Bioinformatics, 38(3), 2022, 770–777
https://doi.org/10.1093/bioinformatics/btab751
Advance Access Publication Date: 30 October 2021
Original Paper

Systems biology

Importance-Penalized Joint Graphical Lasso (IPJGL): differential network inference via GGMs

Jiacheng Leng [1,2] **and Ling-Yun Wu** [1,2,*]

[1]IAM, MADIS, NCMIS, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, China and [2]School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing 100049, China

*To whom correspondence should be addressed.

Associate Editor: Lenore Cowen

Received on March 12, 2021; revised on October 3, 2021; editorial decision on October 26, 2021; accepted on October 27, 2021

Abstract

Motivation: Differential network inference is a fundamental and challenging problem to reveal gene interactions and regulation relationships under different conditions. Many algorithms have been developed for this problem; however, they do not consider the differences between the importance of genes, which may not fit the real-world situation. Different genes have different mutation probabilities, and the vital genes associated with basic life activities have less fault tolerance to mutation. Equally treating all genes may bias the results of differential network inference. Thus, it is necessary to consider the importance of genes in the models of differential network inference.

Results: Based on the Gaussian graphical model with adaptive gene importance regularization, we develop a novel Importance-Penalized Joint Graphical Lasso method (IPJGL) for differential network inference. The presented method is validated by the simulation experiments as well as the real datasets. Furthermore, to precisely evaluate the results of differential network inference, we propose a new metric named APC2 for the differential levels of gene pairs. We apply IPJGL to analyze the TCGA colorectal and breast cancer datasets and find some candidate cancer genes with significant survival analysis results, including SOST for colorectal cancer and RBBP8 for breast cancer. We also conduct further analysis based on the interactions in the Reactome database and confirm the utility of our method.

Availability and implementation: R source code of Importance-Penalized Joint Graphical Lasso is freely available at https://github.com/Wu-Lab/IPJGL.

Contact: lywu@amss.ac.cn

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

As biological data becomes more accessible, researchers are looking to extract key information from the differences in data between normal and disease samples. Although many studies analyze differences between data and extract knowledge from a single-gene perspective (Love *et al.*, 2014; Robinson *et al.*, 2010), there is still a wealth of information to be mined in the differential data. Studies have shown that the co-expression pattern between a pair of genes may have changed significantly even in the absence of significant differences in the expression levels of either one of these genes (de la Fuente, 2010). Therefore, it is necessary to study biological mechanism from the perspective of gene interactions. And studies have shown that biological networks differ a lot in different states, different organizations, and even at different times (Bandyopadhyay *et al.*, 2010; Greene *et al.*, 2015; Ha *et al.*, 2015). Therefore, in order to study key genes which are the root cause of diseases, we need to focus on the differential networks. In general, we focus more on the differential expression between normal samples and disease samples, i.e. given two groups of gene expression data, our problem becomes how to build a differential co-expression network to explore the key genes by differential network analysis (DiNA). However, it is not simple, because there are always numerous false positives in the network inferred by various methods. One of the reasons is that many interactions in co-expression networks result from indirect influence (Cecchini *et al.*, 2018). To eliminate such indirect influence, many models have been developed. One kind of the models widely developed by researchers is the Gaussian graphical models (GGMs) because of its good property: if the entry of the precision matrix Θ_{ij} is zero, it is equivalent that variables i and j are conditionally independent, i.e. there is no co-expression interactions between i and j (Friedman *et al.*, 2008). Many researchers have conducted research based on the GGMs and proposed various methods to meet different assumptions.

Under usual circumstances, the number of genes $p \gg n$, the number of samples, which results in the sample covariance matrix not

being invertible. This means the precision matrix $\Theta = \Sigma^{-1}$ could not be computed directly, where Σ is the real covariance matrix. Graphical Lasso (GLasso) was a popular method to estimate the precision matrix Θ (Banerjee *et al.*, 2008; Friedman *et al.*, 2008; Yuan and Lin, 2007). This method added the Lasso penalty term on the likelihood function to obtain a sparse solution, guiding the development of the GGMs. Subsequently, many researchers have researched the differential networks based on this model. Danaher *et al.* (2014) first used the differential matrix as the penalty term assuming that the networks from two groups should be similar in the overall structure despite the relatively subtle differences and hence proposed Joint Graphical Lasso (JGL). On this basis, Mohan *et al.* (2014) proposed Perturbed-Node Joint Graphical Lasso (PNJGL) and Co-hub Node Joint Graphical Lasso (CNJGL) to solve node-based problems: the first one assumes that there are hub nodes in differential networks, and the second assumes that different states have the same hub nodes. The differential network can not only be obtained by the direct subtraction between networks of two states but can also be directly solved as a variable (He and Deng, 2019; Tang *et al.*, 2020; Tian *et al.*, 2016; Yuan *et al.*, 2017; Zhao *et al.*, 2014), on the premise of ensuring accuracy while greatly reducing the number of samples required.

However, the above methods have a common shortcoming, i.e. they put all genes in the same position. This is not the case in the real world. Different genes have different mutation probabilities. Even the same gene has different mutation rates in different tissues and locations (Scally, 2016). In general, the more important genes are less likely to mutate. This is because, on the one hand, if extremely important genes mutated in an individual, such as genes that regulate essential activities of life, the individual will have a high probability of not being sampled due to rapid death. On the other hand, studies have shown that even if a mutation occurs, important genes will be repaired preferentially, i.e. one of the reasons for the different probabilities of gene mutations is the DNA repair mechanism (Supek and Lehner, 2015). Some methods adjust the weights of different genes (Lyu *et al.*, 2018; Ou-Yang *et al.*, 2020; Zuo *et al.*, 2017). However, these methods need the prior information from the database. Once the prior information is insufficient, or the information in the database is a little biased, the results may also have certain deviations. Some method does not rely on prior information (Sulaimanov *et al.*, 2019), but it requires the time-consuming iterative generation of weights; moreover, it only considers a single state, which is not developed for inferring differential networks.

To address the above issues, we proposed a novel differential network inference method to meet the hypothesis that the important genes are less likely to mutate. In this proposed model, we add an importance penalty to each gene, and the penalties vary with the solution. The genes that are significantly altered between two states will be penalized if they also have large degrees in either one of two estimated networks. The comprehensive simulation experiments proved that the proposed Importance-Penalized Joint Graphical Lasso (IPJGL) method outperforms several state-of-the-art differential network inference methods based on GGMs, on the datasets simulated under realistic and reasonable assumptions. To better evaluate the results of differential network inference, we also propose a new metric named APC2 for the differential levels of gene pairs. We further applied IPJGL to the TCGA colorectal cancer and breast cancer datasets. The networks obtained by IPJGL showed some connections within the same family, as well as those already existed in BioGRID (Stark *et al.*, 2006) and Cytoscape Reactome database (Jassal *et al.*, 2020; Shannon *et al.*, 2003; Sidiropoulos *et al.*, 2017; Wu *et al.*, 2010), which confirmed the validity of our method. Moreover, in the enrichment analysis, our method successfully obtained a gene set among about 1000 genes, which was significantly enriched in the pathway most relevant to the disease. On a deeper level, through survival analysis, we discovered a gene SOST that may be closely related to colorectal cancer. It is an inhibitor of WNT signaling (Wang *et al.*, 2016), and WNT signaling happens to be a signaling pathway related to multiple cancers, especially colorectal cancer (Clevers, 2006). Research on these genes may reveal the pathological mechanism of colorectal cancer.

2 Materials and methods

2.1 JGL and its variants

Given the gene expression matrix of two sets of samples, normal samples, $X^{(1)} = \left(x_{ij}^{(1)}\right)_{n1\times p}$, and disease samples, $X^{(2)} = \left(x_{ij}^{(2)}\right)_{n2\times p}$, where p represents the number of genes, and n_1 and n_2 represent the number of normal samples and disease samples, respectively. Let $S^{(1)} = \left(s_{ij}^{(1)}\right)_{p\times p}$ and $S^{(2)} = \left(s_{ij}^{(2)}\right)_{p\times p}$ be the corresponding sample covariance matrices, $\Theta^{(1)} = \left(\Theta_{ij}^{(1)}\right)_{p\times p}$ and $\Theta^{(2)} = \left(\Theta_{ij}^{(2)}\right)_{p\times p}$ be the corresponding precision matrices. Also, we define the matrix norms, $||X||_1 = \sum_{i,j=1}^{p} |x_{ij}|$ and $||X||_F = \sqrt{\sum_{i,j=1}^{p} x_{ij}^2}$. For vectors $V = (v_1, v_2, \ldots, v_p)$, the vector norm is defined as $|V|_1 = \sum_{i=1}^{p} |v_i|$ and $|V|_2 = \sqrt{\sum_{i=1}^{p} v_i^2}$.

Under the assumption that the variable $x = (x_1, x_2, \ldots, x_p)$ are from the p-dimensional Gaussian distribution, Danaher *et al.* (2014) proposed the JGL model. They followed Friedman *et al.* (2008) and added a penalty term to make the precision matrices of the two states share an overall similar structure as follows:

$$\min_{\Theta^{(1)},\Theta^{(2)}\in\mathbb{S}_{++}^{p}} -L\left(\Theta^{(1)},\Theta^{(2)}\right) + \lambda_1 \sum_{k=1}^{2}\sum_{i\neq j}\left|\Theta_{ij}^{(k)}\right| + \lambda_2 P\left(\Theta^{(1)},\Theta^{(2)}\right),$$

where $\mathbb{S}_{++}^{p}$ is the space of p-dimensional positive definite matrices and $L\left(\Theta^{(1)},\Theta^{(2)}\right) = \sum_{i=1}^{2} n_i[\log(\det(\Theta^{(i)})) - \mathrm{tr}(S^{(i)}\Theta^{(i)})]$ is the likelihood function of the GGMs. The first penalty term is the sparsity penalty. The second penalty term is the similarity penalty, which is defined as

$$P\left(\Theta^{(1)},\Theta^{(2)}\right) = \sum_{i,j}\left|\Theta_{ij}^{(1)} - \Theta_{ij}^{(2)}\right|.$$

This penalty term penalizes edges that are different between two states, and its weight is λ_2.

On this basis, Mohan *et al.* (2014) proposed PNJGL and CNJGL, of which the first term is the same likelihood function, but the first penalty is slightly different in that they penalized the diagonal values, i.e. $\sum_{k=1}^{2}\sum_{i,j}\left|\Theta_{ij}^{(k)}\right|$, and the second penalty is modified as

$$P\left(\Theta^{(1)},\Theta^{(2)}\right) = \sum_{j=1}^{p}|V_j|_q$$

$$\text{s.t. } \Theta^{(1)} - \Theta^{(2)} = V + V^T$$

and

$$P\left(\Theta^{(1)},\Theta^{(2)}\right) = \sum_{j=1}^{p}\left|\begin{bmatrix} V^{(1)} \\ V^{(2)} \end{bmatrix}_j\right|_q$$

$$\text{s.t. } \Theta^{(i)} - \mathrm{diag}(\Theta^{(i)}) = V^{(i)} + \left(V^{(i)}\right)^T,\ i = 1, 2,$$

where $q = 1, 2$, respectively. They decomposed the matrix symmetrically to ensure that there are hub nodes in the corresponding network structure. PNJGL makes the hub structures exist in the differential matrix, and CNJGL supposes two states have the same hub nodes. In these models, all genes are treated equally, which does not tally with the actual situation.

2.2 Importance-Penalized Joint Graphical Lasso

We proposed the IPJGL model based on the following assumptions: (i) the expressions of each gene in the samples of each state follow a Gaussian distribution; (ii) the importance of genes is correlated with their degrees in the network and the important genes are less likely

to mutate; (iii) if a gene mutates, its existed or new interactions with other genes are dropped or established with certain probabilities, respectively.

The IPJGL model is given as follows:

$$\min_{\Theta^{(1)},\Theta^{(2)}\in\mathbb{S}^p_{++},V} -L\left(\Theta^{(1)},\Theta^{(2)}\right)+\lambda_1\left(||\Theta^{(1)}||_1+||\Theta^{(2)}||_1\right)$$

$$+\lambda_2\sum_{j=1}^{p}\left(\left|\Theta_j^{(1)}\right|_2^2+\left|\Theta_j^{(2)}\right|_2^2\right)|V_j|_q$$

$$s.t.\ \Theta^{(1)}-\Theta^{(2)}=V+V^T$$

where $\sum_{j=1}^{p}\left(\left|\Theta_j^{(1)}\right|_2^2+\left|\Theta_j^{(2)}\right|_2^2\right)|V_j|_q$ is the importance penalty and $w_j=\left|\Theta_j^{(1)}\right|_2^2+\left|\Theta_j^{(2)}\right|_2^2$ is the importance weight of gene j. We followed PNJGL and decomposed the differential matrix symmetrically so that V is not necessarily symmetric, and such decomposition will make the differential matrix easier to present the structure with hubs (Mohan *et al.*, 2014). When the precision matrices $\Theta^{(1)}$ and $\Theta^{(2)}$ are adjacency matrices, i.e. binary matrices, the importance weight w_j represents the sum of the degrees of gene j in each of the two networks. The vital genes are often hubs in the networks and therefore have large importance penalties. According to our assumptions, these genes are penalized to involve the differential edges between two states.

In brief, IPJGL adjusts the magnitude of its second penalty term according to the importance weights $\{w_j\}_{j=1,..p}$ of genes which are adaptively updated from the estimated networks of two states. With the differentiated importance penalties, IPJGL gives more chances to reveal those less important genes involved in the differential network. Figure 1 illustrates the influence of the importance penalty using a small toy example descirbed in Supplementary Figure S1. The normalized degrees in the estimated differential network are used to represent the inferred changes of gene activity between two states. When the importance weights of genes in IPJGL are bigger than those in PNJGL, the degrees of these genes may be repressed in IPJGL, e.g. the genes 9, 5 and 10. On the contrary, when the genes have smaller importance weights in IPJGL than PNJGL, the degrees of these genes may be increased in IPJGL, e.g. the genes 1, 7, 8 and 6. Therefore, the genes involved in the differential network but with low degrees in the networks of the two states, such as gene 6, have a higher chance to be identified (Fig. 1 and Supplementary Fig. S1), while these genes may be ignored by other algorithms such as PNJGL. It is worth noting that the Joint Graphical Lasso models are very complicated so that the higher or lower importance weights do not necessarily imply the repression or increase of gene activities in the differential network, respectively. For example, the degree of gene 4 is slightly increased while gene 2 totally vanishes in the result of IPJGL (Fig. 1). If there is strong evidence from the data that a gene is significantly changed between two states, the gene still can be revealed even with high importance penalty. If the change of interactions containing a gene is small, even if the importance penalty is greatly reduced, the result of the gene will not be affected too much. Therefore, the major difference between IPJGL and other general methods is that our method pays more attention to less important genes with the significant changes between the two states.

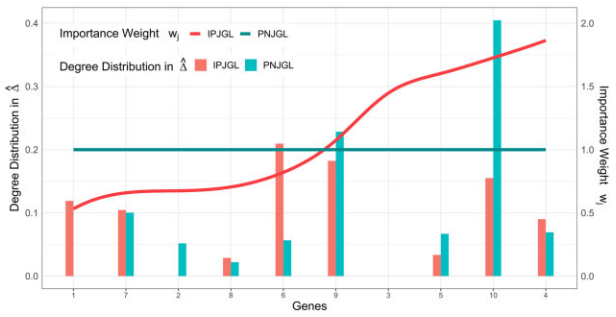

Fig. 1. Schematic diagram for the influence of importance penalty. The *X*-axis refers to genes sorted by their importance weights, the left *Y*-axis refers to the normalized degrees in the estimated differential network of each method, and the right *Y*-axis refers to the importance weights. The red curve represents the importance weight of each gene in IPJGL, while the green curve indicates the importance weight is identical for all genes in PNJGL. The relative degrees of the genes with high importance weights, such as 9, 5 and 10, are repressed. Consequently, other genes such as 1 and 6, have higher chance to be revealed

We used the ADMM (alternating direction method of multipliers) algorithm to solve the IPJGL model. For details of the algorithm, please refer to Section 1 of Supplementary Materials.

2.3 Performance metric

It is well known that the partial correlation coefficient matrix $P=(\rho_{ij})_{p\times p}$ can be obtained from the precision matrix

$$\rho_{ij}=\begin{cases}-\dfrac{\Theta_{ij}}{\sqrt{\Theta_{ii}\Theta_{jj}}}, i\neq j\\ 1,\ i=j\end{cases}.$$

Obviously $\Theta_{ij}=0\Longleftrightarrow\rho_{ij}=0$, so the corresponding network structures of the two matrices are the same. Moreover, the value of the partial correlation coefficient reflects the strength of the conditional correlation, therefore we consider using $\Delta_{true}=P^{(1)}-P^{(2)}$ as the real differential matrix, $I_{\{\Delta_{true}\neq 0\}}$ as the label, $\hat{\Delta}=\hat{P}^{(1)}-\hat{P}^{(2)}$ as the predicted differential matrix, and $|\hat{\Delta}|=\left(|\hat{\Delta}_{ij}|\right)_{p\times p}$ as the score to calculate AUPR (area under the precision–recall curve). We did not use AUC (area under the receiver operating characteristic curve) as an evaluation indicator because for the sparse network inference problem, the proportion of positive edges is extremely low, so there is not much information in the comparison of AUC.

Furthermore, we proposed a novel metric, the absolute value of the second principal component (APC2) to conduct gene-pair survival analysis, which can test whether the interaction between a pair of genes is significantly changed for individual samples. In general survival analysis, we often consider a single gene and group samples according to its expression level. However, many diseases are unlikely to be regulated by a single gene. For example, Combarros *et al.* (2009) have studied Alzheimer's disease and found that there are some carriers of apolipoprotein E4 who have an increased risk of developing the disease, but not all people who carry apolipoprotein E4 will get the disease. They, therefore, identified three genes that interact with apolipoprotein E4 by studying the interactions between genes, and these interactions have impacts on Alzheimer's disease. Therefore, it is necessary to consider the effect of the interaction of a pair of genes on the survival probability of patients.

Specifically, APC2 is the absolute value of the second principal component based on principal component analysis. We denote $(X^T)_{n\times 2}$ as the expression values of a pair of genes on n samples, which is centered on $(0,0)$. $X=U\Sigma V^T$ is the singular value decomposition of X, where $(U)_{2\times 2}$ is the eigenvector matrix of XX^T, $(\Sigma)_{2\times n}$ is the singular value matrix, $(V)_{n\times n}$ is the eigenvector matrix of X^TX. The principal components of X can be obtained by $Z=V\Sigma^T$, where Z is a $n\times 2$ matrix: the first column Z_1 is the first principal component (PC1) and Z_2 is PC2. Therefore, the APC2 of sample i is $|Z_{i2}|$. The first principal component can be considered as the most correlated direction in all samples, and the APC2 of a sample captures its deviation from the most correlated direction (Supplementary Figs S22–S24). In the gene-pair survival analysis, given a pair of genes, we grouped the samples according to their APC2 values. We divided samples into two groups: half of the samples with smaller APC2 are in the group 'high correlation', and the other is in the group 'low correlation'. Using gene-pair survival analysis based on APC2, we were able to identify some gene pairs that were significantly associated with patient survival time, whereas single-gene survival analysis on any of these genes alone did not yield significant results.

3 Simulation experiments

3.1 Data generation

Barabási and Albert (1999) showed that many biological networks have a scale-free structure and proposed a BA algorithm for simulating the scale-free networks. In this article, we first used the BA algorithm to generate a scale-free network with the adjacency matrix $(A^{(1)})_{p\times p}$, where p refers to the total number of genes. Then we generated a weight matrix $U^{(1)}$ of the same size as $A^{(1)}$ whose each entry is sampled from the piecewise uniform distribution $[-u_{max}, -u_{min}] \cup [u_{min}, u_{max}]$, and combine it with $A^{(1)}$ to obtain $\Theta^{(1)}$, i.e. $\Theta^{(1)} = A^{(1)} \odot U^{(1)}$, where $\odot$ refers to Hadamard product. This experiment takes $u_{max} = 0.8,\ u_{min} = 0.3$.

Second, we need to decide which genes are mutated. According to the assumptions, the importance of genes is correlated with their degrees in the network. Therefore, we divided the genes into three sets according to their degrees in the adjacency matrix $A^{(1)}$: (i) G_{High}: genes ranked by degrees in the top 20%; (ii) G_{Middle}: genes ranked by degrees in the top 20–40%; (iii) G_{Low}: genes ranked by degrees in the bottom 60%. To simulate the differential network, we need to select m mutated genes from the above three sets, where m is the preset total number of the mutated genes. To examine the performance of the DiNA algorithms under different conditions, we set three pure differential modes and two mixed differential modes, as shown in Table 1. For example, for diffmode5, we selected $0.1m$ genes from G_{High}, $0.4m$ from G_{Middle} and $0.5m$ genes from G_{Low} as mutated genes.

Next, we need to decide which interactions associated with the mutated genes are altered in the second network $\Theta^{(2)}$. To simulate the real biological process as much as possible, we divided the interaction $\Theta^{(2)}_{ij}$ which connects gene i and gene j after mutation into three categories (because it is an undirected graph, $\Theta^{(2)}_{ji} = \Theta^{(2)}_{ij}$), and used different ways to generate simulation data for each category: (i) unchanged: that is $\Theta^{(1)}_{ij} = \Theta^{(2)}_{ij}$; (ii) dropped: the original interaction is lost, i.e. $\Theta^{(1)}_{ij} \neq 0, \Theta^{(2)}_{ij} = 0$; (iii) connected: a new interaction is established, i.e. $\Theta^{(1)}_{ij} = 0, \Theta^{(2)}_{ij} \neq 0$. After we selected mutated gene k, each entry of $\Theta^{(1)}$ connected with gene k, i.e. $\Theta^{(1)}_{kj} \neq 0$, will lose the interaction with a certain probability, i.e. $\Theta^{(2)}_{kj} = 0$, as well, each entry of $\Theta^{(1)}$ unconnected with gene k, i.e. $\Theta^{(1)}_{kj} = 0$, will establish the interaction with a certain probability, i.e. $\Theta^{(2)}_{kj} \neq 0$. The probabilities of dropped and connected interactions are set separately since the probability of establishing a new interaction by mutation is generally much smaller than the probability of losing an existed interaction by mutation. We used $r_c,\ r_d$ to represent the probability of connected and dropped events, respectively, and set $r_c \in \{0.1, 0.2, 0.3\},\ r_d \in \{0.3, 0.5, 0.7\}$ for different experiment settings. After deciding which interactions need to be generated, each of these interactions will be sampled from the piecewise uniform distribution $[-u_{max}, -u_{min}] \cup [u_{min}, u_{max}]$. Up to this point, we generated two preliminary precision matrices for the two states, i.e. $\Theta^{(1)}, \Theta^{(2)}$.

Because the precision matrices need to be positive definite, we let $e_{min} = \min_{k\in\{1,2\}}(e^{(k)})$, where $e^{(k)}$ represents the eigenvalue set of $\Theta^{(k)}$, and add $(|e_{min}| + 0.1)$ to the diagonals of $\Theta^{(k)}$. So far, we have got the final precision matrices $\Theta^{(1)}, \Theta^{(2)}$. Finally, we used the inverse matrices of the obtained precision matrices, i.e. $(\Theta^{(1)})^{-1},\ (\Theta^{(2)})^{-1}$, as covariance matrices, respectively, to generate the gene expression matrices $X^{(1)}, X^{(2)}$ with zero means using the multivariate normal distribution, $\mathcal{N}(0, (\Theta^{(k)})^{-1})$, and then input them or their sample covariance matrices $S^{(1)}, S^{(2)}$ to the differential network inference methods.

Table 1. The distribution of mutated genes for five differential modes

diffmode	1	2	3	4	5
G_{High}	m	0	0	$0.5m$	$0.1m$
G_{Middle}	0	m	0	0	$0.4m$
G_{Low}	0	0	m	$0.5m$	$0.5m$

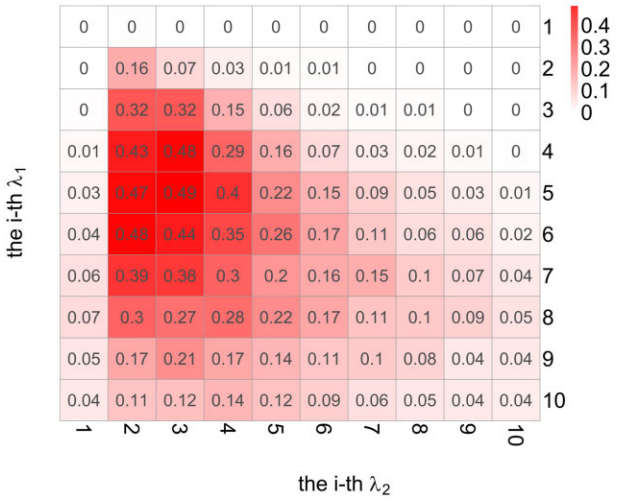

Fig. 2. Frequency heatmap of parameter settings for IPJGL ($q = 2$). *X*-axis refers to the ith value of λ_2 in the parameter space Λ_2, *Y*-axis refers to the ith value of λ_1 in the parameter space Λ_1. The max value of a cell is 1

3.2 Parameter selection

All the methods to be compared use two parameters $\lambda_1,\ \lambda_2$. Therefore, we adopted a more strict framework for parameter selection following Tian *et al.* (2016) and Mohan *et al.* (2014). To compare the performance under each parameter combination in a comprehensive and detailed manner, we first determined the parameter space Λ for each method. We looked for $\lambda_{1,max}$ (fixed $\lambda_2 = 0$) to make both $\hat{\Theta}^{(1)}$ and $\hat{\Theta}^{(2)}$ have at least 10% edges of the complete graph, and got λ_1 space $\Lambda_1 = \left\{0, \frac{\lambda_{1,max}}{10}, \frac{2\lambda_{1,max}}{10}, \ldots, \frac{9\lambda_{1,max}}{10}, \lambda_{1,max}\right\}$. Then, we looked for $\lambda_{2,max}$ (fixed $\lambda_1 = 0$) to make $\hat{\Delta}$ have at least 2% edges (max #edges of the true networks) of the complete graph, and got λ_2 space $\Lambda_2 = \left\{0, \frac{\lambda_{2,max}}{10}, \frac{2\lambda_{2,max}}{10}, \ldots, \frac{9\lambda_{2,max}}{10}, \lambda_{2,max}\right\}$. Finally, we got the overall parameter space $\Lambda = \Lambda_1 \times \Lambda_2$, which can cover most network sparsity. For each dataset $D_p(\text{diffmode}, r_c, r_d)$ generated by the different combination of simulation parameters, we repeated 20 times and then recorded the (λ_1, λ_2) once if its AUPR is greater than 0.95 times the maximum AUPR of all parameter combinations. Finally, we obtained the frequency heatmap and select the best parameter combination $(\lambda_1^*, \lambda_2^*)$ with the highest frequency as the default parameter of each method (Fig. 2 and Supplementary Figs S2–S4).

3.3 Simulation results

To show the effect of the proposed importance penalty, we compared IPJGL with several state-of-the-art methods JGL (Danaher *et al.*, 2014), CNJGL, PNJGL (Mohan *et al.*, 2014), because all these methods use similar GGMs to infer differential networks. We set $n \in \{25, 50, 100\},\ p = 100,\ q = 2,\ m = 10$ to satisfy the small samples assumption. It can be seen from the equation that when $q = 1$, PNJGL is equivalent to JGL, and in our experiments, $q = 2$ always outperformed $q = 1$. Therefore, to show the results more clearly, we only show the results with $q = 2$ in the article. We used the selected default parameter $(\lambda_1^*, \lambda_2^*)$ for each method to predict differential networks. Each method kept the top 1000 to 50 edges in the differential network as the prediction results, and we repeated 50 times in each case and calculated AUPR (Supplementary Figs S5–S19). To evaluate the overall performance of each method on different numbers of kept edges, we show the average AUPR of each method. Here, we only show the case $n = 100$ (Fig. 3), the rest we have put in

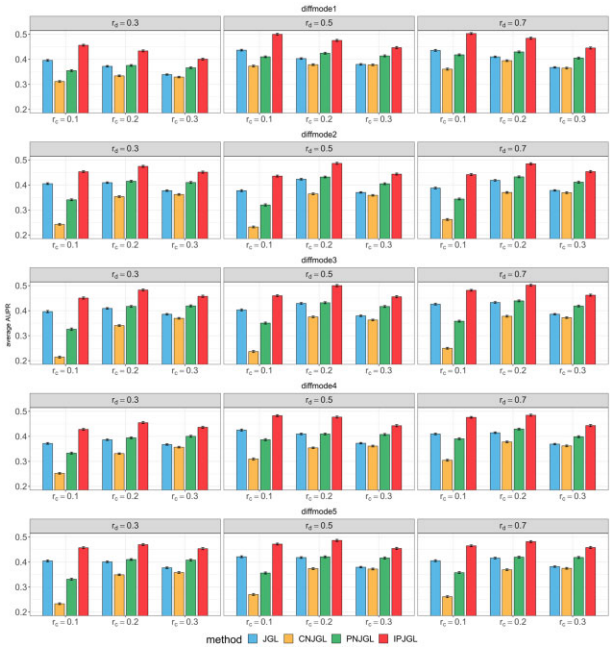

Fig. 3. Average AUPR of different methods on simulated datasets, $n = 100$. The Y-axis refers to the average AUPR and starts from 0.2.

Section 4 of Supplementary Materials (Supplementary Figs S20 and S21).

The results of simulation experiments show that all methods have better performance when the connected rate increases from 0.1 to 0.3 (Fig. 3 and Supplementary Figs S20 and S21). One of the possible reasons is the changes between two states become more significant therefore easier to detect. On the other hand, the performance of all methods is more sensitive to the connected rate than the dropped rate. This is because the original network is scale-free and sparse, so the interactions to be connected are much more than those to be dropped.

In all simulation experiments, IPJGL is uniformly superior to other methods (Fig. 3 and Supplementary Figs S20 and S21). The improvements of IPJGL over PNJGL and CNJGL are especially high when the connected rate is relatively low. This is because the changes between two states are too subtle to detect for the cases of low connected rates, while IPJGL can improve the accuracy by reducing the importance penalties of those genes with decreased degrees.

The performance of IPJGL on the datasets of different modes is consistent with our assumptions. The improvements of IPJGL over PNJGL and CNJGL are largest on the datasets of diffmode 2, 3 and 5, in which none or a few mutated genes are from the group of high importance. The improvements are the smallest on the datasets of diffmode 1, in which all mutated genes are from the group of high importance.

4 Applications in TCGA data

4.1 Datasets preparation

Colorectal cancer and breast cancer are common and plague the normal life of many people. To explore if our method can extract some insights from real-world data, we applied IPJGL to the transcriptome profiles downloaded from the TCGA project. The datasets of HTSeq counts are used in experiments. For colorectal cancer, which belongs to adenomas and adenocarcinomas, there are a total of 39 normal samples and 398 cancer samples according to annotations. For breast cancer, which belongs to ductal and lobular neoplasms, there are a total of 112 normal samples and 1066 cancer samples according to annotations.

Due to the huge number of genes, we performed the following screening. First, we selected 10 and 9 pathways related to KEGG's colorectal and breast cancer pathway including itself, which include 1034 and 958 genes, respectively. Next, we filtered out genes that did not express in at least 80% of the samples. Finally, we got 1009 and 953 genes, which are still relatively large and challenging gene sets for differential network inference problems, for colorectal and breast cancer, respectively. We used $\log_2(x+1)$ to scale the data, and then standardize it as the final input.

4.2 Comparison with single-gene-based analyses

As a widely used technique, differential expression analysis (DEA) typically focuses on a single gene and then compares the difference between the two states for each gene, e.g. in expression levels. DEA is not only fast but also simple, straightforward and easy to understand, yet sometimes some genes are not significantly differentially expressed but their patterns of interacting with other genes are dramatically altered (de la Fuente, 2010). Fortunately, DiNA can determine which interactions between genes have produced changes, and if most interactions of some gene have changed, that gene can be considered the main factor in the changes between states. To investigate the power of DiNA, we are particularly interested in the results that were only discovered by DiNA method but not found by the traditional method of DEA.

We first applied the DiNA methods with the default parameters to the real dataset to obtain the initial differential networks, and then keep the top k edges as the result (denoted as 'method$k$', e.g. the network generated by IPJGL with the top 500 edges is IPJGL500). Next, we used the 'DESeq2' package (Love *et al.*, 2014) in R to perform DEA. The inputs to 'DESeq2' are the gene expression matrices in normal and disease states. The output of 'DESeq2', which we used, is a list of P-values for the genes. If the P-value for a gene is <0.05, we consider the gene to be significantly differential in expression level between the two states. We then marked the insignificant genes (P-value > 0.05) with red color, which means those genes are less likely found by DEA but discovered by DiNA (Fig. 4).

To better exhibit the differential gene pairs discovered by DiNA, we used the metric APC2 as described in Section 2.3 to conduct gene-pair survival analysis on the gene pairs connected in differential networks. We downloaded the clinical data of TCGA and performed survival analysis using the R package 'survminer' on the single genes and the gene pairs. We fitted the data with the Kaplan–Meier model (Kaplan and Meier, 1958), which determines whether a difference in a factor leads to a difference in the probability of patient survival. For the single-gene survival analysis, the factor is the expression level of a gene. We divide the samples into two groups, with the gene expression levels out of the top 50% being called the high expression group and the bottom 50% being called the low expression group. For the gene-pair survival analysis, the procedure is the same except that the factor becomes the APC2 score of a gene pair. Gene-pair survival analysis can detect some genes which work together to regulate cancers, while single-gene survival analysis may ignore those genes. For example, by our proposed gene-pair survival analysis on colorectal cancer, DCC-PRKCG is significant with $P = 0.013$ while

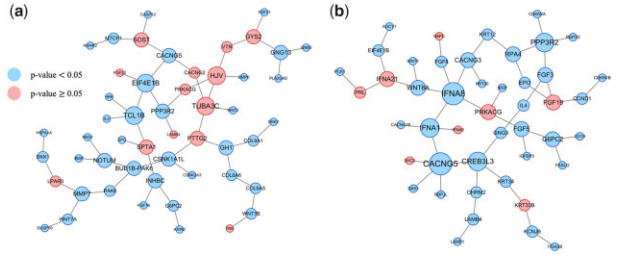

Fig. 4. The main component of IPJGL100 of colorectal cancer and breast cancer. (**a**) colorectal cancer and (**b**) breast cancer. The nodes and edges are extracted from the maximal connected component of the differential network with the top 100 edges identified by IPJGL. The node size represents the sum of the absolute weights of its adjacent interactions. The node color shows the result of DEA, i.e. blue nodes represent significantly differentially expressed genes ($P \leq 0.05$) and red nodes refer to the insignificant genes ($P > 0.05$), where P-values are calculated by DESeq2

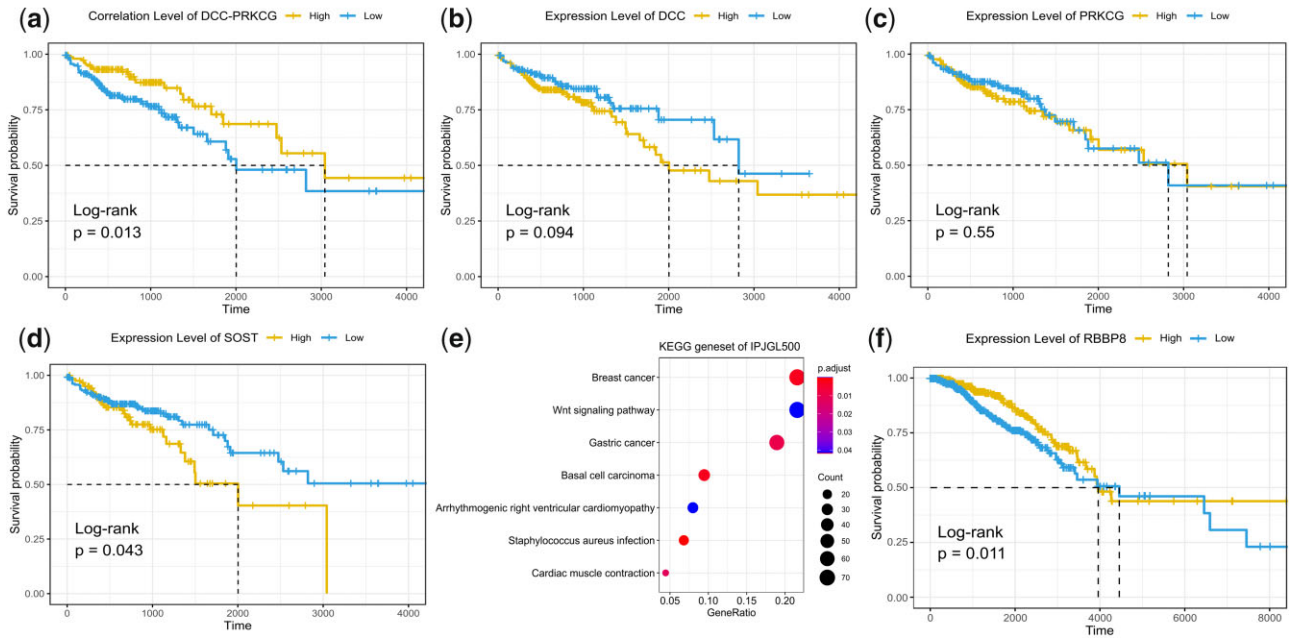

Fig. 5. Real data analysis results on colorectal and breast cancers. (**a**) Gene-pair survival analysis of DCC-PRKCG in colorectal cancer. The *X*-axis represents time in days. *Y*-axis represents the probability of surviving or the proportion of people surviving. The yellow line represents the high correlation group, and the blue line represents the low correlation group. The log-rank *P*-value indicates the significance of the difference between the two sample groups. If $P < 0.05$, we believe that the correlation level of this gene pair has a significant impact on the survival time of patients. (**b**) Single-gene survival analysis of DCC in colorectal cancer. The yellow line represents the high expression group, the blue line represents the low expression group. The log-rank *P*-value indicates the significance of the difference between the two sample groups. If $P < 0.05$, we believe that the expression level of this gene has a significant impact on the survival time of patients. (**c**) Single-gene survival analysis of PRKCG in colorectal cancer. (**d**) Single-gene survival analysis of SOST in colorectal cancer. (**e**) KEGG pathway enrichment analysis of genes in IPJGL500 of breast cancer. The *X*-axis (GeneRatio) represents the proportion of genes in the corresponding pathway that are covered by IPJGL500. The node size represents the number of genes in the intersection. The node color represents the adjusted *P*-value, where red indicates a smaller *P*-value. We successfully recover the breast cancer pathway using TCGA breast cancer data. The second significant pathway is the WNT signaling pathway, which is relative to a certain number of cancers. (**f**) Single-gene survival analysis of RBBP8 in breast cancer

either DCC or PRKCG is not significant ($P > 0.05$) (Fig. 5a–c). DCC functions as a tumor suppressor and is frequently mutated or downregulated in colorectal cancer (Kataoka *et al.*, 2000). In total, we get 92 significant gene pairs while 68 of them cannot be detected by single-gene survival analysis (Supplementary Table S6).

4.3 Colorectal cancer analysis

We compared IPJGL only with PNJGL for clarity of results on the real datasets because PNJGL is second only to our method in the previous simulation experiments. We kept the top 10 genes ranked by degree in method10, method50, method100, method500 (Table 2 and Supplementary Tables S3–S5). For example, top 10 genes in method500 for colorectal cancer are shown in Table 2, where genes that cannot be found by DEA but can be discovered by DiNA are marked in bold. We integrated all bold genes in these four tables as the key genes identified by the corresponding method. IPJGL got 8 key genes: CACNG2, PTTG2, TUBA3C, HJV, PPP3R2, SOST, TBL1Y, EIF4E1B, and PNJGL got 7 key genes: TUBA3C, EIF4E1B, SPTA1, CACNG2, TBL1Y, PPP3R2, FGF22. By applying single-gene survival analysis on these genes (Supplementary Fig. S25), we finally found a significant differential gene SOST (Fig. 5d) obtained only by IPJGL, which regulates the synthesis of sclerostin (Delgado-Calle *et al.*, 2017), but no direct relevant studies are suggesting its direct association with colorectal cancer. It may be a novel key gene associated with the molecular mechanisms of colorectal cancer.

To explore how SOST relates to colorectal cancer, we observed the adjacent genes of SOST in IPJGL500 and found that some interactions do appear in the related literature, such as 'SOST-WNT2', 'SOST-MMP7' (Kusu *et al.*, 2003; Semënov *et al.*, 2005). We further used Cytoscape's plugin ReatomeFIVz (Kusu *et al.*, 2003) to explore the relationship between SOST's neighbors and colorectal cancer. A lot of evidence was found to prove the relationship between its neighbors and colorectal cancer, which further validates the correlation between SOST and colorectal cancer (Section 6.3 of Supplementary Materials and Figs S26 and S27). We also compared the relevance of the gene sets in the differential networks obtained by different algorithms in the Reactome database. We found that genes picked by IPJGL have more interactions in Reactome than that by PNJGL when keeping the same edges as the output (Supplementary Fig. S28), which implies the gene set found by IPJGL is more closely related in the known colorectal cancer database.

Table 2. Comparison of the top 10 genes in 500-edges differential networks for colorectal cancer

	IPJGL500			PNJGL500		
Rank	Gene	Degree	*P*-value	Gene	Degree	*P*-value
1	TCL1B	15	0.030	**PPP3R2**	**19**	**0.635**
2	**HJV**	**14**	**0.735**	TCL1B	17	0.030
3	PAK6	12	0.010	INHBC	16	0.009
4	**TUBA3C**	**12**	**0.071**	DKK1	16	0.000
5	**PPP3R2**	**11**	**0.635**	WNT7A	16	0.000
6	**SOST**	**11**	**0.487**	**TUBA3C**	**14**	**0.071**
7	GH1	11	0.000	FGF20	14	0.000
8	INHBC	11	0.009	**FGF22**	**14**	**0.209**
9	**TBL1Y**	**11**	**0.921**	GH1	14	0.000
10	FGF8	11	0.000	PAK6	13	0.010

Note: The *P*-value is calculated using DESeq2. Genes with *P*-value >0.05 are marked in bold, which means they cannot be found by DEA but can be discovered by DiNA. If the degrees are the same, we order them by the sum of their interaction values.

4.4 Breast cancer analysis

We performed KEGG and GO enrichment analysis on genes in the differential networks with 50, 100 and 500 edges for breast cancer

(Supplementary Figs S29 and S30). Because our candidate gene set is selected from the KEGG pathways, we expected that the genes obtained by the algorithm should enrich the pathways related to breast cancer. The results confirmed that the genes obtained by IPJGL have strong functional enrichment properties, and only in the genes of IPJGL500, we obtained the expected results, as shown in Figure 5e. Although the dataset is relatively rough, the sample ratio is unbalanced, and the expression value may be biased, our method can still identify the corresponding pathway (breast cancer).

Further, we followed the same data analysis procedure in previous section to obtain key genes for breast cancer. We first retained all genes in the differential network method500 with P-value >0.05 in DESeq2. Then we performed the survival analysis for these genes to obtain the final set of key genes (Supplementary Fig. S31). Among them, there are 5 key genes obtained by IPJGL: SFRP5, FGF19, HES5, IL7R, RBBP8, and 3 key genes obtained by PNJGL: SFRP5, FGF19, HES5. The unique key gene RBBP8 identified by IPJGL is also called CTIP, whose low-level expression is associated with a poor prognosis in breast cancer (Wang *et al.*, 2016), as shown in our survival analysis (Fig. 5f).

5 Conclusion

In this article, we developed a novel algorithm IPJGL for differential network inference by adding an importance regularization term in the GGM. The comprehensive simulation experiments of comparing the proposed method with several state-of-the-art differential network inference methods based on GGMs confirmed the advantages and validity of the new algorithm. We also clarified the difference between DiNA and DEA in that DEA focuses on individual genes, whereas DiNA focuses on gene interactions, which are more complex but can also reveal more information. For this purpose, we proposed a novel metric APC2 for evaluating the interaction between a pair of genes for individual samples, which can be used in the downstream analyses of DiNA such as the gene-pair survival analysis. Finally, by applying the new method on the TCGA datasets, we obtained some significant cancer genes, such as SOST and RBBP8 for colorectal cancer and breast cancer, respectively, and found a lot of evidence related to cancers. In a nutshell, we hope our algorithm, simulation studies, and gene-pair interaction measurement can help biological and medical researchers find more candidate key genes about various kinds of cancers, while there is still a lot of work that needs to do based on the assumption of gene importance.

Acknowledgements

The results shown here are in part based upon data generated by the TCGA Research Network: https://www.cancer.gov/tcga.

Funding

This work was supported by the National Key Research and Development Program of China [2020YFA0712402] and the National Natural Science Foundation of China [11631014].

Conflict of Interest: none declared.

References

Bandyopadhyay,S. *et al.* (2010) Rewiring of genetic networks in response to DNA damage. *Science*, **330**, 1385–1389.

Banerjee,O. *et al.* (2008) Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data. *J. Mach. Learn. Res.*, **9**, 485–516.

Barabási,A.L. and Albert,R. (1999) Emergence of scaling in random networks. *Science*, **286**, 509–512.

Cecchini,G. *et al.* (2018) Improving network inference: the impact of false positive and false negative conclusions about the presence or absence of links. *J. Neurosci. Methods*, **307**, 31–36.

Clevers,H. (2006) Wnt/β-Catenin signaling in development and disease. *Cell*, **127**, 469–480.

Combarros,O. *et al.* (2009) Epistasis in sporadic Alzheimer's disease. *Neurobiol. Aging*, **30**, 1333–1349.

Danaher,P. *et al.* (2014) The joint graphical lasso for inverse covariance estimation across multiple classes. *J. R. Stat. Soc. Ser. B Stat. Methodol.*, **76**, 373–397.

de la Fuente,A. (2010) From 'differential expression' to 'differential networking'—identification of dysfunctional regulatory networks in diseases. *Trends Genet.*, **26**, 326–333.

Delgado-Calle,J. *et al.* (2017) Role and mechanism of action of sclerostin in bone. *Bone*, **96**, 29–37.

Friedman,J. *et al.* (2008) Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, **9**, 432–441.

Greene,C.S. *et al.* (2015) Understanding multicellular function and disease with human tissue-specific networks. *Nat. Genet.*, **47**, 569–576.

Ha,M.J. *et al.* (2015) DINGO: differential network analysis in genomics. *Bioinformatics*, **31**, 3413–3420.

He,S. and Deng,M. (2019) Direct interaction network and differential network inference from compositional data via lasso penalized D-trace loss. *PLoS One*, **14**, e0207731.

Jassal,B. *et al.* (2020) The Reactome pathway knowledgebase. *Nucleic Acids Res.*, **48**, D498–D503.

Kaplan,E.L. and Meier,P. (1958) Nonparametric estimation from incomplete observations. *J. Am. Stat. Assoc.*, **53**, 457–481.

Kataoka,M. *et al.* (2000) Aberration of p53 and DCC in gastric and colorectal cancer. *Oncol. Rep.*, **7**, 99–103.

Kusu,N. *et al.* (2003) Sclerostin is a novel secreted osteoclast-derived bone morphogenetic protein antagonist with unique ligand specificity. *J. Biol. Chem.*, **278**, 24113–24117.

Love,M.I. *et al.* (2014) Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biol.*, **15**, 550.

Lyu,Y. *et al.* (2018) Condition-adaptive fused graphical lasso (CFGL): an adaptive procedure for inferring condition-specific gene co-expression network. *PLoS Comput. Biol.*, **14**, e1006436.

Mohan,K. *et al.* (2014) Node-based learning of multiple Gaussian graphical models. *J. Mach. Learn. Res.*, **15**, 445–488.

Ou-Yang,L. *et al.* (2020) Differential network analysis via weighted fused conditional Gaussian graphical model. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, **17**, 2162–2169.

Robinson,M.D. *et al.* (2010) edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics*, **26**, 139–140.

Scally,A. (2016) The mutation rate in human evolution and demographic inference. *Curr. Opin. Genet. Dev.*, **41**, 36–43.

Semënov,M. *et al.* (2005) SOST is a ligand for LRP5/LRP6 and a Wnt signaling inhibitor. *J. Biol. Chem.*, **280**, 26770–26775.

Shannon,P. *et al.* (2003) Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res.*, **13**, 2498–2504.

Sidiropoulos,K. *et al.* (2017) Reactome enhanced pathway visualization. *Bioinformatics*, **33**, 3461–3467.

Stark,C. *et al.* (2006) BioGRID: a general repository for interaction datasets. *Nucleic Acids Res.*, **34**, D535–D539.

Sulaimanov,N. *et al.* (2019) Inferring gene expression networks with hubs using a degree weighted Lasso approach. *Bioinformatics*, **35**, 987–994.

Supek,F. and Lehner,B. (2015) Differential DNA mismatch repair underlies mutation rate variation across the human genome. *Nature*, **521**, 81–84.

Tang,Z. *et al.* (2020) A fast iterative algorithm for high-dimensional differential network. *Comput. Stat.*, **35**, 95–109.

Tian,D. *et al.* (2016) Identifying gene regulatory network rewiring using latent differential graphical models. *Nucleic Acids Res.*, **44**, e140.

Wang,J. *et al.* (2016) Loss of CtIP disturbs homologous recombination repair and sensitizes breast cancer cells to PARP inhibitors. *Oncotarget*, **7**, 7701–7714.

Wu,G. *et al.* (2010) A human functional protein interaction network and its application to cancer data analysis. *Genome Biol.*, **11**, R53.

Yuan,H. *et al.* (2017) Differential network analysis via lasso penalized D-trace loss. *Biometrika*, **104**, 755–770.

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

Zhao,S.D. *et al.* (2014) Direct estimation of differential networks. *Biometrika*, **101**, 253–268.

Zuo,Y. *et al.* (2017) Incorporating prior biological knowledge for network-based differential gene expression analysis using differentially weighted graphical LASSO. *BMC Bioinform.*, **18**, 99.