Electronic Journal of Statistics
Vol. 14 (2020) 1269–1301
ISSN: 1935-7524
https://doi.org/10.1214/20-EJS1691

Differential network inference via the fused D-trace loss with cross variables

Yichong Wu and Tiejun Li

LMAM and School of Mathematical Sciences,
Peking University,
Beijing 100871, China
e-mail: wuyichongyt@pku.edu.cn; tieli@pku.edu.cn

Xiaoping Liu

School of Mathematics and Statistics,
Shandong University,
Weihai 264209, China
e-mail: xpliu@sdu.edu.cn

Luonan Chen

Key Laboratory of Systems Biology,
CAS Center for Excellence in Molecular Cell Science,
Institute of Biochemistry and Cell Biology,
Chinese Academy of Sciences,
Shanghai 200031, China
CAS Center for Excellence in Animal Evolution and Genetics,
Chinese Academy of Sciences, Kunming 650223, China
e-mail: lnchen@sibs.ac.cn

Abstract: Detecting the change of biological interaction networks is of great importance in biological and medical research. We proposed a simple loss function, named as CrossFDTL, to identify the network change or differential network by estimating the difference between two precision matrices under Gaussian assumption. The CrossFDTL is a natural fusion of the D-trace loss for the considered two networks by imposing the ℓ_1 penalty to the differential matrix to ensure sparsity. The key point of our method is to utilize the cross variables, which correspond to the sum and difference of two precision matrices instead of using their original forms. Moreover, we developed an efficient minimization algorithm for the proposed loss function and further rigorously proved its convergence. Numerical results showed that our method outperforms the existing methods in both accuracy and convergence speed for the simulated and real data.

MSC 2010 subject classifications: 62P10, 92B15, 65K10.
Keywords and phrases: Differential network, Gaussian assumption, fused D-trace loss, cross variables, coordinate descent.

Received July 2019.

Contents

1. Introduction

Network inference based on the observed biological data is a fundamental topic in network biology along with the rapid developments of high-throughput technologies. A typical approach in gene regulatory network inference is to utilize the Gaussian graphical model [18]. In such a model, gene expression levels are assumed to be a p-dimensional random vector $x \sim \mathcal{N}(0, \Sigma)$. Then, two genes i and j are conditionally independent given the other components if and only if the corresponding entry of the inverse correlation matrix, i.e. precision matrix, $\Sigma^{-1}_{ij} = 0$. Therefore, under the Gaussian assumption, the network inference problem is equivalent to determinng the sparsity pattern of the precision matrix, which is consistent with the covariance selection problem [4].

There has been some proposals on solving the covariance selection problem. Meinshausen and Bühlmann [19] proposed a neighborhood selection scheme in which one can estimate the support of precision matrix row by row. Yuan and Lin [30] proposed the ℓ_1 penalized log-likelihood estimator and used the MaxDet algorithm to solve it. The ADMM algorithm is utilized by Scheinberg, Ma and Goldfarb [22] to solve the ℓ_1 penalized log-likelihood maximization problem. Cai, Liu and Luo [2] proposed a constrained ℓ_1 minimization estimator and established its convergence rates under the elementwise ℓ_∞ norm and Frobenius norm. Hsieh et al. [10, 11] utilized the coordinate descent method to compute the ℓ_1 penalized log-likelihood estimator which shows good efficiency. Zhang and Zou [33] proposed the D-trace loss function whose minimizer is also a precision matrix but with a simpler mathematical formulation.

In practical biological applications, many other indices are proposed to detect the direct associations by considering the non-Gaussian effect [23]. Such attempts include the Pearson correlation coefficient [5], partial correlation [26], mutual information [17], conditional mutual information [34], partial association [24] and so on. Two similar yet more general approaches than finding the precision matrix, the deconvolution [7] and silencing methods [1], are also proposed to remove the indirect effects from the whole correlations of the considered variables.

In many cases of medical research, what we are more interested in is not a particular network but how the network changes from one state to the other state, i.e. the differential network between two states X and Y. Borrowing the idea of Gaussian assumption in network inference, we suppose $x \sim \mathcal{N}(0, \Sigma_X), y \sim \mathcal{N}(0, \Sigma_Y)$. The differential network can be defined as $\Delta = \Sigma_Y^{-1} - \Sigma_X^{-1}$ or $C_Y^{-1} - C_X^{-1}$, where C_X and C_Y are the responding correlation matrices. To estimate the differential network, we can compute Σ_Y^{-1} and Σ_X^{-1} separately and then take difference. However, this naive approach cannot take advantage of the sparsity of Δ, which is a usual case in practice, and the available data for X and Y simultaneously. Danaher, Wang and Witten [3] proposed the joint graphical lasso (JGL) model, which can jointly estimate $\Sigma_Y^{-1}, \Sigma_X^{-1}$ and Δ. However, there is no theoretical results to guarantee the consistency and convergence of their algorithm. Zhao, Cai and Li [35] extended the work for precision matrix estimation by Cai, Liu and Luo [2] to differential network analysis. But the computational complexity and memory requirement of their method are both around $O(p^4)$, where p is the size of matrix Δ. Yuan et al. [31] proposed the D-trace loss function for differential network to directly estimate Δ with lasso penalty. However, their computation time is usually in the order of hours or days even when $p \sim O(10^3)$.

Here, we proposed a new fused D-trace loss function to estimate Δ in this paper. It can be simply viewed as a fusion of the loss in Zhang and Zou [33], i.e. the sum of two D-trace loss functions for the networks X and Y. However, the key novelty of our method is to utilize a transformed formulation of the loss through cross variables, which correspond to the sum and difference of two precision matrices instead of using their original form. We call it CrossFDTL formulation for the differential network inference. As we will see, the CrossFDTL form permits the construction of an efficient optimization algorithm and rigorous proof of its convergence, which is not feasible for the original form. Simulation studies and real data analysis show that our method outperforms the existing methods in both accuracy and convergence speed. Especially, the final algorithm can identify the differential network in tens of minutes, depending on the sparsity of Δ, when the matrix size $p \sim O(10^4)$ or more.

The rest of this paper is organized as follows. In Section 2, we present our model and optimization algorithm. The convergence and consistency are studied in Section 3. In Sections 4 and 5, we summarize our numerical results for the simulation and real data, respectively. Further discussions on possible issues of the differential network inference are left in Section 6.

2. Methods

In this section, we will mainly present our CrossFDTL loss function and the optimization algorithm. The implementation details are also summarized.

2.1. Fused D-trace loss in cross-variable formulation

To estimate $\Delta = \Sigma_Y^{-1} - \Sigma_X^{-1}$, we construct the following CrossFDTL loss function

$$L_F(S, \Delta, \Sigma_X, \Sigma_Y) = \frac{1}{2}\mathrm{tr}((S+\Delta)^2\Sigma_Y) + \frac{1}{2}\mathrm{tr}((S-\Delta)^2\Sigma_X) - 2\mathrm{tr}(S). \quad (1)$$

Straightforward derivation shows that L_F has the unique minimizer

$$S^* = \frac{1}{2}(\Sigma_Y^{-1} + \Sigma_X^{-1}), \quad \Delta^* = \frac{1}{2}(\Sigma_Y^{-1} - \Sigma_X^{-1}) \quad (2)$$

when Σ_X and Σ_Y are invertible.

L_F can be understood as the fusion of D-trace loss functions L_X, L_Y for networks X and Y

$$L_X(\Theta_X, \Sigma_X) = \frac{1}{2}\mathrm{tr}(\Theta_X^2\Sigma_X) - \mathrm{tr}(\Theta_X), \quad L_Y(\Theta_Y, \Sigma_Y) = \frac{1}{2}\mathrm{tr}(\Theta_Y^2\Sigma_Y) - \mathrm{tr}(\Theta_Y) \quad (3)$$

but with the cross variables $S := \frac{1}{2}(\Theta_Y + \Theta_X)$ and $\Delta := \frac{1}{2}(\Theta_Y - \Theta_X)$. We use the terminology *cross variables* for S and Δ versus the original variables Θ_X and Θ_Y by drawing the analogy between them and the cross diagonals versus adjacent sides in a parallelogram. Note that Σ_X and Σ_Y almost have symmetric status in CrossFDTL formulation except the sign of Δ. We will show that this formulation is essential for the construction of efficient optimization algorithm and rigorous proof of the convergence.

In real computations, the sample covariance matrices $\hat{\Sigma}_X$ and $\hat{\Sigma}_Y$ may not be invertible, so we can estimate Δ by considering the following minimization problem with suitable penalties for S and Δ

$$\min_{\Delta=\Delta^T, S=S^T} L(S, \Delta) = L_F(S, \Delta, \hat{\Sigma}_X, \hat{\Sigma}_Y) + \lambda\|\Delta\|_1 + \rho/2(\|S\|_F^2 + \|\Delta\|_F^2), \quad (4)$$

where $\lambda > 0, \rho \geq 0$ are tuning parameters. The ℓ_1 penalty for Δ ensures sparsity, while the ℓ_2 penalty for both S and Δ ensures strict convexity of the loss function and boundedness of the minimization. When the sample sizes n_X and n_Y are smaller than p, we should choose ρ to be positive. We will denote the minimizer of (4) by $(\hat{S}, \hat{\Delta})$.

We remark that in Eq. (4) we interpret the matrix S as an auxiliary variable, thus we do not require the positivity of S in the minimization. We can also reformulate (4) in an S-free form

$$\min_{\Delta=\Delta^T} \tilde{L}(\Delta) = L(S(\Delta), \Delta) = L_F(S(\Delta), \Delta, \hat{\Sigma}_X, \hat{\Sigma}_Y) + \lambda\|\Delta\|_1 + \rho/2\|\Delta\|_F^2, \quad (5)$$

where $S(\Delta) = \arg\min_{S=S^{\mathrm{T}}} L(S, \Delta)$ which has the explicit solution (9) as shown in next subsection. However, we will keep the form (4) with S since it is more convenient for numerics.

2.2. Minimization algorithm

We minimize (4) by alternating optimization method. Specifically, given $S^{(k)}, \Delta^{(k)}$ at the kth step, we update the estimates as the following

$$S^{(k+1)} = \arg\min_{S=S^{\mathrm{T}}} L(S, \Delta^{(k)}), \tag{6}$$

$$\Delta^{(k+1)} = \arg\min_{\Delta=\Delta^{\mathrm{T}}} L(S^{(k+1)}, \Delta). \tag{7}$$

For (6), we have

$$S^{(k+1)} = \arg\min_{S=S^{\mathrm{T}}} \left[\frac{1}{2}\mathrm{tr}(S^2(\hat{\Sigma}_X + \hat{\Sigma}_Y + \rho I)) - \frac{1}{2}\mathrm{tr}\left(S(4I + \Delta^{(k)}(\hat{\Sigma}_X - \hat{\Sigma}_Y) + (\hat{\Sigma}_X - \hat{\Sigma}_Y)\Delta^{(k)})\right)\right].$$

Let $G(A, B)$ denote the solution of the optimization problem

$$\min_{S=S^{\mathrm{T}}} \frac{1}{2}\mathrm{tr}(S^2 A) - \mathrm{tr}(SB), \quad A \succ 0, B = B^{\mathrm{T}}. \tag{8}$$

We have the explicit form

$$S = G(A, B) = U_A \left((U_A^{\mathrm{T}} B U_A) \circ C \right) U_A^{\mathrm{T}}, \tag{9}$$

where $\circ$ denotes the Hadamard product of matrices, $A = U_A \Sigma_A U_A^{\mathrm{T}}$ is the eigenvalue decomposition of A with ordered eigenvalues $\sigma_1 \geq \cdots \geq \sigma_p$, and $C_{ij} = 2/(\sigma_i + \sigma_j)$. Thus we obtain

$$S^{(k+1)} = G\left(\hat{\Sigma}_X + \hat{\Sigma}_Y + \rho I, 2I + \frac{1}{2}\Delta^{(k)}(\hat{\Sigma}_X - \hat{\Sigma}_Y) + \frac{1}{2}(\hat{\Sigma}_X - \hat{\Sigma}_Y)\Delta^{(k)}\right). \tag{10}$$

To update $\Delta^{(k+1)}$, we have

$$\Delta^{(k+1)} = \arg\min_{\Delta=\Delta^{\mathrm{T}}} L_Q(\Delta) := Q(\Delta) + \lambda\|\Delta\|_1 \tag{11}$$

where

$$Q(\Delta) := \frac{1}{2}\mathrm{tr}(\Delta^2 A) + \mathrm{tr}(\Delta B), \tag{12}$$

and $A = \hat{\Sigma}_X + \hat{\Sigma}_Y + \rho I$, $B = \frac{1}{2}S^{(k+1)}(\hat{\Sigma}_Y - \hat{\Sigma}_X) + \frac{1}{2}(\hat{\Sigma}_Y - \hat{\Sigma}_X)S^{(k+1)}$. We use the coordinate descent method [8, 10, 11, 28, 32] to solve (11).

Consider the coordinate descent update for the variable Δ_{ij} with $i < j$ that preserves symmetry: $\tilde{\Delta} = \Delta + \mu(e_i e_j^{\mathrm{T}} + e_j e_i^{\mathrm{T}})$. We need to solve the following single variable optimization problem

$$\mu_{ij} = \arg\min_{\mu} \frac{1}{2}\mathrm{tr}(\tilde{\Delta}^2 A) + \mathrm{tr}(\tilde{\Delta}B) + \lambda\|\tilde{\Delta}\|_1. \tag{13}$$

Expanding the terms in (13), we get

$$\mu_{ij} = \arg\min_{\mu} \frac{1}{2}(A_{ii} + A_{jj})\mu^2 + [(A\Delta)_{ij} + (A\Delta)_{ji} + B_{ij} + B_{ji}]\,\mu + 2\lambda|\Delta_{ij} + \mu|. \tag{14}$$

Let $a = A_{ii} + A_{jj}$, $b = (A\Delta)_{ij} + (A\Delta)_{ji} + B_{ij} + B_{ji}$, and $c = \Delta_{ij}$. Then the minimum is achieved when

$$\mu_{ij} = -c + \mathcal{S}(c - b/a, 2\lambda/a) \tag{15}$$

where

$$\mathcal{S}(z, r) = \operatorname{sgn}(z)\max\{|z| - r, 0\} \tag{16}$$

is the soft-thresholding function. Since a and c are easy to compute, the main cost lies when evaluating b. Thanks to the sparsity of Δ, $A\Delta$ can be obtained with a relatively small cost in each update. Specifically, it needs $O(fp)$ flops, where $f = \#\{(k,l)|\Delta_{kl} \neq 0\}$ is the number of nonzero elements in Δ.

In the coordinate descent step, we only update a subset of the variables of Δ which we call the *free* set. We identify these variables based on the value of the gradient. The *free* set S_{free} and the *fixed* set S_{fixed} are defined as:

$$\begin{aligned} &\Delta_{ij} \in S_{\text{fixed}} \quad \text{if } \left|\frac{\partial Q(\Delta)}{\partial \Delta_{ij}}\right| \leq \lambda \text{ and } \Delta_{ij} = 0; \\ &\Delta_{ij} \in S_{\text{free}} \quad \text{otherwise.} \end{aligned} \tag{17}$$

Actually, the coordinate descent update restricted to the component $\Delta_{ij} \in S_{\text{fixed}}$ would not change its value due to the following Proposition 1 [10]. Here we restate it for its simplicity.

Proposition 1. *For any Δ and corresponding fixed and free sets S_{fixed} and S_{free} as defined by (17), $\delta = 0$ is the solution of the following minimization problem:*

$$\min_{\delta = \delta^T} L_Q(\Delta + \delta) \quad \textit{with constraints } \delta_{kl} = 0 \quad \textit{for } (k,l) \in S_{free}. \tag{18}$$

Proof. Any optimal solution δ for (18) must satisfy

$$0 \in \frac{\partial L_Q(\Delta + \delta)}{\partial \delta_{ij}}, \quad (i,j) \in S_{\text{fixed}} \text{ and } \delta_{kl} = 0 \text{ for } (k,l) \in S_{\text{free}}.$$

This is equivalent to

$$\frac{\partial Q(\Delta + \delta)}{\partial \delta_{ij}} \begin{cases} = -\lambda, & \text{if } \Delta_{ij} > 0, \\ = \lambda, & \text{if } \Delta_{ij} < 0, \\ \in [-\lambda, \lambda], & \text{if } \Delta_{ij} = 0. \end{cases} \tag{19}$$

for $(i,j) \in S_{\text{fixed}}$. The definition of S_{fixed} in (17) ensures (19). □

Based on the above proposition, we perform the inner loop coordinate descent updates only restricted to the free set. Therefore, the number of variables over

which we perform the coordinate descent step can be potentially reduced from p^2 to the number of nonzero elements in $\Delta^{(k)}$. All of these settings guarantee that we can save huge computational cost when the solution is sparse. We finally remark that the D-trace loss function considered in Yuan et al. [31] is not suitable for coordinate descent method since the sparsity is not preserved during the updates.

2.3. Implementation details

The choice of penalty parameters is always an issue in practice. For the ℓ_1 penalty parameter λ, some information criteria such as the Akaike information criterion (AIC) or Bayesian information criterion (BIC) are usually suggested [31]. However, for a general loss function, the corresponding likelihood is difficult to get and one has to consider different surrogates of AIC or BIC, which may not give good enough results. Therefore in our computations, we mainly choose λ by experience. The ℓ_2 penalty, which is also called ridge penalty in literature, can be considered as a complement to the ℓ_1 penalty term. In regression theory, such combination of the lasso and ridge penalty is also called elastic net penalty [36]. For the choice of ℓ_2 penalty parameter ρ, it depends on the sample size n_x, n_y and the matrix size p. The less n_x, n_y is than p, the larger ρ should be chosen. In the case that n_x and n_y are far larger than p, ρ can be set as 0. In our practice, the final results are not sensitive to the choice of ρ.

In the coordinate descent step, we first find the descent direction D where $D_{ij} = \mu_{ij}$ for $(i,j) \in S_{\text{free}}$ and $D_{ij} = 0$ for $(i,j) \in S_{\text{fixed}}$. Then we adopt the Armijo rule and try step-size $\alpha \in \{\beta^0, \beta^1, \beta^2, \cdots\}$ with a constant decrease rate $0 < \beta < 1$ (typically $\beta = 0.5$) until we find the smallest $k \in \mathbb{N}$ such that $\tilde{\Delta} = \Delta + \alpha D$ with $\alpha = \beta^k$ satisfies the following sufficient decrease condition:

$$L_Q(\Delta + \alpha D) \leq L_Q(\Delta) + \alpha\sigma\delta, \quad \delta = \operatorname{tr}(G \cdot D) + \|\Delta + D\|_1 - \|\Delta\|_1,$$

where $0 < \sigma < 0.5$, and G denotes the gradient matrix of G with respect to Δ with components $G_{ij} = \partial_{\Delta_{ij}} Q$. We terminate the coordinate descent inner loop when $|L_Q(\tilde{\Delta}) - L_Q(\Delta)| \leq \varepsilon |L_Q(\Delta)|$, where $\varepsilon = 10^{-3}$ is taken in our computation.

Now we summarize the overall algorithm in Algorithm 1.

We take the following termination criterion for the overall alternating minimization algorithm in our numerical experiments

$$\|\Delta^{(k+1)} - \Delta^{(k)}\|_F < 10^{-3} \max(1, \|\Delta^{(k)}\|_F, \|\Delta^{(k+1)}\|_F).$$

In practical computations, we will limit the upper bound of the iteration number in the inner loop of Step 2(b). As we mentioned above, the coordinate descent step needs only $O(fp)$ flops, where f is the number of nonzero elements. So Step 2(b) will not cost too much when the solution is sparse. We speculate that the efficiency of coordinate descent method is due that it makes the updates in a strongly targeted way to the selected elements. The simulation studies in Section 4 verify this point.

Algorithm 1 Alternating minimization algorithm for the CrossFDTL formulation.

1. Initialization: $S^0 = 0, \Delta^0 = 0$.
2. Given $S^{(k)}, \Delta^{(k)}$ at the kth step, make the following updates at the $(k+1)$th step:
   (a). $S^{(k+1)} = G(\hat{\Sigma}_X + \hat{\Sigma}_Y + \rho I, 2I + \frac{1}{2}\Delta^{(k)}(\hat{\Sigma}_X - \hat{\Sigma}_Y) + \frac{1}{2}(\hat{\Sigma}_X - \hat{\Sigma}_Y)\Delta^{(k)})$.
   (b). Apply the following coordinate descent to get $\Delta^{(k+1)}$ as below:
   Initialization: $\Delta = \Delta^{(k)}$.
   Repeat the inner loops (1)-(3) until convergence:
   (1). Partition the variables into free and fixed sets based on (17).
   (2). Use coordinate descent to find the optimizing direction D of (11) over the set of free variables.
   (3). Use an Armijo-rule based step-size selection to get α such that there is a sufficient decrease in the objective function.
   Output the final result as $\Delta^{(k+1)}$.
3. Repeat (a)-(b) until the convergence criterion is satisfied.
4. Output $\Delta^{(k+1)}$ as the estimator of the differential matrix Δ.

3. Theoretical results

3.1. Convergence of the alternating minimization

We will prove the convergence of the iterations (6)-(7) in this subsection. Let us first state a simple yet important lemma.

Lemma 1. *Let $F(x,y) = f(x,y) + \|g(x,y)\|_1$, where f and g are differentiable convex functions of (x,y). If g has the separable form, i.e. $g(x,y) = (g_1(x), g_2(y))$, where $g_1(x)$ and $g_2(y)$ are differentiable convex functions of x and y, then we have*

$$0 \in \frac{\partial F}{\partial x}\Big|_{(x^*,y^*)}, 0 \in \frac{\partial F}{\partial y}\Big|_{(x^*,y^*)} \quad \textit{implies} \quad 0 \in \partial F\Big|_{(x^*,y^*)}, \tag{20}$$

where $\frac{\partial F}{\partial x}, \frac{\partial F}{\partial y}, \partial F$ means the subgradients of F.

Proof. Since $g(x,y) = (g_1(x), g_2(y))$, then $\|g(x,y)\|_1 = |g_1(x)| + |g_2(y)|$ and

$$\partial\|g(x,y)\|_1 = \partial|g_1(x)| + \partial|g_2(y)|.$$

Therefore

$$0 \in \frac{\partial F}{\partial x}\Big|_{(x^*,y^*)} = \frac{\partial f}{\partial x}\Big|_{(x^*,y^*)} + \frac{\partial|g_1|}{\partial x}\Big|_{x^*},$$
$$0 \in \frac{\partial F}{\partial y}\Big|_{(x^*,y^*)} = \frac{\partial f}{\partial y}\Big|_{(x^*,y^*)} + \frac{\partial|g_2|}{\partial y}\Big|_{y^*}.$$

We have

$$\begin{aligned}\partial F\Big|_{(x^*,y^*)} &= \partial f\Big|_{(x^*,y^*)} + \partial\|g\|_1\Big|_{(x^*,y^*)} \\ &= \left(\frac{\partial f}{\partial x}\Big|_{(x^*,y^*)} + \frac{\partial|g_1|}{\partial x}\Big|_{x^*}, \frac{\partial f}{\partial y}\Big|_{(x^*,y^*)} + \frac{\partial|g_2|}{\partial y}\Big|_{y^*}\right)^{\mathrm{T}} \ni 0. \quad \square\end{aligned}$$

Without the *separable* condition on g in this lemma, the optimization in alternating directions may be trapped at a meaningless point. This point has the property that it is a minimum point in each direction but not an optimum at all. This can be easily checked for the example $F(x,y) = x^2 + y^2 + |2x + 2y|$, where $g(x,y) = 2x+2y$ is not of separable form and the alternating minimization in x and y may be trapped at any $(x,y) = (a,-a)$ with $a \in [-1,1]$. But the global minimum is at $(x,y) = (0,0)$. This means that the convergence is not guaranteed for JGL proposed in Danaher, Wang and Witten [3].

With this lemma, our algorithm has the convergence.

Theorem 1. *For the CrossFDTL minimization problem (4), if $L(S,\Delta)$ is strictly convex with respect to S and Δ, the algorithm (6)-(7) converges to the unique minimum.*

Proof. Denote $z^{(k)} = (S^{(k)}, \Delta^{(k)})$. We have

$$L(S^{(k)}, \Delta^{(k)}) \geq L(S^{(k+1)}, \Delta^{(k)}) \geq L(S^{(k+1)}, \Delta^{(k+1)}) \tag{21}$$

and correspondingly

$$0 \in \frac{\partial L}{\partial S}\Big|_{(S^{(k)},\Delta^{k-1})}, \quad 0 \in \frac{\partial L}{\partial \Delta}\Big|_{(S^{(k)},\Delta^{(k)})}. \tag{22}$$

By the strict convexity of $L(S,\Delta)$, $z^{(k)}$ is bounded. Thus we can find a subsequence $\{z^{(k_l)}\}_{l=1}^{\infty}$ and an accumulation point z^{∞} such that $z^{(k_l)} \to z^{\infty}$.

For any (i,j), if $\Delta_{ij}^{\infty} > 0$ and l is sufficiently big, we have

$$\Delta_{ij}^{(k_l)} > 0, \quad \frac{\partial L}{\partial \Delta_{ij}}\Big|_{(S^{(k_l)},\Delta^{(k_l)})} = \frac{\partial L_1}{\partial \Delta_{ij}}\Big|_{(S^{(k_l)},\Delta^{(k_l)})} + \lambda = 0,$$

where we denote $L(S,\Delta) = L_1(S,\Delta) + \lambda\|\Delta\|_1$. Therefore

$$\frac{\partial L}{\partial \Delta_{ij}}\Big|_{(S^{\infty},\Delta^{\infty})} = \lim_{l\to\infty} \frac{\partial L_1}{\partial \Delta_{ij}}\Big|_{(S^{(k_l)},\Delta^{(k_l)})} + \lambda = 0.$$

The case for $\Delta_{ij}^{\infty} < 0$ is similar. If $\Delta_{ij}^{\infty} = 0$, as a result of (22), we have

$$\frac{\partial L_1}{\partial \Delta_{ij}}\Big|_{(S^{(k)},\Delta^{(k)})} \in [-\lambda,\lambda] \quad \text{thus} \quad \frac{\partial L_1}{\partial \Delta_{ij}}\Big|_{(S^{\infty},\Delta^{\infty})} \in [-\lambda,\lambda].$$

Therefore

$$0 \in \frac{\partial L}{\partial \Delta_{ij}}\bigg|_{(S^\infty,\Delta^\infty)} = \frac{\partial L_1}{\partial \Delta_{ij}}\bigg|_{(S^\infty,\Delta^\infty)} + [-\lambda,\lambda].$$

Combining the above results, we get

$$\Delta^\infty = \arg\min_{\Delta} L(S^\infty, \Delta) \tag{23}$$

and it is the unique minimum element in Δ direction due to the strict convexity of $L(S,\Delta)$.

Next let us consider the subgradient of L with respect to S at z^∞. We will first show that $\Delta^{(k_l-1)} \to \Delta^\infty$. Otherwise we can find a further subsequence, still denoted as $\{\Delta^{(k_l)}\}$, such that

$$\Delta^{(k_l-1)} \to \Delta^\# \neq \Delta^\infty.$$

Then we have

$$L(S^\infty,\Delta^\infty) = \lim_{k\to\infty} L(S^{(k)},\Delta^{(k)}) = \lim_{l\to\infty} L(S^{(k_l)},\Delta^{(k_l-1)}) = L(S^\infty,\Delta^\#)$$

by the squeeze inequalities in (21). This shows that $\Delta^\infty = \Delta^\#$ by (23), which is a contradiction. Therefore $\Delta^{(k_l-1)} \to \Delta^\infty$. Since $L(S,\Delta)$ is differentiable with respect to S, we have

$$\frac{\partial L}{\partial S}\bigg|_{(S^\infty,\Delta^\infty)} = \lim_{l\to\infty} \frac{\partial L}{\partial S}\bigg|_{(S^{(k_l)},\Delta^{(k_l-1)})} = 0.$$

Now we have already proved that

$$0 \in \frac{\partial L}{\partial S}\bigg|_{(S^\infty,\Delta^\infty)}, \quad 0 \in \frac{\partial L}{\partial \Delta}\bigg|_{(S^\infty,\Delta^\infty)}.$$

According to Lemma 1, $0 \in \partial L|_{z^\infty}$ implies that z^∞ is the unique minimum element of the strictly convex function $L(S,\Delta)$. The limit $z^{(k)} \to z^\infty$ is due to the fact that any subsequence limit of $z^{(k)}$ is z^∞. The proof is completed. □

We remark that the convergence rate of $\|\Delta^{(k)} - \hat{\Delta}\|$ with respect to k is not discussed in the above theorem.

3.2. Consistency and rate of convergence

Our ultimate goal is $\Sigma_Y^{-1} - \Sigma_X^{-1}$, especially its structure of nonzero entries. In practice, we only have sample covariance matrices $\hat{\Sigma}_X$ and $\hat{\Sigma}_Y$, and we have to consider the approximability of Δ^* and the minimizer $\hat{\Delta}$ of (4). This amounts to study the consistency and rate of convergence of $\|\hat{\Delta} - \Delta^*\|$ with respect to the sample sizes n_X and n_Y. In this section, we will first present the irrepresentability condition (25) for establishing the consistency of our estimator,

which is based on similar tricks as those done in Ravikumar et al. [21]. Then we give an estimate of the convergence rate. We will suppose $\rho = 0$ all along in this section.

Let $S^+ = \text{supp}(\Delta^*)$ denote the support of Δ^* and S^c the complement of S^+, $s = |S^+|$. For any matrix $M \in \mathbb{R}^{p\times p}$, denote

$$\Gamma(M) = \frac{1}{2}(M \oplus M) = \frac{1}{2}(M \otimes I + I \otimes M), \tag{24}$$

where $\otimes$ is the Kronecker product and $\oplus$ is the Kronecker sum. We have that $\Gamma(M)$ is a $p^2 \times p^2$ matrix indexed by vertex pairs and

$$\Gamma(M)_{(j,k)(l,m)} = M_{k,m}\delta(j,l) + M_{j,l}\delta(k,m),$$

where $\delta(j,l) = 1$ if $j = l$ and $\delta(j,l) = 0$ otherwise. For simplicity, we denote

$$\begin{aligned} \Gamma^{*(1)} &= \Gamma(\Sigma_Y + \Sigma_X), \quad \Gamma^{*(2)} = \Gamma(\Sigma_Y - \Sigma_X), \\ \hat{\Gamma}^{(1)} &= \Gamma(\hat{\Sigma}_Y + \hat{\Sigma}_X), \quad \hat{\Gamma}^{(2)} = \Gamma(\hat{\Sigma}_Y - \hat{\Sigma}_X). \end{aligned}$$

Then the irrepresentability condition is

$$\max_{e\in S^c} \|E_e^{*Z}\|_1 < 1, \tag{25}$$

where E_e^{*Z} is defined as

$$\begin{aligned} E_e^{*Z} \quad = \quad & -\Gamma^{*(1)}_{eS^+}\left(\Gamma^{*(1)}_{S^+S^+} - \Gamma^{*(2)}_{S^+.}{\Gamma^{*(1)}}^{-1}\Gamma^{*(2)}_{.S^+}\right)^{-1} \\ & +\Gamma^{*(2)}_{e.}\left(\Gamma^{*(1)} - \Gamma^{*(2)}_{.S^+}{\Gamma^{*(1)}}^{-1}_{S^+S^+}\Gamma^{*(2)}_{S^+.}\right)^{-1}\Gamma^{*(2)}_{.S^+}{\Gamma^{*(1)}}^{-1}_{S^+S^+}. \end{aligned}$$

Here the notation $\Gamma_{S^+.}$ means the sub-matrix formed by extracting the components of Γ with the first index $(j,k) \in S^+$ and the second index (l,m) being arbitrary. The other sub-matrices are defined similarly. The derivation of (25) is shown in the Appendix.

With the irrepresentability condition, we can establish the consistency and the rate of convergence of our estimator based on the assumption that X and Y are subject to sub-Gaussian distribution. A zero-mean random vector $X \in \mathbb{R}^p$ with covariance matrix Σ is said to be sub-Gaussian if there exists a constant $\sigma > 0$ such that

$$\mathbb{E}\left[\exp\left(tX_i(\Sigma_{i,i})^{-1/2}\right)\right] \le \exp(\sigma^2 t^2/2), \quad \forall t \in \mathbb{R}, i = 1,\ldots,p.$$

Here X_i is the ith coordinate of the random vector X.

Theorem 2. *Assume that X and Y are sub-Gaussian with parameters σ_X and σ_Y. Under the irrepresentability condition (25), if*

$$\frac{22}{\alpha}C_\lambda\delta \le \lambda \le \frac{24}{\alpha}C_\lambda\delta, \tag{26}$$

where $\alpha := 1 - \max_{e\in S^c}\|E_e^{*Z}\|_1$ *and* $\delta := \max\{\delta_{f_X}(n_X,p^\eta),\delta_{f_Y}(n_Y,p^\eta)\}$ *for some* $\eta > 2$ *and* $\min(n_X,n_Y) > C_G\bar{\delta}^{-2}(\eta\log p+\log 4)$*, then with probability greater than* $1-2/p^{\eta-2}$*, the support of* $\hat{\Delta}$ *lies in the support of* Δ^* *and*

$$\|\hat{\Delta}-\Delta^*\|_\infty \le M_G\left\{\frac{\eta\log p+\log 4}{\min(n_X,n_Y)}\right\}^{\frac{1}{2}}, \|\hat{\Delta}-\Delta^*\|_F \le M_G\left\{\frac{\eta\log p+\log 4}{\min(n_X,n_Y)}\right\}^{\frac{1}{2}} s^{\frac{1}{2}},$$

where $\|A\|_\infty := \max_{ij}|A_{i,j}|$ *instead of the usual* ℓ_∞ *norm in matrix analysis [25]. Morever, if* $\min(n_X,n_Y) > M_G^2(\eta\log p+\log 4)/(\min_{j,k:\Delta^*_{j,k}\neq 0}|\Delta^*_{j,k}|)^2$*, then the support of* $\hat{\Delta}$ *equals to the support of* Δ^**. Here* $\bar{\delta}, M, C_\lambda, C_G, M_G$ *are constants depending on* $\Sigma_X,\Sigma_Y,\sigma_X,\sigma_Y$*. Their definition and the constants* $\delta_{f_X}(n_X,p^\eta),\delta_{f_Y}(n_Y,p^\eta)$ *are given in the Appendix.*

Although this theorem looks very complicated, its meaning is clear. The lower bound on sample sizes n_X and n_Y limits the error between covariance matrices Σ_X,Σ_Y and sample covariance matrices $\hat{\Sigma}_X,\hat{\Sigma}_Y$ with some probability. The lower bound on λ ensures that the solution of the l_1-penalized optimization problem (4) is sparse enough. On the other hand, the upper bound on λ limits the effect of the regularization term. The estimation (26) on λ is not tight, and it is consistent with the results in [21, 31, 33], which simply take a specific λ in the admissible range. The detailed proof of Theorem 2 is given in the Appendix.

More generally, this theorem can be extended to random vectors under polynomial tail conditions if we ignore the Gaussian set-up and the goal is just to estimate the difference on precision matrices [6, 20, 21]. We will skip the details on this point for its irrelevance.

4. Simulation studies

To show the virtue of our CrossFDTL formulation, in this section, we use the simulation data to compare the performance of our estimator with that by D-trace loss function (DTL) [31].

In the simulation study, the data are generated from two independent multivariate normal distributions $\mathcal{N}(0,\Sigma_X)$ and $\mathcal{N}(0,\Sigma_Y)$.

Model 1: Highly structured differential matrix.

The precision matrices are set as

$$\Sigma_X^{-1}(i,j) = 0.5^{|i-j|}, \qquad \Sigma_Y^{-1}(i,j) = \begin{cases} 0.9, & |i-j| = \lfloor p/4\rfloor \\ 0.5^{|i-j|}, & \text{otherwise.}\end{cases}$$

To ensure the positive definiteness, we added 1.2 to their diagonal elements. It is obvious that the differential matrix Δ is sparse.

TABLE 1
Problems with large size p (The numbers in parentheses mean the percentage of time cost in Step 2(b) of Algorithm 1)

	$p = 2000, n = 500$				$p = 2000, n = 1000$			
	TPR	TNR	TDR	time	TPR	TNR	TDR	time
CrossFDTL	0.9594	0.9363	0.0703	17 s (76)	0.9902	0.9767	0.1762	8 s (30)
DTL	0.9076	0.9115	0.0490	1561 s	0.9888	0.9419	0.0788	1086 s
	$p = 4000, n = 500$				$p = 4000, n = 1000$			
	TPR	TNR	TDR	time	TPR	TNR	TDR	time
CrossFDTL	0.9493	0.9531	0.0483	102 s (63)	0.9826	0.9909	0.2127	45 s (22)
DTL	0.9178	0.9222	0.0287	15942 s	0.9609	0.9866	0.1525	8596 s

Model 2: Randomly chosen differential matrix.

We first generate the random matrix Σ_X^{-1} by $\Sigma_X^{-1} = D^{-\frac{1}{2}}(Z_2)Z_2D^{-\frac{1}{2}}(Z_2)$, where $Z_2 = Z_1Z_1^{\mathrm{T}}$, Z_1 is a random matrix with entries sampled independently from $\mathcal{N}(0,1)$, and $D(Z_2)$ is the diagonal matrix with diagonal elements from Z_2. For the choice of Δ, we first randomly set $5p$ elements of its lower triangular part to nonzero values sampled from symmetric Bernoulli distribution valued in $\{-0.5, 0.5\}$, then do the same change to the upper triangular part to make Δ symmetric. Σ_Y^{-1} was then taken to be $\Sigma_X^{-1} + \Delta$. Finally to ensure that both Σ_X^{-1} and Σ_Y^{-1} are positive definite, we add a constant to their diagonals, which is taken as $0.1 - \min\left\{\lambda(\Sigma_X^{-1}), \lambda(\Sigma_Y^{-1})\right\}$ in our numerical experiments below.

Figure 1 shows the receiver operating characteristic (ROC) curves (Figure 1 (a), (c)) and the precision-recall (PR) curves (Figure 1 (b), (d)) for both models with DTL and CrossFDTL formulations. We take $p = 100, n = 50$ in Model 1 (Figure 1 (a), (b)) and $p = 1000, n = 500$ in Model 2 (Figure 1 (c), (d)), respectively. In the plots, each point corresponds to one choice of the tuning parameter λ and the number beside it shows the time cost in seconds. We only list several of them for clarity. The values *auc1* and *auc2* in each subfigure correspond to the AUC values, i.e. the area under the ROC or PR curves, for the DTL and CrossFDTL methods, respectively. It can be found that the CrossFDTL method outperforms the DTL method in both computational efficiency and accuracy.

To further show the efficiency of the CrossFDTL method, we have computed problems with a larger size p. In Model 2, we take $p = 2000, 4000$, $n = 500, 1000$ with a suitable parameter λ. The comparisons with DTL are shown in Table 1.

We can see that the CrossFDTL not only gives more accurate results (the true positive rate (TPR), true negative rate (TNR), and true discovery rate (TDR, i.e., the precision) are all higher than those obtained by DTL), but also cost less time. In the above examples, the DTL costs about 100 times or even longer computation time than CrossFDTL. Even with early stopping, the DTL still needs at least 10 times longer time than CrossFDTL to achieve similar accuracy in our numerical experiments. In this sense, the CrossFDTL can be utilized to solve quite large scale problems.

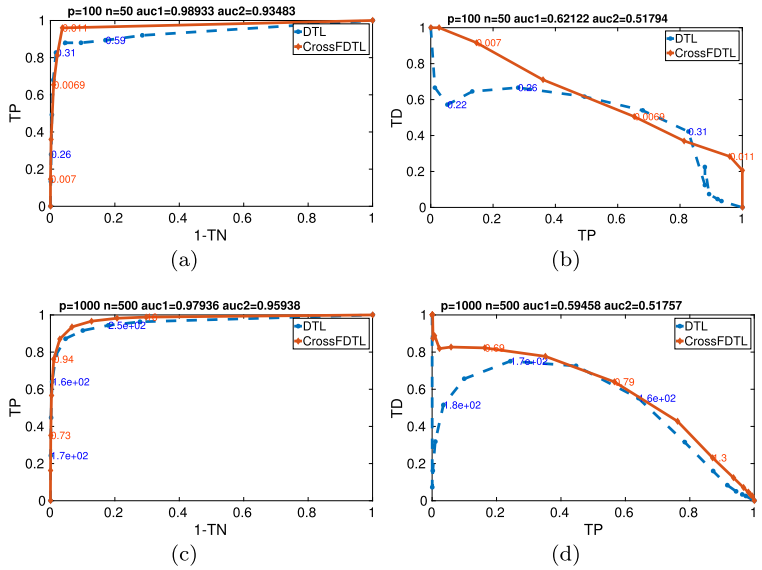

FIG 1. *The left panels (a)-(c) show the receiver operating characteristic curves for the support recovery of* Δ^**, with (a) for Model 1 and (c) for Model 2. The right panels (b)-(d) show the precision-recall curves for the support recovery of* Δ^**, with (b) for Model 1 and (d) for Model 2. The matrix size* p *and sample size* n *are listed in the subtitles, and the values auc*1 *and auc*2 *give the area under the curves corresponding to CrossFDTL and DTL. The solid and dashed lines correspond to CrossFDTL and DTL, respectively.*

In the simulation studies listed in Table 1, we also show the percentage of time cost in Step 2(b) of Algorithm 1 when we utilize CrossFDTL. We can observe that the higher TNR is, the lower percentage it costs. It is reasonable since higher TNR cases will have lower number of nonzero elements, thus Step 2(b) costs less. Furthermore, our empirical computations show that the inner loop of Step 2(b) also converges fast. In each iteration of Step 2(b), there are no more than 5 iterations in the inner loop although we set an upper bound of 50.

5. Real data analysis

In this section, we apply our CrossFDTL method to the gene expression data for gastric cancer patients and make comparisons with DTL.

The gene expression profiles for gastric cancer are obtained from GSE27342 dataset of GEO database (https://www.ncbi.nlm.nih.gov/geo/). The dataset contains 160 samples from cancer tissues and the adjacent non-cancerous tissues of 80 gastric cancer patients. We attempt to find the difference between disease and normal gene networks. The pathway information we used was the *Pathways in cancer* available in the KEGG pathway database. 409 genes in this pathway known to play important roles in cancers. We use only genes that the

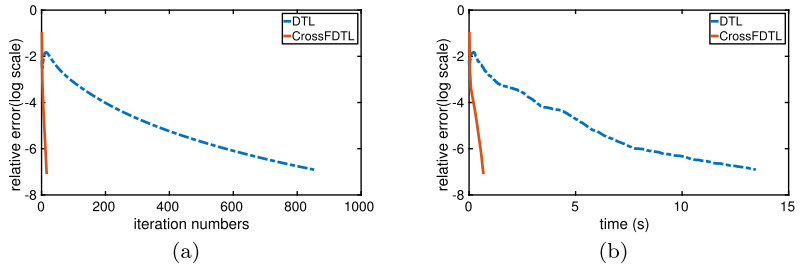

FIG 2. *The convergence history of CrossFDTL and DTL in terms of iteration steps and computation time. It can be observed that the iteration numbers of CrossFDTL (the overall iterations in Algorithm 1) is much less than that of DTL, and the CrossFDTL costs much less time correspondingly.*

gene expression data are complete enough (more than 80% samples had this gene expression data). This left 358 genes to be tested further.

For the consistency of scales, we use the differences between the inverse correlation matrices to represent the differential network. The correlation matrix can be considered as the normalized version of covariance matrix, therefore it can eliminate the influence of the units or scales of the data for different genes. In Figure 3, we showed the top 10 genes according to their importance in the differential gene regulatory networks inferred by the two methods. Here the importance of a gene is measured by the sum of the strength of the edges linked to this gene in the differential network, i.e., the importance of gene i is defined as

$$I(i) = \sum_{j \neq i} |\hat{\Delta}_{ij}|,$$

where $\hat{\Delta}$ is the inferred differential matrix. We can see that similar results are obtained by CrossFDTL and DTL under Gaussian assumptions in the sense that most of genes in the two differential networks were common.

The identified important genes in the differential network have distinct biological meaning. The ERBB2 (also called HER2) identified in both CrossFDTL and DTL is an important cancer causal gene in gastric cancer. FGFR2 is a member of the fibroblast growth factor receptor family, and reported association with gastric cancer [12, 29]. AXIN2 indentified from DTL plays an important role in the regulation in the Wnt signaling pathway, and is reported to associate with breast cancer [14, 15] and colorectal cancer [16, 9], but there is no report about the association between AXIN2 and gastric cancer. The MET is also an important gene in gastric cancer [13, 27] identified in CrossFDTL while not in DTL.

To further confirm the obtained results, we use the KEGG gastric cancer pathway to see whether these genes have been previously reported (Fig. 4). The biggest circle contains 358 genes in the pathways of all types of cancers. The medium-sized circle contains 107 genes in the gastric pathway. And the smallest circle contains the top 10 genes related to gastric cancer identified by

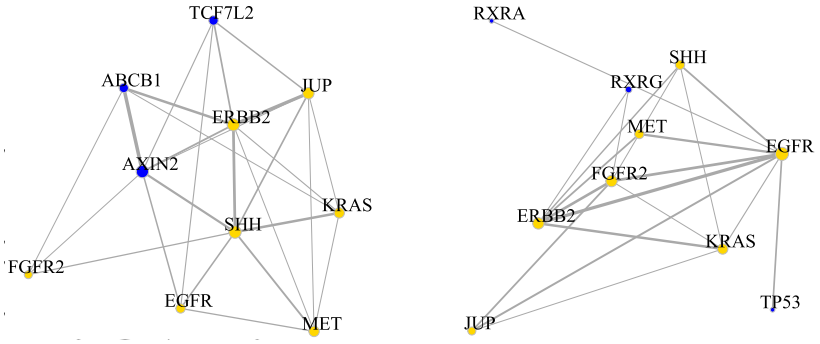

FIG 3. *The top 10 genes according to their importance in the differential gene regulatory networks between disease and normal issues inferred by the DTL (left panel) and CrossFDTL (right panel) methods. The common genes identified by both methods are shown in yellow, while the non-common genes are shown in blue. The width of the edges shows the strength of the links, and the size of nodes shows the sum of the strength of edges linked to them.*

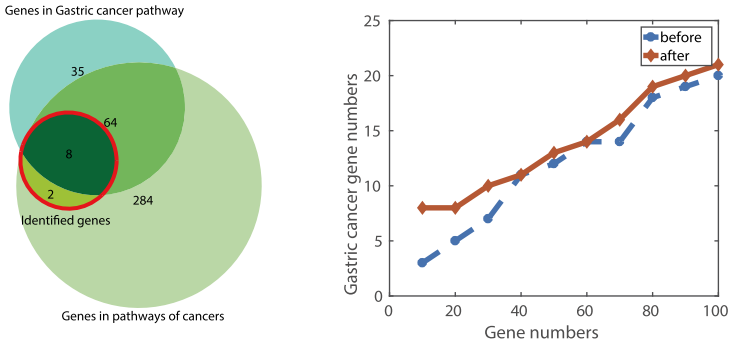

FIG 4. *Predicted genes and known gastric cancer related genes. (a) 107 known gastric cancer genes are obtained from the KEGG pathway database, 8 of which appear in our top 10 predictions. (b) The number of gastric cancer related genes is significantly higher after our prediction, especially in the forward genes.*

CrossFDTL. We can see that the identified top 10 genes have 8 common genes with those in the gastric cancer pathway. Furthermore, there are 64 common genes between the 358 genes in the biggest circle and the genes in the gastric cancer pathway. This shows that the identified genes have significantly high incidence in gastric cancer, with a p-value of 0.007 by Fisher's test.

6. Discussion

In this paper, we constructed a fused D-trace loss function with an efficient optimization algorithm, the CrossFDTL formulation, to infer the differential

network between two precision matrices based on the Gaussian assumption. We established the consistency and rate of convergence of the proposed method. The theoretical and computational results show the virtue of our formulation compared with the existing methods. It will be interesting to study the convergence speed of the proposed method with respect to the alternating minimization steps, and generalize the current methodology to non-Gaussian case. Further application to other practical examples will be pursued as a future task.

Acknowledgement

T. Li is supported by Beijing Academy of Artificial Intelligence (BAAI) and the NSFC under grants Nos. 11421101 and 11825102. L. Chen is supported by National Key R&D Program of China under grant No. 2017YFA0505500 and NSFC under grant No. 31771476. The authors thank Profs. Weiguo Gao and Minghua Deng for stimulating discussions.

Appendix

A.1. Tail conditions

In Sections A.1 and A.2, we give the detailed proof of Theorem 2. The idea of the proof can be summarized in the following three steps.

(1) Estimate the error between the covariance matrices Σ_X, Σ_Y and sample covariance matrices $\hat{\Sigma}_X, \hat{\Sigma}_Y$ through tail conditions. It will introduce the lower bound on sample sizes n_X and n_Y.
(2) Derive the condition under which the solution of l_1-penalized optimization problem (4) keeps the structure of nonzero entries. That is exactly what Lemma 2 shows. Only the optimality condition and some ordinary inequalities are used to obtain the result.
(3) Apply the error estimates obtained in (1) to the condition inferred in (2). Besides sample sizes, we also need to give an appropriate range of λ to ensure the condition holds. The technique is to repeatedly use the elementary Lemmas 3 and 4.

Below we present the proof details.

According to Ravikumar et al. [21], if a zero-mean random vector X has a sub-Gaussian tail, then X satisfies the tail condition $\mathcal{T}(f, \nu_*)$, i.e., there exists a constant $\nu_* > 0$ and a function $f : \mathbb{N} \times (0, \infty) \to (0, \infty)$ such that

$$\mathbb{P}(|\hat{\Sigma}^n_{i,j} - \Sigma_{i,j}| \geq \delta) \leq 1/f(n, \delta) \quad (i, j = 1, \ldots, p, 0 < \delta < 1/\nu_*),$$

where Σ is the covariance matrix of X and $\hat{\Sigma}^n$ is the sample covariance matrix of X from n samples. For each fixed $\delta > 0$ and n, we can define the inverse functions of f for $r \geq 1$ as

$$n_f(\delta, r) = \max\{n : f(n, \delta) \leq r\}, \quad \delta_f(n, r) = \max\{\delta : f(n, \delta) \leq r\}.$$

If X has a sub-Gaussian tail with parameter σ, we have [21]

$$\begin{aligned}
\nu_* &= \left\{\max_i \Sigma^*_{i,i} 8(1+4\sigma^2)\right\}^{-1}, \\
f(n,\delta) &= \exp(c_* n\delta^2)/4, \quad c_* = \left\{123(1+4\sigma^2)^2 \max_i (\Sigma^*_{i,i})^2\right\}^{-1}, \\
\delta_f(n,p^\eta) &= \left\{128(1+4\sigma^2)^2 \max_i (\Sigma^*_{i,i})^2 (\eta\log p + \log 4)/n\right\}^{1/2}, \\
n_f(\delta,p^\eta) &= 128(1+4\sigma^2)^2 \max_i (\Sigma^*_{i,i})^2 (\eta\log p + \log 4)/\delta^2.
\end{aligned}$$

With the help of above results, we can obtain the following important lemma.

Lemma 2. *The support of $\hat{\Delta}$ lies in the support of Δ^* if*

$$\max_{e\in S^c} |\hat{E}^I_e \mathrm{vec}(I)| + \lambda \|\hat{E}^Z_e\|_\infty \le \lambda, \tag{A.1}$$

where $\mathrm{vec}(I)$ *is the p^2-vector formed by stacking the columns of matrix I. $\hat{E}^I \in \mathbb{R}^{p^2}$, $\hat{E}^I \in \mathbb{R}^s$ have the form*

$$\begin{aligned}
\hat{E}^I_e =& -2\hat{\Gamma}^{(1)}_{eS^+}\left(\hat{\Gamma}^{(1)}_{S^+S^+} - \hat{\Gamma}^{(2)}_{S^+.}\hat{\Gamma}^{(1)^{-1}}\hat{\Gamma}^{(2)}_{.S^+}\right)^{-1}\hat{\Gamma}^{(2)}_{S^+.}\hat{\Gamma}^{(1)^{-1}} \\
&+ 2\hat{\Gamma}^{(2)}_{e.}\left(\hat{\Gamma}^{(1)} - \hat{\Gamma}^{(2)}_{.S^+}\hat{\Gamma}^{(1)^{-1}}_{S^+S^+}\hat{\Gamma}^{(2)}_{S^+.}\right)^{-1}, \\
\hat{E}^Z_e =& -\hat{\Gamma}^{(1)}_{eS^+}\left(\hat{\Gamma}^{(1)}_{S^+S^+} - \hat{\Gamma}^{(2)}_{S^+.}\hat{\Gamma}^{(1)^{-1}}\hat{\Gamma}^{(2)}_{.S^+}\right)^{-1} \\
&+ \hat{\Gamma}^{(2)}_{e.}\left(\hat{\Gamma}^{(1)} - \hat{\Gamma}^{(2)}_{.S^+}\hat{\Gamma}^{(1)^{-1}}_{S^+S^+}\hat{\Gamma}^{(2)}_{S^+.}\right)^{-1}\hat{\Gamma}^{(2)}_{.S^+}\hat{\Gamma}^{(1)^{-1}}_{S^+S^+}.
\end{aligned}$$

Proof. Consider

$$(\tilde{\Delta},\tilde{S}) = \mathop{\arg\min}_{S=S^{\mathrm{T}},\Delta=\Delta^{\mathrm{T}},\Delta_{S^c}=0} L(S,\Delta). \tag{A.2}$$

From the optimality condition, we obtain

$$\begin{aligned}
&\hat{\Gamma}^{(1)}_{S^+S^+}\mathrm{vec}(\tilde{\Delta})_{S^+} + \hat{\Gamma}^{(2)}_{S^+.}\mathrm{vec}(\tilde{S}) = -\lambda\mathrm{vec}(Z)_{S^+}, \\
&\hat{\Gamma}^{(2)}_{.S^+}\mathrm{vec}(\tilde{\Delta})_{S^+} + \hat{\Gamma}^{(1)}\mathrm{vec}(\tilde{S}) = 2\mathrm{vec}(I).
\end{aligned} \tag{A.3}$$

Here $Z = \mathrm{sgn}(\tilde{\Delta})$ is derived from the subgradient of l_1 penalty. Solving (A.3), we get

$$\begin{aligned}
\mathrm{vec}(\tilde{\Delta})_{S^+} =& -\left(\hat{\Gamma}^{(1)}_{S^+S^+} - \hat{\Gamma}^{(2)}_{S^+.}\hat{\Gamma}^{(1)^{-1}}\hat{\Gamma}^{(2)}_{.S^+}\right)^{-1}\left(\lambda\mathrm{vec}(Z)_{S^+} + 2\hat{\Gamma}^{(2)}_{S^+.}\hat{\Gamma}^{(1)^{-1}}\mathrm{vec}(I)\right), \\
\mathrm{vec}(\tilde{S}) =& \left(\hat{\Gamma}^{(1)} - \hat{\Gamma}^{(2)}_{.S^+}\hat{\Gamma}^{(1)^{-1}}_{S^+S^+}\hat{\Gamma}^{(2)}_{S^+.}\right)^{-1}\left(2\mathrm{vec}(I) + \lambda\hat{\Gamma}^{(2)}_{.S^+}\hat{\Gamma}^{(1)^{-1}}_{S^+S^+}\mathrm{vec}(Z)_{S^+}\right).
\end{aligned} \tag{A.4}$$

And we have

$$\max_{e\in S^c} |\hat{E}^I_e \mathrm{vec}(I) + \lambda\hat{E}^Z_e \mathrm{vec}(Z)_{S^+}| \le \max_{e\in S^c} \|\hat{E}^I_e\|_\infty + \lambda\|\hat{E}^Z_e\|_\infty \le \lambda. \tag{A.5}$$

Combining (A.4) and (A.5), we obtain

$$\max_{e\in S^c}\left|\hat{\Gamma}^{(1)}_{eS^+}\mathrm{vec}(\tilde{\Delta})_{S^+}+\hat{\Gamma}^{(2)}_{e\cdot}\mathrm{vec}(\tilde{S})\right|\le\lambda. \tag{A.6}$$

(A.3) and (A.6) show that

$$(\tilde{\Delta},\tilde{S})=\underset{S=S^{\mathrm{T}},\Delta=\Delta^{\mathrm{T}}}{\arg\min}\ L(S,\Delta),$$

which implies $(\tilde{\Delta},\tilde{S})=(\hat{\Delta},\hat{S})$. Therefore $\hat{\Delta}_{S^c}=0$, i.e. the support of $\hat{\Delta}$ lies in the support of Δ^*. □

We remark that the term $\hat{E}^I\mathrm{vec}(I)$ exists due to the sample covariance matrix. On the other hand, we have $E^I\mathrm{vec}(I)=0$, which can be inferred by similar analysis for (Δ^*,S^*). The term $\hat{E}^Z\mathrm{vec}(Z)$ is related to the sample covariance and the l_1 penalty, but is independent of the magnitude of λ. Since $\|\mathrm{vec}(Z)\|_\infty\le 1$, then $|E^Z_e\mathrm{vec}(Z)_{S^+}|\le\|E^Z_e\|_1$. This is the source of the irrepresentability condition (25).

A.2. Proof of Theorem 2

For simplicity, we first define some notations:

$$\begin{aligned}
C_1^*&=\Gamma^{*(1)}_{S^+S^+}-\Gamma^{*(2)}_{S^+\cdot}{\Gamma^{*(1)}}^{-1}\Gamma^{*(2)}_{\cdot S^+},\quad C_2^*=\Gamma^{*(1)}-\Gamma^{*(2)}_{\cdot S^+}{\Gamma^{*(1)}_{S^+S^+}}^{-1}\Gamma^{*(2)}_{S^+\cdot},\\
\hat{C}_1&=\hat{\Gamma}^{(1)}_{S^+S^+}-\hat{\Gamma}^{(2)}_{S^+\cdot}{\hat{\Gamma}^{(1)}}^{-1}\hat{\Gamma}^{(2)}_{\cdot S^+},\quad \hat{C}_2=\hat{\Gamma}^{(1)}-\hat{\Gamma}^{(2)}_{\cdot S^+}{\hat{\Gamma}^{(1)}_{S^+S^+}}^{-1}\hat{\Gamma}^{(2)}_{S^+\cdot},\\
e_c^{(1)}&=|||\hat{C}_1^{-1}-C_1^{*-1}|||_\infty,\quad e_c^{(2)}=|||\hat{C}_2^{-1}-C_2^{*-1}|||_\infty,\\
e^{(1)}&=|||{\hat{\Gamma}^{(1)}}^{-1}-{\Gamma^{*(1)}}^{-1}|||_\infty,\quad e^{(2)}=|||{\hat{\Gamma}^{(1)}_{S^+S^+}}^{-1}-{\Gamma^{*(1)}_{S^+S^+}}^{-1}|||_\infty,
\end{aligned}$$

$$\begin{aligned}
M=\max\Big\{&\|{\Gamma^{*(1)}}^{-1}\|_\infty,\|{\Gamma^{*(1)}_{S^+S^+}}^{-1}\|_\infty,\|\Gamma^{*(2)}\|_\infty,\\
&\|\Gamma^{*(2)}_{S^+\cdot}{\Gamma^{*(1)}}^{-1}\|_\infty,\|\Gamma^{*(2)}_{\cdot S^+}{\Gamma^{*(1)}_{S^+S^+}}^{-1}\|_\infty,\|C_1^*\|_\infty,\|C_2^*\|_\infty,\\
&\|{C_1^*}^{-1}\|_\infty,\|{C_2^*}^{-1}\|_\infty,\|\Gamma^{*(1)}_{eS^+}{C_1^*}^{-1}\|_\infty,\|\Gamma^{*(2)}_{e\cdot}{C_2^*}^{-1}\|_\infty,1\Big\},
\end{aligned}$$

$$r=\max\left\{p^2,s\right\}=p^2,$$

where $|||\cdot|||_\infty$ is the operator norm induced by the ℓ_∞ norm of vectors. We use this notation to avoid the confusion with $\|\cdot\|_\infty$ defined before.

Since X and Y have sub-Gaussian tails, they satisfy the tail condition $\mathcal{T}(f_X,\nu_{X*})$ or $\mathcal{T}(f_Y,\nu_{Y*})$, respectively. Let $\nu_*=\max(\nu_{X*},\nu_{Y*})$, then X and Y also satisfy the tail condition $\mathcal{T}(f_X,\nu_*)$ and $\mathcal{T}(f_Y,\nu_*)$. Let

$$\bar{\delta}=\min\left\{\alpha(22r^9M^8)^{-1},1/\nu_*\right\}.$$

For sub-Gaussian tail, we have

$$\bar{\delta} = \min\left\{\alpha(22r^9M^8)^{-1}, \min_{X,Y}\left\{\max_i \Sigma^*_{X_{i,i}} 8(1+4\sigma_X^2), \max_i \Sigma^*_{Y_{i,i}} 8(1+4\sigma_Y^2)\right\}\right\}.$$

In the following, for $\eta > 2$, we assume $n_X > n_{f_X}(\bar{\delta}, p^\eta), n_Y > n_{f_Y}(\bar{\delta}, p^\eta)$ and

$$\frac{22}{\alpha}C_\lambda\delta \le \lambda \le \frac{24}{\alpha}C_\lambda\delta,$$

where $C_\lambda = r^9M^8$ and $\delta := \max\{\delta_{f_X}(n_X, p^\eta), \delta_{f_Y}(n_Y, p^\eta)\}$. Let

$$C_G = 128\left\{1 + 4\max(\sigma_X^2, \sigma_Y^2)\right\}^2 \max_i(\Sigma_{X_{i,i}}, \Sigma_{Y_{i,i}})^2,$$
$$C_P = 4\max_i(\Sigma_{X_{i,i}}, \Sigma_{Y_{i,i}})^2\left\{\max(K_{X_m}, K_{Y_m}) + 1\right\}^{1/m},$$

then $n_X > n_{f_X}(\bar{\delta}, p^\eta)$, and we have $\delta_{f_X}(n_X, p^\eta) < \bar{\delta}$ with probability at least $1 - 1/p^{\eta-2}$. Similar result holds for Y.

Now we will prove (A.1), and then complete the proof of Theorems 2 through Lemma 2. The estimation of (A.1) is relatively technical. We will utilize the following two important lemmas.

Lemma 3. *For $\hat{v}, v \in \mathbb{R}^m, \hat{A}, A \in \mathbb{R}^{m\times n}, \hat{B}, B \in \mathbb{R}^{n\times l}$, we have*

$$\|\hat{v}^{\mathrm{T}}\hat{A} - v^{\mathrm{T}}A\|_1 \le \|\hat{v} - v\|_1 \,|||\, \hat{A} - A \,|||_\infty + \|\hat{v} - v\|_1 \,|||\, A \,|||_\infty + \|v\|_1 \,|||\, \hat{A} - A|||_\infty,$$
$$|||\hat{A}\hat{B} - AB|||_\infty \le |||\, \hat{A} - A \,|||_\infty \,|||\hat{B} - B \,|||_\infty + |||\, \hat{A} - A \,|||_\infty \,|||B|||_\infty + |||A \,|||_\infty \,|||\hat{B} - B \,|||_\infty .$$

Lemma 4. *For matrices A and E, if $|||A^{-1} \,|||_\infty \,|||E|||_\infty < 1$, then we have*

$$|||(A+E)^{-1} - A^{-1}|||_\infty \le \frac{|||A^{-1} \,|||_\infty^2 \,|||E|||_\infty}{1 - |||A^{-1} \,|||_\infty \,|||E|||_\infty}.$$

Lemma 3 is trivial. It only uses triangle inequality and the facts $\|v^{\mathrm{T}}A\|_1 \le \|v\|_1 \,|||\, A|||_\infty$, $|||AB|||_\infty \le |||A \,|||_\infty \,|||B|||_\infty$. Lemma 4 is a result of matrix pertubation theory [25]. Below we will frequently use these two lemmas without further explanation.

Firstly, according to the definition of Γ, we can easily obtain

$$\|\hat{\Gamma}^{(1)} - \Gamma^{*(1)}\|_\infty \le \max(\delta_{f_X}(n_X, p^\eta), \delta_{f_Y}(n_Y, p^\eta)) =: \delta,$$
$$|||\, \hat{\Gamma}^{(1)}_{S^+S^+} - \Gamma^{*(1)}_{S^+S^+} \,|||_\infty \le |||\hat{\Gamma}^{(1)} - \Gamma^{*(1)}|||_\infty \le r\delta.$$

Then

$$e^{(1)} \le \frac{|||\Gamma^{*(1)^{-1}} \,|||_\infty^2 \,|||\hat{\Gamma}^{(1)} - \Gamma^{*(1)}|||_\infty}{1 - |||\Gamma^{*(1)^{-1}} \,|||_\infty \,|||\hat{\Gamma}^{(1)} - \Gamma^{*(1)}|||_\infty}$$

$$\begin{aligned} &\leq \frac{(rM)^2 \cdot r\delta}{1 - rM \cdot r\delta} = \frac{r^3 M^2 \delta}{1 - r^2 M\delta} \\ &\leq 2r^3 M^2 \delta, \end{aligned}$$

and similarly

$$e^{(2)} \leq 2r^3 M^2 \delta.$$

With the estimation for $e^{(1)}$, we get

$$\begin{aligned} |||\hat{\Gamma}^{(2)}_{S^+.} \hat{\Gamma}^{(1)^{-1}} - \Gamma^{*(2)}_{S^+.} \Gamma^{*(1)^{-1}}|||_\infty \leq & |||\hat{\Gamma}^{(2)}_{S^+.} - \Gamma^{*(2)}_{S^+.}|||_\infty |||\hat{\Gamma}^{(1)^{-1}} - \Gamma^{*(1)^{-1}}|||_\infty \\ & + |||\hat{\Gamma}^{(2)}_{S^+.} - \Gamma^{*(2)}_{S^+.}|||_\infty |||\Gamma^{*(1)^{-1}}|||_\infty \\ & + |||\Gamma^{*(2)}_{S^+.}|||_\infty |||\hat{\Gamma}^{(1)^{-1}} - \Gamma^{*(1)^{-1}}|||_\infty \\ \leq & r\delta \cdot e^{(1)} + r\delta \cdot rM + rM \cdot e^{(1)} \\ \leq & 2r^4 M^2 \delta^2 + r^2 M\delta + 2r^4 M^3 \delta \\ \leq & 3r^4 M^3 \delta, \end{aligned} \tag{A.7}$$

and

$$\begin{aligned} & |||\hat{\Gamma}^{(2)}_{S^+.} \hat{\Gamma}^{(1)^{-1}} \hat{\Gamma}^2_{.S^+} - \Gamma^{*(2)}_{S^+.} \Gamma^{*(1)^{-1}} \Gamma^{*(2)}_{.S^+}|||_\infty \\ \leq & |||\hat{\Gamma}^{(2)}_{S^+.} \hat{\Gamma}^{(1)^{-1}} - \Gamma^{*(2)}_{S^+.} \Gamma^{*(1)^{-1}}|||_\infty |||\hat{\Gamma}^{(2)}_{.S^+} - \Gamma^{*(2)}_{.S^+}|||_\infty \\ & + |||\hat{\Gamma}^{(2)}_{S^+.} \hat{\Gamma}^{(1)^{-1}} - \Gamma^{*(2)}_{S^+.} \Gamma^{*(1)^{-1}}|||_\infty |||\Gamma^{*(2)}_{.S^+}|||_\infty \\ & + |||\Gamma^{*(2)}_{S^+.} \Gamma^{*(1)^{-1}}|||_\infty |||\hat{\Gamma}^{(2)}_{.S^+} - \Gamma^{*(2)}_{.S^+}|||_\infty \\ \leq & 3r^4 M^3 \delta(r\delta + rM) + rM \cdot r\delta \\ = & 3r^5 M^3 \delta^2 + r^2 M\delta + 3r^5 M^4 \delta. \end{aligned} \tag{A.8}$$

According to the definition of $\hat{C}_1$ and C^*_1, we get

$$|||\hat{C}_1 - C^*_1|||_\infty \leq r\delta + 3r^5 M^3 \delta^2 + r^2 M\delta + 3r^5 M^4 \delta \leq 4r^5 M^4 \delta. \tag{A.9}$$

Similar as the estimates in (A.7) and (A.8), we get

$$|||\hat{C}_2 - C^*_2|||_\infty \leq 4r^5 M^4 \delta. \tag{A.10}$$

Utilizing (A.9), we obtain

$$\begin{aligned} e^{(1)}_c \leq & \frac{|||C^*_1|||^2_\infty |||\hat{C}_1 - C^*_1|||_\infty}{1 - |||C^*_1|||_\infty |||\hat{C}_1 - C^*_1|||_\infty} \\ \leq & \frac{(rM)^2 \cdot 4r^5 M^4 \delta}{1 - rM \cdot 4r^5 M^4 \delta} \leq 8r^7 M^6 \delta, \end{aligned} \tag{A.11}$$

and similarly

$$e^{(2)}_c \leq 8r^7 M^6 \delta. \tag{A.12}$$

Applying (A.11) and (A.12), we have

$$\|\hat{\Gamma}^{(1)}_{eS^+} \hat{C}_1^{-1} - \Gamma^{*(1)}_{eS^+} C^{*^{-1}}_1\|_1 \leq r\delta e^{(1)}_c + r\delta \cdot rM + rM e^{(1)}_c$$

$$\begin{aligned}&\leq 8r^8M^6\delta^2 + r^2M\delta + 8r^8M^7\delta\\&\leq 9r^8M^7\delta\end{aligned} \tag{A.13}$$

and

$$\|\hat{\Gamma}^{(2)}_{e.}\hat{C}_2^{-1} - \Gamma^{*(2)}_{e.}C_2^{*^{-1}}\|_1 \leq 9r^8M^7\delta. \tag{A.14}$$

Combining (A.7) and (A.13), we get

$$\begin{aligned}&\|\hat{\Gamma}^{(1)}_{eS^+}\hat{C}_1^{-1}\hat{\Gamma}^2_{S^+.}\hat{\Gamma}^{(1)^{-1}} - \Gamma^{*(1)}_{eS^+}C_1^{*^{-1}}\Gamma^{*(2)}_{S^+.}\Gamma^{*(1)^{-1}}\|_1\\\leq&\|\hat{\Gamma}^{(1)}_{eS^+}\hat{C}_1^{-1} - \Gamma^{*(1)}_{eS^+}C_1^{*^{-1}}\|_1 \,|||\, \hat{\Gamma}^{(2)}_{S^+.}\hat{\Gamma}^{(1)^{-1}} - \Gamma^{*(2)}_{S^+.}\Gamma^{*(1)^{-1}}\,|||_\infty\\&+\|\hat{\Gamma}^{(1)}_{eS^+}\hat{C}_1^{-1} - \Gamma^{*(1)}_{eS^+}C_1^{*^{-1}}\|_1 \,|||\, \Gamma^{*(2)}_{S^+.}\Gamma^{*(1)^{-1}}\,|||_\infty\\&+\|\Gamma^{*(1)}_{eS^+}C_1^{*^{-1}}\|_1 \,|||\, \hat{\Gamma}^2_{S^+.}\hat{\Gamma}^{(1)^{-1}} - \Gamma^{*(2)}_{S^+.}\Gamma^{*(1)^{-1}}\,|||_\infty\\\leq&9r^8M^7\delta\cdot 3r^4M^3\delta + 9r^8M^7\delta\cdot rM + rM\cdot 3r^4M^3\delta\\\leq&10r^9M^8\delta,\end{aligned} \tag{A.15}$$

and similarly

$$\|\hat{\Gamma}^{(2)}_{e.}\hat{C}_2^{-1}\hat{\Gamma}^{(2)}_{.S^+}\hat{\Gamma}^{(1)^{-1}}_{S^+S^+} - \Gamma^{*(2)}_{e.}C_2^{*^{-1}}\Gamma^{*(2)}_{.S^+}\Gamma^{*(1)^{-1}}_{S^+S^+}\|_1 \leq 10r^9M^8\delta. \tag{A.16}$$

Recall the definitions of $\hat{E}^I_e$ and $\hat{E}^Z_e$ and utilize (A.13), (A.14), (A.15) and (A.16), we get

$$\begin{aligned}\|\hat{E}^I_e - E^{*^I}_e\|_1 \leq&10r^9M^8\delta + 9r^8M^7\delta \leq 11r^9M^8\delta,\\\|\hat{E}^Z_e - E^{*^Z}_e\|_1 \leq&10r^9M^8\delta + 9r^8M^7\delta \leq 11r^9M^8\delta.\end{aligned} \tag{A.17}$$

From the definition

$$(\Delta^*, S^*) = \underset{\Delta^{\mathrm{T}}=\Delta, S^{\mathrm{T}}=S}{\arg\min}\ L_F(S, \Delta, \Sigma_X, \Sigma_Y),$$

we have the following optimality conditions

$$\begin{aligned}&\Gamma^{*(1)}_{S^+S^+}\mathrm{vec}(\Delta^*)_{S^+} + \Gamma^{*(2)}_{S^+.}\mathrm{vec}(S^*) = 0,\\&\Gamma^{*(2)}_{.S^+}\mathrm{vec}(\Delta^*)_{S^+} + \Gamma^{*(1)}\mathrm{vec}(S^*) = 2\mathrm{vec}(I),\\&\Gamma^{*(1)}_{eS^+}\mathrm{vec}(\Delta^*)_{S^+} + \Gamma^{*(2)}_{e.}\mathrm{vec}(S^*) = 0, \quad e \in S^c.\end{aligned} \tag{A.18}$$

Simple calculation shows that

$$E^{*^I}_e\mathrm{vec}(I) = 0, \quad e \in S^c. \tag{A.19}$$

Combining (A.17) and (A.19), we obtain

$$\begin{aligned}&\left|\hat{E}^I_e\mathrm{vec}(I) + \lambda\hat{E}^Z_e\mathrm{vec}(Z)_{S^+}\right|\\=&\left|\left(\hat{E}^I_e - E^{*^I}_e\right)\mathrm{vec}(I) + \lambda\left(\hat{E}^Z_e - E^{*^Z}_e + E^{*^Z}_e\right)\mathrm{vec}(Z)_{S^+}\right|\end{aligned}$$

$$
\begin{aligned}
&\leq \|\hat{E}_e^I - {E_e^*}^I\|_1 + \lambda\left(\|\hat{E}_e^Z - {E_e^*}^Z\|_1 + \|{E_e^*}^Z\|_1\right)\\
&\leq 11r^9M^8\delta + \lambda(11r^9M^8\delta + 1 - \alpha)\\
&= \lambda + 11r^9M^8\delta(\lambda+1) - \alpha\lambda\\
&< \lambda + \frac{\alpha}{2}\lambda + 11r^9M^8\delta - \alpha\lambda\\
&= \lambda + 11r^9M^8\delta - \frac{\alpha}{2}\lambda\\
&< \lambda.
\end{aligned}
$$

Thus the condition of Lemma 2 is confirmed, and we have the support of $\hat{\Delta}$ lies in the support of Δ^*.

From (A.18) we get

$$
\mathrm{vec}(\Delta^*)_{S^+} = -\left(\Gamma^{*(1)}_{S^+S^+} - \Gamma^{*(2)}_{S^+\cdot}{\Gamma^{*(1)}}^{-1}\Gamma^{*(2)}_{\cdot S^+}\right)^{-1} 2\Gamma^{*(2)}_{S^+\cdot}{\Gamma^{*(1)}}^{-1}\mathrm{vec}(I).
$$

Combining with (A.4) we obtain

$$
\begin{aligned}
&\|\hat{\Delta} - \Delta^*\|_\infty = \|\mathrm{vec}(\hat{\Delta})_{S^+} - \mathrm{vec}(\Delta^*)_{S^+}\|_\infty\\
=&\| -\hat{C}_1^{-1}2\hat{\Gamma}^{*(2)}_{S^+}\hat{\Gamma}^{*(1)^{-1}}\mathrm{vec}(I) + {C_1^*}^{-1}2\Gamma^{*(2)}_{S^+}{\Gamma^{*(1)}}^{-1}\mathrm{vec}(I) - \lambda\hat{C}_1^{-1}\mathrm{vec}(Z)_{S^+}\|_\infty\\
\leq& \|\!| -\hat{C}_1^{-1}2\hat{\Gamma}^{*(2)}_{S^+}\hat{\Gamma}^{*(1)^{-1}} + {C_1^*}^{-1}2\Gamma^{*(2)}_{S^+}{\Gamma^{*(1)}}^{-1}\|\!|_\infty + \lambda\,\|\!|\,\hat{C}_1^{-1}\|\!|_\infty\\
\leq& 2(8r^7M^6\delta\cdot 3r^4M^3\delta + rM\cdot 3r^4M^3\delta + 8r^7M^6\delta\cdot rM) + \lambda(rM + 8r^7M^6\delta)\\
\leq& 25r^{10}M^9\delta.
\end{aligned}
$$

Since X and Y are sub-Gaussian, we have

$$
\|\hat{\Delta} - \Delta^*\|_\infty \leq M_G\left\{\frac{\eta\log p + \log 4}{\min(n_X, n_Y)}\right\}^{1/2},
$$

where

$$
M_G = 200\sqrt{2}r^{10}M^9(1+4\sigma^2)\max_i(\Sigma^*_{X_{i,i}}, \Sigma^*_{Y_{i,i}}).
$$

Since we have known that the support of $\hat{\Delta}$ lies in the support of Δ^*,

$$
\|\hat{\Delta} - \Delta^*\|_F \leq M_G\left\{\frac{\eta\log p + \log 4}{\min(n_X, n_Y)}\right\}^{1/2} s^{1/2}.
$$

If $\min(n_X, n_Y) > M_G^2(\eta\log p + \log 4)/(\min_{j,k:\Delta^*_{j,k}\neq 0}|\Delta^*_{j,k}|)^2$, we have

$$
\|\hat{\Delta} - \Delta^*\|_\infty < \min_{j,k:\Delta^*_{j,k}\neq 0}|\Delta^*_{j,k}|.
$$

Therefore $\mathrm{sgn}(\hat{\Delta}_{i,j}) = \mathrm{sgn}(\Delta^*_{i,j})$ for all i, j with probability at least $1 - 2/p^{\eta-2}$.

A.3. Supplementary figures for simulation studies

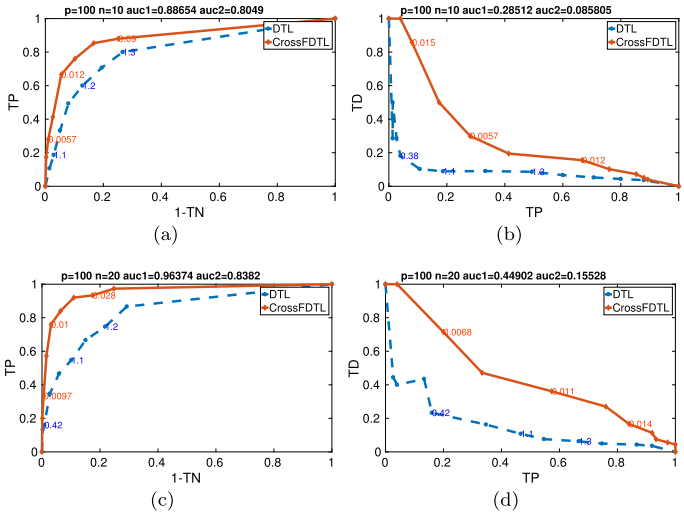

FIG 5. *The left panels show the receiver operating characteristic curves for the support recovery of* Δ^**, and the right panels show the precision-recall curves, with (a)(b)* $p = 100, n = 10$*, (c)(d)* $p = 100, n = 20$ *in model1. The values auc*1 *and auc*2 *are the area under the curves correspond to CrossFDTL and DTL. The solid and dashed lines correspond to the CrossFDTL and DTL, respectively.*

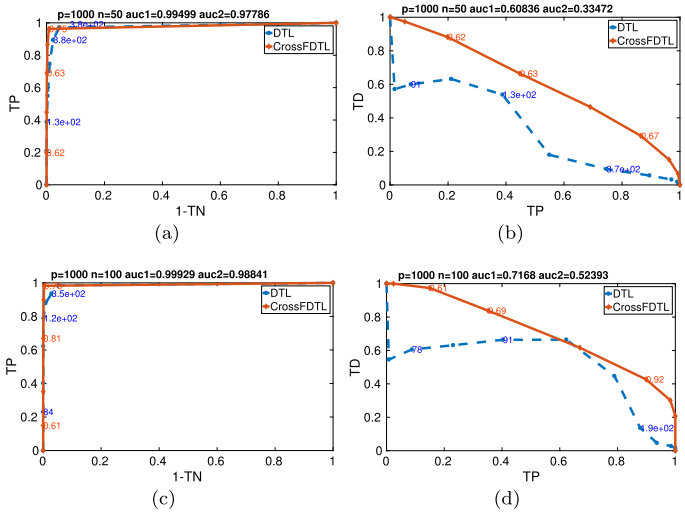

FIG 6. *The left panels show the receiver operating characteristic curves for the support recovery of* Δ^**, and the right panels show the precision-recall curves, with (a)(b)* $p = 1000, n = 50$*, (c)(d)* $p = 1000, n = 100$ *in model1. The values auc*1 *and auc*2 *are the area under the curves correspond to CrossFDTL and DTL. The solid and dashed lines correspond to the CrossFDTL and DTL, respectively.*

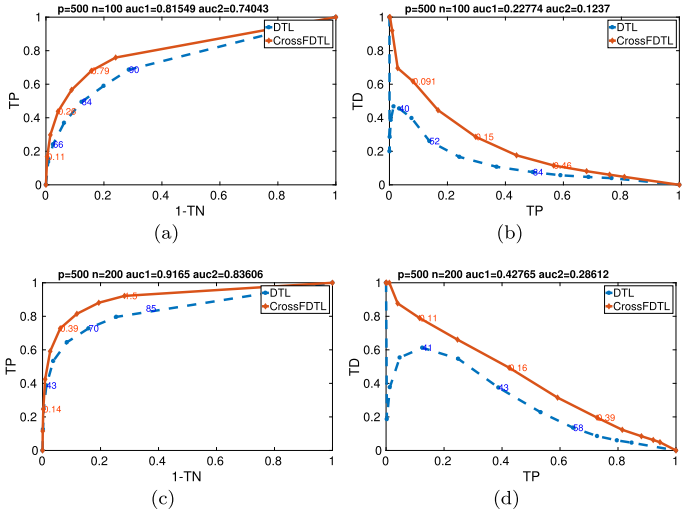

FIG 7. *The left panels show the receiver operating characteristic curves for the support recovery of* Δ^**, and the right panels show the precision-recall curves, with (a)(b)* $p = 500, n = 100$*, (c)(d)* $p = 500, n = 200$ *in model2. The values* auc1 *and* auc2 *are the area under the curves correspond to CrossFDTL and DTL. The solid and dashed lines correspond to the CrossFDTL and DTL, respectively.*

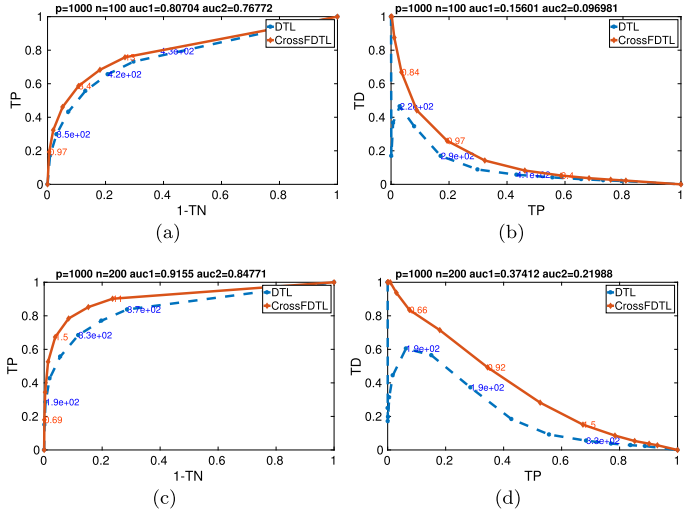

FIG 8. *The left panels show the receiver operating characteristic curves for the support recovery of* Δ^**, and the right panels show the precision-recall curves, with (a)(b)* $p = 1000, n = 100$*, (c)(d)* $p = 1000, n = 200$ *in model2. The values* auc1 *and* auc2 *are the area under the curves correspond to CrossFDTL and DTL. The solid and dashed lines correspond to the CrossFDTL and DTL, respectively.*

A.4. Supplementary figures for real data analysis

We apply our CrossFDTL method to different types of cancer data from TCGA and make comparisons with DTL. For each type of cancer, we showed the inferred differential gene regulatory networks in two ways. The only difference is the way of the choices of genes. We showed the top 10 genes according to their importance in the differential networks inferred by the two methods. Here are two different definitions of the importance of gene i used in this paper:

Sum of strength:

$$I(i) = \sum_{j \neq i} |\hat{\Delta}_{ij}|,$$

Degree:

$$I(i) = \#\left\{j | j \neq i, \hat{\Delta}_{ij} \neq 0\right\}.$$

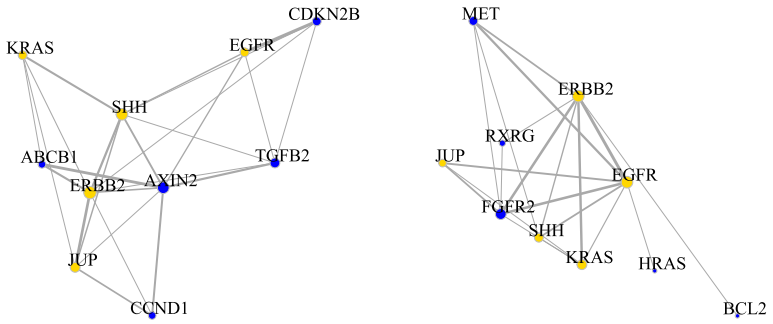

FIG 9. *Gastric cancer, top 10 genes identified by the degree (top 10 genes identified by the sum of strength are shown in the main text).*

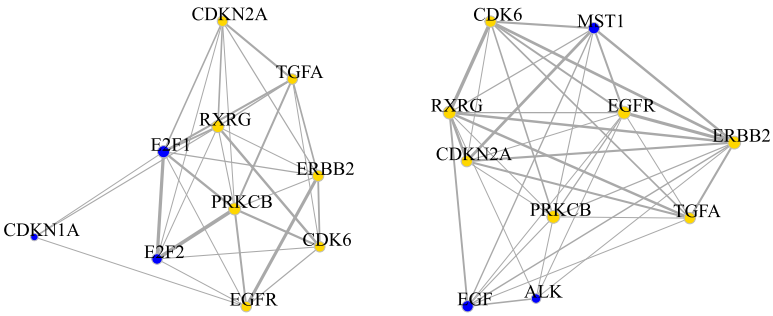

FIG 10. *Lung adenocarcinoma (LUAD), top 10 genes identified by the sum of strength.*

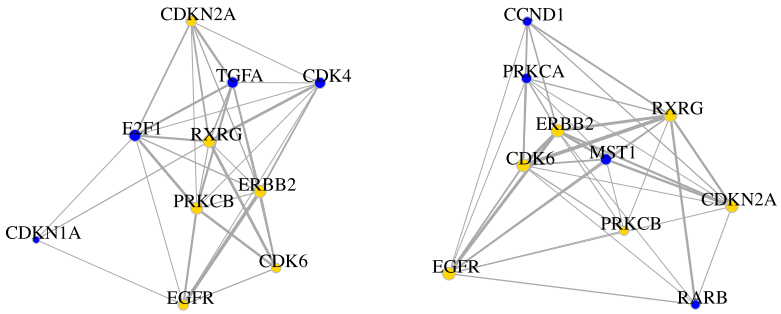

FIG 11. *Lung adenocarcinoma (LUAD), top 10 genes identified by the degree.*

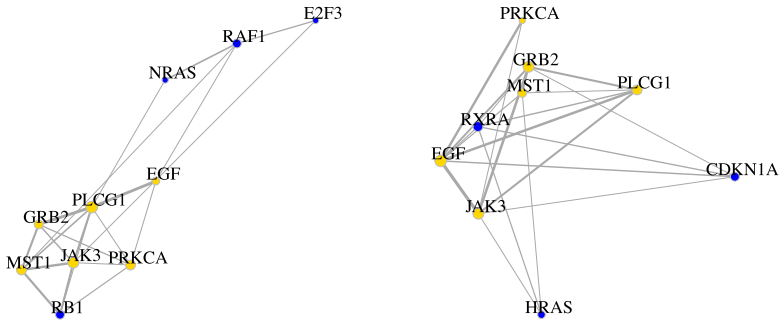

FIG 12. *Lung squamous cell carcinoma (LUSC), top 10 genes identified by the sum of strength.*

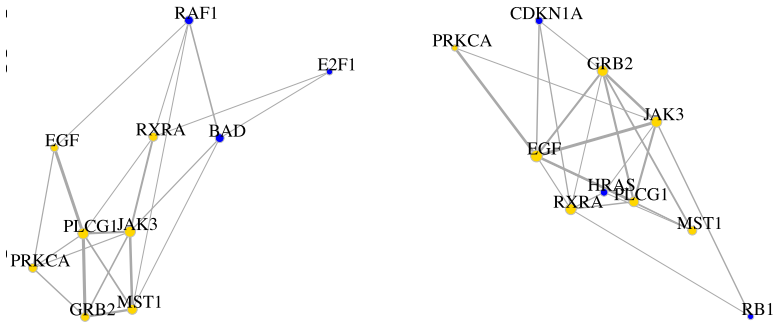

FIG 13. *Lung squamous cell carcinoma (LUSC), top 10 genes identified by the degree.*

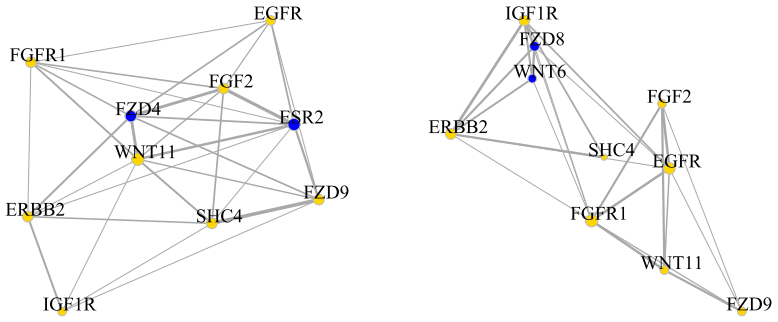

FIG 14. *Breast invasive carcinoma (BRCA), top 10 genes identified by the sum of strength.*

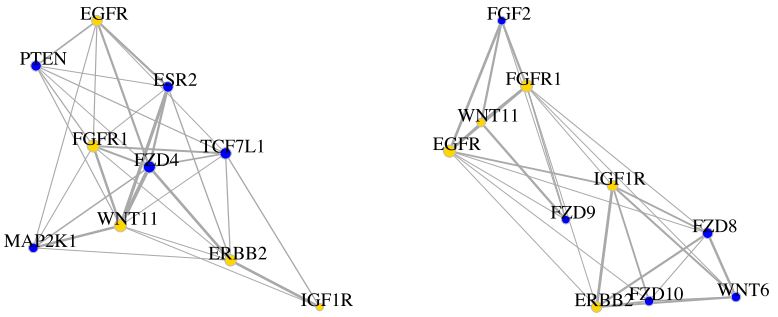

FIG 15. *Breast invasive carcinoma (BRCA), top 10 genes identified by the degree.*

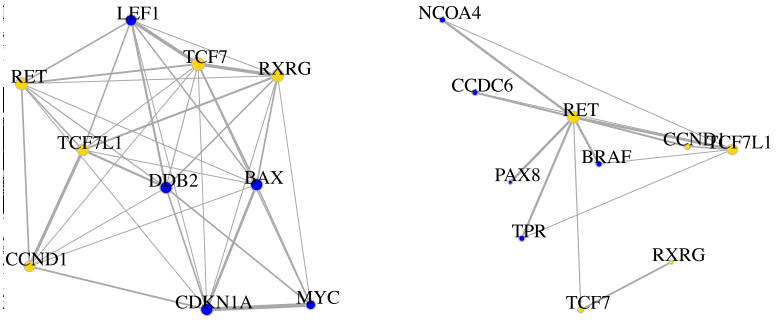

FIG 16. *Thyroid carcinoma (THCA), top 10 genes identified by the sum of strength.*

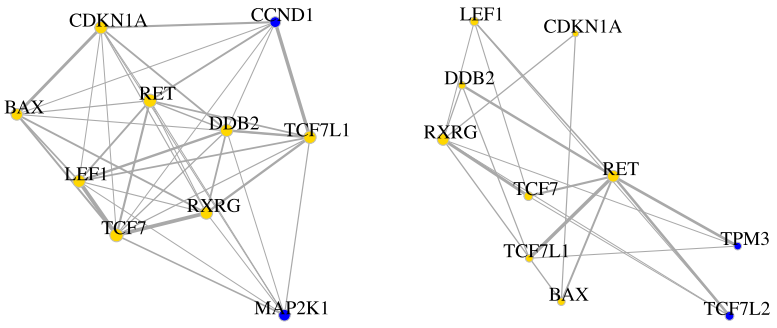

FIG 17. *Thyroid carcinoma (THCA), top 10 genes identified by the degree.*

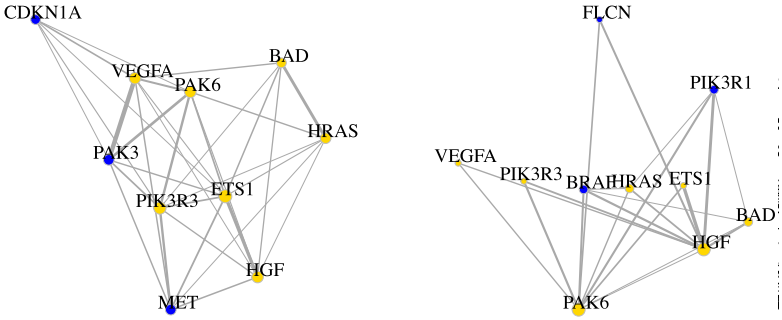

FIG 18. *Kidney renal clear cell carcinoma (KIRC), top 10 genes identified by the sum of strength.*

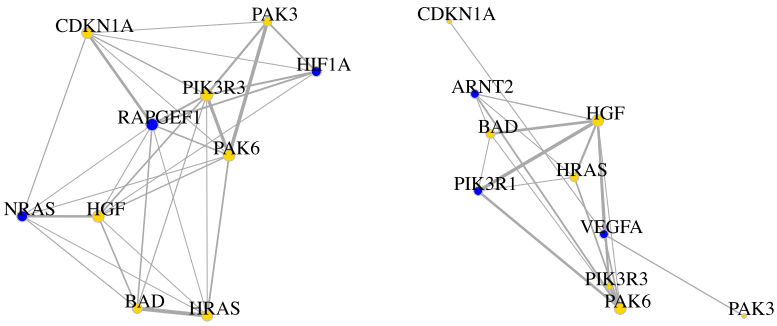

FIG 19. *Kidney renal clear cell carcinoma (KIRC), top 10 genes identified by the degree.*

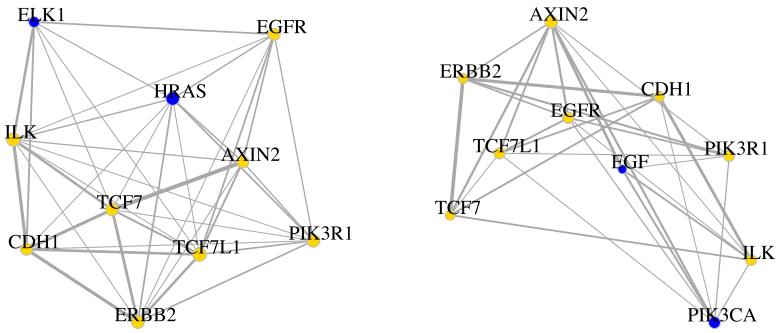

FIG 20. *Uterine corpus endometrial carcinoma (UCEC), top 10 genes identified by the sum of strength.*

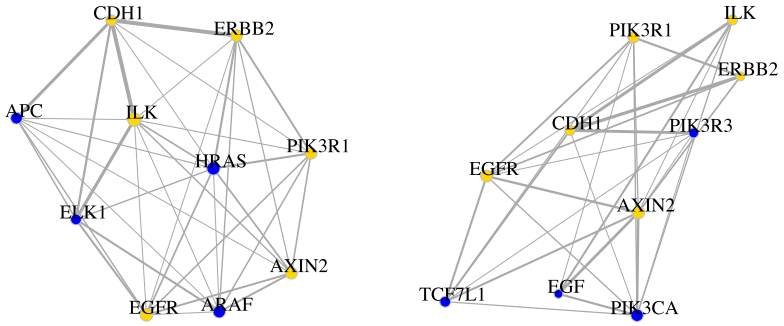

FIG 21. *Uterine corpus endometrial carcinoma (UCEC), top 10 genes identified by the degree.*

References

[1] BARZEL, B. and BARABÁSI, A.-L. (2013). Network link prediction by global silencing of indirect correlations. *Nature Biotechnology* **31** 720–725.

[2] CAI, T., LIU, W. and LUO, X. (2011). A constrained l_1 minimization approach to sparse precision matrix estimation. *Journal of the American Statistical Association* **106** 594–607. MR2847973

[3] DANAHER, P., WANG, P. and WITTEN, D. M. (2014). The joint graphical lasso for inverse covariance estimation across multiple classes. *Journal of the Royal Statistical Society* **76** 373–397. MR3164871

[4] DEMPSTER, A. P. (1972). Covariance selection. *Biometrics* **28** 157–175. MR3931974

[5] EISEN, M. B., SPELLMAN, P. T., BROWN, P. O. and BOTSTEIN, D. (1998). Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences* **95** 14863–14868.

[6] FAN, J., WANG, W. and ZHU, Z. (2016). A shrinkage principle for heavy-tailed data: High-dimensional robust low-rank matrix recovery. *arXiv*

preprint arXiv:1603.08315.

[7] Feizi, S., Marbach, D., Médard, M. and Kellis, M. (2013). Network deconvolution as a general method to distinguish direct dependencies in networks. *Nature Biotechnology* **31** 726–733.

[8] Friedman, J., Hastie, T. and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software* **33** 1–22.

[9] Hlouskova, A., Bielik, P., Bonczek, O., Balcar, V. and O, S. (2017). Mutations in AXIN2 gene as a risk factor for tooth agenesis and cancer: A review. *Neuro Endocrinology Letters* **38** 131–137.

[10] Hsieh, C.-J., Dhillon, I. S., Ravikumar, P. K. and Sustik, M. A. (2011). Sparse inverse covariance matrix estimation using quadratic approximation. In *Advances in Neural Information Processing Systems* **24**. MIT Press, Cambridge.

[11] Hsieh, C.-J., Sustik, M. A., Dhillon, I. S. and Ravikumar, P. (2014). QUIC: Quadratic approximation for sparse inverse covariance estimation. *Journal of Machine Learning Research* **15** 2911–2947. MR3277149

[12] Huang, T., Wang, L., Liu, D., Li, P., Xiong, H., Zhuang, L., Sun, L., Yuan, X. and Qiu, H. (2017). FGF7/FGFR2 signal promotes invasion and migration in human gastric cancer through upregulation of thrombospondin-1. *International Journal of Oncology* **50** 1501.

[13] Kim, H. S., Chon, H. J., Kim, H., Shin, S. J., Wacheck, V., Gruver, A. M., Kim, J. S., Rha, S. Y. and Chung, H. C. (2018). MET in gastric cancer with liver metastasis: The relationship between MET amplification and Met overexpression in primary stomach tumors and liver metastasis. *Journal of Surgical Oncology* **117** 1679–1686.

[14] Li, Y., Jin, K., van Pelt, G. W., Van, D. H., Yu, X., Mesker, W. E., Ten, D. P., Zhou, F. and Zhang, L. (2016). c-Myb enhances breast cancer invasion and metastasis through the Wnt/β-catenin/Axin2 pathway. *Cancer Research* **76** 3364.

[15] Liu, H., Mastriani, E., Yan, Z. Q., Yin, S. Y., Zheng, Z., Hong, W., Li, Q. H., Liu, H. Y., Wang, X. and Bao, H. X. (2016). SOX7 co-regulates Wnt/β-catenin signaling with Axin-2: Both expressed at low levels in breast cancer. *Scientific Reports* **6** 26136.

[16] MA, R.-R., AR, A.-V., LI, W.-C., P, B.-N., MP, G.-A., SE, F.-M. and J, S.-C. (2016). AXIN2 polymorphisms and its association with colorectal cancer in Mexican patients. *Genetic Testing and Molecular Biomarkers* **20**.

[17] Margolin, A. A., Nemenman, I., Basso, K., Wiggins, C., Stolovitzky, G., Favera, R. D. and Califano, A. (2006). ARACNE: An algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics* **7** S7.

[18] Markowetz, F. and Spang, R. (2007). Inferring cellular networks – a review. *BMC Bioinformatics* **8** S5.

[19] Meinshausen, N. and Bühlmann, P. (2006). High-dimensional graphs and variable selection with the Lasso. *Annals of Statistics* **34** 1436–1462. MR2278363

[20] Minsker, S. (2018). Sub-Gaussian estimators of the mean of a random matrix with heavy-tailed entries. *The Annals of Statistics* **46** 2871–2903. MR3851758

[21] Ravikumar, P., Wainwright, M. J., Raskutti, G. and Yu, B. (2008). High-dimensional covariance estimation by minimizing l_1-penalized log-determinant divergence. *Electronic Journal of Statistics* **5** 935–980. MR2836766

[22] Scheinberg, K., Ma, S. and Goldfarb, D. (2010). Sparse inverse covariance selection via alternating linearization methods. In *Advances in Neural Information Processing Systems* **23** 2101–2109. MIT Press, Cambridge.

[23] Shi, J., Zhao, J., Li, T. and Chen, L. (2019a). Detecting direct associations in a network by information theoretic approaches. *Science China Mathematics* **62** 823–838. MR3938498

[24] Shi, J., Zhao, J., Liu, X., Chen, L. and Li, T. (2019b). Quantifying direct dependencies in biological networks by multiscale association analysis. *IEEE Transactions on Computational Biology and Bioinformatics*.

[25] Stewart, G. W. and Sun, J. (1990). *Matrix perturbation theory*. Academic Press, Boston. MR1061154

[26] Stuart, J. M., Segal, E., Koller, D. and Kim, S. K. (2003). A gene-coexpression network for global discovery of conserved genetic modules. *Science* **302** 249–255.

[27] Waheed, A. and Shadduck, R. K. (1988). Effect of pH on binding and dissociation of colony-stimulating factor. *Proceedings of the Society for Experimental Biology & Medicine Society for Experimental Biology & Medicine* **187** 69.

[28] Wu, T. T. and Lange, K. (2008). Coordinate descent algorithms for lasso penalized regression. *Annals of Applied Statistics* **2** 224–244. MR2415601

[29] Yu, Y., Yu, X., Liu, H., Song, Q. and Yang, Y. (2018). miR-494 inhibits cancer-initiating cell phenotypes and reverses resistance to lapatinib by downregulating FGFR2 in HER2-positive gastric cancer. *International Journal of Molecular Medicine* **42** 998–1007.

[30] Yuan, M. and Lin, Y. (2007). Model selection and estimation in the Gaussian graphical model. *Biometrika* **94** 19–35. MR2367824

[31] Yuan, H., Xi, R., Chen, C. and Deng, M. (2017). Differential network analysis via lasso penalized D-trace loss. *Biometrika* **104** 755–770. MR3737302

[32] Yun, S. and Toh, K.-C. (2011). A coordinate gradient descent method for ℓ_1-regularized convex minimization. *Computational Optimization and Applications* **48** 273–307. MR2783427

[33] Zhang, T. and Zou, H. (2014). Sparse precision matrix estimation via lasso penalized D-trace loss. *Biometrika* **101** 103–120. MR3180660

[34] Zhang, X., Zhao, X.-M., He, K., Lu, L., Cao, Y., Liu, J., Hao, J.-K., Liu, Z.-P. and Chen, L. (2012). Inferring gene regulatory networks from gene expression data by path consistency algorithm based on conditional mutual information. *Bioinformatics* **28** 98–104.

[35] Zhao, S. D., Cai, T. T. and Li, H. (2015). Direct estimation of differential networks. *Biometrika* **2** 253–268. MR3215346

[36] Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society* **67** 768–768. MR2210692