Condition-adaptive fused graphical lasso (CFGL): An adaptive procedure for inferring condition-specific gene co-expression network

Yafei Lyu, Lingzhou Xue, Feipeng Zhang, Hillary Koch, Laura Saba, Katerina Kechris, Qunhua Li

1 Bioinformatics and Genomics, the Huck Institute of the Life Science, Pennsylvania State University, State College, Pennsylvania, United States of America, 2 Department of Statistics, Pennsylvania State University, State College, Pennsylvania, United States of America, 3 Department of Pharmaceutical Sciences, Skaggs School of Pharmacy and Pharmaceutical Sciences, University of Colorado Anschutz Medical Campus, Aurora, Colorado, United States of America, 4 Department of Biostatistics and Informatics, Colorado School of Public Health, University of Colorado Anschutz Medical Campus, Aurora, Colorado, United States of America

* qunhua.li@psu.edu

Abstract

Co-expression network analysis provides useful information for studying gene regulation in biological processes. Examining condition-specific patterns of co-expression can provide insights into the underlying cellular processes activated in a particular condition. One challenge in this type of analysis is that the sample sizes in each condition are usually small, making the statistical inference of co-expression patterns highly underpowered. A joint network construction that borrows information from related structures across conditions has the potential to improve the power of the analysis. One possible approach to constructing the co-expression network is to use the Gaussian graphical model. Though several methods are available for joint estimation of multiple graphical models, they do not fully account for the heterogeneity between samples and between co-expression patterns introduced by condition specificity. Here we develop the condition-adaptive fused graphical lasso (CFGL), a data-driven approach to incorporate condition specificity in the estimation of co-expression networks. We show that this method improves the accuracy with which networks are learned. The application of this method on a rat multi-tissue dataset and The Cancer Genome Atlas (TCGA) breast cancer dataset provides interesting biological insights. In both analyses, we identify numerous modules enriched for Gene Ontology functions and observe that the modules that are upregulated in a particular condition are often involved in condition-specific activities. Interestingly, we observe that the genes strongly associated with survival time in the TCGA dataset are less likely to be network hubs, suggesting that genes associated with cancer progression are likely to govern specific functions or execute final biological functions in pathways, rather than regulating a large number of biological processes. Additionally, we observed that the tumor-specific hub genes tend to have few shared edges with normal tissue, revealing tumor-specific regulatory mechanism.
Author summary

Gene co-expression networks provide insights into the mechanism of cellular activity and gene regulation. Condition-specific mechanisms may be identified by constructing and comparing co-expression networks of multiple conditions. We propose a novel statistical method to jointly construct co-expression networks for gene expression profiles from multiple conditions. By using a data-driven approach to capture condition-specific co-expression patterns, this method is effective in identifying both co-expression patterns that are specific to a condition and that are common across conditions. The application of this method to real datasets reveals interesting biological insights.

Introduction

Gene co-expression network analysis is a useful tool for studying the complex regulatory machinery in organisms [1][2][3][4]. When the gene expression profiles under multiple conditions are available, comparing co-expression networks across conditions could reveal co-expression patterns that are common across conditions and those that are unique to a condition [5][6][7][8][9], providing insights on how genes work together to regulate biological processes under different conditions. It has been demonstrated that complex diseases are likely to be regulated by condition-specific mechanisms while condition-specific hub genes are likely to be drug targets [10][11][12].

The Gaussian graphical model and its variants have been widely used for studying biological networks [13][14][15][16][17][18][19]. This method models the joint distribution of a set of variables and characterizes the conditional dependence between each pair of variables given all the other variables through the precision matrix (a.k.a. inverse covariance matrix) of the joint distribution [20]. Unlike co-expression models based on marginal correlation, e.g. WGCNA [21], which do not distinguish the direct and indirect (e.g. through intermediate genes) relationship between genes, the direct relationship between a pair of genes can be inferred from the conditional independence estimated from the Gaussian graphical model. Many algorithms have been proposed to obtain a sparse estimate for the precision matrix, for example, graphical lasso [22] and neighborhood selection [23]. These algorithms make it possible to construct gene co-expression networks using graphical models. A graph generated from this estimate, where genes are represented as nodes and entries in the estimated precision matrix as edges, provides a useful tool for visualizing the relationships between genes and for generating biological hypotheses.

In a multi-condition gene expression study, the co-expression profiles across conditions typically are related, for example, due to shared pathways in different tumor subtypes, or common regulatory mechanisms for housekeeping genes in different tissues. A joint analysis that borrows information across conditions potentially can reveal common structures and increase the power of statistical inference, which is especially useful when the sample sizes are small. Recently, several methods have been proposed to jointly analyze multiple graphical models. Meinshausen et al. [24] incorporated a non-convex hierarchical group lasso penalty into the graphical lasso to encourage common 0’s (i.e. absence of edges) in the precision matrix across conditions. Danaher et al. [6] proposed a joint graphical lasso model by adding an additional convex penalty to the graphical lasso objective function. They proposed two choices for the convex penalty: a group penalty that encourages a shared pattern of sparsity and a fused lasso penalty that encourages similarities in both network sparsity and edge weights.
Despite their differences, these methods encourage similarities equally across all edges and all conditions. This inherently assumes that the similarity across conditions is similar for all edges and that the precision matrices in all conditions are equally similar to each other. For gene co-expression networks across different conditions, however, both assumptions are violated due to the heterogeneity across genes and across conditions. First, edges in the networks often have different levels of conservation across conditions. For example, in a network consisting of multiple pathways, the pathways involving basic cellular functions tend to be more conserved across tissues than those involving tissue-specific functions. Second, when there are multiple conditions, some conditions may be more similar to each other than others. For example, tissues with the same embryonic origin may have more similar pathways than those with different origins. More recently, several methods have been proposed to allow more structural heterogeneity in joint estimation. Zhu et al. [25] introduced a non-convex truncated $l_1$ penalty on the pairwise differences between the precision matrices to encourage elementwise clustering of similar entries across conditions. To incorporate external information on shared subgraphs across conditions, Ma et al. [26] grouped edges shared across conditions based on external information and extended the neighborhood selection method to a joint analysis with the proposed penalty. To handle heterogeneity in similarities across conditions, Seagusa et al. [27] proposed a Laplacian shrinkage penalty to incorporate the pairwise distance between conditions, and proposed using hierarchical clustering to obtain the pairwise distance when it is unknown a priori. While these methods improve the flexibility in estimation, they do not completely address the issues in studying condition-specific co-expression networks. For example, though the approach in Zhu et al. [25] allows abrupt elementwise difference across conditions, it still implicitly assumes that the majority of edges are common across conditions and penalizes condition-specificity. The approach in Ma et al. [26] relies on the availability and the quality of external information, which is still limited for gene co-expression relationships. The approach in Seagusa et al. [27] uses external information or hierarchical clustering to define the weighted subpopulation network and only partially addresses the issue of condition specificity.

In this work, we propose an adaptive approach to simultaneously addressing condition specificity and heterogeneity across conditions in the estimation of multiple co-expression networks. Our strategy is to incorporate a binary weight matrix that contains information on whether or not an edge is common between conditions in the fused graphical lasso framework. We propose a strategy to learn this matrix adaptively from the data based on a test for differential co-expression, though it can also be obtained from external sources. The incorporation of this matrix not only accounts for the difference between condition-common edges and condition-specific edges but also makes the estimation adaptive to the distance between different conditions. In this way, one can borrow information across conditions for common edges, while estimating differential edges in a condition-specific manner. We provide a computationally efficient implementation using the alternating direction method of multipliers (ADMM) algorithm. Our simulations show that this method generates more accurate results in both edge detection and edge weight estimation. We applied our method to a rat multi-tissue dataset and a TCGA breast cancer dataset (TCGA BRCA) and obtained interesting biological insights.

Results

Review of graphical lasso model and fused graphical lasso model

We first briefly describe the Graphical Lasso (GL) method [22] and the Fused Graphical Lasso (FGL) method [6]. Suppose the gene expression profiles are available across K conditions,
where conditions are, for example, different tissues or disease statuses. Denote the gene expression levels $\mathbf{Y}^{(k)}$ for the condition $k$, $k = 1, 2, \ldots, K$, as a $n_k \times p$ matrix, where $p$ is the number of genes, which is common across all conditions, and $n_k$ is the number of observations, which can vary across conditions. Suppose that gene expression levels within each condition, $y_{1}^{(k)}, y_{2}^{(k)}, \ldots, y_{n_k}^{(k)} \in \mathbb{R}^p$, are identically drawn from $N(\mu_k, \Sigma_k)$, where $\mu_k \in \mathbb{R}^p$ and $\Sigma_k$ is a positive definite $p \times p$ matrix. Then zero entries in the precision matrix $\Sigma_k^{-1}$ correspond to the pairs of genes that are conditionally independent given all other genes in the dataset. Based on the precision matrix $\Theta^{(k)} = \Sigma_k^{-1}$, a gene co-expression network can be constructed by representing the genes as nodes and conditional dependencies as edges in a graph.

The most direct way to analyze such data is to estimate $K$ individual graphical models separately. We can use the graphical lasso method to compute a separate $l_1$ penalized estimator of $\Sigma_k^{-1}$ for each condition by solving,

$$\text{maximize}_{\{\Theta^{(k)}\}} \left( \log \{ \det(\Theta^{(k)}) \} - \text{tr}(S^{(k)} \Theta^{(k)}) - \lambda_1 \| \Theta^{(k)} \|_1 \right),$$

where $S^{(k)} = (\mathbf{Y}^{(k)})^T \mathbf{Y}^{(k)}/n_k$ is the empirical covariance matrix of $\mathbf{Y}^{(k)}$, $\lambda_1 \| \Theta^{(k)} \|_1$ is a penalty term with non-negative tuning parameter $\lambda_1$ and $\| \Theta^{(k)} \|_1$ is the $l_1$ norm of $\Theta^{(k)}$. However, when the conditions are related, separate estimation ignores the common structure shared across conditions and can also mask differences critical in understanding condition-specificity in the co-expression pattern.

To address this issue, Danaher et al. [6] developed a fused graphical lasso model to jointly estimate multiple graphical models from related conditions. This model incorporates the generalised fused lasso penalty $P(\{\Theta\})$ [28] to the log-likelihood,

$$l(\{\Theta\}) = \sum_{k=1}^{K} n_k \{ \log \{ \det(\Theta^{(k)}) \} - \text{tr}(S^{(k)} \Theta^{(k)}) \} - P(\{\Theta\}),$$

such that information can be borrowed across conditions. The penalty $P(\{\Theta\})$ is a convex penalty with two terms,

$$P(\{\Theta\}) = \lambda_1 \sum_{k=1}^{K} \sum_{(i,j) \neq (i,j)} |\theta_{i,j}^{(k)}| + \lambda_2 \sum_{k=1}^{K} \sum_{(i,j) \neq (i,j)} \sum_{(i',j') \neq (i,j')} |\theta_{i,j}^{(k)} - \theta_{i',j'}^{(k')}|,$$

where $\lambda_1$ and $\lambda_2$ are non-negative tuning parameters, and $\theta_{i,j}^{(k)}$ is the $(i,j)$-th element of the matrix $\Theta^{(k)}$. The first term, which is the lasso penalty in GL [23][22], is applied to the off-diagonal entries of the $K$ precision matrices to encourage sparsity. The second term, which is the fused lasso penalty [28], is applied to the differences between elements of each pair of precision matrices to encourage similarity between conditions. A large $\lambda_2$ leads to similar edge patterns across conditions. It has been shown that FGL outperforms GL when conditions are related [6].

**Condition-adaptive fused graphical lasso**

While borrowing strength across conditions is helpful for enlarging effective sample sizes, differences in co-expression patterns are present between different conditions. For example, if one were studying tumor-specific co-expression by analyzing two subtypes of tumor tissues and a normal tissue jointly (Fig 1), some edges in the co-expression networks may be common across all three conditions, while others may be specific to one condition or both tumor subtypes. A primary interest of the study would be to identify the tumor-specific or subtype-specific edges. If FGL is used to construct the co-expression network, it would encourage similarities among all edges across all conditions equally by imposing a constant penalty parameter. This has two drawbacks. First, it does not distinguish between shared edges and
those unique to a condition, thus condition-specificity of edges is not preserved. Second, it imposes an equal amount of similarities across all pairs regardless of whether the pair consists of two tumor subtypes or a tumor tissue and a normal tissue. This is problematic as the two tumor subtypes are likely to be more similar to each other than to the normal tissue.

To address these issues, we extend the fused graphical lasso method to incorporate condition-specificity in the integration of networks across conditions. Our strategy is to add a binary screening matrix $W^{(kk_0)}$ to the fused lasso penalty as follows,

$$P(\Theta) = \lambda_1 \sum_{k=1}^{K} \sum_{i \neq j} |\theta_{ij}^{(k)}| + \lambda_2 \sum_{k<k'} \sum_{i \neq j} W^{(kk_0)}_{ij} |\theta_{ij}^{(k)} - \theta_{ij}^{(k')}|,$$

where $W^{(kk_0)}_{ij}$ is the $(i,j)$-th element of $W^{(kk_0)}$ with

$$W^{(kk_0)}_{ij} = \begin{cases} 1, & \text{if } \theta_{ij}^{(k)} \text{ and } \theta_{ij}^{(k')} \text{ are nondifferential between conditions } k \text{ and } k' \\ 0, & \text{if } \theta_{ij}^{(k)} \text{ and } \theta_{ij}^{(k')} \text{ are differential between conditions } k \text{ and } k' \end{cases}.$$

The matrix $W^{(kk_0)}$ controls whether similarity should or should not be encouraged between each pair of condition for each edge. It allows different edges to be penalized differently, and also allows the penalties for different pairs of conditions to vary according to the distance between the conditions. In doing so, one can borrow strength across conditions for estimating common edges, while allowing differential edges to be estimated in a condition-specific way. Therefore, we call our method condition-adaptive fused graphical lasso (CFGL). Fig 2 illustrates the workflow of our method.

Fig 1. An illustrative example of condition-specific co-expression patterns. Yellow: module common across all three conditions; Blue: tumor-specific module shared across the tumor subtypes; Unshaded: module specific to each condition. The co-expression networks in the two tumor subtypes are more similar to each other than to normal tissue.

https://doi.org/10.1371/journal.pcbi.1006436.g001
Determine the screening matrix $W^{(kk')}$

The screening matrix can be obtained using prior knowledge, learning directly from the data, or a combination of both strategies. To determine the screening matrix using prior knowledge, one may extract information on co-expression regulation from public databases, such as the KEGG pathway database [29], COXPRESdb [30] or MSigDB [31]. For example, if a pathway is known to be conserved across tissues [5][32], one may specify the corresponding elements in $W^{(kk')}$ as 1 to reflect the conservation of co-expression regulation.

However, it is difficult to construct the entire screening matrix solely based on prior information, because the gene relationships in the databases often are not provided in a condition-specific way (e.g. not available for a specific disease type) and the genes of interest may not be included. Therefore, we propose a data-driven strategy to estimate the screening matrix $W^{(kk')}$ from the data. As $W^{(kk')}$ reflects the status of differentiation between a pair of conditions, we determine $W^{(kk')}$ by identifying differential entries between the precision matrices of the two conditions, $\Sigma^{-1}_k$ and $\Sigma^{-1}_0$, through a hypothesis test. If the test determines that the entry $ij$ is differential, we set $w^{(kk')}_{ij} = 0$, otherwise we set $w^{(kk')}_{ij} = 1$. As $w^{(kk')}_{ij}$ is binary, this approach is equivalent to using a $l_0$ penalty to determine the support of the condition-specific edges. It is
somewhat analogous to the Sure Independent Screening procedure for feature selection methods such as the lasso, Dantzig selector, and SCAD [33], where an elementwise screening is first performed to reduce the dimension from ultra-high to moderate before variable selection.

Here we test for differentiation using the test proposed by Xia et al.[34]. This method tests for a difference between a pair of precision matrices and reports differential entries in the precision matrices with proper false discovery rate (FDR) control. It directly estimates the difference between precision matrices, bypassing the estimation of the individual precision matrices. Other tests for differential entries are available[35][36], but we selected Xia’s test as it has been shown to provide more accurate estimates than the tests that require separate estimation of precision matrices due to leveraging information on the sparsity of the difference between precision matrices [37]. To avoid falsely imposing similarity for edges that are moderately differential, we use a relaxed FDR threshold in the test to encourage similarity only to the edges that are obviously non-differential across conditions.

**Parameter estimation and selection of penalty parameters**

Similar to FGL and other penalty-based methods, this model can be estimated using the ADMM algorithm. We used BIC to guide the selection of penalty parameters. In the real data application, when the sample size is reasonably large to afford subsampling, we performed an additional stability selection [38] step. Instead of constructing networks using all the samples, the stability selection procedure constructs networks for a large set of subsamples generated from the original data and keeps only the edges that frequently occur across subsamples to obtain robust edges. Details on the stability selection procedure can be found in Methods.

**Simulation studies**

We used simulation studies to evaluate the performance of our method and compare it to FGL and GL. We first considered the two-condition scenario, evaluating the performance of these methods at different levels of differentiation between conditions. Then, we increased the complexity by introducing a third condition and allowing the level of differentiation to vary across all three conditions.

In the first set of simulations, we generated the gene expression profiles from a co-expression network of 400 genes for 2 conditions. The network consists of 8 co-expression modules, each of 50 genes. To generate different levels of differentiation between conditions, we simulated four scenarios (S1-S4) with a progressively increasing number of differential edges between conditions, where the networks in the two conditions are identical in S1 and are different at various levels in S2-S4. While S1 is extremely rare in practice, it exactly follows the model assumptions of FGL and thus illustrates the methods performance under conditions ideal to FGL. Two samples size (50, 100) are considered for each scenario.

To simulate a network, we first simulated its constituent modules. To create different levels of differentiation, three types of modules were simulated: (1) identical network structure and identical edge weights between conditions (II), (2) identical network structure but different edge weights between conditions (ID), and (3) different network structures and different edge weights between conditions (DD). We then combined these modules in various configurations to achieve the desired level of differentiation for the networks in different scenarios. In all scenarios, the 8 modules are evenly split into two groups, each of which consists of 4 modules of the same type. The configurations of modules in these scenarios are summarized in Table 1. Detailed information on the data generating process are in Methods.

For each simulation, we constructed the co-expression network using our method, FGL, and GL. To evaluate how the accuracy of the estimated screening matrix affects the
performance of our method, we also included a version of CFGL with the true screening matrix, which is labeled as CFGL-oracle (CFGLO) (see the Methods section). We compared the performance of these methods based on the estimation of network topology and edge weight across a grid of $\lambda_1$ and $\lambda_2$. The accuracy of estimated network structure was evaluated according to the network topology, i.e. the presence or absence of edges. Specifically, we compared the estimated network topology with the true topology, then computed the sensitivity and specificity of the edge detection. If an edge is present in the true network but missed in the estimated one (i.e. estimated edge weight = 0), then it was counted as a false negative. If an edge is absent in the true network but identified in the estimated one (i.e. estimated edge weight $>$ 0), then it was counted as a false positive. The accuracy of edge weight estimation was assessed by computing the sum of squared error (SSE) between the estimated edge weight and the true precision matrix. We plotted the ROC curve for edge detection and the SSE for edge weight estimation at a varying level of $\lambda_1$ with $\lambda_2$ fixed at the value that achieves the minimal BIC value ($\lambda_2 = 0.15$ for $n = 50$ and $\lambda_2 = 0.10$ for $n = 100$). Because edges detected at a low false positive rate are of primary interest, we computed the partial area under the curve (pAUC) from the ROC curve for the range of FPR $<$ 0.05.

Fig 3 shows the results at $n = 50$. In all scenarios, our approach (CFGL) had a higher partial AUC (pAUC: S1: 0.711, S2: 0.671, S3: 0.688, S4: 0.620) than GL (pAUC: S1: 0.583, S2: 0.594, S3: 0.588, S4: 0.590), and also a lower SSE. The gain is more apparent when the two conditions are relatively similar (S1-S3). This is because data integration improves the accuracy of edge detection, especially when networks are similar between conditions. The advantage is especially obvious when $n = 50$ (S1 Table and S2 Fig for $n = 100$), as this small sample size is likely not enough to support accurate estimation with GL based on the samples from a single condition. FGL performs well in these scenarios too (pAUC: S1: 0.714, S2: 0.609, S3: 0.660); however, when the two conditions are fairly different (S4), FGL performs worse than GL (pAUC: 0.578 vs. 0.590). Compared with FGL, our method has a higher AUC and an apparently lower SSE in all scenarios with between-condition differences (S2-S4). Even in the scenario without between-condition differences (S1), i.e. the ideal setting for FGL, our method is still competitive: it has an almost identical ROC curve as FGL and a slightly higher SSE than FGL. In practice, it is much more common to encounter S2-S4 than S1, as the networks of two different conditions are likely to be different. CFGLO has the best AUC (pAUC: S1: 0.714, S2: 0.717, S3: 0.715, S4: 0.723) and SSE among all the methods, suggesting that the performance of CFGLO can be further improved by improving the estimation of screening matrix, for example, by incorporating external information. We also reported simulation results under several other $\lambda_2$’s. The results are similar and can be found in S1 Table.

Next, we allowed the level of differentiation between conditions to vary across conditions. Such a situation commonly arises when one performs co-expression network analysis for multiple conditions. Here, we simulated the gene expression profiles under three conditions for

Table 1. Configurations of simulation scenarios. The level of differentiation between two conditions in the constituent modules is shown in Columns 2–3. II: identical network structure and identical edge weight; ID: identical network structure and different edge weights; DD: different network structures and different edge weights.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Group 1 (4 modules)</th>
<th>Group 2 (4 modules)</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>II</td>
<td>II</td>
</tr>
<tr>
<td>S2</td>
<td>II</td>
<td>ID</td>
</tr>
<tr>
<td>S3</td>
<td>ID</td>
<td>ID</td>
</tr>
<tr>
<td>S4</td>
<td>ID</td>
<td>DD</td>
</tr>
</tbody>
</table>

https://doi.org/10.1371/journal.pcbi.1006436.t001
450 genes comprised of 9 modules of 50 genes each. Similar to the 2-condition simulation, we included two groups of 4 modules of the same type. To better imitate real networks, we also included an additional type II module to mimic housekeeping co-expression across 3 conditions. In total, we considered 4 scenarios. Table 2 summarizes the configurations of these simulations. S1 and S2 represent the cases where pairwise similarities between conditions are constant across conditions, with a higher similarity in S1 than in S2. S3 and S4 represent the case where pairwise similarities vary across conditions, with a higher similarity in S3 than S4. In this simulation, we compared CFGL, FGL and GL. CFGLO was not included as its performance is similar to the previous case.

In all scenarios, our approach has a higher AUC (pAUC: S1: 0.649, S2: 0.508, S3: 0.650, S4: 0.513) and a lower SSE than both GL (pAUC: S1: 0.597, S2: 0.474, S3: 0.593, S4: 0.472) and FGL (pAUC: S1: 0.605, S2: 0.494, S3: 0.615, S4: 0.499) (Fig 4, S2 Table). The gain over GL is most apparent when the differentiation between conditions is low (S1). This is again because data integration is most beneficial when networks are similar across conditions. FGL performs well in this case too. However, when the distance between conditions is different across conditions (S3), the advantage of FGL over GL diminishes; and when the differentiation between conditions is relatively high (S2 and S4), FGL performs worse than GL. This is expected, as imposing similarity across conditions as in FGL is improper for these scenarios. However, our method performs well in all scenarios.

Taken together, we attribute the gain of our methods to its adaptive way of enforcing similarities. When networks are highly similar across conditions, enforcing similarities across all edges, as in FGL, is optimal. Our method adapts to this situation and produces similar results to FGL. In contrast, when networks are different across conditions, similarity should be
encouraged only among the shared edges in data integration. Our method is again adaptive to the differential patterns across conditions, thus shows more gain when the difference between conditions is present.

Application to rat expression data
We applied our method to a microarray dataset collected from a recombinant inbred (RI) rat panel and compared with FGL, GL and WGCNA, which is a widely used network analysis method based on marginal correlation [21]. The gene expression profiles in the brain and heart tissues were measured for 19 rat strains using Affymetrix Rat Exon Array 1.0 ST. Details on data processing and normalization are provided in Methods. Because of the small sample size, we restricted the network construction to the 500 most differentially expressed (DE) genes between brain and heart (see Methods). We used BIC to guide the selection of penalty parameters for CFGL, FGL and GL and used default parameters for WGCNA analysis.

Tissue specificity for edges. Since brain and heart have different embryonic origins and functions, a considerable number of tissue-specific co-expression relationships are expected. We first compared the co-expression networks constructed by each method in terms of tissue specificity (S2 and S3 Figs). Fig 5A shows the number of edges identified by each method categorized by their tissue specificity. For graphical model based methods, the number of edges are reported at the optimal BIC ($\lambda_1 = 0.0010$ and $\lambda_2 = 0.0008$ for both CFGL and FGL, and $\lambda_1 = 0.0009$ for GL). Our method and FGL identify substantially more tissue-common edges than GL and WGCNA. For example, at $\lambda_2 = 0.0008$ (Fig 5A), 26.0% (356 out of 1374) and 35.6% (354 out of 994) of edges detected by our method and FGL, respectively, are common between tissues; whereas only 0.2% (3 out of 1491) and 0.3% (5 out of 1495) of edges detected by GL and WGCNA are common between tissues (S3 Table). This is expected, as the fused penalty in CFGL and FGL enforces similarities across tissues. Compared with FGL, our method detects substantially more tissue-specific edges. To check if this pattern is related to the choices of $\lambda_2$, we also performed the same analysis at $\lambda_2 = 0.0010$ and 0.0012 (Fig 5B and 5C). This observation persists at all $\lambda_2$ levels. This difference is especially obvious when a relatively high $\lambda_2$ is applied: the proportion of tissue-specific edges detected by FGL rapidly reduces, whereas our method still maintains a considerable proportion of tissue-specific edges. For example, when $\lambda_2 = 0.0012$ (Fig 5C), our method detects 23.6% (191) brain-specific edges and 20.6% (161) heart-specific edges, whereas FGL detects only 1.1% (7) brain-specific edges and no heart-specific edges.

Tissue-specific hub genes identified by CFGL demonstrate highly tissue-specific biological functions. Because the estimated heart- and brain-specific networks are both considerably dense, it is difficult to identify disjoint co-expression modules. Instead, we first identified

### Table 2. Configurations of scenarios in the 3-condition simulation. The pairwise similarity between condition 1 and other conditions is reported.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Condition</th>
<th>Group 1 (4 modules)</th>
<th>Group 2 (4 modules)</th>
<th>Housekeeping module</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>C1-C2</td>
<td>II</td>
<td>ID</td>
<td>II</td>
</tr>
<tr>
<td></td>
<td>C1-C3</td>
<td>II</td>
<td>ID</td>
<td>II</td>
</tr>
<tr>
<td>S2</td>
<td>C1-C2</td>
<td>ID</td>
<td>DD</td>
<td>II</td>
</tr>
<tr>
<td></td>
<td>C1-C3</td>
<td>ID</td>
<td>DD</td>
<td>II</td>
</tr>
<tr>
<td>S3</td>
<td>C1-C2</td>
<td>II</td>
<td>ID</td>
<td>II</td>
</tr>
<tr>
<td></td>
<td>C1-C3</td>
<td>ID</td>
<td>DD</td>
<td>II</td>
</tr>
<tr>
<td>S4</td>
<td>C1-C2</td>
<td>II</td>
<td>ID</td>
<td>II</td>
</tr>
<tr>
<td></td>
<td>C1-C3</td>
<td>DD</td>
<td>DD</td>
<td>II</td>
</tr>
</tbody>
</table>

https://doi.org/10.1371/journal.pcbi.1006436.t002
tissue-specific hub genes. Then we formed a tissue-specific module for each hub gene using the hub gene and the genes directly connected with the hub by tissue-specific edges. To identify these hub genes, we counted the number of edges that are specific to each tissue for each gene and reported the five genes with the highest number of tissue-specific edges identified by our method (at $\lambda_1 = 0.001, \lambda_2 = 0.0008$) in Table 3. Because the genes used to construct networks in this analysis are differentially expressed genes, any random set of them presumably would show some tissue specificity. However, a random set of DE genes is much less likely to

Fig 4. Performance comparison for simulation with 3 conditions. Top row (A-D): ROC curve for edge detection for the settings of S1-S4; Bottom row (E-H): SSE for edge weights estimation for the settings of S1-S4. Red line: CFGL, Green line: FGL, Blue line: GL.

https://doi.org/10.1371/journal.pcbi.1006436.g004

Fig 5. The numbers of tissue-specific and tissue-common edges detected in two tissues (B: brain, H: heart) by our method, GL, and FGL. (A) $\lambda_2 = 0.0008$, (B) $\lambda_2 = 0.0010$ and (C) $\lambda_2 = 0.0012$.

https://doi.org/10.1371/journal.pcbi.1006436.g005
form a functionally coherent module than a set of co-expressed DE genes. Thus the functional coherence of a module helps establish confidence in the identified co-expression pattern. To examine the functional coherence for these modules, we performed a Gene Ontology (GO) enrichment analysis using the non-hub genes in each module and checked its agreement with the functionality of the hub gene. As a comparison, we also performed a GO enrichment analysis using a set of the most differentially expressed genes with the same size as that of the identified module.

Among the brain-specific hub genes, Mobp has been found to be specifically expressed in oligodendrocytes. It encodes the protein related to sheath compaction in rat brain and spinal cord [39]. Its peripheral genes also show a significant GO enrichment in myelin sheath (FDR = 1.558E-3), consistent with the function of Mobp. Interestingly, this GO term was not enriched using the set of 30 most differentially expressed genes in brain, providing an example of how differential co-expression analysis can uncover biologically important findings not revealed by differential expression analysis alone. Cacng3 is a protein-coding gene that encodes type I trans-membrane AMPA (α-amino-3-hydroxy-5-methyl-4-isoxazolepropionic acid) receptor regulatory protein (TARP). Its product, TARP gamma-3, is abundant in the cerebral cortex and amygdala, and has been found to be associated with childhood absence epilepsy in humans [40][41]. The genes specifically co-expressed with Cacng3 in brain are also enriched in the GO terms related to neuron differentiation (FDR = 1.332E-2), anterograde trans-synaptic signaling (FDR = 1.332E-2) and synapse (FDR = 4.161E-6). Among heart-specific hub genes, Nppb (also known as BNP) is a member of the natriuretic peptide family and encodes a secreted protein that functions as a cardiac hormone. It has been reported that Nppb is associated with an intra-cardiac counterregulatory mechanism that prevents the development of cardiac fibrosis in vivo. It has been suggested that this gene serves as a local regulator during the process of ventricular remodeling [42]. The genes specifically connected with Nppb in heart are also enriched in the GO terms related to cardiac muscle tissue development (FDR = 3.886E-3). Another heart-specific hub gene, Xirp2 (also known as CMYA3), has been reported to be related to the formation of intercalated disc (ICD), which is a juncture that links cardiac muscle cells and plays vital roles in signaling among cardiomyocytes[43]. The genes connected to Xirp2 are also enriched in GO terms related to muscle fiber development (FDR = 1.517E-2).

### Table 3. Top-5 tissue-specific hub genes identified by our method

The numbers of tissue-specific edges linked to the hub and the corresponding rankings are reported, in comparison with the results from FGL and GL. ‘-’ indicates that the ranking is outside of top 50.

<table>
<thead>
<tr>
<th>Tissue</th>
<th>Gene</th>
<th>CFGL #edge</th>
<th>FGL #edge</th>
<th>GL #edge</th>
<th>WGCNA #edge</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Ranking</td>
<td>Ranking</td>
<td>Ranking</td>
<td>Ranking</td>
</tr>
<tr>
<td>Brain</td>
<td>Cox8b</td>
<td>59</td>
<td>1</td>
<td>54</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Scrg1</td>
<td>39</td>
<td>2</td>
<td>34</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Cryab</td>
<td>37</td>
<td>3</td>
<td>36</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Cacng3</td>
<td>31</td>
<td>4</td>
<td>29</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td>Mobp</td>
<td>30</td>
<td>5</td>
<td>28</td>
<td>5</td>
</tr>
<tr>
<td>Heart</td>
<td>Nppb</td>
<td>38</td>
<td>1</td>
<td>9</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>LOC1003635047</td>
<td>18</td>
<td>2</td>
<td>14</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>Xirp2</td>
<td>17</td>
<td>3</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Eno3</td>
<td>17</td>
<td>4</td>
<td>1</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Cxcl11</td>
<td>7</td>
<td>5</td>
<td>1</td>
<td>-</td>
</tr>
</tbody>
</table>

[https://doi.org/10.1371/journal.pcbi.1006436.t003](https://doi.org/10.1371/journal.pcbi.1006436.t003)
FGL identifies the same set of brain-specific hubs and a similar brain-specific subnetwork as our method. However, our method detects more heart-specific hubs than FGL, and each hub harbors more heart-specific edges (Table 2). GL and WGCNA report drastically different results (S4 and S5 Tables) from FGL and our method in both tissues. None of the hubs reported by FGL and CFGL (Table 2) are ranked among the top 5 by these two methods. We also identified co-expression modules for rat brain and heart tissue using WGCNA. In total, 7 modules were identified for brain tissue (module size: 26–102 genes) and 3 for heart tissue (module size: 89–202 genes), with enriched GO terms related to neuron projection morphogenesis and chemical synaptic transmission for brain and transmembrane transporter activity and channel activity for heart (S6 Table). These modules generally are larger than those identified by CFGL. We discuss the observed differences in the Conclusion.

Application to TCGA BRCA data set

We applied our method to the breast cancer data from the TCGA project [44]. Breast cancer is the most common cancer among women [43]. According to the presence and absence of the estrogen receptor (ER) in cancer cells, breast cancer can be classified into two subtypes, ER+ and ER-. Approximately two-thirds of breast cancer are ER+ at the time of diagnosis, and the rest are ER-. The ER status provides important clinical implications for both mechanisms of carcinogenesis and therapeutic treatment [45].

The TCGA BRCA project [44] has collected gene expression RNA-seq data for 1100 breast cancer patients. Among them, 112 individuals have both tumor tissue and matched peripheral normal tissue. Our goal is to identify co-expression modules that are specific to ER+ or ER-subtype and those that are shared between the two tumor subtypes but are not present in normal tissue. To ensure the independence of the samples in our analysis, we used the normal tissue samples from these 112 individuals and tumor samples that are annotated as ER+ (187 samples) or ER- (98 samples) from different individuals. Due to the limited sample size, we restrict our analysis to a subset of 1000 genes that either show a significant association with survival time in a Cox model or have been reported to be related to breast cancer (Details in Methods). To obtain robust co-expression networks, we applied the stability selection procedure in [38] in conjunction with CFGL, FGL and GL. In addition, we also performed WGCNA on the same dataset as a comparison. Details of the data processing steps and the procedure of the stability selection can be found in Methods.

Disease type specificity of the co-expression edges. To investigate disease type specificity of the edges, we partitioned the identified edges into seven mutually exclusive categories: normal tissue only, ER+ subtype only, ER- subtype only, normal and ER+ shared, normal and ER- shared, ER+ and ER- shared, and all tissue common. The number of edges identified by each method in each category is summarized in S7 Table. Fig 6 compared the disease type specificity of the edges identified by the three methods. Similar to the rat dataset, the majority of the edges identified by GL and WGCNA are unique to one tissue with a very small percentage (GL: 68/5978 = 1.1%, WGCNA: 12/2834 = 4.2%) of edges shared by all tissues. In contrast, a high percentage of edges identified by FGL (1330/4448 = 30.0%) and our method (684/2624 = 26.1%) are common across all tissues.

We also evaluated the pairwise similarity of the co-expression networks between each pair of tissues (Fig 6). As ER+ and ER- tumors both are subtypes of breast cancer, they are expected to be more similar to each other than to normal tissue. However, we observed that FGL and GL show a higher proportion of shared edges between normal tissue and ER+ tumor (FGL: 446/4448 = 10.0%; GL: 290/5978 = 4.9%) than between the two tumor subtypes (FGL: 226/4448 = 5.1%; GL: 218/5978 = 3.6%). WGCNA identifies a very small proportion of shared
edges between any two tissues. Among them, more edges are shared between normal tissue and ER- tumor (30/2834 = 1%) than between the two tumor subtypes (22/2834 = 0.7%). In contrast, our method shows a much higher proportion of shared edges between the two tumor subtypes (332/2624 = 12.6%) than between the normal tissue and either of the tumor subtypes (normal and ER+: 250/2624 = 9.5%; normal and ER-: 60/2624 = 2.3%), reflecting the expected biological similarity. The proportion of common edges between the two tumor subtypes is also substantially higher in the network constructed by our method than in those constructed by the other three methods (CFGL: 332 (12.6%), FGL: 226 (5.1%), GL: 218 (3.6%), and WGCNA: 22 (0.7%)).

Genes most significantly associated with survival time usually are not hubs. To investigate the biological relevance of the hub genes in the co-expression network constructed by our method, we examined the relationship between the hubness of each gene, i.e. the number of edges connected to the gene, and its association with survival time. Strikingly, we found a clear negative correlation between the hubness of a gene and the significance of its association with survival time (Fig 7): the genes that have the most significant p-values in the Cox model usually are not hub genes, whereas hub genes tend to have less significant p-values. One possible explanation is that many genes that are significantly associated with survival time govern very
particular functions or are located downstream in pathways to execute the final biological functions, thus they are not correlated with the expression of many other genes. On the other hand, the regulators upstream in pathways, which are often hub genes, are less directly involved in the execution of final biological functions, thus they are less correlated with the survival time. This agrees with the observation in the previous analysis [5] of the co-expression network for 35 human tissues in the GTEx project, in which genes with tissue-specific functions were observed to have fewer co-expression edges than average. Certainly, cancer progression is also determined by many other factors, such as mutation, CNV or other structural alterations. Further biological or clinical evidence is still needed to confirm this interpretation.

Hubs specific to a disease type tend to not share edges across all tissues. To understand the role that a gene plays in condition-specific and condition-common co-expression, we classified edges according to their disease type specificity and compared the hubness of a gene in the network shared by all three tissues (3T) and the network specific to one tissue (1T). We observed a clear negative correlation between these two types of hubness (Fig 8), especially in the two tumor tissues. For all genes with at least 5 edges, the Spearman correlation between the numbers of 1T edges and the numbers of 3T edges is -0.23, -0.69, and -0.43 in normal, ER+ tumor, and ER- tumor tissue, respectively. It can be clearly seen that several ER+ or ER- specific hubs have very few edges shared across all tissues. This indicates that co-expression hubs specifically triggered in tumor tissues are usually not the co-expression hubs in normal tissue. These hubs may provide important insights in carcinogenesis and cancer treatment. The tumor-specific hubs and their possible biological functions are discussed in the next sections.

Biological functions of tumor-related modules. Next, we characterized the subnetwork shared between ER+/ER-, ER+ specific subnetwork and the ER- specific subnetwork in order to further study the biological function of the co-expression network of tumor tissues. We obtained each subnetwork by extracting the corresponding disease type specific edges from the networks constructed using our method. For each subnetwork, we identified major disjoint modules and then annotated their biological functions using GO enrichment analysis. To determine the hub gene in each module, we counted the number of edges for each node in the module. For ER+ specific and ER- specific modules, we characterized all genes that have more than 10 edges. Because the size of tumor-shared modules is relatively small, we only characterized the genes with the most edges in the module.
**Tumor-shared subnetwork.** The tumor-shared subnetwork consists of 203 genes and 332 co-expression edges (S4 Fig). There are 6 disjoint co-expression modules with more than 10 genes (S1 Fig). All 6 modules show significant enrichment (FDR < 0.05) in the GO term analysis. They are enriched in GO terms of Immunity, antigen processing and presentation, Nucleus, and DNA damage (Table 4).

Among the hubs of the 6 modules, 5 of them (PTPN22, BRIP1, CEACAM6, LTF, and CCNT1) have been previously found to be associated with breast cancer [46][47][48][49][50][51][52][53], and the other one, MXRA5, has been reported to be related to non-small cell lung cancer[54][55]. PTPN22 encodes a protein tyrosine phosphatase that is involved in the signaling pathways associated with immune response (Fig 9). Previous studies have shown that overexpression of PTPN22 significantly inhibits the growth of human breast cancer cells. Its product also blocks cancer cell xenografts and their metastases[46]. BRIP1 (also known as BACH1, FANCJ) together with BRCA1, is involved in the repair process of DNA double-strand breaks. It has been reported that BRIP1 acts as a master regulator of breast cancer [56]. Previous studies have reported that overexpression of BRIP1 promotes the migration and invasion of cancer cells, while knockdown of BRIP1 suppresses this process[57]. ZKSCAN1 (also known as KOX18, ZNF139), a node with three connections in the same module with BRIP1, has also been reported to play regulatory roles in migration and invasion of human gastric cancer cells [58][59]. CEACAM6 encodes a protein in the carcinoembryonic antigen family, which has been shown to be associated with cell adhesion. Previous studies have shown that CEACAM6 is detected in approximately 70% of solid tumors, including breast cancer [43][60]. It has been suggested that CEACAM6 is associated with tumor progression stage [61], inhibition of cell differentiation and anoikis, and promotion of cell adhesion, invasion, and metastasis [48]. LTF was previously found to inhibit the growth of solid tumors and the development of experimental metastases[49][50][51]. Overexpression of CCNT1 was found as an implication of tumor growth[53].

Table 4. Modules and their significantly enriched GO terms.

<table>
<thead>
<tr>
<th>Disease status</th>
<th>Modules ID (nodes number)</th>
<th>Hub(s)</th>
<th>GO term enrichment</th>
<th>FDR</th>
</tr>
</thead>
<tbody>
<tr>
<td>ER+ ER- shared</td>
<td>Module 1 (20)</td>
<td>PTPN22</td>
<td>antigen binding</td>
<td>2.60E-2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>regulation of immune response</td>
<td>2.50E-5</td>
</tr>
<tr>
<td></td>
<td>Module 2 (12)</td>
<td>MXRA5</td>
<td>cell adhesion molecule binding</td>
<td>4.60E-2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Cone-shaped epiphyses fused within their metaphyses</td>
<td>1.28E-2</td>
</tr>
<tr>
<td></td>
<td>Module 3 (17)</td>
<td>LTF</td>
<td>Perinuclear endoplasmic reticulum membrane</td>
<td>3.79E-2</td>
</tr>
<tr>
<td></td>
<td>Module 4 (20)</td>
<td>BRIP1</td>
<td>Nucleus</td>
<td>5.01E-6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>DNA damage</td>
<td>2.12E-4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Nucleoplasm</td>
<td>1.20E-3</td>
</tr>
<tr>
<td></td>
<td>Module 5 (21)</td>
<td>CCNT1</td>
<td>monocarboxylic acid binding</td>
<td>6.99E-3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>lysine-acetylated histone binding</td>
<td>8.28E-3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>transcription elongation from RNA polymerase II promoter</td>
<td>9.81E-4</td>
</tr>
<tr>
<td></td>
<td>Module 6 (10)</td>
<td>CEACAM6</td>
<td>holocytochrome-c synthase activity</td>
<td>1.99E-2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>manganese-transporting ATPase activity</td>
<td>1.99E-2</td>
</tr>
<tr>
<td>ER+</td>
<td>Module 7 (224)</td>
<td>LTF, NPY1R, CEACAM6</td>
<td>regulation of apoptotic process</td>
<td>1.99E-3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>regulation of intracellular signal transduction</td>
<td>6.31E-3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>regulation of secretion</td>
<td>3.32E-2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>response to estrogen</td>
<td>4.52E-2</td>
</tr>
<tr>
<td>ER-</td>
<td>Module 8 (108)</td>
<td>FOXA1, CEACAM6</td>
<td>acute inflammatory response</td>
<td>4.39E-3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>oncostatin-M receptor activity</td>
<td>3.72E-2</td>
</tr>
</tbody>
</table>

https://doi.org/10.1371/journal.pcbi.1006436.t004
ER+ specific subnetwork. The ER+ specific subnetwork consists of 554 edges and 296 nodes (S5 Fig). One major co-expression module with 224 nodes is detected. Table 4 shows the enriched GO terms and the hub genes for this module. This module is enriched in the GO term related to the response to estrogen. One of its hub genes, NYPIR, has been reported to be involved in the activation of estrogen signaling pathway in breast carcinoma. It is up-regulated in ER+ tumor compared to ER- tumor [62][63]. This agrees well with the classification of ER+ subtype, which is characterized by the presence of estrogen receptors. The other two hub genes, LTF and CEACAM6, are also hubs genes in the tumor-shared modules.

ER- specific subnetwork. The ER- specific network consists of 360 edges and 223 nodes (S6 Fig). One major co-expression module with 108 nodes is detected. The GO term analysis shows that ER- specific module is enriched in the activity of the oncostatin-M receptor. This receptor is involved in the signaling event of oncostatin-M, a growth regulator that inhibits the proliferation of a number of tumor cell lines. Interestingly, we observed that, though CEACAM6 is a hub in both ER+ and ER- specific modules, it connects with different sets of genes in the two subtypes (Table 4). This indicates that CEACAM6, which is associated with tumor progression stage [61], may regulate cancer progression through different mechanisms in these two tumor subtypes.

WGCNA analysis. We identified the hub genes for ER+, ER- tumor specific network and tumor shared network based on the TOM matrix estimated from WGCNA (S8 Table). Because very few edges are shared between any pair of networks inferred by WGCNA, there is only one hub gene (CCNT1) in the tumor-shared network. This gene is also identified as a tumor-shared hub gene by our method. In the ER+ and ER- specific networks, WGCNA and our method identified distinct hub genes. We also identified the co-expression modules using WGCNA (S9 Table). Similar to the rat analysis, WGCNA modules are generally larger (on average 266 genes in each module) than the modules from CFGL. These differences are further discussed in the Conclusion.

Conclusions
In this paper, we present a method, called condition-adaptive fused graphical lasso (CFGL), to construct gene co-expression networks for multiple conditions simultaneously. By incorporating a data-driven penalty that reflects the condition-specific co-expression pattern in the FGL.
framework, this method takes condition specificity into account while borrowing information across conditions in the network construction. Our results have shown that it effectively accounts for heterogeneity between samples and between co-expression patterns introduced by condition specificity. It outperforms GL and FGL methods in both edge detection and estimation of edge weights across a range of scenarios in simulation studies.

Our analysis on a rat multi-tissue dataset and TCGA breast cancer data reveals interesting biological insights. In both datasets, the modules in the condition-specific subnetwork identified by our method consistently show biologically relevant functions, demonstrating the suitability of our method for studying tissue-specific or disease-specific co-expression networks. The analysis on TCGA breast cancer data also reveals several interesting findings related to the mechanism of ER+ and ER- tumor subtypes. We found that the genes most significantly associated with survival time are less likely to be hubs. This suggests that most genes associated with cancer progression may govern specific functions or locate downstream in pathways to execute the final biological functions, rather than regulating a large number of biological processes. Similarly, we also observed that the hub genes in the tumor-specific subnetworks tend to not harbor edges shared with normal tissue. Several previously known cancer-related genes, including PTPN22, BRIP1, and CEACAM6, were found as hubs in the tumor-related subnetworks. Together, these results confirm the biological relevance of the results from our method.

Interestingly, we noticed that the methods that construct networks separately for each condition (GL and WGCNA) consistently produce very few condition-common edges (< 5%), far fewer than the joint analysis methods (FGL and CFGL). This is even the case when some common biology is expected to be shared between conditions, for example, ER+ and ER- breast cancer subtypes. The results reported by these analysis methods, and their suitability for studying condition specificity, are therefore questionable. Another interesting observation is that the graphical lasso based methods (GL, FGL and CFGL) generally report smaller modules than WGCNA. This is partly because the former uses sparse estimation and partly because the former evaluates conditional independence between genes, rather than marginal independence as in the latter. It is generally believed that very large gene sets may encompass multiple cellular processes and make GO enrichment results less specific [64][65][66], thus smaller modules may have the benefit of improving the interpretability of results.

Though our method was motivated by co-expression networks, it is suitable for other data applications with multiple conditions but shared network structures, such as learning condition-specific binary networks with sparse Ising models [67][68]. The R package for our method CFGL is available on GitHub, https://github.com/Yafei611/CFGL.

Method

Optimization of CFGL network

CFGL estimates the precision matrices \( \{ \Theta \} \) by solving

\[
\text{maximize}_{\Theta} \left( \sum_{k=1}^{K} n_k \log(\det(\Theta^{(k)})) - \text{tr}(S^{(k)}\Theta^{(k)}) - P(\{\Theta\}) \right).
\]

The penalty term \( P(\{\Theta\}) \) is

\[
P(\{\Theta\}) = \lambda_1 \sum_{k=1}^{K} \sum_{i\neq j} |\theta^{(k)}_{ij}| + \lambda_2 \sum_{k<k'} \sum_{i\neq j} w^{(k,k')}_{ij} |\theta^{(k)}_{ij} - \theta^{(k')}_{ij}|,
\]

where \( n_k \) is the sample size of \( k \)th condition and \( S^{(k)} = (Y^{(k)})^T Y^{(k)}/n_k \) is the empirical covariance matrix for the \( k \)th expression data set.

We implemented the ADMM algorithm [69] to solve the above problem. The detailed optimization procedure can be found in the supplementary materials (S1 Text).
**Tuning parameter selection**

We determine the tuning parameters $\lambda_1$ and $\lambda_2$ ($\lambda_2$ only for CFGL and FGL) according to the Bayesian information criterion (BIC) [70].

$$\text{BIC}(\lambda_1, \lambda_2) = \sum_{k=1}^K [n_k \text{tr}(S_k\hat{\Theta}_{(\lambda_1, \lambda_2)}^{(k)}) - n_k \log(\det(\hat{\Theta}_{(\lambda_1, \lambda_2)}^{(k)}))] + p_k \log(n_k),$$

where $\hat{\Theta}_{(\lambda_1, \lambda_2)}^{(k)}$ is the estimated precision matrix for the $k$th condition obtained at $(\lambda_1, \lambda_2)$, and $p_k$ is the number of non-zero elements in $\hat{\Theta}_{(\lambda_1, \lambda_2)}^{(k)}$. We ran the analysis on a series of combinations of $\lambda_1$ and $\lambda_2$, then chose the tuning parameters that achieve the minimal BIC value.

In the simulation, we used BIC to select tuning parameters. For the rat data analysis, because the sample size is very small, it is difficult to obtain meaningful estimates from sub-samples in the stability selection. Instead, we first identified the model that achieves the minimum BIC and the models with similar BIC values, then selected the model that had the fewest edges to obtain biologically interpretable results. In the TCGA data analysis, we applied a stability selection procedure to identify reliable edges (see stability selection section).

**Determining screening matrix**

We determined the screening matrix for CFGL by testing the differences between two precision matrices using the method proposed by Xia et al. [34]. This method tests whether the difference ($\Delta = \Sigma_{11}^{-1} - \Sigma_{00}^{-1}$) between two precision matrices is 0, i.e. $H_0: \Delta = 0$ vs $H_1: \Delta \neq 0$. To avoid falsely imposing similarity for edges that are moderately differential, we used a relaxed FDR threshold (FDR = 0.4) to determine differential entries, such that only the edges that are obviously non-differential across conditions (i.e. FDR $> 0.4$) were encouraged to be similar ($w_{ij}^{kk} = 1$). We implemented this method in R and included it in the CFGL package.

**Generation of Synthetic data**

In the simulation study, we generate the gene expression data for multiple network configurations. Suppose each condition contains $M$ disjoint modules and each module consists of $p$ genes. For each module, the gene constitution is constant across conditions, but the connectivity and the edge weight may vary across conditions. To generate conditions with a specified level of similarity, we first generate the network for condition 1, and then generate the network for other conditions based on their similarities to condition 1.

**Step 1: Simulating network for condition 1.** To generate a module in the network we first create an unweighted scale-free network, according to the Barabasi-Albert model [71] with an exponent of 1, to mimic real-world biological networks structure [72]. Then, we obtain the weighted network $A_1^{(m)}$ by assigning the edge weights as follows,

$$A_1^{(m)}(i,j) = \begin{cases} 
1, & i = j \\
0, & i \neq j, \text{there is no coexpression between gene i and gene j} \\
\sim U(D), & i \neq j, \text{there is coexpression between gene i and gene j}
\end{cases}$$

where $U(D)$ is a uniform distribution with $D = [-1,0.6] \cup [0.6,1]$. To ensure $A_1^{(m)}$ is positive definite, we add values on the matrix diagonal to get the modified matrix $B_1^{(m)}$:

$$B_1^{(m)} = A_1^{(m)} + \delta I,
$$

where $\delta$ is the minimal eigenvalue of the matrix $A_1^{(m)}$. Based on the matrix $B_1^{(m)}$, the covariance
matrix $\Sigma^{(m)}_1 = [\Sigma^{(m)}_1(i,j)]$ is determined by

$$
\Sigma^{(m)}_1(i,j) = [B^{(m)}_1]^{-1}(i,j)/\sqrt{[B^{(m)}_1]^{-1}(i,i)[B^{(m)}_1]^{-1}(j,j)}
$$

(6)

Finally, we obtained the covariance matrix for condition 1, $\Sigma_1$, by combining $\Sigma^{(m)}_1$ for each module.

$$
\Sigma_1 = \begin{bmatrix}
\Sigma^{(1)}_1 \\
\Sigma^{(2)}_1 \\
\vdots \\
\Sigma^{(M)}_1
\end{bmatrix}
$$

(7)

The expression data for condition 1 were generated from $N(0, \Sigma_1)$.

**Step 2: Simulating network for the other conditions.** The modules in other conditions were simulated based on their similarities to the corresponding module in condition 1. Three types of similarities were considered: (1) identical network structure and identical edge weights across conditions (II), (2) identical network structure but different edge weights across conditions (ID), and (3) different network structures and different edge weights (DD). The generation procedure to obtain $\Sigma^{(m)}_2$ from $\Sigma^{(m)}_1$ for these three types of similarities is as follows.

1. **II modules:** the covariance matrix for condition 2 is identical to that of condition 1.

   $$
   \Sigma^{(m)}_2 = \Sigma^{(m)}_1
   $$

2. **ID modules:** to maintain the network structure as in condition 1 but altering edge weights, a matrix $U$ is added to $B^{(m)}_1$:

   $$
   B^{(m)}_2 = B^{(m)}_1 + U
   $$

   where $U$ is a $p \times p$ matrix with elements:

   $$
   U(i,j) = \begin{cases} 
   \sim U(D^*) & B^{(m)}_1(i,j) \neq 0, i \neq j \\
   0 & \text{otherwise}
   \end{cases}
   $$

   Ans $U(D^*)$ is a uniform distribution with $D^* = [-0.6, 0.6]$. Then, $\Sigma^{(m)}_2$ can be obtained from

3. **DD modules:** $\Sigma_2$ is generated independently as described in step 1.

**Determining true screening matrix in simulation study**

To assess the performance of CFGL-oracle, we obtained the true screening matrix as

$$
W^{(k)}_\text{true}(i,j) = \begin{cases} 
1 & \Sigma^{-1}_k(i,j) - \Sigma^{-1}_0(i,j) < 0.01 \\
0 & \text{otherwise}
\end{cases}
$$

where $\Sigma^{-1}_k$ is the simulated precision matrix for the $k$th condition.
Accuracy of edge detection and edge weight estimation in the simulation study

The accuracy of edge identification is assessed by checking if the presence of edges is correct in the estimated matrix $\Sigma_k^{-1}$. We define true positive as $\Sigma_k^{-1}(i,j) \neq 0$ and $\Sigma_k^{-1}(i,j) \neq 0$ and false positive as $\Sigma_k^{-1}(i,j) = 0$ and $\Sigma_k^{-1}(i,j) \neq 0(i > j)$. The accuracy of edge weight estimation is assessed by the sum of square error (SSE) between the true and estimated edge weights

$$\text{SSE} = \sum_{k=1}^{K} \sum_{i=2}^{P} \sum_{j=1}^{i-1} (\Sigma_k^{-1}(i,j) - \Sigma_k^{-1}(i,j))^2,$$

where $K$ is number of conditions and $P$ is number of nodes (genes). For each $\lambda_2$, we generate an ROC curve by computing the true positive rate and false positive rate over a grid of $\lambda_1$. Similarly, SSE is computed over a grid of $\lambda_1$.

To compare the performance of different methods, we calculated partial AUC (pAUC), which is the area under the ROC curve over a restricted range of false positive rate (FPR). Because the primary interests are edges detected at a low false positive rate, we compute pAUC on the FPR range of (0,0.05) in the simulation study.

WGCNA analysis

We performed WGCNA analysis for rat and TCGA BRAC expression data. Because WGCNA does not allow joint analysis for more than one condition, we performed WGCNA for each condition separately.

We used the WGCNA R package (version 1.63) and chose the tuning parameters according to its manual [21]. For the rat expression data, the soft threshold was set to 8 for brain tissue and 7 for heart tissue, and the module size was set to 20 to accommodate the relatively small total number of genes. For TCGA data, the soft threshold was set to 6 for normal tissue, 5 for the ER+ tumor tissue and 4 for the ER- tumor tissue, and the module size was set to 50. Default settings were used for all the other parameters.

To study hub genes and edges using WGCNA, we obtained edge weights from the topological overlap matrix (TOM) calculated from WGCNA. The TOM is a quantity computed by WGCNA for measuring the topological similarity between genes. Each entry can be viewed as an edge weight between a pair of genes. Unlike graphical lasso based methods, which provide sparse networks, the TOM matrix is dense. To ensure the comparison with graphical lasso based methods is on the same basis, we only kept the edges with high values of TOM and removed other edges. The number of nonzero edges is chosen according to the number of edges identified by CFGL in the same dataset.

Construction of co-expression network for rat data

Heart and brain RNA expression levels were measured in a recombinant inbred (RI) rat panel, HXB/BXH, using the Affymetrix Rat Exon 1.0 ST Array (Affymetrix, Santa Clara, CA). This rat panel was originally generated using gender reciprocal crossing between the congenic Brown Norway strain with the polydactyly-luxate syndrome (BN-Lx/Cub) and the spontaneous hypertensive rat strain (SHR/OlaIpcv), with sixty generations of brother/sister mating after the F2 generation [73]. The CEL files for the heart and brain RNA expression data from 3 to 4 male rats per strain (19 strains) are publicly available through the PhenoGen website (http://phenogen.ucdenver.edu) [74] along with a probe mask for the ‘core’ (Affymetrix defined) transcript clusters that eliminates probes that do not align uniquely to the RN6 version of the rat genome or align to a region of the genome that harbors a single nucleotide
polymorphism between either of the parental strains (SHR and BN-Lx) and the reference genome. Further detail about this type of probe mask are available in Saba et al 2015 [75]. Transcript cluster estimates on the log2 scale were estimated using the rma-sketch pipeline for normalization and aggregation using Affymetrix Power Tools (Irizarry et al 2003; Lockstone 2011) [76][77]. Individual rat estimates were summarized as strain mean values for each transcript cluster and strain combination.

Given the small sample size, we restricted the network construction to the 500 most differentially expressed genes between the two tissues. The differential expression was determined using the R package LIMMA with the default parameter settings.

We ran CFGL and FGL for a grid of $\lambda_1$ and $\lambda_2$. They both achieved the lowest BIC at $\lambda_1 = 0.001$ and $\lambda_2 = 0.0008$. To investigate the effect of $\lambda_2$, we report the results at $\lambda_1 = 0.001$ and $\lambda_2 = 0.0008,0.0010,0.0012$. We set $\lambda = 0.0009$ for GL since it gives similar sparsity (edge number) to the other two methods. For WGCNA analysis, we kept the edges with the highest TOM values (1000 for brain and 500 for heart) from the WGCNA results, such that the number of edges is consistent with that of the estimated network from CFGL/FGL.

Construction of co-expression network with TCGA breast cancer data

TCGA has collected gene expression RNA-seq data for 1092 breast cancer patients [44]. We used the normal tissue samples from the individuals (n = 112) who have both tumor tissue and matched peripheral normal tissue, and all tumor samples from different individuals that were annotated with ER+ (n = 187) and ER- (n = 98)[44]. We obtained the gene expression level by downloading the RNA-seq V2 data, which are reads counts normalized by RSEM, from the TCGA website (https://cancergenome.nih.gov/). We then took log transformation for the expression level (with 0.5 added to the counts of each gene to avoid 0) and standardized the transformed expression level to mean 0 and standard deviation 1. Prior to network construction, we first removed genes with very low counts (less or equal than 5) in more than 10% (40) samples. After this step, the log summed read counts over all samples approximately follow a normal distribution.

Due to the limitation of sample size, we restricted our analysis to a subset of 1000 genes. To select 1000 genes in the analysis, we first included 39 genes that were previously reported as breast cancer-related genes (S10 Table) [23][44]. Then we selected other 961 genes that are most strongly associated with the survival time based on a univariate Cox regression:

$$h(t) = h_0(t) \times \exp(\beta g)$$

where $t$ is the survival time and $x_g$ is the expression level of the $g$th gene.

We constructed the co-expression network using CFGL/FGL/GL in conjunction with the stability selection procedure (See next section). For the WGCNA-based network, we kept the edges with the highest TOM values (920 edges for normal tissue, 1154 edges for ER+ tumor tissue and 840 edge for ER- tumor tissue).

Stability selection for TCGA data set

In order to obtain reliable co-expression networks, we applied the stability selection procedure in [78] to CFGL, FGL, and GL. This procedure first generates a large set of subsamples from the original data and then builds networks based on the subsamples. The edges that frequently occur in subsamples are kept. This method provides an upper bound for the FDR control and has been shown to outperform the standard GL when being applied to GL [78][38].

In our analysis, we created 100 subsamples, each of which contains half of the original samples. To reduce the computational load, we first determined the optimal choice of $\lambda_1$ based on
the original dataset, and then used this value for all subsamples. For all methods, $\lambda_1 = 0.2$ achieves both reasonable sparsity and low BIC across a series $\lambda_1 (0.01–0.50)$ on the original dataset, thus we fixed $\lambda_1 = 0.2$ in all subsamples. For FGL and CFGL, we performed the analysis on a series of $\lambda_2$ for each subsample and then used the tuning parameters that achieve the minimal BIC value to select edges. The minimal BIC for all subsamples was found in the range of $\lambda_2 = 0.002–0.02$. We keep the edges that appear in more than 90% subsamples. According to the false discovery rate (FDR) calculation in [38], this threshold guarantees that the number of wrong edges is less than 800 among the 499500 possible edges in the graph.

**GO enrichment analysis**

The GO enrichment analyses were conducted using ToppFun[79], which is publicly available at https://toppgene.cchmc.org/enrichment.jsp. All parameters are used at their default setting.

**Supporting information**

S1 Fig. Comparison of performance for simulations with two conditions with sample size $n = 100$. Top row (A-D): ROC curves for edge detection in the four simulation settings (S1-S4). Bottom row (E-H): SSE for edge weight estimation in the four simulation settings (S1-S4). Red line: CFGL, Green line: FGL, Blue line: GL, Purple line: CFGL-oracle.

(TIF)

S2 Fig. The rat brain specific network.
(TIFF)

S3 Fig. The rat heart specific network.
(TIFF)

S4 Fig. The ER+/ER- shared subnetwork. Red: Genes that are up-regulated in both tumor tissues in comparison with normal tissue. Blue: Genes that are down-regulated in both tumor tissues. Yellow: Genes that are up-regulated in one tumor tissue but down-regulated in another.

(TIFF)

S5 Fig. The ER+ specific subnetwork. Red: Genes that are up-regulated in both tumor tissues in comparison with normal tissue. Blue: Genes that are down-regulated in both tumor tissues. Yellow: Genes that are up-regulated in one tumor tissue but down-regulated in another.

(TIFF)

S6 Fig. The ER- specific subnetwork. Red: Genes that are up-regulated in both tumor tissues in comparison with normal tissue. Blue: Genes that are down-regulated in both tumor tissues. Yellow: Genes that are up-regulated in one tumor tissue but down-regulated in another.

(TIFF)

S1 Table.
(DOCX)

S2 Table.
(DOCX)

S3 Table.
(DOCX)

S4 Table.
(DOCX)
Author Contributions

Conceptualization: Yafei Lyu, Qunhua Li.

Formal analysis: Yafei Lyu.

Funding acquisition: Lingzhou Xue, Qunhua Li.

Investigation: Yafei Lyu, Laura Saba, Katerina Kechris, Qunhua Li.

Methodology: Yafei Lyu, Lingzhou Xue, Feipeng Zhang, Qunhua Li.

Project administration: Qunhua Li.

Resources: Laura Saba, Katerina Kechris.


Supervision: Qunhua Li.

Validation: Yafei Lyu, Qunhua Li.

Visualization: Yafei Lyu, Qunhua Li.

Writing – original draft: Yafei Lyu, Qunhua Li.

Writing – review & editing: Yafei Lyu, Lingzhou Xue, Hillary Koch, Laura Saba, Katerina Kechris, Qunhua Li.

References


17. Lee KH, Xue L. Nonparametric finite mixture of Gaussian graphical models. Technometrics. 2017; Forthcoming.
40. Moss FJ, Dolphin AC, Clare JJ. Human neuronal stargazin-like proteins, $\gamma_2$, $\gamma_3$ and $\gamma_4$; an investigation of their specific localization in human brain and their influence on Ca V 2.1 voltage-dependent calcium channels expressed in Xenopus oocytes. BMC Neurosci. 2003; 4(1):23.


