)
	5	0.78	0.17	18.64	105.38	0.65	0.14	14.47	105.28	0.82	0.14	14.42	104.94
		(0.03)	(0.01)	(0.88)	(0.23)	(0.02)	(0.01)	(0.38)	(0.21)	(0.02)	(0.01)	(0.57)	(0.19)

Abbreviations: AGM, assisted graphical model; FPR, false positive rate; GM, graphical model; TPR, true positive rate.

TABLE B3

Simulation V: summary statistics on the models selected using cross validation

Erdos-Renyi Network					Scale-Free Network				Nearest-Neighbor Network				
	p_f	TPR	FPR	ER	dKL	TPR	FPR	ER	dKL	TPR	FPR	ER	dKL
$\chi^2 - 2.5$													
GM	-	0.68	0.14	21.39	106.08	0.59	0.12	15.83	105.92	0.77	0.12	15.64	105.36
		(0.03)	(0.01)	(0.90)	(0.25)	(0.02)	(0.01)	(0.41)	(0.24)	(0.02)	(0.01)	(0.65)	(0.25)
AGM	10	0.77	0.17	18.87	105.43	0.64	0.14	14.86	105.41	0.82	0.14	14.33	104.91
		(0.03)	(0.02)	(0.95)	(0.22)	(0.02)	(0.01)	(0.42)	(0.21)	(0.02)	(0.01)	(0.64)	(0.22)
	50	0.75	0.17	19.53	105.63	0.63	0.14	15.30	105.68	0.81	0.14	14.69	105.09

	p_f	Erdos-Renyi Network				Scale-Free Network				Nearest-Neighbor Network			
		TPR	FPR	ER	dKL	TPR	FPR	ER	dKL	TPR	FPR	ER	dKL
		(0.03)	(0.01)	(0.87)	(0.25)	(0.03)	(0.02)	(0.42)	(0.21)	(0.02)	(0.01)	(0.59)	(0.22)
	90	0.74	0.17	20.02	105.76	0.63	0.14	15.37	105.76	0.80	0.14	15.07	105.22
		(0.03)	(0.02)	(0.91)	(0.24)	(0.03)	(0.02)	(0.42)	(0.22)	(0.02)	(0.01)	(0.56)	(0.23)
		ln(x)											
GM	-	0.68	0.14	21.39	106.08	0.59	0.12	15.83	105.92	0.77	0.12	15.64	105.36
		(0.03)	(0.01)	(0.90)	(0.25)	(0.02)	(0.01)	(0.41)	(0.24)	(0.02)	(0.01)	(0.65)	(0.25)
AGM	10	0.77	0.17	18.80	105.41	0.63	0.14	14.91	105.41	0.82	0.14	14.26	104.87
		(0.03)	(0.01)	(0.83)	(0.21)	(0.02)	(0.01)	(0.37)	(0.20)	(0.02)	(0.01)	(0.56)	(0.21)
	50	0.76	0.17	19.48	105.57	0.63	0.14	15.32	105.66	0.82	0.14	14.63	105.03
		(0.03)	(0.01)	(0.82)	(0.23)	(0.02)	(0.01)	(0.44)	(0.23)	(0.02)	(0.01)	(0.54)	(0.21)
	90	0.75	0.18	19.90	105.68	0.62	0.14	15.45	105.78	0.81	0.15	14.92	105.14
		(0.03)	(0.01)	(0.81)	(0.22)	(0.03)	(0.01)	(0.47)	(0.24)	(0.02)	(0.01)	(0.60)	(0.23)

Abbreviations: AGM, assisted graphical model; FPR, false positive rate; GM, graphical model; TPR, true positive rate.

REFERENCES

1. Zhang B, Horvath S. A general framework for weighted gene co-expression network analysis. *Stat Appl Genet Mol Biol*. 2005;4(1).
2. Meinshausen N, Bühlmann P. High-dimensional graphs and variable selection with the lasso. *Ann Stat*. 2006;34:1436–1462.
3. Yuan M, Lin Y. Model selection and estimation in the Gaussian graphical model. *Biometrika*. 2007;94(1):19–35.
4. Friedman J, Hastie T, Tibshirani R. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*. 2008;9(3):432–441. [PubMed: 18079126]
5. Cai T, Liu W, Luo X. A constrained ℓ_1 minimization approach to sparse precision matrix estimation. *J Am Stat Assoc*. 2011;106(494):594–607.
6. Witten DM, Friedman JH, Simon N. New insights and faster computations for the graphical lasso. *J Comput Graph Stat*. 2011;20(4):892–900.
7. Rothman AJ, Bickel PJ, Levina E, Zhu J. Sparse permutation invariant covariance estimation. *Electron J Stat*. 2008;2:494–515.
8. Lam C, Fan J. Sparsistency and rates of convergence in large covariance matrix estimation. *Ann Stat*. 2009;37(6B):4254–4278. [PubMed: 21132082]
9. Guo J, Levina E, Michailidis G, Zhu J. Joint estimation of multiple graphical models. *Biometrika*. 2011;98(1):1–15. [PubMed: 23049124]
10. Danaher P, Wang P, Witten DM. The joint graphical lasso for inverse covariance estimation across multiple classes. *J R Stat Soc Ser B Stat Methodol*. 2014;76(2):373–397.
11. Chai H, Shi X, Zhang Q, Zhao Q, Huang Y, Ma S. Analysis of cancer gene expression data with an assisted robust marker identification approach. *Genetic Epidemiology*. 2017;41(8):779–789. [PubMed: 28913902]
12. Hidalgo SJT, Wu M, Ma S. Assisted clustering of gene expression data using ANCut. *BMC Genomics*. 2017;18(1):623. [PubMed: 28814280]
13. Kim DC, Kang M, Zhang B, Wu X, Liu C, Gao J. Integration of DNA methylation, copy number variation, and gene expression for gene regulatory network inference and application to psychiatric disorders. In: 2014 IEEE International Conference on Bioinformatics and Bioengineering (BIBE); 2014; Boca Raton, FL.
14. Zhu R, Zhao Q, Zhao H, Ma S. Integrating multidimensional omics data for cancer outcome. *Biostatistics*. 2016;17(4):605–618. [PubMed: 26980320]

15. Shi X, Zhao Q, Huang J, Xie Y, Ma S. Deciphering the associations between gene expression and copy number alteration using a sparse double Laplacian shrinkage approach. *Bioinformatics*. 2015;31(24):3977–3983. [PubMed: 26342102]
16. Zhang CH. Nearly unbiased variable selection under minimax concave penalty. *Ann Stat*. 2010;38(2):894–942.
17. Yin J, Li H. Adjusting for high-dimensional covariates in sparse precision matrix estimation by ℓ_1 -penalization. *J Multivar Anal*. 2013;116:365–381. [PubMed: 23687392]
18. Fan J, Fan Y, Lv J. High dimensional covariance matrix estimation using a factor model. *J Econ*. 2008;147(1):186–197.
19. Fan J, Liao Y, Mincheva M. High dimensional covariance matrix estimation in approximate factor models. *Ann Stat*. 2011;39(6):3320–3356. [PubMed: 22661790]
20. Cheng MY, Honda T, Li J. Efficient estimation in semivarying coefficient models for longitudinal/clustered data. *Ann Stat*. 2016;44(5):1988–2017.
21. Jiang Y, He Y, Zhang H. Variable selection with prior information for generalized linear models via the prior lasso method. *J Am Stat Assoc*. 2016;111(513):355–376. [PubMed: 27217599]
22. Fan J, Peng H. Nonconcave penalized likelihood with a diverging number of parameters. *Ann Stat*. 2004;32(3):928–961.
23. Ravikumar P, Wainwright MJ, Raskutti G, Yu B. High-dimensional covariance estimation by minimizing l_1 -penalized log-determinant divergence. *Electron J Stat*. 2011;5:935–980.
24. Breheny P, Huang J. Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *Ann Appl Stat*. 2011;5(1):232–253. [PubMed: 22081779]
25. Li S, Li H, Peng J, Wang P. Bootstrap inference for network construction with an application to a breast cancer microarray study. *Ann Appl Stat*. 2013;7(1):391–417. [PubMed: 24563684]
26. Mohan K, London P, Fazel M, Witten D, Lee SI. Node-based learning of multiple Gaussian graphical models. *J Mach Learn Res*. 2014;15(1):445–488. [PubMed: 25309137]
27. Ciriello G, Gatza ML, Beck AH, et al. Comprehensive molecular portraits of invasive lobular breast cancer. *Cell*. 2015;163(2):506–519. [PubMed: 26451490]
28. Box GEP, Cox DR. An analysis of transformations. *J R Stat Soc Ser B Stat Methodol*. 1964;26(2):211–252.
29. Khare K, Oh SY, Rajaratnam B. A convex pseudolikelihood framework for high dimensional partial correlation estimation with convergence guarantees. *J R Stat Soc Ser B Stat Methodol*. 2015;77(4):803–825.
30. Bickel PJ, Levina E. Regularized estimation of large covariance matrices. *Ann Stat*. 2008;36:199–227.

FIGURE 1. Simulated network structures: Erdos-Renyi (left), scale-free (middle), and nearest-neighbor (right) [Colour figure can be viewed at wileyonlinelibrary.com]

FIGURE 2.

Simulation I with $p = 100$ and a block B . Red curves: AGM with $\alpha=0.4$; Blue curves: AGM with 0.6; Black curves: GM. Left/middle/right column: Erdos-Renyi/scale-free/nearest-neighbor network. Row 1: ROC curves; Row 2: precision-recall curves; Row 3: sum of squared errors versus NUM (the number of edges); Row 4: dKL versus NUM. AGM, assisted graphical model; GM, graphical model; PPV, positive predictive value; ROC, receiver operating characteristic; TPR, true positive rate [Colour figure can be viewed at wileyonlinelibrary.com]

FIGURE 3.

Simulation I with $p = 200$ and a block B . Red curves: AGM with $\alpha=0.4$; Blue curves: AGM with 0.6 ; Black curves: GM. Left/middle/right column: Erdos-Renyi/scale-free/nearest-neighbor network. Row 1: ROC curves; Row 2: precision-recall curves; Row 3: sum of squared errors versus NUM (the number of edges); Row 4: dKL versus NUM. AGM, assisted graphical model; GM, graphical model; PPV, positive predictive value; ROC, receiver operating characteristic; TPR, true positive rate [Colour figure can be viewed at wileyonlinelibrary.com]

FIGURE 4.

Simulation I with $p = 100$ and a banded B . Red curves: AGM with $\alpha=0.4$; Blue curves: AGM with 0.6 ; Black curves: GM. Left/middle/right column: Erdos-Renyi/scale-free/nearest-neighbor network. Row 1: ROC curves; Row 2: precision-recall curves; Row 3: sum of squared errors versus NUM (the number of edges); Row 4: dKL versus NUM. AGM, assisted graphical model; GM, graphical model; PPV, positive predictive value; ROC, receiver operating characteristic; TPR, true positive rate [Colour figure can be viewed at wileyonlinelibrary.com]

FIGURE 5.

Simulation I with $p = 200$ and a banded B . Red curves: AGM with $\alpha=0.4$; Blue curves: AGM with 0.6 ; Black curves: GM. Left/middle/right column: Erdos-Renyi/scale-free/nearest-neighbor network. Row 1: ROC curves; Row 2: precision-recall curves; Row 3: sum of squared errors versus NUM (the number of edges); Row 4: dKL versus NUM. AGM, assisted graphical model; GM, graphical model; PPV, positive predictive value; ROC, receiver operating characteristic; TPR, true positive rate [Colour figure can be viewed at wileyonlinelibrary.com]

FIGURE 6.

Data analysis. Top left panel: AGM; Top right panel: GM; Bottom left panel: difference between AGM and GM; Bottom middle: difference between moderate connections; Bottom right: difference between strong connections. AGM, assisted graphical model; GM, graphical model [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 1

Simulation I: summary statistics on the models selected using cross validation

		$p = 100$				$p = 200$			
		TPR	FPR	ER	dKL	TPR	FPR	ER	dKL
Block B									
Erdos-Renyi network									
GM	0.73	0.15	18.01	106.11	0.22	0.04	125.08	211.99	
	(0.03)	(0.01)	(0.84)	(0.25)	(0.02)	(0.01)	(1.61)	(0.21)	
AGM	0.81	0.17	15.46	105.28	0.34	0.07	116.59	211.17	
	(0.02)	(0.01)	(0.67)	(0.23)	(0.02)	(0.01)	(1.63)	(0.22)	
Scale-free network									
GM	0.57	0.12	16.75	106.11	0.53	0.07	35.63	213.72	
	(0.03)	(0.01)	(0.53)	(0.25)	(0.02)	(0.00)	(0.86)	(0.43)	
AGM	0.67	0.15	14.89	105.49	0.60	0.08	32.86	212.26	
	(0.03)	(0.02)	(0.63)	(0.23)	(0.02)	(0.01)	(0.73)	(0.36)	
Nearest-neighbor network									
GM	0.78	0.12	16.23	105.39	0.70	0.07	36.07	212.50	
	(0.03)	(0.01)	(0.56)	(0.20)	(0.02)	(0.00)	(0.85)	(0.33)	
AGM	0.83	0.14	14.41	104.79	0.77	0.08	32.29	211.15	
	(0.02)	(0.01)	(0.42)	(0.16)	(0.02)	(0.01)	(0.88)	(0.28)	
Banded B									
Erdos-Renyi network									
GM	0.74	0.14	17.83	106.04	0.22	0.04	125.05	212.04	
	(0.03)	(0.01)	(0.79)	(0.23)	(0.02)	(0.01)	(1.41)	(0.21)	
AGM	0.85	0.17	12.68	104.39	0.43	0.08	106.66	210.25	
	(0.02)	(0.01)	(0.69)	(0.18)	(0.02)	(0.01)	(1.82)	(0.22)	
Scale-free network									
GM	0.58	0.13	16.70	106.14	0.53	0.07	35.68	213.69	
	(0.03)	(0.01)	(0.51)	(0.25)	(0.02)	(0.00)	(0.68)	(0.43)	
AGM	0.71	0.16	13.26	104.66	0.63	0.08	29.95	210.39	
	(0.03)	(0.01)	(0.55)	(0.20)	(0.01)	(0.01)	(0.56)	(0.31)	
Nearest-neighbor network									
GM	0.78	0.12	16.21	105.37	0.70	0.07	36.26	212.55	
	(0.02)	(0.01)	(0.61)	(0.20)	(0.02)	(0.00)	(0.82)	(0.31)	
AGM	0.87	0.14	11.92	103.94	0.81	0.08	27.36	209.24	
	(0.02)	(0.01)	(0.48)	(0.17)	(0.01)	(0.00)	(0.70)	(0.22)	

Abbreviations: AGM, assisted graphical model; FPR, false positive rate; GM, graphical model; TPR, true positive rate.

TABLE 2

Simulation II: summary statistics on the models selected using cross validation

r_j		Erdos-Renyi Network				Scale-Free Network				Nearest-Neighbor Network			
		TPR	FPR	ER	dKL	TPR	FPR	ER	dKL	TPR	FPR	ER	dKL
$\alpha(0.1, 0.2)$	GM	0.75	0.15	16.54	106.14	0.62	0.13	15.75	106.05	0.76	0.13	15.19	105.48
		(0.02)	(0.01)	(0.67)	(0.25)	(0.03)	(0.01)	(0.65)	(0.22)	(0.03)	(0.01)	(0.73)	(0.25)
	AGM	0.78	0.16	15.57	105.87	0.64	0.14	15.20	105.82	0.79	0.13	14.37	105.23
		(0.02)	(0.01)	(0.64)	(0.26)	(0.04)	(0.01)	(0.64)	(0.22)	(0.03)	(0.01)	(0.73)	(0.26)
$\alpha(0.5, 0.6)$	GM	0.76	0.15	16.54	106.21	0.61	0.13	15.87	106.07	0.77	0.13	15.13	105.44
		(0.02)	(0.01)	(0.76)	(0.27)	(0.03)	(0.01)	(0.54)	(0.23)	(0.02)	(0.01)	(0.62)	(0.23)
	AGM	0.85	0.18	13.81	105.20	0.71	0.17	13.61	105.19	0.84	0.15	13.05	104.63
		(0.02)	(0.02)	(0.66)	(0.22)	(0.03)	(0.02)	(0.48)	(0.20)	(0.02)	(0.02)	(0.56)	(0.18)
$\alpha(0.8, 0.9)$	GM	0.76	0.15	16.61	106.21	0.62	0.13	15.83	106.09	0.75	0.13	15.37	105.57
		(0.02)	(0.01)	(0.73)	(0.27)	(0.04)	(0.01)	(0.66)	(0.24)	(0.03)	(0.01)	(0.66)	(0.26)
	AGM	0.81	0.16	14.57	105.50	0.67	0.14	14.34	105.48	0.80	0.13	13.60	104.97
		(0.03)	(0.01)	(0.75)	(0.22)	(0.03)	(0.01)	(0.62)	(0.24)	(0.02)	(0.01)	(0.52)	(0.23)

Abbreviations: AGM, assisted graphical model; FPR, false positive rate; GM, graphical model; TPR, true positive rate.