Journal of Machine Learning Research 21 (2020) 1-61 Submitted 2/19; Revised 2/20; Published 6/20

Bayesian Model Selection with Graph Structured Sparsity

Youngseok Kim YOUNGSEOK@UCHICAGO.EDU
Chao Gao CHAOGAO@GALTON.UCHICAGO.EDU
Department of Statistics
University of Chicago
Chicago, IL 60637, USA

Editor: Francois Caron

Abstract

We propose a general algorithmic framework for Bayesian model selection. A spike-and-slab Laplacian prior is introduced to model the underlying structural assumption. Using the notion of effective resistance, we derive an EM-type algorithm with closed-form iterations to efficiently explore possible candidates for Bayesian model selection. The deterministic nature of the proposed algorithm makes it more scalable to large-scale and high-dimensional data sets compared with existing stochastic search algorithms. When applied to sparse linear regression, our framework recovers the EMVS algorithm (Ročková and George, 2014) as a special case. We also discuss extensions of our framework using tools from graph algebra to incorporate complex Bayesian models such as biclustering and submatrix localization. Extensive simulation studies and real data applications are conducted to demonstrate the superior performance of our methods over its frequentist competitors such as ℓ_0 or ℓ_1 penalization.

Keywords: spike-and-slab prior, graph laplacian, variational inference, expectation maximization, sparse linear regression, biclustering

1. Introduction

Bayesian model selection has been an important area of research for several decades. While the general goal is to estimate the most plausible sub-model from the posterior distribution (Barry and Hartigan, 1993; Diebolt and Robert, 1994; Richardson and Green, 1997; Bottolo and Richardson, 2010) for a wide class of learning tasks, most of the developments of Bayesian model selection have been focused on variable selection in the setting of sparse linear regression (Hans et al., 2007; Li and Zhang, 2010; Ghosh and Clyde, 2011; Ročková and George, 2014; Wang et al., 2018). One of the main challenges of Bayesian model selection is its computational efficiency. Recently, Ročková and George (2014) discovered that Bayesian variable selection in sparse linear regression can be solved by an EM algorithm Dempster et al. (1977); Neal and Hinton (1998) with a closed-form update at each iteration. Compared with previous stochastic search type of algorithms such as Gibbs sampling (George and McCulloch, 1993, 1997), this deterministic alternative greatly speeds up computation for large-scale and high-dimensional data sets.

The main thrust of this paper is to develop of a general framework of Bayesian models that includes sparse linear regression, change-point detection, clustering and many other models as special cases. We will derive a general EM-type algorithm that efficiently explores possible candidates for Bayesian model selection. When applied to sparse linear regression, our model and algorithmic frameworks naturally recover the proposal of Ročková and George (2014). The general framework proposed in this paper can be viewed as an algorithmic counterpart of the theoretical framework for Bayesian high-dimensional structured linear models in Gao et al. (2015). While the work Gao et al. (2015) is focused on optimal posterior contraction rate and oracle inequalities, the current paper pursues a general efficient and scalable computational strategy.

In order to study various Bayesian models from a unified perspective, we introduce a spike-and-slab Laplacian prior distribution on the model parameters. The new prior distribution is an extension of the classical spike-and-slab prior (Mitchell and Beauchamp, 1988; George and McCulloch, 1993, 1997) for Bayesian variable selection. Our new definition incorporates the graph Laplacian of the underlying graph representing the model structure, and thus gives the name of the prior. Under this general framework, the problem of Bayesian model selection can be recast as selecting a subgraph of some base graph determined by the statistical task. Here, the base graph and its subgraphs represent the structures of the full model and the corresponding sub-models, respectively. Various choices of base graphs lead to specific statistical estimation problems such as sparse linear regression, clustering and change-point detection. In addition, the connection to graph algebra further allows us to build prior distributions for even more complicated models. For example, using graph products such as Cartesian product or Kronecker product (Imrich and Klavzar, 2000; Leskovec et al., 2010), we can construct prior distributions for biclustering models from the Laplacian of the graph products of row and column clustering structures. This leads to great flexibility in analyzing real data sets of complex structures.

Our Bayesian model selection follows the procedure of Ročková and George (2014) that evaluates the posterior probabilities of sub-models computed from the solution path of the EM algorithm. However, the derivation of the EM algorithm under our general framework is indeed nontrivial task. When the underlying base graph of the model structure is a tree, the derivation of the EM algorithm is straightforward by following the arguments in Ročková and George (2014). On the other hand, for a general base graph that is not a tree, the arguments in Ročková and George (2014) do not apply. To overcome this difficulty, we introduce a relaxation through the concept of effective resistance (Lovász, 1993; Ghosh et al., 2008; Spielman, 2007) that adapts to the underlying graphical structure of the model. The lower bound given by this relaxation is then used to derive a variational EM algorithm that works under the general framework.

Model selection with graph structured sparsity has also been studied in the frequentist literature. For example, generalized Lasso (Tibshirani and Taylor, 2011; Arnold and Tibshirani, 2014) and its multivariate version network Lasso (Hallac et al., 2015) encode the graph structured sparsity with ℓ_1 regularization. Algorithms based on ℓ_0 regularization have also been investigated recently (Fan and Guan, 2018; Xu and Fan, 2019). Compared with these frequentist methods, our proposed Bayesian model selection procedure tends to achieve better model selection performance in terms of false discovery proportion and power in a wide range of model scenarios, which will be shown through an extensive numerical study under various settings.

The rest of the paper is organized as follows. In Section 2, we introduce the general framework of Bayesian models and discuss the spike-and-slab Laplacian prior. The EM algorithm will be derived in Section 3 for both the case of trees and general base graphs. In Section 4, we discuss how to incorporate latent variables and propose a new Bayesian clustering models under our framework. Section 5 introduces the techniques of graph products and several important extensions of our framework. We will also discuss a non-Gaussian spike-and-slab Laplacian prior in Section 6 with a natural application to reduced isotonic regression (Schell and Singh, 1997). Finally, extensive simulated and real data analysis will be presented in Section 7.

2. A General Framework of Bayesian Models

In this section, we describe a general framework for building Bayesian structured models on graphs. To be specific, the prior structural assumption on the parameter $\theta \in \mathbb{R}^p$ will be encoded by a graph. Throughout the paper, $G = (V, E)$ is an undirected graph with $V = [p]$ and some $E \subset \{(i,j) : 1 \leq i < j \leq p\}$. It is referred to as the *base graph* of the model, and our goal is to learn a sparse subgraph of G from the data. We use $p = |V|$ and $m = |E|$ for the node size and edge size of the base graph.

2.1. Model Description

We start with the Gaussian linear model $y \,|\, \beta, \sigma^2 \sim N(X\beta, \sigma^2 I_n)$ that models an n-dimensional observation. The design matrix $X \in \mathbb{R}^{n\times p}$ is determined by the context of the problem. Given some nonzero vector $w \in \mathbb{R}^p$, the Euclidean space $\mathbb{R}^p$ can be decomposed as a direct sum of the one-dimensional subspace spanned by w and its orthogonal complement. In other words, we can write

$$\beta = \frac{1}{\|w\|^2} ww^T\beta + \left(I_p - \frac{1}{\|w\|^2} ww^T\right)\beta.$$

The structural assumption will be imposed by a prior on the second term above. To simplify the notation, we introduce the space $\Theta_w = \left\{\theta \in \mathbb{R}^p : w^T\theta = 0\right\}$. Then, any $\beta \in \mathbb{R}^p$ can be decomposed as $\beta = \alpha w + \theta$ for some $\alpha \in \mathbb{R}$ and $\theta \in \Theta_w$. The likelihood is thus given by

$$y \,|\, \alpha, \theta, \sigma^2 \sim N(X(\alpha w + \theta), \sigma^2 I_n). \tag{1}$$

The prior distribution on the vector $\alpha w + \theta$ will be specified by independent priors on α and θ. They are given by

$$\alpha \,|\, \sigma^2 \quad \sim \quad N(0, \sigma^2/\nu), \tag{2}$$

$$\theta \,|\, \gamma, \sigma^2 \quad \sim \quad p(\theta \,|\, \gamma, \sigma^2) \propto \prod_{(i,j)\in E} \exp\left(-\frac{(\theta_i - \theta_j)^2}{2\sigma^2[v_0\gamma_{ij} + v_1(1-\gamma_{ij})]}\right)\mathbb{I}\{\theta \in \Theta_w\}. \tag{3}$$

Under the prior distribution, α is centered at 0 and has precision ν/σ^2. The parameter θ is modeled by a prior distribution on Θ_w that encodes a pairwise relation between θ_i and θ_j. Here, v_0 is a very small scalar and v_1 is a very large scalar. For a pair $(i,j) \in E$ in the base graph, the prior enforces the closedness between θ_i and θ_j when $\gamma_{ij} = 1$. Our goal is then to learn the most probable subgraph structure encoded by $\{\gamma_{ij}\}$, which will be estimated from the posterior distribution.

We finish the Bayesian modeling by putting priors on γ and σ^2. They are given by

$$\begin{aligned}
\gamma \,|\, \eta &\sim p(\gamma \,|\, \eta) \propto \prod_{(i,j)\in E} \eta^{\gamma_{ij}}(1-\eta)^{1-\gamma_{ij}} \mathbb{I}\{\gamma \in \Gamma\}, && (4)\\
\eta &\sim \text{Beta}(A, B), && (5)\\
\sigma^2 &\sim \text{InvGamma}(a/2, b/2). && (6)
\end{aligned}$$

Besides the standard conjugate priors on η and σ^2, the independent Bernoulli prior on γ is restricted on a set $\Gamma \subset \{0,1\}^m$. This restriction is sometimes useful for particular models, but for now we assume that $\Gamma = \{0,1\}^m$ until it is needed in Section 4.

The Bayesian model is now fully specified. The joint distribution is

$$p(y, \alpha, \theta, \gamma, \eta, \sigma^2) = p(y \,|\, \alpha, \theta, \sigma^2) p(\alpha \,|\, \sigma^2) p(\theta \,|\, \gamma, \sigma^2) p(\gamma \,|\, \eta) p(\eta) p(\sigma^2). \tag{7}$$

Among these distributions, the most important one is $p(\theta|\gamma, \sigma^2)$. To understand its properties, we introduce the *incidence matrix* $D \in \mathbb{R}^{m\times p}$ for the base graph $G = (V, E)$. The matrix D has entries $D_{ei} = 1$ and $D_{ej} = -1$ if $e = (i, j)$, and $D_{ek} = 0$ if $k \neq i, j$. We note that the definition of D depends on the order of edges $\{(i, j)\}$ even if G is an undirected graph. However, this does not affect any application that we will need in the paper. We then define the Laplacian matrix

$$L_\gamma = D^T \operatorname{diag}\left(v_0^{-1}\gamma + v_1^{-1}(1-\gamma)\right) D.$$

It is easy to see that L_γ is the graph Laplacian of the weighted graph with adjacency matrix $\{v_0^{-1}\gamma_{ij} + v_1^{-1}(1-\gamma_{ij})\}$. Thus, we can write (3) as

$$p(\theta \,|\, \gamma, \sigma^2) \propto \exp\left(-\frac{1}{2\sigma^2}\theta^T L_\gamma \theta\right) \mathbb{I}\{\theta \in \Theta_w\}. \tag{8}$$

Given its form, we name (8) the *spike-and-slab Laplacian prior.*

Proposition 1 *Suppose $G = (V, E)$ is a connected base graph. For any $\gamma \in \{0,1\}^m$ and $v_0, v_1 \in (0, \infty)$, the graph Laplacian L_γ is positive semi-definite and has rank $p-1$. The only eigenvector corresponding to its zero eigenvalue is proportional to $\mathbb{1}_p$, the vector with all entries 1. As a consequence, as long as $\mathbb{1}_p^T w \neq 0$, the spike-and-slab Laplacian prior is a non-degenerate distribution on Θ_w. Its density function with respect to the Lebesgue measure restricted to Θ_w is*

$$p(\theta \,|\, \gamma, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{(p-1)/2}} \sqrt{det_w(L_\gamma)} \exp\left(-\frac{1}{2\sigma^2}\theta^T L_\gamma \theta\right) \mathbb{I}\{\theta \in \Theta_w\},$$

where $det_w(L_\gamma)$ is the product of all nonzero eigenvalues of the positive semi-definite matrix $\left(I_p - \frac{1}{\|w\|^2}ww^T\right) L_\gamma \left(I_p - \frac{1}{\|w\|^2}ww^T\right)$.

The proposition reveals two important conditions that lead to the well-definedness of the spike-and-slab Laplacian prior: the connectedness of the base graph $G = (V, E)$ and $\mathbb{1}_p^T w \neq 0$. Without either condition, the distribution would be degenerate on Θ_w. Extensions to a base graph that is not necessarily connected is possible. We leave this task to Section 4 and Section 5, where tools from graph algebra are introduced.

2.2. Examples

The Bayesian model (7) provides a very general framework. By choosing a different base graph $G = (V, E)$, a design matrix X, a grounding vector $w \in \mathbb{R}^p$ and a precision parameter ν, we then obtain a different model. Several important examples are given below.

Example 1 (Sparse linear regression) *The sparse linear regression model $y \,|\, \theta, \sigma^2 \sim N(X\theta, \sigma^2 I_n)$ is a special case of (1). To put it into the general framework, we can expand the design matrix $X \in \mathbb{R}^{n\times p}$ and the regression vector $\theta \in \mathbb{R}^p$ by $[0_n, X] \in \mathbb{R}^{n\times(p+1)}$ and $[\theta_0; \theta] \in \mathbb{R}^{p+1}$. With the grounding vector $w = [1; 0_p]$, the sparse linear regression model can be recovered from (1). For the prior distribution, the base graph G consists of nodes $V = \{0, 1, ..., p\}$ and edges $\{(0, i) : i \in [p]\}$. We set $\nu = \infty$, so that $\theta_0 = 0$ with prior probability one. Then, (3) is reduced to*

$$\theta \,|\, \gamma, \sigma^2 \sim p(\theta \,|\, \gamma, \sigma^2) \propto \prod_{i=1}^{p} \exp\left(-\frac{\theta_i^2}{2\sigma^2[v_0\gamma_{0i} + v_1(1-\gamma_{0i})]}\right).$$

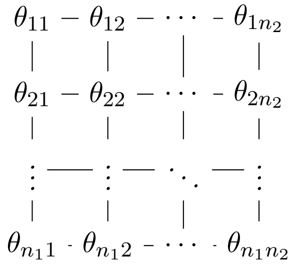
That is, $\theta_i|\gamma, \sigma^2 \sim N(0, \sigma^2[v_0\gamma_{0i}+v_1(1-\gamma_{0i})])$ independently for all $i \in [n]$. This is recognized as the spike-and-slab Gaussian prior for Bayesian sparse linear regression considered by George and McCulloch (1993, 1997); Ročková and George (2014).

Example 2 (Change-point detection) *Set $n = p$, $X = I_n$, and $w = \mathbb{1}_n$. We then have $y_i \,|\, \theta_i, \sigma^2 \sim N(\alpha + \theta_i, \sigma^2)$ independently for all $i \in [n]$ from (1). For the prior distribution on α and θ, we consider $\nu = 0$ and a one-dimensional chain graph $G = (V, E)$ with $E = \{(i, i+1) : i \in [n-1]\}$. This leads to a flat prior on α, and the prior on θ is given by*

$$\theta \,|\, \gamma, \sigma^2 \sim p(\theta \,|\, \gamma, \sigma^2) \propto \prod_{i=1}^{n-1} \exp\left(-\frac{(\theta_i - \theta_{i+1})^2}{2\sigma^2[v_0\gamma_{i,i+1} + v_1(1-\gamma_{i,i+1})]}\right) \mathbb{I}\{\mathbb{1}_p^T\theta = 0\}.$$

A more general change-point model on a tree can also be obtained by constructing a tree base graph G.

Example 3 (Two-dimensional image denoising) *Consider a rectangular set of observations $y \in \mathbb{R}^{n_1\times n_2}$. With the same construction in Example 2 applied to $\mathsf{sec}(y)$, we obtain $y_{ij} \,|\, \theta_{ij}, \sigma^2 \sim N(\alpha + \theta_{ij}, \sigma^2)$ independently for all $(i, j) \in [n_1] \times [n_2]$ from (1). To model images, we consider a prior distribution that imposes closedness to nearby pixels. Consider $\nu = 0$ and a base graph $G = (V, E)$ shown in the picture below.*



We then obtain a flat prior on α, and

$$\theta \,|\, \gamma, \sigma^2 \sim p(\theta \,|\, \gamma, \sigma^2) \propto \prod_{(ik,jl)\in E} \exp\left(-\frac{(\theta_{ik}-\theta_{jl})^2}{2\sigma^2[v_0\gamma_{ik,jl} + v_1(1-\gamma_{ik,jl})]} \right) \mathbb{I}\{\mathbb{1}_{n_1}^T \theta \mathbb{1}_{n_2} = 0\}.$$

Note that G is not a tree in this case.

3. EM Algorithm

In this section, we will develop efficient EM algorithms for the general model. It turns out that the bottleneck is the computation of $\det_w(L_\gamma)$ given some $\gamma \in \{0,1\}^m$.

Lemma 2 *Let spt(G) be the set of all spanning trees of G. Then*

$$det_w(L_\gamma) = \frac{(\mathbb{1}_p^T w)^2}{\|w\|^2} \sum_{T\in spt(G)} \prod_{(i,j)\in T} \left[v_0^{-1}\gamma_{ij} + v_1^{-1}(1-\gamma_{ij}) \right].$$

In particular, if G is a tree, then $det_w(L_\gamma) = \frac{(\mathbb{1}_p^T w)^2}{\|w\|^2} \prod_{(i,j)\in E} \left[v_0^{-1}\gamma_{ij} + v_1^{-1}(1-\gamma_{ij}) \right]$.

The lemma suggests that the hardness of computing $\det_w(L_\gamma)$ depends on the number of spanning trees of the base graph G. When the base graph is a tree, $\det_w(L_\gamma)$ is factorized over the edges of the tree, which greatly simplifies the derivation of the algorithm. We will derive a closed-form EM algorithm in Section 3.1 when G is a tree, and the algorithm for a general G will be given in Section 3.2.

3.1. The Case of Trees

We treat γ as latent. Our goal is to maximize the marginal distribution after integrating out the latent variables. That is,

$$\max_{\alpha,\theta\in\Theta_w,\eta,\sigma^2} \log \sum_{\gamma} p(y,\alpha,\theta,\gamma,\eta,\sigma^2), \tag{9}$$

where $p(y,\alpha,\theta,\gamma,\eta,\sigma^2)$ is given by (7). Since the summation over γ is intractable, we consider an equivalent form of (9), which is

$$\max_{q} \max_{\alpha,\theta\in\Theta_w,\eta,\sigma^2} \sum_{\gamma} q(\gamma) \log \frac{p(y,\alpha,\theta,\gamma,\eta,\sigma^2)}{q(\gamma)}. \tag{10}$$

Then, the EM algorithm is equivalent to iteratively updating $q, \alpha, \theta \in \Theta_w, \eta, \sigma^2$ (Neal and Hinton, 1998).

Now we illustrate the EM algorithm that solves (10). The E-step is to update $q(\gamma)$ given the previous values of θ, η, σ. In view of (7), we have

$$q^{\text{new}}(\gamma) \propto p(y,\alpha,\theta,\gamma,\eta,\sigma^2) \propto p(\theta \,|\, \gamma, \sigma^2) p(\gamma \,|\, \eta). \tag{11}$$

According to (2), $p(\theta \,|\, \gamma, \sigma^2)$ can be factorized when the base graph $G = (V, E)$ is a tree. Therefore, with a simpler notation $q_{ij} = q(\gamma_{ij} = 1)$, we can write the update for q as $q^{\text{new}}(\gamma) = \prod_{(i,j)\in E} (q_{ij}^{\text{new}})^{\gamma_{ij}} (1 - q_{ij}^{\text{new}})^{1-\gamma_{ij}}$, where

$$q_{ij}^{\text{new}} = \frac{\eta\phi(\theta_i - \theta_j; 0, \sigma^2 v_0)}{\eta\phi(\theta_i - \theta_j; 0, \sigma^2 v_0) + (1-\eta)\phi(\theta_i - \theta_j; 0, \sigma^2 v_1)}. \tag{12}$$

Here, $\phi(\cdot; \mu, \sigma^2)$ stands for the density function of $N(\mu, \sigma^2)$.

To derive the M-step, we introduce the following function

$$F(\alpha, \theta; q) = \|y - X(\alpha w + \theta)\|^2 + \nu\alpha^2 + \theta^T L_q \theta, \tag{13}$$

where L_q is obtained by replacing γ with q in the definition of the graph Laplacian L_γ. The M-step consists of the following three updates,

$$\begin{aligned}
(\alpha^{\text{new}}, \theta^{\text{new}}) &= \operatorname*{argmin}_{\alpha, \theta \in \Theta_w} F(\alpha, \theta; q^{\text{new}}), & (14)\\
(\sigma^2)^{\text{new}} &= \operatorname*{argmin}_{\sigma^2} \left[\frac{F(\alpha^{\text{new}}, \theta^{\text{new}}; q^{\text{new}}) + b}{2\sigma^2} + \frac{p+n+a+2}{2} \log(\sigma^2) \right], & (15)\\
\eta^{\text{new}} &= \operatorname*{argmax}_{\eta} \left[(A - 1 + q_{\text{sum}}^{\text{new}}) \log \eta + (B - 1 + p - 1 - q_{\text{sum}}^{\text{new}}) \log(1-\eta) \right], & (16)
\end{aligned}$$

where the notation $q_{\text{sum}}^{\text{new}}$ stands for $\sum_{(i,j)\in E} q_{ij}^{\text{new}}$. While (14) is a simple quadratic programming, (15) and (16) have closed forms, which are given by

$$(\sigma^2)^{\text{new}} = \frac{F(\alpha^{\text{new}}, \theta^{\text{new}}; q^{\text{new}}) + b}{p+n+a+2} \quad \text{and} \quad \eta^{\text{new}} = \frac{A - 1 + q_{\text{sum}}^{\text{new}}}{A+B+p-3}. \tag{17}$$

We remark that the EMVS algorithm (Ročková and George, 2014) is a special case for the sparse linear regression problem discussed in Example 1. When G is a tree, the spike-and-slab graph Laplacian prior (8) is proportional to the product of individual spike-and-slab priors

$$p(\theta \,|\, \gamma, \sigma^2) \propto \prod_{(i,j)\in E} \exp\left(-\frac{(\theta_i - \theta_j)^2}{2\sigma^2[v_0\gamma_{ij} + v_1(1-\gamma_{ij})]} \right),$$

supported on Θ_w, as we have seen in Example 1 and 2. In this case, the above EM algorithm we have developed can also be extended to models with alternative prior distributions, such as the spike-and-slab Lasso prior (Ročková and George, 2018) and the finite normal mixture prior (Stephens, 2016).

3.2. General Graphs

When the base graph G is not a tree, the E-step becomes computationally infeasible due to the lack of separability of $p(\theta|\gamma, \sigma^2)$ in γ. In fact, given the form of the density function in Proposition 1, the main problem lies in the term $\sqrt{\det_w(L_\gamma)}$, which cannot be factorized over $(i,j) \in E$ when the base graph $G = (V, E)$ is not a tree (Lemma 2). To overcome the difficulty, we consider optimizing a lower bound of the objective function (10). This means we need to find a good lower bound for $\log\det_w(L_\gamma)$. Similar techniques are also advocated in the context of learning exponential family graphical models (Wainwright and Jordan, 2008).

By Lemma 2, we can write

$$\log \det_w(L_\gamma) = \log \sum_{T\in \mathrm{spt}(G)} \prod_{(i,j)\in T} \left[v_0^{-1}\gamma_{ij} + v_1^{-1}(1-\gamma_{ij})\right] + \log \frac{(\mathbb{1}_p^T w)^2}{\|w\|^2}. \tag{18}$$

We only need to lower bound the first term on the right hand side of the equation above, because the second term is independent of γ. By Jensen's inequality, for any non-negative sequence $\{\lambda(T)\}_{T\in \mathrm{spt}(G)}$ such that $\sum_{T\in \mathrm{spt}(G)} \lambda(T) = 1$, we have

$$\begin{aligned}
& \log \sum_{T\in \mathrm{spt}(G)} \prod_{(i,j)\in T} \left[v_0^{-1}\gamma_{ij} + v_1^{-1}(1-\gamma_{ij})\right] \\
\geq & \sum_{T\in \mathrm{spt}(G)} \lambda(T) \log \prod_{(i,j)\in T} \left[v_0^{-1}\gamma_{ij} + v_1^{-1}(1-\gamma_{ij})\right] - \sum_{T\in \mathrm{spt}(G)} \lambda(T)\log\lambda(T) \\
= & \sum_{(i,j)\in E} \left(\sum_{T\in \mathrm{spt}(G)} \lambda(T)\mathbb{I}\{(i,j)\in T\} \right) \log \left[v_0^{-1}\gamma_{ij} + v_1^{-1}(1-\gamma_{ij})\right] - \sum_{T\in \mathrm{spt}(G)} \lambda(T)\log\lambda(T).
\end{aligned}$$

One of the most natural choices of the weights $\{\lambda(T)\}_{T\in \mathrm{spt}(G)}$ is the uniform distribution

$$\lambda(T) = \frac{1}{|\mathrm{spt}(G)|}.$$

This leads to the following lower bound

$$\begin{aligned}
& \log \sum_{T\in \mathrm{spt}(G)} \prod_{(i,j)\in T} \left[v_0^{-1}\gamma_{ij} + v_1^{-1}(1-\gamma_{ij})\right] \\
\geq & \sum_{(i,j)\in E} r_{ij} \log \left[v_0^{-1}\gamma_{ij} + v_1^{-1}(1-\gamma_{ij})\right] + \log|\mathrm{spt}(G)|,
\end{aligned} \tag{19}$$

where

$$r_{ij} = \frac{1}{|\mathrm{spt}(G)|} \sum_{T\in \mathrm{spt}(G)} \mathbb{I}\{(i,j)\in T\}. \tag{20}$$

The quantity r_{ij} defined in (20) is recognized as the *effective resistance* between the ith and the jth nodes (Lovász, 1993; Ghosh et al., 2008). Given a graph, we can treat each edge as a resistor with resistance 1. Then, the effective resistance between the ith and the jth nodes is the resistance between i and j given by the whole graph. That is, if we treat the entire graph as a resistor. Let L be the (unweighted) Laplacian matrix of the base graph $G = (V, E)$, and L^+ its pseudo-inverse. Then, an equivalent definition of (20) is given by the formula

$$r_{ij} = (e_i - e_j)^T L^+ (e_i - e_j),$$

where e_j is the basis vector with the ith entry 1 and the remaining entries 0. Therefore, computation of the effective resistance can leverage fast Laplacian solvers in the literature (Spielman and Teng, 2004; Livne and Brandt, 2012). Some important examples of effective resistance are listed below:

- When G is the complete graph of size p, then $r_{ij} = 2/p$ for all $(i,j) \in E$.
- When G is the complete bipartite graph of sizes p and k, then $r_{ij} = \frac{p+k-1}{pk}$ for all $(i,j) \in E$.
- When G is a tree, then $r_{ij} = 1$ for all $(i,j) \in E$.
- When G is a two-dimensional grid graph of size $n_1 \times n_2$, then $r_{ij} \in [0.5, 0.75]$ depending on how close the edge (i,j) is from its closest corner.
- When G is a lollipop graph, the conjunction of a linear chain with size p and a complete graph with size k, then $r_{ij} = 1$ or $2/k$ depending on whether the edge (i,j) belongs to the chain or the complete graph.

By (19), we obtain the following lower bound for the objective function (10),

$$\max_q \max_{\alpha,\theta\in\Theta_w,\eta,\sigma^2} \sum_\gamma q(\gamma) \log \frac{p(y \,|\, \alpha,\theta,\sigma^2)p(\alpha \,|\, \sigma^2)\widetilde{p}(\theta \,|\, \gamma,\sigma^2)p(\gamma \,|\, \eta)p(\eta)p(\sigma^2)}{q(\gamma)}, \tag{21}$$

where the formula of $\widetilde{p}(\theta \,|\, \gamma,\sigma^2)$ is obtained by applying the lower bound (19) in the formula of $p(\theta \,|\, \gamma,\sigma^2)$ in Proposition 1. Since $\widetilde{p}(\theta \,|\, \gamma,\sigma^2)$ can be factorized over $(i,j) \in E$, the E-step is given by $q^{\text{new}}(\gamma) = \prod_{(i,j)\in E}(q_{ij}^{\text{new}})^{\gamma_{ij}}(1-q_{ij}^{\text{new}})^{1-\gamma_{ij}}$, where

$$q_{ij}^{\text{new}} = \frac{\eta v_0^{-r_{ij}/2} e^{-(\theta_i-\theta_j)^2/2\sigma^2 v_0}}{\eta v_0^{-r_{ij}/2} e^{-(\theta_i-\theta_j)^2/2\sigma^2 v_0} + (1-\eta) v_1^{-r_{ij}/2} e^{-(\theta_i-\theta_j)^2/2\sigma^2 v_1}}. \tag{22}$$

Observe that the lower bound (19) is independent of $\alpha, \theta, \eta, \sigma^2$, and thus the M-step remains the same as in the case of a tree base graph. The formulas are given by (14)-(16), except that (16) needs to be replaced by

$$\eta^{\text{new}} = \frac{A - 1 + q_{\text{sum}}^{\text{new}}}{A+B+m-2}.$$

The EM algorithm for a general base graph can be viewed as a natural extension of that of a tree base graph. When $G = (V,E)$ is a tree, it is easy to see from the formula (20) that $r_{ij} = 1$ for all $(i,j) \in E$. In this case, the E-step (22) is reduced to (12), and the inequality (19) becomes an equality.

3.3. Bayesian Model Selection

The output of the EM algorithm $\widehat{q}(\gamma)$ can be understood as an estimator of the posterior distribution $p(\gamma|\widehat{\alpha}, \widehat{\theta}, \widehat{\sigma}^2, \widehat{\eta})$, where $\widehat{\alpha}, \widehat{\theta}, \widehat{\sigma}^2, \widehat{\eta}$ are obtained from the M-step. Then, we get a subgraph according to the thresholding rule $\widehat{\gamma}_{ij} = \mathbb{I}\{\widehat{q}_{ij} \geq 1/2\}$. It can be understood as a model learned from the data. The sparsity of the model critically depends on the values of v_0 and v_1 in the spike-and-slab Laplacian prior. With a fixed large value of v_1, we can obtain the solution path of $\widehat{\gamma} = \widehat{\gamma}(v_0)$ by varying v_0 from 0 to v_1. The question then is how to select the best model along the solution path of the EM algorithm.

The strategy suggested by Ročková and George (2014) is to calculate the posterior score $p(\gamma|y)$ with respect to the Bayesian model of $v_0 = 0$. While the meaning of $p(\gamma|y)$

corresponding to $v_0 = 0$ is easily understood for the sparse linear regression setting in Ročková and George (2014), it is less clear for a general base graph $G = (V, E)$.

In order to define a version of (7) for $v_0 = 0$, we need to introduce the concept of *edge contraction.* Given a $\gamma \in \{0, 1\}^m$, the graph corresponding to the adjacency matrix γ induces a partition of disconnected components $\{\mathcal{C}_1, ..., \mathcal{C}_s\}$ of $[p]$. In other words, $\{i, j\} \subset \mathcal{C}_l$ for some $l \in [s]$ if and only if there is some path between i and j in the graph γ. For notational convenience, we define a vector $z \in [s]^n$ so that $z_i = l$ if and only if $i \in \mathcal{C}_l$. A membership matrix $Z_\gamma \in \{0, 1\}^{p \times s}$ is defined with its (i, l)th entry being the indicator $\mathbb{I}\{z_i = l\}$.

We let $\widetilde{G} = (\widetilde{V}, \widetilde{E})$ be a graph obtained from the base graph $G = (V, E)$ after the operation of edge contraction. In other words, every node in $\widetilde{G}$ is obtained by combining nodes in G according to the partition of $\{\mathcal{C}_1, ..., \mathcal{C}_s\}$. To be specific, $\widetilde{V} = [s]$, and $(k, l) \in \widetilde{E}$ if and only if there exists some $i \in \mathcal{C}_k$ and some $j \in \mathcal{C}_l$ such that $(i, j) \in E$.

Now we are ready to define a limiting version of (3) as $v_0 \to 0$. Let $\widetilde{L}_\gamma = D^T \operatorname{diag}(v_1^{-1}(1-\gamma))D$, which is the graph Laplacian of the weighted graph with adjacency matrix $\{v_1^{-1}(1 - \gamma_{ij})\}$. Then, define

$$p(\widetilde{\theta} \,|\, \gamma, \sigma^2) = \frac{1}{(2\pi\sigma^2)^{(s-1)/2}} \sqrt{\det_{Z_\gamma^T w}(Z_\gamma^T \widetilde{L}_\gamma Z_\gamma)} \exp\left(-\frac{\widetilde{\theta}^T Z_\gamma^T \widetilde{L}_\gamma Z_\gamma \widetilde{\theta}}{2\sigma^2} \right) \mathbb{I}\{\widetilde{\theta} \in \Theta_{Z_\gamma^T w}\}. \quad (23)$$

With $\widetilde{G} = (\widetilde{V}, \widetilde{E})$ standing for the contracted base graph, the prior distribution (23) can also be written as

$$p(\widetilde{\theta} \,|\, \gamma, \sigma^2) \propto \exp\left(-\sum_{(k,l) \in \widetilde{E}} \frac{\omega_{kl}(\widetilde{\theta}_k - \widetilde{\theta}_l)^2}{2\sigma^2 v_1} \right) \mathbb{I}\{\widetilde{\theta} \in \Theta_{Z_\gamma^T w}\}, \quad (24)$$

where $\omega_{kl} = \sum_{(i,j) \in E} \mathbb{I}\{z(i) = k, z(j) = l\}$, which means that the edges $\{(i, j)\}_{z(i)=k, z(j)=l}$ in the base graph $G = (V, E)$ are contracted as a new edge (k, l) in $\widetilde{G} = (\widetilde{V}, \widetilde{E})$ with ω_{kl} as the weight.

Proposition 3 *Suppose $G = (V, E)$ is connected and $\mathbb{1}_p^T w \neq 0$. Let Z_γ be the membership matrix defined as above. Then for any $\gamma \in \{0, 1\}^m$, (23) is a well-defined density function on the $(s - 1)$-dimensional subspace $\{\widetilde{\theta} \in \mathbb{R}^s : w^T Z_\gamma \widetilde{\theta} = 0\}$. Moreover, for an arbitrary design matrix $X \in \mathbb{R}^{n \times p}$, the distribution of θ that follows (3) weakly converges to that of $Z_\gamma \widetilde{\theta}$ as $v_0 \to 0$.*

Motivated by Proposition 3, a limiting version of (7) for $v_0 = 0$ is defined as follows,

$$y \,|\, \alpha, \widetilde{\theta}, \gamma, \sigma^2 \sim N(X(\alpha w + Z_\gamma \widetilde{\theta}), \sigma^2 I_n). \quad (25)$$

Then, $p(\widetilde{\theta} \,|\, \gamma, \sigma^2)$ is given by (23), and $p(\alpha|\sigma^2)$, $p(\gamma|\eta)$, $p(\eta)$, $p(\sigma^2)$ are specified in (2) and (4)-(6). The posterior distribution of γ has the formula

$$\begin{aligned} p(\gamma \,|\, y) &\propto \int\int\int\int p(y, \alpha, \widetilde{\theta}, \gamma, \eta, \sigma^2)\, d\alpha\, d\widetilde{\theta}\, d\eta\, d\sigma^2 \\ &= \int p(\sigma^2) \int p(\alpha \,|\, \sigma^2) \int p(y \,|\, \alpha, \widetilde{\theta}, \gamma, \sigma^2) p(\widetilde{\theta} \,|\, \gamma, \sigma^2)\, d\widetilde{\theta}\, d\alpha\, d\sigma^2 \int p(\gamma \,|\, \eta) p(\eta) d\eta. \end{aligned}$$

A standard calculation using conjugacy gives

$$\begin{aligned} p(\gamma \,|\, y) \propto & \left(\frac{\det_{Z_\gamma^T w}(Z_\gamma^T \widetilde{L}_\gamma Z_\gamma)}{\det_{Z_\gamma^T w}(Z_\gamma^T (X^T X + \widetilde{L}_\gamma) Z_\gamma)} \right)^{1/2} \left(\frac{\nu}{\nu + w^T X^T (I_n - R_\gamma) X w} \right)^{1/2} \\ & \times \left(y^T (I_n - R_\gamma) y - \frac{|w^T X^T (I_n - R_\gamma) y|^2}{\nu + w^T X^T (I_n - R_\gamma) X w} + b \right)^{-\frac{n+a}{2}} \\ & \times \frac{\mathrm{Beta}\left(\sum_{(i,j)\in E} \gamma_{ij} + A - 1, \, \sum_{(i,j)\in E} (1 - \gamma_{ij}) + B - 1 \right)}{\mathrm{Beta}(A, B)}, \end{aligned} \tag{26}$$

where

$$R_\gamma = X Z_\gamma (Z_\gamma^T (X^T X + \widetilde{L}_\gamma) Z_\gamma)^{-1} Z_\gamma^T X^T.$$

This defines the model selection score $g(\gamma) = \log p(\gamma \,|\, y)$ up to a universal additive constant. The Bayesian model selection procedure evaluates $g(\gamma)$ on the solution path $\{\widehat{\gamma}(v_0)\}_{0 < v_0 \leq v_1}$ and selects the best model with the highest value of $g(\gamma)$.

3.4. Summary of Our Approach

The parameter v_0 plays a critical role in our Bayesian model selection procedure. Recall that the joint distribution of our focus has the expression

$$p(y \,|\, \theta) p_{v_0}(\theta \,|\, \gamma) p(\gamma), \tag{27}$$

where $p(y \,|\, \theta)$ is the linear model parametrized by θ, $p_{v_0}(\theta \,|\, \gamma)$ is the spike-and-slab prior with tuning parameter v_0, and $p(\gamma)$ is the prior on the graph[1]. The tuning parameter v_0 is the variance of the spike component of the prior. Different choices of v_0 are used as different components in our entire model selection procedure.

1. The ideal choice of $v_0 = 0$ models exact sparsity in the sense that $\gamma_{ij} = 1$ implies $\theta_i = \theta_j$. In this case, the exact posterior

$$p_{v_0 = 0}(\gamma \,|\, y) \propto \int p(y \,|\, \theta) p_{v_0 = 0}(\theta \,|\, \gamma) p(\gamma) d\theta$$

can be calculated according to the formulas that we derive in Section 3.3. Then the ideal Bayes model selection procedure would be the one that maximizes the posterior $p_{v_0=0}(\gamma \,|\, y)$ over all γ. However, since this would require evaluating $p_{v_0=0}(\gamma \,|\, y)$ for exponentially many γ's, it is sensible to maximize $p_{v_0=0}(\gamma \,|\, y)$ only over a carefully chosen subset of γ that has a reasonable size for computational efficiency.

2. The choice of $v_0 > 0$ models approximate sparsity in the sense that γ_{ij} implies $\theta_i \approx \theta_j$. Though $v_0 > 0$ does not offer interpretation of exact sparsity, a nonzero v_0 leads to efficient computation via the EM algorithm. That is, for a $v_0 > 0$, we can maximize

$$\max_q \max_\theta \sum_\gamma q(\gamma) \log \frac{p(y \,|\, \theta) p_{v_0}(\theta \,|\, \gamma) p(\gamma)}{q(\gamma)},$$

[1] We have ignored other parameters such as α, η, σ^2 in order to make the discussion below clear and concise.

which is the objective function of EM. Denote the output of the algorithm by $q_{v_0}(\gamma) = \prod_{ij} q_{ij,v_0}$, we then obtain our model by $\widehat{\gamma}_{ij}(v_0) = \mathbb{I}\{q_{ij,v_0} > 0.5\}$. As we vary v_0 on a grid from 0 to v_1, we obtain a path of models $\{\widehat{\gamma}(v_0)\}_{0<v_0\leq v_1}$. It covers models that ranges from very parsimonious ones to the fully saturated one.

The proposed model selection procedure in Section 3.3 is

$$\max_{\gamma\in\{\widehat{\gamma}(v_0)\}_{0<v_0\leq v_1}} p_{v_0=0}(\gamma\,|\,y), \tag{28}$$

where $p_{v_0=0}(\gamma\,|\,y) \propto \int p(y\,|\,\theta)p_{v_0=0}(\theta\,|\,\gamma)p(\gamma)d\theta$. That is, we optimize the posterior distribution $p_{v_0=0}(\gamma\,|\,y)$ *only* over the EM solution path. The best one among all the candidate models will be selected according to $p_{v_0=0}(\gamma\,|\,y)$, which is the exact/full posterior of γ. There are two ways to interpret our model selection procedure (28):

1. The procedure (28) can be understood as a computationally efficient approximation strategy to the ideal Bayes model selection procedure $\max_\gamma p_{v_0=0}(\gamma\,|\,y)$ that is infeasible to compute. The EM algorithm with various choices of v_0 simply provides a short list of candidate models. From this perspective, the proposed procedure (28) is fully Bayes.

2. The procedure (28) can be also understood as a method for selecting the tuning parameter v_0, because the maximizer of (28) must be in the form of $\widehat{\gamma}(\widehat{v}_0)$ for some data-driven $\widehat{v}_0$. In this way, the solution $\widehat{\gamma} = \widehat{\gamma}(\widehat{v}_0)$ also has a non-Bayesian interpretation since it is obtained by post-processing the EM solution with the tuning parameter $\widehat{v}_0$ selected by (28). In this regard, $\widehat{\gamma}$ also can be thought of as an empirical Bayes estimator.

In summary, our proposed procedure (28) is motivated by both statistical and computational considerations.

4. Clustering: A New Deal

4.1. A Multivariate Extension

Before introducing our new Bayesian clustering model, we need a multivariate extension of the general framework (7) to model a matrix observation $y \in \mathbb{R}^{n\times d}$. With a design matrix $X \in \mathbb{R}^{n\times p}$, the dimension of θ is now $p \times d$. We denote the ith row of θ by θ_i. With the grounding vector $w \in \mathbb{R}^p$, the distribution $p(y|\alpha,\theta,\sigma^2)p(\alpha|\sigma^2)p(\theta|\gamma,\sigma^2)$ is given by

$$\begin{aligned} y\,|\,\alpha,\theta,\sigma^2 &\sim N(X(w\alpha^T+\theta),\sigma^2 I_n\otimes I_d), && (29)\\ \alpha\,|\,\sigma^2 &\sim N\left(0,\frac{\sigma^2}{\nu}I_d\right), && (30)\\ \theta\,|\,\gamma,\sigma^2 &\sim p(\theta|\gamma,\sigma^2) \propto \prod_{(i,j)\in E}\exp\left(-\frac{\|\theta_i-\theta_j\|^2}{2\sigma^2[v_0\gamma_{ij}+v_1(1-\gamma_{ij})]}\right)\mathbb{I}\{\theta\in\Theta_w\}, && (31)\end{aligned}$$

where $\Theta_w = \{\theta \in \mathbb{R}^{p\times d} : w^T\theta = 0\}$. The prior distributions on γ,η,σ^2 are the same as (4)-(6). Moreover, the multivariate spike-and-slab Laplacian prior (31) is supported on a

$d(p-1)$-dimensional subspace Θ_w, and is well-defined as long as $\mathbb{1}_p^T w \neq 0$ for the same reason stated in Proposition 1.

The multivariate extension can be understood as the task of learning d individual graphs for each column of θ. Instead of modeling the d graph separately by $\gamma^{(1)}, ..., \gamma^{(d)}$ using (7), we assume the d columns of θ share the same structure by imposing the condition $\gamma^{(1)} = ... = \gamma^{(d)}$.

An immediate example is a Bayesian multitask learning problem with group sparsity. It can be viewed as a multivariate extension of Example 1. With the same argument in Example 1, (29)-(31) is specialized to

$$\begin{aligned} y \,|\, \theta, \sigma^2 &\sim N(X\theta, \sigma^2 I_n \otimes I_d), \\ \theta \,|\, \gamma, \sigma^2 &\sim p(\theta \,|\, \gamma, \sigma^2) \propto \prod_{i=1}^{p} \exp\left(-\frac{\|\theta_i\|^2}{2\sigma^2[v_0\gamma_i + v_1(1-\gamma_i)]}\right). \end{aligned}$$

To close this subsection, let us mentions that the model (29)-(31) can be easily modified to accommodate a heteroscedastic setting. For example, one can replace the $\sigma^2 I_n \otimes I_d$ in (29) by a more general $I_n \otimes \mathrm{diag}(\sigma_1^2, ..., \sigma_d^2)$, and then make corresponding changes to (30) and (31) as well.

4.2. Model Description

Consider the likelihood

$$y \,|\, \alpha, \theta, \sigma^2 \sim N(\mathbb{1}_n \alpha^T + \theta, \sigma^2 I_n \otimes I_d), \tag{32}$$

with the prior distribution of $\alpha|\sigma^2$ specified by (30). The clustering model uses the following form of (31),

$$p(\theta, \mu \,|\, \gamma, \sigma^2) \propto \prod_{i=1}^{n}\prod_{j=1}^{k} \exp\left(-\frac{\|\theta_i - \mu_j\|^2}{2\sigma^2[v_0\gamma_{ij} + v_1(1-\gamma_{ij})]}\right) \mathbb{I}\{\mathbb{1}_n^T \theta = 0\}. \tag{33}$$

Here, both vectors θ_i and μ_j are in $\mathbb{R}^d$. The prior distribution (33) can be derived from (31) by replacing θ in (31) with (θ, μ) and specifying the base graph as a complete bipartite graph between θ and μ. We impose the restriction that

$$\sum_{j=1}^{k} \gamma_{ij} = 1, \tag{34}$$

for all $i \in [n]$. Then, $\mu_1, ..., \mu_k$ are latent variables that can be interpreted as the clustering centers, and each θ_i is connected to one of the clustering centers.

To fully specify the clustering model, the prior distribution of γ is given by (4) with Γ being the set of all $\{\gamma_{ij}\}$ that satisfies (34). Equivalently,

$$(\gamma_{i1}, ..., \gamma_{ik}) \sim \mathrm{Uniform}(\{e_j\}_{j=1}^{k}), \tag{35}$$

independently for all $i \in [n]$, where e_j is a vector with 1 on the jth entry and 0 elsewhere. Finally, the prior of σ^2 is given by (6).

4.3. EM Algorithm

The EM algorithm can be derived by following the idea developed in Section 3.2. In the current setting, the lower bound (19) becomes

$$\begin{aligned} & \log \sum_{T\in \mathrm{spt}(K_{n,k})} \prod_{i=1}^{n}\prod_{j=1}^{k}[v_0^{-1}\gamma_{ij}+v_1^{-1}(1-\gamma_{ij})] \\ \geq & \sum_{i=1}^{n}\sum_{j=1}^{k} r_{ij}\log\left[v_0^{-1}\gamma_{ij}+v_1^{-1}(1-\gamma_{ij})\right]+\log|\mathrm{spt}(K_{n,k})|, \end{aligned} \tag{36}$$

where $K_{n,k}$ is the complete bipartite graph. By symmetry, the effective resistance $r_{ij}=r$ is a constant independent of (i,j). Thus, (36) can be written as

$$\begin{aligned} & r\sum_{i=1}^{n}\sum_{j=1}^{k}\log\left[v_0^{-1}\gamma_{ij}+v_1^{-1}(1-\gamma_{ij})\right]+\log|\mathrm{spt}(K_{n,k})| \qquad (37) \\ = & \; r\log(v_0^{-1})\sum_{i=1}^{n}\sum_{j=1}^{k}\gamma_{ij}+r\log(v_1^{-1})\sum_{i=1}^{n}\sum_{j=1}^{k}(1-\gamma_{ij})+\log|\mathrm{spt}(K_{n,k})| \\ = & \; rn\log(v_0^{-1})+rn(k-1)\log(v_1^{-1})+\log|\mathrm{spt}(K_{n,k})|, \qquad (38) \end{aligned}$$

where the last equality is derived from (34). Therefore, for the clustering model, the lower bound (19) is a constant independent of $\{\gamma_{ij}\}$. As a result, the lower bound of the objective function of the EM algorithm becomes

$$\sum_{\gamma} q(\gamma)\log\frac{p(y\,|\,\alpha,\theta,\sigma^2)p(\alpha\,|\,\sigma^2)\widetilde{p}(\theta,\mu\,|\,\gamma,\sigma^2)p(\gamma)p(\sigma^2)}{q(\gamma)}, \tag{39}$$

with

$$\widetilde{p}(\theta,\mu\,|\,\gamma,\sigma^2)=\text{const}\times\frac{1}{(2\pi\sigma^2)^{(n+k-1)d/2}}\prod_{i=1}^{n}\prod_{j=1}^{k}\exp\left(-\frac{\|\theta_i-\mu_j\|^2}{2\sigma^2[v_0\gamma_{ij}+v_1(1-\gamma_{ij})]}\right),$$

and the algorithm is to maximize (39) over $q,\alpha,\theta\in\{\theta:\mathbb{1}_n^T\theta=0\}$ and σ^2.

Maximizing (39) over q, we obtain the E-step as

$$q_{ij}^{\text{new}}=\frac{\exp\left(-\frac{\|\theta_i-\mu_j\|^2}{2\sigma^2\bar{v}}\right)}{\sum_{l=1}^{k}\exp\left(-\frac{\|\theta_i-\mu_l\|^2}{2\sigma^2\bar{v}}\right)}, \tag{40}$$

independently for all $i\in[n]$, where $\bar{v}^{-1}=v_0^{-1}-v_1^{-1}$, and we have used the notation

$$q_{ij}=q\left((\gamma_{i1},...,\gamma_{ik})=e_j\right).$$

It is interesting to note that the E-step only depends on v_0 and v_1 through $\bar{v}$. Maximizing (39) over $\alpha,\theta\in\{\theta:\mathbb{1}_n^T\theta=0\},\mu,\sigma^2$, we obtain the M-step as

$$\begin{aligned} (\alpha^{\text{new}},\theta^{\text{new}},\mu^{\text{new}}) &= \underset{\alpha,\mathbb{1}_n^T\theta=0,\mu}{\operatorname{argmin}}\; F(\alpha,\theta,\mu;q^{\text{new}}), \qquad (41) \\ (\sigma^2)^{\text{new}} &= \frac{F(\alpha^{\text{new}},\theta^{\text{new}},\mu^{\text{new}};q^{\text{new}})+b}{(2n+k)d+a+2}, \end{aligned}$$

where

$$F(\alpha,\theta,\mu;q)=\|y-\mathbb{1}_n\alpha^T-\theta\|_{\mathrm{F}}^2+\nu\|\alpha\|^2+\sum_{i=1}^n\sum_{j=1}^k\left(\frac{q_{ij}}{v_0}+\frac{1-q_{ij}}{v_1}\right)\|\theta_i-\mu_j\|^2.$$

Note that for all θ such that $\mathbb{1}_n^T\theta=0$, we have

$$\|y-\mathbb{1}_n\alpha^T-\theta\|_{\mathrm{F}}^2=\|n^{-1}\mathbb{1}_n\mathbb{1}_n^Ty-\mathbb{1}_n\alpha^T\|_{\mathrm{F}}^2+\|y-n^{-1}\mathbb{1}_n\mathbb{1}_n^Ty-\theta\|_{\mathrm{F}}^2,$$

and thus the M-step (41) can be solved separately for α and (θ,μ).

4.4. A Connection to Bipartite Graph Projection

The clustering model (33) involves latent variables μ that do not appear in the likelihood (32). This allows us to derive an efficient EM algorithm in Section 4.3. To better understand (33), we connect the bipartite graphical structure between θ and μ to a graphical structure on θ alone. Given a $\gamma=\{\gamma_{ij}\}$ that satisfies (34), we call it non-degenerate if $\sum_{i=1}^n\gamma_{ij}>0$ for all $j\in[k]$. In other words, none of the k clusters is empty.

Proposition 4 *Let the conditional distribution of $\theta,\mu\,|\,\gamma,\sigma^2$ be specified by (33) with some non-degenerate γ. Then, the distribution of $\theta\,|\,\gamma,\sigma^2$ weakly converges to*

$$p(\theta\,|\,\gamma,\sigma^2)\propto\prod_{1\le i<l\le n}\exp\left(-\frac{\lambda_{il}\|\theta_i-\theta_l\|^2}{2\sigma^2v_0}\right)\mathbb{I}\{\mathbb{1}_n^T\theta=0\},\tag{42}$$

as $v_1\to\infty$, where $\lambda_{il}=\sum_{j=1}^k\gamma_{ij}\gamma_{lj}/n_j$ with $n_j=\sum_{i=1}^n\gamma_{ij}$ being the size of the jth cluster.

The formula (42) resembles (3), except that λ encodes a clustering structure. By the definition of λ_{il}, if θ_i and θ_l are in different clusters, $\lambda_{il}=0$, and otherwise, λ_{il} takes the inverse of the size of the cluster that both θ_i and θ_l belong to. The relation between (33) and (42) can be understood from the operations of graph projection and graph lift, in the sense that the weighted graph λ, with nodes θ, is a projection of γ, a bipartite graph between nodes θ and nodes μ. Conversely, γ is said to be a graph lift of λ. Observe that the clustering structure of (42) is combinatorial, and therefore it is much easier to work with the bipartite structure in (33) with latent variables.

4.5. A Connection to Gaussian Mixture Models

We establish a connection to Gaussian mixture models. We first give the following result.

Proposition 5 *Let the conditional distribution of $\theta,\mu\,|\,\gamma,\sigma^2$ be specified by (33). Then, as $v_0\to 0$, this conditional distribution weakly converges to the distribution specified by the following sampling process: $\theta=\gamma\mu$ and $\mu\,|\,\gamma,\sigma^2$ is sampled from*

$$p(\mu\,|\,\gamma,\sigma^2)\propto\prod_{1\le j<l\le k}\exp\left(-\frac{(n_j+n_l)\|\mu_j-\mu_l\|^2}{2\sigma^2v_1}\right)\mathbb{I}\{\mathbb{1}_n^T\gamma\mu=0\},\tag{43}$$

where $n_j=\sum_{i=1}^n\gamma_{ij}$ being the size of the jth cluster.

With this proposition, we can see that as $v_0 \to 0$, the clustering model specified by (32), (30) and (33) becomes

$$y \,|\, \alpha, \mu, \gamma, \sigma^2 \sim N(\mathbb{1}_n \alpha^T + \gamma\mu, \sigma^2 I_n \otimes I_d), \tag{44}$$

with $\alpha \,|\, \sigma^2$ distributed by (30) and $\mu \,|\, \gamma, \sigma^2$ distributed by (43). The likelihood function (44) is commonly used in Gaussian mixture models, which encodes an exact clustering structure. Therefore, with a finite v_0, the model specified by (32), (30) and (33) can be interpreted as a relaxed version of the Gaussian mixture models that leads to an approximate clustering structure.

4.6. Adaptation to the Number of Clusters

The number k in (33) should be understood as an upper bound of the number of clusters. Even though the EM algorithm outputs k cluster centers $\{\widehat{\mu}_1, ..., \widehat{\mu}_k\}$, these k cluster centers will be automatically grouped according to their own closedness as we vary the value of v_0. Generally speaking, for a very small v_0 (the Gaussian mixture model in Section 4.5, for example), $\{\widehat{\mu}_1, ..., \widehat{\mu}_k\}$ will take k vectors that are not close to each other. As we increase v_0, the clustering centers $\{\widehat{\mu}_1, ..., \widehat{\mu}_k\}$ start to merge, and eventually for a sufficiently large v_0, they will all converge to a single vector. In short, v_0 parametrizes the solution path of our clustering algorithm, and on this solution path, the *effective* number of clusters increases as the value of v_0 increases.

We illustrate this point by a simple numerical example. Consider the observation $y = (4, 2, -2, 4)^T \in \mathbb{R}^{4\times 1}$. We fit our clustering model with $k \in \{2, 3, 4\}$. Figure 1 visualizes the output of the EM algorithm $(\widehat{\mu}, \widehat{\theta})$ as v_0 varies. It is clear that the solution path always starts at $\{\widehat{\mu}_1, ..., \widehat{\mu}_k\}$ of different values. Then, as v_0 increases, the solution path has various phase transitions where the closest two $\widehat{\mu}_j$'s merge. In the end, for a sufficiently large v_0, the clustering centers $\{\widehat{\mu}_1, ..., \widehat{\mu}_k\}$ all merge to a common value.

To explain this phenomenon, it is most clear to investigate the case $k = 2$. Then, the M-step (41) updates μ according to

$$\begin{aligned} \mu_1^{\text{new}} &= \underset{\mu_1}{\text{argmin}} \sum_{i=1}^n \left(\frac{q_{i1}}{v_0} + \frac{1 - q_{i1}}{v_1} \right) \|\theta_i - \mu_1\|^2 \\ \mu_1^{\text{new}} &= \underset{\mu_2}{\text{argmin}} \sum_{i=1}^n \left(\frac{q_{i2}}{v_0} + \frac{1 - q_{i2}}{v_1} \right) \|\theta_i - \mu_2\|^2. \end{aligned}$$

Observe that both μ_1^{new} and μ_1^{new} are weighted averages of $\{\theta_1, ..., \theta_n\}$, and the only difference between μ_1^{new} and μ_1^{new} lies in the weights. According to the E-step (40),

$$\begin{aligned} q_{i1} &= \frac{\exp\left(-\frac{\|\theta_i - \mu_1\|^2}{2\sigma^2 \bar{v}}\right)}{\exp\left(-\frac{\|\theta_i - \mu_1\|^2}{2\sigma^2 \bar{v}}\right) + \exp\left(-\frac{\|\theta_i - \mu_2\|^2}{2\sigma^2 \bar{v}}\right)} \\ q_{i2} &= \frac{\exp\left(-\frac{\|\theta_i - \mu_2\|^2}{2\sigma^2 \bar{v}}\right)}{\exp\left(-\frac{\|\theta_i - \mu_1\|^2}{2\sigma^2 \bar{v}}\right) + \exp\left(-\frac{\|\theta_i - \mu_2\|^2}{2\sigma^2 \bar{v}}\right)}, \end{aligned}$$

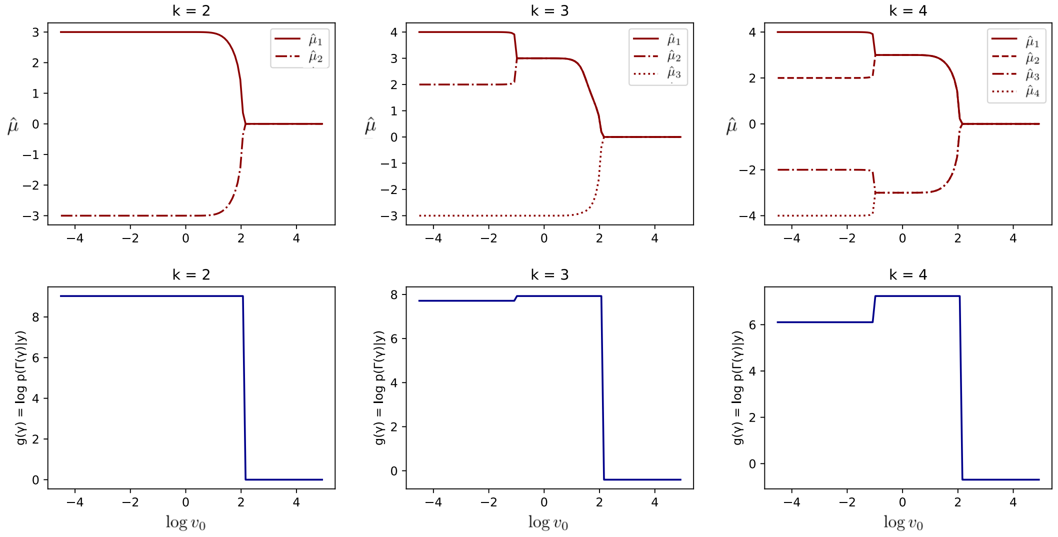


Figure 1: (Top) Solution paths of $\widehat{\mu}$ with different choices of k; (Bottom) Model selection scores on the solution paths.

and we recall that $\bar{v}^{-1} = v_0^{-1} - v_1^{-1}$. Therefore, as $v_0 \to \infty$, $q_{i1} \to 1/2$ and $q_{i2} \to 1/2$, which results in the phenomenon that μ_1^{new} and μ_2^{new} merge to the same value. The same reasoning also applies to $k \geq 2$.

4.7. Model Selection

In this section, we discuss how to select a clustering structure from the solution path of the EM algorithm. According to the discussion in Section 4.6, we should understand k as an upper bound of the number of clusters, and the estimator of the number of clusters will be part of the output of the model selection procedure. The general recipe of our method follows the framework discussed in Section 3.3, but some nontrivial twist is required in the clustering problem. To make the presentation clear, the model selection procedure will be introduced in two parts. We will first propose our model selection score, and then we will describe a method that extracts a clustering structure from the output of the EM algorithm.

4.7.1. The model selection score

For any $\gamma \in \{0,1\}^{n \times k}$ that satisfies (34), we can calculate the posterior probability $p(\gamma \,|\, y)$ with $v_0 = 0$. This can be done by the connection to Gaussian mixture models discussed in Section 4.5. To be specific, the calculation follows the formula

$$p(\gamma \,|\, y) = \int\int\int p(y \,|\, \alpha, \mu, \gamma, \sigma^2) p(\alpha \,|\, \sigma^2) p(\mu \,|\, \gamma, \sigma^2) p(\sigma^2) p(\gamma) \, d\alpha \, d\mu \, d\sigma^2,$$

where $p(y\,|\,\alpha,\mu,\gamma,\sigma^2)$, $p(\alpha|\sigma^2)$, $p(\mu|\gamma,\sigma^2)$, $p(\sigma^2)$, and $p(\gamma)$ are specified by (32), (30), (43), (6) and (35). A standard calculation gives the formula

$$\begin{aligned} p(\gamma|y) \propto \left(\frac{\nu}{\nu+n}\times\frac{\det_{\gamma^T\mathbb{1}_n}(\bar{L}_\gamma)}{\det_{\gamma^T\mathbb{1}_n}(\bar{L}_\gamma+\gamma^T\gamma)}\right)^{d/2} &\times \Bigg[\frac{\nu n}{\nu+n}\|n^{-1}\mathbb{1}_n y\|^2 \\ &+ \mathsf{Tr}\left((y-n^{-1}\mathbb{1}_n\mathbb{1}_n^T y)^T(I_n-\gamma(\bar{L}_\gamma+\gamma^T\gamma)\gamma^T)(y-n^{-1}\mathbb{1}_n\mathbb{1}_n^T y)\right)\Bigg]^{-\frac{nd+a}{2}}, \end{aligned} \tag{45}$$

where $\bar{L}_\gamma = (\mathbb{1}_{k\times n}\gamma + \gamma^T\mathbb{1}_{n\times k} - 2\gamma^T\gamma)/v_1$ is the graph Laplacian of the weighted adjacency matrix, which satisfies

$$\mathsf{Tr}(\mu^T\bar{L}_\gamma\mu) = \sum_{1\leq j<l\leq k}\frac{n_j+n_l}{v_1}\|\mu_j-\mu_l\|^2.$$

Recall that $n_j = \sum_{i=1}^n \gamma_{ij}$ is the size of the jth cluster.

However, the goal of the model selection is to select a clustering structure, and it is possible that different γ's may correspond to the same clustering structure due to label permutation. To overcome this issue, we need to sum over all equivalent γ's. Given a $\gamma \in \{0,1\}^{n\times k}$ that satisfies (34), define a symmetric membership matrix $\Gamma(\gamma) \in \{0,1\}^{n\times n}$ by

$$\Gamma_{il}(\gamma) = \mathbb{I}\left\{\sum_{j=1}^k \gamma_{ij}\gamma_{lj} = 1\right\}.$$

In other words, $\Gamma_{il}(\gamma) = 1$ if and only if i and l are in the same cluster. It is easy to see that every clustering structure can be uniquely represented by a symmetric membership matrix. We define the posterior probability of a clustering structure Γ by

$$p(\Gamma\,|\,y) = \sum_{\gamma\in\{\gamma:\Gamma(\gamma)=\Gamma\}} p(\gamma\,|\,y).$$

The explicit calculation of the above summation is not necessary. A shortcut can be derived from the fact that $p(\gamma\,|\,y) = p(\gamma'\,|\,y)$ if $\Gamma(\gamma) = \Gamma(\gamma')$. This immediately implies that $p(\Gamma\,|\,y) = |\Gamma|\cdot p(\gamma\,|\,y)$ for any γ that satisfies $\Gamma(\gamma) = \Gamma$. Suppose Γ encodes a clustering structure with $\widetilde{k}$ nonempty clusters, and then we have $|\Gamma| = \binom{k}{\widetilde{k}}\widetilde{k}!$. This leads to the model selection score

$$g(\gamma) = \log p(\Gamma(\gamma)\,|\,y) = \log p(\gamma\,|\,y) + \log\left[\binom{k}{\widetilde{k}}\widetilde{k}!\right], \tag{46}$$

for any $\gamma \in \{0,1\}^{n\times k}$ that satisfies (34), and $\widetilde{k}$ above is calculated by $\widetilde{k} = \sum_{j=1}^k \max_{1\leq i\leq n}\gamma_{ij}$, the effective number of clusters.

4.7.2. Extraction of clustering structures from the EM algorithm

Let $\widehat{\mu}$ and $\widehat{q}$ be outputs of the EM algorithm, and we discuss how to obtain $\widehat{\gamma}$ that encodes a meaningful clustering structure to be evaluated by the model selection score (46). It is very tempting to directly threshold $\widehat{q}$ as is done in Section 3.3. However, as has been discussed in Section 4.6, the solution paths of $\{\mu_1, ..., \mu_k\}$ merge at some values of v_0. Therefore, we should treat the clusters whose clustering centers merge together as a single cluster.

Given $\widehat{\mu}_1, ..., \widehat{\mu}_k$ output by the M-step, we first merge $\widehat{\mu}_j$ and $\widehat{\mu}_l$ whenever $\|\widehat{\mu}_j - \widehat{\mu}_l\| \leq \epsilon$. The number ϵ is taken as 10^{-8}, the square root of the machine precision, in our code. This forms a partition $[k] = \cup_{l=1}^{\widehat{k}} \mathcal{G}_l$ for some $\widehat{k} \leq k$. Then, by taking average within each group, we obtain a reduced collection of clustering centers $\widetilde{\mu}_1, ..., \widetilde{\mu}_{\widehat{k}}$. In other words, $\widetilde{\mu}_l = |\mathcal{G}_l|^{-1} \sum_{j \in \mathcal{G}_l} \widehat{\mu}_j$.

The $\widehat{q} \in [0,1]^{n \times k}$ output by the E-step should also be reduced to $\widetilde{q} \in [0,1]^{n \times \widehat{k}}$ as well. Note that $\widehat{q}_{ij}$ is the estimated posterior probability that the ith node belongs to the jth cluster. This means that $\widetilde{q}_{il}$ is the estimated posterior probability that the ith node belongs to the lth reduced cluster. An explicit formua is given by $\widetilde{q}_{il} = \sum_{j \in \mathcal{G}_l} \widehat{q}_{ij}$.

With the reduced posterior probability $\widetilde{q}$, we simply apply thresholding to obtain $\widehat{\gamma}$. We have $(\widehat{\gamma}_{i1}, ..., \widehat{\gamma}_{ik}) = e_j$ if $j = \text{argmax}_{1 \leq l \leq \widehat{k}} \widetilde{q}_{il}$. Recall that e_j is a vector with 1 on the jth entry and 0 elsewhere. Note that according to this construction, we always have $\widehat{\gamma}_{ij} = 0$ whenever $j > \widehat{k}$. This does not matter, because the model selection score (46) does not depend on the clustering labels. Finally, the $\widehat{\gamma}$ constructed according to the above procedure will be evaluated by $g(\widehat{\gamma})$ defined by (46).

In the toy example with four data points $y = (4, 2, -2, 4)^T$, the model selection score is computed along the solution path. According to Figure 1, the model selection procedure suggests that a clustering structure with two clusters $\{4, 2\}$ and $\{-2, -4\}$ is the most plausible one. We also note that the curve of $g(\gamma)$ has sharp phase transitions whenever the solution paths μ merge.

5. Extensions with Graph Algebra

In many applications, it is useful to have a model that imposes both row and column structures on a high-dimensional matrix $\theta \in \mathbb{R}^{p_1 \times p_2}$. We list some important examples below.

1. *Biclustering.* In applications such as gene expression data analysis, one needs to cluster both samples and features. This task imposes a clustering structure for both rows and columns of the data matrix (Hartigan, 1972; Cheng and Church, 2000).

2. *Block sparsity.* In problems such as planted clique detection (Feige and Krauthgamer, 2000) and submatrix localization (Hajek et al., 2017), the matrix can be viewed as the sum of a noise background plus a submatrix of signals with unknown locations. Equivalently, it can be modeled by simultaneous row and column sparsity (Ma and Wu, 2015).

3. *Sparse clustering.* Suppose the data matrix exhibits a clustering structure for its rows and a sparsity structure for its columns, then we have a sparse clustering problem (Witten and Tibshirani, 2010). For this task, we need to select nonzero column features in order to accurately cluster the rows.

For the problems listed above, the row and column structures can be modeled by graphs γ_1 and γ_2. Then, the structure of the matrix θ is induced by a notion of graph product of γ_1

and γ_2. In this section, we introduce tools from graph algebra including Cartesian product and Kronecker product to build complex structure from simple components.

We first introduce the likelihood of the problem. To cope with many useful models, we assume that the observation can be organized as a matrix $y \in \mathbb{R}^{n_1 \times n_2}$. Then, the specific setting of a certain problem can be encoded by design matrices $X_1 \in \mathbb{R}^{n_1 \times p_1}$ and $X_2 \in \mathbb{R}^{n_2 \times p_2}$. The likelihood is defined by

$$y \,|\, \alpha, \theta, \sigma^2 \sim N(X_1(\alpha w + \theta)X_2^T, \sigma^2 I_{n_1} \otimes I_{n_2}). \tag{47}$$

The matrix $w \in \mathbb{R}^{p_1 \times p_2}$ is assumed to have rank one, and can be decomposed as $w = w_1 w_2^T$ for some $w_1 \in \mathbb{R}^{p_1}$ and $w_2 \in \mathbb{R}^{p_2}$. The prior distribution of the scalar is simply given by

$$\alpha \,|\, \sigma^2 \sim N(0, \sigma^2/\nu). \tag{48}$$

We then need to build prior distributions of θ that is supported on $\Theta_w = \{\theta \in \mathbb{R}^{p_1 \times p_2} : \mathsf{Tr}(w\theta^T) = 0\}$ using Cartesian and Kronecker products.

5.1. Cartesian Product

We start with the definition of the Cartesian product of two graphs.

Definition 6 *Given two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, their Cartesian product $G = G_1 \,\square\, G_2$ is defined with the vertex set $V_1 \times V_2$. Its edge set contains $((x_1, x_2), (y_1, y_2))$ if and only if $x_1 = y_1$ and $(x_2, y_2) \in E_2$ or $(x_1, y_1 suno) \in E_1$ and $x_2 = y_2$.*

According to the definition, it can be checked that for two graphs of sizes p_1 and p_2, the adjacency matrix, the Laplacian and the incidence matrix of the Cartesian product enjoy the relations

$$\begin{aligned} A_{1\,\square\,2} &= A_2 \otimes I_{p_1} + I_{p_2} \otimes A_1, \\ L_{1\,\square\,2} &= L_2 \otimes I_{p_1} + I_{p_2} \otimes L_1, \\ D_{1\,\square\,2} &= [D_2 \otimes I_{p_1}; I_{p_2} \otimes D_1]. \end{aligned}$$

Given graphs γ_1 and γ_2 that encode row and column structures of θ, we introduce the following prior distribution

$$\begin{aligned} p(\theta \,|\, \gamma_1, \gamma_2, \sigma^2) \;\propto\; & \prod_{(i,j) \in E_1} \exp\left(-\frac{\|\theta_{i*} - \theta_{j*}\|^2}{2\sigma^2[v_0\gamma_{1,ij} + v_1(1-\gamma_{1,ij})]}\right) \\ & \times \prod_{(k,l) \in E_2} \exp\left(-\frac{\|\theta_{*k} - \theta_{*l}\|^2}{2\sigma^2[v_0\gamma_{2,kl} + v_1(1-\gamma_{2,kl})]}\right) \mathbb{I}\{\theta \in \Theta_w\}. \end{aligned} \tag{49}$$

Here, E_1 and E_2 are the base graphs of the row and column structures. According to its form, the prior distribution (49) models both pairwise relations of rows and those of columns based on γ_1 and γ_2, respectively. To better understand (49), we can write it in the following equivalent form,

$$p(\theta \,|\, \gamma_1, \gamma_2, \sigma^2) \propto \exp\left(-\frac{1}{2\sigma^2} \mathsf{vec}(\theta)^T \left(L_{\gamma_2} \otimes I_{p_1} + I_{p_2} \otimes L_{\gamma_1}\right) \mathsf{vec}(\theta)\right) \mathbb{I}\{\theta \in \Theta_w\}, \tag{50}$$

where $L_{\gamma_1} \in \mathbb{R}^{p_1 \times p_1}$ and $L_{\gamma_2} \in \mathbb{R}^{p_2 \times p_2}$ are Laplacian matrices of the weighted graphs $\{v_0\gamma_{1,ij} + v_1(1-\gamma_{1,ij})\}$ and $\{v_0\gamma_{2,kl} + v_1(1-\gamma_{2,kl})\}$, respectively. Therefore, by Definition 6, $p(\theta \,|\, \gamma_1, \gamma_2, \sigma^2)$ is a spike-and-slab Laplacian prior $p(\theta \,|\, \gamma, \sigma^2)$ defined in (3) with $\gamma = \gamma_1 \,\square\, \gamma_2$, and the well-definedness is guaranteed by Proposition 1.

To complete the Bayesian model, the distribution of $(\gamma_1, \gamma_2, \sigma^2)$ are specified by

$$\begin{aligned} \gamma_1, \gamma_2 \,|\, \eta_1, \eta_2 &\sim \prod_{(i,j)\in E_1} \eta_1^{\gamma_{1,ij}}(1-\eta_1)^{1-\gamma_{1,ij}} \prod_{(i,j)\in E_2} \eta_2^{\gamma_{2,kl}}(1-\eta_2)^{1-\gamma_{2,kl}}, &(51)\\ \eta_1, \eta_2 &\sim \text{Beta}(A_1, B_1) \bigotimes \text{Beta}(A_2, B_2), &(52)\\ \sigma^2 &\sim \text{InvGamma}(a/2, b/2). &(53) \end{aligned}$$

We remark that it is possible to constrain γ_1 and γ_2 in some subsets Γ_1 and Γ_2 like (4). This extra twist is useful for a biclustering model that will be discussed in Section 5.3.

Note that in general the base graph $G = G_1 \,\square\, G_2$ is not a tree, and the derivation of the EM algorithm follows a similar argument in Section 3.2. Using the same argument in (19), we lower bound $\log \sum_{T\in\text{spt}(G)} \sum_{e\in T}[v_0^{-1}\gamma_e + v_1^{-1}(1-\gamma_e)]$ by

$$\sum_{e\in E_1 \,\square\, E_2} r_e \log[v_0^{-1}\gamma_e + v_1^{-1}(1-\gamma_e)] + \log|\text{spt}(G)|. \tag{54}$$

Since

$$E_1 \,\square\, E_2 = \{((i,k),(j,k)) : (i,j) \in E_1, k \in V_2\} \cup \{((i,k),(i,l)) : i \in V_1, (k,l) \in E_2\},$$

and $\gamma = \gamma_1 \,\square\, \gamma_2$, we can write (54) as

$$\begin{aligned} &\sum_{(i,j)\in E_1} \sum_{k=1}^{p_2} r_{(i,k),(j,k)} \log[v_0^{-1}\gamma_{1,ij} + v_1^{-1}(1-\gamma_{1,ij})]\\ &+ \sum_{(k,l)\in E_2} \sum_{i=1}^{p_1} r_{(i,k),(i,l)} \log[v_0^{-1}\gamma_{2,kl} + v_1^{-1}(1-\gamma_{2,kl})]\\ = &\sum_{(i,j)\in E_1} r_{1,ij} \log[v_0^{-1}\gamma_{1,ij} + v_1^{-1}(1-\gamma_{1,ij})] + \sum_{(k,l)\in E_2} r_{2,kl} \log[v_0^{-1}\gamma_{2,kl} + v_1^{-1}(1-\gamma_{2,kl})], \end{aligned}$$

where

$$r_{1,ij} = \sum_{k=1}^{p_2} r_{(i,k),(j,k)} = \frac{1}{|\text{spt}(G)|} \sum_{k=1}^{p_2} \sum_{T\in\text{spt}(G)} \mathbb{I}\{((i,k),(j,k)) \in T\},$$

and $r_{2,kl}$ is similarly defined.

Using the lower bound derived above, it is direct to derive the an EM algorithm, which consists of the following iterations,

$$\begin{align}
q_{1,ij}^{\text{new}} &= \frac{\eta_1 v_0^{-r_{1,ij}/2} e^{-\|\theta_{i*}-\theta_{j*}\|^2/2\sigma^2 v_0}}{\eta_1 v_0^{-r_{1,ij}/2} e^{-\|\theta_{i*}-\theta_{j*}\|^2/2\sigma^2 v_0} + (1-\eta_1) v_1^{-r_{1,ij}/2} e^{-\|\theta_{i*}-\theta_{j*}\|^2/2\sigma^2 v_1}}, \tag{55}\\
q_{2,kl}^{\text{new}} &= \frac{\eta_2 v_0^{-r_{2,kl}/2} e^{-\|\theta_{*k}-\theta_{*l}\|^2/2\sigma^2 v_0}}{\eta_2 v_0^{-r_{2,kl}/2} e^{-\|\theta_{*k}-\theta_{*l}\|^2/2\sigma^2 v_0} + (1-\eta_2) v_1^{-r_{2,kl}/2} e^{-\|\theta_{*k}-\theta_{*l}\|^2/2\sigma^2 v_1}}, \tag{56}\\
(\alpha^{\text{new}}, \theta^{\text{new}}) &= \underset{\alpha,\theta\in\Theta_w}{\operatorname{argmin}} F(\alpha, \theta; q_1^{\text{new}}, q_2^{\text{new}}), \tag{57}\\
(\sigma^2)^{\text{new}} &= \frac{F(\alpha^{\text{new}}, \theta^{\text{new}}; q_1^{\text{new}}, q_2^{\text{new}}) + b}{n_1 n_2 + p_1 p_2 + a + 2}, \notag\\
\eta_1^{\text{new}} &= \frac{A_1 - 1 + \sum_{(i,j)\in E_1} q_{1,ij}^{\text{new}}}{A_1 + B_1 - 2 + m_1}, \notag\\
\eta_2^{\text{new}} &= \frac{A_2 - 1 + \sum_{(k,l)\in E_2} q_{2,ij}^{\text{new}}}{A_2 + B_2 - 2 + m_2}. \tag{58}
\end{align}$$

The definition of the function $F(\alpha, \theta; q_1, q_2)$ is given by

$$F(\alpha, \theta; q_1, q_2) = \|y - X_1(\alpha w + \theta)X_2^T\|_{\text{F}}^2 + \nu\alpha^2 + \mathsf{vec}(\theta)^T \left(L_{q_2} \otimes I_{p_1} + I_{p_2} \otimes L_{q_1}\right) \mathsf{vec}(\theta)$$

Though the E-steps (55) and (56) are straightforward, the M-step (57) is a quadratic programming of dimension $p_1 p_2$, which may become the computational bottleneck of the EM algorithm when the size of the problem is large. We will introduce a Dykstra-like algorithm to solve (57) in Appendix E.

The Cartesian product model is useful for simultaneous learning the row structure γ_1 and the column structure and γ_2 of the coefficient matrix θ. Note that when $X_1 = X$, $X_2 = I_d$, and $E_2 = \varnothing$, the Cartesian product model becomes the multivariate regression model described in Section 4.1. In this case, the model only regularizes the row structure of θ. Another equally interesting example is obtained when $X_1 = X$, $X_2 = I_d$, and $E_1 = \varnothing$. In this case, the model only regularizes the column structure of θ, and can be interpreted as multitask learning with task clustering.

5.2. Kronecker Product

The Kronecker product of two graphs is defined below.

Definition 7 *Given two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, their Kronecker product $G = G_1 \otimes G_2$ is defined with the vertex set $V_1 \times V_2$. Its edge set contains $((x_1, x_2), (y_1, y_2))$ if and only if $(x_1, y_1) \in E_1$ and $(x_2, y_2) \in E_2$.*

It is not hard to see that the adjacency matrix of two graphs has the formula $A_{1\otimes 2} = A_1 \otimes A_2$, which gives the name of Definition 7. The prior distribution of θ given row and column graphs γ_1 and γ_2 that we discuss in this subsection is

$$p(\theta \,|\, \gamma_1, \gamma_2, \sigma^2) \propto \prod_{(i,j)\in E_1} \prod_{(k,l)\in E_2} \exp\left(-\frac{(\theta_{ik} - \theta_{jl})^2}{2\sigma^2[v_0\gamma_{1,ij}\gamma_{2,kl} + v_1(1 - \gamma_{1,ij}\gamma_{2,kl})]}\right) \mathbb{I}\{\theta \in \Theta_w\}. \tag{59}$$

Again, E_1 and E_2 are the base graphs of the row and column structures. According to its form, the prior imposes a nearly block structure on θ based on the graphs γ_1 and γ_2. Moreover, $p(\theta \,|\, \gamma_1, \gamma_2, \sigma^2)$ can be viewed as a spike-and-slab Laplacian prior $p(\theta \,|\, \gamma, \sigma^2)$ defined in (3) with $\gamma = \gamma_1 \otimes \gamma_2$. The distribution of $(\gamma_1, \gamma_2, \sigma^2)$ follows the same specification in (51)-(53).

To derive an EM algorithm, we follow the strategy in Section 3.2 and lower bound $\log \sum_{T \in \mathrm{spt}(G)} \sum_{e \in T} [v_0^{-1} \gamma_e + v_1^{-1}(1 - \gamma_e)]$ by

$$\sum_{(i,j) \in E_1} \sum_{(k,l) \in E_2} r_{(i,k),(j,l)} \log[v_0^{-1} \gamma_{1,ij} \gamma_{2,kl} + v_1^{-1}(1 - \gamma_{1,ij} \gamma_{2,kl})]. \tag{60}$$

Unlike the Cartesian product, the Kronecker product structure has a lower bound (60) that is not separable with respect to γ_1 and γ_2. This makes the E-step combinatorial, and does not apply to a large-scale problem. To alleviate this computational barrier, we consider a variational EM algorithm that finds the best posterior distribution of γ_1, γ_2 that can be factorized. In other words, instead of maximizing over all possible distribution q, we maximize over the mean-filed class $q \in \mathcal{Q}$, with $\mathcal{Q} = \{q(\gamma_1, \gamma_2) = q_1(\gamma_1) q_2(\gamma_2) : q_1, q_2\}$. Then, the objective becomes

$$\max_{q_1, q_2} \max_{\alpha, \theta \in \Theta_2, \delta, \eta, \sigma^2} \sum_{\gamma_1, \gamma_2} q_1(\gamma_1) q_2(\gamma_2) \log \frac{\widetilde{p}(y, \alpha, \theta, \delta, \gamma_1, \gamma_2, \eta, \sigma^2)}{q_1(\gamma_1) q_2(\gamma_2)},$$

where $\widetilde{p}(y, \alpha, \theta, \delta, \gamma_1, \gamma_2, \eta, \sigma^2)$ is obtained by replacing $p(\theta \,|\, \gamma_1, \gamma_2, \sigma^2)$ with $\widetilde{p}(\theta \,|\, \gamma_1, \gamma_2, \sigma^2)$ in the joint distribution $p(y, \alpha, \theta, \delta, \gamma_1, \gamma_2, \eta, \sigma^2)$. Here, $\log \widetilde{p}(\theta \,|\, \gamma_1, \gamma_2, \sigma^2)$ is a lower bound for $\log p(\theta \,|\, \gamma_1, \gamma_2, \sigma^2)$ with (60). The E-step of the variational EM is

$$q_1^{\mathrm{new}}(\gamma_1) \propto \exp\left(\sum_{\gamma_2} q_2(\gamma_2) \log \widetilde{p}(y, \alpha, \theta, \delta, \gamma_1, \gamma_2, \eta, \sigma^2) \right),$$

$$q_2^{\mathrm{new}}(\gamma_2) \propto \exp\left(\sum_{\gamma_1} q_1^{\mathrm{new}}(\gamma_1) \log \widetilde{p}(y, \alpha, \theta, \delta, \gamma_1, \gamma_2, \eta, \sigma^2) \right).$$

After some simplification, we have

$$q_{1,ij}^{\mathrm{new}} = \left(1 + \frac{(1 - \eta_1) \prod_{(k,l) \in E_2} \left(v_1^{-r_{(i,k),(j,l)}/2} e^{-(\theta_{ik} - \theta_{jl})^2 / 2\sigma^2 v_1} \right)^{q_{2,kl}}}{\eta_1 \prod_{(k,l) \in E_2} \left(v_0^{-r_{(i,k),(j,l)}/2} e^{-(\theta_{ik} - \theta_{jl})^2 / 2\sigma^2 v_0} \right)^{q_{2,kl}}} \right)^{-1}, \tag{61}$$

$$q_{2,kl}^{\mathrm{new}} = \left(1 + \frac{(1 - \eta_2) \prod_{(i,j) \in E_1} \left(v_1^{-r_{(i,k),(j,l)}/2} e^{-(\theta_{ik} - \theta_{jl})^2 / 2\sigma^2 v_1} \right)^{q_{1,kl}^{\mathrm{new}}}}{\eta_2 \prod_{(i,j) \in E_1} \left(v_0^{-r_{(i,k),(j,l)}/2} e^{-(\theta_{ik} - \theta_{jl})^2 / 2\sigma^2 v_0} \right)^{q_{1,kl}^{\mathrm{new}}}} \right)^{-1}. \tag{62}$$

The M-step can be derived in a standard way, and it has the same updates as in (57)-(58), with a new definition of $F(\alpha, \theta; q_1, q_2)$ given by

$$\begin{aligned} F(\alpha, \theta; q_1, q_2) &= \|y - X_1(\alpha w + \theta) X_2\|^2 + \nu \alpha^2 \\ &\quad + \sum_{(i,j) \in E_1} \sum_{(k,l) \in E_2} \left(\frac{q_{1,ij} q_{2,kl}}{v_0} + \frac{1 - q_{1,ij} q_{2,kl}}{v_1} \right) (\theta_{ik} - \theta_{jl})^2. \end{aligned}$$

5.3. Applications in Biclustering

When both row and column graphs encode clustering structures discussed in Section 4.2, we have the biclustering model. In this section, we discuss both biclustering models induced by Kronecker and Cartesian products. We start with a special form of the likelihood (47), which is given by

$$y\,|\,\alpha,\theta,\sigma^2 \sim N(\alpha\mathbb{1}_{n_1}\mathbb{1}_{n_2}^T+\theta,\sigma^2 I_{n_1}\otimes I_{n_2}),$$

and the prior distribution on α is given by (48). The prior distribution on θ will be discussed in two cases.

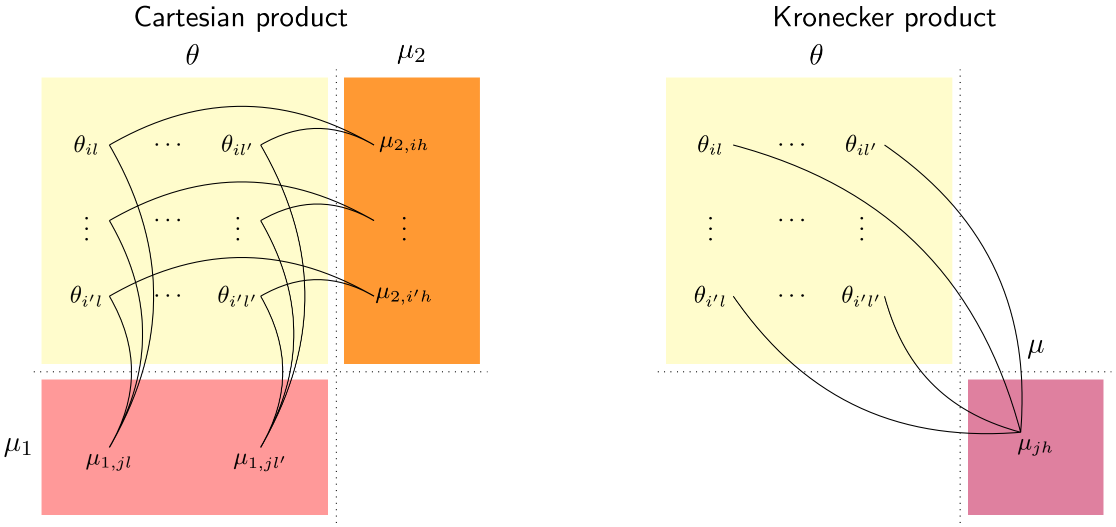


Figure 2: Structure diagrams for the two biclustering methods. The Cartesian product biclustering model (Left) and the Kronecker product biclustering model (Right) have different latent variables and base graphs. While the Cartesian product models the row and column clustering structures by separate latent variable matrices $\mu_1 \in \mathbb{R}^{k_1\times n_2}$ and $\mu_2 \in \mathbb{R}^{n_1\times k_2}$, the Kronecker product directly models the checkerboard structure by a single latent matrix $\mu \in \mathbb{R}^{k_1\times k_2}$.

5.3.1. Cartesian product biclustering model

Let $k_1 \in [n_1]$ and $k_2 \in [n_2]$ be upper bounds of the numbers of row and column clusters, respectively. We introduce two latent matrices $\mu_1 \in \mathbb{R}^{k_1\times n_2}$ and $\mu_2 \in \mathbb{R}^{n_1\times k_2}$ that serve as row and column clustering centers. The prior distribution is then specified by

$$\begin{aligned}p(\theta,\mu_1,\mu_2\,|\,\gamma_1,\gamma_2,\sigma^2) \propto& \prod_{i=1}^{n_1}\prod_{j=1}^{k_1}\exp\left(-\frac{\|\theta_{i*}-\mu_{1,j*}\|^2}{2\sigma^2[v_0\gamma_{1,ij}+v_1(1-\gamma_{1,ij})]}\right)\\ &\times\prod_{l=1}^{n_2}\prod_{h=1}^{k_2}\exp\left(-\frac{\|\theta_{*l}-\mu_{2,*h}\|^2}{2\sigma^2[v_0\gamma_{2,lh}+v_1(1-\gamma_{2,lh})]}\right)\mathbb{I}\{\mathbb{1}_{n_1}^T\theta\mathbb{1}_{n_2}=0\},\end{aligned}$$

which can be regarded as an extension of (33) in the form of (49). The prior distributions on γ_1 and γ_2 are independently specified by (35) with (k,n) replaced by (k_1,n_1) and (k_2,n_2). Finally, σ^2 follows the inverse Gamma prior (6).

We follow the framework of Section 3.2. The derivation of the EM algorithm requires lower bounding $\log\sum_{T\in\text{spt}(G)}\sum_{e\in T}[v_0^{-1}\gamma_e+v_1^{-1}(1-\gamma_e)]$. Using the same argument in Section 5.1, we have the following lower bound

$$\sum_{i=1}^{n_1}\sum_{j=1}^{k_1}r_{1,ij}\log[v_0^{-1}\gamma_{1,ij}+v_1^{-1}(1-\gamma_{1,ij})]+\sum_{l=1}^{n_2}\sum_{h=1}^{k_2}r_{2,lh}\log[v_0^{-1}\gamma_{2,lh}+v_1^{-1}(1-\gamma_{2,lh})].\quad(63)$$

By the symmetry of the complete bipartite graph, $r_{1,ij}$ is a constant that does not depend on (i,j). Then use the same argument in (37)-(38), and we obtain the fact that $\sum_{i=1}^{n_1}\sum_{j=1}^{k_1}r_{1,ij}\log[v_0^{-1}\gamma_{1,ij}+v_1^{-1}(1-\gamma_{1,ij})]$ is independent of $\{\gamma_{1,ij}\}$, and the same conclusion also applies to the second term of (63).

Since the lower bound (63) does not dependent on γ_1,γ_2, the determinant factor in the density function $p(\theta,\mu_1,\mu_2\,|\,\gamma_1,\gamma_2,\sigma^2)$ does not play any role in the derivation of the EM algorithm. With some standard calculations, the E-step is given by

$$q_{1,ij}^{\text{new}}=\frac{\exp\left(-\frac{\|\theta_{i*}-\mu_{1,j*}\|^2}{2\sigma^2\bar{v}}\right)}{\sum_{u=1}^{k_1}\exp\left(-\frac{\|\theta_{i*}-\mu_{1,u*}\|^2}{2\sigma^2\bar{v}}\right)},\quad q_{1,lh}^{\text{new}}=\frac{\exp\left(-\frac{\|\theta_{*l}-\mu_{2,*h}\|^2}{2\sigma^2\bar{v}}\right)}{\sum_{v=1}^{k_2}\exp\left(-\frac{\|\theta_{*l}-\mu_{2,*v}\|^2}{2\sigma^2\bar{v}}\right)},$$

where $\bar{v}^{-1}=v_0^{-1}-v_1^{-1}$. The M-step is given by

$$(\alpha^{\text{new}},\theta^{\text{new}},\mu_1^{\text{new}},\mu_2^{\text{new}})=\underset{\alpha,\mathbb{1}_{n_1}^T\theta\mathbb{1}_{n_2}=0,\mu_1,\mu_2}{\text{argmin}}F(\alpha,\theta,\mu_1,\mu_2;q_1^{\text{new}},q_2^{\text{new}}),$$
$$(\sigma^2)^{\text{new}}=\frac{F(\alpha^{\text{new}},\theta^{\text{new}},\mu_1^{\text{new}},\mu_2^{\text{new}};q_1^{\text{new}},q_2^{\text{new}})+b}{2n_1n_2+n_1k_2+n_2k_1+a+2},$$

where

$$\begin{aligned}F(\alpha,\theta,\mu_1,\mu_2;q_1,q_2)&=\|y-\alpha\mathbb{1}_{n_1}\mathbb{1}_{n_2}^T-\theta\|_{\text{F}}^2+\nu\|\alpha\|^2\\&+\sum_{i=1}^{n_1}\sum_{j=1}^{k_1}\left(\frac{q_{1,ij}}{v_0}+\frac{1-q_{1,ij}}{v_1}\right)\|\theta_{i*}-\mu_{1,j*}\|^2\\&+\sum_{l=1}^{n_2}\sum_{h=1}^{k_2}\left(\frac{q_{2,lh}}{v_0}+\frac{1-q_{2,lh}}{v_1}\right)\|\theta_{*l}-\mu_{2,*h}\|^2.\end{aligned}$$

5.3.2. Kronecker product biclustering model

For the Kronecker product structure, we introduce a latent matrix $\mu\in\mathbb{R}^{k_1\times k_2}$. Since the biclustering model implies a block-wise constant structure for θ. Each entry of μ serves as a center for a block of the matrix θ. The prior distribution is defined by

$$\begin{aligned}&p(\theta,\mu\,|\,\gamma_1,\gamma_2,\sigma^2)\\&\propto\prod_{i=1}^{n_1}\prod_{j=1}^{k_1}\prod_{l=1}^{n_2}\prod_{h=1}^{k_2}\exp\left(-\frac{(\theta_{il}-\mu_{jh})^2}{2\sigma^2[v_0\gamma_{1,ij}\gamma_{2,lh}+v_1(1-\gamma_{1,ij}\gamma_{2,lh})]}\right)\mathbb{I}\{\mathbb{1}_{n_1}^T\theta\mathbb{1}_{n_2}=0\}.\end{aligned}$$

The prior distribution is another extension of (33), and it is in a similar form of (59). To finish the Bayesian model specification, we consider the same priors for $\gamma_1, \gamma_2, \sigma^2$ as in the Cartesian product case.

Recall that the lower bound of $\log \sum_{T \in \text{spt}(G)} \sum_{e \in T} [v_0^{-1}\gamma_e + v_1^{-1}(1-\gamma_e)]$ is given by (60) for a general Kronecker product structure. In the current setting, a similar argument gives the lower bound

$$\sum_{i=1}^{n_1}\sum_{j=1}^{k_1}\sum_{l=1}^{n_2}\sum_{h=1}^{k_2} r_{(i,l),(j,h)} \log[v_0^{-1}\gamma_{1,ij}\gamma_{2,lh} + v_1^{-1}(1-\gamma_{1,ij}\gamma_{2,lh})].$$

Since $r_{(i,l),(j,h)} \equiv r$ is independent of $(i,l),(j,h)$ by the symmetry of the complete bipartite graph, the above lower bound can be written as

$$\begin{aligned} & r\sum_{i=1}^{n_1}\sum_{j=1}^{k_1}\sum_{l=1}^{n_2}\sum_{h=1}^{k_2} \log[v_0^{-1}\gamma_{1,ij}\gamma_{2,lh} + v_1^{-1}(1-\gamma_{1,ij}\gamma_{2,lh})] \\ &= r\log(v_0^{-1})\sum_{i=1}^{n_1}\sum_{j=1}^{k_1}\sum_{l=1}^{n_2}\sum_{h=1}^{k_2}\gamma_{1,ij}\gamma_{2,lh} + r\log(v_1^{-1})\sum_{i=1}^{n_1}\sum_{j=1}^{k_1}\sum_{l=1}^{n_2}\sum_{h=1}^{k_2}(1-\gamma_{1,ij}\gamma_{2,lh}) \\ &= rn_1n_2\log(v_0^{-1}) + rn_1n_2(k_1k_2-1)\log(v_1^{-1}), \end{aligned}$$

which is independent of γ_1, γ_2. The inequality (5.3.2) is because both γ_1 and γ_2 satisfy (34).

Again, the determinant factor in the density function $p(\theta, \mu \mid \gamma_1, \gamma_2, \sigma^2)$ does not play any role in the derivation of the EM algorithm, because the lower bound (5.3.2) does not depend on (γ_1, γ_2). Since we are working with the Kronecker product, we will derive a variational EM algorithm with the E-step finding the posterior distribution in the mean filed class $\mathcal{Q} = \{q(\gamma_1, \gamma_2) = q_1(\gamma_1)q_2(\gamma_2) : q_1, q_2\}$. By following the same argument in Section 5.2, we obtain the E-step as

$$\begin{aligned} q_{1,ij}^{\text{new}} &= \frac{\exp\left(-\sum_{l=1}^{n_2}\sum_{h=1}^{k_2}\frac{q_{2,lh}(\theta_{il}-\mu_{jh})^2}{2\sigma^2\bar{v}}\right)}{\sum_{u=1}^{k_1}\exp\left(-\sum_{l=1}^{n_2}\sum_{h=1}^{k_2}\frac{q_{2,lh}(\theta_{il}-\mu_{uh})^2}{2\sigma^2\bar{v}}\right)}, \\ q_{2,lh}^{\text{new}} &= \frac{\exp\left(-\sum_{i=1}^{n_1}\sum_{j=1}^{k_1}\frac{q_{1,ij}^{\text{new}}(\theta_{il}-\mu_{jh})^2}{2\sigma^2\bar{v}}\right)}{\sum_{v=1}^{k_2}\exp\left(-\sum_{i=1}^{n_1}\sum_{v=1}^{k_1}\frac{q_{1,ij}^{\text{new}}(\theta_{il}-\mu_{lv})^2}{2\sigma^2\bar{v}}\right)} \end{aligned}$$

where $\bar{v}^{-1} = v_0^{-1} - v_1^{-1}$. The M-step is given by

$$\begin{aligned} (\alpha^{\text{new}}, \theta^{\text{new}}, \mu^{\text{new}}) &= \underset{\alpha, \mathbb{1}_{n_1}^T\theta\mathbb{1}_{n_2}=0, \mu}{\operatorname{argmin}} F(\alpha, \theta, \mu; q_1^{\text{new}}, q_2^{\text{new}}), \\ (\sigma^2)^{\text{new}} &= \frac{F(\alpha^{\text{new}}, \theta^{\text{new}}, \mu^{\text{new}}; q_1^{\text{new}}, q_2^{\text{new}}) + b}{2n_1n_2 + n_1k_2 + n_2k_1 + a + 2}, \end{aligned}$$

where

$$\begin{aligned} F(\alpha, \theta, \mu_1, \mu_2; q_1, q_2) &= \|y - \alpha\mathbb{1}_{n_1}\mathbb{1}_{n_2}^T - \theta\|_{\text{F}}^2 + \nu\|\alpha\|^2 \\ &\quad + \sum_{i=1}^{n_1}\sum_{j=1}^{k_1}\sum_{l=1}^{n_2}\sum_{h=1}^{k_2}\left(\frac{q_{1,ij}q_{2,lh}}{v_0} + \frac{1-q_{1,ij}q_{2,lh}}{v_1}\right)(\theta_{il}-\mu_{jh})^2. \end{aligned}$$

6. Reduced Isotonic Regression

The models that we have discussed so far in our general framework all involve Gaussian likelihood functions and Gaussian priors. It is important to develop a natural extension of the framework to include non-Gaussian models. In this section, we discuss a reduced isotonic regression problem with a non-Gaussian prior distribution, while a full extension to non-Gaussian models will be considered as a future project.

Given a vector of observation $y \in \mathbb{R}^n$, the reduced isotonic regression seeks the best piecewise constant fit that is nondecreasing (Schell and Singh, 1997; Gao et al., 2018). It is an important model that has applications in problems with natural monotone constraint on the signal. With the likelihood $y|\alpha, \theta, \sigma^2 \sim N(\alpha \mathbb{1}_n + \theta, \sigma^2 I_n)$, we need to specify a prior distribution on θ that induces both piecewise constant and isotonic structures. We propose the following prior distribution,

$$\theta \,|\, \gamma, \sigma^2 \sim p(\theta \,|\, \gamma, \sigma^2) \propto \prod_{i=1}^{n-1} \exp\left(-\frac{(\theta_{i+1}-\theta_i)^2}{2\sigma^2[v_0\gamma_i + v_1(1-\gamma_i)]}\right) \mathbb{I}\{\theta_i \le \theta_{i+1}\} \mathbb{I}\{\mathbb{1}_n^T \theta = 0\}. \quad (64)$$

We call (64) the spike-and-slab half-Gaussian distribution. Note that the support of the distribution is the intersection of the cone $\{\theta : \theta_1 \le \theta_2 \le ... \le \theta_n\}$ and the subspace $\{\theta : \mathbb{1}_n^T \theta = 0\}$. The parameters v_0 and v_1 play similar roles as in (3), which model the closedness between θ_i and θ_{i+1} depending on the value of γ_i.

Proposition 8 *For any $\gamma \in \{0,1\}^{n-1}$ and $v_0, v_1 \in (0, \infty)$, the spike-and-slab half-Gaussian prior (64) is well defined on $\{\theta : \theta_1 \le \theta_2 \le ... \le \theta_n\} \cap \{\theta : \mathbb{1}_n^T \theta = 0\}$, and its density function with respect to the Lebesgue measure restricted on the support is given by*

$$p(\theta \,|\, \gamma, \sigma^2) = 2^{n-1} \frac{1}{(2\pi\sigma^2)^{(n-1)/2}} \sqrt{n \prod_{i=1}^{n-1} [v_0^{-1}\gamma_i + v_1^{-1}(1-\gamma_i)]}$$
$$\times \exp\left(-\sum_{i=1}^{n-1} \frac{(\theta_{i+1}-\theta_i)^2}{2\sigma^2[v_0\gamma_i + v_1(1-\gamma_i)]}\right) \mathbb{I}\{\theta_1 \le \theta_2 \le ... \le \theta_n\} \mathbb{I}\{\mathbb{1}_n^T \theta = 0\}.$$

Note that the only place that Proposition 8 deviates from Proposition 1 is the extra factor 2^{n-1} due to the isotonic constraint $\{\theta : \theta_1 \le \theta_2 \le ... \le \theta_n\}$ and the symmetry of the density. We complete the model specification by put priors on $\alpha, \gamma, \eta, \sigma^2$ that are given by (2), (4), (5) and (6).

Now we are ready to derive the EM algorithm. Since the base graph is a tree, the EM algorithm for reduced isotonic regression is exact. The E-step is given by

$$q_i^{\text{new}} = \frac{\eta\phi(\theta_i - \theta_{i-1}; 0, \sigma^2 v_0)}{\eta\phi(\theta_i - \theta_{i-1}; 0, \sigma^2 v_0) + (1-\eta)\phi(\theta_i - \theta_{i-1}; 0, \sigma^2 v_1)}.$$

The M-step is given by

$$(\alpha^{\text{new}}, \theta^{\text{new}}) = \underset{\alpha, \theta_1 \le \theta_2 \le ... \le \theta_n, \mathbb{1}_n^T \theta = 0}{\operatorname{argmin}} F(\alpha, \theta; q^{\text{new}}), \quad (65)$$

where

$$F(\alpha, \theta; q) = \|y - \alpha \mathbb{1}_n - \theta\|^2 + \nu\alpha^2 + \sum_{i=1}^{n-1} \left(\frac{q_i}{v_0} + \frac{1-q_i}{v_1} \right) (\theta_i - \theta_{i-1})^2,$$

and the updates of σ^2 and η are given by (17) with $p = n$. The M-step (65) can be solved by a very efficient optimization technique. Since $\|y - \alpha\mathbb{1}_n - \theta\|^2 = \|(\bar{y} - \alpha)\mathbb{1}_n\|^2 + \|y - \bar{y}\mathbb{1}_n - \theta\|^2$ by $\mathbb{1}_n^T \theta = 0$, α and θ can be updated independently. It is easy to see that $\alpha^{\text{new}} = \frac{n}{n+\nu}\bar{y}$. The update of θ can be solved by SPAVA (Burdakov and Sysoev, 2017).

Similar to the Gaussian case, the parameter v_0 determines the complexity of the model. For each v_0 between 0 and v_1, we apply the EM algorithm above to calculate $\widehat{q}$, and then let $\widehat{\gamma}_i = \widehat{\gamma}_i(v_0) = \mathbb{I}\{\widehat{q}_i \geq 1/2\}$ form a solution path. The best model will be selected from the EM-solution path by the limiting version of the posterior distribution as $v_0 \to 0$.

Given a $\gamma \in \{0,1\}^{n-1}$, we write $s = 1 + \sum_{i=1}^{n-1}(1 - \gamma_i)$ to be the number of pieces, and $Z_\gamma \in \{0,1\}^{n \times s}$ is the membership matrix defined in Section 3.3. As $v_0 \to 0$, a slight variation of Proposition 3 implies that θ that follows (64) weakly converges to $Z_\gamma \widetilde{\theta}$, where $\widetilde{\theta}$ is distributed by

$$p(\widetilde{\theta} \,|\, \gamma, \sigma^2) \propto \exp\left(-\sum_{l=1}^{s} \frac{(\widetilde{\theta}_l - \widetilde{\theta}_{l+1})^2}{2\sigma^2 v_1} \right) \mathbb{I}\{\widetilde{\theta}_1 \leq \widetilde{\theta}_2 \leq ... \leq \widetilde{\theta}_s\} \mathbb{I}\{\mathbb{1}_n^T Z_\gamma \widetilde{\theta} = 0\}. \tag{66}$$

The following proposition determines the normalizing constant of the above distribution.

Proposition 9 *The density function of (66) is given by*

$$\begin{aligned} p(\widetilde{\theta} \,|\, \gamma, \sigma^2) = 2^{s-1} (2\pi\sigma^2)^{-(s-1)/2} \sqrt{det_{Z_\gamma^T \mathbb{1}_n}(Z_\gamma^T \widetilde{L}_\gamma Z_\gamma)} \times \\ \exp\left(-\sum_{l=1}^{s} \frac{(\widetilde{\theta}_l - \widetilde{\theta}_{l+1})^2}{2\sigma^2 v_1} \right) \mathbb{I}\{\widetilde{\theta}_1 \leq \widetilde{\theta}_2 \leq ... \leq \widetilde{\theta}_s\} \mathbb{I}\{\mathbb{1}_n^T Z_\gamma \widetilde{\theta} = 0\}, \end{aligned} \tag{67}$$

where Z_γ and $\widetilde{L}_\gamma$ are defined in Section 3.3.

Interestingly, compared with the formula (23), (67) has an extra 2^{s-1} due to the isotonic constraint $\{\widetilde{\theta}_1 \leq ... \leq \widetilde{\theta}_s\}$.

Following Section 3.3, we consider a reduced version of the likelihood $y \,|\, \alpha, \widetilde{\theta}, \gamma, \sigma^2 \sim N(\alpha\mathbb{1}_n + Z_\gamma \widetilde{\theta}, \sigma^2 I_n)$. Then, with the prior distributions on $\alpha, \widetilde{\theta}, \gamma, \sigma^2$ specified by (2), (67), (4), (5) and (6), we obtain the joint posterior distribution $p(\alpha, \widetilde{\theta}, \gamma, \sigma^2 \,|\, y)$. Ideally, we would like to integrate out $\alpha, \widetilde{\theta}, \sigma^2$ and use $p(\gamma \,|\, y)$ for model selection. However, the integration with respect to $\widetilde{\theta}$ is intractable due to the isotonic constraint. Therefore, we propose to maximize out $\alpha, \widetilde{\theta}, \sigma^2$, and then the model selection score for reduced isotonic regression is given by

$$g(\gamma) = \max_{\alpha, \widetilde{\theta}_1 \leq ... \leq \widetilde{\theta}_s, \mathbb{1}_n^T Z_\gamma \widetilde{\theta} = 0, \sigma^2} \log p(\alpha, \widetilde{\theta}, \gamma, \sigma^2 \,|\, y).$$

For each γ, the optimization involved in the evaluation of $g(\gamma)$ can be done efficiently, which is very similar to the M-step updates.

7. Numerical Results

In this section, we test the performance of the methods proposed in the paper and compare the accuracy in terms of sparse signal recovery and graphical structure estimation with existing methods. We name our method BayesMSG (Bayesian Model Selection on Graphs) throughout the section. All simulation studies and real data applications were conduced on a standard laptop (2.6 GHz Intel Core i7 processor and 16GB memory) using R and Julia programming languages.

Our Bayesian method outputs a subgraph defined by

$$\widehat{\gamma} = \text{argmax}\left\{g(\gamma) : \gamma \in \{\widehat{\gamma}(v_0)\}_{0<v_0\leq v_1}\right\},$$

which is a sub-model selected by the model selection score $g(\gamma)$ on the EM solution path (see Section 3.3 for details). Suppose γ^* is the underlying true subgraph that generates the data, we measure the performance of $\widehat{\gamma}$ by false discovery proportion and power. The definitions are

$$\mathsf{FDP} = \frac{\sum_{(i,j)\in E}(1-\widehat{\gamma}_{ij})\gamma^*_{ij}}{\sum_{(i,j)\in E}(1-\widehat{\gamma}_{ij})} \quad \text{and} \quad \mathsf{POW} = 1 - \frac{\sum_{(i,j)\in E}(1-\gamma^*_{ij})\widehat{\gamma}_{ij}}{\sum_{(i,j)\in E}(1-\gamma^*_{ij})},$$

where we adopt the convention that $0/0 = 1$. Note that the above FDP and POW are not suitable for the clustering/biclustering model, because clustering structures are equivalent up to arbitrary clustering label permutations.

The sub-model indexed by $\widehat{\gamma}$ also induces a point estimator for the model parameters. This can be done by calculating the posterior mean of the reduced model specified by the likelihood (25) and priors (2) and (23). With notations $p(y|\alpha,\widetilde{\theta},\gamma,\sigma^2)$, $p(\alpha|\sigma^2)$ and $p(\widetilde{\theta}|\gamma,\sigma^2)$ for (25), (2) and (23), the point estimator is defined by $\widehat{\beta} = \alpha^{\text{est}}w + Z_{\widehat{\gamma}}\widetilde{\theta}^{\text{est}}$, where Z_γ is the membership matrix defined in Section 3.3, and the definition of $(\alpha^{\text{est}},\widetilde{\theta}^{\text{est}})$ is given by

$$(\alpha^{\text{est}},\widetilde{\theta}^{\text{est}}) = \underset{\alpha,\widetilde{\theta}\in\{\widetilde{\theta}:w^TZ_{\widehat{\gamma}}\widetilde{\theta}=0\}}{\text{argmax}} \log\left[p(y|\alpha,\widetilde{\theta},\widehat{\gamma},\sigma^2)p(\alpha|\sigma^2)p(\widetilde{\theta}|\widehat{\gamma},\sigma^2)\right],$$

which is a simple quadratic programming whose solution does not depend on σ^2. Note that the definition implies that $\widehat{\beta}$ is the posterior mode of the reduced model. Since the posterior distribution is Gaussian, $\widehat{\beta}$ is also the posterior mean. The performance of $\widehat{\beta}$ will be measured by the mean squared error

$$\mathsf{MSE} = \frac{1}{n}\|X(\widehat{\beta}-\beta^*)\|^2,$$

where β^* is the true parameter that generates the data.

The hyper-parameters a, b, A, B in (5) and (6) are all set as the default value 1. The same rule is also applied to the extensions in Sections 4-6.

7.1. Simulation Studies

In this section, we compare the proposed Bayesian model selection procedure with existing methods in the literature. There are two popular generic methods for graph-structured

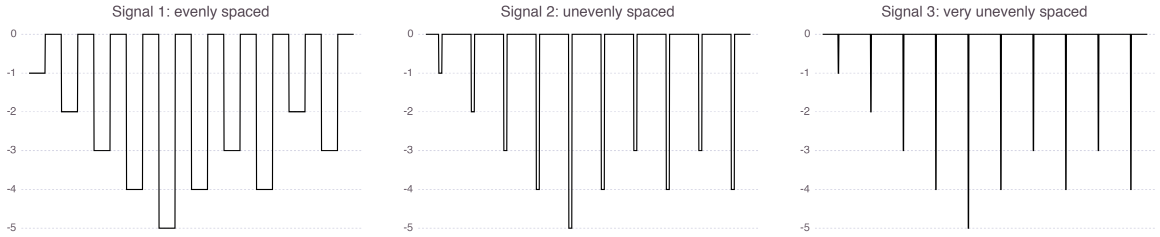


Figure 3: Three different signals for the linear chain graph. All signals have 20 pieces. Signal 1 has evenly spaced changes (each piece has length 50), Signal 2 has unevenly spaced changes (a smaller piece has length 10), and Signal 3 has very unevenly spaced changes (a smaller one has length 2).

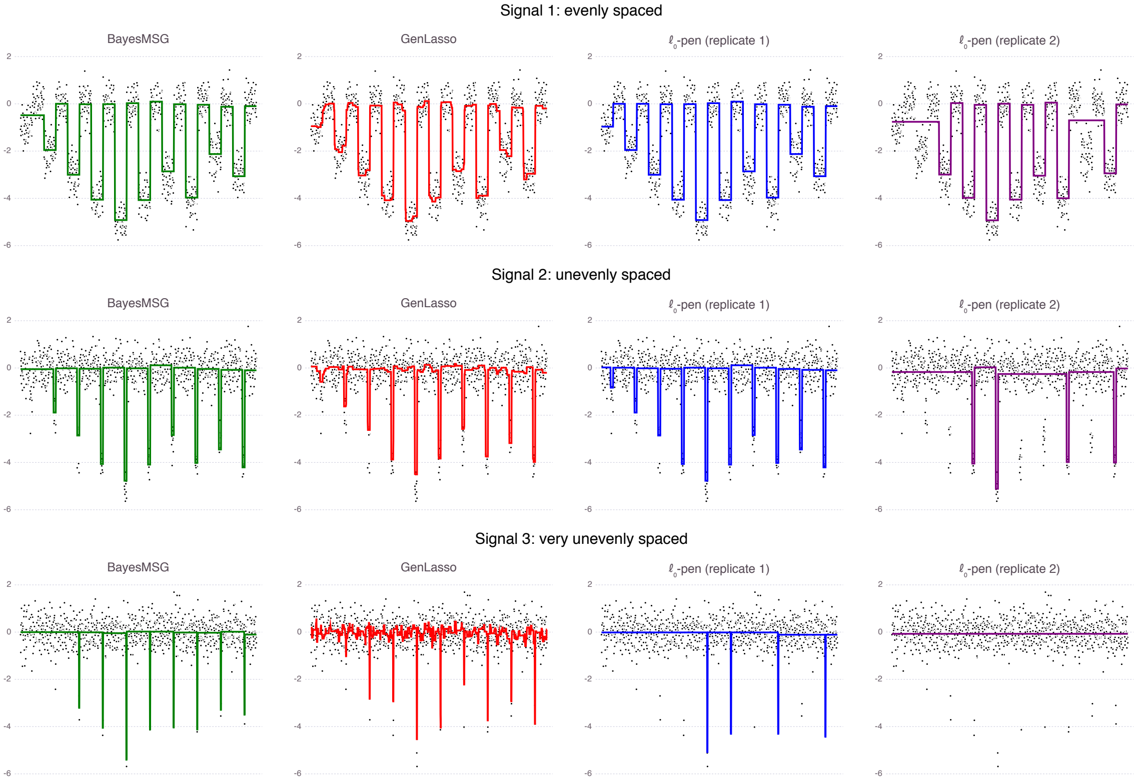


Figure 4: Visualization of typical solutions of the three methods when $\sigma = 0.5$. Since ℓ_0-pen is very unstable, we plot contrasting solutions from two independent replicates. (Top) Evenly spaced signal; (Center) Unequally spaced signal; (Bottom) Very unevenly spaced signal; (Far Left) BayesMSG; (Left) GenLasso; (Right and Far Right) Two independent replicates of ℓ_0-pen.

model selection in the literature. The first method is the generalized Lasso (or simlply GenLasso henceforth) (Tibshirani et al., 2005; She, 2010; Tibshirani and Taylor, 2011), defined by

$$\widehat{\beta} = \frac{1}{2}\|y - X\beta\|^2 + \lambda \sum_{(i,j)\in E} |\beta_i - \beta_j|. \tag{68}$$

The second method is the ℓ_0-penalized least-squares (Barron et al., 1999; Friedrich et al., 2008; Fan and Guan, 2018), defined by

$$\widehat{\beta} = \frac{1}{2}\|y - X\beta\|^2 + \lambda \sum_{(i,j)\in E} \mathbb{I}\{\beta_i \neq \beta_j\}. \tag{69}$$

For both methods, an estimated subgraph is given by

$$\widehat{\gamma}_{ij} = \mathbb{I}\{|\widehat{\beta}_i - \widehat{\beta}_j| \leq \epsilon\},$$

for all $(i,j) \in E$. Here, the number ϵ is taken as 10^{-8}. The two methods are referred to by GenLasso and ℓ_0-pen from now on. In addition to GenLasso and ℓ_0-pen, various other methods (Pérez and de Los Campos, 2014; Bondell and Reich, 2008; Govaert and Nadif, 2003; Tan and Witten, 2014; Gao et al., 2016; Chi et al., 2017; Mair et al., 2009; Gao et al., 2018; Tibshirani et al., 2011) that are specific to different models will also be compared in our simulation studies.

7.1.1. Linear Chain Graph

We first consider the simplest linear chain graph, which corresponds to the change-point mdoel explained in Example 2. We generate data according to $y \sim N(\beta^*, \sigma^2 I_n)$ with $n = 1000$ and $\sigma \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$. The mean vector $\beta^* \in \mathbb{R}^n$ is specified in three different cases as shown in Figure 3.

We compare the performances of the proposed Bayesian method, GenLasso and ℓ_0-pen. For the linear chain graph, GenLasso is the same as fused Lasso (Tibshirani et al., 2005). Its tuning parameter λ in (68) is selected by cross validation using the default method of the R package `genlasso` (Arnold and Tibshirani, 2014). For ℓ_0-pen, the λ in (69) is selected using the method suggested by Fan and Guan (2018).

The results are summarized in Table 1. Some typical solutions of the three methods are plotted in Figure 4. In terms of MSE, our Bayesian method achieves the smallest error among the three methods when σ is small, and GenLasso has the best performance when σ is large. For model selection performance measured by FDP and POW, the Bayesian method is the best, and ℓ_0-pen is better than GenLasso. We also point out that the solutions of ℓ_0-pen is highly unstable, as shown in Figure 4. In terms of computational time, BayesMSG, GenLasso and ℓ_0-pen require 5.2, 11.8 and 19.0 seconds on average.

It is not surprising that GenLasso achieves the lowest MSE in the low signal strength regime. This is because Lasso is known to produce estimators with strong bias towards zero, and therefore it is favored when the true parameters are close to zero. The other two methods, BayesMSG and ℓ_0 are designed to achieve nearly unbiased estimators when the signals are strong, and therefore show their advantages over GenLasso when the signal strength is large.

	σ	Even			Uneven			Very uneven		
		MSE	FDP	POW	MSE	FDP	POW	MSE	FDP	POW
BayesMSG	0.1	0.00019	0.00	1.00	0.00949	0.00	1.00	0.00217	0.00	0.80
	0.2	0.00585	0.00	0.98	0.01010	0.00	0.97	0.00279	0.00	0.81
	0.3	0.01620	0.01	0.96	0.01116	0.01	0.97	0.00349	0.00	0.81
	0.4	0.01940	0.05	0.95	0.01693	0.02	0.96	0.00837	0.00	0.79
	0.5	0.04667	0.10	0.95	0.03682	0.02	0.96	0.01803	0.05	0.78
GenLasso	0.1	0.00094	0.81	1.00	0.00116	0.90	1.00	0.00570	0.96	1.00
	0.2	0.00374	0.81	1.00	0.00458	0.90	1.00	0.01152	0.94	1.00
	0.3	0.00842	0.81	0.98	0.01024	0.89	1.00	0.02084	0.93	0.99
	0.4	0.01494	0.81	0.98	0.01813	0.88	0.98	0.03376	0.92	0.98
	0.5	0.02345	0.82	0.98	0.02818	0.89	0.98	0.04984	0.92	0.97
ℓ_0-pen	0.1	0.00505	0.00	0.98	0.00288	0.00	0.97	0.02042	0.00	0.81
	0.2	0.00545	0.00	0.98	0.00888	0.00	0.94	0.06049	0.00	0.63
	0.3	0.00399	0.01	0.98	0.00918	0.02	0.94	0.06121	0.00	0.63
	0.4	0.00826	0.02	0.97	0.01119	0.02	0.93	0.06250	0.00	0.63
	0.5	0.06512	0.03	0.92	0.04627	0.02	0.93	0.06452	0.00	0.63

Table 1: Comparisons of the three methods for the linear chain graph.

An interesting question would be if it is possible to design a method that works well in all of the three criteria (MSE, POW, FDP) with both low and strong signal? Unfortunately, a recent paper (Song and Cheng, 2018) proves that this is impossible. The result of Song and Cheng (2018) rigorously establishes the incompatibility phenomenon between selection consistency and rate-optimality in high-dimensional sparse regression. We therefore believe our proposed BayesMSG, which performs very well in terms of the three criteria (MSE, POW, FDP) except for the MSE in the low signal regime is a very good solution in view of this recent impossibility result.

7.1.2. Regression with Graph-structured Coefficients

For this experiment, we consider a linear regression setting with graph structured sparsity on the regression coefficients. We sample random Gaussian measurements $X_{ij} \sim N(0,1)$ and measurement errors $\epsilon_i \sim N(0,1)$. Then we construct a design matrix $X \in \mathbb{R}^{n \times p}$ and a response vector $y = X\theta + \epsilon \in \mathbb{R}^n$, where θ is a vector of node attributes of an underlying graph G.

We fix $n = 500$ throughout this simulation study, and consider three different graph settings listed in Table 2. When G is a star graph with the center node at 0, our proposed model in Section 2 corresponds to the sparse linear regression problem. We compare our proposed approach BayesMSG with the following baseline methods implemented in R programming language: Lasso (`glmnet` R package) (Friedman et al., 2010), and Bayesian Spike-And-Slab linear regression (BSAS) via MCMC (`BGLR` R package) (Pérez and de Los Campos, 2014). All the R packages listed here are implemented using their default setting and their recommended model selection methods.

Next, when G is a linear chain graph, the model corresponds to the linear regression problem with a sparse graph difference vector $(\theta_2 - \theta_1, \cdots, \theta_p - \theta_{p-1})$. This problem setting is particularly studied for fused Lasso (Tibshirani et al., 2005) and the approximate ℓ_0

Graph	# of nodes	# of edges	Description
Star	1,001 (1 fixed node)	1,001	regression with sparse coefficients
Linear chain	1,000	999	regression with gradient-sparse signals
Complete	200	19,900	regression with clustered coefficients

Table 2: Simulation Settings for Gaussian Design.

Graph	Star Graph			Linear Chain Graph			Complete Graph		
Method	BMSG	Lasso	BSAS	BMSG	GLasso	ITALE	BMSG	GLasso	OSCAR
MSE	0.579	0.792	0.530	0.099	0.276	0.159	0.399	0.472	0.438
Time	5.425	1.663	11.32	3.161	5.791	6.543	27.52	50.62	18.18
FDP	0.324	0.662	-	0.000	0.953	0.106	0.224	0.614	0.467
POW	0.978	0.995	-	0.954	0.980	0.988	0.996	1.000	1.000

Graph	Star Graph			Linear Chain Graph			Complete Graph		
Method	BMSG	Lasso	BSAS	BMSG	GLasso	ITALE	BMSG	GLasso	OSCAR
MSE	0.624	0.789	0.592	0.099	0.274	0.176	0.403	0.472	0.442
Time	5.396	1.721	12.75	3.436	5.904	6.554	22.85	51.42	15.80
FDP	0.319	0.658	-	0.000	0.930	0.032	0.205	0.290	0.208
POW	0.983	0.982	-	0.994	0.988	0.996	0.998	1.000	0.994

Table 3: Simulation Results for Gaussian Design $C = 0.5$ (above) $C = 1$ (below).

regression setting (ITALE) (Xu and Fan, 2019). Therefore, we compare BayesMSG with the above baseline approaches: fused Lasso (`genlasso` R package) and ITALE (`ITALE` R package) on the linear chain graph.

Finally, when G is a complete graph, the model corresponds to the linear regression problem with clustered coefficients, i.e. β_j's may be clustered together. This particular problem setting is also considered in the studies of GenLasso (Tibshirani and Taylor, 2011) and OSCAR (Bondell and Reich, 2008). We compare BayesMSG with the following baseline methods: GenLasso and OSCAR (`lqa` R package). In brief, OSCAR seeks to solve

$$\text{minimize}_\beta \quad \frac{1}{2}\|y - X\theta\|_2^2 + \lambda \sum_{j=1}^{p} (c(j-1)+1)|\theta|_{(j)}.$$

A true graph-structured sparse signal is constructed as follows. For the case of star graphs, $\theta_j^* = 0.5C$ for $j = 1, \cdots, 40$ and 0 otherwise. For the cases of linear change graphs and complete graphs, $\theta_j^* = C$ for $j = 1, \cdots, 0.4p$, $\theta_j^* = 2C$ for $j = 0.4p+1, \cdots, 0.7p$, $\theta_j^* = 3C$ for $j = 0.7p+1, \cdots, 0.9p$ and $4C$ otherwise. Tables 3 displays the estimation error (MSEs) $\|\theta^* - \widehat{\theta}\|_2$ on the test data sets, and computation times. BMSG, BSAS and GenLasso are abbreviations for BayesMSG, Bayesian Spike-And-Slab regression and Generalized Lasso. Each reported error value is averaged across 10 independent simulations with different random seeds.

The results show that our proposed BayesMSG method is the overall winners across all models in both estimation error (MSE) and model selection error (FDP and POW). The advantage is especially obvious for the linear chain graph and the complete graph. The only case where BayesMSG cannot beat its competitor (BSAS) is the estimation error in the

Name	# of nodes	# of edges	mean.ER	sd.ER	diameter	# of CC
Chicago roadmap	4126	4308	0.9575	0.0499	324	1
Enron email	4112	14520	0.2831	0.2341	14	1
Facebook egonet	4039	88234	0.0457	0.0608	8	1

Table 4: Graph properties of the three real networks.

Name	# of clust	# of nodes in each cluster	# of cuts	total variation
Chicago roadmap	4	$(576, 678, 835, 2037)$	31	$31 \times \kappa$
Enron email	4	$(384, 538, 1531, 1659)$	4570	$5047 \times \kappa$
Facebook egonet	4	$(750, 753, 778, 1758)$	651	$1220 \times \kappa$

Table 5: Important features of the signals on the three networks.

star graph case (sparse linear regression). However, we note that BSAS is an MCMC-based method that does not involve model selection but perform Bayesian model averaging. Thus, the solution of BSAS is not sparse. On the other hand, BayesMSG is designed for model selection, and therefore performs much better in terms of model selection error (FDP and POW).

7.1.3. Two-Dimensional Grid Graph

We consider the two-dimensional grid graph described in Example 3. The data is generated according to $y_{ij} \sim N(\kappa\mu^*_{ij}, 1)$ for $i = 1, ..., 21$ and $j = 1, ..., 21$, where

$$\mu^*_{ij} = \left\lceil 2.8 \cos\left(\frac{\sqrt{i^2 + j^2}}{2\pi}\right) - 0.2 \right\rceil,$$

and $\kappa \in \{1, 2, ..., 10\}$ is used to control the signal strength. Note that μ^*_{ij} has a piecewise constant structure because of the operation by $\lceil\cdot\rceil$ that denotes the integer part. In fact, μ^*_{ij} only takes 5 possible values as shown in Figure 6.

Since the R package `genlasso` does not provide a tuning method for the λ in (68) for the two-dimensional grid graph setting, we report MSE based on the λ selected by cross validation, and FDP and POW are reported based on the λ that minimizes the FDP. The λ in ℓ_0-pen is tuned by the method in Fan and Guan (2018).

The results are shown in Figure 5. It is clear that our method outperforms the other two in terms of all the evaluation criteria when the signal strength is not very small. When the signal strength is very small, GenLasso achieves the lowest MSE but shows poor model selection performance. We also illustrate the solution path of our method in Figure 6. Typical solutions of GenLasso and ℓ_0-pen are visualized in Figure 7. We observe that ℓ_0-pen tends to oversmooth the data, while GenLasso tends to undersmooth. In terms of the computational time, BayesMSG, GenLasso and ℓ_0-pen require 21.2, 26.7 and 8.4 seconds on average.

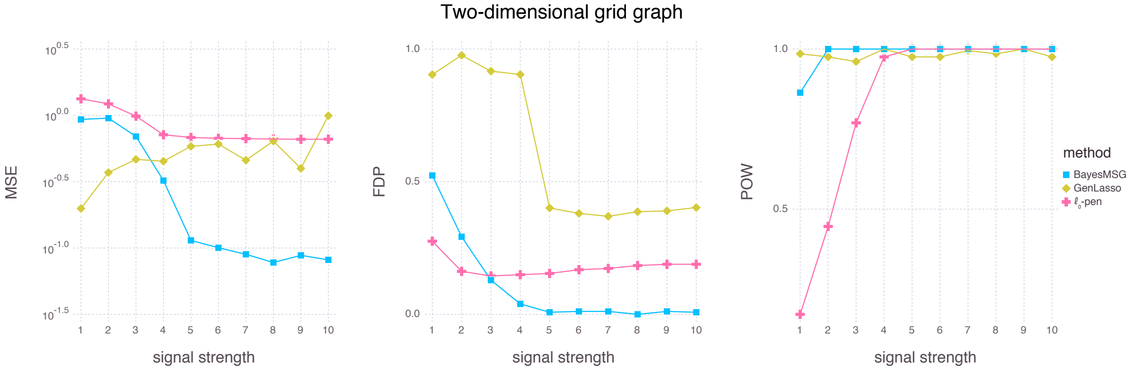


Figure 5: Comparison of the three methods for the two-dimensional grid graph. (Left) MSE; (Center) FDP; (Right) POW.

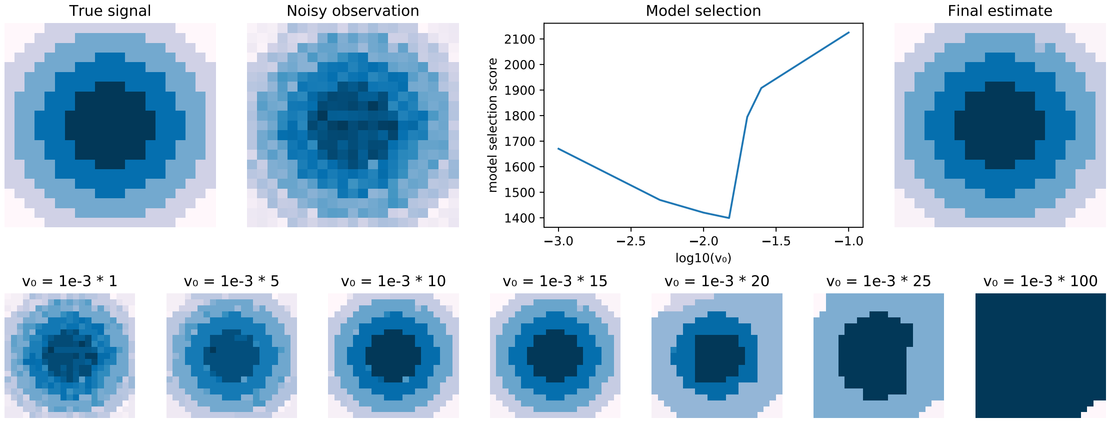


Figure 6: (Top panels) True signal, noisy observations, model selection score, and final estimate; (Bottom panels) A regularization path from $v_0 = 10^{-3}$ to $v_0 = 10^{-1}$.

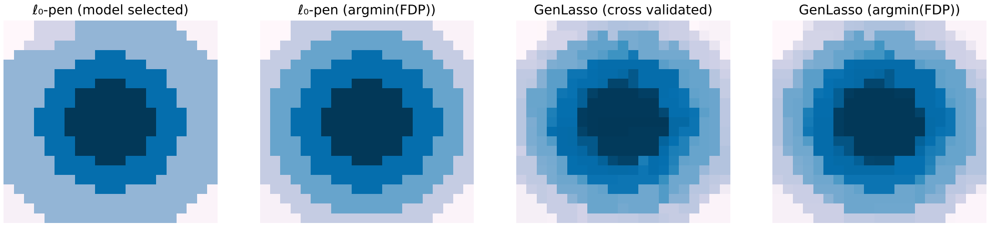


Figure 7: (Far Left) ℓ_0-pen with λ selected using the method in Fan and Guan (2018); (Left) ℓ_0-pen with λ that minimizes FDP; (Right) GenLasso with λ selected by cross validation; (Far Right) GenLasso with λ that minimizes FDP.

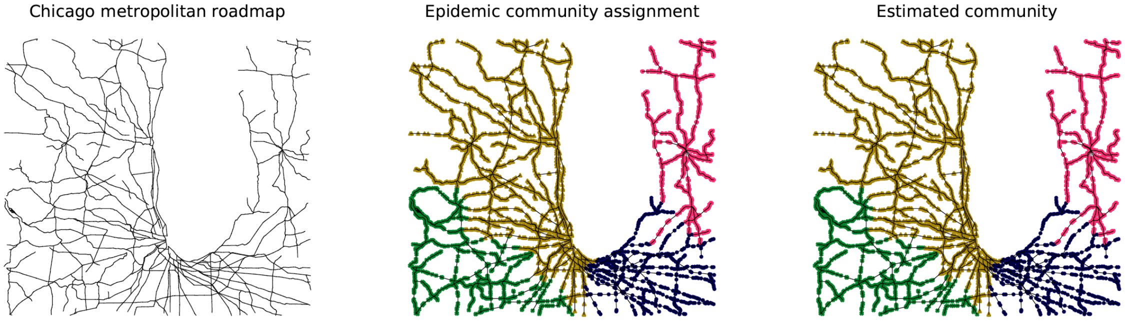


Figure 8: The Chicago roadmap network with signals exhibiting four clusters.

7.1.4. Generic Graphs

In this section, we consider some graphical structures that naturally arise in real world applications. The three graphs to be tested are the Chicago metropolitan area road network[2], the Enron email network[3], and the Facebook egonet network[4]. For all the three networks, we extract induced subgraphs of sizes about 4000. Graph properties for the three networks are summarized in Table 4. For each network, we calculate its number of nodes, number of edges, mean and standard deviation of effective resistances, diameter, and number of connected components. We observe that the three networks behave very differently. The Chicago roadmap network is locally and globally tree-like, since its number of edges is very close to its number of nodes, and the distribution of its effective resistances highly concentrates around 1. The other two networks, the Enron email network and the Facebook egonet, are denser graphs but their effective resistances behave in very different ways.

For each network, we generate data according to $y_i \sim N(\kappa\mu_i^*, 1)$ on its set of nodes, with the signal strength varies according to $\kappa \in \{1, 2, ..., 5\}$. The signal μ^* for each graph is generated as follows:

1. Pick four anchor nodes from the the set of all nodes uniformly at random.

2. For each node, compute the the length of the shortest path to each of the four anchor nodes.

3. Code the ith node by j if the jth anchor node is the closest one to the ith node. This gives four clusters for each graph.

4. Generate a piecewise constant signal $\mu_i^* = j$.

Some properties of the signals are summarized in Table 5, where the number of cuts of μ^* with respect to the base graph $G = (V, E)$ is defined by $\sum_{(i,j)\in E} \mathbb{I}\{\mu_i^* \neq \mu_j^*\}$, and the total variation of μ^* means $\sum_{(i,j)\in E} |\mu_i^* - \mu_j^*|$. We also plot the signal on the Chicago roadmap network in Figure 8.

[2] The data set can be retrieved from `http://www.cs.utah.edu/~lifeifei/SpatialDataset.htm`.

[3] The data set can be retrieved from `http://snap.stanford.edu/data/email-Enron.html`.

[4] The data set can be retrieved from `http://snap.stanford.edu/data/ego-Facebook.html`.

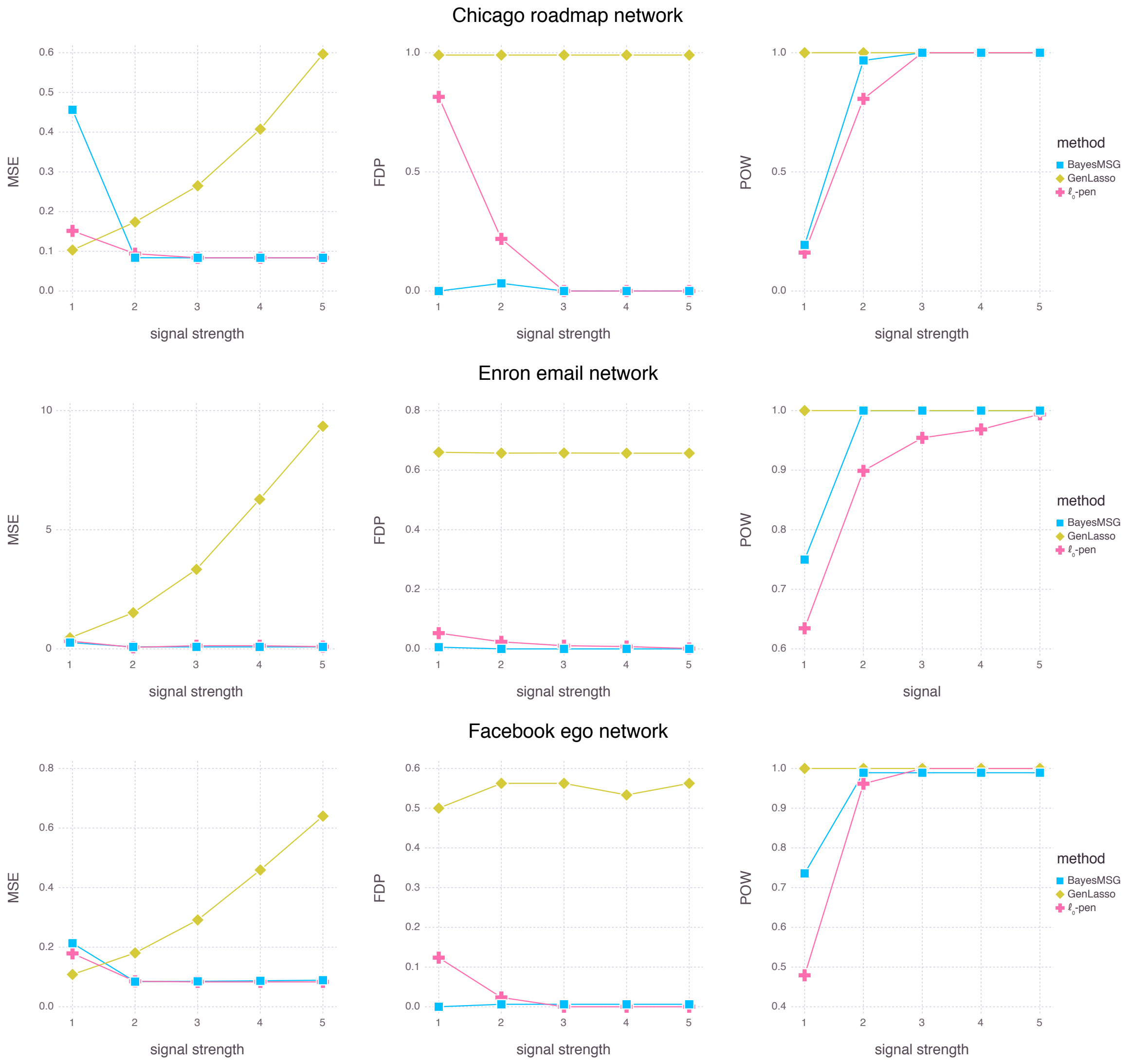


Figure 9: Comparison of the three methods on generic graphs. (Top) Chicago Roadmap network; (Center) Enron Email network; (Bottom) Facebook Ego network.

Since the R package `genlasso` does not provide a tuning method for the λ in (68) for a generic graph, we report MSE based on the λ selected by cross validation, and FDP and POW are reported based on the λ that minimizes the FDP. The λ in ℓ_0-pen is tuned by the method in Fan and Guan (2018).

The results are shown in Figure 9. It is clear that our method outperforms the other two. When the signal strength κ is small, we observe that GenLasso sometimes has the smallest MSE, but its MSE grows very quickly as κ increases. For most κ's, our method and ℓ_0-pen are similar in terms of MSE. In terms of the model selection performance, GenLasso is not competitive, and our method outperforms ℓ_0-pen.

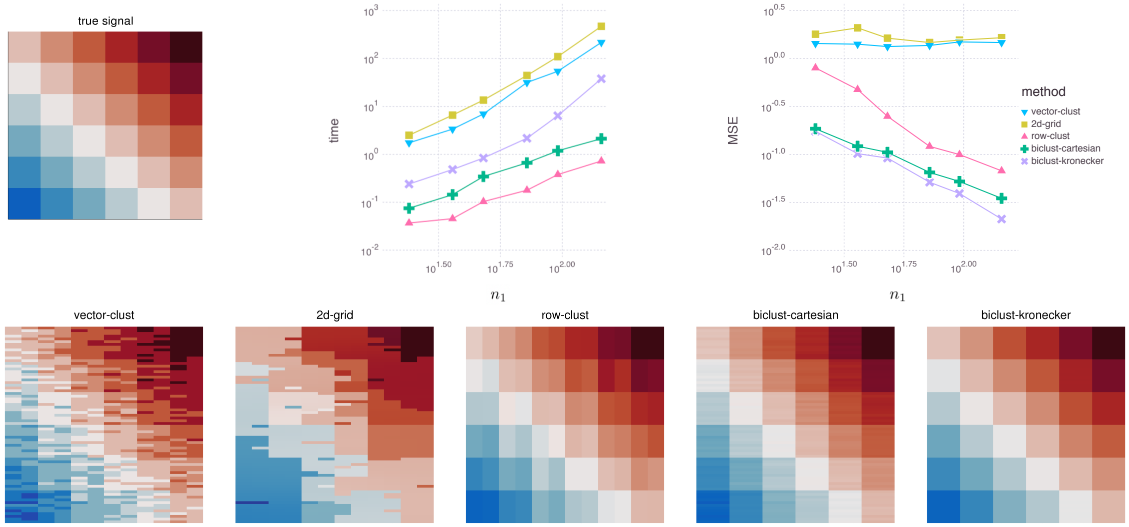


Figure 10: (Top left) True signal; (Top center) Computational time; (Top right) MSE; (Bottom) Heatmaps of estimators using different models ($n_1 = 72$).

7.1.5. Comparison of Different Base Graphs

One key ingredient of our Bayesian model selection framework is the specification of the base graph. For the same problem, there can be multiple ways to specify the base graph that lead to completely different models and methods. In this section, we consider an example and compare the performances of different Bayesian methods with different base graphs.

We consider observations $y_{ij} \sim N(\theta^*_{ij}, 1)$ for $i \in [n_1]$ and $j \in [n_2]$. We fix $n_2 = 12$ and vary n_1 from 24 to 144. The signal matrix $\theta^* \in \mathbb{R}^{n_1 \times n_2}$ has a checkerboard structure as shown in Figure 10. That is, the $n_1 \times n_2$ matrix is divided into 6×6 equal-sized blocks. On the (u, v)th block, $\theta^*_{ij} = 2(u + v - 6)$.

The following models are considered to fit the observations:

1. *Vector clustering.* We regard the matrix $y \in \mathbb{R}^{n_1 \times n_2}$ as a $n_1 n_2$-dimensional vector and apply the clustering model described in Section 4.2 with $n = k = n_1 n_2$.

2. *Two-dimensional grid graph.* The two-dimensional image denoising model described in Example 3 is fit to the observations.

3. *Row clustering.* We regard the matrix $y \in \mathbb{R}^{n_1 \times n_2}$ as $n = n_1$ observations in $\mathbb{R}^d$ with $d = n_2$, and then fit the clustering model described in Section 4.2 to the rows of y with $n = k = n_1$.

4. *Cartesian product biclustering.* The biclustering model induced by the Cartesian product described in Section 5.3 is fit to the observations.

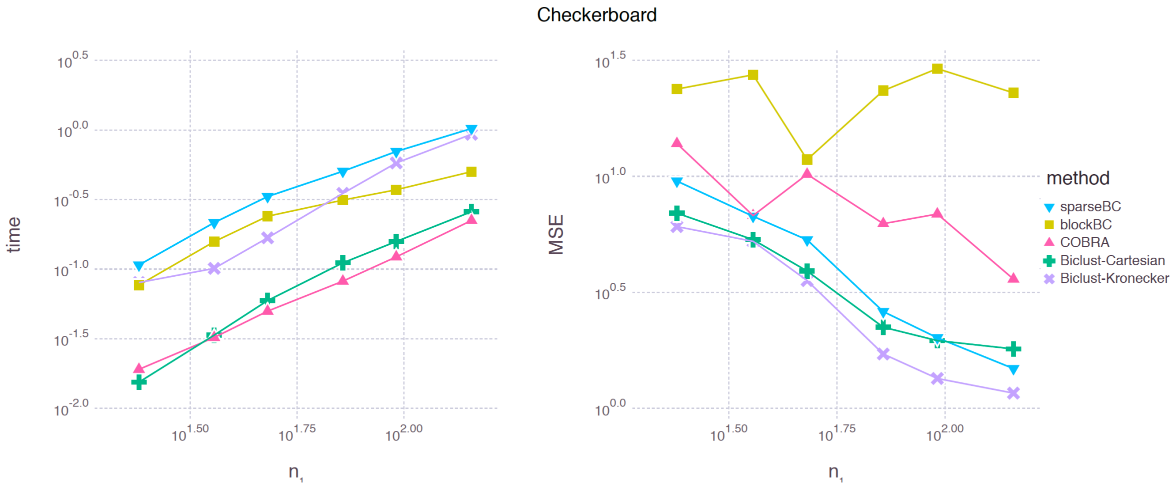


Figure 11: Comparisons of the 5 biclustering methods for the checkerboard data. (Left) Time; (Right) MSE.

5. *Kronecker product biclustering.* The biclustering model induced by the Kronecker product described in Section 5.3 is fit to the observations.

Figure 10 summarizes the results. In terms of MSE, the vector clustering and the two-dimensional grid graph do not fully capture the structure of the data and thus perform worse than all other methods. Both the biclustering models are designed for the checkerboard structure, and they therefore have the best performances. Between the two biclustering models, the one induced by the Kronecker product has a smaller MSE at the cost of a higher computational time.

To summarize the comparisons, we would like to emphasize that the right choice of the base graph has an enormous impact to the result. This also highlights the flexibility of our Bayesian model selection framework that is able to capture various degrees of structures of the data.

7.1.6. Biclustering

To evaluate performance of our biclustering methods in comparison to existing methods, we provide a simulation study using the same data in Section 7.1.5. The true parameter $\theta \in \mathbb{R}^{n_1 \times n_2}$ has a checkerboard structure and is visualized in Figure 10. More precisely, we consider observations $y_{ij} \sim N(\theta^*_{ij}, 1)$ for $i \in [n_1]$ and $j \in [n_2]$. We fix $n_2 = 12$ and vary n_1 from 24 to 144. The signal matrix $\theta^* \in \mathbb{R}^{n_1 \times n_2}$ has a checkerboard structure as shown in Figure 10. That is, the $n_1 \times n_2$ matrix is divided into 6×6 equal-sized blocks. On the (u, v)th block, $\theta^*_{ij} = 2(u + v - 6)$.

For comparison, we consider the three competitors, blockBC (Govaert and Nadif, 2003), sparseBC (Tan and Witten, 2014), and COBRA (Chi et al., 2017), with the implementation via R packages `blockcluster`, `sparseBC` and `cvxbiclustr`. In summary, blockBC and sparseBC are based on the minimization of $\sum_{i,j}(y_{ij} - \mu_{z_1(i)z_2(j)})^2$ given the number k_1 and

k_2 of row and column clusters, respectively, where $\mu \in \mathbb{R}^{k_1 \times k_2}$ is a matrix of latent hidden bicluster means and z_1 and z_2 are row and column cluster assignments, respectively. The methods blockBC and sparseBC use different approaches for the estimation of row and column clusters, i.e. blockBC uses a block EM algorithm. sparseBC solves a penalized linear regression with the ℓ_1-penalty $\lambda\|\mu\|_1$. COBRA is an alternating direction method of multipliers (ADMM) algorithm based on the minimization of $\sum_{i,j}(y_{ij} - \theta_{ij})) + \text{pen}_{\text{row}}(\theta) + \text{pen}_{\text{col}}(\theta)$ where $\text{pen}_{\text{row}}(\theta) = \sum_{i<j} w_{ij}\|\theta_{i*} - \theta_{j*}\|_2$ and pen_{col} is defined similarly.

The result is displayed in Figure 11. The result shows that BayesMSG Kronecker outperforms other methods in terms of MSE. BayesMSG Cartesian also produces competitive solutions in terms of MSE within short amount of computation time. Note that BayesMSG and COBRA are regularization path based methods, and require less computation time than other approaches (blockBC and sparseBC) based on cross validation.

7.2. Real Data Applications

In this section, we apply our methods to three different data sets.

7.2.1. Global warming data

The global warming data has been studied previously by Wu et al. (2001); Tibshirani et al. (2011). It consists of 166 data points in degree Celsius from 1850 to 2015. Here we fit the Bayesian reduced isotonic regression discussed in Section 6. Our results are shown in Figure 12.

When v_0 is nearly zero, the solution is very close to the regular isotonic regression that can be solved efficiently by the pool-adjacent-violators algorithm (PAVA) (Mair et al., 2009). When $v_0 = 0.005$, we obtain a fit with 24 pieces. The PAVA outputs a very similar fit also with 24 pieces. In contrast, the Bayesian model selection procedure suggests a model with $v_0 = 0.06$, which has only 6 pieces, a significantly more parsimonious and a more interpretable fit. This may suggest global warming is accelerating faster in recent years. The same conclusion cannot be obtained from the suboptimal fit with 24 pieces.

To compare with existing methods, we have implemented reduced isotonic regression with dynamic programming (DP) algorithm introduced by Gao et al. (2018) and the near isotonic (NearIso) regression (Tibshirani et al., 2011). Figure 13 shows the estimated signals (left panel) and the estimated differences $\{\theta_{i+1} - \theta_i : i = 1, \cdots, n-1\}$ between the two adjacent years (right panel). Since the DP algorithm does not have a practical model selection procedure, we use the number of pieces $k = 9$ selected by BayesMSG. NearIso uses Mallow's C_p for model selection (Tibshirani et al., 2011). The left panel of Figure 13 shows that BayesMSG outputs a more sparse but still reasonable isotonic fit. On the other hand, NearIso relaxes the isotonic constraint, allowing non-monotone signals but penalizing the decreasing portion of variations. Indeed, one can see from the right panel of Figure 13 that the changepoints of NearIso contain those of PAVA. The PAVA solution is the least parsimonious isotonic fit by definition, and thus we can conclude that NearIso does not seem to find a more parsimonious fit.

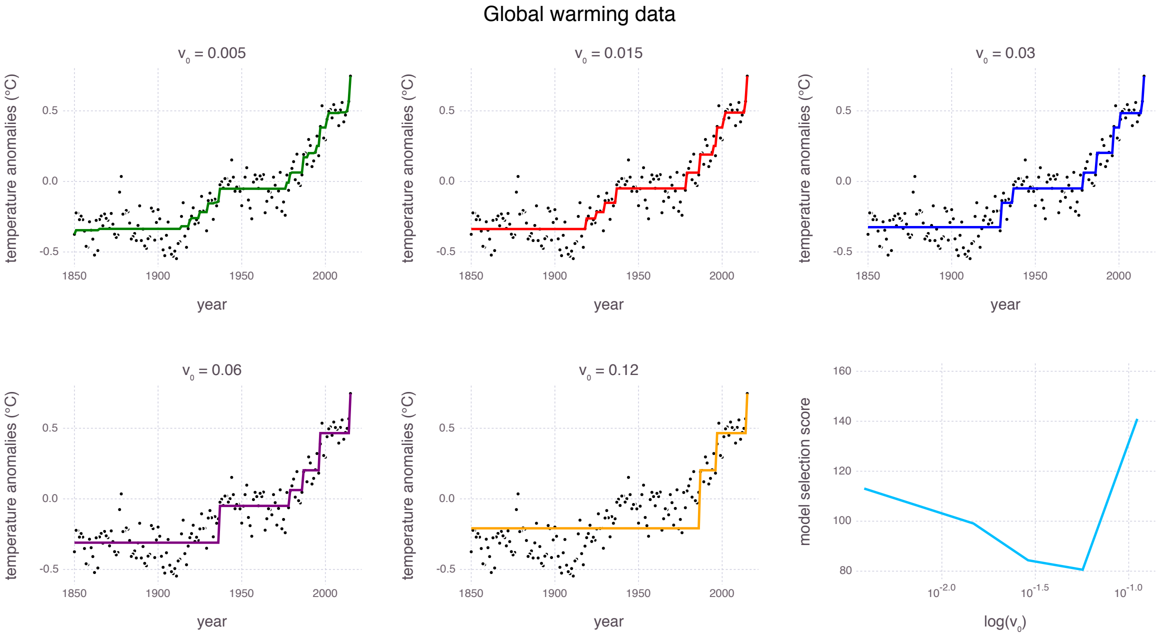


Figure 12: The solution path (smaller v_0 at top left and larger v_0 at bottom right) for Bayesian reduced isotonic regression.

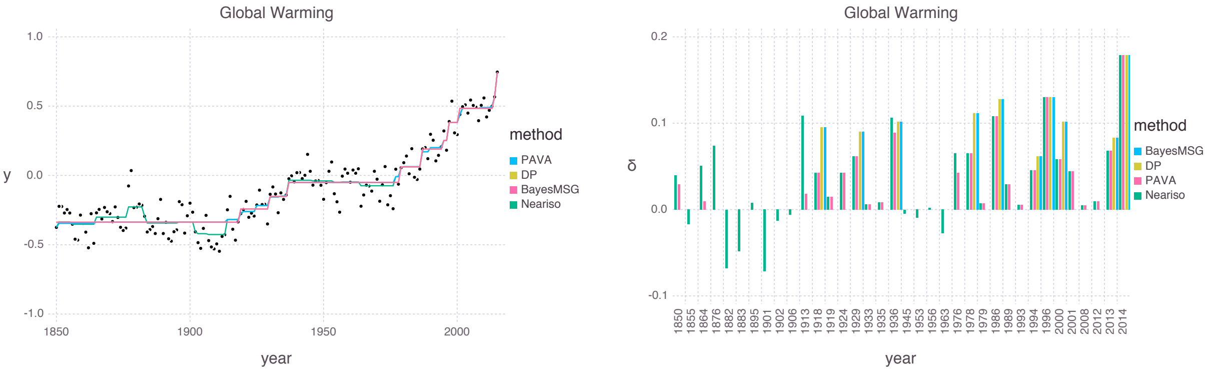


Figure 13: Comparison of various isotonic regression methods. (Left) The estimated isotonic signals (PAVA, DP, BayesMSG, NearIso); let us mention that DP and BayesMSG signals exactly coincide. (Right) The estimated differences between the two adjacent years; the only years (x-axis) with at least one nonzero differences are reported.

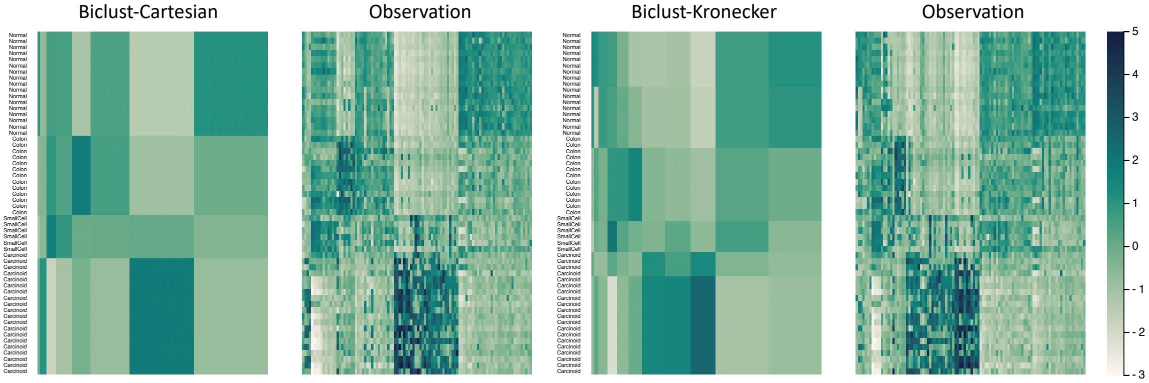


Figure 14: Results of biclustering for the lung cancer data^5.

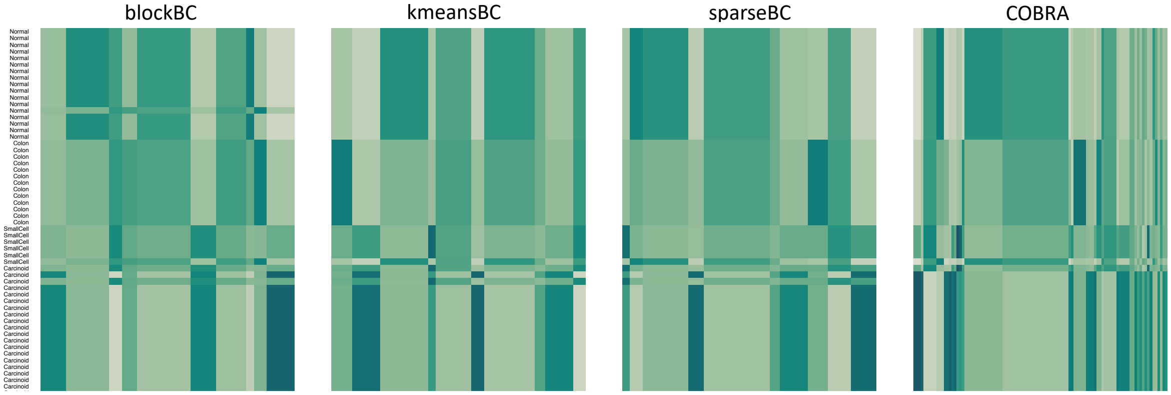


Figure 15: Comparison of existing biclustering methods.

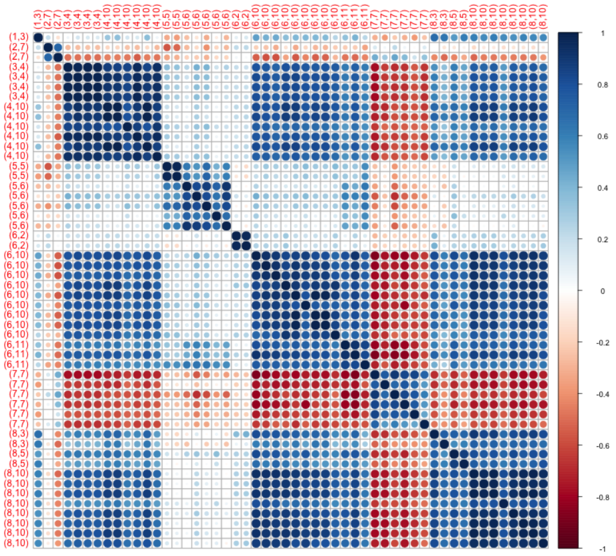


Figure 16: A correlation plot of the selected genes.

Label	Gene description/GenBank ID
$(2,7)$	"proteoglycan 1, secretory granule", "AI932613: Homo sapiens cDNA, 3 end"
$(3,4)$	"AI147237: Homo sapiens cDNA, 3 end", "S71043: Ig A heavy chain allotype 2", "advanced glycosylation end product-specific receptor", "leukocyte immunoglobulin-like receptor, subfamily B"
$(5,5)$	'immunoglobulin lambda locus", "glypican 3"
$(5,6)$	"glutamate receptor, ionotropic, AMPA 2", "small inducible cytokine subfamily A", "W60864: Homo sapiens cDNA, 3 end", "secreted phosphoprotein 1", "LPS-induced TNF-alpha factor"
$(6,2)$	"interleukin 6", "carcinoembryonic antigen-related cell adhesion molecule 5"
$(6,11)$	"secretory granule, neuroendocrine protein 1", "alcohol dehydrogenase 2", "neurofilament, light polypeptide"
$(8,3)$	"fmajor histocompatibility complex, class II", "glycoprotein (transmembrane) nmb"
$(8,5)$	"N90866: Homo sapiens cDNA, 3 end", 'receptor (calcitonin) activity modifying protein 1"

Table 6: Description or GenBank ID of the selected gene clusters of size at least 2 and at most 5.

7.2.2. Lung cancer data

We illustrate the Bayesian biclustering models by a gene expression data set from a lung cancer study. The same data set has also been used by Bhattacharjee et al. (2001); Lee et al. (2010); Sill et al. (2011); Chi et al. (2017). Following Chi et al. (2017), we study a subset with 56 samples and 100 genes. The 56 samples comprise 20 pulmonary carcinoid samples (Carcinoid), 13 colon cancer metastasis samples (Colon), 17 normal lung samples (Normal) and 6 small cell lung carcinoma samples (SmallCell). We also apply the row and column normalizations as has been done in Chi et al. (2017).

Our goal is to identify sets of biologically relevant genes, for example, that are significantly expressed for certain cancer types. We fit both Bayesian biclustering models (Section 5.3) induced by the Cartesian and Kronecker products to the data with $n_1 = 56$, $n_2 = 100$, $k_2 = 10$, and $k_2 = 20$. Recall that k_1 and k_2 are upper bounds of the numbers of row and column clusters, and the actual numbers of row and column clusters will be learned through Bayesian model selection. To pursue a more flexible procedure of model selection, we use two independent pairs of (v_0, v_1) for the row structure and the column structure. To be specific, let (v_0, v_1) be the parameters for the row structure, and the parameters for the column structure are set as (cv_0, cv_1) with some $c \in \{1/10, 1/5, 1/2, 1, 2, 5, 10\}$. Then, the model selection scores are computed with both v_0 and c varying in their ranges.

The results are shown in Figure 14. The two methods select different models with different interpretations. The Cartesian product fit gives 4 row clusters and 8 column clusters, while the Kroneker product fit gives 6 row clusters and 11 column clusters. Even though we have not used the information of the row labels for both biclustering methods, the row clustering structure output by the Cartesian product model almost coincides with these labels except one. On the other hand, the Kronecker product model leads to a finer row clustering structure, with potential discoveries of subtypes of both normal lung samples and pulmonary carcinoid samples.

[5] The rows of the four heatmaps are ordered in the same way according to the labels of tumor types.

Using the same lung cancer data set, we have also compared the BayesMSG Cartesian product method with several existing biclustering methods implemented via R packages `blockcluster`, `sparseBC` and `cvxbiclustr`. Figure 11 displays the final estimates of the four competitors, blockBC (Govaert and Nadif, 2003), sparseBC (Tan and Witten, 2014), kmeansBC (Gao et al., 2016) and COBRA (Chi et al., 2017). One can qualitatively compare these results with the BayesMSG solutions in Figure 15.

The results show that BayesMSG and all the competitors select models with $\widehat{k}_1 = 4$ row clusters. However, BayesMSG selects a smallest number of column clusters. This implies that BayesMSG favors a more parsimonious model compared to the others. Also, the selected BayesMSG Cartesian model achieves a lowest misclassification error for the row (cancer type).

An important goal of biclustering is to simultaneously identify gene and tumor types. To be specific, we seek to find genes that show different expression levels for different types of samples. To this end, we report those genes that are clustered together by both the Cartesian and Kronecker product structures. Groups of genes with size between 2 and 5 are reported in Table 6. Note that our gene clustering is assisted by the sample clustering in the biclustering framework, which is different from gene clustering methods that are only based on the correlation structure (Bhattacharjee et al., 2001). As a sanity check, the correlation matrix of the subset of the selected genes is plotted in Figure 16, and we can observe a clear pattern of block structure.

7.2.3. Chicago crime data

The Chicago crime data is publicly available at Chicago Police Department website[6]. The report in the website contains time, type, district, community area, latitude and longitude of each crime occurred. After removing missing data, we obtain 6.0 millions of crimes that occurred in 22 police districts from 2003 to 2018 (16 years). Here we restrict ourselves to the analysis of the spatial and temporal structure of Chicago crimes within the past few years, ignoring the types or categories of the crimes.

Since the 22 districts have different area sizes, we divide the total numbers of crimes in each district by its population density (population per unit area). We will call this quantity the *Chicago crime rate*. We observe that the Chicago crime rates exhibit decreasing patterns over the years. Since our study is focused on the relative comparisons among different police districts, we divide each entry by the sum of the yearly Chicago crime rates over all the 22 district in its current year. We will call this quantity the *relative crime rate*. Admittedly this preprocessing step does not reflect the difference between the *residential* and the *floating* populations in each district which might be important for the analysis of the crime data. For instance, around O'Hare international airport, it is very likely that the floating and the residential populations differ a lot. After the preprocessing, we obtain a three-way tensor with size $22 \times 16 \times 4$, for 22 districts, 16 years, and 4 seasons, which is visualized in Figure 17.

Our main interest is to understand the geographical structure of the relative crime rates and how the structure changes over the year. A Bayesian model is constructed for this purpose by using the graphical tools under the proposed framework. We define a

[6] The Chicago crime data set can be retrieved from `https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present`.

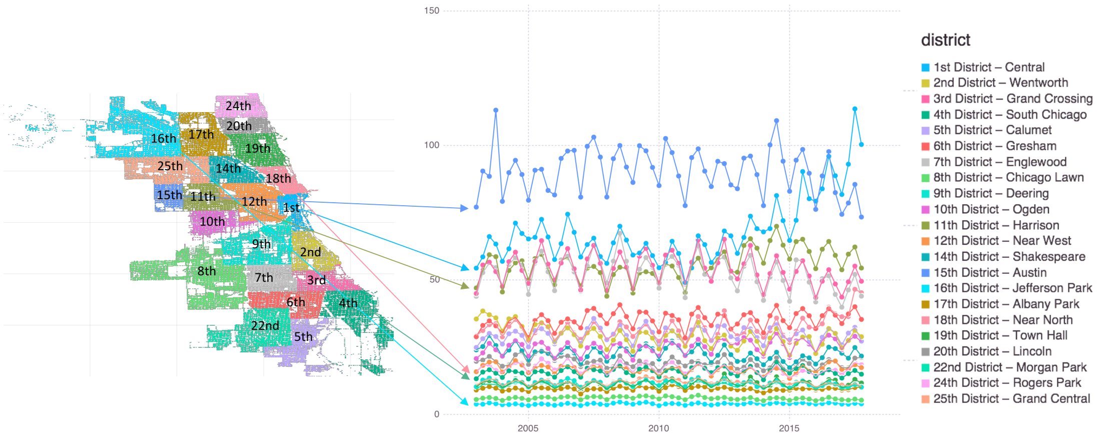


Figure 17: Visualization of the Chicago crime data after preprocessing.

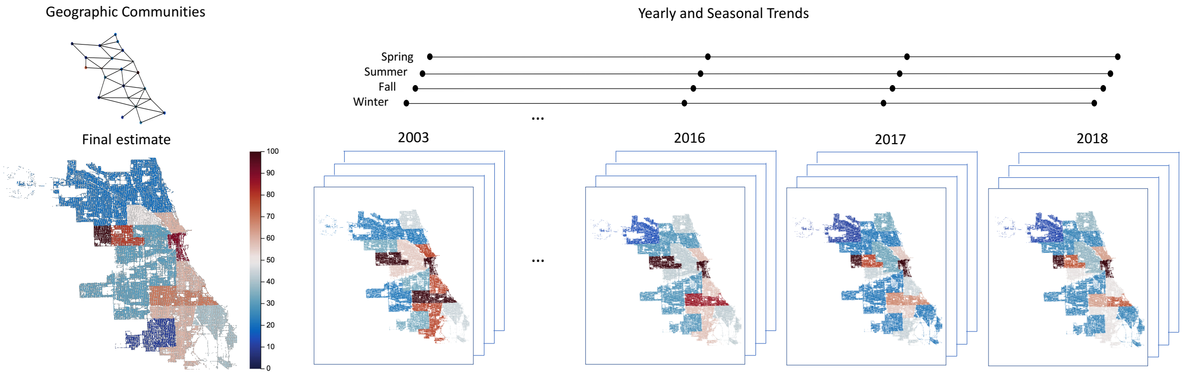


Figure 18: Visualization of Bayesian model selection for the Chicago crime data. (Left) The overall geographical pattern; (Right) Four different patterns from 2003 to 2018.

graph characterizing the geographical effect by $G_1 = (V_1, E_1)$ with $V_1 = \{1, 2, \cdots, 22\}$ and $E_1 = \{(i, j) : \text{the } i\text{th and } j\text{th districts are adjacent}\}$. A graph characterizing the temporal effect is given by $G_2 = (V_2, E_2)$ with $V_2 = \{1, 2, ..., 16\}$ and $E_2 = \{(i, i+1) : i = 1, ..., 15\}$. Then, the $22 \times 16 \times 4$ tensor is modeled by a spike-and-slab Laplacian prior with the base graph $G_1 \square G_2$, in addition to a multivariate extension (Section 4.1) along the dimension of 4 seasons.

The result of the Bayesian model selection for the Chicago crime data is visualized in Figure 18. The geographical structure of the relative crime rates exhibit four different patterns according to the partition $\{2003, 2004, ..., 2015\}$, $\{2016\}$, $\{2017\}$, $\{2018\}$. While geographical compositions of the crimes are similar from 2003 to 2015, our results reveal that the last three years have witnessed dramatic changes. In particular, in these three years, the relative crime rates of Districts 11 and 15 were continuously decreasing, and the relative crime rates of Districts 1 and 18 show the opposite trend. This implies that the overall crime pattern is moving away from historically dangerous areas to downtown areas in Chicago.

Acknowledgments

We thank Matthew Stephens for helpful discussions. Research of CG is supported in part by NSF grant DMS-1712957 and NSF CAREER award DMS-1847590.

Appendix A. Some Basics on Linear Algebra

For a symmetric matrix Γ with rank r, it has an eigenvalue decomposition $\Gamma = UDU^T$ with some orthonormal matrix $U \in \mathcal{O}(p,r) = \{V \in \mathbb{R}^{p\times r} : V^TV = I_r\}$ and some diagonal matrix D whose diagonal entires are all positive. Then, the Moore-Penrose pseudo inverse of Γ is defined by

$$\Gamma^+ = UD^{-1}U^T.$$

Lemma 10 *Consider a symmetric and invertible matrix $R \in \mathbb{R}^{r\times r}$. Then, we have*

$$R = V^T(VR^{-1}V^T)^+V, \tag{70}$$

for any $V \in \mathcal{O}(p,r)$.

Proof To prove (70), we write R as its eigenvalue decomposition $W\Lambda W^T$ for some $W \in \mathcal{O}(r,r)$ and some invertible diagonal matrix Λ. Then, it is easy to see that $VW \in \mathcal{O}(p,r)$, and we thus have

$$(VR^{-1}V^T)^+ = (VW\Lambda^{-1}W^TV^T)^+ = VW\Lambda W^TV^T,$$

and then

$$V^T(VR^{-1}V^T)^+V = V^TVW\Lambda W^TV^TV = W\Lambda W^T = R.$$

■

Lemma 11 *Let $A, B \in \mathbb{R}^{n\times m}$ be matrices of full column rank m (i.e. $m \leq n$). Let Z_A and Z_B span the nullspaces of A and B, respectively. That is, $Z_A, Z_B \in \mathbb{R}^{n\times(n-m)}$ and*

$$A^TZ_A = 0, \quad B^TZ_B = 0.$$

Then, we have

$$I_n - A(B^TA)^{-1}B^T = Z_B(Z_A^TZ_B)^{-1}Z_A^T.$$

Proof Let $C = I_n - A(B^TA)^{-1}B^T$, and then it is easy to check that

$$CZ_B = Z_B, \quad Z_A^TC = Z_A^T, \quad CA = 0, \quad B^TC = 0.$$

Note that the above four equations determine the singular value decomposition of C and are also satisfied by $Z_B(Z_A^TZ_B)^{-1}Z_A^T$, which immediately implies $C = Z_B(Z_A^TZ_B)^{-1}Z_A^T$. ■

Lemma 12 *Suppose for symmetric matrices $S, H \in \mathbb{R}^{p\times p}$, we have $\mathcal{M}([S;H]) = \mathbb{R}^p$, where the notation $\mathcal{M}(\cdot)$ means the subspace spanned by the columns of a matrix. Then, we have*

$$(tS + H)^{-1} \to R(R^THR)^{-1}R^T,$$

as $t \to \infty$, where R is any matrix such that $\mathcal{M}(R)$ is the null space of S.

Proof We first prove the special case of $H = I_p$. Denote the rank of S by r, and then S has an eigenvalue decomposition $S = UDU^T$ for some $U \in \mathcal{O}(p, r)$ and some diagonal matrix D with positive diagonal entries. Since $\mathcal{M}(R)$ is the null space of S, we have $I_p = UU^T + R(R^TR)^{-1}R^T$ by Lemma 11. Then,

$$(tS + I_p)^{-1} = (U(tD + I_r)U^T + R(R^TR)^{-1}R^T)^{-1} = U(tD + I_r)^{-1}U^T + R(R^TR)^{-1}R^T,$$

which converges to $R(R^TR)^{-1}R^T$ as $t \to \infty$. The second equality above follows from $U^TR = 0$. Now we assume a general H of full rank, which means $H = Q^TQ$ for some $Q \in \mathbb{R}^{p\times p}$ that is invertible. Then,

$$(tS + H)^{-1} = Q^{-1}(t(Q^T)^{-1}SQ^{-1} + I_p)^{-1}(Q^T)^{-1}.$$

Since the null space of $(Q^T)^{-1}SQ^{-1}$ is $\mathcal{M}(QR)$, we have

$$(t(Q^T)^{-1}SQ^{-1} + I_p)^{-1} \to QR(R^TQ^TQR)^{-1}R^TQ^T = QR(R^THR)^{-1}R^TQ^T,$$

and therefore $(tS + H)^{-1} \to R(R^THR)^{-1}R^T$. For a general H that is not necessarily full rank, since $\mathcal{M}([S; H]) = \mathbb{R}^p$, $S + H$ is a matrix of full rank. Then,

$$(tS + H)^{-1} = ((t - 1)S + S + H)^{-1} \to R(R^T(S + H)R)^{-1}R^T = R(R^THR)^{-1}R^T,$$

and the proof is complete. ■

Appendix B. Degenerate Gaussian Distributions

A multivariate Gaussian distribution is fully characterized by its mean vector and covariance matrix. For $N(\mu, \Sigma)$ with some $\mu \in \mathbb{R}^p$ and a positive semidefinite $\Sigma \in \mathbb{R}^{p\times p}$, we call the distribution degenerate if $\mathsf{rank}(\Sigma) < p$. Given any Σ such that $\mathsf{rank}(\Sigma) = r < p$, we have the decomposition $\Sigma = AA^T$ for some $A \in \mathbb{R}^{p\times r}$. Therefore, $X \sim N(\mu, \Sigma)$ if and only if

$$X = \mu + AZ, \tag{71}$$

where $Z \sim N(0, I_r)$. The latent variable representation (71) immediately implies that $X - \mu \in \mathcal{M}(A) = \mathcal{M}(\Sigma)$ with probability one.

The density function of $N(\mu, \Sigma)$ is given by

$$(2\pi)^{-r/2}\frac{1}{\sqrt{\det_+(\Sigma)}}\exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^+(x-\mu)\right)\mathbb{I}\{x - \mu \in \mathcal{M}(\Sigma)\}. \tag{72}$$

The formula (72) can be found in Khatri (1968); Songgui and Shein-Chung (1994). Note that the density function (72) is defined with respect to the Lebesgue measure on the subspace $\{x : x - \mu \in \mathcal{M}(\Sigma)\}$. Here, the $\det_+(\cdot)$ is used for the product of all nonzero eigenvalues of a symmetric matrix, and Σ^+ is the Moore-Penrose inverse of the covariance matrix Σ. The two characterizations (71) and (72) of $N(\mu, \Sigma)$ are equivalent to each other.

The property is useful for us to identify whether a formula leads to a well-defined density function of a degenerate Gaussian distribution.

Lemma 13 *Suppose $f(x) = \exp(-\frac{1}{2}(x-\mu)^T\Omega(x-\mu))\mathbb{I}\{x-\mu \in \mathcal{M}(V)\}$ for some $\mu \in \mathbb{R}^p$, some positive semidefinite $\Omega \in \mathbb{R}^{p\times p}$ and some $V \in \mathcal{O}(p,r)$. As long as $\mathcal{M}(\Omega V) = \mathcal{M}(V)$, we have $\int f(x)dx < \infty$, and $f(x)/\int f(x)dx$ is the density function of $N(\mu, \Sigma)$ with $\Sigma = V(V^T\Omega V)^{-1}V^T$.*

Proof Without loss of generality, assume $\mu = 0$. Since $\mathcal{M}(\Omega V) = \mathcal{M}(V)$, $V^T\Omega V$ is an invertible matrix, and thus Σ is well defined. It is easy to see that $\mathcal{M}(V) = \mathcal{M}(\Sigma)$. Therefore, in view of (72), we only need to show

$$x^T\Omega x = x^T(V(V^T\Omega V)^{-1}V^T)^+x,$$

for all $x \in \mathcal{M}(V)$. Since $x = VV^Tx$ for all $x \in \mathcal{M}(V)$, it suffices to show

$$V^T\Omega V = V^T(V(V^T\Omega V)^{-1}V^T)^+V,$$

which is immediately implied by (70) with $R = V^T\Omega V$. The proof is complete. ■

We remark that Lemma 13 also holds for a $V \in \mathbb{R}^{p\times r}$ that satisfies $\mathsf{rank}(V) = r$ but is not necessarily orthonormal. This is because $V(V^T\Omega V)^{-1}V^T = W(W^T\Omega W)^{-1}W^T$ whenever $\mathcal{M}(V) = \mathcal{M}(W)$.

Appendix C. Proofs of Propositions

Proof of Proposition 1

The property of the Laplacian matrix L_γ is standard in spectral graph theory (Spielman, 2007). We apply Lemma 13 with $\mu = 0$, $\Omega = \sigma^{-2}L_\gamma$ and V is chosen arbitrarily from $\mathcal{O}(p, p-1)$ such that $w^TV = 0$. Then, the condition $\mathcal{M}(\Omega V) = \mathcal{M}(V)$ is equivalent to $\mathbb{1}_p^Tw \neq 0$, because $\mathbb{1}_p$ spans the null space of L_γ. Note that (70) immediately implies $VRV^T = VV^T(VR^{-1}V^T)^+VV^T = (VR^{-1}V^T)^+$. We then have

$$\begin{aligned}\det{}_+(V(V^T\Omega V)^{-1}V^T) &= \sigma^{2(p-1)}\det{}_+(V(V^TL_\gamma V)^{-1}V^T) \\ &= \sigma^{2(p-1)}\det{}_+((VV^TL_\gamma VV^T)^+) \\ &= \sigma^{2(p-1)}\frac{1}{\det_+(VV^TL_\gamma VV^T)}.\end{aligned}$$

The proof is complete by realizing that $VV^T = I_p - ww^T/\|w\|^2$ from Lemma 11. ■

Proof of Proposition 3

Recall the incidence matrix $D \in \mathbb{R}^{m\times p}$ defined in Section 2.1. With the new notations $\Phi = D^T\operatorname{diag}(\gamma)D$ and $\Psi = D^T\operatorname{diag}(1-\gamma)D$, we can write $L_\gamma = v_0^{-1}\Phi + v_1^{-1}\Psi$ and $\widetilde{L}_\gamma = v_1^{-1}\Psi$.

We first prove that (23) is well defined. By Lemma 13, it is sufficient to show $\widetilde{V}^TZ_\gamma^T\Psi Z_\gamma\widetilde{V}$ is invertible. This is because $\mathcal{M}([\Phi;\Psi;w]) = \mathbb{R}^p$, and the columns of $Z_\gamma\widetilde{V}$ are all orthogonal to $\mathcal{M}([\Phi;w])$. Therefore, (23) is well defined and its covariance matrix is given by (73).

Now we will show the distribution (8) converges to that of $Z_\gamma\widetilde{\theta}$ with $\widetilde{\theta}$ distributed by (23) as $v_0 \to 0$. Let $\widetilde{V} \in \mathbb{R}^{r\times r-1}$ be a matrix of rank $r-1$ that satisfies $\widetilde{V}^T Z_\gamma^T w = 0$. Then, by Lemma 13, the distribution (23) can be written as

$$\widetilde{\theta} \sim N(0, \sigma^2 \widetilde{V}(\widetilde{V}^T Z_\gamma^T \widetilde{L}_\gamma Z_\gamma \widetilde{V})^{-1}\widetilde{V}^T), \tag{73}$$

which implies

$$Z_\gamma\widetilde{\theta} \sim N(0, \sigma^2 Z_\gamma\widetilde{V}(\widetilde{V}^T Z_\gamma^T \widetilde{L}_\gamma Z_\gamma \widetilde{V})^{-1}\widetilde{V}^T Z_\gamma^T).$$

On the other hand, the distribution (8) can be written as

$$\theta \sim N(0, \sigma^2 V(V^T L_\gamma V)^{-1}V^T),$$

where the matrix $V \in \mathbb{R}^{p\times p-1}$ can be chosen to be any matrix of rank $p-1$ that satisfies $V^T w = 0$. In particular, we can choose V that takes the form of

$$V = [Z_\gamma\widetilde{V}; U],$$

where $U \in \mathcal{O}(p, p-r)$ and U satisfies $U^T w = 0$ and $U^T Z_\gamma\widetilde{V} = 0$. Since both θ and $Z_\gamma\widetilde{\theta}$ are Gaussian random vectors, we need to prove

$$V(V^T L_\gamma V)^{-1}V^T \to Z_\gamma\widetilde{V}(\widetilde{V}^T Z_\gamma^T \widetilde{L}_\gamma Z_\gamma \widetilde{V})^{-1}\widetilde{V}^T Z_\gamma^T, \tag{74}$$

as $v_0 \to 0$. Note that (74) is equivalent to

$$V(v_0^{-1}V^T\Phi V + v_1^{-1}V^T\Psi V)^{-1}V^T \to v_1 Z_\gamma\widetilde{V}(\widetilde{V}^T Z_\gamma^T \Psi Z_\gamma \widetilde{V})^{-1}\widetilde{V}^T Z_\gamma^T, \tag{75}$$

as $v_0 \to 0$. By the definition of Z_γ and the property of graph Laplacian, the null space of Φ is $\mathcal{M}(Z_\gamma)$. By the construction of V, the null space of $V^T\Phi V$ is $\mathcal{M}(V^T Z_\gamma)$. Moreover, we have $\mathcal{M}([V^T\Phi V; V^T\Psi V]) = \mathbb{R}^{p-1}$. Therefore, Lemma 12 implies that

$$(v_0^{-1}V^T\Phi V + v_1^{-1}V^T\Psi V)^{-1} \to v_1 V^T Z_\gamma(Z_\gamma^T V V^T \Psi V V^T Z_\gamma)^{-1} Z_\gamma^T V,$$

and thus

$$V(v_0^{-1}V^T\Phi V + v_1^{-1}V^T\Psi V)^{-1}V^T \to v_1 V V^T Z_\gamma(Z_\gamma^T V V^T \Psi V V^T Z_\gamma)^{-1} Z_\gamma^T V V^T.$$

Since $VV^T Z_\gamma = Z_\gamma\widetilde{V}\widetilde{V}^T Z_\gamma^T Z_\gamma$, we have $\mathcal{M}(VV^T Z_\gamma) = \mathcal{M}(Z_\gamma\widetilde{V})$. This implies

$$VV^T Z_\gamma(Z_\gamma^T V V^T \Psi V V^T Z_\gamma)^{-1} Z_\gamma^T V V^T = Z_\gamma\widetilde{V}(\widetilde{V}^T Z_\gamma^T \Psi Z_\gamma \widetilde{V})^{-1}\widetilde{V}^T Z_\gamma^T,$$

and therefore we obtain (75). The proof is complete. ■

Proof of Proposition 4

We only prove the case with $d = 1$. The general case with $d \geq 2$ follows the same argument with more complicated notation of covariance matrices (such as Kronecker products). By Lemma 13, (33) is the density function of $N(0, \Sigma)$, with

$$\Sigma = \sigma^2 U(U^T L_\gamma U)^{-1}U^T,$$

where

$$L_\gamma = \begin{bmatrix} (v_0^{-1} - v_1^{-1})I_n + v_1^{-1}kI_n & -(v_0^{-1} - v_1^{-1})\gamma - v_1^{-1}\mathbb{1}_{n\times k} \\ -(v_0^{-1} - v_1^{-1})\gamma^T - v_1^{-1}\mathbb{1}_{k\times n} & (v_0^{-1} - v_1^{-1})\gamma^T\gamma + v_1^{-1}nI_k \end{bmatrix}.$$

The matrix U is defined by

$$U = \begin{bmatrix} V & 0_{n\times k} \\ 0_{k\times(n-1)} & I_k \end{bmatrix},$$

and $V \in \mathbb{R}^{n\times(n-1)}$ is a matrix of rank $n-1$ that satisfies $\mathbb{1}_n^T V = 0$. Note that $\theta|\gamma, \sigma^2$ follows $N(0, \Sigma_{[n]\times[n]})$. That is, the covariance matrix is the top $n \times n$ submatrix of Σ. A direct calculation gives

$$\Sigma_{[n]\times[n]} = \sigma^2 V[V^T(A - BC^{-1}B^T)V]^{-1}V^T,$$

where

$$\begin{aligned} A &= (v_0^{-1} - v_1^{-1})I_n + v_1^{-1}kI_n, \\ B &= -(v_0^{-1} - v_1^{-1})\gamma - v_1^{-1}\mathbb{1}_{n\times k}, \\ C &= (v_0^{-1} - v_1^{-1})\gamma^T\gamma + v_1^{-1}nI_k. \end{aligned}$$

Letting $v_1 \to \infty$, we have

$$\Sigma_{[n]\times[n]} \to \sigma^2 v_0 V[V^T(I_n - \gamma(\gamma^T\gamma)^{-1}\gamma^T)V]^{-1}V^T.$$

The existence of $(\gamma^T\gamma)^T$ is guaranteed by the condition that γ is non-degenerate. By Lemma 13, $p(\theta|\gamma, \sigma^2) \propto \prod_{1\le i<l\le n} \exp\left(-\frac{\lambda_{il}\|\theta_i-\theta_l\|^2}{2\sigma^2 v_0}\right)\mathbb{I}\{\mathbb{1}_n^T\theta = 0\}$ is the density function of $N(0, \sigma^2 v_0 V[V^T(I_n - \gamma(\gamma^T\gamma)^{-1}\gamma^T)V]^{-1}V^T)$, which completes the proof. ■

Proof of Proposition 5

We only prove the case with $d = 1$. The general case with $d \geq 2$ follows the same argument with more complicated notation of covariance matrices (such as Kronecker products). The proof is basically an application of Proposition 3. That is, as $v_0 \to 0$, the distribution of $(\theta^T, \mu^T)^T$ weakly converges to that of $Z_\gamma\widetilde{\mu}$. In the current setting, we have

$$Z_\gamma = \begin{bmatrix} \gamma \\ I_k \end{bmatrix}.$$

The random vector $\widetilde{\mu}$ is distributed by (24). Note that the contracted base graph is a complete graph on $\{1, ..., k\}$, and $w_{jl} = n_j + n_l$ in the current setting. The density (24) thus becomes

$$p(\widetilde{\mu}|\gamma, \sigma^2) \propto \prod_{1\le j<l\le k} \exp\left(-\frac{(n_j + n_l)(\widetilde{\mu}_j - \widetilde{\mu}_l)^2}{2\sigma^2 v_1}\right)\mathbb{I}\{\mathbb{1}_n^T\gamma\widetilde{\mu} = 0\}.$$

Finally, the relations $\theta = \gamma\widetilde{\mu}$ and $\mu = \widetilde{\mu}$ lead to the desired conclusion. ■

Proof of Proposition 8

Note that the integration is with respect to the Lebesgue measure on the $(n-1)$-dimensional subspace $\{\theta : \mathbb{1}_n^T\theta = 0\}$. Consider a matrix $V \in \mathbb{R}^{n\times n-1}$ of rank $n-1$ that satisfies $\mathbb{1}_n^T V = 0$, which means that the columns of $[\mathbb{1}_n : V] \in \mathbb{R}^{n\times n}$ form a nondegenerate basis. Then, we can write $d\theta$ in the integral as $\frac{1}{\sqrt{\det(V^TV)}}d(V^T\theta)$. For the Laplacian matrix L_γ that satisfies $\theta^T L_\gamma\theta = \sum_{i=1}^{n-1}\frac{(\theta_{i+1}-\theta_i)^2}{v_0\gamma_i+v_1(1-\gamma_i)}$, we have $\mathbb{1}_n^T L_\gamma = 0$. In particular, we choose V such that its ith column is $V_{*i} = e_i - e_{i+1}$, where e_i is a vector whose ith entry is 1 and 0 elsewhere. Then, we have $L_\gamma = VS_\gamma V^T$ with $S_\gamma = \operatorname{diag}(v_0^{-1}\gamma + v_1^{-1}(1-\gamma))$, and the integral becomes

$$\begin{aligned}
&\int_{\mathbb{1}_n^T\theta=0,\theta_1\leq\ldots\leq\theta_n} 2^{(n-1)}\frac{1}{(2\pi\sigma^2)^{(n-1)/2}}\sqrt{\det_{\mathbb{1}_n}(L_\gamma)}\exp\left(-\frac{1}{2\sigma^2}\theta^T L_\gamma\theta\right)d\theta\\
&= \int_{\theta_1\leq\ldots\leq\theta_n} 2^{(n-1)}\frac{1}{(2\pi\sigma^2)^{(n-1)/2}}\sqrt{\frac{\det_{\mathbb{1}_n}(L_\gamma)}{\det(V^TV)}}\exp\left(-\frac{1}{2\sigma^2}\theta^T VS_\gamma V^T\theta\right)d(V^T\theta)\\
&= \int_{\delta_1\leq 0,\ldots,\delta_n\leq 0} 2^{(n-1)}\frac{1}{(2\pi\sigma^2)^{(n-1)/2}}\sqrt{\frac{\det_{\mathbb{1}_n}(L_\gamma)}{\det(V^TV)}}\exp\left(-\frac{1}{2\sigma^2}\delta^T S_\gamma\delta\right)d\delta\\
&= \int \frac{1}{(2\pi\sigma^2)^{(n-1)/2}}\sqrt{\frac{\det_{\mathbb{1}_n}(L_\gamma)}{\det(V^TV)}}\exp\left(-\frac{1}{2\sigma^2}\delta^T S_\gamma\delta\right)d\delta \qquad (76)\\
&= \sqrt{\frac{\det_{\mathbb{1}_n}(L_\gamma)}{\det(S_\gamma)\det(V^TV)}} \qquad (77)\\
&= 1,
\end{aligned}$$

where the last equality is by $\det_{\mathbb{1}_n}(L_\gamma) = \det_+(VS_\gamma V^T) = \det(S_\gamma)\det(V^TV)$. The equality (76) is by the symmetry of $\exp\left(-\frac{1}{2\sigma^2}\delta^T S_\gamma\delta\right)$, and (77) is by Lemma 13. Finally, Lemma 2 says that

$$\det_{\mathbb{1}_n}(L_\gamma) = n\prod_{i=1}^{n-1}[v_0^{-1}\gamma_i + v_1^{-1}(1-\gamma_i)].$$

This completes the proof. ■

Proof of Proposition 9

We need to calculate

$$\int_{\mathbb{1}_n^T Z_\gamma\widetilde{\theta}=0,\ \widetilde{\theta}_1\leq\cdots\leq\widetilde{\theta}_s}\prod_{l=1}^{s}\exp\left(-\frac{(\widetilde{\theta}_l-\widetilde{\theta}_{l+1})^2}{2\sigma^2 v_1}\right)d\widetilde{\theta},$$

where the integral is taken with respect to the Lebesgue measure on the low-dimensional subspace $\{\widetilde{\theta} : \mathbb{1}_n^T Z_\gamma\widetilde{\theta} = 0\}$. Choose $\widetilde{V}_{*l} = e_l - e_{l+1}$, where $e_l \in \{0,1\}^s$ is a vector whose lth entry is 1 and 0 elsewhere. Then the columns of $[Z_\gamma^T\mathbb{1}_n : (Z_\gamma^T Z_\gamma)^{-1}\widetilde{V}] \in \mathbb{R}^{s\times s}$ form a non-degenerate basis of $\mathbb{R}^s$. This is because

$$Z_\gamma^T\mathbb{1}_n = (n_1,\cdots,n_s)^T, \quad Z_\gamma^T Z_\gamma = \operatorname{diag}(n_1,\cdots,n_s),$$

where n_l is the size of lth cluster. Furthermore, $Z_\gamma^T \mathbb{1}_n$ and $(Z_\gamma^T Z_\gamma)^{-1}\widetilde{V}$ are orthogonal to each other. We write $\widetilde{W} = (Z_\gamma^T Z_\gamma)^{-1}\widetilde{V}$ for simplicity. Then,

$$
\begin{aligned}
&\int_{\mathbb{1}_n^T Z_\gamma \widetilde{\theta}=0,\ \widetilde{\theta}_1 \le \cdots \le \widetilde{\theta}_s} \prod_{l=1}^{s} \exp\left(- \frac{(\widetilde{\theta}_l - \widetilde{\theta}_{l+1})^2}{2\sigma^2 v_1} \right) d\widetilde{\theta} \\
=\ & \frac{1}{\sqrt{\det \widetilde{W}^T \widetilde{W}}} \int_{\widetilde{\theta}_1 \le \cdots \le \widetilde{\theta}_s} \prod_{l=1}^{s} \exp\left(- \frac{(\widetilde{\theta}_l - \widetilde{\theta}_{l+1})^2}{2\sigma^2 v_1} \right) d(\widetilde{W}^T \widetilde{\theta}) \\
=\ & \frac{\det \widetilde{W}^T \widetilde{W} (\widetilde{V}^T \widetilde{W})^{-1}}{\sqrt{\det \widetilde{W}^T \widetilde{W}}} \int_{\widetilde{\theta}_1 \le \cdots \le \widetilde{\theta}_s} \prod_{l=1}^{s-1} \exp\left(- \frac{(\widetilde{\theta}_l - \widetilde{\theta}_{l+1})^2}{2\sigma^2 v_1} \right) d(\widetilde{V}^T \widetilde{\theta}) \\
=\ & \frac{\det \widetilde{W}^T \widetilde{W} (\widetilde{V}^T \widetilde{W})^{-1}}{\sqrt{\det \widetilde{W}^T \widetilde{W}}} \int_{\widetilde{\delta}_1, \cdots, \widetilde{\delta}_{s-1} \le 0} \prod_{l=1}^{s-1} \exp\left(- \frac{\widetilde{\delta}_l^2}{2\sigma^2 v_1} \right) d\widetilde{\delta} \\
=\ & \frac{\sqrt{\det \widetilde{W}^T \widetilde{W}}}{\det \widetilde{V}^T \widetilde{W}} \int_{\widetilde{\delta}_1, \cdots, \widetilde{\delta}_{s-1} \le 0} \prod_{l=1}^{s-1} \exp\left(- \frac{\widetilde{\delta}_l^2}{2\sigma^2 v_1} \right) d\widetilde{\delta} \\
=\ & \frac{(2\pi\sigma^2 v_1)^{(s-1)/2}}{2^{s-1}} \times \frac{\sqrt{\det \widetilde{W}^T \widetilde{W}}}{\det \widetilde{V}^T \widetilde{W}}.
\end{aligned}
$$

The first equality is from the Lebesgue integration on the reduced space and the last equality is by symmetry. The second equality follows from the change of variables formula, because for any $\widetilde{\theta}$ such that $\mathbb{1}_n^T Z_\gamma \widetilde{\theta} = 0$, $\widetilde{\theta} = \widetilde{W} U$ for some U, which leads to $\widetilde{W}^T \widetilde{W} (\widetilde{V}^T \widetilde{W})^{-1} \widetilde{V}^T \widetilde{\theta} = \widetilde{W}^T \widetilde{W} U = \widetilde{W}^T \widetilde{\theta}$. Finally, we observe that

$$
\begin{aligned}
v_1^{(s-1)/2} \sqrt{\det_{Z_\gamma^T \mathbb{1}_n} (Z_\gamma^T \widetilde{L}_\gamma Z_\gamma)} &= \left(\frac{\det(\widetilde{W}^T \widetilde{V} \widetilde{V}^T \widetilde{W})}{\det \widetilde{W}^T \widetilde{W}} \right)^{1/2} \\
&= \left(\frac{(\det \widetilde{V}^T \widetilde{W})^2}{\det \widetilde{W}^T \widetilde{W}} \right)^{1/2} = \frac{\det \widetilde{V}^T \widetilde{W}}{\sqrt{\det \widetilde{W}^T \widetilde{W}}},
\end{aligned}
$$

since $\widetilde{V}\widetilde{V}^T = v_1 \widetilde{L}_\gamma$ is the reduced graph Laplacian and the columns of $\widetilde{W}$ spans the nullspace of $Z_\gamma^T \mathbb{1}_n$. The proof is complete. ■

Appendix D. Proof of Lemma 2

We let $U \in \mathcal{O}(p, p-1)$ be an orthonormal matrix that satisfies $\mathbb{1}_p^T U = 0$, and $V \in \mathcal{O}(p, p-1)$ be an orthonormal matrix that satisfies $w^T V = 0$. Write $I_p - U(V^T U)^{-1} V^T$ as R. Then, $RU = 0$, which implies that R has rank at most one. The facts $Rw = w$ and $\mathbb{1}_p^T R = \mathbb{1}_p^T$, together with Lemma 11, imply that

$$
I_p - U(V^T U)^{-1} V^T = \frac{1}{\mathbb{1}_p^T w} w \mathbb{1}_p^T. \tag{78}
$$

Therefore,

$$\begin{aligned}
\det_w(L_\gamma) &= \det_+(VV^TL_\gamma VV^T) \\
&= \det(V^TL_\gamma V) \\
&= \det(V^TUU^TL_\gamma UU^TV) \\
&= (\det(V^TU))^2\det(U^TL_\gamma U).
\end{aligned} \tag{79}$$

The inequality (79) is because UU^T is a projection matrix to the null space of L_γ. We are going to calculate $\det(V^TU)$ and $\det(U^TL_\gamma U)$ separately. For $\det(V^TU)$, we have

$$\begin{aligned}
1 &= \det\left(\begin{bmatrix} V & \|w\|^{-1}w\end{bmatrix}^T\begin{bmatrix} U & p^{-1/2}\mathbb{1}_p\end{bmatrix}\right) \\
&= \det\left(\begin{bmatrix} V^TU & p^{-1/2}V^T\mathbb{1}_p \\ \|w\|^{-1}w^TU & w^T\mathbb{1}_p/(p^{1/2}\|w\|)\end{bmatrix}\right) \\
&= \det(V^TU)\det(w^T(I_p-U(V^TU)^{-1}V^T)\mathbb{1}_p)/(p^{1/2}\|w\|) \\
&= \det(V^TU)\frac{p^{1/2}\|w\|}{\mathbb{1}_p^Tw},
\end{aligned}$$

and we thus get

$$\big(\det(V^TU)\big)^2 = \frac{\big(\mathbb{1}_p^Tw\big)^2}{p\|w\|^2}. \tag{80}$$

We use (78) for the equality (80).

The calculation of $\det(U^TL_\gamma U)$ requires the Cauchy-Binet formula. For any given matrices $A, B \in \mathbb{R}^{n\times m}$ with $n \le m$, we have

$$\det(AB^T) = \sum_{\{S\subset[m]:|S|=n\}}\det(B_{*S}^TA_{*S}). \tag{81}$$

The version (81) can be found in Tao (2012) and references therein. Let $A = B = U^TD^T(v_0^{-1/2}\operatorname{diag}(\gamma)+v_1^{-1/2}\operatorname{diag}(1-\gamma))$, and we have

$$\det(U^TL_\gamma U) = \sum_{\{S\subset E:|S|=p-1\}}\left(\prod_{(i,j)\in S}\left[v_0^{-1}\gamma_{ij}+v_1^{-1}(1-\gamma_{ij})\right]\right)\det(D_{S*}UU^TD_{S*}^T).$$

Note that $\det(D_{S*}UU^TD_{S*}^T) = \det(U^TD_{S*}^TD_{S*}U)$, and $D_{S*}^TD_{S*}$ is the graph Laplacian of a subgraph of the base graph with the edge set S. Since $|S| = p-1$, S is either a spanning tree ($\det(U^TD_{S*}^TD_{S*}U) = p$) or is disconnected ($\det(U^TD_{S*}^TD_{S*}U) = 0$), we have

$$\det(U^TL_\gamma U) = p\sum_{T\in\operatorname{spt}(G)}\prod_{(i,j)\in T}\left[v_0^{-1}\gamma_{ij}+v_1^{-1}(1-\gamma_{ij})\right]. \tag{82}$$

Therefore, by plugging (80) and (82) into (79), we obtain the desired conclusion.

Algorithm 1 A fast DLPA

```
Input: Initialize $u_1, u_2$ and $z_2$.
  repeat
    $z_1^{new} = (I_{n_1} + L_{q_1})^{-1}(z_2 + u_2)$
    $z_2^{new} = (z_1^{new} + u_1)(I_{n_2} + L_{q_2})^{-1}$
    $u_1^{new} = z_2 + u_2 - z_1^{new}$,  $u_2^{new} = z_1^{new} + u_2 - z_2^{new}$
    $z_1 = z_1^{new}$, $z_2 = z_2^{new}$, $u_1 = u_1^{new}$, $u_2 = u_2^{new}$
  until convergence criteria met
Output: $\theta = z_2$
```

Appendix E. Some Implementation Details

In this section, we present a fast algorithm that solves the M-step (57). To simplify the notation in the discussion, we consider a special case with $X_1 = I_{n_1}$, $X_2 = I_{n_2}$, $w = \mathbb{1}_{n_1}\mathbb{1}_{n_2}^T$ and $\nu = 0$, which is the most important setting that we need for the biclustering problem. In other words, we need to optimize $F(\theta; q_1, q_2)$ over $\theta \in \Theta_w$ for any q_1 and q_2, where

$$\begin{aligned} F(\theta; q_1, q_2) &= \|y - \bar{y}\mathbb{1}_{n_1}\mathbb{1}_{n_2}^T - \theta\|_{\mathrm{F}}^2 + \mathsf{vec}(\theta)^T\left(L_{q_2} \otimes I_{p_1} + I_{p_2} \otimes L_{q_1}\right)\mathsf{vec}(\theta) \quad (83)\\ &= \|y - \bar{y}\mathbb{1}_{n_1}\mathbb{1}_{n_2}^T - \theta\|_{\mathrm{F}}^2 + \left\langle \theta\theta^T, L_{q_1}\right\rangle + \left\langle \theta^T\theta, L_{q_2}\right\rangle. \quad (84)\end{aligned}$$

Algorithm 1 is a Dykstra-like proximal algorithm (DLPA) (Dykstra, 1983) that iteratively solves the optimization problem. It is shown that Algorithm 1 has a provable linear convergence Combettes and Pesquet (2011). If we initialize $u_1 = u_2 = 0_{n_1 \times n_2}$ and $z_2 = y - \bar{y}\mathbb{1}_{n_1}\mathbb{1}_{n_2}^T$ with $\hat{\alpha} = \bar{y}$, then the first two steps of Algorithm 1 can be written as the following update

$$\theta^{\text{new}} = (I_{n_1} + L_{q_1})^{-1}\left(y - \bar{y}\mathbb{1}_{n_1}\mathbb{1}_{n_2}^T\right)(I_{n_2} + L_{q_2})^{-1}. \quad (85)$$

In practice, we suggest using (85) as approximate M-step updates in the first few iterations of the EM algorithm. Then, the full version of Algorithm 1 can be implemented in later iterations to ensure convergence.

Appendix F. Accuracy of Variational Approximation

In this section, we conduct an empirical study of the accuracy of the variational approximation. The variational lower bound we used is

$$\begin{aligned} &\log \sum_{T \in \mathrm{spt}(G)} \prod_{(i,j) \in T} \left[v_0^{-1}\gamma_{ij} + v_1^{-1}(1 - \gamma_{ij})\right] \\ &\geq \sum_{(i,j) \in E} r_{ij} \log\left[v_0^{-1}\gamma_{ij} + v_1^{-1}(1 - \gamma_{ij})\right] + \log|\mathrm{spt}(G)|, \end{aligned}$$

where $r_{ij} = |\mathrm{spt}(G)|^{-1}\sum_{T \in \mathrm{spt}(G)} \mathbb{I}\{(i,j) \in T\}$ is an effective resistance of an edge (i,j). An effective resistance r_e of an edge e measures how important the edge e is in the whole graph, provided that their local resistance is 1. For instance, if (i,j) is the only edge connecting

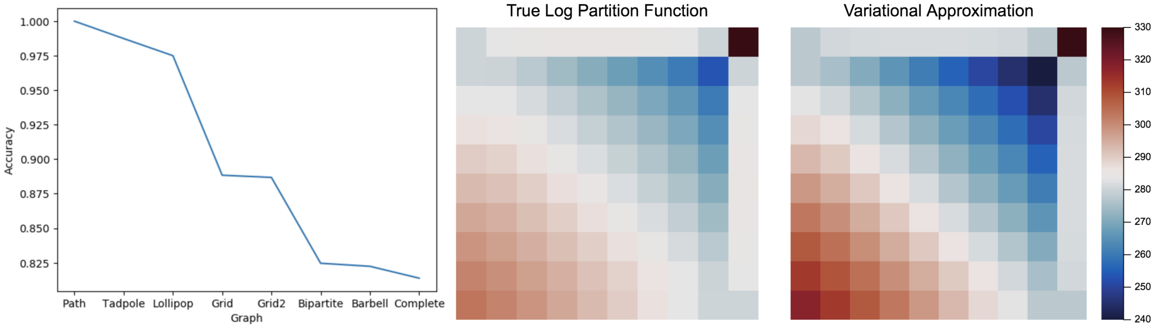


Figure 19: (Left) Accuracy of variational approximation to the normalization constant; (Center and Right) The quality of variational approximation on Grid(10,10).

Tadpole(50,50)	Lollipop(80,20)	Grid(10,10)	Grid(20,5)	Bipartite(80,20)	Barbell(50,50)
$P_{50} \vee C_{50}$	$P_{80} \vee K_{20}$	$P_{10} \otimes P_{10}$	$P_{20} \otimes P_5$	$K_{80,20}$	$K_{50} \vee K_{50}$

Table 7: A List of Graphs Used for Evaluating Quality of Variational Approximation

two mutually exclusive subgraphs containing i and j respectively then $r_{ij} = 1$. In this case, determining whether $\gamma_{ij} = 1$ has a direct effect on separation of i and j, one may want to put larger weights on γ_{ij} in the objective function. In the right hand side of (19), $\log[v_0^{-1}\gamma_{ij} + v_1^{-1}(1-\gamma_{ij})]$ is weighted by the effective resistance r_{ij} of the edge (i, j). This implies that the variational lower bound or the right hand side of the above inequality puts larger weights to more "important" edges. This intuitive explanation can be verified by the following numerical experiments.

We compare the true log partition function

$$f_1(\gamma) = \log \sum_{T \in \mathrm{spt}(G)} \prod_{(i,j)\in T} \left[v_0^{-1}\gamma_{ij} + v_1^{-1}(1-\gamma_{ij})\right]$$

with our variational lower bound

$$f_2(\gamma) = \sum_{(i,j)\in E} r_{ij} \log \left[v_0^{-1}\gamma_{ij} + v_1^{-1}(1-\gamma_{ij})\right] + \log |\mathrm{spt}(G)|$$

for various choices of graphs. To this end, we randomly sample two weighted graphs G_1 and G_2 and then put independent random uniform weights on the edges of G_1 and G_2, respectively. Then we compare the differences between the true normalization constants and the differences between the variational lower bounds. We fix $n = 100$, $v_0 = 10^{-1}$ and $v_1 = 10^3$. Let us write the linear chain graph as P_n and the complete graph as K_n. Let $\vee$ be a graph operator joining two graphs by adding exactly one connecting edge between two graphs. We compare P_n, K_n and the 6 graphs in Table 7. The graph $K_{80,20}$ is sampled under the additional constraint that $\sum_j \gamma_{ij} = 1$.

The left panel of Figure 19 plots the accuracy of approximation, measured by $e^{f_2(\gamma)}/e^{f_1(\gamma)}$ against the complexity of graphs. It shows that the approximation is more accurate for a sparser graph, but even for the complete graph we still obtain an accuracy over 0.8.

The center and the right panels of Figure 19 displays the true log-partition function and the variational approximation when Grid(10,10) is clustered into two parts. To be specific, the true signal $\theta^* \in \mathbb{R}^{10\times 10}$ is defined on the nodes of G and has exactly 2 clusters, the bottom left square of (a, b) and the rest. That is,

$$\mathcal{C}_1^{(a,b)} = \{\theta_{ij} : i \leq a \text{ and } j \leq b\} \quad \mathcal{C}_2^{(a,b)} = \{\theta_{ij} : i > a \text{ or } j > b\}.$$

For instance, if $a = 3$ and $b = 4$, then $\mathcal{C}_1$ has 12 nodes. Let $\gamma^{(a,b)}$ be the corresponding latent structure parameter, and $\gamma_{ij}^{(a,b)} = 1$ if i and j are in the same cluster and $\gamma_{ij}^{(a,b)} = 0$ otherwise. For $a = 1, \cdots, 10$ and $b = 1, \cdots, 10$, we compare the true log-normalization constant $f_1(\gamma^{(a,b)})$ and its variational approximation $f_2(\gamma^{(a,b)})$, and summarize the result in the center and the right panels of Figure 19. The result is displayed by heat-maps, and the values of $f_1(\gamma^{a,b})$ and $f_2(\gamma^{a,b})$ are reported in the (a, b)-th entries of the two heatmaps. The results support our conclusion that the variational approximations are reasonable for various graphical structures.

References

Taylor B Arnold and Ryan J Tibshirani. Genlasso: path algorithm for generalized lasso problems. *R package version*, 1, 2014.

Andrew Barron, Lucien Birgé, and Pascal Massart. Risk bounds for model selection via penalization. *Probability theory and related fields*, 113(3):301–413, 1999.

Daniel Barry and John A Hartigan. A Bayesian analysis for change point problems. *Journal of the American Statistical Association*, 88(421):309–319, 1993.

Arindam Bhattacharjee, William G Richards, Jane Staunton, Cheng Li, Stefano Monti, Priya Vasa, Christine Ladd, Javad Beheshti, Raphael Bueno, and Michael Gillette. Classification of human lung carcinomas by mrna expression profiling reveals distinct adenocarcinoma subclasses. *Proceedings of the National Academy of Sciences*, 98(24):13790–13795, 2001.

Howard D Bondell and Brian J Reich. Simultaneous regression shrinkage, variable selection, and supervised clustering of predictors with oscar. *Biometrics*, 64(1):115–123, 2008.

Leonard Bottolo and Sylvia Richardson. Evolutionary stochastic search for Bayesian model exploration. *Bayesian Analysis*, 5(3):583–618, 2010.

Oleg Burdakov and Oleg Sysoev. A dual active-set algorithm for regularized monotonic regression. *Journal of Optimization Theory and Applications*, 172(3):929–949, 2017.

Y. Cheng and G.M. Church. Biclustering of expression data. In *Proceedings of the eighth international conference on intelligent systems for molecular biology*, pages 93–103, 2000.

Eric C Chi, Genevera I Allen, and Richard G Baraniuk. Convex biclustering. *Biometrics*, 73(1):10–19, 2017.

Patrick L Combettes and Jean-Christophe Pesquet. Proximal splitting methods in signal processing. In *Fixed-point algorithms for inverse problems in science and engineering*, pages 185–212. Springer, 2011.

Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977.

Jean Diebolt and Christian P Robert. Estimation of finite mixture distributions through Bayesian sampling. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 363–375, 1994.

Richard L Dykstra. An algorithm for restricted least squares regression. *Journal of the American Statistical Association*, 78(384):837–842, 1983.

Zhou Fan and Leying Guan. Approximate ℓ_0-penalized estimation of piecewise-constant signals on graphs. *The Annals of Statistics*, 46(6B):3217–3245, 2018.

Uriel Feige and Robert Krauthgamer. Finding and certifying a large hidden clique in a semirandom graph. *Random Structures & Algorithms*, 16(2):195–208, 2000.

Jerome Friedman, Trevor Hastie, and Rob Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1, 2010.

Felix Friedrich, Angela Kempe, Volkmar Liebscher, and Gerhard Winkler. Complexity penalized M-estimation: Fast computation. *Journal of Computational and Graphical Statistics*, 17(1):201–224, 2008.

Chao Gao, Aad W van der Vaart, and Harrison H Zhou. A general framework for Bayes structured linear models. *arXiv preprint arXiv:1506.02174*, 2015.

Chao Gao, Yu Lu, Zongming Ma, and Harrison H Zhou. Optimal estimation and completion of matrices with biclustering structures. *The Journal of Machine Learning Research*, 17 (1):5602–5630, 2016.

Chao Gao, Fang Han, and Cun-Hui Zhang. On estimation of isotonic piecewise constant signals. *The Annals of Statistics, to appear*, 2018.

Edward I George and Robert E McCulloch. Variable selection via Gibbs sampling. *Journal of the American Statistical Association*, 88(423):881–889, 1993.

Edward I George and Robert E McCulloch. Approaches for Bayesian variable selection. *Statistica sinica*, pages 339–373, 1997.

Arpita Ghosh, Stephen Boyd, and Amin Saberi. Minimizing effective resistance of a graph. *SIAM review*, 50(1):37–66, 2008.

Joyee Ghosh and Merlise A Clyde. Rao–blackwellization for Bayesian variable selection and model averaging in linear and binary regression: a novel data augmentation approach. *Journal of the American Statistical Association*, 106(495):1041–1052, 2011.

Gérard Govaert and Mohamed Nadif. Clustering with block mixture models. *Pattern Recognition*, 36(2):463–473, 2003.

Bruce Hajek, Yihong Wu, and Jiaming Xu. Submatrix localization via message passing. *The Journal of Machine Learning Research*, 18(1):6817–6868, 2017.

David Hallac, Jure Leskovec, and Stephen Boyd. Network lasso: clustering and optimization in large graphs. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 387–396. ACM, 2015.

Chris Hans, Adrian Dobra, and Mike West. Shotgun stochastic search for large p regression. *Journal of the American Statistical Association*, 102(478):507–516, 2007.

John A Hartigan. Direct clustering of a data matrix. *Journal of the American Statistical Association*, 67(337):123–129, 1972.

Wilfried Imrich and Sandi Klavzar. *Product Graphs: Structure and Recognition.* Wiley, 2000.

Chinubhai G Khatri. Some results for the singular normal multivariate regression models. *Sankhyā: The Indian Journal of Statistics, Series A*, pages 267–280, 1968.

Mihee Lee, Haipeng Shen, Jianhua Z Huang, and JS Marron. Biclustering via sparse singular value decomposition. *Biometrics*, 66(4):1087–1095, 2010.

Jure Leskovec, Deepayan Chakrabarti, Jon Kleinberg, Christos Faloutsos, and Zoubin Ghahramani. Kronecker graphs: an approach to modeling networks. *Journal of Machine Learning Research*, 11(Feb):985–1042, 2010.

Fan Li and Nancy R Zhang. Bayesian variable selection in structured high-dimensional covariate spaces with applications in genomics. *Journal of the American Statistical Association*, 105(491):1202–1214, 2010.

Oren E Livne and Achi Brandt. Lean algebraic multigrid (lamg): fast graph laplacian linear solver. *SIAM Journal on Scientific Computing*, 34(4):B499–B522, 2012.

László Lovász. Random walks on graphs: a survey. *Combinatorics, Paul erdos is eighty*, 2 (1):1–46, 1993.

Zongming Ma and Yihong Wu. Volume ratio, sparsity, and minimaxity under unitarily invariant norms. *IEEE Transactions on Information Theory*, 61(12):6939–6956, 2015.

Patrick Mair, Kurt Hornik, and Jan de Leeuw. Isotone optimization in R: pool-adjacent-violators algorithm (pava) and active set methods. *Journal of statistical software*, 32(5): 1–24, 2009.

Toby J Mitchell and John J Beauchamp. Bayesian variable selection in linear regression. *Journal of the American Statistical Association*, 83(404):1023–1032, 1988.

Radford M Neal and Geoffrey E Hinton. A view of the EM algorithm that justifies incremental, sparse, and other variants. In *Learning in graphical models*, pages 355–368. Springer, 1998.

Paulino Pérez and Gustavo de Los Campos. Genome-wide regression and prediction with the BGLR statistical package. *Genetics*, 198(2):483–495, 2014.

Sylvia Richardson and Peter J Green. On Bayesian analysis of mixtures with an unknown number of components (with discussion). *Journal of the Royal Statistical Society: series B (statistical methodology)*, 59(4):731–792, 1997.

Veronika Ročková and Edward I George. EMVS: the EM approach to Bayesian variable selection. *Journal of the American Statistical Association*, 109(506):828–846, 2014.

Veronika Ročková and Edward I George. The spike-and-slab lasso. *Journal of the American Statistical Association*, 113(521):431–444, 2018.

Michael J Schell and Bahadur Singh. The reduced monotonic regression method. *Journal of the American Statistical Association*, 92(437):128–135, 1997.

Yiyuan She. Sparse regression with exact clustering. *Electronic Journal of Statistics*, 4: 1055–1096, 2010.

Martin Sill, Sebastian Kaiser, Axel Benner, and Annette Kopp-Schneider. Robust biclustering by sparse singular value decomposition incorporating stability selection. *Bioinformatics*, 27(15):2089–2097, 2011.

Qifan Song and Guang Cheng. Optimal false discovery control of minimax estimator. *arXiv preprint arXiv:1812.10013*, 2018.

Wang Songgui and Chow Shein-Chung. Advanced linear models: Theory and applications, 1994.

Daniel A Spielman. Spectral graph theory and its applications. In *Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium on*, pages 29–38. IEEE, 2007.

Daniel A Spielman and Shang-Hua Teng. Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*, pages 81–90. ACM, 2004.

Matthew Stephens. False discovery rates: a new deal. *Biostatistics*, 18(2):275–294, 2016.

Kean Ming Tan and Daniela M Witten. Sparse biclustering of transposable data. *Journal of Computational and Graphical Statistics*, 23(4):985–1008, 2014.

Terence Tao. *Topics in Random Matrix Theory*, volume 132. American Mathematical Soc., 2012.

Robert Tibshirani, Michael Saunders, Saharon Rosset, Ji Zhu, and Keith Knight. Sparsity and smoothness via the fused lasso. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(1):91–108, 2005.

Ryan J Tibshirani and Jonathan Taylor. The solution path of the generalized lasso. *The Annals of Statistics*, 39(3):1335–1371, 2011.

Ryan J Tibshirani, Holger Hoefling, and Robert Tibshirani. Nearly-isotonic regression. *Technometrics*, 53(1):54–61, 2011.

Martin J Wainwright and Michael I Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.

Gao Wang, Abhishek K Sarkar, Peter Carbonetto, and Matthew Stephens. A simple new approach to variable selection in regression, with application to genetic fine-mapping. *bioRxiv*, page 501114, 2018.

Daniela M Witten and Robert Tibshirani. A framework for feature selection in clustering. *Journal of the American Statistical Association*, 105(490):713–726, 2010.

Wei Biao Wu, Michael Woodroofe, and Graciela Mentz. Isotonic regression: another look at the changepoint problem. *Biometrika*, 88(3):793–804, 2001.

Sheng Xu and Zhou Fan. Iterative alpha expansion for estimating gradient-sparse signals from linear measurements. *arXiv preprint arXiv:1905.06097*, 2019.