Supplementary Materials for "Bayesian Structure Learning in Multi-layered Genomic Networks"

S1 Background: Markov properties on chain graph models

We consider a chain graph model for a probability distribution over p random variables $\boldsymbol{Y} = (Y_1, \ldots, Y_p)^{\mathrm{T}} \in \mathbb{R}^p$. A graph of random variables $\boldsymbol{Y}$ can be denoted by $G = (V, E)$, where V contains p vertices $\{1, 2, \ldots, p\}$ that correspond to $\{Y_1 \ldots, Y_p\}$, and E may contain both directed ($\rightarrow$) and undirected edges ($-$). A *partially directed cycle* in a graph G is a sequence of m distinct vertices $v_1, \ldots, v_m$ ($m \geq 3$) and $v_1 = v_{n+1}$, such that

(a) $\forall\ i\ (1 \leq i \leq m)$ either $v_i - v_{i+1}$ or $v_i \leftarrow v_{i+1}$, and

(b) $\exists j\ (1 \leq j \leq m)$ such that $v_j \leftarrow v_{j+1}$.

(Lauritzen and Richardson, 2002). A graph G that has no partially-directed cycles is called a *chain graph*. The *chain components* of a chain graph G are the connected components of the undirected graph obtained by deleting all directed edges from G. A directed acyclic graph (DAG) is a chain graph where all chain components are singletons. Let $\mathcal{T} = \{\tau_k | 1 \leq k \leq q\}$, $q \leq p$, be a family of pairwise disjoint *ordered* blocks of vertices $\tau_k \neq \emptyset$, such that each τ_k is a union of chain components and $\cup_{1 \leq k \leq q} \tau_k = V$. The ordered partitioning $\mathcal{T}$ implies that any edges between blocks are directed. The partitioning $\mathcal{T}$ of V is called a *dependence chain* for G, if

$$k < l \implies v \nrightarrow u\ \forall u \in \tau_k, v \in \tau_l.$$

In other words, when $k < l$, any edges between τ_k and τ_l point from a vertex in τ_k to a vertex in τ_l. For a dependence chain $\mathcal{T}$, we define the *cumulatives* to be the set $C_l = \cup_{k \leq l} \tau_k$ for $1 \leq l \leq q$. For each $v \in V$, let $1 \leq t(v) \leq q$ be the index, such that $v \in \tau_{t(v)}$. We assume without loss of generality, that the vertices are labelled such that

$$t(u) < t(v) \implies u < v.$$

If $u \to v$, the vertex u is a *parent* of v, and if $u - v$, u is a *neighbor* of v. $\mathcal{P}(v) = \{u \in V : u \to v \in E\}$ and $\mathcal{C}(v) = \{u \in V : u - v \in E\}$ be the set of parents of v and the set of neighbors of v, respectively. For a subset $A \subseteq V$, we let $\mathcal{P}(A) = \cup_{v \in A} \mathcal{P}(v) \setminus A$ and $\mathcal{C}(A) = \cup_{v \in A} \mathcal{C}(v) \setminus A$. From the non-existence of partially-directed cycles, the joint probability distribution of $\boldsymbol{Y}$ can be factorized as,

$$P(\boldsymbol{Y}) = \prod_{\tau \in \mathcal{T}} P(\boldsymbol{Y}_\tau | \boldsymbol{Y}_{\mathcal{P}(\tau)}). \tag{1}$$

This factorization is equivalent to directed acyclic graphs (DAG), where each node is replaced by a chain component.

Lauritzen and Wermuth (1989); Frydenberg (1990) (LWF) introduced a Markov property for chain graphs that generalizes the Markov properties for both UGs and DAGs. Each factors, $P(\boldsymbol{Y}_\tau | \boldsymbol{Y}_{\mathcal{P}(\tau)})$ in (1) further factorizes into a product of densities of the cliques of the undirected graph, which has $\tau \cup \mathcal{P}(\tau)$ as nodes and undirected edges, $u - v$ if either both of these are in $\mathcal{P}(\tau)$ or there is an (directed or undirected) edge between them in the chain graph G (Lauritzen, 1996). The corresponding pairwise Markov property for G states that

$$u - v \notin E \text{ and } u \to v \notin E \implies Y_u \perp\!\!\!\perp Y_v | Y_{C_{t(v)} \setminus \{u,v\}}. \tag{2}$$

A missing edge between two random variables between Y_u and Y_v implies that they are conditionally independent, given all other variables in $\tau_1, \ldots, \tau_{t(v)}$. On the other hand, Andersson et al. (2001) proposed an alternative Markov property (AMP) for chain graphs and states the pairwise Markov property for G,

$$u - v \notin E \implies Y_u \perp\!\!\!\perp Y_v | Y_{C_{t(v)} \setminus \{u,v\}} \text{ for } t(u) = t(v) \tag{3}$$

$$u \to v \notin E \implies Y_u \perp\!\!\!\perp Y_v | Y_{C_{t(v)-1} \setminus \{u\}} \text{ for } t(u) < t(v). \tag{4}$$

A missing (directed) edge between two random variables Y_u and Y_v implies that they are conditionally independent, given all other variables in $\tau_1, \ldots, \tau_{t(v)-1}$, while the conditional sets of missing (undirected) edges are the same as the LWF Markov property. For example, in Figure S1, the LWF Markov property implies the conditional independencies, $Y_1 \perp\!\!\!\perp Y_2$, $Y_1 \perp\!\!\!\perp Y_4 | Y_2, Y_3$, and $Y_2 \perp\!\!\!\perp Y_3 | Y_1, Y_4$. On the other hand, the AMP Markov property states the conditional independencies, such that $Y_1 \perp\!\!\!\perp Y_2$, $Y_1 \perp\!\!\!\perp Y_4 | Y_2$, $Y_2 \perp\!\!\!\perp Y_3 | Y_1$. From the conditional distribution of $\boldsymbol{Y}_\tau$ given $\boldsymbol{Y}_{\mathtt{pa}_\tau}$, the matrix of regression coefficients $\mathbf{B}$ is $-\boldsymbol{\mathcal{K}}_\tau^{-1}\boldsymbol{\mathcal{K}}_{\tau,\mathtt{pa}_\tau}$. By the Gaussian chain graph model in (3), the directed edges under the AMP model can be directly read from the zero structure in $\mathbf{B}$. However, the LWF model is equivalent to the less easily interpretable conditions on $\boldsymbol{\mathcal{K}}_{\tau,\mathtt{pa}_\tau}$ (Andersson et al., 2001). Sohn and Kim (2012); McCarter and Kim (2014) proposed a penalized likelihood method with a sparsity assumption on $\boldsymbol{\mathcal{K}}_{\tau,\mathtt{pa}_\tau}$. In the case of continuous variables with a joint multivariate normal distribution, the AMP Markov property (not the LWF Markov property) is coherent with data generation by a block-recursive linear system (Cox and Wermuth, 1993; Andersson et al., 2001; Drton and Eichler, 2006). In the case of continuous variables with a joint multivariate normal distribution, the AMP Markov property (not the LWF Markov property) is coherent, with data generation via a block-recursive linear system (Cox and Wermuth, 1993; Andersson et al., 2001; Drton and Eichler, 2006). Thus, AMP models are preferable for regression-based model selection frameworks.

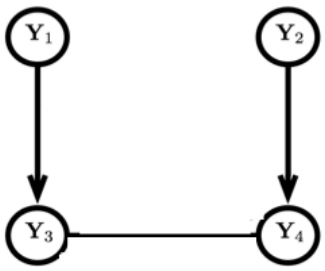

Figure S1: An example of chain graph.

S2 Proofs

Proposition 1.

Proof. From model in equation (7),

$$\begin{aligned}
\boldsymbol{\epsilon}_{\mathcal{C}_v} &= (\mathbf{I}-\mathbf{B})_{\mathcal{C}_v,\mathcal{C}_v\cup\mathcal{P}_v}\boldsymbol{Y}_{\mathcal{C}_v\cup\mathcal{P}_v} \\
&= \begin{pmatrix}(\mathbf{I}-\mathbf{B})_{\mathcal{C}_v,\mathcal{C}_v} & (\mathbf{I}-\mathbf{B})_{\mathcal{C}_v,\mathcal{P}_v}\end{pmatrix}\begin{pmatrix}\boldsymbol{Y}_{\mathcal{C}_v}\\ \boldsymbol{Y}_{\mathcal{P}_v}\end{pmatrix} \\
&= \boldsymbol{Y}_{\mathcal{C}_v} + (\mathbf{I}-\mathbf{B})_{\mathcal{C}_v,\mathcal{P}_v}\boldsymbol{Y}_{\mathcal{P}_v}.
\end{aligned}$$

Thus,

$$\begin{aligned}
\epsilon_v &= \boldsymbol{\epsilon}_{\mathcal{C}_v}^{\mathrm{T}}\boldsymbol{\alpha}_v + e_v \\
&= \boldsymbol{Y}_{\mathcal{C}_v}^{\mathrm{T}}\boldsymbol{\alpha}_v + \boldsymbol{Y}_{\mathcal{P}_v}^{\mathrm{T}}(\mathbf{I}-\mathbf{B})_{\mathcal{C}_v,\mathcal{P}_v}^{\mathrm{T}}\boldsymbol{\alpha}_v + e_v \\
&= \boldsymbol{Y}_{\mathcal{C}_v}^{\mathrm{T}}\boldsymbol{\alpha}_v - \boldsymbol{Y}_{\mathcal{P}_v}^{\mathrm{T}}\mathbf{B}_{\mathcal{C}_v,\mathcal{P}_v}^{\mathrm{T}}\boldsymbol{\alpha}_v + e_v.
\end{aligned}$$

□

S3 Sampling scheme

Let $\boldsymbol{\mathcal{E}} = \begin{pmatrix}\boldsymbol{\eta}_1 & \boldsymbol{\eta}_2 & \dots\end{pmatrix}^{\mathrm{T}}$. Let $P(\eta_{vw}=1) = P(\eta_{wv}=1) = p_{vw}$ for $v \in V$ and $w \in \mathcal{C}(v)$ and $P(\gamma_{vw}=1) = q_{vw}$ for $v \in V$ and $w \in \mathcal{P}_v$. For each vertex $v \in \tau \subseteq V$, we perform the following sampling.

Update undirected edges

Set $\tilde{\mathbf{y}}_v = \mathbf{y}_v - \mathbf{Y}_{\mathcal{P}_v}\mathbf{b}_v$ and $\mathbf{X}_v = \mathbf{Y}_{\mathcal{C}_v} - \mathbf{Y}_{\mathcal{P}_v}\mathbf{B}_{\mathcal{C}_v,\mathcal{P}_v}^{\mathrm{T}}$.

1. Metropolis Hastings algorithm: $\boldsymbol{\mathcal{E}} \sim p(\boldsymbol{\mathcal{E}}|\mathbf{Y},\boldsymbol{\kappa})$

   -s is the current state

   (1) Add-delete or swap. With probability 1/2,

   - Sample w_1 from $\{u : u \in \mathcal{C}_v\}$ and set $\eta^*_{vw_1} = \eta^*_{w_1v} = 1 - \eta^s_{vw_1}$, or

   - Sample w_2 from $\{w : \eta_{vw} = 0\} \setminus \{v\}$, and w_3 from $\{w : \eta_{vw} = 1\}$ and set $\eta^*_{vw_2} = \eta_{w_2v} = 1$, and $\eta^*_{vw_3} = \eta_{w_3v} = 0$

(2) Compute the acceptance ratio:

$$R = \prod_{r \in \{v, w_1\}} \frac{p(\tilde{\mathbf{y}}_r | \mathbf{X}_r, \boldsymbol{\eta}_r^*, \kappa_{rr}) p(\boldsymbol{\eta}_r^*)}{p(\tilde{\mathbf{y}}_r | \mathbf{X}_r, \boldsymbol{\eta}_r^s, \kappa_{rr}) p(\boldsymbol{\eta}_r^s)} \text{ (add-delete)}$$

$$R = \prod_{r \in \{v, w_2, w_3\}} \frac{p(\tilde{\mathbf{y}}_r | \mathbf{X}_r, \boldsymbol{\eta}_r^*, \kappa_{rr}) p(\boldsymbol{\eta}_r^*)}{p(\tilde{\mathbf{y}}_r | \mathbf{X}_r, \boldsymbol{\eta}_r^s, \kappa_{rr}) p(\boldsymbol{\eta}_r^s)} \text{ (swap)}$$

where $p(\tilde{\mathbf{y}}_v | \mathbf{X}_v, \boldsymbol{\eta}_v, \kappa_{vv})$ is the density of $N\left(\mathbf{0}, \frac{1}{\kappa_{vv}}\left(\mathbf{I} - \mathbf{X}_v^\eta \left(\mathbf{X}_v^{\eta\mathrm{T}} \mathbf{X}_v^\eta + \mathbf{D}_v^{-1}\right)^{-1} \mathbf{X}_v^{\eta\mathrm{T}}\right)^{-1}\right)$ for $\mathbf{D}_v = \frac{1}{\lambda_\tau}\mathbf{I}$ and $p(\boldsymbol{\eta}_r) = \prod_{k \in \mathcal{C}(r)} p_{rk}^{\eta_{rk}} (1 - p_{rk})^{1 - \eta_{rk}}$.

(3) Sample $U \sim Uniform(0, 1)$, setting $\boldsymbol{\mathcal{E}}^{s+1} = \boldsymbol{\mathcal{E}}^*$ if $U < R$ and setting $\boldsymbol{\mathcal{E}}^{s+1} = \boldsymbol{\mathcal{E}}^s$.

2. Gibbs sampling:
$\boldsymbol{\alpha}_r | \tilde{\mathbf{y}}_r, \mathbf{X}_r, \boldsymbol{\eta}_r, \kappa_{rr} \sim N\left(\left(\mathbf{X}_r^{\eta\mathrm{T}} \mathbf{X}_r^\eta + \mathbf{D}_r^{-1}\right)^{-1} \mathbf{X}_r^{\eta\mathrm{T}} \tilde{\mathbf{y}}_r, \frac{1}{\kappa_{rr}}\left(\mathbf{X}_r^{\eta\mathrm{T}} \mathbf{X}_r^\eta + \mathbf{D}_r^{-1}\right)^{-1}\right)$ for all r in $\{v, w_1\}$ or $\{v, w_2, w_3\}$.

3. Gibbs sampling:
$\kappa_{rr} \sim p(\kappa_{rr} | \tilde{\mathbf{y}}_r, \mathbf{X}_r, \boldsymbol{\eta}_r, \boldsymbol{\alpha}_r)$
$= Gamma\left(\frac{n + \delta_\tau + |\tau| - 1 + \|\boldsymbol{\eta_r}\|_0 + \|\boldsymbol{\gamma}_r\|_0}{2}, \frac{\lambda_\tau}{2} + \frac{1}{2}\left[(\tilde{\mathbf{y}}_r - \mathbf{X}_r^\eta \boldsymbol{\alpha}_r^\eta)^{\mathrm{T}} (\tilde{\mathbf{y}}_r - \mathbf{X}_r^\eta \boldsymbol{\alpha}_r^\eta) + \mathbf{b}_r^{\gamma\mathrm{T}} \mathbf{C}_v^{-1} \mathbf{b}_r^\gamma + (\lambda_\tau + |\tau| - 1) \boldsymbol{\alpha}_r^{\eta\mathrm{T}} \boldsymbol{\alpha}_r^\eta\right]\right)$ for all r in $\{v, w_1\}$ or $\{v, w_2, w_3\}$, and $\|\cdot\|_0$ is L_0 norm (number of nonzero values).

Update directed edges

Let G be the current graph and $u_1, u_2, \ldots$ be vertices in $\mathtt{ne}_v$ obtained from G. Set $\tilde{\mathbf{y}}_v = \mathbf{y}_v - \mathbf{Y}_{\mathcal{C}_v} \boldsymbol{\alpha}_v$,
$\mathbf{X}_\tau = \begin{pmatrix} \mathbf{Y}_{\mathcal{P}_v} & -\alpha_{vu_1} \mathbf{Y}_{\mathcal{P}_v} & -\alpha_{vu_2} \mathbf{Y}_{\mathcal{P}_v} & \ldots \end{pmatrix}$
, and $\mathbf{C}_\tau = \text{Blockdiag}\left(\frac{1}{\kappa_{vv}} \mathbf{C}_v, \frac{1}{\kappa_{u_1 u_1}} \mathbf{C}_{u_1}, \ldots\right)$

1. Metropolis Hastings algorithm: $\boldsymbol{\Gamma}_{\tau, \mathcal{P}_\tau} \sim p(\boldsymbol{\Gamma}_{\tau, \mathcal{P}_\tau} | \mathbf{y}, \mathbf{X}_\tau, \boldsymbol{\kappa}_{\tau\tau})$

s is the current state.

(1) Add-delete or swap. With probability 1/2,

- Sample k_1 from $\{k : k \in \mathcal{P}_v\}$, then set $\gamma_{v,k_1}^* = 1 - \gamma_{v,k_1}^s$ (delete-add), or

- Sample k_2 from $\{k : k \in \mathcal{P}_v\}$ and k_3 from $\{k : k \notin \mathcal{P}_v\}$, then set $\gamma_{v,k_2} = 0$ and $\gamma_{v,k_3} = 1$ (swap).

(2) Compute acceptance ratio:

$$R = \frac{p(\tilde{\mathbf{y}}_v | \mathbf{X}_\tau, \boldsymbol{\Gamma}_{\tau, \mathcal{P}_\tau}^*, \boldsymbol{\kappa}_{\tau\tau}) p(\boldsymbol{\Gamma}_{\tau, \mathcal{P}_\tau}^*)}{p(\tilde{\mathbf{y}}_v | \mathbf{X}_\tau, \boldsymbol{\Gamma}_{\tau, \mathcal{P}_\tau}^s, \boldsymbol{\kappa}_{\tau\tau}) p(\boldsymbol{\Gamma}_{\tau, \mathcal{P}_\tau}^s)}$$

where $p(\tilde{\mathbf{y}}_v|\mathbf{X}_\tau, \mathbf{\Gamma}_{\tau,\mathcal{P}_\tau}, \boldsymbol{\kappa}_{\tau\tau})$ is density of $N\left(\mathbf{0}, \frac{1}{\kappa_{vv}}\left(\mathbf{I} - \kappa_{vv}\mathbf{X}_\tau^\gamma\left(\kappa_{vv}\mathbf{X}_\tau^{\gamma\mathrm{T}}\mathbf{X}_\tau^\gamma + \mathbf{C}_\tau^{-1}\right)^{-1}\mathbf{X}_\tau^{\gamma\mathrm{T}}\right)^{-1}\right)$ and $p(\mathbf{\Gamma}^*_{\tau,\mathcal{P}_\tau}) = \prod_{v\in\tau, w\in\mathcal{P}_\tau} q_{vw}^{\gamma_{vw}}(1-q_{vw})^{1-\gamma_{vw}}$.

(3) Sample $U \sim Uniform(0,1)$ and set $\mathbf{\Gamma}^{s+1}_{\tau,\mathcal{P}_\tau} = \mathbf{\Gamma}^*_{\tau,\mathcal{P}_\tau}$ if $U < R$ and set $\mathbf{\Gamma}^{s+1}_{\tau,\mathcal{P}_\tau} = \mathbf{\Gamma}^s_{\tau,\mathcal{P}_\tau}$.

2. Gibbs sampling:
$vec(\mathbf{B}^{\mathrm{T}}_{\tau,\mathcal{P}_\tau})|\tilde{\mathbf{y}}_v, \mathbf{X}_\tau, \mathbf{\Gamma}_{\tau,\mathcal{P}_\tau}, \boldsymbol{\kappa}_{\tau\tau} \sim N\left(\kappa_{vv}\left(\kappa_{vv}\mathbf{X}_\tau^{\gamma\mathrm{T}}\mathbf{X}_\tau^\gamma + \mathbf{C}_\tau^{-1}\right)^{-1}\mathbf{X}_\tau^{\gamma\mathrm{T}}\tilde{\mathbf{y}}_v, \left(\kappa_{vv}\mathbf{X}_\tau^{\gamma\mathrm{T}}\mathbf{X}_\tau^\gamma + \mathbf{C}_\tau^{-1}\right)^{-1}\right)$.

3. Gibbs sampling:
$\kappa_{vv}|\tilde{\mathbf{y}}_v, \mathbf{X}_\tau, \mathbf{\Gamma}_{\tau,\mathcal{P}_\tau}, \mathbf{B}_{\tau,\mathcal{P}_\tau}$
$\sim Gamma\left(\frac{n+\delta_\tau+|\tau|-1+\|\boldsymbol{\eta}_\tau\|_0+\|\mathbf{\Gamma}_{\tau,\mathcal{P}_\tau}\|_0}{2}, \frac{\lambda_\tau}{2} + \frac{1}{2}\left[(\tilde{\mathbf{y}}_v - \mathbf{X}_\tau^\gamma\tilde{\mathbf{b}}_\tau^\gamma)^{\mathrm{T}}(\tilde{\mathbf{y}}_v - \mathbf{X}_\tau^\gamma\tilde{\mathbf{b}}_\tau^\gamma) + \tilde{\mathbf{b}}_\tau^{\gamma\mathrm{T}}\mathbf{C}_\tau^{-1}\tilde{\mathbf{b}}_\tau^\gamma + (\lambda_\tau + |\tau| - 1)\boldsymbol{\alpha}_v^{\eta\mathrm{T}}\boldsymbol{\alpha}_v^\eta\right]\right)$, where $\tilde{\mathbf{b}}_\tau = vec(\mathbf{B}_{\tau,\mathcal{P}_\tau})$, and $\|\cdot\|_0$ is L_0 norm (number of nonzero values).

S4 Sensitivity analysis to hyper parameters

We assess sensitivity of our model by the choice of the hyper-parameters, λ_τ and δ_τ when we set $c^2_{vw} = 1/\lambda_\tau$ for $v \in \tau$ and $w \in \mathcal{P}(\tau)$ in equation (11). The δ_τ and λ_τ are the shape and scale parameters of the Gamma prior on the precision (inverse variance), κ_{vv} of node $v \in \tau$. Moreover, λ_τ, the κ_{vv} contribute to the variance of the non-zero regression coefficients in $\boldsymbol{\alpha}_v$ and $\boldsymbol{b}_v$. The results given in the simulation section were obtained using the setting $\lambda = 5$ and $\delta = 1$ for all τ, which reflects the mean precision of the non-zero regression coefficients, $E(\lambda_\tau\kappa_{vv}) = \delta_\tau + |\tau| - 1$, to be the size of the corresponding layer plus 1, $|\tau| + 1$, and the mean of κ_{vv} to be $|\tau|/5$. To examine the effect of varying λ and δ, we fitted chain graphs for the simulation setting $(p, q, p_E) = (20, 6, 0.3)$ by changing λ and δ from 1 to 10 with fixed $\delta = 1$ and $\lambda = 5$, respectively, for all of the 6 layers. The average PPIs for non-zero edges was consistently higher (around 0.9) than those for zero edges (around 0.1) in the range from 1 to 10 of λ_τ and δ_τ (Figure S2). The average PPIs showed slight increasing and decreasing patterns for non-zero and zero edges in the true graph, until around $\lambda = 5$, while δ values provide flat patterns of the average PPIs for both non-zero and zero edges.

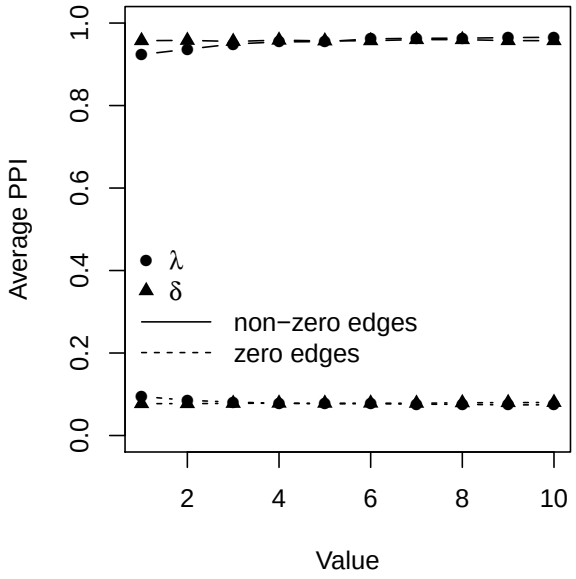

Figure S2: Sensitivity analysis for the simulation setting, $(p, q, p_E) = (20, 6, 0.3)$. Averages of PPIs for edges and gaps in the true graph, and averages of posterior means of κ for all $p = 20$ nodes, according to different λ and δ values.

S5 Convergence

We illustrate posterior inference using simulated datasets from a chain graph with the setting, $(p, n, q, p_E) = (20, 200, 6, 0.3)$. To obtain a sample from the posterior distribution, we ran the MCMC sampler described in Section 5.1 with 10,000 iterations as burn-in, and 20,000 iterations as the basis of inference. Figure S3 shows the traces of the number of edges and log-likelihood included in the chain graph. These plots show good mixing around a stable model size and no strong trends.

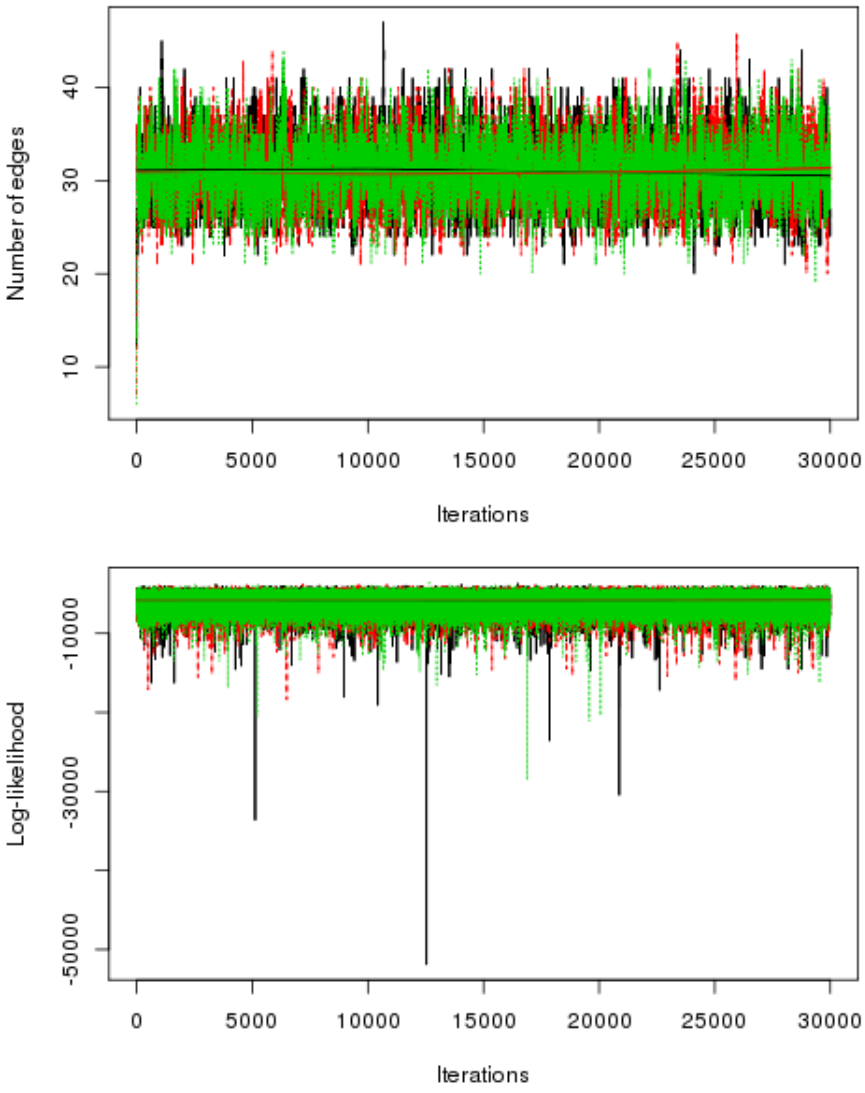

Figure S3: Trace plot from parallel runs of 3 chains for $(p, n, q, p_E) = (20, 200, 6, 0.3)$

S6 BANS-parallel

We implemented a new node-wise sampling scheme. From node-wise regression for $v \in \tau \subseteq V$ of the working model in Proposition 1, we learn the neighbor of v within τ, and all the directed edges from the preceding vertices $\mathcal{P}_v$ to τ. For the inference of directed edges toward a node $v \in \tau$, we use only MCMC samples of γ_v although we obtain MCMC samples for $\{\gamma_v : v \in \tau\}$ for each node-wise regression. For the undirected graphical structure, we use post-hoc symmetrization using posterior means of $\{\eta_{vw}\}$: in that, the posterior marginal probability of edge inclusion for an undirected edge $v-w$ is defined by the maximum value of the posterior means of η_{vw} and η_{wv}, which is analogous to the OR rule in (Meinshausen and Bühlmann, 2006). For the BANS-parallel method, the MCMC algorithm for updating undirected edges (Section 5.1) is modified as follows:

1. Undirected edges: for a node $v \in \tau \subseteq V$,

   1.1 Set $\tilde{\mathbf{y}}_v = \mathbf{y}_v - \mathbf{Y}_{\mathcal{P}_v}\mathbf{b}_v$ and $\mathbf{X}_v = \mathbf{Y}_{\mathcal{C}_v} - \mathbf{Y}_{\mathcal{P}_v}\mathbf{B}^{\mathrm{T}}_{\mathcal{C}_v,\mathcal{P}_v}$.

   1.2 Update $\boldsymbol{\eta}_v$ and set $\mathtt{ne}_v^{(t)} = \{w \in \tau : \eta_{vw} \neq 0\}$.

   1.3 Update $\boldsymbol{\alpha}_v$, and κ_{vv}.

The detailed sampling scheme for undirected edges in Section S3, Supplementary Materials are changed to:

Update undirected edges

Set $\tilde{\mathbf{y}}_v = \mathbf{y}_v - \mathbf{Y}_{\mathcal{P}_v}\mathbf{b}_v$ and $\mathbf{X}_v = \mathbf{Y}_{\mathcal{C}_v} - \mathbf{Y}_{\mathcal{P}_v}\mathbf{B}^{\mathrm{T}}_{\mathcal{C}_v,\mathcal{P}_v}$.

1. Metropolis Hastings algorithm: $\boldsymbol{\eta}_v \sim p(\boldsymbol{\eta}_v|\mathbf{Y}, \kappa_{vv})$

   -s is the current state

   (1) Add-delete or swap. With probability 1/2,

   - Sample w_1 from $\{u : u \in \mathcal{C}_v\}$ and set $\eta^*_{vw_1} = 1 - \eta^s_{vw_1}$, or

   - Sample w_2 from $\{w : \eta_{vw} = 0\} \setminus \{v\}$, and w_3 from $\{w : \eta_{vw} = 1\}$ and set $\eta^*_{vw_2} = 1$, and $\eta^*_{vw_3} = 0$

   (2) Compute the acceptance ratio:

$$R = \frac{p(\tilde{\mathbf{y}}_v|\mathbf{X}_v, \boldsymbol{\eta}^*_v, \kappa_{vv})p(\boldsymbol{\eta}^*_v)}{p(\tilde{\mathbf{y}}_v|\mathbf{X}_v, \boldsymbol{\eta}^s_v, \kappa_{vv})p(\boldsymbol{\eta}^s_v)} \text{ (add-delete)}$$

$$R = \frac{p(\tilde{\mathbf{y}}_v|\mathbf{X}_v, \boldsymbol{\eta}^*_v, \kappa_{vv})p(\boldsymbol{\eta}^*_v)}{p(\tilde{\mathbf{y}}_v|\mathbf{X}_v, \boldsymbol{\eta}^s_v, \kappa_{vv})p(\boldsymbol{\eta}^s_v)} \text{ (swap)}$$

where $p(\tilde{\mathbf{y}}_v|\mathbf{X}_v, \boldsymbol{\eta}_v, \kappa_{vv})$ is the density of $N\left(\mathbf{0}, \frac{1}{\kappa_{vv}}\left(\mathbf{I} - \mathbf{X}_v^{\eta}\left(\mathbf{X}_v^{\eta\mathrm{T}}\mathbf{X}_v^{\eta} + \mathbf{D}_v^{-1}\right)^{-1}\mathbf{X}_v^{\eta\mathrm{T}}\right)^{-1}\right)$ for $\mathbf{D}_v = \frac{1}{\lambda_\tau}\mathbf{I}$ and $p(\boldsymbol{\eta}_r) = \prod_{k\in\mathcal{C}(r)} p_{rk}^{\eta_{rk}}(1-p_{rk})^{1-\eta_{rk}}$.

(3) Sample $U \sim Uniform(0,1)$, setting $\boldsymbol{\mathcal{E}}^{s+1} = \boldsymbol{\mathcal{E}}^*$ if $U < R$ and setting $\boldsymbol{\mathcal{E}}^{s+1} = \boldsymbol{\mathcal{E}}^s$.

2. Gibbs sampling:
$\boldsymbol{\alpha}_v|\tilde{\mathbf{y}}_v, \mathbf{X}_v, \boldsymbol{\eta}_v, \kappa_{vv} \sim N\left(\left(\mathbf{X}_v^{\eta\mathrm{T}}\mathbf{X}_v^{\eta} + \mathbf{D}_v^{-1}\right)^{-1}\mathbf{X}_v^{\eta\mathrm{T}}\tilde{\mathbf{y}}_v, \frac{1}{\kappa_{vv}}\left(\mathbf{X}_v^{\eta\mathrm{T}}\mathbf{X}_v^{\eta} + \mathbf{D}_v^{-1}\right)^{-1}\right)$.

3. Gibbs sampling:
$\kappa_{vv} \sim p(\kappa_{vv}|\tilde{\mathbf{y}}_v, \mathbf{X}_v, \boldsymbol{\eta}_v, \boldsymbol{\alpha}_v)$
$= Gamma\left(\frac{n+\delta_\tau+|\tau|-1+\|\boldsymbol{\eta}_v\|_0+\|\boldsymbol{\gamma}_v\|_0}{2}, \frac{\lambda_\tau}{2} + \frac{1}{2}\left[(\tilde{\mathbf{y}}_v - \mathbf{X}_v^{\eta}\boldsymbol{\alpha}_v^{\eta})^{\mathrm{T}}(\tilde{\mathbf{y}}_v - \mathbf{X}_v^{\eta}\boldsymbol{\alpha}_v^{\eta}) + \mathbf{b}_v^{\gamma\mathrm{T}}\mathbf{C}_v^{-1}\mathbf{b}_v^{\gamma} + (\lambda_\tau + |\tau| - 1)\boldsymbol{\alpha}_v^{\eta\mathrm{T}}\boldsymbol{\alpha}_v^{\eta}\right]\right)$,
where $\|\cdot\|_0$ is L_0 norm (number of nonzero values).

The simulation performance of BANS-parallel is evaluated in Section S7.4.

S7 Simulations

S7.1 Simulation Setup for DNA Damage Response pathway for TCGA LUSC samples

We investigated the degree distribution of our estimated networks in the Section 7. The distribution of degree (total number of undirected and directed connections) is displayed in Figure S4 for LUSC cancer type, where the $\log_{10}$ scale density of $\log_{10}\nu$ is overlaid. The linearity between $\log_{10}\nu$ and $\log_{10}P(\nu)$ supports that the estimated network follows power law ($P(\nu) \sim \nu^{-k}$), thus has scale-free feature (Barabási and Albert, 1999). We have added a new simulation setting emulating the real data application example for DNA damage response network for 309 TCGA lung squamous cell carcinoma (LUSC) samples. We used the parameter estimates, $\hat{\mathbf{B}}$ and $\hat{\boldsymbol{\mathcal{K}}}$ from BANS in Section 7 to generate simulation datasets. The chain graph structure is displayed in Figure 8. Figure S8 displays the ROC curves and MCC curves for the four different methods, BANS, BANS-parallel, MRCE and CAPME.

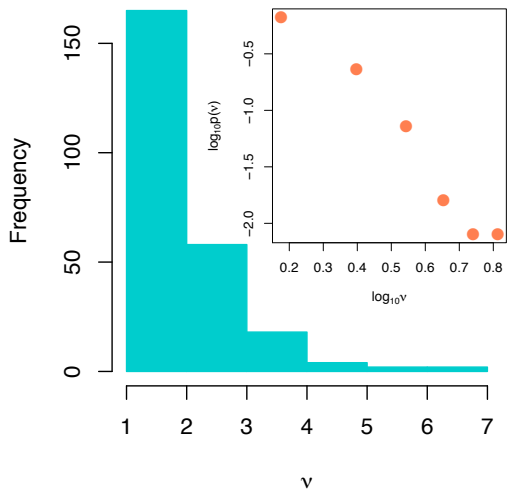

Figure S4: Histogram of degree ν for the estimated network for LUSC samples across the 10 pathways, where the $\log_{10}$ scale density of $\log_{10}\nu$ is overlaid.

S7.2 Non-Gaussian Example

Since BANS includes MRF estimation for single-layered networks as special case, we compared the XMRF method (Wan et al., 2016) developed by Genevera Allen's group for fitting MRF from mixed data. To analyze RNA-seq data, they employ log-linear graphical model that estimates the graphical structure via neighborhood selection by L1 penalized log-linear regressions. By using neighborhood selection approach, they can capture both positive and negative dependencies. We simulated 50 replication datasets from multivariate poisson distributions with scale-free network structure using the `XMRF.Sim` function in the `XMRF` R packages. The `XMRF` function provides optimal network by measuring stability using bootstrap samples. For BANS method, we selected a network by controlling FDR of 0.1. We considered sample size of $n = 100$ and number of nodes of $p = 20$ (the dimensionality used as example in the XMRF package) and compared Matthew's correlation coefficient (MCC) values for 50 replication datasets. The violin plot is displayed in Figure S5 and shows that the BANS performs better than XMRF in edge selection in terms of false and true positives showing higher MCC values across 50 simulated datasets. One source of the inaccuracy of the XMRF procedure might be their regularization method using 100 bootstrap samples (chosen by default)- across 50 replication datasets, it tends to choose much more edges than the true network.

Our modeling framework is viable for non–gaussian distribution with heavier tail as the induced marginal distribution becomes t-distribution by modeling the variances and covariances.

From a practical perspective, when RNA-seq count data is not sparse and skewed, then application of the BANS after standardizing (for each gene) provides accurate estimation of MRF. However, when the empirical distribution of the data are skewed and include a lot of zero values, appropriate preprocessing steps should be used for network estimation based on normality assumption — as is done usually (Ritchie et al., 2015; Ha and Sun, 2018). For applying BANS, for the data in the first layer e.g. DNA-level data such as methylation and copy number variation, it is not necessary to assume normality – since they appear as covariates in the model – and only regulatory (directed) edges to the downstream layers such as mRNA and protein expression data can be incorporated if normality assumption on such data are not valid.

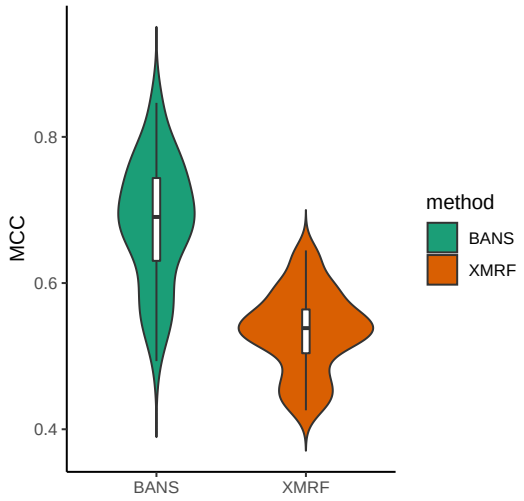

Figure S5: Violin plots of MCC values across 50 replication datasets.

S7.3 Comparison with Horseshoe prior

Shrinkage-based priors provide a viable alternative to fitting our models. We have now added additional simulation studies, to compare the horseshoe prior in estimating MRF structure using

neighborhood selection approach with BANS. We considered the horseshoe prior:

$$\begin{aligned}
\alpha_{vw} &\sim N(0, \kappa_{vv}\lambda_{vw}^2\eta_v^2), \\
\lambda_{vw} &\sim C^+(0,1), \\
\eta_v &\sim C^+(0,1), \\
\kappa_{vv} &\sim 1/\sigma^2,
\end{aligned}$$

where $C^+(0,1)$ denotes a half-Cauchy distribution. For BANS, as described in Section 4.2, we employed mixture priors on the regression coefficients for the undirected edges and gamma priors on the inverse variances:

$$\begin{aligned}
\alpha_{vw} &\sim \eta_{vw}N(1, 1/(\lambda_\tau\kappa_{vv})) + (1-\eta_{vw})\delta_0, \\
\kappa_{vv} &\sim Gamma(\frac{\delta_\tau + |\tau| - 1}{2}, \frac{\lambda_\tau}{2}),
\end{aligned}$$

where $P(\eta_{vw} = 1) = q_{vw}$ with fixed q_{vw}. λ_τ and δ_τ are fixed hyper-parameters for layer τ.

Using the `horseshoe` R package following the implementation procedure in (Bhattacharya et al., 2016), we implemented the neighborhood selection framework for learning MRF. We compared the performance of our BANS method using point mass prior and horseshoe prior in learning UG for a single layer when $p = 100$ (number of variables), $n = 30$ (sample size), and $p_E = 0.02$ (degree of sparsity). We followed the same simulation data generation procedure in Section 6.1 for single-layered case. After performing all $p = 100$ multiple regressions using the horseshoe prior, the posterior mean of the regression coefficients were used for inference after symmetrization step $\rho_{vw} = \rho_{wv} = \sqrt{|\hat{\alpha}_{vw}\hat{\alpha}_{wv}|}$ and $\{\rho_{vw}|v < w\}$ (absolute values of partial correlations, (Peng et al., 2009; Ha and Sun, 2014)) were used to compute ROC curves and MCC curves. Figure S6 shows the averaged results across 30 replicated graph structures and the corresponding datasets. The point mass prior in our BANS method showed better performance than horseshoe prior in recovering undirected graph structure using node-wise regressions with respect to ROC curves and MCC across different different sparsity. We note that in the simulation studies in Section 6.2.1, we have extensively compared the structure estimation performance with MRCE that uses Lasso penalty, that is comparable to Laplace prior. In summary, we construct a regression-based formulation

that converts the mlGGM into a more tractable node-wise multiple regression model and utilize point-mass priors on the regression coefficients to learn the graphical structure. In estimating MRF using neighborhood selection approach, we compared the point-mass and Horseshoe priors while a recent literature (Li et al., 2019) proposed graphical Horseshoe estimator, where the priors are specified for each of the off-diagonal elements on the precision matrix.

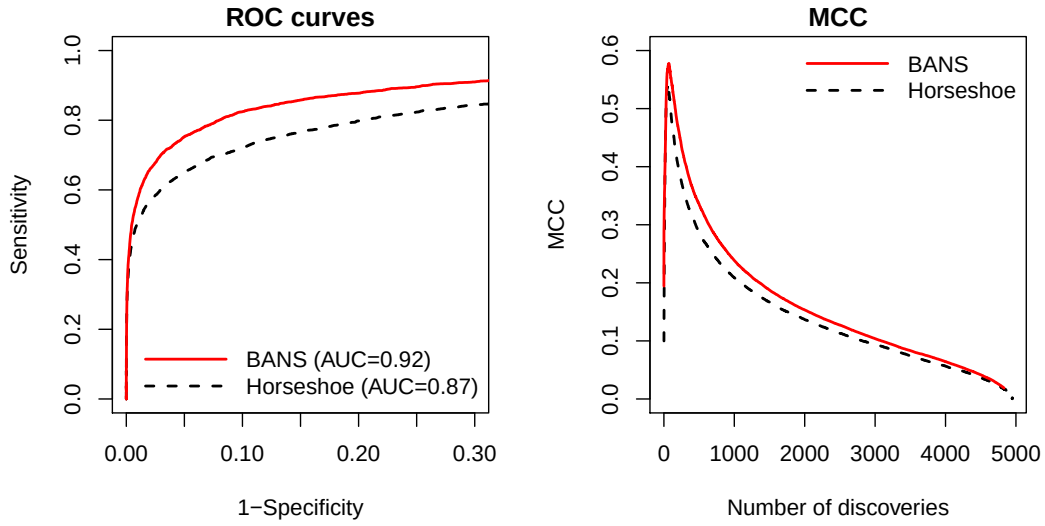

Figure S6: ROC curves and MCC curves for one-layered graph structure learning for (p, n, q, p_E) = (100,30,1,0.02).

S7.4 Comparison with objective Bayes fractional Bayes factor (OBFBF)

The objective Bayes fractional Bayes factor (OBFBF) method (**?**) also proposed a joint selection approach of the nonzero elements of the directed ($\mathbf{B}$) and undirected ($\boldsymbol{\mathcal{K}}$) edges. However, our BANS approach performs more flexible modeling for selecting both directed and undirected edges for the following reasons. First, the structure of undirected edges in (**?**) is limited to decomposable graphs using hyper-inverse Wishart prior and the method, moreover, selects entire columns of the coefficient matrix $\mathbf{B}_{\tau_k, C_{k-1}}$, and fails to select single elements of this matrix (i.e., a variable in an upper layer is either connected to all variables in a lower layer or to none), and the same precision matrix $\boldsymbol{\mathcal{K}}_{\tau_k}$ informs the selection of the coefficient matrix $\mathbf{B}_{\tau_k, C_{k-1}}$. Assumptions on both $\mathbf{B}_{\tau_k, C_{k-1}}$ and $\boldsymbol{\mathcal{K}}_{\tau_k}$, while improving computational efficiency due to conjugate formulation and allowing for

the exact calculation of the marginal likelihood of the graph, impose the artificial restriction on both undirected and directed structures, and may potentially result in misspecification of the networks structures. In particular, the space of decomposable graphs corresponding to $\boldsymbol{\mathcal{K}}_{\tau_k}$ is increasingly sparse with the increasing size of τ_k. For example, the percentages of graphs that are decomposable decrease as 95%, 80%, 55%, 29% and 12% for $|\tau_k| = 4$, 5, 6, 7, and 8, respectively (Armstrong, 2005).

We tested the model selection accuracy of OBFBF with the same model averaging approach as BANS using simulation studies. We used the simulation setting emulating the real data application example of DNA damage response network for 309 TCGA lung squamous cell carcinoma (LUSC) samples. Data were simulated using the parameter estimates $\hat{\mathbf{B}}$ and $\hat{\boldsymbol{\mathcal{K}}}$ from the data analysis described in Section 7 (the chain graph structure is displayed in Figure 8). The network includes 39 nodes across 4 platforms and the directed edges allowed across platforms are described in Figure 5. Note that the R implementation of OBFBF (https://github.com/stefanopel/OBFBF) does not include the option of fitting undirected networks for the first two platforms (Methylation and CNA), and forces the selection of parent sets that are connected to all nodes in the child layer. We then decided to evaluate the selection performances based on only the conditional undirected graph structure of the third and fourth layer (corresponding to mRNA and protein expressions). Using the posterior marginal probability of edge inclusion for each edge g_{vw} for both OBFBF and BANS, we computed ROC and MCC curves in Figure S7. While BANS showed the perfect selection accuracy based on the AUC values of 1, OBFBF resulted in a AUC of 0.79 and low MCC values across different levels of sparsity of the selected graphs.

S7.5 Performance of BANS-parallel

We evaluated the empirical performance of the BANS-parallel (Section S6) with respect to BANS, MRCE, and CAPME under a new simulation setting, emulating the DNA damage response network for 309 TCGA lung squamous cell carcinoma (LUSC) samples estimated in Section 7 and displayed in Figure 8 (in the revised paper). We used the chain graph structure and regression coefficient

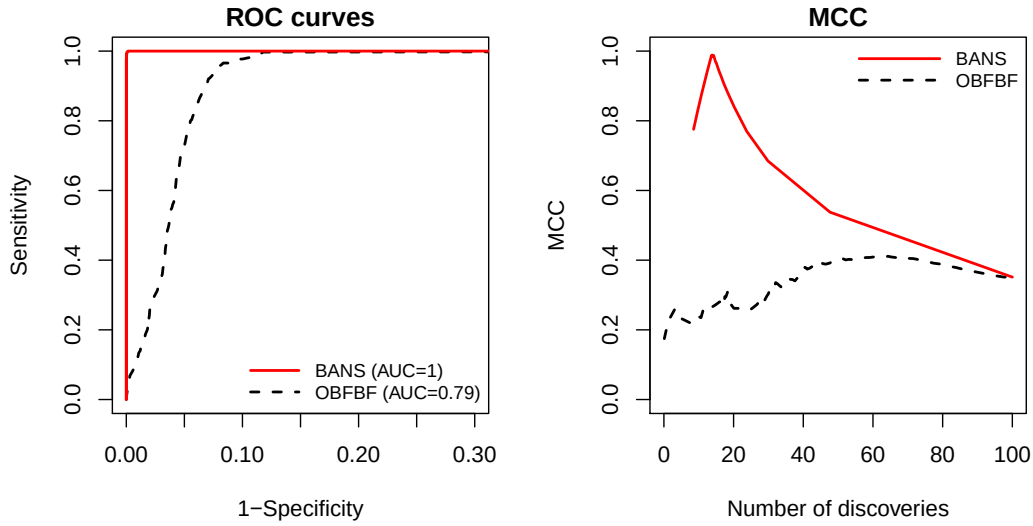

Figure S7: ROC curves and MCC curves for graph structure learning for (p, n, q) = (39,309,4), emulating real data example of DNA damage response pathway for TCGA LUSC samples.

estimates to generate simulation datasets following the same procedure, described in Section 6.1. Figure S8 displays the ROC curves and MCC curves averaged over 25 replications, which demonstrate that BANS still performs the best. However, the difference in structural accuracies between BANS and BANS-parallel is marginal given the simulation setting, compared to other methods such as MRCE and CAPME. Based on the computation time (Figure S9), BANS-parallel took 15 minutes for 5000 MCMC iterations to perform a node-wise regression for number of nodes from 20 to 100 across two-layers, while BANS took 37 hours to learn the p=100 network jointly. Thus, BANS-parallel is potentially useful considering the gain in computational efficiency.

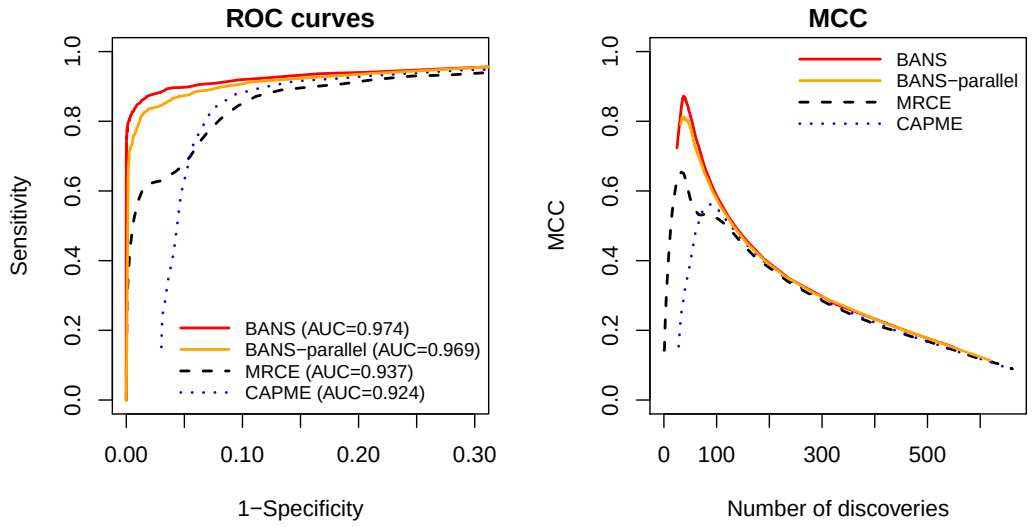

Figure S8: ROC curves and MCC curves for graph structure learning for (p, n, q) = (39,309,4), emulating real data example of DNA damage response pathway for TCGA LUSC samples.

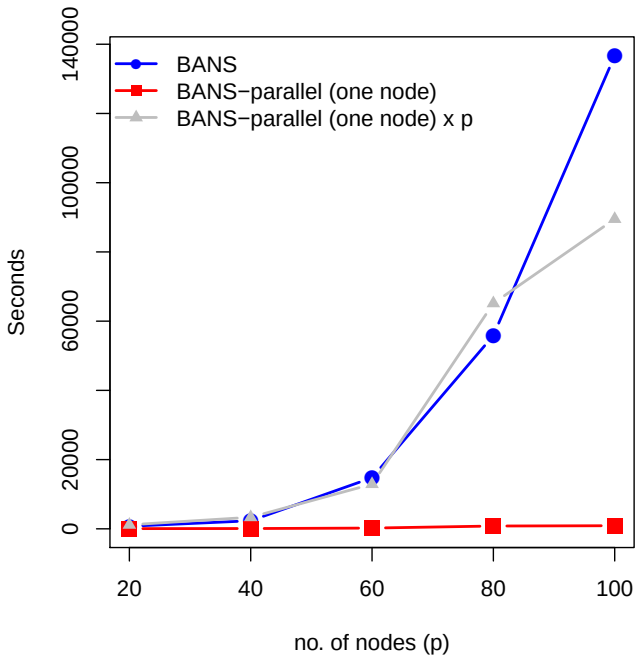

Figure S9: Computation time for number of nodes (p) from 20 to 100 for two layers with equal number of nodes.

S7.6 Posterior inference on the signs

The main focus of this section is to make inference on the signs of the edges, conditioned on the estimated undirected and directed structures using our node-wise regression approach. We define the sign for an undirected edge as the sign for its corresponding partial correlation. The partial correlation for an edge $v - w$ is $-\kappa_{vw}/\sqrt{\kappa_{vv}\kappa_{ww}}$, for which the sign is the same as that of the regression coefficient α_{vw} or α_{wv}. The sign for a directed edge is straightforward, $sign(b_{vw})$ for $w \to v$. Thus, the signs of all estimated edges are obtained by structured estimation of the nonzero elements in $\mathbf{B}$ and $\boldsymbol{\mathcal{K}}$. The structured MCMC sampling scheme can be followed by the same procedure described in Section 5.1 and Section S3, given $\boldsymbol{\eta}$ and $\boldsymbol{\gamma}$. Then the posterior probabilities for negative or positive signs can be obtained for each edge. We investigate the performance of

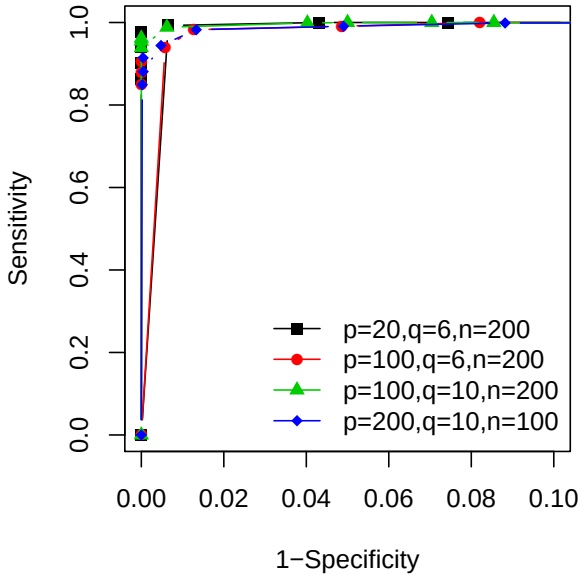

Figure S10: ROC curves for posterior inference on predicting positive signs from the structured estimation. The AUCs were 0.99 for all the scenarios.

the inference on the sign using a simulation study. We declare that an undirected edge $v - w$ is positive, if $P(\alpha_{vw} > 0|data) > \xi$, or negative otherwise; and a directe edge $w \to v$ is positive, if $P(b_{vw} > 0|data) > \xi$, or negative otherwise. Figure S10 shows ROC curves for predicting positive signs from the structured estimation, which was averaged over 50 replications. With AUCs 0.99 for all the four settings, we can conclude that signs are accurately estimated from our structured estimation given $\boldsymbol{\eta}$ and $\boldsymbol{\gamma}$, using our node-wise regression model.

S8 Supplementary figures and tables

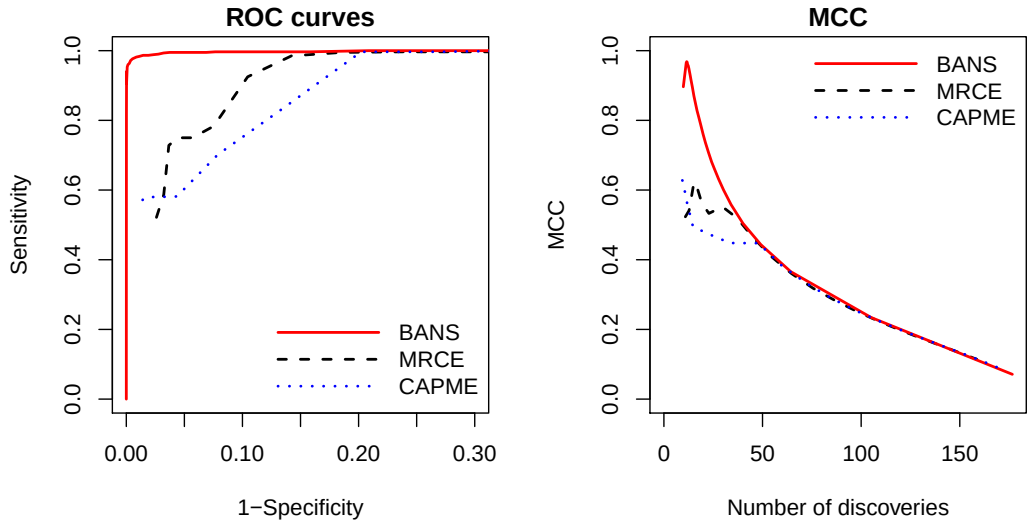

Figure S11: ROC curves and MCC curves for graph structure learning for $(p, n, q, p_E) =$ (20,200,6,0.3).

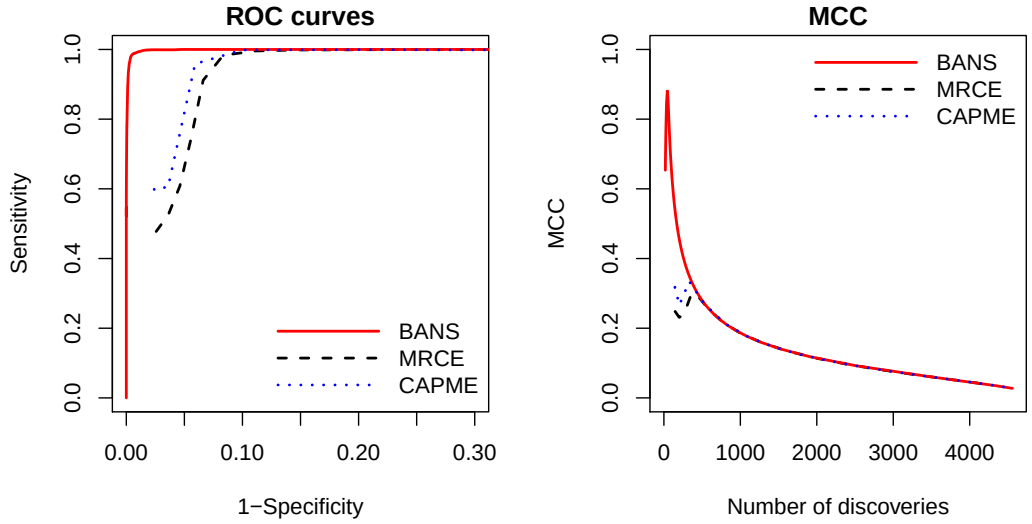

Figure S12: ROC curves and MCC curves for graph structure learning for $(p, q, p_E) =$ (100,200,6,0.03).

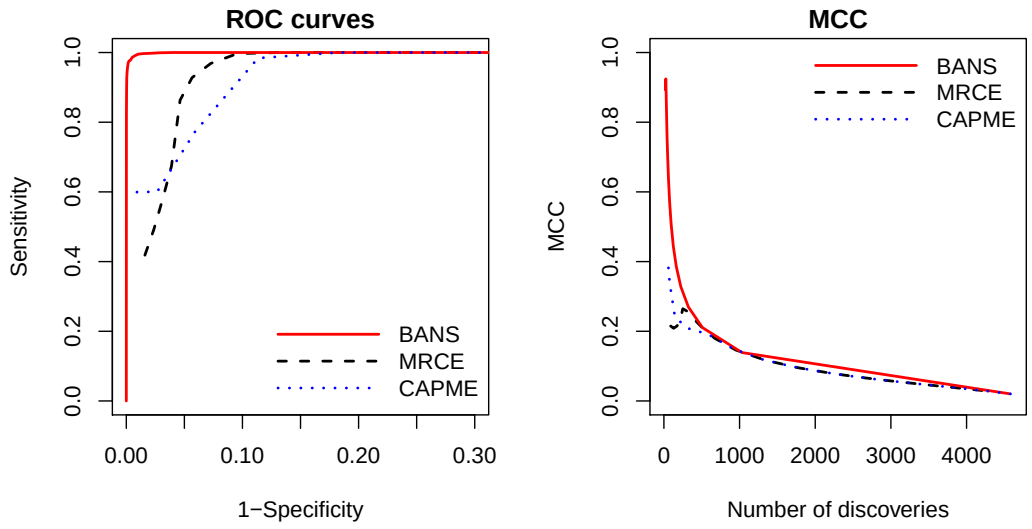

Figure S13: ROC curves and MCC curves for graph structure learning for $(p, q, p_E) = (100,200,10,0.03)$.

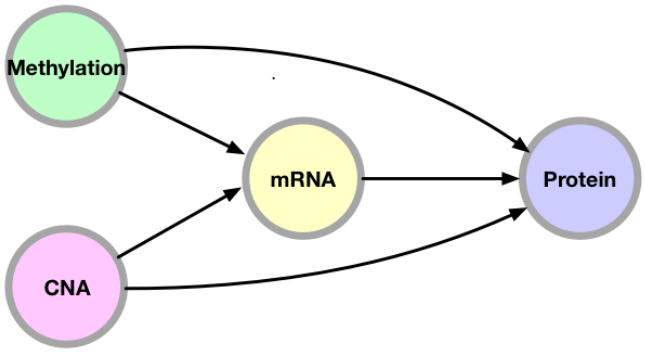

Figure S14: Inter-relationships between multi-platform data, copy number aberration (CNA), DNA methylation, nRNA expression, and protein.

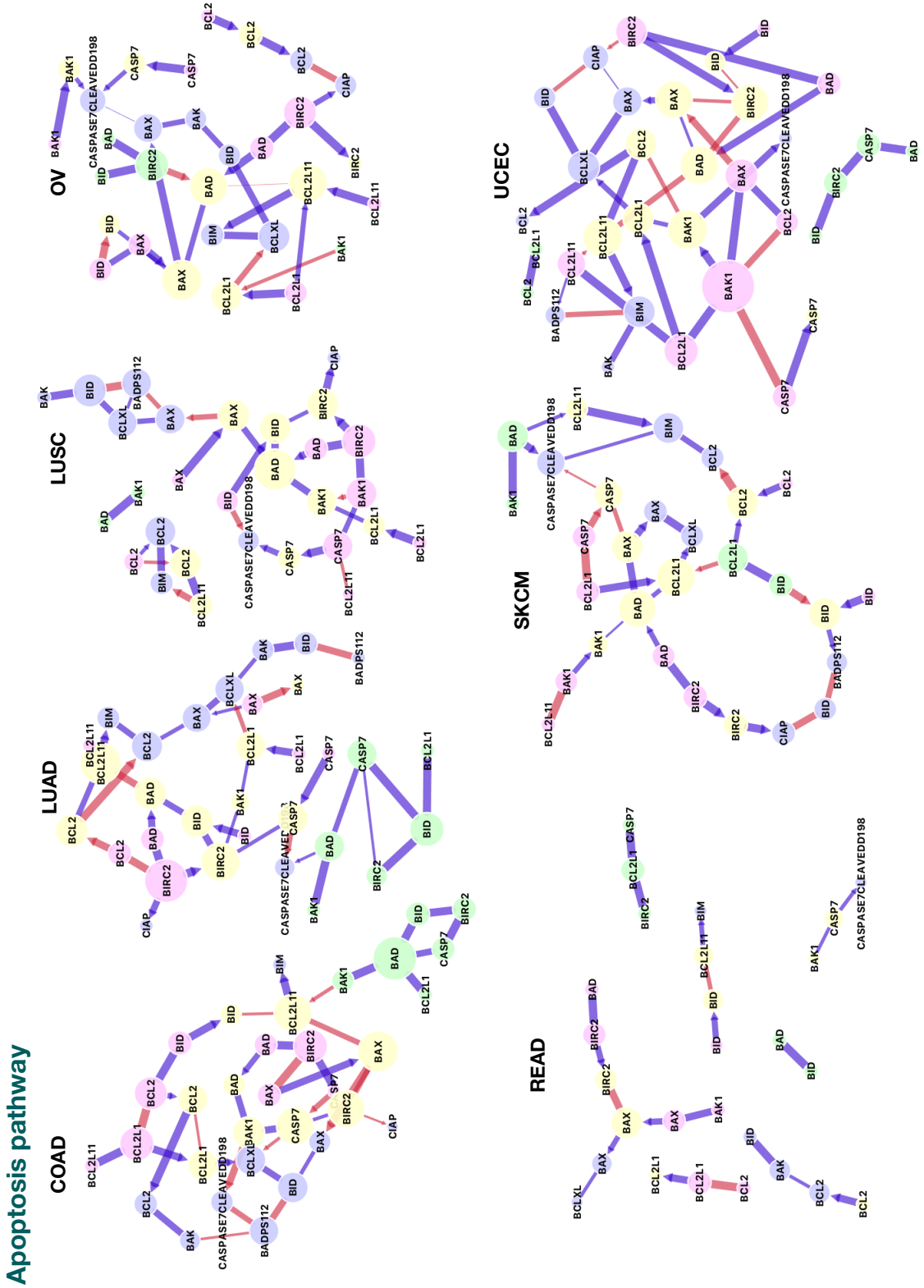

Figure S15: mlGGM for apoptosis pathway

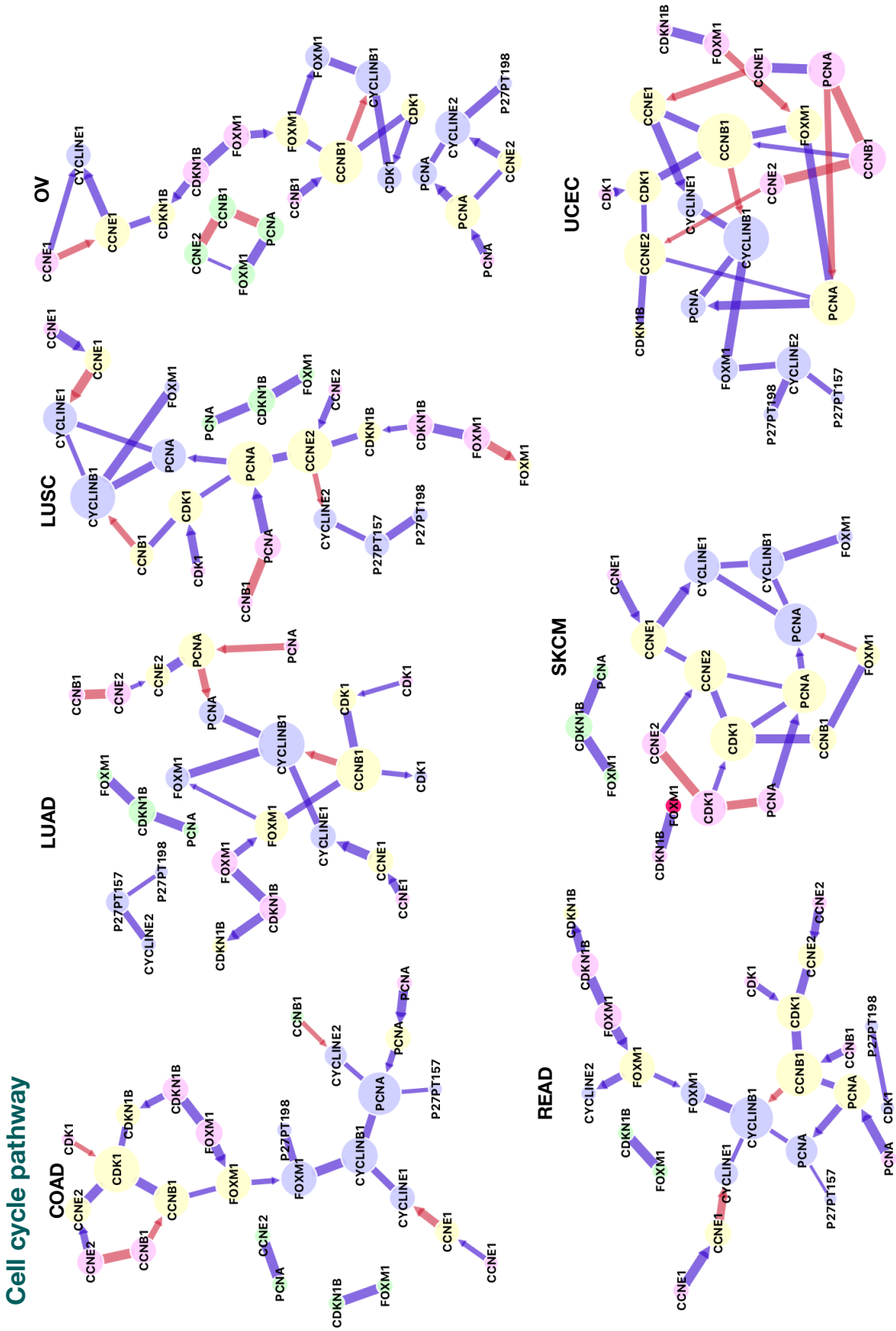

Figure S16: mlGGM for cell cycle pathway

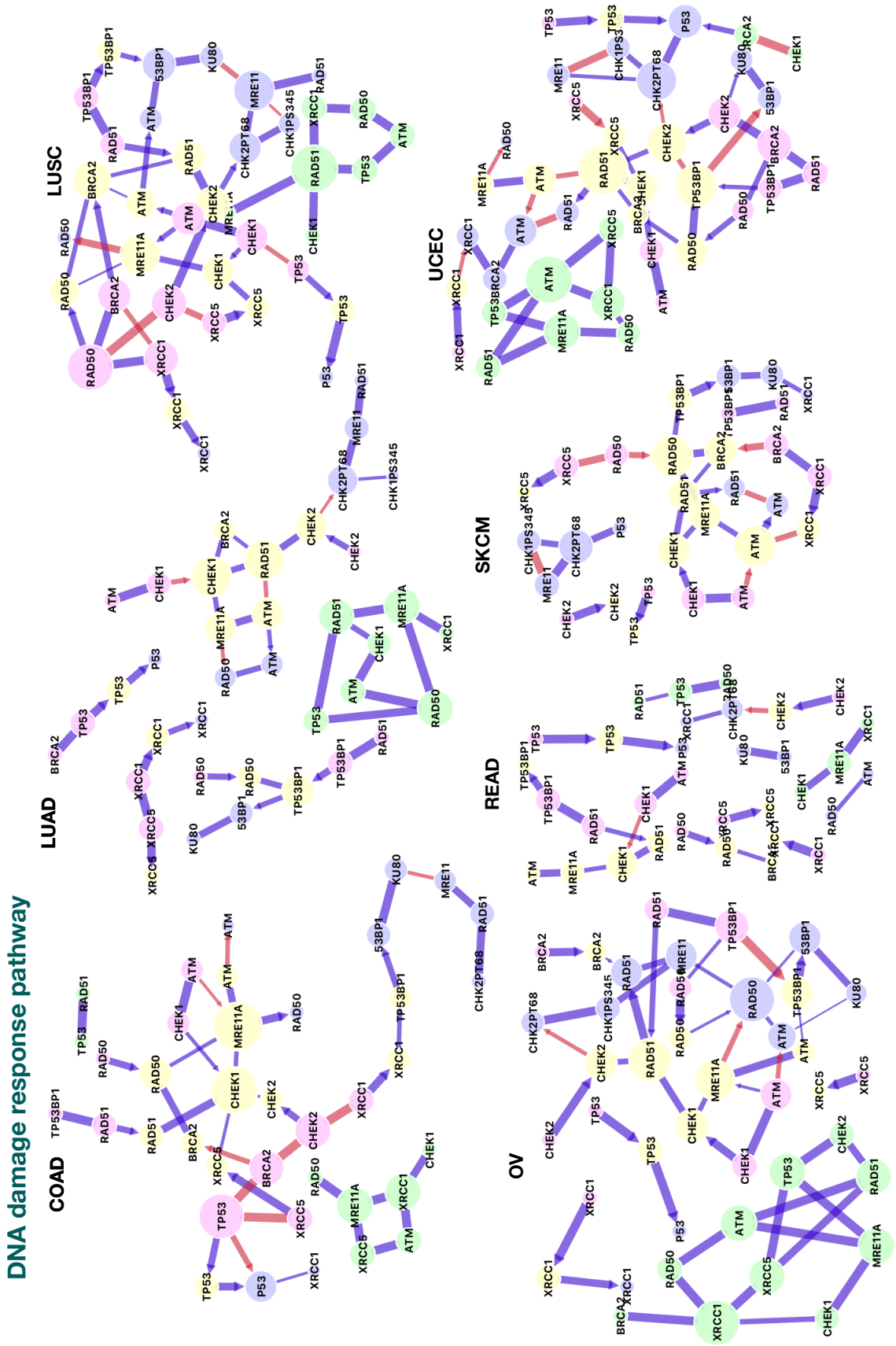

Figure S17: mlGGM for DNA damage response pathway

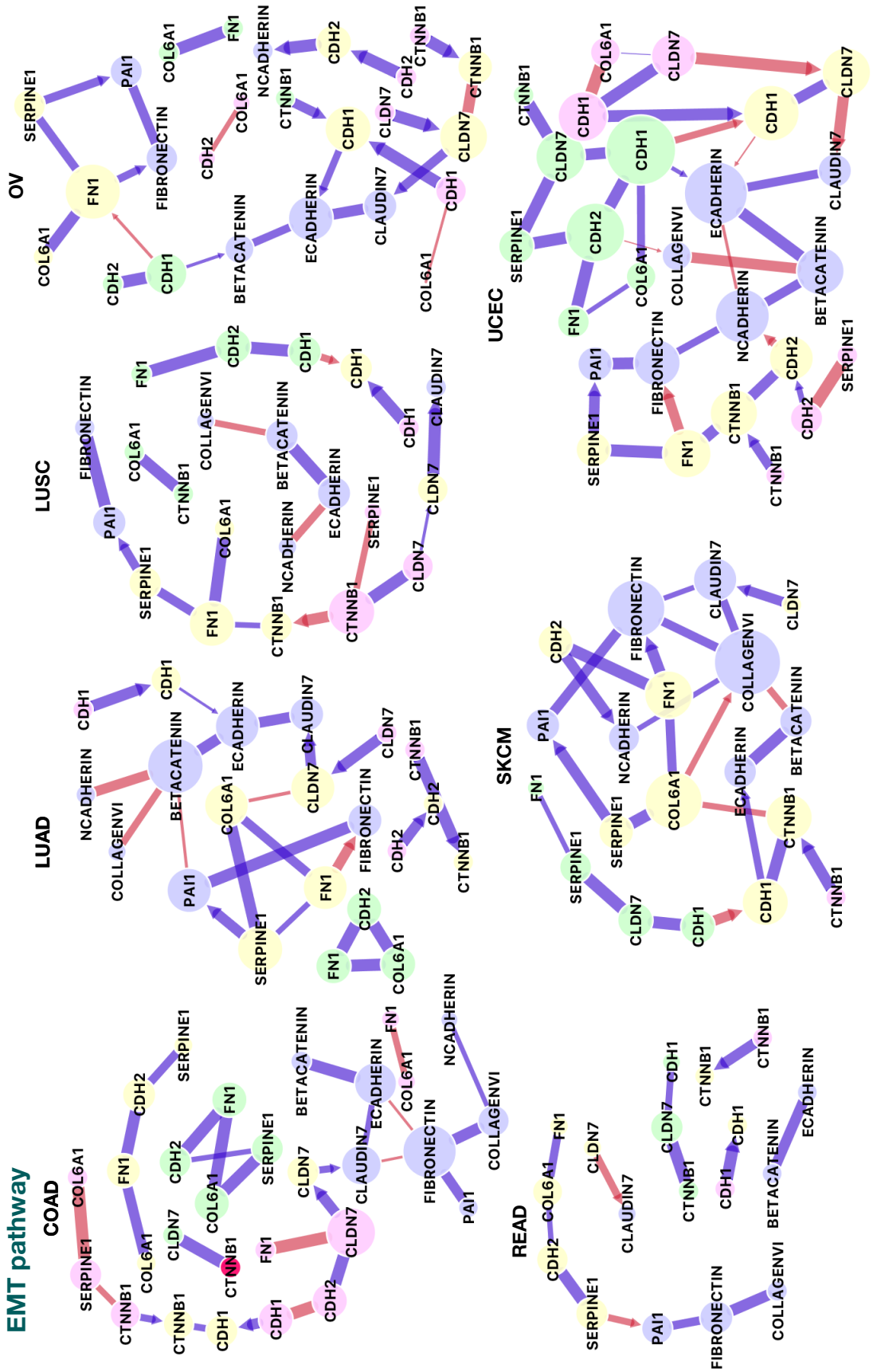

Figure S18: mlGGM for EMT pathway

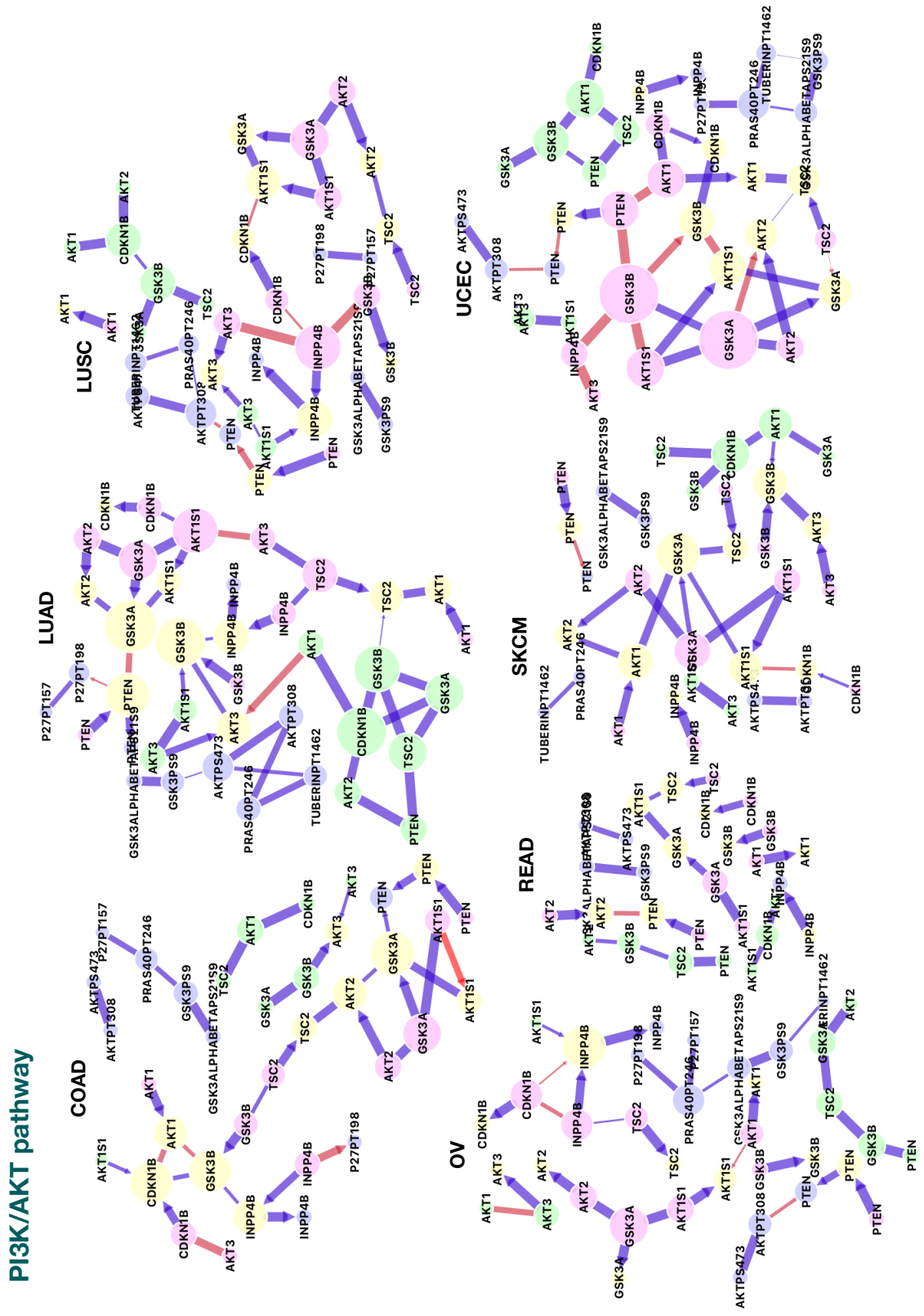

Figure S19: mlGGM for PI3K/AKT pathway

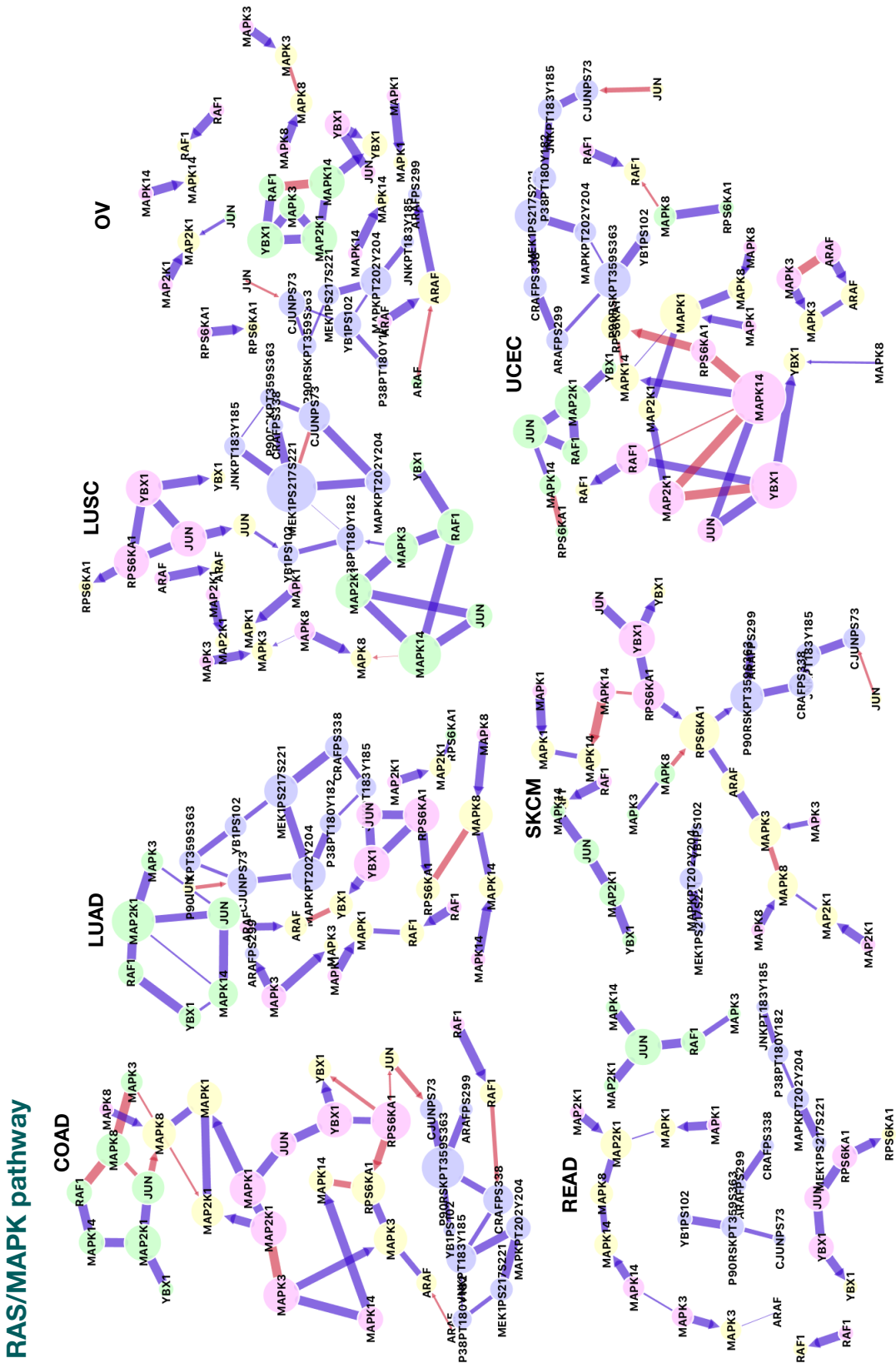

Figure S20: mlGGM for RAS/MAPK pathway

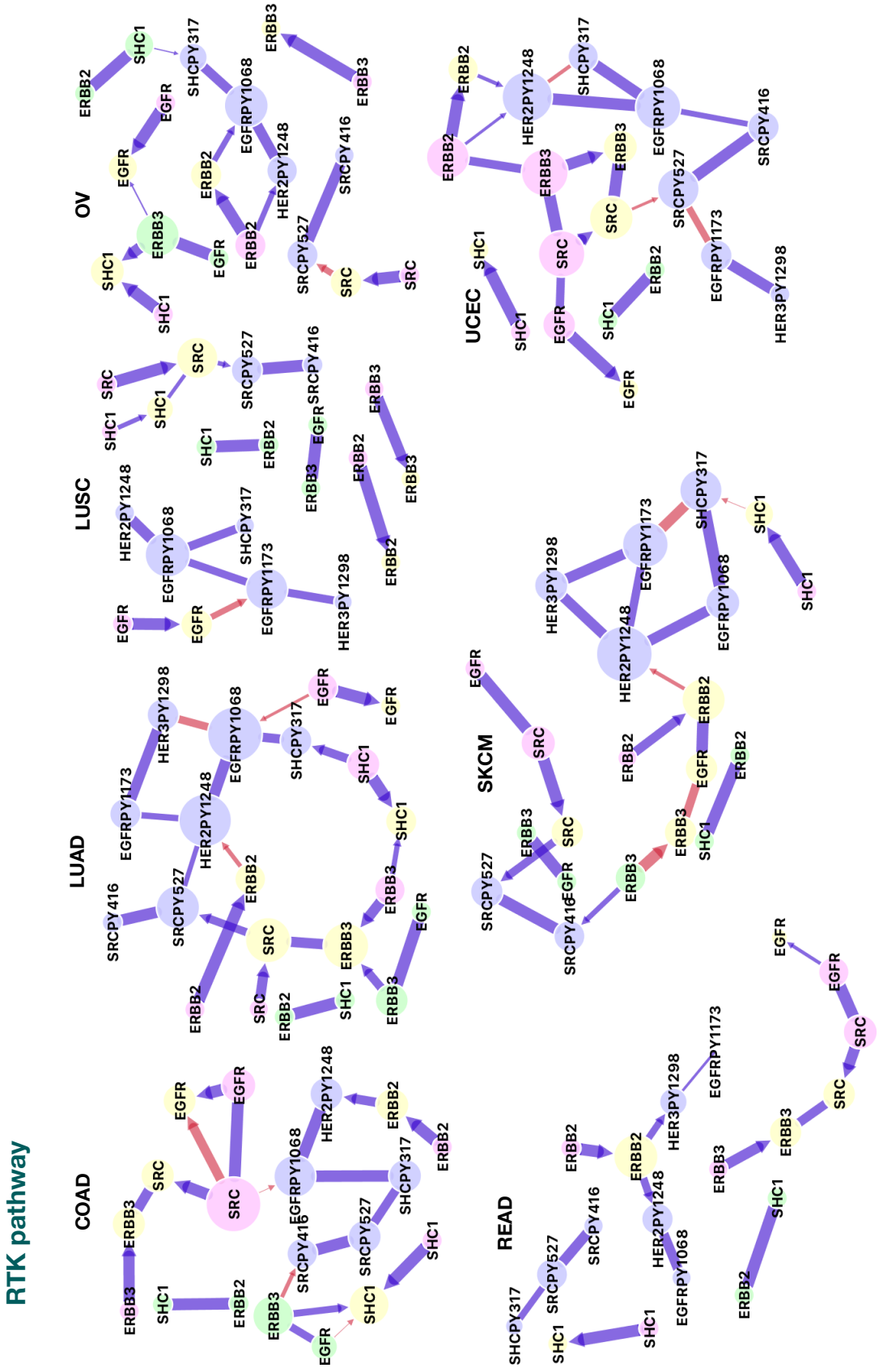

Figure S21: mlGGM for RTK pathway

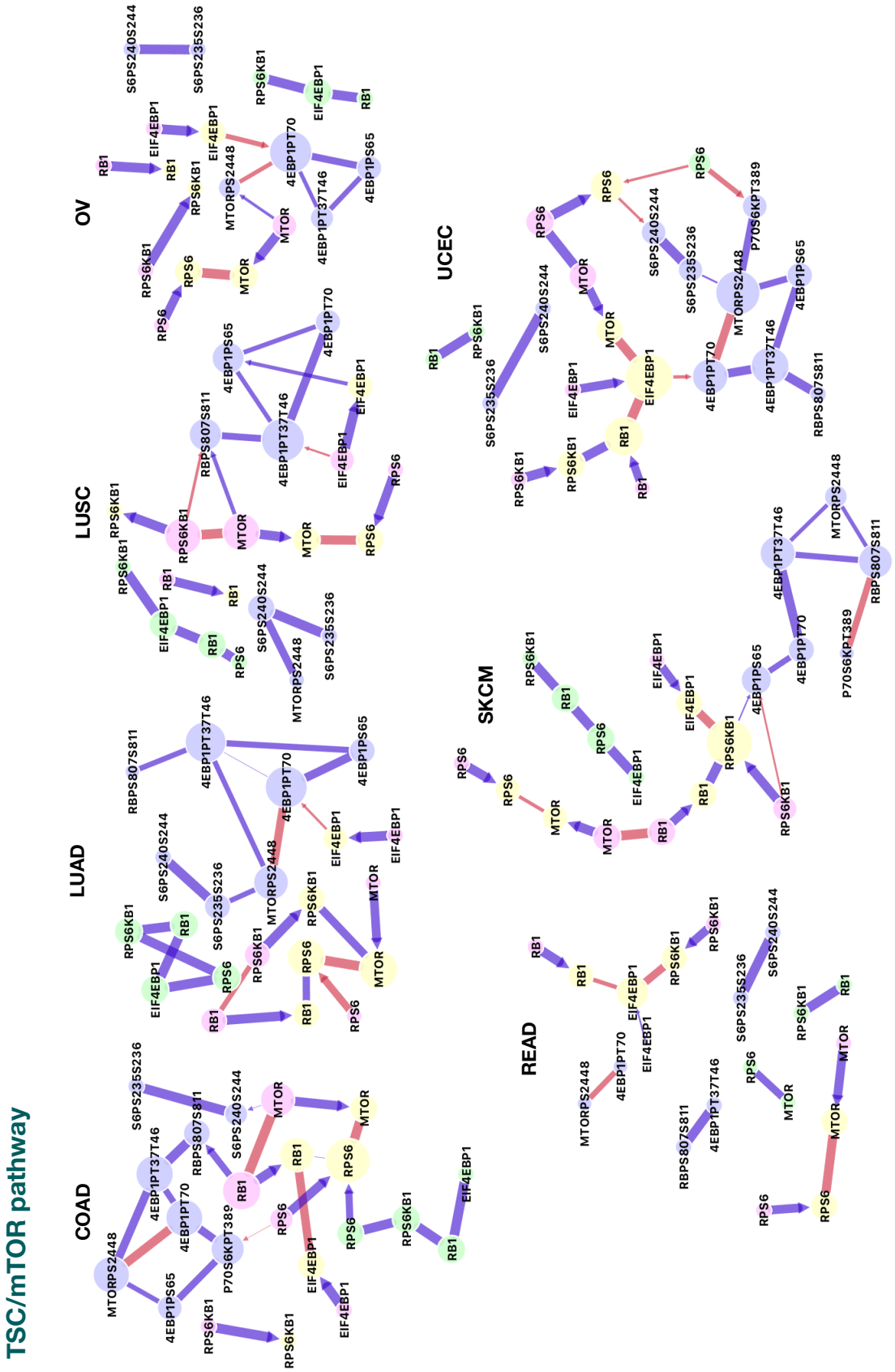

Figure S22: mlGGM for TSC/mTOR pathway

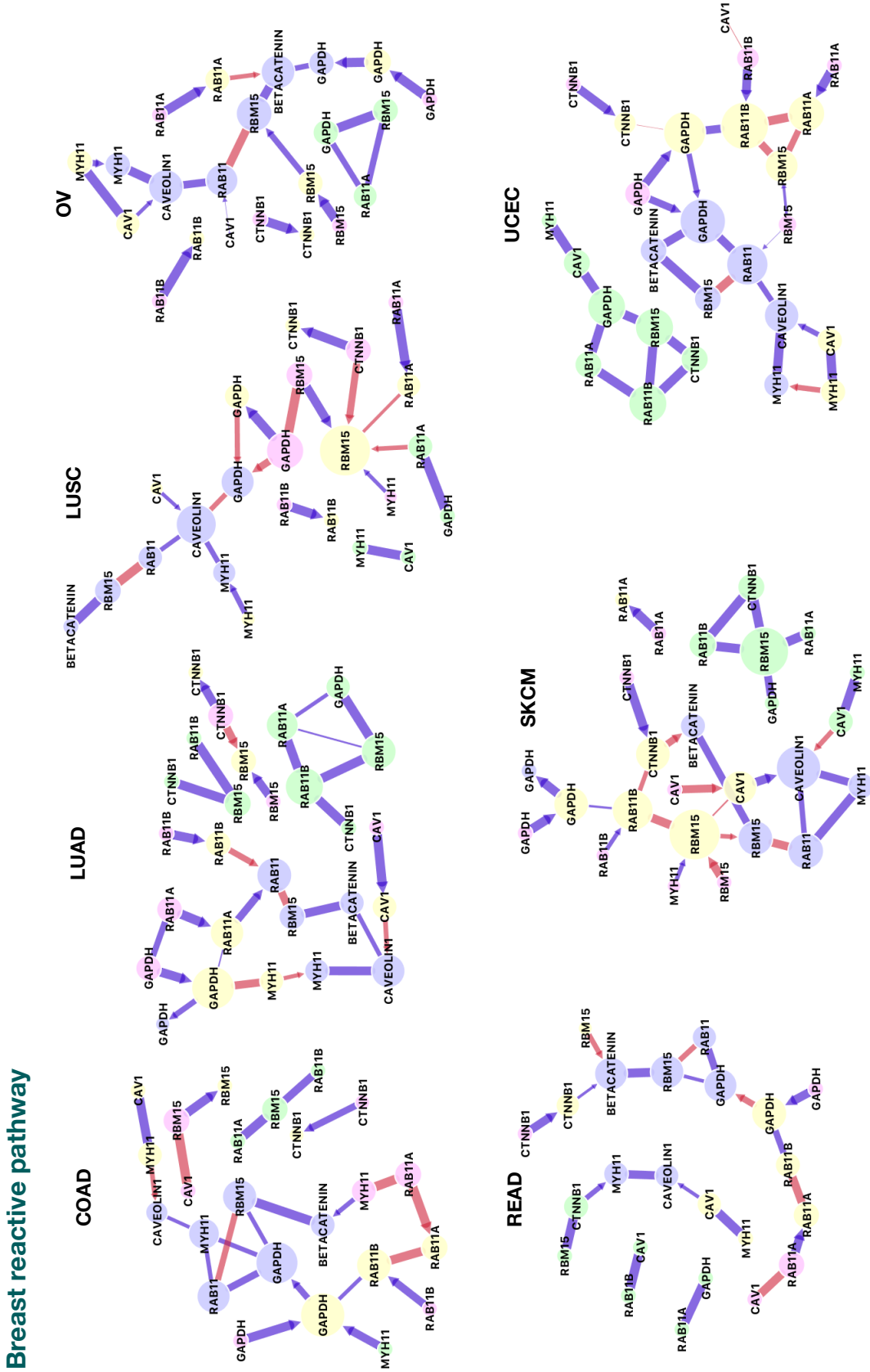

Figure S23: mlGGM for breast reactive pathway

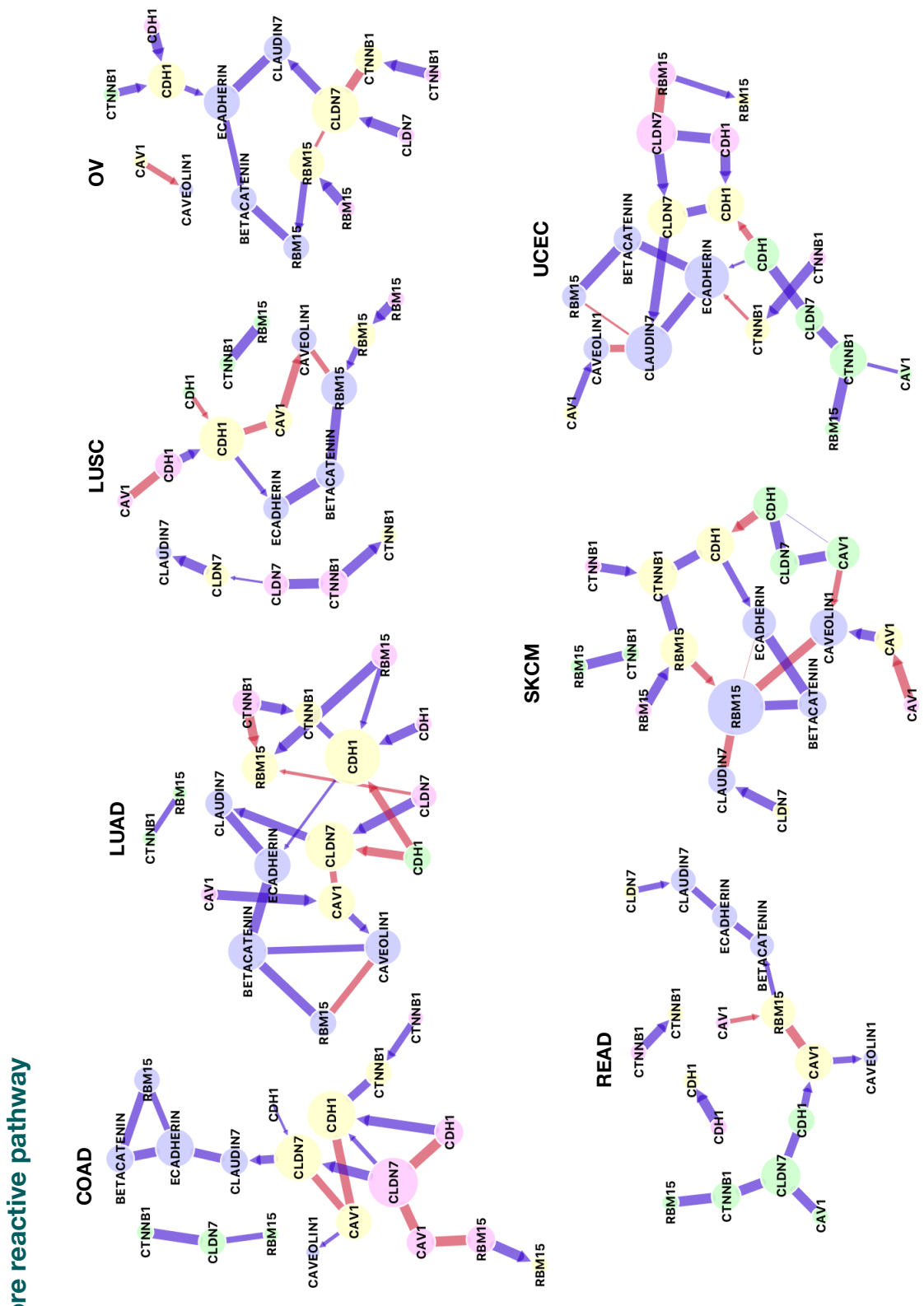

Figure S24: mlGGM for core reactive pathway

Table S1: Simulation results: model selection performances as measured by sensitivity, specificity, Matthew's correlation coefficient (MCC), the number of edges detected, partial area under the curve (pAUC) at specificity=0.8 and area under the curve (AUC), for four different simulation scenarios, based on 50 replications. Numbers in parentheses are the simulation standard errors.

Setting (p, n, q, p_E)	Method	Sensitivity	Specificity	MCC	Number of discoveries	pAUC(0.8)*	AUC
(20,200,6,0.3)	BANS	0.99 (0.042)	0.99 (0.005)	0.9 (0.041)	14.3 (0.974)	1 (0.015)	1 (0.003)
# true edges= 12	MRCE	0.99 (0.039)	0.81 (0.046)	0.46 (0.062)	45.64 (8.312)	0.79 (0.048)	0.96 (0.015)
	CAPME	1 (0.016)	0.74 (0.056)	0.39 (0.049)	58.84 (9.960)	0.74 (0.055)	0.95 (0.009)
(100,200,6,0.03)	BANS	0.94 (0.034)	1 (0.001)	0.87 (0.034)	49.62 (1.947)	1 (0.001)	1 (0)
# true edges= 43	MRCE	1 (0.008)	0.9 (0.021)	0.27 (0.029)	529.08 (100.605)	0.81 (0.015)	0.96 (0.003)
	CAPME	1 (0.005)	0.79 (0.028)	0.18 (0.014)	1087.76 (137.793)	0.85 (0.006)	0.97 (0.002)
(100,200,10,0.03)	BANS	0.97 (0.021)	1 (0.001)	0.83 (0.031)	34.66 (1.825)	1 (0.001)	1 (0)
# true edges= 25	MRCE	1 (0.008)	0.93 (0.013)	0.25 (0.025)	367.02 (64.609)	0.86 (0.008)	0.97 (0.002)
	CAPME	1 (0.008)	0.85 (0.021)	0.17 (0.014)	776.6 (105.138)	0.88 (0.011)	0.98 (0.002)
(200,100,10,0.03)	BANS	0.74 (0.024)	1 (0.002)	0.83 (0.015)	91 (3.536)	0.99 (0.002)	1 (0)
# true edges=116	MRCE	0.47 (0.050)	0.98 (0.002)	0.22 (0.017)	469.04 (40.141)	0.87 (0.009)	0.93 (0.013)
	CAPME	0.48 (0.029)	0.94 (0.009)	0.13 (0.009)	1327.98 (174.406)	0.79 (0.007)	0.96 (0.004)

* scaled to be located between 0 and 1.

Table S2: Pathways and gene names

	Pathway	Genes
1	Apoptosis	BAK1, BAX, BID, BCL2L11, CASP7, BAD, BCL2, BCL2L1, BIRC2
2	Breast reactive	CAV1, MYH11, RAB11A, RAB11B, CTNNB1, GAPDH, RBM15
3	Cell cycle	CDK1, CCNB1, CCNE1, CCNE2, CDKN1B, PCNA, FOXM1
4	Core reactive	CAV1, CTNNB1, RBM15, CDH1, CLDN7
5	DNA damage response	TP53BP1, ATM, BRCA2, CHEK1, CHEK2, XRCC5, MRE11A, TP53, RAD50, RAD51, XRCC1
6	EMT	FN1, CDH2, COL6A1, CLDN7, CDH1, CTNNB1, SERPINE1
7	PI3K/AKT	AKT1, AKT2, AKT3, GSK3A, GSK3B, CDKN1B, AKT1S1, TSC2, INPP4B, PTEN
8	RAS/MAPK	ARAF, JUN, RAF1, MAPK8, MAPK1, MAPK3, MAP2K1, MAPK14, RPS6KA1, YBX1
9	RTK	EGFR, ERBB2, ERBB3, SHC1, SRC
10	TSC/mTOR	EIF4EBP1, RPS6KB1, MTOR, RPS6, RB1

References

Andersson, S. A., Madigan, D., and Perlman, M. D. (2001). Alternative markov properties for chain graphs. *Scandinavian journal of statistics*, 28(1):33–85.

Armstrong, H. (2005). *Bayesian estimation of decomposable Gaussian graphical models*. PhD thesis, The University of New South Wales.

Barabási, A.-L. and Albert, R. (1999). Emergence of scaling in random networks. *science*, 286(5439):509–512.

Bhattacharya, A., Chakraborty, A., and Mallick, B. K. (2016). Fast sampling with gaussian scale mixture priors in high-dimensional regression. *Biometrika*, page asw042.

Consonni, G., La Rocca, L., and Peluso, S. (2017). Objective bayes covariate-adjusted sparse graphical model selection. *Scandinavian Journal of Statistics*, pages n/a–n/a. 10.1111/sjos.12273.

Cox, D. R. and Wermuth, N. (1993). Linear dependencies represented by chain graphs. *Statistical science*, pages 204–218.

Drton, M. and Eichler, M. (2006). Maximum likelihood estimation in gaussian chain graph models under the alternative markov property. *Scandinavian journal of statistics*, 33(2):247–257.

Frydenberg, M. (1990). The chain graph markov property. *Scandinavian Journal of Statistics*, pages 333–353.

Ha, M. J. and Sun, W. (2014). Partial correlation matrix estimation using ridge penalty followed by thresholding and re-estimation. *Biometrics*, 70(3):762–770.

Ha, M. J. and Sun, W. (2018). Estimation of high-dimensional directed acyclic graphs with surrogate intervention. *Biostatistics*.

Lauritzen, S. (1996). *Graphical models*, volume 17. Oxford University Press, USA.

Lauritzen, S. L. and Richardson, T. S. (2002). Chain graph models and their causal interpretations. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(3):321–348.

Lauritzen, S. L. and Wermuth, N. (1989). Graphical models for associations between variables, some of which are qualitative and some quantitative. *The Annals of Statistics*, pages 31–57.

Li, Y., Craig, B. A., and Bhadra, A. (2019). The graphical horseshoe estimator for inverse covariance matrices. *Journal of Computational and Graphical Statistics*, pages 1–24.

McCarter, C. and Kim, S. (2014). On sparse gaussian chain graph models. In *Advances in Neural Information Processing Systems*, pages 3212–3220.

Meinshausen, N. and Bühlmann, P. (2006). High-dimensional graphs and variable selection with the lasso. *The annals of statistics*, pages 1436–1462.

Peng, J., Wang, P., Zhou, N., and Zhu, J. (2009). Partial correlation estimation by joint sparse regression models. *Journal of the American Statistical Association*, 104(486):735–746.

Ritchie, M. E., Phipson, B., Wu, D., Hu, Y., Law, C. W., Shi, W., and Smyth, G. K. (2015). limma powers differential expression analyses for rna-sequencing and microarray studies. *Nucleic acids research*, 43(7):e47–e47.

Sohn, K.-A. and Kim, S. (2012). Joint estimation of structured sparsity and output structure in multiple-output regression via inverse-covariance regularization. In *AISTATS*, pages 1081–1089.

Wan, Y.-W., Allen, G. I., Baker, Y., Yang, E., Ravikumar, P., Anderson, M., and Liu, Z. (2016). Xmrf: an r package to fit markov networks to high-throughput genetics data. *BMC systems biology*, 10(3):69.