Hui Zhou and Tian Zheng*

Bayesian hierarchical graph-structured model for pathway analysis using gene expression data

Abstract: In genomic analysis, there is growing interest in network structures that represent biochemistry interactions. Graph structured or constrained inference takes advantage of a known relational structure among variables to introduce smoothness and reduce complexity in modeling, especially for high-dimensional genomic data. There has been a lot of interest in its application in model regularization and selection. However, prior knowledge on the graphical structure among the variables can be limited and partial. Empirical data may suggest variations and modifications to such a graph, which could lead to new and interesting biological findings. In this paper, we propose a Bayesian random graph-constrained model, *rGrace*, an extension from the *Grace* model, to combine *a priori* network information with empirical evidence, for applications such as pathway analysis. Using both simulations and real data examples, we show that the new method, while leading to improved predictive performance, can identify discrepancy between data and a prior known graph structure and suggest modifications and updates.

Keywords: gene expression; network analysis; Bayesian anslysis.

***Corresponding author: Tian Zheng,** Department of Statistics, Columbia University, New York, NY 10027, USA, e-mail: tzheng@stat.columbia.edu
Hui Zhou: Department of Biostatistics, Columbia University, New York, NY 10032, USA

1 Introduction

In genomics, there are many genome-wide networks constructed based on high-throughput experiments, such as protein-protein interaction networks (Franke et al., 2006) and gene synergy networks (Watkinson et al., 2008). Prior subject knowledge may lead to gains in statistical efficiency in data analysis. Indeed, there is an emerging class of methods that perform analysis based on prevailing knowledge of gene sets or modules. Baranzini et al. (2009) proposed to first identify gene subnetworks and then search for significant modules that are related to multiple sclerosis, an approach that can recover genes with a modest signal. Elbers et al. (2009) studied significantly overrepresented pathways using different pathway classification tools. Emily et al. (2009) searched for SNP interactions, but focusing only on those located near genes that have interactions, physically or functionally. However, such approaches completely rely on the quality of the *a priori* biological knowledge, which is incomplete and constantly being updated. It is therefore desirable to update such information according to data under study. Another limitation of current biological databases is that they usually indicate deterministic relations between variables (e.g., genes) that do not reflect the stochastic, highly inter-dependent, and conditional nature of biological interactions (Rzhetsky et al., 2006).

The Bayesian framework provides a natural way of utilizing empirical evidence to update prior knowledge. Network information can be introduced using a suitable prior. Werhli and Husmeier (2007) constructed priors over network structures to combine different sources of the biological prior knowledge in a Bayesian network framework. Li and Zhang (2010) imposed an Ising prior on indicators of whether individual covariates should be included in the model and related this prior to a known network structure of the covariates. Stingo et al. (2011) incorporated pathway membership and gene network information through priors on latent indicators, which determine the inclusion of both pathways and genes. Such priors lead to graph-structured dependence in variable selection. Liu and Lozano (2011) proposed a Bayesian regularization method with a graph Laplacian prior, which characterizes the dependence between variables. In this way, the structure

among the variables can be inferred directly. Such a prior also promotes graph-structured smoothness among coefficients estimates.

Li and Li (2008, 2010) proposed a regression model, *Grace*, with a penalty utilizing a given gene-gene network structure. Pan et al. (2010) proposed a similar procedure with different forms of penalty functions. In this paper, we develop a multilevel Bayesian regression model based on a variation of the Grace penalized regression model (Li and Li, 2008, 2010). Instead of using a fixed known graph structure as in Grace, we allow the graph structure to be random and adopt an informative prior centered at that *a priori* graph. Such a Bayesian formulation of penalized regression provides certain inferential benefits, such as a joint posterior distribution of the coefficients, better estimation of residual variance (Kyung et al., 2010), and potential generalization to broader model classes, answering the need of genomic data analysis (Yi and Xu, 2008). More specifically, in addition to results on individual covariates' coefficients and their predictive performance, our method is able to combine prior knowledge with empirical information in the data into a posterior distribution of graph structures. This posterior distribution may suggest a different graph as the most probable relational structure among the covariates and will also indicate the probabilities of the interaction states between two covariates (positive, negative, or no interaction). We call the new method the random graph constrained (rGrace) model.

To overcome computational complexity due to the large number of possible random graphs, we further consider possible grouping structure among the covariates. The group lasso penalty is widely adopted to induce structured sparseness (Yuan and Lin, 2006; Meier et al., 2008; Friedman et al., 2010). Pan et al. (2010) studied group penalty based on L_γ norm with $\gamma>1$. To encourage a grouping structure, in our penalized regression model (rGrace), instead of using the conventional L_1 plus weighted L_2 penalty, we use a group lasso penalty plus a weighted L_2 penalty with grouping decided by the connected subgraphs.

This rest of the paper is organized as follows. Section 2 describes the model and the Markov chain Monte Carlo (MCMC) procedure for model inference. Section 3 provides a simulation study that compares our method to the Grace/aGrace procedures, followed by a real data application to brain aging in Section 4. Section 5 concludes the paper with a discussion.

2 Methods

2.1 Notation

Let $\mathbf{X}=[x_{ij}]\in\mathbb{R}^{n\times p}$ denote the matrix of gene expression measurements, with x_{ij} being the j th gene for the i th individual, while $\mathbf{Y}=(y_1,\ldots,y_n)^T\in\mathbb{R}^n$ denotes the response vector for n individuals. Assume the following model:

$$\mathbf{Y}=\mathbf{X}\boldsymbol{\beta}+\boldsymbol{\varepsilon},$$

where $\boldsymbol{\varepsilon}\sim N(0,\sigma^2\mathbf{I})$ and $\boldsymbol{\beta}=(\beta_1,\ldots,\beta_p)^T$. Throughout the paper, we assume that the response vector $\mathbf{Y}$ is centered at zero, and the measurement matrix $\mathbf{X}$ is normalized so that each covariate is centered at zero and $\sum_{i=1}^n x_{ij}^2=n-1$ for $j=1,\ldots,p$. If $\mathbf{X}$ can be naturally partitioned into J groups, corresponding to a certain graph structure, we assume $\mathbf{X}=(\mathbf{X}_1,\mathbf{X}_2,\ldots,\mathbf{X}_J)$, where $\mathbf{X}_j$ is an $n\times p_j$ matrix, $\sum_{j=1}^J p_j=p$. The coefficient corresponding to $\mathbf{X}_j$ is denoted $\boldsymbol{\beta}_j$.

Consider a labeled and unweighted graph $G=(V, \mathrm{E})$ with P nodes, representing a known fixed graph based on prior knowledge, where $V=\{1,\ldots,p\}$, each node corresponding to one covariate (gene), and $E=\{u\sim v\}$ is the set of edges, representing the relational structure among the covariates. Two nodes are considered adjacent if they are connected by an edge in the graph. Let $\mathbf{A}$ be the $p\times$p adjacency matrix such that $A_{u,v}$ equals one if and only if u and v are adjacent and zero otherwise. Let d_u be the degree for node u, i.e., the number of edges incident to u, and let $\mathbf{D}$ be the $p\times p$ diagonal matrix with $D_{u,u}=d_u$. Define the Laplacian matrix $\tilde{\mathbf{L}}=\mathbf{D}-\mathbf{A}$ of G as

$$\tilde{L}_{u,v}=\begin{cases} d_u & \text{if } u=v \text{ and } d_u\neq 0,\\ -1 & \text{if } u \text{ and } v \text{ are adjacent},\\ 0 & \text{otherwise.}\end{cases}$$

The normalized Laplacian matrix **L** (Chung, 1997) is defined as follows:

$$L_{u,v}=\begin{cases} 1 & \text{if } u=v \text{ and } d_u\neq 0,\\ -1/\sqrt{d_u d_v} & \text{if } u \text{ and } v \text{ are adjacent},\\ 0 & \text{otherwise.}\end{cases}$$

Both the Laplacian and the normalized Laplacian matrix are semi-positive definite.

2.2 Random graph constrained (rGrace) model

Li and Li (2008) introduced the graph-constrained estimation of regression coefficients (Grace), defined as

$$\begin{aligned}\hat{\beta}_{Grace}&=\underset{\beta}{\operatorname{argmin}}\left\{||\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}||_2^2+\lambda_1||\boldsymbol{\beta}||_1+\lambda_2\boldsymbol{\beta}^T\mathbf{L}\boldsymbol{\beta}\right\}\\&=\underset{\beta}{\operatorname{argmin}}\left\{||\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}||_2^2+\lambda_1||\boldsymbol{\beta}||_1+\lambda_2\sum_{u\sim v}\left(\frac{\beta_u}{\sqrt{d_u}}-\frac{\beta_v}{\sqrt{d_v}}\right)^2\right\},\end{aligned}$$

where, an L_1 penalty is used for sparseness and a weighted L_2 penalty is used to introduce smoothness in the coefficients along the edges of the graph, for better generative performance in prediction. In addition, Li and Li (2010) proposed another procedure, adaptive Grace (aGrace), which allows the regression coefficients of linked covariates to take opposite signs. The signs were determined by an initial step of ordinary least-square or elastic net regression (Zou and Hastie, 2005) that produces an estimate $\tilde{\boldsymbol{\beta}}$. More specifically,

$$\hat{\beta}_{aGrace}=\underset{\beta}{\operatorname{argmin}}\left\{||\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}||_2^2+\lambda_1||\boldsymbol{\beta}||_1+\lambda_2\boldsymbol{\beta}^T\mathbf{S}\mathbf{L}\mathbf{S}\boldsymbol{\beta}\right\},$$

where $\mathbf{S}=diag(sign(\tilde{\beta}_1),...,sign(\tilde{\beta}_p))$.

Biological studies have shown that gene networks consist of modules defined as genes that are regulated together as a group (Segal et al., 2003). Bar-Joseph et al. (2003) suggested that gene modules that partition the genetic network aid in the reduction of graph complexity without significant loss of explanatory power and interpreted genes within a same module as having a common biological function. Langfelder and Horvath (2008) advocated analyzing highly connected modules as a biologically motivated data reduction approach. Gene modules can be formed based on expression profiles (Segal et al., 2003; Langfelder and Horvath, 2008; Kim et al., 2011). However, Ravasz et al. (2002) suggested that topological similarity can be used to define more stable gene modules and Bar-Joseph et al. (2003) argued that genes with similar expression patterns could be governed by distinct regulatory mechanisms. Multiple approaches have been proposed to discovery gene modules directly based on adjacency matrix of the genes (Newman, 2006; Yip and Horvath, 2007; Ruan and Zhang, 2008). In particular, Yip and Horvath (2007) developed a node dissimilarity measure to identify nodes that have high topological overlap. For a large graph that is partitioned into several connected components, a natural way to extend Grace/aGrace is to consider group-Grace/group-aGrace,

$$\begin{aligned}\hat{\beta}_{group-Grace}&=\underset{\beta}{\operatorname{argmin}}\left\{||\mathbf{Y}-\mathbf{X}\beta||_2^2+\lambda_1\sum_{j=1}^{J}\sqrt{p_j}||\boldsymbol{\beta}_j||_2+\lambda_2\beta^T\tilde{\mathbf{L}}\beta\right\},\\\hat{\beta}_{group-aGrace}&=\underset{\beta}{\operatorname{argmin}}\left\{||\mathbf{Y}-\mathbf{X}\beta||_2^2+\lambda_1\sum_{j=1}^{J}\sqrt{p_j}||\boldsymbol{\beta}_j||_2+\lambda_2\beta^T\mathbf{S}\tilde{\mathbf{L}}\mathbf{S}\beta\right\}.\end{aligned}$$

The grouping based on connected subgraphs automatically leads to a block diagonal Laplacian matrix $\tilde{\mathbf{L}}=diag(\tilde{\mathbf{L}}_1,\ldots,\tilde{\mathbf{L}}_J)$. It is expected that connected genes share related biological functions as well as similar regression coefficients (Zhang and Horvath, 2005; Liu et al., 2013). Therefore, in this paper, we use the Laplacian matrix $\tilde{\mathbf{L}}$ in our model instead of the normalized Laplacian matrix $\mathbf{L}$, since we found the results from the former easier to interpret biologically. Our method is not affected by this choice.

Penalized regression models have been adapted to the Bayesian framework by choosing suitable priors, as for the Bayesian lasso (Park and Casella, 2008), Bayesian adaptive lasso (Griffin and Brown, 2007; Sun et al., 2009), Bayesian elastic net (Li and Lin, 2010), and Bayesian group lasso (Raman et al., 2009). Kyung et al. (2010) gave an overview of the Bayesian formulation of penalized regression methods and also gave full conditionals for Bayesian fused lasso.

We now introduce the random graph constrained model (rGrace) as a multilevel model extension of Grace under the Bayesian framework. Following Park and Casella (2008) and Li and Lin (2010), we consider a fully Bayesian hierarchical model (conditioning on $\mathbf{X}$ is implicit):

$$\begin{aligned}
\mathbf{Y}|\boldsymbol{\beta},\sigma^2 &\sim N(\mathbf{X}\boldsymbol{\beta},\sigma^2\mathbf{I}),\\
p(\boldsymbol{\beta}|\sigma^2,G,\lambda_1,\lambda_2) &\propto \exp\left\{-\frac{1}{2\sigma^2}\left(\lambda_1\sum_{j=1}^{J}\sqrt{p_j}\,||\boldsymbol{\beta}_j||_2+\lambda_2\boldsymbol{\beta}^T\tilde{\mathbf{L}}\boldsymbol{\beta}\right)\right\},\\
\sigma^2 &\sim Gamma(\alpha,\theta),\\
\lambda_1^2 &\sim Gamma(\alpha_1,\theta_1),\\
\lambda_2 &\sim Gamma(\alpha_2,\theta_2),\\
G &\sim \prod_{i<j} Pr(A_{ij}).
\end{aligned}$$

The form of $\tilde{\mathbf{L}}$ is

$$\begin{pmatrix}
\sum_{j\neq 1}|A_{1j}| & -A_{12} & \cdots & -A_{1p}\\
-A_{21} & \sum_{j\neq 2}|A_{2j}| & \cdots & -A_{2p}\\
\vdots & \vdots & \ddots & \vdots\\
-A_{p1} & -A_{p2} & \cdots & \sum_{j\neq p}|A_{pj}|
\end{pmatrix},$$

with $A_{ij}=A_{ij}$.

The gamma prior on σ^2 is proper but vague with a small positive α and a large θ. In addition, gamma priors on λ_1^2 (not λ_1) and λ_2 permit easier implementation via the Gibbs sampler shown in the next section. Hyperparameters α_1, θ_1, α_2, and θ_2 are set such that during the MCMC procedure the ranges of sampled λ_1, λ_2 are comparable to the range of the searching grid when solving group-Grace or group-aGrace. We also examine the sensitivity of the inference results to the value of these hyperparameters, including α and θ, by running a parallel analysis on a few combinations of hyperparameters in the simulation study later on. Through imposing a prior on each element of the adjacency matrix corresponding to a graph G with p nodes, independent of σ^2, we may overcome the drawback of a fixed graph structure based on an incomplete knowledge on biological pathways.

The aGrace estimator of Li and Li (2010) was motivated by the fact that two adjacent genes might have opposite effects on Y. Furthermore, gene regulatory networks explain the causality of gene expression regulation via activators and suppressors. Mason et al. (2009) also reported the advantage of allowing positive and negative signs in gene co-expression networks. Therefore, we expect the regression coefficients, β, of two linked genes to show identical or opposite signs, depending on the underlying functional relation. For each edge in the graph, we allow the corresponding entry in the adjacency matrix to have a sign. Specifically, between any two nodes, there might be a positive edge (A_{ij}=1), a negative edge (A_{ij}=–1), or no edge (A_{ij}=0). The Laplacian matrix equals D–A, as previously defined, and remains semipositive-definite.

For gene expression analysis, there are publicly available genomic network databases, such as KEGG, that provide information on whether an edge exists between two nodes (genes). However, the sign for an edge is usually not provided and is treated as positive by default. Denote the initial graph structure as G^0 with adjacency matrix A^0. We aim to update a graph structure using information from empirical data, while

maintaining high confidence in the prior knowledge. We achieve this by adopting the following informative prior on A_{ij}. Given two cut off value $-1< c_l<0$, and $0<c_u<1$, if $A^0_{ij}=0$, then

$$A_{ij|b_0}=\begin{cases}-1 & \text{with prob. } \mathrm{P}(Z_0\in[-1,c_l)),\\ 0 & \text{with prob. } \mathrm{P}(Z_0\in[c_l,c_u)),\\ 1 & \text{with prob. } \mathrm{P}(Z_0\in[c_u,1]),\end{cases}$$

where Z_0 follows a scaled beta distribution with parameters (b_0,b_0). Here a scaled beta distribution is defined as two times beta distribution minus one, so that it ranges from negative one to one. The scaled beta distribution with parameters (α,β) has density

$$p(y;\alpha,\beta)=\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)2^{\alpha+\beta-1}}(1+y)^{\alpha-1}(1-y)^{\beta-1},-1\le y\le 1.$$

Or if $A^0_{ij}=1$,

$$A_{ij|b_1}=\begin{cases}-1 & \text{with prob. } \mathrm{P}(Z_1\in[-1,c_l)),\\ 0 & \text{with prob. } \mathrm{P}(Z_1\in[c_l,c_u)),\\ 1 & \text{with prob. } \mathrm{P}(Z_1\in[c_u,1]),\end{cases}$$

where Z_1 follows a scaled beta distribution with parameters (b_1, b_1). Hyperpriors are distributed as

$$b_0\sim\mathrm{Unif}(1,B_0),$$
$$b_1\sim\mathrm{Unif}(B_1,1),$$

given hyperparameters $B_0>1$ and $B_1<1$. Essentially, we assume A_{ij} is the truncation of a continuous latent variable following a scaled symmetric beta distribution. If $A^0_{ij}=0$, then the latent beta distribution has a shape parameter b_0 larger than one with the mode at zero; otherwise the latent beta distribution has a shape parameter b_1 smaller than one, with two peaks at negative one and one. This idea is depicted in Figure 1. Such a prior structure discourages removing or adding an edge between any two nodes and induces equal probabilities for the edge sign. Hyperpriors B_0 and B_1 control the informativeness of the prior.

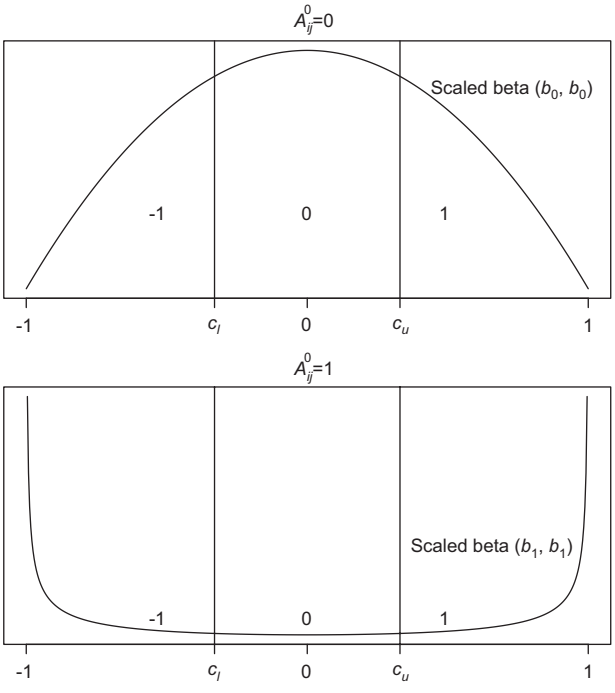

Figure 1 Initial edge and latent scaled beta distribution.

2.3 MCMC procedure for rGrace model inference

Motivated by the connection between the Laplace distribution and scale mixture of normal distributions (Andrews and Mallows, 1947), Park and Casella (2008) connected the lasso to the Bayesian paradigm. Also making use of this connection, we introduce instrumental variables s=(s_1,..., s_J) and treat $\boldsymbol{\beta}|\sigma^2$,$G$ alternatively as $\int_s p(\boldsymbol{\beta}|\sigma^2,\mathrm{s},G)\,p(\mathrm{s}|\sigma^2,G)ds$, where

$$\begin{aligned}\boldsymbol{\beta}|\mathbf{s},\sigma^2,G &\sim N\left(0,\left(\mathbf{D}_s+\frac{\lambda_2}{\sigma^2}\tilde{\mathbf{L}}\right)^{-1}\right),\\ p(\mathbf{s}|\sigma^2,G) &\propto \prod_{j=1}^{J}\frac{I(s_j>0)}{\sqrt{s_j}\prod_{l=1}^{p_j}\left(\sqrt{\frac{1}{s_j}+\frac{\lambda_2 w_{j_l}}{\sigma^2}}\right)}\exp\left(-\frac{\lambda_1^2 p_j}{8\sigma^4}s_j\right).\end{aligned}$$

Here, $\mathbf{D}_s$ is a block diagonal matrix with J blocks, and the jth block $\mathbf{D}_{s_j}$ is $\frac{1}{s_j}\mathbf{I}_{p_j\times p_j}$ and w_{j_l} is the lth eigenvalue of matrix $\tilde{L}_j$. The derivation is given in the Appendix.

Note that $p(\boldsymbol{\beta}|\,\mathrm{s},\sigma^2,G)$ is proper since $\mathbf{D}_s+\frac{\lambda_2}{\sigma^2}\tilde{\mathbf{L}}$ is positive definite. In addition, for any j,

$$\int_0^\infty \frac{1}{\sqrt{s}\prod_{l=1}^{p_j}\left(\sqrt{\frac{1}{s}+\frac{\lambda_2 w_{j_l}}{\sigma^2}}\right)}\exp\left(-\frac{\lambda_1^2 p_j}{8\sigma^4}s\right)ds<\int_0^\infty s^{\left(\frac{p_j+1}{2}-1\right)}\exp\left(-\frac{\lambda_1^2 p_j}{8\sigma^4}s\right)ds<\infty.$$

Hence, $p(\mathrm{s}|\sigma^2)$ is proper, and the same is so for the prior

$$p(\boldsymbol{\beta},\mathbf{s},\sigma^2,G)=p(\boldsymbol{\beta}|\mathbf{s},\sigma^2)\,p(\mathbf{s},\sigma^2)\,p(\sigma^2)\,p(G).$$

In fact,

$$\begin{aligned}p(\boldsymbol{\beta},\mathbf{s},\sigma^2|G)=(2\pi)^{-\frac{p}{2}}\exp\left[-\frac{1}{2}\boldsymbol{\beta}^T\left(\mathbf{D}_s+\frac{\lambda_2}{\sigma^2}\tilde{\mathbf{L}}\right)\boldsymbol{\beta}\right]\prod_{j=1}^{J}m_{s_j}(\sigma^2)^{-1}\\ \frac{1}{\sqrt{s_1\cdots s_J}}\exp\left(-\frac{\lambda_1^2}{8\sigma^4}\sum_{j=1}^{J}p_j s_j\right)\frac{(\sigma^2)^{\alpha-1}}{\Gamma(\alpha)\theta^\alpha}\exp\left(-\frac{\sigma^2}{\theta}\right),\end{aligned}$$

where

$$m_{s_j}(\sigma^2)=\int_0^\infty\frac{1}{\sqrt{s_j}\prod_{l=1}^{p_j}\left(\sqrt{\frac{1}{s_j}+\frac{\lambda_2 w_{j_l}}{\sigma^2}}\right)}\exp\left(-\frac{\lambda_1^2 p_j}{8\sigma^4}s_j\right)ds_j. \tag{1}$$

The MCMC algorithm for the rGrace computation contains the following two main steps:

1. *Update parameters given a fixed graph structure G.* Given a graph structure G, it is straightforward to compute the full conditional distributions.
   - Sample $\boldsymbol{\beta}$, given other parameters:

     $$\boldsymbol{\beta}|\mathbf{Y},\sigma^2,\mathrm{s},\lambda_1,\lambda_2,G,b_0,b_1\sim N(\mathbf{U}^{-1}\mathbf{X}^T\mathbf{Y},\sigma^2\mathbf{U}^{-1}),$$

     where $\mathbf{U}=\mathbf{X}^T\mathbf{X}+\mathbf{D}_s\sigma^2+\lambda_2\tilde{\mathbf{L}}$.
   - Sample σ^2, given other parameters:

$$p(\sigma^2|\mathbf{Y},\boldsymbol{\beta},\mathbf{s},\lambda_1,\lambda_2,G,b_0,b_1)\propto\left(\frac{1}{\sigma^2}\right)^{n/2+1-\alpha}$$
$$\prod_{j=1}^{J}\frac{I(s_j>0)}{m_{s_j}(\sigma^2)}\exp\left(-\frac{||\mathbf{Y}-\mathbf{X}\boldsymbol{\beta}||_2^2}{2\sigma^2}-\frac{\lambda_2}{2\sigma^2}\boldsymbol{\beta}^T\tilde{\mathbf{L}}\boldsymbol{\beta}-\frac{\lambda_1^2}{8\sigma^4}\sum_{j=1}^{J}p_j s_j-\frac{\sigma^2}{\theta}\right).$$

To sample from $\sigma^2|\mathbf{Y},\boldsymbol{\beta}$, s, λ_1, λ_2, G, b_0, b_1, we apply the Metropolis-Hastings algorithm given an appropriate proposal density and evaluate (1) by numerical integration. For simplicity, we use a normal density with modest variance (always rejecting negative samples) as proposal densities. Computationally, with the block diagonal structure of the network, we are able to accurately evaluate (1) as a product of J integrals of one-dimensional functions. Without this assumption, we would have to deal with a single p-dimensional integral, which is usually numerically infeasible.

– Sample s, given other parameters:

$$p(\mathbf{s}|\mathbf{Y},\boldsymbol{\beta},\sigma^2,\lambda_1,\lambda_2,G,b_0,b_1)$$
$$\propto exp\left(\frac{1}{2}\boldsymbol{\beta}^T D_s\boldsymbol{\beta}\right)\prod_{j=1}^{J}\left(\frac{I(s_j>0)}{\sqrt{s_j}}exp\left(-\frac{\lambda_1^2 p_j}{8\sigma^4}s_j\right)\right)$$
$$\propto\prod_{j=1}^{J}\left(\frac{I(s_j>0)}{\sqrt{s_j}}exp\left(-\frac{\lambda_1^2 p_j}{8\sigma^4}s_j-\frac{\boldsymbol{\beta}_j^T\boldsymbol{\beta}_j}{2s_j}\right)\right).$$

Therefore,

$$\mathbf{s}|\mathbf{Y},\boldsymbol{\beta},\sigma^2,\lambda_1,\lambda_2,G,b_0,b_1\sim\prod_{j=1}^{J}GIG\left(a=\frac{\lambda_1^2 p_j}{4\sigma^4},b=\boldsymbol{\beta}_j^T\boldsymbol{\beta}_j,p=\frac{1}{2}\right),$$

where GIG(a,b,p) stands for the generalized inverse Gaussian distribution with the density

$$f(x;a,b,p)=\frac{(a/b)^{p/2}}{2K_p(\sqrt{ab})}x^{p-1}\exp\left(-\frac{1}{2}\left(ax+\frac{b}{x}\right)\right),x>0$$

with K_p a modified Bessel function of the third kind. In particular,

$$K_{\frac{1}{2}}(x)=\sqrt{\frac{\pi}{2x}}\exp(-x).$$

To sample from $\mathbf{s}|\mathbf{Y},\boldsymbol{\beta},\sigma^2,G,b_0,b_1$, the product of generalized inverse Gaussian distributions, we make use of the *R* function *rgig* in the package *HyperbolicDist*.

– Sample λ_1^2 given other parameters

$$p\left(\lambda_1^2|\mathbf{Y},\boldsymbol{\beta},\sigma^2,\mathbf{s},\lambda_2,G,b_0,b_1\right)\propto\frac{\left(\lambda_1^2\right)^{\alpha_1-1}}{\prod_{j=1}^{J}m_{s_j}}exp\left(-\left(\theta_1+\frac{\sum_{j=1}^{J}p_j s_j}{8\sigma^4}\right)\lambda_1^2\right).$$

We again apply the Metropolis-Hastings algorithm with proposal density $Gamma\left(\alpha_1,\theta_1+\frac{\sum_{j=1}^{J}p_j s_j}{8\sigma^4}\right)$ to sample λ_1^2.

– Sample λ_2 given other parameters

$$p(\lambda_2|\mathbf{Y},\boldsymbol{\beta},\sigma^2,\mathbf{s},\lambda_1,G,b_0,b_1)\propto\frac{\lambda_2^{\alpha_2-1}}{\prod_{j=1}^{J}m_{s_j}}exp\left(-\left(\theta_2+\frac{\boldsymbol{\beta}^T\tilde{\mathbf{L}}\boldsymbol{\beta}}{2\sigma^2}\right)\lambda_2\right).$$

Similarly, it is sampled via the Metropolis-Hastings algorithm with proposal density $Gamma\left(\alpha_2,\theta_2+\frac{\boldsymbol{\beta}^T\tilde{\mathbf{L}}\boldsymbol{\beta}}{2\sigma^2}\right)$

2. *Update graph structure G.* When updating the graph structure, or equivalently the adjacency matrix, we constrain the model space to all graphs with the same group membership as the initial structure by allowing any deletion of edges but only the addition of edges within each group. Deletion of edges may result in isolated vertices or unconnected components within a group.
   – Sample A, given other parameters:

Direct sampling from a conditional posterior is difficult. To obtain an efficient Gibbs sampler, we first augment the parameter space by defining the latent variable a_{ij} corresponding to A_{ij}, where a_{ij} follows the scaled beta distribution with parameter b_0 or b_1, depending on whether A_{ij}^0 equals zero or one. Consequently,

$$A_{ij}=\begin{cases}-1 & \text{if } -1\le a_{ij}\le c_l,\\ 0 & \text{if } c_l<a_{ij}<c_u,\\ 1 & \text{if } c_u\le a_{ij}\le 1.\end{cases}$$

Update $\{a_{ij}\}$ block by block: for the jth group

$$\begin{aligned}&P(a_{il}\text{ in group } j\,|\,\mathbf{Y},\boldsymbol{\beta},\sigma^2,\mathbf{s},\lambda_1,\lambda_2,G,b_0,b_1)\\ &\propto\frac{1}{m_{s_j}}exp\left(-\frac{1}{2}\boldsymbol{\beta}_j^T\tilde{\mathbf{L}}_j\boldsymbol{\beta}_j\right)\prod_{i,l}f(a_{il};A_{il},A_{il}^0),\end{aligned}$$

With

$$f\left(a_{il};A_{il},A_{il}^0\right)=\begin{cases}(1+a_{il}^2)^{b_0-1}I(-1\le a_{il}\le c_l) & \text{if } A_{ij}^0=0,A_{ij}=-1,\\ (1+a_{il}^2)^{b_0-1}I(c_l<a_{il}<c_u) & \text{if } A_{ij}^0=0,A_{ij}=0,\\ (1+a_{il}^2)^{b_0-1}I(c_u\le a_{il}\le 1) & \text{if } A_{ij}^0=0,A_{ij}=1,\\ (1+a_{il}^2)^{b_1-1}I(-1\le a_{il}\le c_l) & \text{if } A_{ij}^0=1,A_{ij}=-1,\\ (1+a_{il}^2)^{b_1-1}I(c_l<a_{il}<c_u) & \text{if } A_{ij}^0=1,A_{ij}=0,\\ (1+a_{il}^2)^{b_1-1}I(c_u\le a_{il}\le 1) & \text{if } A_{ij}^0=1,A_{ij}=1.\end{cases}$$

In the Metropolis-Hastings algorithm, conditioning on the value of A_{ij}^0, we propose new a_{ij} from the truncated scaled beta distribution, with parameter b_0 or b_1, truncated depending on the value of A_{ij}. Once a_{ij} is accepted, it is truncated to get A_{ij}.

– Sample b_0, given other parameters:

$$\begin{aligned}&P(b_0\,|\,\mathbf{Y},\boldsymbol{\beta},\sigma^2,\mathbf{s},\lambda_1,\lambda_2,G,b_1)\\ &\propto p_{0,-1}(b_0)^{N_{0,-1}}p_{0,0}(b_0)^{N_{0,0}}p_{0,1}(b_0)^{N_{0,1}}I(1<b_0<B_0),\end{aligned}$$

with

$$\begin{aligned}p_{0,-1}(b_0)&=\int_{-1}^{c_l}\frac{\Gamma(2b_0)}{\Gamma(b_0)\Gamma(b_0)}(1-y^2)^{b_0-1}dy,\\ p_{0,0}(b_0)&=\int_{c_l}^{c_u}\frac{\Gamma(2b_0)}{\Gamma(b_0)\Gamma(b_0)}(1-y^2)^{b_0-1}dy,\\ p_{0,1}(b_0)&=\int_{c_u}^{1}\frac{\Gamma(2b_0)}{\Gamma(b_0)\Gamma(b_0)}(1-y^2)^{b_0-1}dy,\\ N_{0,K}&=\sum_{j=1}^{J}\#\{(i,l)\text{ in group } j\colon A_{il}^0=0,A_{il}=K\},\end{aligned}$$

where $K=\{0, -1, 1\}$. We simply pick the proposal distribution to be $Unif(1, B_0)$ in the Metropolis-Hastings algorithm to sample b_0.

– Sample b_1, given other parameters:

$$P(b_1|\mathbf{Y},\boldsymbol{\beta},\sigma^2,\mathbf{s},\lambda_1,\lambda_2,G,b_0)$$
$$\propto p_{1,-1}(b_1)^{N_{1,-1}}\, p_{1,0}(b_1)^{N_{1,0}}\, p_{1,1}(b_1)^{N_{1,1}}\, I(B_1<b_1<1),$$

with

$$p_{1,-1}(b_1)=\int_{-1}^{c_l}\frac{\Gamma(2b_1)}{\Gamma(b_1)\Gamma(b_1)}(1-y^2)^{b_1-1}dy,$$
$$p_{1,0}(b_1)=\int_{c_l}^{c_u}\frac{\Gamma(2b_1)}{\Gamma(b_1)\Gamma(b_1)}(1-y^2)^{b_1-1}dy,$$
$$p_{1,1}(b_1)=\int_{c_u}^{1}\frac{\Gamma(2b_1)}{\Gamma(b_1)\Gamma(b_1)}(1-y^2)^{b_1-1}dy,$$
$$N_{1,K}=\sum_{j=1}^{J}\#\{(i,l)\ in\ group\ j\colon A_{il}^0=1, A_{il}=K\},$$

where $K=\{-1, 0, 1\}$. Choose proposal distribution $Unif(B_1, 1)$ in the Metropolis-Hastings algorithm to sample b_1.

2.4 Variable selection

Following Kang and Guo (2009), with a series of posterior draws after a burn-in period, we first choose the optimal tuning parameters (λ_1, λ_2) that minimize prediction error based on the tuning data. We then draw samples from the conditional posterior, given the fixed optimal $(\lambda_1^*, \lambda_2^*)$, and make inferences, including variable selection.

In the Bayesian framework, variable selection can be dealt with by a Bayesian spike and slap approach (Ishwaran and Rao, 2005; Li and Zhang, 2010) with a suitable prior or treated as a hypothesis-testing problem based on posterior samples. One can simply apply a hard-threshold rule with a pre-specified number δ so that β_j is regarded as zero if its posterior mode is located in $[-\delta, \delta]$ (Yi and Xu, 2008; Kang and Guo, 2009). Li et al. (2002) and Bae and Mallick (2004) explicitly parameterized the variance of each β_j with prior distribution as Λ_j and deleted the predictor if posterior Λ_j fell below a threshold. Alternatively, we can exclude a covariate if its posterior variance has a value below a small number c (Li and Lin, 2010). In this article, we employ three selection approaches:

1. M-cut: Select a coefficient whose absolute posterior mean exceeds $\delta=0.05$,
2. S-cut: Select a coefficient whose posterior standard deviation exceeds $c=0.05$,
3. Z-cut: Select a coefficient whose absolute Z statistics exceeds $Z=1.96$, which is the ratio of the posterior mean and posterior standard deviation

$$Z=\frac{\hat{\beta}}{\hat{\sigma}(\beta)}.$$

3 Results

3.1 Toy example

We consider a hypothetical graph that consists of only four nodes $\{A, B, C, D\}$, representing four genes, with the causal relationship depicted in upper left panel of Figure 2. For a healthy individual, D is suppressed

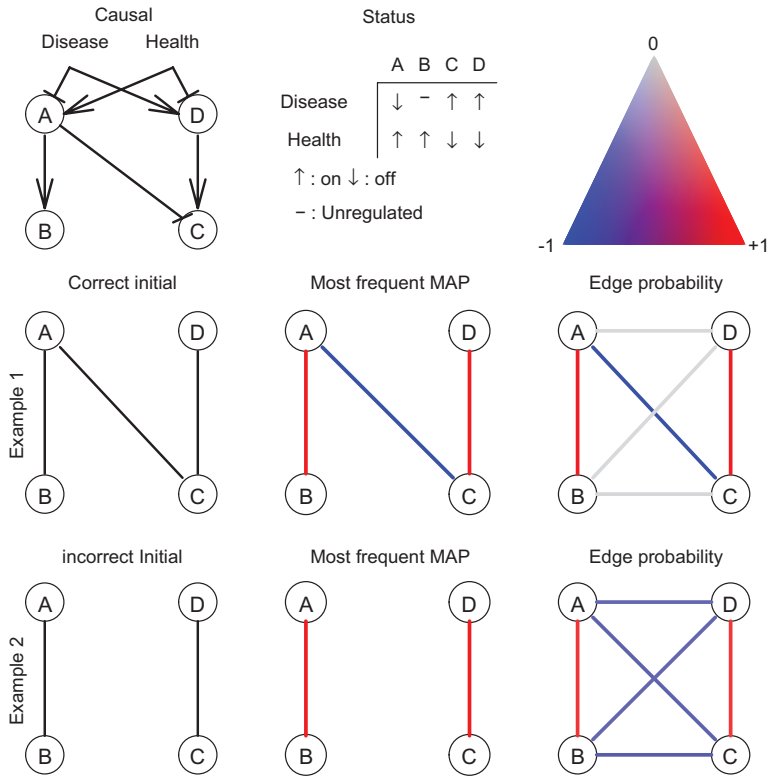

Figure 2 Simulation model and identified MAP graphs under different priors. The simulation model is based on causal relationships between genes {*A, B, C,D*} (upper-left panel). We consider two initial graph structures for setting up the informative priors. The most common MAP graphs and the probability distributions of each edge status identified in the MAP graph using the proposed methods are plotted (see the upper right panel for the color legend).

while A is activated, which in turn activates B and suppresses C; otherwise, A is suppressed and D is activated, which in turn activates C, but the status of B is unregulated. The status of each gene is shown in the upper middle panel of Figure 2. To model this relationship, depending on gene's on/off status, we generate the expression level of each gene u, x_u, according to the following rule:

$$x_u \sim \begin{cases} N(\mu_1,\sigma^2) & \text{if } u \text{ is on,} \\ N(\mu_0,\sigma^2) & \text{if } u \text{ is off,} \\ \tau N(\mu_1,\sigma^2)+(1-\tau)N(\mu_0,\sigma^2) & \text{if } u \text{ is unregulated.} \end{cases}$$

For the following examples, we take μ_1=1, μ_1=-1, σ^2=0.08, and τ=0.1. For an individual with disease, response Y is set to one and otherwise negative one. We simulate datasets that consist of 100 cases and 100 controls. For each simulated trial, we generate a training dataset, a tuning dataset, and a testing dataset, each with equal sample size 200 from the same model. The tuning parameters (λ_1 for lasso, and λ_1, λ_2 for Grace/aGrace) are chosen to minimize the residual sum of squares based on the tuning dataset. For all the methods, the corresponding regression coefficients are used to compute prediction errors based on the testing dataset. The regression coefficients for rGrace are taken as the posterior mean of the coefficients generated after we determine the optimal tuning parameters. Given a training set, rGrace starts from lasso estimates of the regression coefficients and after 5000 burn-in iterations, runs the MCMC procedure for 10,000 iterations to select optimal tuning parameters. Given the selected tuning parameters, draw another 10,000 samples from the conditional posterior. Set hyperparameters (α, θ, α_1, θ_1, α_2, θ_2, B_0, B_1) to (0.1, 10, 2, 0.1, 2, 0.1, 10, 0.1). The sensitivity of the inference to the specification of the hyperparameter is formally investigated in the next section.

In the first example, correct network structure is provided for Grace, aGrace and rGrace, but without signs. With a correct initial network structure, we compute the maximum *a posteriori* probability (MAP) graph structure (the graph structure that occurs most frequently among the posterior draws) for each replicate. The MAP graph for 92 replicates out of 100 equals the assumed causal graph structure, in other words, the positive edges between A and B, and between C and D, and the negative edge between A and C. The second column of Figure 2 shows the most frequent MAP graphs among the 100 replicates. Based on the posterior MAP samples using rGrace, we compute for each edge the probability of being *positive*, *negative*, or *no edge*. The third column of Figure 2 shows these probabilities of each edge with a color reflecting the inferred sign and strength for rGrace. As indicated in the upper right panel in Figure 2, red represents a *positive edge* (corresponding to +1 in the adjacency matrix A), blue represents a *negative edge* (–1 in A), and gray denotes *no edge*. Given the correct structure, rGrace is able to recover the true signs with great certainty.

In the second example, we provide an incorrect network structure with two edges connecting A to B, and C to D. The positive signs for these two edges can be recovered successfully as shown Figure 2, but it is uncertain where the negative edge is between the groups $\{A, B\}$ and $\{C,D\}$. Such uncertainty comes from the lack of prior information. If a correct initial graph structure was given as a prior, with the prior in Section 2, we would prefer the edge AC over the other three edges, BC, AD and BD. In this example, however, both edges are equally penalized. The causal relationships between these four genes only imply similar regression coefficients for A and B, similar coefficients for C and D, and distinct signs for the two groups. In this sense, any negative edge between two nodes, one from either group, is equivalent to another.

To better understand the effects of different priors, we directly calculate, under both the correct and incorrect network priors, the Bayes factors of all 729 (each edge has three possible signs, yielding a total of 3^6=729) graph structures. We run 729 MCMC chains on the same simulated data set, one for each structure, and use the posterior samples to calculate the Bayes factor. We then rank all graph structures by their Bayes factors. We repeat this procedure for 50 independently generated samples and add the ranks for each graph structure across these 50 samples. Figure 3 shows the top five models based on the sum of ranks and any plotted edge takes a *positive* (red) or *negative* (blue) value. When the correct graph structure (without signs) is used as the prior, rGrace is able to recover the assumed causal structure. In example 1, where the correct graph structure is provided, each of the top five structures contains the AC edge. In example 2, all top five structures contain the AB and CD edges, consistent with the prior. However, since there is no prior information about the AC edge, the top five graphs include all four possible ways of connecting $\{A, B\}$ and $\{C, D\}$ by adding one edge.

Based on 100 replicates, Table 1 shows the estimated prediction errors (with standard errors) using lasso, Grace, aGrace, and rGrace. Using the proposed methods, we observe a notable reduction of mean prediction errors under both correct and incorrect prior specifications.

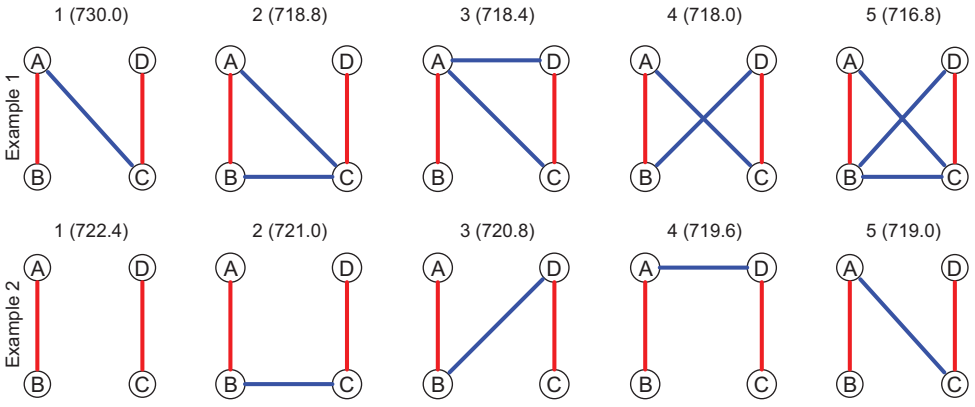

Figure 3 Top five structures based on Bayes factors under two priors. The ranking is based on average rank (shown in parentheses) among 729 models, using 50 simulations.

Table 1 Mean prediction error and standard error (std. err) based on 100 simulated replicates, using lasso, Grace, aGrace, and rGrace.

Graph	Lasso	Grace	aGrace	rGrace
Example 1	1.283	1.025	1.022	**0.992**
Std.err	0.0302	0.0117	0.0117	0.0095
Example 2	–	1.029	1.028	**1.005**
Std.err	–	0.0118	0.0118	0.0103

Each bold value is the smallest number in the corresponding row.

3.2 Simulation studies

In this section, the data are generated based on a linear regression model $\mathbf{Y}=\mathbf{X}^T\boldsymbol{\beta}+\epsilon$. For each replicate, the size of the dataset, n, equals 100 for a training, a tuning and a test set. We assume predictors X form 10 groups ($\mathbf{X}_1$, ..., $\mathbf{X}_{10}$), each consisting of 21 variables; hence p=210. Predictors within each group are marginally standard normal, with compound symmetry correlation ρ=0.2. The significant variables are chosen to be the first two groups. The true coefficient vector $\boldsymbol{\beta}$ is given by ($\boldsymbol{\beta}_1$, $\boldsymbol{\beta}_2$, $\mathbf{0}$, ..., $\mathbf{0}$). Vector $\boldsymbol{\beta}_1$ is of length 21 with all elements equal to three, and all elements in $\boldsymbol{\beta}_2$ equal to –2. The correct graph structure consist of 10 separate connected components, each a fully connected subgraph with 21 nodes and 210 edges. The initial graph structure has the same grouping as the true graph, but with a ring-shaped network with 21 edges in each group. The independent and identically distributed error term ϵ follows a normal distribution with zero mean and variance $\boldsymbol{\beta}^T\boldsymbol{\beta}/4$. As mentioned in the previous section, after 5000 burn-in iterations, rGrace runs for 10,000 iterations to select optimal tuning parameters, and then runs another 10,000 iterations to make inferences. The Grace/aGrace solution is computed via the coordinate-descent algorithm provided by Li and Li (2010). We use the *SGL()* function in the *SGL* R package (Simon et al., 2012) to obtain the group-Grace/aGrace solution with artificial $\mathbf{Y}^*$ and $\mathbf{X}^*$, defined as

$$\mathbf{Y}^*=\begin{pmatrix}\mathbf{Y}\\0\end{pmatrix},$$

$$\mathbf{X}^*=\begin{pmatrix}\mathbf{X}\\\sqrt{\lambda_2}(\mathbf{S}\tilde{\mathbf{L}})^{1/2}\end{pmatrix}.$$

To examine how the choice of hyperparameters affects the inference results, we consider the three combinations of hyperparameters (α, θ, α_1, θ_1, α_2, θ_2, B_0, B_1) shown in Table 2.

In setting the hyperparameters (B_0, B_1) for the distribution of the scale parameters (b_0, b_1), a smaller B_1 (larger B_0) leads to a curvier scaled beta distribution, concentrating on the initial graph. On the other hand, a larger B_1 (smaller B_0) induces a flatter scaled beta distribution. To evaluate the performance of each method, we calculate the number of true positives (TP), true negatives (NP), false positives (FP), and false negatives (FN) and report the average false positive rate (FPR) and average false negative rate (FNR), defined as

Table 2 Three combination of hyperparameters.

	*C*1	*C*2	*C*3
α	0.1	0.1	2
θ	10	10	50
α_1	2	2	5
θ_1	0.1	0.1	0.05
α_2	2	2	5
θ_2	0.1	0.1	0.05
B_0	10	3	3
B_1	0.1	0.3	0.3

$$TPR=\frac{TP}{TP+FN},$$
$$TNR=\frac{TN}{TN+FP},$$

respectively. Moreover, the proportion of selected edges relative to the number of true edges (%Edge) together with the proportion of selected edges with the correct sign (%Sign-Edge) for each MAP graph will be reported. For a particular dataset, the sample trace plots of two significant regression coefficients, an instrumental variable and a tuning parameter for the last 5000 iterations (before the optimal tuning parameters are chosen) are shown in Figure 4. Table 3 gives the means of the four selected nonzero coefficients ($\beta_1, \beta_{21}, \beta_{22}, \beta_{42}$), TPR, TNR, %Edge, and %Sign-Edge, where the hyperparameter combination ($C1$) is used. Compared to competing methods, rGrace exhibits promising performance in terms of smaller prediction errors, more accurate parameter estimates and larger *TNR* values. Although the initial graph structure contains only 10% of the actual edges, rGrace is able to recover about one-fourth of the total edges. With a less informative prior, say, the hyperparameter combination $C2$, rGrace is capable of discovering around half of the true edges, as shown in Table 4. As suggested by Table 4, a different choice of ($\alpha, \theta, \alpha_1, \theta_1, \alpha_2, \theta_2$) does not have a significant impact on the inference results.

3.3 Application to a gene expression study of brain aging

Li and Li (2010) analyzed gene expression data measured in the human brain (Lu et al., 2004), with the logarithm of the individual age as the response and $\log_{10}$ of the expression levels as covariates. Using the same network structure as for Li and Li (2010) and applying the default algorithm of Yip and Horvath (2007), we identified 174 separate gene modules with a total of 1237 genes and 3478 intra-modular edges. The largest module contains 76 genes. To estimate the regression coefficients, the tuning parameters are chosen based on a five-fold cross validation (CV) applied to the entire dataset for lasso, Grace/aGrace, and group-Grace/aGrace. For rGrace, instead of choosing the tuning parameters that minimize CV-error, we compute the average of the optimal tuning parameters in each fold. With selected tuning parameters, rGrace runs MCMC for 100,000 iterations to sample from the posterior distribution (total computing time 270 hours on an Inteli5-

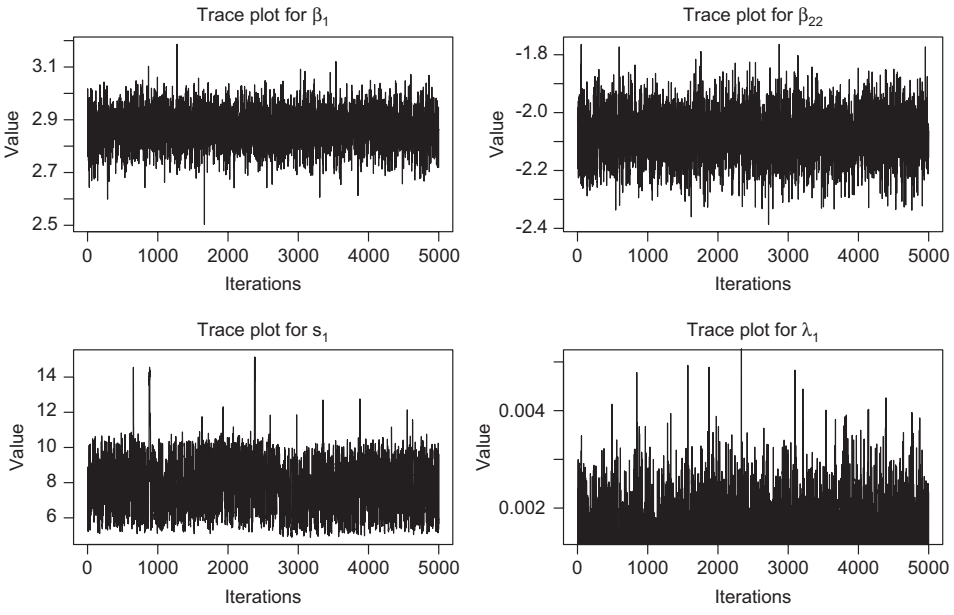

Figure 4 Sample trace plots of two significant regression coefficients, an instrumental variable, and a tuning parameter.

Table 3 Prediction errors (PE), the means of four nonzero coefficients ($\beta_1, \beta_{21}, \beta_{22}, \beta_{42}$), the true positive rate (TPR), the true negative rate (TNR), the proportion of selected true edges (%Edge), and the proportion of selected true edges of correct sign (%Sign-Edge), based on 100 replicates.

	Lasso	Grace	aGrace	Group-Grace	Group-aGrace	bGrace			rGrace		
						M-cut	S-cut	Z-cut	M-cut	S-cut	Z-cut
PE	50.3	11.3	10.3	8.2	8.6	7.2	–	–	4.4	–	–
Std.err	(17.6)	(3.3)	(3.0)	(1.8)	(2.0)	(1.6)	–	–	(1.3)	–	–
β_1=3	2.33	2.76	2.80	2.90	2.83	2.83	–	–	2.87	–	–
Std.err	(0.95)	(0.52)	(0.55)	(0.20)	(0.36)	(0.19)	–	–	(0.18)	–	–
β_{21}=3	2.29	2.81	2.82	2.91	2.86	2.82	–	–	2.85	–	–
Std.err	(0.83)	(0.52)	(0.60)	(0.20)	(0.26)	(0.20)	–	–	(0.17)	–	–
β_{22}=−2	−1.28	−1.67	−1.73	−1.81	−1.77	−1.84	–	–	−1.85	–	–
Std.err	(0.76)	(0.46)	(0.48)	(0.25)	(0.32)	(0.17)	–	–	(0.18)	–	–
β_{42}=−2	−1.28	−1.72	−1.71	−1.85	−1.76	−1.82	–	–	−1.84	–	–
Std.err	(0.59)	(0.44)	(0.52)	(0.29)	(0.35)	(0.18)	–	–	(0.15)	–	–
TPR	98.2%	99.1%	99.0%	100%	100%	99.2%	99.3%	99.4%	100%	99.4%	99.0%
Std.err	(0.06)	(0.01)	(0.01)	(0.00)	(0.00)	(0.01)	(0.01)	(0.04)	(0.00)	(0.05)	(0.01)
TNR	45.6%	52.1%	52.3%	54.6%	52.7%	55.6%	46.8%	84.8%	63.7%	57.2%	88.9%
Std.err	(0.11)	(0.06)	(0.07)	(0.06)	(0.08)	(0.07)	(0.12)	(0.16)	(0.09)	(0.11)	(0.17)
%Edge	10%	10%	10%	10%	10%	10%	–	–	26.4%	–	–
Std.err	–	–	–	–	–	–	–	–	(2.23%)	–	–
%Sign-Edge	10%	10%	10%	10%	10%	10%	–	–	26.3%	–	–
Std.err	–	–	–	–	–	–	–	–	(2.41%)	–	–

Table 4 Prediction errors (PE), the means of four nonzero coefficients ($\beta_1, \beta_{21}, \beta_{22}, \beta_{42}$), the true positive rate (TPR), the true negative rate (TNR), the proportion of selected true edges (%Edge), and the proportion of selected true edges of correct sign (%Sign-Edge), for bGrace and rGrace (use the Z-cut method for variable selection) with three sets of hyperparameters based on 100 replicates.

	bGrace			rGrace		
	*C*1	*C*2	*C*3	*C*1	*C*2	*C*3
PE	7.2	–	7.1	4.4	3.8	4.0
Std.err	(1.6)	–	(1.6)	(1.3)	(1.2)	(1.3)
β_1=3	2.83	–	2.81	2.87	2.87	2.82
Std.err	(0.19)	–	(0.18)	(0.18)	(0.16)	(0.19)
β_{21}=3	2.82	–	2.80	2.85	2.83	2.83
Std.err	(0.20)	–	(0.19)	(0.17)	(0.17)	(0.18)
β_{22}=−2	−1.84	–	−1.82	−1.85	−1.82	−1.86
Std.err	(0.17)	–	(0.16)	(0.18)	(0.17)	(0.20)
β_{41}=−2	−1.82	–	−1.82	−1.84	−1.87	−1.82
Std.err	(0.18)	–	(0.17)	(0.15)	(0.15)	(0.17)
TPR	99.4%	–	98.9%	99.0%	97.9%	98.9%
Std.err	(0.04)	–	(0.04)	(0.01)	(0.09)	(0.03)
TNR	84.8%	–	84.3%	88.9%	95.2%	93.3%
Std.err	(0.16)	–	(0.14)	(0.17)	(0.06)	(0.06)
%Edge	10%	–	–	26.4%	55.0%	55.4%
Std.err	–	–	–	(2.23%)	(4.45%)	(6.19%)
%Sign-Edge	10%	–	–	26.3%	54.8%	55.5%
Std.err	–	–	–	(2.41%)	(4.46%)	(6.30%)

2320, 3 GHz processor, 6 GB RAM). To estimate the prediction errors, we apply a nested CV procedure, with an outer three-fold CV loop and an inner five-fold CV loop (Varma and Simon, 2006). Table 5 shows the prediction errors, and the number of genes and edges based on the regression coefficients for various methods.

Table 5 Prediction errors based on brain aging gene expression data, using lasso, Grace, aGrace, group-Grace, group-aGrace and rGrace.

	Lasso	Grace	aGrace	Group-Grace	Group-aGrace	rGrace
PE	0.099	0.081	0.080	0.107	0.107	**0.067**
#Genes	19	61	84	99	105	58
#Edges	0	4	22	58	61	38

Each bold value is the smallest value in the row.

Our proposed methods achieve better prediction performance than Grace/aGrace without using information about the potential signs of the regression coefficients. Table 6 displays the nonzero edges among significant genes selected by rGrace using Z-cut. The identified genes CAV1 and CAV2 are associated with progressive optic nerve degeneration (Wiggs et al., 2011). Gene CD247 is reported to be significantly enriched in neurological disease (de Jong et al., 2012) and gene CDK5 is related to adult-onset neuro-degeneration, as

Table 6 Edges among significant genes obtained by rGrace based on brain aging gene expression data.

Gene pair		Sign	Sign in initial graph
NCR2	TYRPOBP	1	1
CDC25B	YWHAB	1	1
CDC25B	YWHAE	1	1
PLAT	PLG	1	1
MPZ	MPZL1	1	1
NLGN1	NRXN1	1	1
DVL1	FRAT2	−1	1
DVL3	FRAT2	−1	1
F12	PLG	−1	1
PLG	SERPINF2	−1	1
MLLT4	SSX2IP	−1	1
CAV1	CD247	1	0
CAV1	NCR3	1	0
CAV1	SHC1	1	0
CAV2	CD247	1	0
CAV2	CDK5	1	0
CAV2	LCK	1	0
CAV2	SHC1	1	0
CD247	SHC1	1	0
CDK5	SHC1	1	0
LCK	NCR3	1	0
NCR2	YES1	1	0
DVL1	DVL3	1	0
F12	SERPINF2	1	0
CAMK2A	PPP3CB	−1	0
CAV1	NCR2	−1	0
CAV1	YES1	−1	0
CAV2	NCR2	−1	0
CAV2	YES1	−1	0
CD247	NCR3	−1	0
CD247	TYROBP	−1	0
CDK5	NCR2	−1	0
CDK5	TYROBP	−1	0
LCK	NCR2	−1	0
NCR2	NCR3	−1	0
NCR2	SHC1	−1	0
TUBB2C	TUBB4	−1	0
PLAT	SERPINF2	−1	0

the lack of CDK5 within the nervous system leads to abnormalities in neuron development (Trunova and Giniger, 2012). Among discovered edges, it is interesting to note that physical interaction was confirmed between gene pairs (CD247, NCR3), (CD247, SHC1), (CAVA1, YES1), and (DVL1, DVL3) based on iRefIndex (Razick et al., 2008). Genes PLAT and SERPINF2 share protein domains based on InterPro (Hunter et al., 2009). Also, gene pairs (F12, SERPINF2) are colocalized (Schadt et al., 2004). This biological evidence supports the validity of rGrace.

4 Discussion

We have proposed a Bayesian hierarchical model, rGrace, that incorporates the network (graph) structure of covariates and produces posterior inference of regression coefficients and a graph structure. Compared to Grace/aGrace, rGrace can discover different gene-gene relations by allowing random graph structure. A simulation study and real data analysis demonstrated that the estimated coefficients have lower prediction error. The MCMC procedure also facilitates the estimation of the posterior probability of the graph structure.

Our prior for the graph structure encourages similar structures as the initial graph. To further induce sparseness or fewer groups, the prior can, for instance, take the form

$$P(G)\propto\exp(-\lambda_G\#\{edges\}),$$

or

$$P(G)\propto\exp(-\lambda_G\#\{groups\}).$$

In general, as suggested by Mukherjee and Speed (2008), one can take a log-linear network prior,

$$P(G)\propto\exp\left(-\lambda_G\sum_i w_i f_i(G)\right),$$

where each $f_i(G)$ maps certain feature of the graph to a real value that increases if the graph deviates more from prior belief, with weight w_i. Such a feature can also include edges within each group, degree distribution, the number of two-stars, triangles, and so forth. This general class of informative network priors is also consistent with exponential random graph models (Robins et al., 2006). However, more equivalent graphs may arise with such specification.

In this paper, we use the MCMC procedure to sample graphs. Alternatively, for a moderate number of possible models (graphs), the Metropolized Carlin and Chib (1995) method can be adopted by setting up pseudopriors. If the number of graph structures of interest is small, one can even run a Gibbs sampler for each fixed graph structure and directly compare the Bayes factor, which is possible since the prior is proper. Other Bayesian model selection methods can be found in the survey of Han and Carlin (2001) and Dellaportas et al. (2002).

The linear model in this paper can be extended to generalized linear models. Holmes and Held (2006) discussed Bayesian logistic regression and multinomial regression based on auxiliary variable methods. Yang and Song (2010) studied a Bayesian probit regression model for disease classification, utilizing a latent variable representation. Intercept and regression coefficients are integrated out to avoid convergence problems in the MCMC algorithm. With suitable implementation, the rGrace procedure can be extended to generalized linear models.

Appendix A Derivation of Sampling Scheme for $\beta|\sigma^2$

According to Andrews and Mallows (1947), for $a>0$, a scale mixture of normal distributions representation of the Laplace distribution is

$$\frac{a}{2}\exp(-a|Z|)=\int_0^\infty \frac{1}{\sqrt{2\pi s}}\exp\left(-\frac{Z^2}{2s}\right)\frac{a^2}{2}\exp\left(-\frac{a^2 s}{2}\right)ds.$$

Let $a=\frac{\lambda_1\sqrt{p_j}}{2\sigma^2}$ and $Z=\|\beta_j\|_2$. Then

$$\frac{\lambda_1\sqrt{p_j}}{4\sigma^2}\exp\left(-\frac{\lambda_1\sqrt{p_j}\|\beta_j\|_2}{2\sigma^2}\right)=\int_0^\infty \frac{1}{\sqrt{2\pi s}}\exp\left(-\frac{\|\beta_j\|_2^2}{2s}\right)\frac{\lambda_1^2 p_j}{8\sigma^4}\exp\left(-\frac{\lambda_1^2 p_j}{8\sigma^4}s\right)ds$$

or

$$\exp\left(-\frac{\lambda_1\sqrt{p_j}\|\beta_j\|_2}{2\sigma^2}\right)=\frac{\lambda_1\sqrt{p_j}}{2\sigma^2\sqrt{2\pi}}\int_0^\infty \frac{1}{\sqrt{s}}\exp\left(-\frac{\|\beta_j\|_2^2}{2s}\right)\exp\left(-\frac{\lambda_1^2 p_j}{8\sigma^4}s\right)ds.$$

Therefore,

$$\begin{aligned}&\exp\left(-\frac{\lambda_1}{2\sigma^2}\sum_{j=1}^J\sqrt{p_j}\|\beta_j\|_2\right)\\&=\prod_{j=1}^J\left(\frac{\lambda_1\sqrt{p_j}}{2\sigma^2\sqrt{2\pi}}\right)\int_0^\infty\cdots\int_0^\infty\frac{1}{\sqrt{s_1\cdots s_J}}\exp\left(-\frac{1}{2}\sum_{j=1}^J\frac{\|\beta_j\|_2^2}{s_j}\right)\exp\left(-\frac{\lambda_1^2}{8\sigma^4}\sum_{j=1}^J p_j s_j\right)ds_1\cdots ds_J.\end{aligned}$$

Hence

$$\begin{aligned}&\exp\left(-\frac{\lambda_1}{2\sigma^2}\sum_{j=1}^J\sqrt{p_j}\|\beta_j\|_2-\frac{\lambda_2}{2\sigma^2}\beta^T\tilde{L}\beta\right)\\&=\prod_{j=1}^J\left(\frac{\lambda_1\sqrt{p_j}}{2\sigma^2\sqrt{2\pi}}\right)\int_0^\infty\cdots\int_0^\infty\frac{1}{\sqrt{s_1\cdots s_J}}\exp\left(-\frac{1}{2}\sum_{j=1}^J\frac{\|\beta_j\|_2^2}{s_j}-\frac{\lambda_2}{2\sigma^2}\beta^T\tilde{L}\beta\right)\\&\exp\left(-\frac{\lambda_1^2}{8\sigma^4}\sum_{j=1}^J p_j s_j\right)ds\\&\propto_{\sigma^2}\int_0^\infty\cdots\int_0^\infty\frac{1}{\sqrt{s_1\cdots s_J}}\exp\left(-\frac{1}{2}\beta^T\left(D_s+\frac{\lambda_2}{\sigma^2}\tilde{L}\right)\beta\right)\\&\exp\left(-\frac{\lambda_1^2}{8\sigma^4}\sum_{j=1}^J p_j s_j\right)ds_{1\cdots}d_{sJ},\end{aligned}\tag{2}$$

where D_s is a block diagonal matrix with J blocks in the diagonal, and the j th block D_{s_j} is $\frac{1}{s_j}\mathbf{I}_{p_j\times p_j}$. Note that, assuming a block structure for $\tilde{L}$,

$$\begin{aligned}&det\left(D_s+\frac{\lambda_2}{\sigma^2}\tilde{L}\right)\\&=\prod_{j=1}^J det\left(D_{s_j}+\frac{\lambda_2}{\sigma^2}\tilde{L}_j\right)\\&=\prod_{j=1}^J det\left(D_{s_j}+\frac{\lambda_2}{\sigma^2}O_j^T\wedge_j O_j\right)\\&=\prod_{j=1}^J det\left(O_j^T\left(D_{s_j}+\frac{\lambda_2}{\sigma^2}\wedge_j\right)O_j\right)\\&=\prod_{j=1}^J\left(\prod_{l=1}^{p_j}\left(\frac{1}{s_j}+\frac{\lambda_2 w_{j_l}}{\sigma^2}\right)\right),\end{aligned}$$

where each O_j is an orthogonal matrix and $\wedge_j$ is a diagonal matrix, that is $\wedge_j = diag(w_{j_1},...,w_{j_{p_j}})$. With the block diagonal structure assumption of $\tilde{L}$, (2) can be written as:

$$\propto_{\sigma_2} \int...\int \frac{\prod_{j=1}^{J}\left(\prod_{l=1}^{p_j}\sqrt{\frac{1}{s_j}+\frac{\lambda_2 w_{j_l}}{\sigma^2}}\right)}{\sqrt{s_1...s_J}\prod_{j=1}^{J}\left(\prod_{l=1}^{p_j}\sqrt{\frac{1}{s_j}+\frac{\lambda_2 w_{j_l}}{\sigma^2}}\right)} \exp\left(-\frac{1}{2}\beta^T\left(D_s+\frac{\lambda_2}{\sigma^2}\tilde{L}\right)\beta\right)$$
$$\exp\left(-\frac{\lambda_1^2}{8\sigma^4}\sum_{j=1}^{J}p_j s_j\right) ds.$$

As a result, we can treat $\beta|\sigma^2$ alternatively as:

$$\beta|\mathbf{s}=(s_1,...,s_J),\sigma^2 \sim N\left(0,\left(D_s+\frac{\lambda_2}{\sigma^2}\tilde{L}\right)^{-1}\right)$$

$$p(\mathbf{s}|\sigma^2)\propto\prod_{j=1}^{J}\left(\frac{I(s_j>0)}{\sqrt{s_j}\prod_{l=1}^{p_j}\left(\sqrt{\frac{1}{s_j}+\frac{\lambda_2 w_{j_l}}{\sigma^2}}\right)}\exp\left(-\frac{\lambda_1^2 p_j}{8\sigma^4}s_j\right)\right).$$

Acknowledgment: This research is, in parts, supported by NSF grants DMS-0714669 and SES-1023176, NIH grant R01 GM070789, and a 2010 Google research award. We would like to thank two anonymous reviewers for their constructive comments.

References

Andrews, D. F. and C. L. Mallows (1947): "Scale mixtures of normal distributions," J. Roy. Stat. Soc. B, 36, 99–102.

Bae, K. and B. K. Mallick (2004): "Gene selection using a two-level hierarchical Bayesian model," Bioinformatics, 20, 3423–3430.

Bar-Joseph, Z., G. K. Gerber, T. I. Lee, N. J. Rinaldi, J. Y. Yoo, F. Robert, D. B. Gordon, E. Fraenkel, T. S. Jaakkola, R. A. Young and D. K. Gifford (2003): "Computational discovery of gene modules and regulatory networks," Nat. Biotechnol., 21, 1337–1342.

Baranzini, S. E., N. W. Galwey, J. Wang, P. Khankhanian, R. Lindberg, D. Pelletier, W. Wu, B. M. Uitdehaag, L. Kappos, C. H. Polman and GeneMSA Consortium (2009): "Pathway and network-based analysis of genome-wide association studies in multiple sclerosis," Hum. Mol. Genet., 18, 2078–2090.

Carlin, B. P. and S. Chib (1995): "Bayesian model choice via Markov chain Monte Carlo methods," J. Roy. Stat. Soc. B, 57, 473–484.

Chung, F. (1997): *Spectral graph theory*, Vol. 92 of CBMS Regional Conferences Series. American Mathematical Society, Providence.

De Jong, S., M. Boks, T. Fuller, E. Strengman, E. Janson, C. De Kovel, A. Ori, N. Vi, F. Mulder, J. Blom, B. Glenthoj, C. Schbart, W. Cahn, R. Kahn, S. Horvath and R. Ophoff (2012): "A gene co-expression network in whole blood of schizophrenia patients is independent of antipsychotic-use and enriched for brain-expressed genes," PLoS One, 7, e39498.

Dellaportas, P., J. J. Forster and I. Ntzoufras (2002): "On Bayesian model and variable selection using MCMC," Stat. Comput., 12, 27–36.

Elbers, C. C., K. R. van Eijk, L. Franke, F. Mulder, Y. T. van der Schouw, C. Wijmenga, and N. C. Onland-Moret (2009): "Using genome-wide pathway analysis to unravel the etiology of complex diseases," Genet. Epidemiol., 33, 419–431.

Emily, M., T. Mailund1, J. Hein, L. Schauser and M. H. Schierup (2009): "Using biological networks to search for interacting loci in genome-wide association studies," Eur. J. Hum. Genet., 17, 1231–1240.

Franke, L., H. van Bakel, L. Fokkens, E. D. de Jong, M. Egmont-Peterson and C. Wijmenga (2006): "Reconstructing of a functional human gene network with an application for prioritizing positional candidate genes," Am. J. Hum. Genet., 78, 1011–1025.

Friedman, J., T. Hastie and R. Tibshirani (2010): "A note on the group lasso and a sparse group lasso," Arxiv, arXiv:1001.0736.

Griffin, J. E. and P. J. Brown (2007): "Bayesian adaptive lasso with non-convex penalization," Aust. NZ. J. Stat., 53, 423–442.

Han, C. and B. P. Carlin (2001): "Markov chain Monte Carlo methods for computing Bayes factors: a comparative review," J. Am. Stat. Assoc., 96, 1122–1132.

Holmes, C. C and L. Held (2006): "Bayesian auxiliary variable models for binary and multinomial regression," Bayesian Analysis, 1, 145–168.

Hunter, S., R. Apweiler, T. K. Attwood, A. Bairoch, A. Bateman, D. Binns, P. Bork, U. Das, L. Daugherty, L. Duquenne, R. D. Finn, J. Gough, D. Haft, N. Hulo, D. Kahn, E. Kelly, A. Laugraud, I. Letunic, D. Lonsdale, R. Lopez, M. Madera, J. Maslen, C. McAnulla, J. McDowall, J. Mistry, A. Mitchell, N. Mulder, D. Natale, C. Orengo, A. F. Quinn, J. D. Selengut, C. J. Sigrist, M. Thimma, P. D. Thomas, F. Valentin, D. Wilson, C. H. Wu and C. Yeats (2009): InterPro: the integrative protein signature database," Nucl. Acids Res., 37, D211–D215.

Ishwaran, H. and J. S. Rao (2005): "Spike and slab variable selection: frequentist and Bayesian strategies," Ann. Stat., 33, 730–773.

Kang, J. and J. Guo (2009): Self-adaptive lasso and its Bayesian estimation. Technical report, University of Michigan. Available at http://www.stat.lsa.umich.edu/~guojian/publications/manuscript_bayesso_arxiv.pdf.

Kim, M., H. Shin, T. S. Chung, J.-G. Joung and J. H. Kim (2011): "Extracting regulatory modules from gene expression data by sequential pattern mining," BMC Genomics, 12(Suppl 3), S5.

Kyung, M., J. Gill, M. Ghosh and G. Casella (2010): Penalized regression, standard errors, and Bayesian lassos," Bayesian Analysis, 5, 369–412.

Langfelder, P. and S. Horvath (2008): "WGCNA: an R package for weighted correlation network analysis," BMC Bioinformatics, 9, 559.

Li, C. and H. Li (2008): "Network-constrained regularization and variable selection for analysis of genomic data," Bioinformatics, 24, 1175–118.

Li, C. and H. Li (2010): "Variable selection and regression analysis for graph-structured covariates with an application to genomics," Ann. Appl. Stat., 4, 1498–1516.

Li, Q. and N. Lin (2010): "The Bayesian elastic net," Bayesian Analysis, 5, 151–170.

Li, F. and N. R. Zhang (2010): "Bayesian variable selection in structured high-dimensional covariate spaces with applications in genomics," J. Am. Stat. Assoc., 105, 1202–1214.

Li, Y., C. Campbell and M. Tipping (2002): "Bayesian automatic relevance determination algorithms for classifying gene expression data," Bioinformatics, 18, 1332–1339.

Liu, F. and A. C. Lozano (2011): "A Graph Laplacian prior for variable selection and grouping," Biometrika, 98, 1–31.

Liu, J., J. Huang and S. Ma (2013): "Incorporating network structure in integrative analysis of cancer prognosis data," Genet. Epidemiol., 37, 173–83.

Lu, T., Y. Pan, S. Kao, C. Li, I. Kohane, J. Chan and B. A. Yankner (2004): "Gene regulation and DNA damage in the ageing human brain," Nature, 429, 883–891.

Mason, M. J., G. Fan, K. Plath, Q. Zhou and S. Horvath (2009): "Signed weighted gene co-expression network analysis of transcriptional regulation in murine embryonic stem cells," BMC Genomics, 10, 327.

Meier, L., S. van de Geer and P. Buhlmann (2008): "The group lasso for logistic regression," J. Roy. Stat. Soc. B, 70, 53–71.

Mukherjee, S. and T. P. Speed (2008): "Network inference using informative priors," Proc. Natl. Acad. Sci., 105, 14313–14318.

Newman, M. E. (2006): "Modularity and community structure in networks," Proc. Natl. Acad. Sci., 103, 8577–8582.

Pan, W., B. Xie and X. Shen (2010): "Incorporating predictor network in penalized regression with application to microarray data," Biometrics, 66, 474–484.

Park, T. and G. Casella (2008): "The Bayesian lasso," J. Am. Stat. Assoc., 103, 681–686.

Raman, S., T. J. Fuchs, P. J. Wild, E. Dahl and V. Roth (2009): The Bayesian group-lasso for analyzing contingency tables. Proceedings of the 26th International Conference on Machine Learning, 881–888.

Razick, S., G. Magklaras and I. M. Donaldson (2008): "iRefIndex: a consolidated protein interaction database with provenance," BMC Bioinformatics, 9, 405.

Ravasz, E., A. L. Somera, D. A. Mongru, Z. N. Oltvai and A. L. Barabasi (2002): "Hierarchical organization of modularity in metabolic networks," Science, 297, 1551–1555.

Robins, G., P. Pattison, Y. Kalish and D. Lusher (2006): "An introduction to exponential random graph models for social networks," Social Networks, 29, 173–191.

Ruan, J. and W. Zhang (2008): "Identifying network communities with a high resolution," Phys. Rev. E, 77, 016104.

Rzhetsky, A., T. Zheng and C. Weinreb (2006): "Self-correcting maps of molecular pathways," PLoS One, 1(1), e61

Schadt, E., S. W. Edwards, D. GuhaThakurta, D. Holder, L. Ying, V. Svetnik, A. Leonardson, K. W. Hart, A. Russell, G. Li, G. Cavet, J. Castle, P. McDonagh, Z. Kan, R. Chen, A. Kasarskis, M. Margarint, R. M. Caceres, J. M. Johnson, C. D. Armour, P. W. Garrett-Engele, N. F. Tsinoremas, and D. D. Shoemaker (2004): "A comprehensive transcript index of the human genome generated using microarrays and computational approaches," Gen. Biol., 5, R73.

Segal, E., M. Shapira, A. Regev, D. Pe'er, D. Botstein, D. Koller and N. Friedman (2003): "Module networks: identifying regulatory modules and their condition specific regulators from gene expression data," Nat. Genet., 34, 166–176.

Simon, N., J. Friedman, T. Hastie and R. Tibshirani (2012): "The sparse group lasso," J. Comput. Graph. Stat., DOI:10.1080/10618600.2012.681250.

Stingo, F. C., Y. A. Chen, M. G. Tadesse and M. Vannucci (2011): "Incorporating biological information into linear models: A Bayesian approach to the selection of pathways and genes," Ann. Appl. Stat., 5, 1978–002.

Sun, W., J. G. Ibrahim, and F. Zou (2009): Variable selection by Bayesian adaptive lasso and iterative adaptive lasso, with application for genome-wide multiple loci mapping. Technical report, University of North Carolina at Chapel Hill, Department of Biostatistics. Available at http://biostats.bepress.com/uncbiostat/art10/.

Trunova, S. and E. Giniger (2012): "Absence of the cdk5 activator p35 causes adult-onset neurodegeneration in the central brain of drosophila," Dis. Mod Mech., 5, 210–219.

Varma, S. and R. Simon (2006): "Bias in error estimation when using cross-validation for model selection," BMC Bioinformatics, 7, 91.

Watkinson, J., X. Wang, T. Zheng and D. Anastassiou (2008): "Identification of gene interactions associated with disease from gene expression data using synergy networks," BMC Syst. Biol., 2, 10.

Werhli, A. V. and D. Husmeier (2007): "Reconstructing gene regulatory networks with Bayesian networks by combining expression data with multiple sources of prior knowledge," Stat. Appl. Genet. Mol. Biol., 6, 15.

Wiggs, J. L., J. H. Kang, B. L. Yaspan, D. B. Mirel, C. Laurie, A. Crenshaw, W. Brodeur, S. Gogarten, L. M. Olson, W. Abdrabou, E. DelBono, S. Loomis, J. L. Haines, L. R. Pasquale, and GENEVA Consortium (2011): "Common variants near CAV1 and CAV2 are associated with primary open-angle glaucoma in Caucasians from the USA," Hum. Mol. Genet., 20, 4707–4713.

Yang, A. and X. Song (2010): "Bayesian variable selection for disease classification using gene expression data," Bioinformatics, 26, 215–222.

Yi, N. and S. Xu (2008): "Bayesian lasso for quantitative trait loci mapping," Genetics, 179, 1045–1055.

Yip, A. and S. Horvath (2007): "The generalized topological overlap matrix for detecting modules in gene network," BMC Bioinformatics, 8, 22.

Yuan, M. and Y. Lin (2006): "Model selection and estimation in regression with grouped variables," J. Roy. Stat. Soc. B, 68, 49–67.

Zhang, B. and S. Horvath (2005): "A general framework for weighted gene co-expression network analysis," Stat. Appl. Genet. Mol. Biol., 4, 17.

Zou, H. and T. Hastie (2005): Regularization and variable selection via the elastic net," J. Roy. Stat. Soc. B, 67, 301–320.