Bayesian Analysis (2024) **TBA**, Number TBA, pp. 1–25

Estimating Sparse Direct Effects in Multivariate Regression With the Spike-and-Slab LASSO

Yunyi Shen[*], Claudia Solís-Lemus[†], and Sameer K. Deshpande[‡]

Abstract. The multivariate regression interpretation of the Gaussian chain graph model simultaneously parametrizes (i) the direct effects of p predictors on q outcomes and (ii) the residual partial covariances between pairs of outcomes. We introduce a new method for fitting sparse versions of these models with spike-and-slab LASSO (SSL) priors. We develop an Expectation Conditional Maximization algorithm to obtain sparse estimates of the $p \times q$ matrix of direct effects and the $q \times q$ residual precision matrix. Our algorithm iteratively solves a sequence of penalized maximum likelihood problems with self-adaptive penalties that gradually filter out negligible regression coefficients and partial covariances. Because it adaptively penalizes individual model parameters, our method is seen to outperform fixed-penalty competitors on simulated data. We establish the posterior contraction rate for our model, buttressing our method's excellent empirical performance with strong theoretical guarantees. Using our method, we estimated the direct effects of diet and residence type on the composition of the gut microbiome of elderly adults.

MSC2020 subject classifications: Primary 62F15; secondary 62F12.

Keywords: penalized likelihood, posterior concentration, variable selection, covariance selection, EM algorithm.

1 Introduction

1.1 Motivation: Gut microbiome composition

Between 10 and 100 trillion microorganisms live within each person's lower intestines. These bacteria, fungi, viruses, and other microbes constitute the human gut *microbiome* (Guinane and Cotter, 2013). The composition of human gut microbiome can substantially affect our health and well-being (Shreiner et al., 2015): in addition to playing an integral role in digestion and metabolic processes, microbes living in the gut can mediate immune response to certain diseases (Kamada and Núñez, 2014) and may even influence disease pathogenesis and progression (Wang et al., 2011).

Emerging evidence suggests that the gut microbiome can mediate the effects of diet and medication use on human health (Singh et al., 2017). That is, these factors may first affect the composition of the gut microbiome, which in turn influences health outcomes. Further, these factors can impact the composition of the microbiome in both direct and

[*] Laboratory for Information & Decision Systems. Massachusetts Institute of Technology

[†] Wisconsin Institute for Discovery & Dept. of Plant Pathology, University of Wisconsin–Madison, solislemus@wisc.edu

[‡] Dept. of Statistics, University of Wisconsin–Madison, sameer.deshpande@wisc.edu

https://doi.org/10.1214/24-BA1430

indirect ways. For instance, many antibiotics target and kill certain microbial species, thereby directly affecting the abundances of the targeted species. However, by killing the targeted species, the antibiotics may reduce the overall competition for nutrients, which allows non-targeted species to proliferate. Thus, by directly reducing the abundance of a small number targeted microbes, antibiotics may indirectly increase the abundance of many other non-targeted species.

In Section 6, we re-analyze a dataset Claesson et al. (2012) containing $n = 178$ covariate-response pairs $(\boldsymbol{x}, \boldsymbol{y})$ where $\boldsymbol{x}$ contains measurements of $p = 11$ factors related to diet, medication use, and residence type and $\boldsymbol{y}$ contains the logit-transformed relative abundances of $q = 14$ different microbial taxa. Our main analytic goal is to identify which lifestyle factors directly affect the relative abundances of which taxa.

1.2 Directly modeling direct effects

It is tempting to fit q separate linear models, one for each outcome Y_k. After all, doing so allows us to estimate the average change in each Y_k associated with a change in X_j, keeping all other covariates fixed. However, as we alluded in Section 1.1, there are two mechanisms through which a change in X_j can produce a change in Y_k. First, it is possible that X_j may directly change Y_k itself. More subtly, it is possible that X_j directly induces a change in some other outcome $Y_{k'}$, which in turns induces a change in Y_k (i.e. an indirect effect). Generally speaking, fitting separate linear models to our data estimates combinations of these direct and indirect effects. Unfortunately, without modeling the residual dependence between outcomes, we cannot disentangle direct effects from the indirect effects.

To that end, a natural starting point for our analysis is the multivariate linear regression model that asserts

$$\boldsymbol{y}|B, \Omega, \boldsymbol{x} \sim \mathcal{N}(B^{\top}\boldsymbol{x}, \Omega^{-1}), \tag{1.1}$$

where $B = (\beta_{j,k})$ is a $p \times q$ matrix of regression coefficients and $\Omega = (\omega_{k,k'})$ is a symmetric, positive definite $q \times q$ residual precision matrix. Under this model, $\omega_{k,k'}$, the (k, k') entry of Ω, quantifies the residual partial covariance between outcomes Y_k and $Y_{k'}$ that remains after adjusting for the effects of the covariates. Unfortunately, $\beta_{j,k}$ does not quantify the direct effect of X_j on outcome Y_k. To see this, observe that

$$\beta_{j,k} = \mathbb{E}[Y_k|X_j = x + 1, X_{-j}] - \mathbb{E}[Y_k|X_j = x, X_{-j}]. \tag{1.2}$$

Notice that the expectations in right-hand side of Equation (1.2) do not condition on the values of the other outcomes $Y_{k'}$ for $k' \neq k$. In other words, $\beta_{j,k}$ represents a certain *marginal* association between X_j and Y_k, keeping all other covariates fixed. Specifically, $\beta_{j,k}$ represents a weighted average of (i) X_j's direct effect on Y_k and (ii) X_j's indirect effect, which is induced through X_j's direct effect on other outcomes $Y_{k'}$'s that themselves may be related to Y_k.

Although the model in Equation (1.1) does not directly parametrize the direct effects of interest, we can nevertheless compute them from B and Ω. Letting $\psi_{j,k}$ be the (j, k)

entry of the matrix $\Psi = B\Omega$, it turns out that

$$\psi_{j,k}/\omega_{k,k} = \mathbb{E}[Y_k|X_j = x_j + 1, Y_{-k}, X_{-j}] - \mathbb{E}[Y_k|X_j = x_j, Y_{-k}, X_{-j}]. \tag{1.3}$$

Based on this decomposition, a straightforward approach to estimating direct effects involves first fitting the model in Equation (1.1) and then computing the matrix Ψ.

This analytic plan is unfortunately inadequate for our purposes. For one thing, although neither p nor q is especially large in our application, the total number of parameters ($pq + q(q+1)/2$) exceeds the total sample size n. While we can overcome this challenge by assuming that B and Ω are sparse (and estimating them accordingly), the resulting matrix of scaled direct effects Ψ tends not to be sparse. The combination of a sparse B, sparse Ω, and dense Ψ corresponds to a situation in (i) a single covariate directly affects multiple outcomes but (ii) appears to be associated (in the usual linear regression sense) to very few. In the context of our microbiome applications, a sparse B and dense Ψ would mean that several of the lifestyle factors directly affect the abundance of several taxa but that we would observe very few marginal associations. Such a scenario is implausible in the context of microbiome data; in fact the exact opposite tends to be true, with a small number of direct effects producing several marginal associations (see, e.g., Yassour et al., 2016; Blaser, 2016; Thorpe et al., 2018; Schwartz et al., 2020; Avis et al., 2021; Fishbein et al., 2023).

We instead propose fitting a re-paramatrized version of the model in Equation (1.1):

$$\boldsymbol{y}|\Psi, \Omega, \boldsymbol{x} \sim \mathcal{N}(\Omega^{-1}\Psi^{\top}\boldsymbol{x}, \Omega^{-1}), \tag{1.4}$$

where Ψ and Ω are now assumed to be sparse. Now, we may interpret $\psi_{j,k} \neq 0$ to mean that X_j has a direct effect on Y_k. Furthermore, whenever $\psi_{j,k} = 0$, we can conclude that any marginal correlation between X_j and Y_k is due solely to X_j's direct effects on other outcomes $Y_{k'}$ that are themselves conditionally correlated with Y_k.

1.3 Our contributions

We introduce the chain graph spike-and-slab LASSO (cgSSL) for fitting the model in Equation (1.4) by placing separate spike-and-slab LASSO priors (Ročková and George, 2018) on the entries of Ψ and on the off-diagonal entries of Ω. We derive an efficient Expectation Conditional Maximization algorithm to compute the *maximum a posteriori* (MAP) estimates of Ψ and Ω. We further quantify the uncertainty around this estimate using Newton et al. (2021)'s weighted Bayesian bootstrap. Our algorithm involves solving a sequence of penalized maximum likelihood problems with individualized penalties for each parameter $\psi_{j,k}$ and $\omega_{k,k'}$. In fact, these individualized penalties are *self-adaptive*: the penalties are updated according to the previous iteration's parameter estimates, with smaller (resp. larger) parameter estimates receiving larger (resp. smaller) penalties. In this way, the algorithm automatically and adaptively *learns* the appropriate amount of shrinkage to apply to each parameter. On synthetic data, our algorithm displays excellent support recovery and estimation performance. We further establish the posterior contraction rate for each of $\Psi, \Omega, \Psi\Omega^{-1}$, and $X\Psi\Omega^{-1}$. Our contraction results provide asymptotic justification for MAP estimation and also upper bound the minimax optimal rates of estimating these quantities in the Frobenius norm. To the best of our knowledge, ours are the first posterior contraction results for sparse Gaussian chain graph models with element-wise priors on Ψ and Ω.

In Section 2, we review the Gaussian chain graph model and the spike-and-slab LASSO. We next introduce the cgSSL prior in Section 3.1 and carefully derive our ECM algorithm for finding the MAP in Sections 3.2. We present our asymptotic results in Section 4 before demonstrating the cgSSL's excellent finite sample performance on several synthetic datasets in Section 5. We apply the cgSSL to our motivating gut microbiome data in Section 6. Finally, in Section 7, we outline avenues for future research in spike-and-slab uncertainty quantification and modeling outcomes of mixed type.

2 Background

2.1 Motivating (1.4) and related graphical models

Under the model in Equation (1.4), we have for each $k = 1, \ldots, q$,

$$Y_k | Y_{-k}, X, \Psi, \Omega \sim \mathcal{N}\left(-\omega_{k,k}^{-1} \sum_{k' \neq k} \omega_{k,k'} y_{k'} + \omega_{k,k}^{-1} \sum_{j=1}^{p} \psi_{j,k} X_j, \omega_{k,k}^{-1}\right). \tag{2.1}$$

In this way, the parameters Ψ and Ω directly encode important conditional dependence relationship between the covariates and outcomes. Specifically, if $\psi_{j,k} = 0$, we can conclude from Equation (2.1) that Y_k is conditionally independent from X_j given all other covariates and outcomes. And if $\omega_{k,k'} = 0$, then Y_k is conditionally independent of $Y_{k'}$, given all other covariates and outcomes. Consequently, despite its somewhat complicated notation, we can represent the conditional dependencies encoded by the model in Equation (1.4) with a simple graphical model.

Specifically, we construct a graph with $p + q$ vertices, one for each covariate X_j and outcome Y_k. We then draw a directed edges from X_j's vertex to Y_k's vertex whenever $\psi_{j,k} \neq 0$. We additionally draw two directed edges (or, equivalently, a single bi-directed edge) between Y_k's and $Y_{k'}$'s vertices whenever $\omega_{k,k'} \neq 0$. Figure 1 shows a cartoon illustration of such a graph with $p = 3$ covariates and $q = 4$ outcomes.

The graph so constructed is a chain graph and is faithful to the model in Equation (1.4) under Cox and Wermuth (1993)'s multivariate regression (MVR) interpretation of chain graph; see Sonntag and Peña (2015) for a comparison of different chain graph interpretations. Based on this interpretation, and following the examples of McCarter and Kim (2014) and Shen and Solís-Lemus (2021), we will refer to the model in Equation (1.4) as the Gaussian chain graph model.

McCarter and Kim (2014) proposing fitting sparse Gaussian chain graph models by maximizing a penalized log-likelihood. They specifically introduced homogeneous L_1 penalties on the entries of Ψ and Ω and used cross-validation to set the penalty parameters for Ψ and Ω. Shen and Solís-Lemus (2021) developed a Bayesian version

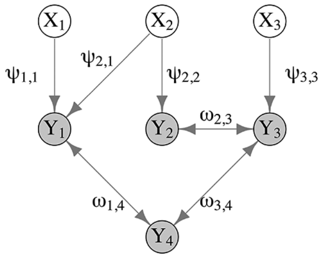


Figure 1: Cartoon illustrations of a Gaussian chain graph model with $p = 3$ covariates and $q = 4$ outcomes. Edges indicate conditional dependence. Edge labels correspond to non-zero parameters in Equation (1.4).

of that chain graphical LASSO and put a gamma prior on the penalty parameters. In this way, they automatically learned the degree to which each $\psi_{j,k}$ and $\omega_{k,k'}$ should be shrunk to zero. Although these papers differ in how they determined the appropriate amount of penalization, both deployed a single fixed penalty on all entries in Ψ and a single fixed penalty on all entries in Ω. With fixed penalties, larger parameter estimates are shrunk towards zero as aggressively as smaller estimates, which can introduce substantial estimation bias.

Fitting a sparse Gaussian chain graph model is related to the covariate-adjusted Gaussian graphical model problem (see, e.g., Consonni et al., 2017; Ni et al., 2019, and references therein). In that problem, interest lies in uncovering the conditional dependencies between q outcomes that remain after controlling for p covariates. Typically, this is done by fitting a sparse version of the marginal model in Equation (1.1) and examining the support of Ω (see, e.g., Cai et al., 2013; Chen et al., 2016, 2018). Within the context of Figure 1, those works focus on detecting Y–Y edges, while we are primarily interested in detecting X–Y edges. As we alluded to in Section 1.2, although it is possible to estimate direct effects by first estimating B and Ω, doing so generally results in a very dense Ψ that is at odds with our expectations, with one important exception: whenever the j-th row of B contains all zeros, the j-th row of Ψ will as well.

Consonni et al. (2017) introduced an Objective Bayes approach to fitting (1.1) with a row-sparse B. Although their method can yield sparse Ψ, it makes the restrictive assumption that each covariate directly affects all or none of the outcomes. In contrast, our proposed procedure places no restrictions on the support of Ψ, allowing covariates to directly affect all, none, or some of the outcomes.

2.2 Spike-and-slab variable selection & asymptotics

The spike-and-slab prior originally comprised a mixture of a point mass at 0 (the "spike") and a uniform distribution over a wide interval (the "slab"; Mitchell and Beauchamp, 1988). George and McCulloch (1993) introduced a continuous relaxation of the original prior, respectively replacing the point mass spike and uniform slab distributions with zero-mean Gaussians with small and large variances. Intuitively, the spike generates the "essentially negligible" model parameters while the slab generates the "significant" parameters values. In high dimensional problems, spike-and-slab priors often produce extremely multimodal posteriors, which render many Markov chain Monte Carlo strategies computationally prohibitive.

In response, Ročková and George (2014) introduced EMVS, a fast Expectation Maximization (EM; Dempster et al., 1977) algorithm targeting the *maximum a posteriori* (MAP) estimate of the regression parameters. They later extended EMVS, which used Gaussian spike and slab distributions, to use Laplacian spike and slab distributions in Ročková and George (2018). The resulting spike-and-slab LASSO (SSL) procedure demonstrated excellent empirical performance. The SSL algorithm solved a sequence of maximum likelihood problems with adaptive L_1-penalties that shrink larger parameter estimates to zero less aggressively than smaller parameter estimates.

Ročková and George (2014)'s general EM technique for maximizing spike-and-slab posteriors has been successfully applied to many problems. For instance, Tang et al. (2017) deployed the SSL to fit sparse generalized linear models while Bai et al. (2020) introduced a grouped version of the SSL that adaptively shrinks groups of parameter values towards zero. Beyond single-outcome regression, continuous spike-and-slab priors have been used to estimate sparse Gaussian graphical models (Li et al., 2019; Gan et al., 2019a,b), sparse factor models (Ročková and George, 2016), and to biclustering (Moran et al., 2021). Deshpande et al. (2019) introduce a multivariate SSL for estimating B and Ω in the marginal regression model in Equation (1.1). In each extension, the adaptive penalization esulted in superior support recovery and parameter estimation compared to fixed penalty methods.

Ročková and George (2018) proved that, under mild regularity conditions, the posterior induced by the SSL prior in high-dimensional, single-outcome linear regression contracts at a near minimax-optimal rate as $n \to \infty$. Bai et al. (2020) extended these results to the group SSL posterior with an unknown variance. In the context of Gaussian graphical models, Gan et al. (2019a) showed that the MAP estimator corresponding to placing spike-and-slab LASSO priors on the off-diagonal elements of a precision matrix is consistent. They did not, however, establish the contraction rate of the posterior. Ning et al. (2020) showed that the joint posterior distribution of (B, Ω) in the multivariate regression model in Equation (1.1) concentrates when using a group spike-and-slab prior with Laplace slab and point mass spike on B and a carefully selected prior on the eigendecomposition of Ω^{-1}. However, the asymptotic properties of the posterior formed by placing SSL priors on the entries of Ψ and Ω have not yet been established.

3 Introducing the cgSSL

3.1 The cgSSL prior

To quantify the prior belief that many entries in Ψ are essentially negligible, we model each $\psi_{j,k}$ as having been drawn either from a spike distribution, which is sharply concentrated around zero, or a slab distribution, which is much more diffuse. More specifically, we take the spike distribution to be Laplace(λ_0) and the slab distribution to be

Laplace(λ_1), where $0 < \lambda_1 \ll \lambda_0$ are fixed positive constants. We further let $\theta \in [0,1]$ be the prior probability that each $\psi_{j,k}$ is drawn from the slab and model the $\psi_{j,k}$'s as conditionally independent given θ. The prior density for Ψ, conditional on θ, is

$$\pi(\Psi|\theta) = \prod_{j=1}^{p}\prod_{k=1}^{q}\left(\frac{\theta\lambda_1}{2}e^{-\lambda_1|\psi_{j,k}|} + \frac{(1-\theta)\lambda_0}{2}e^{-\lambda_0|\psi_{j,k}|}\right). \tag{3.1}$$

Since Ω is symmetric, it is enough to specify a prior on the entries $\omega_{k,k'}$ where $k \le k'$. To this end, we begin by placing an entirely analogous spike-and-slab prior on the off-diagonal entries. That is, for $k < k'$, we model each $\omega_{k,k'}$ as being drawn from a Laplace(ξ_1) with probability $\eta \in [0,1]$ or a Laplace(ξ_0) with probability $1-\eta$, where $0 < \xi_1 \ll \xi_0$. We similarly model each $\omega_{k,k'}$ as conditionally independent given η and place independent exponential Exp(ξ_1) priors on the diagonal entries of Ω. We truncate the resulting distribution of $\Omega|\theta$ to the positive definite cone, yielding the prior density

$$\begin{aligned}\pi(\Omega|\eta) \propto &\left(\prod_{1\le k<k'\le q}\left[\frac{\eta\xi_1}{2}e^{-\xi_1|\omega_{k,k'}|} + \frac{(1-\eta)\xi_0}{2}e^{-\xi_0|\omega_{k,k'}|}\right]\right) \\ &\times\left(\prod_{k=1}^{q}e^{-\xi_1\omega_{k,k}}\right)\times\mathbb{1}(\Omega \succ 0).\end{aligned} \tag{3.2}$$

Note that the truncation implicitly introduces dependence between the entries in Ω. In Section S1.5 of the Supplementary Materials (Shen et al., 2024), we demonstrate that the truncated prior still marginally shrinks every $\omega_{k,k'}$ towards zero. We have also observed empirically that, at least for small q, the marginal prior of $\omega_{k,k'}$ tends to be more concentrated around zero after truncation than before truncation (see Figure S1 in the Supplementary Materials (Shen et al., 2024)).

The quantities $1-\theta$ and $1-\eta$ respectively capture the proportion of essentially negligible entries in Ψ and Ω. We specify independent Beta priors for θ and η: $\theta \sim$ Beta(a_θ, b_θ) and $\eta \sim$ Beta(a_η, b_η), where $a_\theta, b_\theta, a_\eta, b_\eta > 0$ are fixed positive constants.

3.2 MAP estimation and uncertainty quantification

MAP estimation

We follow Ročková and George (2018) and approximate the *maximum a posteriori* (MAP) estimate of $(\Psi, \theta, \Omega, \eta)$. Throughout, we assume that the columns of X are centered and scaled to have norm $\sqrt{n}$. For $\Omega \succ 0$, the log posterior density is, up to an additive constant,

$$\begin{aligned}\log\pi(\Psi,\theta,\Omega,\eta|\boldsymbol{y}) = \frac{n}{2}\times\log|\Omega| - \frac{1}{2}\mathrm{tr}\left(\left(\boldsymbol{Y}-X\Psi\Omega^{-1}\right)^{\top}\Omega\left(\boldsymbol{Y}-X\Psi\Omega^{-1}\right)\right) \\ + \sum_{k=1}^{q}\sum_{j=1}^{p}\log\left(\theta\lambda_1 e^{-\lambda_1|\psi_{j,k}|} + (1-\theta)\lambda_0 e^{-\lambda_0|\psi_{j,k}|}\right)\end{aligned}$$

$$
\begin{aligned}
&+\sum_{k=1}^{q}\left[-\xi_1\omega_{k,k}+\sum_{k'=k+1}^{q}\log\left(\eta\xi_1 e^{-\xi_1|\omega_{k,k'}|}+(1-\eta)\xi_0 e^{-\xi_0|\omega_{k,k'}|}\right)\right]\\
&+(a_\theta-1)\log(\theta)+(b_\theta-1)\log(1-\theta)\\
&+(a_\eta-1)\log(\eta)+(b_\eta-1)\log(1-\eta),
\end{aligned}
$$

where the first line is the log-likelihood implied by the model in Equation (1.4).

Optimizing $\log\pi(\Psi,\theta,\Omega,\eta|\boldsymbol{Y})$ directly is complicated by the non-concavity of $\log\pi(\Omega|\eta)$ (i.e. the term in the third line above). Instead, we iteratively optimize a surrogate objective using an EM-like algorithm. To motivate this approach, observe that we can obtain the prior density $\pi(\Omega|\eta)$ in Equation (3.2) by marginalizing an *augmented* prior

$$\pi(\Omega|\eta)=\int\pi(\Omega|\boldsymbol{\delta})\pi(\boldsymbol{\delta}|\eta)d\boldsymbol{\delta}$$

where $\boldsymbol{\delta}=\{\delta_{k,k'}:1\le k<k'\le q\}$ is a collection of $q(q-1)/2$ i.i.d. Bernoulli(η) variables,

$$
\begin{aligned}
\pi(\Omega|\boldsymbol{\delta})\propto&\left(\prod_{1\le k<k'\le q}\left(\xi_1 e^{-\xi_1|\omega_{k,k'}|}\right)^{\delta_{k,k'}}\left(\xi_0 e^{-\xi_0|\omega_{k,k'}|}\right)^{1-\delta_{k,k'}}\right)\\
&\times\left(\prod_{k=1}^{q}e^{-\xi_1\omega_{k,k}}\right)\times\mathbb{1}(\Omega\succ 0),
\end{aligned}
$$

and $\delta_{k,k'}$ encodes whether $\omega_{k,k'}$ is drawn from the slab ($\delta_{k,k'}=1$) or the spike ($\delta_{k,k'}=0$).

The above marginalization immediately suggests an EM algorithm: rather than optimize $\log\pi(\Psi,\theta,\Omega,\eta|\boldsymbol{Y})$ directly, we can iteratively optimize a surrogate objective formed by marginalizing the augmented log posterior density. That is, starting from some initial guess $(\Psi^{(0)},\theta^{(0)},\Omega^{(0)},\eta^{(0)})$, for $t>1$, the t^{th} iteration of our algorithm consists of two steps. In the first step, we compute the surrogate objective

$$F^{(t)}(\Psi,\theta,\Omega,\eta)=\mathbb{E}_{\boldsymbol{\delta}|\cdot}[\log\pi(\Psi,\theta,\Omega,\eta,\boldsymbol{\delta}|\boldsymbol{y})|\Psi{=}\Psi^{(t-1)},\theta{=}\theta^{(t-1)},\Omega{=}\Omega^{(t-1)},\eta{=}\eta^{(t-1)}],$$

where the expectation is taken with respect to the conditional posterior distribution of the indicators $\boldsymbol{\delta}$ given the current value of $(\Psi,\theta,\Omega,\eta)$. Then in the second step, we maximize the surrogate objective and set $(\Psi^{(t)},\theta^{(t)},\Omega^{(t)},\eta^{(t)})=\arg\max F^{(t)}(\Psi,\theta,\Omega,\eta)$. We defer closed form expressions for the log densities of the augmented posterior (i.e., $\log\pi(\Psi,\theta,\Omega,\eta,\boldsymbol{\delta}|\boldsymbol{Y})$) and the surrogate objective function (i.e., $F^{(t)}(\Psi,\theta,\Omega,\eta)$) to Equations S1.5 and S1.6 in the Supplementary Materials (Shen et al., 2024).

Given Ω and η, the indicators $\delta_{k,k'}$ are conditionally independent, making it simple to derive a closed form expression for the surrogate objective $F^{(t)}$. Unfortunately, maximizing $F^{(t)}$ is still difficult. Consequently, we carry out two conditional maximizations, first optimizing with respect to (Ψ,θ) while holding (Ω,η) fixed, and then optimizing with respect to (Ω,η) while holding (Ψ,θ) fixed. That is, in the second step of each iteration of our algorithm, we set

$$(\Psi^{(t)}, \theta^{(t)}) = \underset{\Psi,\theta}{\arg\max}\ F^{(t)}(\Psi, \theta, \Omega^{(t-1)}, \eta^{(t-1)}) \tag{3.3}$$

$$(\Omega^{(t)}, \eta^{(t)}) = \underset{\Omega,\eta}{\arg\max}\ F^{(t)}(\Psi^{(t)}, \theta^{(t)}, \Omega, \eta). \tag{3.4}$$

In summary, we propose finding the MAP estimate of $(\Psi, \theta, \Omega, \eta)$ using an Expectation Conditional Maximization (ECM; Meng and Rubin, 1993) algorithm.

The objective function $F^{(t)}(\Psi, \theta, \Omega^{(t-1)}, \eta^{(t-1)})$ in Equation (3.3) can be written as the sum of a function of Ψ alone and a function of θ alone. So we separately compute $\Psi^{(t)}$ and $\theta^{(t)}$ while fixing $(\Omega, \eta) = (\Omega^{(t-1)}, \eta^{(t-1)})$. The objective function in Equation (3.4) is similarly separable and we separately compute $\Omega^{(t)}$ and $\eta^{(t)}$ while fixing $(\Psi, \theta) = (\Psi^{(t)}, \theta^{(t)})$. Computing $\theta^{(t)}$ and $\eta^{(t)}$ is relatively straightforward: we compute $\theta^{(t)}$ using Newton's method and there is a closed form expression for $\eta^{(t)}$.

The main computational challenge lies in computing each of $\Psi^{(t)}$ and $\Omega^{(t)}$ conditionally given all other parameters. At a high level, we compute $\Psi^{(t)}$ with a cyclical coordinate descent algorithm that blends soft- and hard-thresholding. Essentially, each entry $\psi_{j,k}$ is thresholded at a level determined by the conditional posterior probability that $\psi_{j,k}$ was drawn from the slab distribution; see Section S1.3 of the Supplementary Materials (Shen et al., 2024) for details. We compute $\Omega^{(t)}$ by solving an optimization problem that is similar to the graphical LASSO (GLASSO; Friedman et al., 2008) objective but includes additional trace terms. We solve that problem by forming a quadratic approximation of the objective and following a Newton direction for a carefully chosen step size. See Sections S1.4 and S2 of the Supplementary Materials (Shen et al., 2024) for the detailed derivation of the algorithm used to update Ω and a proof that the algorithm converges to the unique optimum.

Implementation considerations

The ability of the proposed ECM algorithm to identify the MAP critically depends on two sets of hyperparameters and the initial estimate $\Psi^{(0}$ and $\Omega^{(0)}$. The first set of hyperparameters consists of the spike and slab penalties $\lambda_0, \lambda_1, \xi_0$ and ξ_1. The second set, containing $a_\theta, b_\theta, a_\eta$, and b_η, encode our initial beliefs about the overall proportion of non-negligible entries in Ψ and Ω. In this section, we recommend default hyperparameter settings and sketch a particular path-following scheme that provides good initialization for the ECM algorithm. We have found these recommendations to work very well in practice and present a systematic hyperparameter sensitivity analysis in Section 3.1 of the Supplementary Materials (Shen et al., 2024).

It is initially tempting to run our ECM algorithm with very small slab penalties λ_1 and ξ_1 and very large spike penalties λ_0 and ξ_0 so that the slabs cover a wide range of non-zero parameter values while the spikes are supported only on narrow ranges of extremely small parameter values. Unfortunately, our algorithm was quite sensitive to initialization with such choices. In fact, in early experiments with synthetic data, with very large spike and very small slab penalties, the algorithm tended to estimate Ψ with a zero matrix and Ω with a diagonal matrix with very small diagonal entries, unless it was initialized close to the true data-generating parameter values. Such initialization is, of course, impossible in practice. To overcome this challenge, rather than run cgSSL with a single set of spike and slab penalties, we run our ECM algorithm along sequences of increasing values of the spike-penalties using warm-starts.

Specifically, we fix the slab penalties λ_1 and ξ_1 and specify grids of L increasing spike penalties $\mathcal{I}_\lambda = \{\lambda_0^{(1)} < \cdots < \lambda_0^{(L)}\}$ and $\mathcal{I}_\xi = \{\xi_0^{(1)} < \cdots < \xi_0^{(L)}\}$. We then run cgSSL with warm-starts for each combination of spike penalties, yielding a set of posterior modes $\{(\Psi^{(s,t)}, \theta^{(s,t)}, \Omega^{(s,t)}, \eta^{(s,t)})\}$ indexed by the choices $(\lambda_0^{(s)}, \xi_0^{(t)})$. To warm-start the estimation of the mode corresponding to $(\lambda_0^{(s)}, \xi_0^{(t)})$, we first compute the modes for $(\lambda_0^{(s-1)}, \xi_0^{(t-1)})$, $(\lambda_0^{(s)}, \xi_0^{(t-1)})$ and $(\lambda_0^{(s-1)}, \xi_0^{(t)})$. Then we initialize the ECM algorithm from the mode with highest density (computed using $(\lambda_0^{(s)}, \xi_0^{(t)})$).

Our path-following strategy mirrors one first introduced in Ročková and George (2014) and subsequently deployed by many others (see, e.g., Ročková and George, 2016, 2018; Moran et al., 2019, 2021). In fact, our strategy exactly matches the one used by Deshpande et al. (2019), who fit sparse versions of the model in Equation (1.1) by placing spike-and-slab LASSO priors on the $\beta_{j,k}$'s and $\omega_{k,k'}$'s. Taking a cue from that literature, we refer to our strategy as *dynamic posterior exploration* (DPE).

It is important to stress that the goal of DPE is not to tune or optimize hyperparameters. Instead, we use DPE to identify a good initialization from which to launch our algorithm with large spike penalties. It specifically does so by computing L^2 posterior modes, one for each combination of spike penalties. We can additionally think of DPE as passing an initial estimate of Ψ and Ω through a sequence of increasingly discerning filters. As the spike penalties increase, DPE filters out more and more negligible parameter values, producing sparser and sparse estimates of Ψ and Ω.

In practice, we recommend setting $\lambda_1 = 1$ and $\xi_1 = 0.01n$ and letting $\mathcal{I}_\lambda$ contain ten evenly spaced values ranging from 10 to n and $\mathcal{I}_\xi$ contain ten evenly spaced values from $0.1n$ to n. We further recommend reporting the final mode computed by DPE, corresponding to the largest spike penalties, as a final parameter estimate. With these choices, the sequence of posterior modes computed by DPE appeared to stabilize in the underlying parameter space. That is, for large penalty values $\lambda_0^{(s)}, \lambda_0^{(s')}, \xi_0^{(t)}$, and $\xi_0^{(t')}$, we have $(\Psi^{(s,t)}, \theta^{(s,t)}, \Omega^{(s,t)}, \eta^{(s,t)}) \approx (\Psi^{(s',t')}, \theta^{(s',t')}, \Omega^{(s',t')}, \eta^{(s',t')})$; see Figure S2 in the Supplementary Materials (Shen et al., 2024) for an illustration.

While experimenting with different penalty choices, we observed that when the spike and slab penalties were approximately equal and small, our ECM algorithm would often return very dense estimates of Ψ and diagonal estimates of Ω with large diagonal entries. Essentially, when the spike and slab distributions are not too different and when neither encourages strong shrinkage, our ECM algorithm tended to overfit the data using a dense Ψ, leaving very little residual variation to be quantified with Ω. We found that we could detect such pathological behavior by examining the condition number of the matrix $Y\Omega - X\Psi$. To avoid propagating dense Ψ's and diagonal Ω's through DPE, we terminate our ECM algorithm early whenever the condition number of $Y\Omega - X\Psi$ exceeds $10n$. We then set the corresponding $\Psi^{(s,t)} = 0_{p\times q}$ and $\Omega^{(s,t)} = I_{q\times q}$ and continue the dynamic exploration from that point. Though it is not foolproof, we have found this heuristic to work well in practice. We also note that Moran et al. (2019) utilized a similar early termination strategy in the single-outcome high-dimensional linear regression setting with unknown variance.

To further discourage our ECM algorithm from over-fitting the data with a dense Ψ, we recommend the default priors $\theta \sim \mathrm{Beta}(1, pq)$ and $\eta \sim \mathrm{Beta}(1, q)$. These priors concentrate probability around models with very few direct covariate effects on the outcomes and very few conditional dependencies between outcomes, after adjusting for the covariates. These choices mirror similar prior choices made by Ročková and George (2018) and Deshpande et al. (2019).

To summarize, we recommend setting $a_\theta = 1, b_\theta = pq, a_\eta = 1$, and $b_\eta = q$, and running DPE with $\lambda_1 = 1$ and $\xi_1 = 0.01n$ and letting $\mathcal{I}_\lambda$ contain ten evenly spaced values ranging from 10 to n and $\mathcal{I}_\xi$ contain ten evenly spaced values from $0.1n$ to n. We implemented these choices as the default options in the R (R Core Team, 2022) package **mSSL**, which is available at `https://github.com/YunyiShen/mSSL`. In Section S3.1 of the Supplementary Materials (Shen et al., 2024), we assess the sensitivity of our DPE implementation to (i) the range of spike penalty values in the grids $\mathcal{I}_\lambda$ and $\mathcal{I}_\xi$; (ii) the size of the grids $\mathcal{I}_\lambda$ and $\mathcal{I}_\xi$; and the choice of prior for θ and η. We found that the final point estimates returned by our default implementation displayed better recovered the supports of Ψ and Ω than those found with larger or smaller penalty values. We additionally did not observe much sensitivity to the number of spike penalties in the grids $\mathcal{I}_\lambda$ and $\mathcal{I}_\xi$ nor to the choice of priors for θ and η.

Uncertainty quantification via the weight Bayesian bootstrap

The cgSSL posterior distribution is not log-concave, rendering efficient MCMC or importance sampling computationally prohibitive. Newton et al. (2021)'s weighted Bayesian bootstrap, on the other hand, offers a computationally practical and embarrassingly parallel alternative. At a high-level, the procedure works by repeatedly solving a MAP estimation problem that randomly re-weights every observation's contribution to the log-likelihood and the log-prior density. We specifically use the following two-step procedure. In the first step, we run cgSSL with DPE and obtain point estimates $\hat{\Psi}, \hat{\theta}, \hat{\Omega}$, and $\hat{\eta}$. Then in the second step, we repeatedly solve the single optimization problem

$$\operatorname*{arg\,min}_{\Psi,\Omega,} \left\{ \sum_{i=1}^{n} w_i \ell_i(\Psi,\Omega) + w_0 \left[\log \pi_{\psi}^{(L)}(\Psi) + \log \pi_{\omega}^{(L)}(\Omega) \right] \right\}, \tag{3.5}$$

where $\boldsymbol{w} = (w_0, w_1, \ldots, w_n)$ is a vector of independent Gamma(1, 1) weights, $\pi^{(L)}(\Psi)$ is the conditional prior density of $\Psi|\theta = \hat{\theta}^{(L,L)}$ with $\lambda_0 = \lambda_0^{(L)}$, and $\pi^{(L)}(\Omega)$ is analogously defined. Note, we do not re-run the full DPE procedure to generate each bootstrap sample. Instead, we fix the values of θ and η and also warm-start our optimization from the stabilized Ψ and Ω estimates.

4 Asymptotic theory of cgSSL

If the Gaussian chain graph model in Equation (1.4) is well-specified — that is, if our data are truly generated according to the model — will the posterior distribution of Ψ and Ω collapse to a point-mass at the true data generating parameters as $n \to \infty$? In this section, we answer the question affirmatively: under some mild assumptions and with some slight modifications, the cgSSL posterior concentrates around the truth. We further establish the rate of concentration, which quantifies the speed at which the posterior distribution shrinks to the true data generating parameters. We begin by briefly reviewing our general proof strategy before precisely stating our assumptions and results. Proofs of our main results are available in Section S5 of the Supplementary Materials (Shen et al., 2024).

4.1 Proof strategy

Following Ning et al. (2020) and Bai et al. (2020), we first showed that the posterior of (Ψ, Ω) concentrates in log-affinity. Posterior concentration of the individual parameters followed as a consequence. To show that the posterior concentrates in log-affinity, we verified the three conditions of Theorem 8.23 of Ghosal and van der Vaart (2017). First, we confirmed that the cgSSL prior places enough prior probability mass in small neighborhoods around every possible choice of (Ψ, Ω). This was done by verifying that for each (Ψ, Ω), the prior probability contained in a small Kullback-Leibler ball around (Ψ, Ω) can be lower bounded by a function of the ball's radius. Then we studied a sequence of likelihood ratio tests defined on sieves of the parameter space that can correctly distinguish between parameter values that are sufficiently far away from each other in log-affinity. In particular, we bounded the error rate of such tests and then bounded the covering number of the sieves.

Ning et al. (2020) studied the sparse marginal regression model in Equation (1.1) instead of the sparse chain graph. Although these are somewhat different models, our overall proof strategy is quite similar to theirs. However, there are important technical differences. First, they placed a prior on Ω's eigendecomposition while we placed an arguably simpler and more natural element-wise prior on Ω. The second and more substantive difference is in how we bound the covering number of sieves of the underlying parameter space. Because they specified exactly sparse priors on the elements of $B = \Psi\Omega^{-1}$, it was enough for them to carefully bound the covering number of exactly low-dimensional sets of the form $\mathcal{A} \times \{0\}^r$ where $\mathcal{A}$ is some subset of a multidimensional Euclidean space and $r > 0$ is a positive integer. In contrast, because we specified absolutely continuous priors on the elements of Ψ, we had to cover "effectively low-dimensional" sets of the form $\mathcal{A} \times [-\delta, \delta]^r$ for small $\delta > 0$. Our key lemma, Lemma S5 of the Supplementary Materials (Shen et al., 2024), provides sufficient conditions on δ for bounding the ϵ-packing number of effectively low-dimensional sets using the ϵ'-packing number of $\mathcal{A}$ for a carefully chosen $\epsilon' > 0$.

4.2 Contraction of cgSSL

In order to establish our posterior concentration results, we first assume that the data were generated according to a sparse Gaussian chain graph model with true parameter Ψ_0 and Ω_0. We additionally modify our prior on Ω by truncating it to the set $\{\Omega \succ \tau\}$ for some small τ that does not depend on n. Finally, we make the following assumptions about the spectra of Ψ_0 and Ω_0 and on the dimensions n, p, and q. Note that for sequences $\{a_n\}$ and $\{b_n\}$, we write $a_n \lesssim b_n$ to mean that there is some constant C independent of n such that $a_n \leq Cb_n$.

A1 Bounded operator norms: $\Psi_0 \in \mathcal{T}_0 = \{\Psi : |||\Psi|||_2 < a_1\}$ and $\Omega_0 \in \mathcal{H}_0 = \{\Omega : \mathrm{eig}(\Omega) \subseteq [1/b_2, 1/b_1]\}$ where $|||\cdot|||$ is the operator norm, eig is the set of eigenvalues, and $a_1, b_1, b_2 > 0$ are fixed positive constants not depending on n.

A2 Dimensionality: We assume that $\log(n) \lesssim \log(q)$; $\log(n) \lesssim \log(p)$; and

$$\max\{p, q, s_0^{\Omega}, s_0^{\Psi}\} \log(\max\{p, q\})/n \to 0,$$

where s_0^{Ω} and s_0^{Ψ} are the number of non-zero free parameters in Ω and Ψ.

A3 Tuning the Ψ prior: We assume that $(1-\theta)/\theta \sim (pq)^{2+a'}$; $\lambda_0 \sim \max\{n, pq\}^{2+b'}$; and $\lambda_1 \asymp 1/n$ where $a' > 0$ and $b' > 1/2$ are fixed constants not depending on n

A4 Tuning the Ω prior: We assume that $(1-\eta)/\eta \sim \max\{Q, pq\}^{2+a}$; $\xi_0 \sim \max\{Q, pq, n\}^{4+b}$; and $\xi_1 \asymp 1/\max\{Q, n\}$, where $Q = q(q-1)/2$ and $a, b > 0$ are fixed constants not depending on n.

Before proceeding, we highlight two key differences between the above assumptions and model introduced in Section 3.1. Although the prior in Section 3.1 restricts Ω to the positive-definite cone, our modified prior and Assumption A1 bound the smallest eigenvalue of Ω away from zero. The stronger assumption ensures that the entries of $\Psi\Omega^{-1}$ do not diverge in our theoretical analysis and parallels those made by Gan et al. (2019a), Ning et al. (2020), and Sagar et al. (2021). Based on these works and ours, we conjecture that bounded eigenvalue assumptions may in fact be necessary. Additionally, like Ročková and George (2018) and Gan et al. (2019a), we restricted our theoretical analysis to the setting where the proportion of non-negligible parameters, θ and η, are fixed and known (Assumptions A3 and A4).

Theorem 1 (Posterior contraction). *Under Assumptions A1–A4, there is a constant $M_1 > 0$ not depending on n such that*

$$\sup_{\Psi \in \mathcal{T}_0, \Omega \in \mathcal{H}_0} \mathbb{E}_0 \Pi\left(\Psi : ||X(\Psi\Omega^{-1} - \Psi_0\Omega_0^{-1})||_F^2 \geq M_1 n\epsilon_n^2 | Y_1, \ldots, Y_n\right) \longrightarrow 0 \tag{4.1}$$

$$\sup_{\Psi \in \mathcal{T}_0, \Omega \in \mathcal{H}_0} \mathbb{E}_0 \Pi\left(\Omega : ||\Omega - \Omega_0||_F^2 \geq M_1 \epsilon_n^2 | Y_1, \ldots, Y_n\right) \longrightarrow 0 \tag{4.2}$$

where $\epsilon_n = \sqrt{\max\{p, q, s_0^{\Omega}, s_0^{\Psi}\} \log(\max\{p, q\})/n}$. Note that $\epsilon_n \to 0$ as $n \to \infty$.

A key step in proving Theorem 1 is Lemma 1. In order to state this lemma, we denote the effective dimensions of Ψ and Ω by $|\nu_{\psi}(\Psi)|$ and $|\nu_{\omega}(\Omega)|$. The effective dimension of Ψ (resp. Ω) counts the number of entries (resp. off-diagonal entries in the lower-triangle) whose absolute value exceeds the intersection point of the spike and slab prior densities.

Lemma 1 (Dimension recovery)**.** *For a sufficiently large constant $C_3' > 0$, we have:*

$$\sup_{\Psi\in\mathcal{T}_0,\Omega\in\mathcal{H}_0} \mathbb{E}_0\Pi\left(\Psi : |\nu_{\psi}(\Psi)| > C_3' s^{\star} | Y_1,\dots,Y_n\right) \to 0 \tag{4.3}$$

$$\sup_{B\in\mathcal{T}_0,\Omega\in\mathcal{H}_0} \mathbb{E}_0\Pi\left(\Omega : |\nu_{\omega}(\Omega)| > C_3' s^{\star} | Y_1,\dots,Y_n\right) \to 0 \tag{4.4}$$

where $s^{\star} = \max\{p, q, s_0^{\Omega}, s_0^{\Psi}\}$.

Lemma 1 guarantees that the cgSSL posterior does not grossly overestimate the number of non-zero entries in Ψ and Ω.

Note that the result in Equation (4.1) shows that the vector $X\Psi\Omega^{-1}$ converges to the vector of evaluations of the true regression function $\Omega_0^{-1}\Psi_0^{\top}\boldsymbol{x}$. Importantly, apart from Assumption A2 about the dimensions of X, Theorem 1 does not require any assumptions about the design matrix X. The contraction rates for Ψ and $\Psi\Omega^{-1}$, however, depend critically on a restricted eigenvalue of X, which we define as

$$\phi^2(s) = \inf_{A\in\mathbb{R}^{p\times q}:0\le|\nu(A)|\le s} \left\{ \frac{\|XA\|_F^2}{n\|A\|_F^2} \right\}.$$

Corollary 1 (Regression coefficients recovery)**.** *Under Assumptions A1–A4, there is a constant $M' > 0$ not depending on n such that*

$$\sup_{\Psi\in\mathcal{T}_0,\Omega\in\mathcal{H}_0} \mathbb{E}_0\Pi\left(||\Psi\Omega^{-1} - \Psi_0\Omega_0^{-1}||_F^2 \ge \frac{M'\epsilon_n^2}{\phi^2(s_0^{\Psi} + C_3' s^{\star})} \right) \to 0 \tag{4.5}$$

$$\sup_{\Psi\in\mathcal{T}_0,\Omega\in\mathcal{H}_0} \mathbb{E}_0\Pi\left(||\Psi - \Psi_0||_F^2 \ge \frac{M'\epsilon_n^2}{\min\{\phi^2(s_0^{\Psi} + C_3' s^{\star}), 1\}} \right) \to 0. \tag{4.6}$$

Corollary 1 shows that the posterior distribution of $\Psi\Omega^{-1}$ can contract at a faster or slower rate than the posterior distributions of $X\Psi\Omega^{-1}$ and Ω, depending on the design matrix. In particular, when X is poorly conditioned, we might expect the rate to be slower. In contrast, the term $\min\{\phi^2(s_0^{\Psi} + C_3' s^{\star}), 1\}$ appearing in the denominator of the rate in Equation (4.6) implies that the posterior distribution of Ψ cannot concentrate at a faster rate than the posterior distributions of $\Psi\Omega^{-1}$ and Ω, regardless of the design matrix. To develop some intuition about this phenomenon, notice that

$$\Psi - \Psi_0 = (\Psi\Omega^{-1} - \Psi_0\Omega_0^{-1})\Omega + (\Psi_0\Omega_0^{-1}(\Omega - \Omega_0)\Omega^{-1})\Omega.$$

Roughly speaking, the decomposition suggests that in order to estimate Ψ well, we must estimate both Ω and $\Psi\Omega^{-1}$ well. That is, estimating Ψ is at least as hard, statistically, as estimating Ω and $\Psi\Omega^{-1}$. Taken together, Corollary 1 suggests that while a carefully constructed design matrix can improve estimation of the matrix of *marginal* effects, $B = \Psi\Omega^{-1}$, it cannot generally improve estimation of the matrix of *direct* effects Ψ.

5 Synthetic experiments

We performed a simulation study to assess how well cgSSL with DPE (`cgSSL-DPE`) (i) recovers the supports of Ψ and Ω and (ii) estimates each matrix. We simulated several synthetic datasets of various dimensions and with different sparsity patterns in Ω (Figure 2) from the model in Equation (1.4). We compared cgSSL to several competitors: a fixed-penalty method (`cgLASSO`), which uses 10-fold cross-validation to select a single penalty λ for the entries in Ψ and a single fixed penalty ξ for the entries in Ω; Shen and Solís-Lemus (2021)'s CAR-LASSO procedure (`CAR`), which puts a common Laplace prior on the entries in Ψ and a common Laplace prior entries in Ω; Shen and Solís-Lemus (2021)'s adaptive CAR-LASSO (`CAR-A`), which puts individualized Laplace priors on the entries in Ψ and Ω; Deshpande et al. (2019)'s `mSSL` procedure that places spike-and-slab LASSO priors on the entries of B and Ω in Equation (1.1); and Consonni et al. (2017)'s Objective Bayes procedure (`OBFB`).

Before proceeding, we note that `mSSL` is inherently misspecified as it assumes $B = \Psi\Omega^{-1}$ is sparse instead of Ψ. Nevertheless, we included `mSSL` in our experiments to investigate how misspecifying the mean structure affects our ability to recover Ω. `OBFB` is similarly misspecified, albeit to a somewhat lesser extent, as it assumes that B (and consequently Ψ) is row-sparse. Unfortunately, the implementation of `OBFB` only returns posterior samples of indicators encoding which rows of B and which entries in Ω are non-zero. We computed a point-estimate of Ω's support using the posterior mode of the relevant indicators. We were unable, however, to reliably reconstruct Ψ's support from `OBFB`'s output. This is because a zero-entry in Ψ can occur when a non-zero row of B is orthogonal to a non-zero column in Ω. For these reasons, we only report `OBFB`'s performance in recovering the support of Ω.

Across all choices of dimension and Ω, we found that `cgSSL-DPE` achieved somewhat lower sensitivity but much higher precision in estimating the supports of both Ψ and Ω than the competing methods. This means that while `cgSSL-DPE` tended to return fewer non-zero parameter estimates than the other methods, we can be much more certain that those parameters are truly non-zero. Put another way, although the other methods can recover more of the truly non-zero signal, they do so at the expense of making many more false positive identifications in the supports of Ψ and Ω than `cgSSL-DPE`.

5.1 Simulation design

We simulated data with three different dimensions $(n, p, q) = (100, 10, 10)$, $(100, 20, 30)$, and $(400, 100, 30)$. For each choice of (n, p, q), we considered seven different Ω's: (i) an AR(1) model for Ω^{-1} so that Ω is tri-diagonal; (ii) an AR(2) model for Ω^{-1} so that $\omega_{k,k'} = 0$ whenever $|k - k'| > 2$; (iii) a block model in which Ω is block-diagonal with two dense $q/2 \times q/2$ diagonal blocks; (iv) a star graph where the off-diagonal entry $\omega_{k,k'} = 0$ unless k or k' is equal to 1; (v) a small-world network; (vi) a tree network; and (viii) dense model with all off-diagonal elements $\omega_{k,k'} = 2$.

In the AR(1) model we set $(\Omega^{-1})_{k,k'} = 0.7^{|k-k'|}$ so that $\omega_{k,k'} = 0$ whenever $|k-k'| > 1$. In the AR(2) model, we set $\omega_{k,k} = 1, \omega_{k-1,k} = \omega_{k,k-1} = 0.5$, and $\omega_{k-2,k} = \omega_{k,k-2} =$

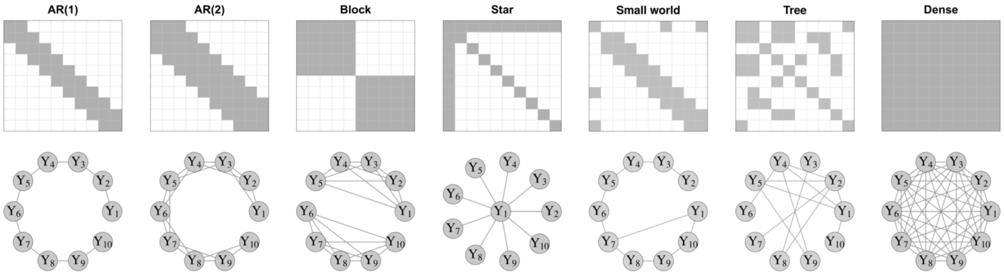


Figure 2: Supports of each Ω for $q = 10$ (top) and corresponding graph (bottom). Gray cells in top row indicate non-zero entries $\omega_{k,k'}$ and white cells indicate zeros.

0.25. For the block model, we partitioned $\Sigma = \Omega^{-1}$ into 4 $q/2 \times q/2$ blocks and set all entries in the off-diagonal blocks of Σ to zero. We then set $\sigma_{k,k} = 1$ and $\sigma_{k,k'} = 0.5$ for $1 \leq k \neq k' \leq q/2$ and for $q/2 + 1 \leq k \neq k' \leq q$. For the star graph, we set $\omega_{k,k} = 1$, $\omega_{1,k} = \omega_{k,1} = 0.1$ for each $k > 1$, and set the remaining off-diagonal elements of Ω equal to zero. For the small-world and tree networks, we first generated an appropriate random graph and then drew Ω from a G-Wishart distribution (Roverato, 2002; Lenkoski, 2013) with three degrees of freedom and an identity scale matrix. We generated the small-world graph using the Watts-Strogatz (Watts and Strogatz, 1998) model with a single community and rewiring probability of 0.1. We generated the tree graph by running a loop-erased random walk on a complete graph.

These seven specifications of Ω (top row of Figure 2) correspond to rather different underlying graphical structure among the outcomes (bottom row of Figure 2). The AR(1) model, for instance, represents an extremely sparse but regular structure while the AR(2) model is somewhat less sparse. While the star model and AR(1) model contain the same number of edges, the underlying graphs have markedly different degree distributions. Compared to the AR(1), AR(2), and star models, the block model is considerably denser. We included a dense Ω to assess how well all of the methods perform in a misspecified regime.

In total, we considered 21 combinations of dimensions (n, p, q) and Ω. We generated Ψ by randomly selecting 20% of entries to be non-zero and drawing the non-zero entries uniformly from $[-2{,}2]$. For each combination of $(n, p, q), \Omega$ and Ψ, we generated 100 synthetic datasets from the Gaussian chain graph model in Equation (1.4). The design matrix X contained independent standard normal entries.

5.2 Results

To assess estimation performance, we computed the Frobenius norm between the estimated matrices and the true data generating matrices. We additionally computed the coverage of our 95% bootstrap intervals, averaged over all entries in Ψ and Ω (`cgSSL-dpe+BB`) To assess the support recovery performance, we counted the number of elements in each of Ψ and Ω that were (i) correctly estimated as non-zero (true positives;

TP); (ii) correctly estimated as zero (true negatives; TN); (iii) incorrectly estimated as non-zero (false positives; FP); and (iv) incorrectly estimated as zero (false negatives; FN). We report the sensitivity (TP/(TP + FN)) and precision (TP/(TP + FP)). We also report the sensitivity and precision of estimating the supports of Ψ and Ω by checking whether zero is contained in the bootstrap interval for each $\psi_{j,k}$ and $\omega_{k,k'}$. Generally speaking, we prefer methods with high sensitivity and high precision. High sensitivity indicates that the method has correctly estimated most of the true non-zero parameters as non-zero. High precision, on the other hand, indicates that most of the estimated non-zero parameters are truly non-zero. For brevity, we only report results for the $(n, p, q) = (400, 100, 30)$ setting in Table 1. Results for the other dimensions were similar; see Tables S1 and S2 of the Supplementary Materials (Shen et al., 2024).

We performed all our experiments in a shared high-throughput computing environment (Center for High Throughput Computing, 2006) on nodes with 5 GB RAM and 2 CPU cores running `R` (v. 4.13; R Core Team, 2022). We ran `cgSSL-DPE` using the default hyperparameter settings recommended in Section 3.2. Similarly, we ran `mSSL` using the default settings that were recommended in Deshpande et al. (2019). For each MCMC method (`CAR`, `CAR-A`, and `OFBF`), we ran four Markov chains for 10,000 iterations each, discarding the first 5,000 samples from each as "burn-in," and retaining all subsequent samples. In general, although the simulated Markov chains did not mix (see below for more discussion), we nevertheless report the results based on the 20,000 obtained samples. Because `cgLASSO`'s cross-validation step often did not finish within 72 hours (the maximum time limit set by our cluster) when run serially, we parallelized our `cgLASSO` implementation, allowing our cluster to schedule separate jobs running each combination of fold and penalty. Because the scheduler sometimes delayed running certain jobs, we were unable to reliably time `cgLASSO` and do not report its runtimes in Table 1. Extrapolating from preliminary runs, however, we estimate that running one full fold would take around ten hours.

In terms of identifying non-zero direct effects (i.e., estimating the support of Ψ), `cgLASSO` consistently achieved the highest sensitivity. But, as the precision results indicate, the majority of `cgLASSO`'s "discoveries" were in fact false positives. On further inspection, we determined that such behavior was the result of the cross-validation step in `cgLASSO`, which tended to select very small penalty values that promoted very little shrinkage. The other fixed penalty method, `CAR`, similarly displayed high sensitivity and low precision. In contrast, methods that deployed adaptive penalties (`CAR-A` and `cgSSL-DPE`), displayed higher precision in estimating the support of Ψ. In fact, at least for estimating the support of Ψ, `cgSSL-DPE` made no false positives in the vast majority of simulation replications.

We observed essentially the same phenomenon for Ω: although `cgSSL-DPE` generally returned fewer non-zero estimates of $\omega_{k,k'}$, the vast majority of these estimates were true positives. In a sense, the fixed penalty methods (`cgLASSO` and `CAR`) cast a very wide net when searching for non-zero signal in Ψ and Ω, leading to large number of false positive identifications in the supports of these matrices. Adaptive penalty methods, on the other hand, were much more discerning. Interestingly, `mSSL` recovered Ω's support substantially better than `OBFB`. We suspect that the discrepancy stems from the fact that `mSSL` deploys adaptive penalization while `OBFB` utilizes a fixed Wishart prior for Ω.

	Ψ recovery			Ω recovery			Runtime
Method	SEN	PREC	FROB	SEN	PREC	FROB	Time (min)
$AR(1)$ model							
`cgLASSO`	**1**	0.2	0.07	0.94	0.46	28.0	–
`CAR`	0.82	0.46	0.02	**1**	0.27	**2.2**	472.7
`CAR-A`	0.86	0.73	0.01	**1**	**0.89**	7.3	564.8
`OBFB`	–	–	–	0.07	0.08	–	54.0
`cgSSL`	0.87	**0.99**	**0.00**	**1**	0.78	3.4	**26.7**
`cgSSL`+BB	0.88	**0.99**	–	**1**	0.83	–	55.4
`mSSL`	0.95	0.25	0.06	**1**	0.82	9.9	51.6
$AR(2)$ model							
`cgLASSO`	**1**	0.2	0.2	0.79	0.63	10.8	–
`CAR`	0.85	0.5	0.01	0.98	0.49	0.4	493.8
`CAR-A`	0.89	0.77	0.01	1	0.94	1.2	458.1
`OBFB`	–	–	–	0.06	0.13	–	49.6
`cgSSL`	0.92	**1**	**0.00**	**1**	0.47	**0.3**	13.0
`cgSSL`+BB	0.92	**1**	–	**1**	0.61	–	30.3
`mSSL`	0.99	0.23	0.03	1	**0.95**	2.5	**0.08**
Block model							
`cgLASSO`	**1**	0.20	0.4	0.87	0.97	10.1	–
`CAR`	0.84	0.46	0.02	0.71	0.76	3.4	480.6
`CAR-A`	0.88	0.70	**0.01**	0.75	**0.99**	4.1	512.4
`OBFB`	–	–	–	0.06	0.50	–	74.5
`cgSSL`	0.86	**0.99**	**0.01**	0.98	0.98	**1.4**	17.3
`cgSSL`+BB	0.88	**0.99**	–	**0.99**	0.98	–	42.6
`mSSL`	0.99	0.21	0.1	0.44	**0.99**	28.4	**2.0**
Star model							
`cgLASSO`	**0.93**	0.83	0.01	0.53	0.59	4.7	–
`CAR`	0.89	0.48	0.01	0.73	0.25	0.6	493.2
`CAR-A`	0.90	0.70	0.01	0.87	0.74	1.1	431.8
`OBFB`	–	–	–	0.06	0.08	–	65.2
`cgSSL`	0.89	**1**	**0.00**	**1**	0.90	**0.3**	1.1
`cgSSL`+BB	0.89	**1**	–	**1**	0.90	–	3.7
`mSSL`	0.90	0.85	0.02	**1**	**1**	0.7	**0.1**
Small world model							
`cgLASSO`	**0.99**	0.20	0.6	0.38	0.49	468.1	–
`CAR`	0.82	0.43	0.03	0.92	0.22	**10.7**	633.7
`CAR-A`	0.85	0.68	**0.01**	0.92	**0.79**	29.3	431.5
`OBFB`	–	–	–	0.06	0.08	–	47.7
`cgSSL`	0.82	**0.99**	**0.01**	**0.95**	0.78	25.8	359.7
`cgSSL`+BB	0.82	**0.99**	–	**0.95**	0.78	–	749.6
`mSSL`	0.88	0.34	0.1	0.58	0.7	122.8	**10.9**
Tree model							
`cgLASSO`	**0.99**	0.20	0.6	0.69	0.48	381.0	–
`CAR`	0.79	0.46	0.03	0.95	0.24	**14.2**	716.1
`CAR-A`	0.84	0.72	0.02	0.95	**0.86**	22.7	519.9
`OBFB`	–	–	–	0.07	0.08	–	44.7
`cgSSL`	0.84	**0.99**	**0.01**	**0.97**	0.61	18.8	398.6
`cgSSL`+BB	0.84	**0.99**	–	0.96	0.61	–	1139.1
`mSSL`	0.92	0.28	0.2	0.9	0.76	25.1	**28.3**
Dense model							
`cgLASSO`	–	–	–	–	–	–	–
`CAR`	0.87	0.39	**0.01**	0	–	964.2	522.7
`CAR-A`	0.88	0.52	**0.01**	0	–	970.0	431.8
`OBFB`	–	–	–	0.06	**1**	–	53.8
`cgSSL`	0.86	**0.98**	0.04	**0.26**	**1**	**918.4**	**1.7**
`cgSSL`+BB	0.86	**0.98**	–	0.24	**1**	–	10.2
`mSSL`	**0.96**	0.27	0.06	0.18	**1**	960.0	8.7

Table 1: Sensitivity, precision, and Frobenius error for Ψ and Ω when $(n, p, q) = (400, 100, 30)$. For each Ω, the best performance is bold-faced. For dense Ω, `cgLASSO` with tuned penalties did not converge within 72 hours.

In terms of estimation performance, with the exception of the dense Ω setting, the fixed penalty methods tended to have much larger Frobenius error than the adaptive penalty methods. Interestingly, for the six sparse Ω's, no method had high Frobenius for Ω but low Frobenius error for Ψ. This finding corroborates our intuition about Corollary 1: in order to estimate Ψ well, we must estimate Ω well. Additionally, our bootstrap intervals for individual parameters were reasonably well-calibrated and achieved near-nominal coverage (0.9 for Ψ and Ω). Using these intervals for support recovery was comparable to `cgSSL-DPE`.

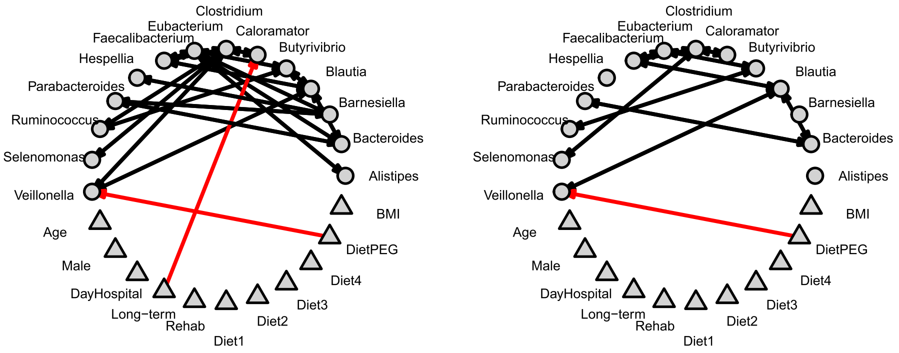
Across all choices of Ω, `cgSSL-DPE` was faster than the two MCMC methods. As alluded to above, the Markov chains simulated by those methods did not appear to have mixed, even after 10,000 iterations: across our experiments, about 25% of the $\omega_{k,k'}$'s had effective sample sizes less than 1,000 and around 5% of the parameters had marginal $\hat{R}$ estimates in excess of 1.1. Interestingly, `mSSL` was sometimes faster than `cgSSL-DPE`, depending on Ω.

6 Re-analysis of the gut microbiome data with cgSSL

Claesson et al. (2012) studied the gut microbiota of elderly adults using data sequenced from fecal samples taken from 178 subjects. They were primarily interested in understanding differences in the gut microbiome composition across several residence types (in the community, day-hospital, rehabilitation, or in long-term residential care) and across several different types of diet. They found that the gut microbiomes of residents in long-term care facilities were considerably less diverse than those of residents dwelling in the community. They additionally reported that diet had a large marginal effect on gut microbe diversity but they did not examine direct effects, which might align more closely with the underlying biological mechanism. We re-analyzed their data using the cgSSL to estimate the direct effects of each type of diet and residence type on gut microbiome composition. Before proceeding, we note that while raw microbiome data consists of counts, we used cgSSL to model the logarithms of relative abundances of each taxa. Section S4 of the Supplementary Materials (Shen et al., 2024) describes how we pre-processed the raw 16s-rRNA data to obtain these log-abundances.

In all, the dataset contains $n = 178$ observations of $p = 11$ predictors and $q = 14$ outcomes. We computed two graphs for these data, which are shown in Figure 3. In Figure 3a, edges correspond to the estimated non-zero entries of Ψ and Ω returned by cgSSL-DPE. Edges in Figure 3b instead correspond to those parameters whose bootstrapped uncertainty intervals did not contain zero.

In both graphs in Figure 3, we observed many more edges between the different species (corresponding to non-zero $\omega_{k,k'}$'s) than edges between covariates and species (corresponding to non-zero $\psi_{j,k}$'s). In both graphical models, we estimated that percutaneous endoscopic gastronomy (PEG), in which a feeding tube is inserted into the abdomen, had a direct effect on the abundance of *Veillonella*, which is involved in lactose fermentation. Our findings reassuringly align with those in Takeshita et al. (2011), who reported a negative effect of PEG on this genus. Although cgSSL-DPE addition-



(a)

(b)

Figure 3: Graphical models for Claesson et al. (2012)'s gut microbiome dataset estimated by cgSSL-DPE (a) and cgSSL-DPE followed by the weighted Bayesian bootstrap (b). Triangles correspond to covariates and circles correspond to responses. Edges corresponding to non-zero $\psi_{j,k}$'s are colored red.

ally identified staying in a day hospital as having a direct effect on *Caloramator*, the corresponding bootstrap interval contained zero.

Our results suggest that the large marginal effects reported by Claesson et al. (2012) are a by-product of only a few direct effects and substantial residual conditional dependence between species. For instance, because PEG has a direct effect on *Veillonella*, which is conditionally correlated with *Clostridium*, *Butyrivibrio*, and *Blautia*, PEG displays a marginal effect on each of these other genus. In this way, the cgSSL can provide a more nuanced understanding of the underlying biological mechanism than simply estimating the matrix of marginal effects $B = \Psi\Omega^{-1}$. We note, however, that Claesson et al. (2012)'s dataset does not contain an exhaustive set of environmental and patient life-style predictors. Accordingly, our re-analysis is limited in the sense that were we able to incorporate additional predictors, the estimated graphical model may be quite different. Further, although we followed a relatively standard pre-processing workflow, we anticipate that our overall findings will be somewhat sensitive to some of the choices made in converting the raw microbiome data into the log abundances that we modeled.

7 Discussion

In the Gaussian chain graph model in Equation (1.4), Ψ is a matrix containing all of the direct effects of p predictors on q outcomes while Ω is the residual precision matrix that encodes the conditional dependence relationships between the outcomes that remain after adjusting for the predictors. We have introduced the cgSSL procedure for obtaining simultaneously sparse estimates of Ψ and Ω. In our procedure, we formally specify spike-and-slab LASSO priors on the free elements of Ψ and Ω and use an ECM algorithm to maximize the posterior density. Our ECM algorithm iteratively solves a sequence of penalized maximum likelihood problem with self-adaptive penalties. Across several simulated datasets, cgSSL demonstrated excellent support recovery and estimation performance, substantially out-performing competitors that deployed constant penalties. We further characterized the asymptotic properties of cgSSL posteriors, establishing posterior contraction rates under relatively mild assumptions. To the best of our knowledge, these are the first posterior contraction results for sparse Gaussian chain graph models with element-wise priors.

Our main asymptotic result (Theorem 1) notwithstanding, quantifying the finite sample uncertainty around the MAP estimate returned by our cgSSL procedure remains a challenging problem. We found that Newton et al. (2021)'s weighted Bayesian bootstrap can produce uncertainty intervals for individual parameters with close-to-nominal frequentist coverage. Although the bootstrap does not exactly sample from the posterior, randomly re-centering the prior densities in Equation (3.5), as suggested by Nie and Ročková (2022), may improve the approximation. Understanding the extent to which solving randomly re-weighted and shifted MAP estimation problems faithfully approximate posterior sampling remains an important open question.

Although our motivating example involves only continuous outcomes, many applications feature outcomes of mixed type — that is both continuous and discrete outcomes. We anticipate that cgSSL could be extended to such settings using a strategy similar to that in Kowal and Canale (2020). Specifically, one would model the discrete outcomes as a truncated and transformed latent Gaussian vector and fit a sparse Gaussian chain graphical model to the latent vector. The main challenge of such an extension lies in adaptively learning an appropriate transformation.

Supplementary Material

Supplementary Material for "Estimating sparse direct effects in multivariate regression with the spike-and-slab LASSO" (DOI: 10.1214/24-BA1430SUPP; .pdf). Full derivation of the cgSSL procedure, additional simulation study results, and proofs of our asymptotic results

References

Avis, T., Wilson, F. X., Khan, N., Mason, C. S., and Powell, D. J. (2021). "Targeted microbiome-sparing antibiotics." *Drug Discovery Today*, 26(9): 2198–2203. 3

Bai, R., Moran, G. E., Antonelli, J. L., Chen, Y., and Boland, M. R. (2020). "Spike-and-slab group LASSOs for grouped regression and sparse generalized additive models." *Journal of the American Statistical Association*. MR4399078. doi: https://doi.org/10.1080/01621459.2020.1765784. 6, 12

Blaser, M. J. (2016). "Antibiotic use and its consequences for the normal microbiome." *Science*, 352(6285): 544–545. 3

Cai, T. T., Li, H., Liu, W., and Xie, J. (2013). "Covariate-adjusted precision matrix estimation with an application in genetical genomics." *Biometrika*, 100(1): 139–156. MR3034329. doi: https://doi.org/10.1093/biomet/ass058. 5

Center for High Throughput Computing (2006). “Center for High Throughput Computing.” 17

Chen, J., Xu, P., Wang, L., Ma, J., and Gu, Q. (2018). “Covariate adjusted precision matrix estimation via nonconvex optimization.” In *Proceedings of the 35th International Conference on Machine Learning*. 5

Chen, M., Ren, Z., Zhao, H., and Zhou, H. (2016). “Asymptotically normal and efficient estimation of covariate-adjusting Gaussian graphical model.” *Journal of the American Statistical Association*, 111(513): 394–406. MR3494667. doi: `https://doi.org/10.1080/01621459.2015.1010039`. 5

Claesson, M. J., Jeffery, I. B., Conde, S., Power, S. E., O’Connor, E. M., Cusack, S., Harris, H. M., Coakley, M., Lakshminarayanan, B., O’Sullivan, O., Fitzgerald, G. F., Deane, J., O’Connor, M., Harnedy, N., O’Connor, K., O’Mahony, D., van Sinderen, D., Wallace, M., Brennan, L., Stanton, C., Marchesi, J. R., Fitzgerald, A. P., Shanahan, F., Hill, C., Ross, R., and O’Toole, P. W. (2012). “Gut microbiota composition correlates with diet and health in the elderly.” *Nature*, 488(7410): 178–184. MR2647488. doi: `https://doi.org/10.1007/s00233-010-9216-3`. 2, 19, 20

Consonni, G., La Rocca, L., and Peluso, S. (2017). “Objective Bayes covariate-adjusted sparse graphical model selection.” *Scandinavian Journal of Statistics*, 44: 741–764. MR3687971. doi: `https://doi.org/10.1111/sjos.12273`. 5, 15

Cox, D. R. and Wermuth, N. (1993). “Linear dependencies represented by chain graphs.” *Statistical Science*, 204–218. MR1243593. 4

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). “Maximum likelihood from incomplete data via the EM algorithm.” *Journal of the Royal Statistical Society: Series B*, 39(1): 1–38. MR0501537. 6

Deshpande, S. K., Ročková, V., and George, E. I. (2019). “Simultaneous variable and covariance selection with the multivariate spike-and-slab LASSO.” *Journal of Computational and Graphical Statistics*, 28(4): 921–931. MR4045858. doi: `https://doi.org/10.1080/10618600.2019.1593179`. 6, 10, 11, 15, 17

Fishbein, S. R., Mahmud, B., and Dantas, G. (2023). “Antibiotic perturbations to the gut microbiome.” *Nature Reviews Microbiology*, 1–17. 3

Friedman, J., Hastie, T., and Tibshirani, R. (2008). “Sparse inverse covariance estimation with the graphical LASSO.” *Biostatistics*, 9(3): 432–441. 9

Gan, L., Narisetty, N. N., and Liang, F. (2019a). “Bayesian regularization for graphical models with unequal shrinkage.” *Journal of the American Statistical Association*, 114(527): 1218–1231. MR4011774. doi: `https://doi.org/10.1080/01621459.2018.1482755`. 6, 13

Gan, L., Yang, X., Narisetty, N. N., and Liang, F. (2019b). “Bayesian joint estimation of multiple graphical models.” In *Advances in Neural Information Processing Systems*. 6

George, E. I. and McCulloch, R. E. (1993). "Variable selection via Gibbs sampling." *Journal of the American Statistical Association*, 88(423): 881–889. 5

Ghosal, S. and van der Vaart, A. (2017). *Fundamentals of Nonparametric Bayesian Inference*. Cambridge University Press. MR3587782. doi: `https://doi.org/10.1017/9781139029834`. 12

Guinane, C. M. and Cotter, P. D. (2013). "Role of the gut microbiota in health and chronic gastrointestinal disease: understanding a hidden metabolic organ." *Therapeutic Advances in Gastroenterology*, 6(4): 295–308. 1

Kamada, N. and Núñez, G. (2014). "Regulation of the immune system by the resident intestinal bacteria." *Gastroenterology*, 146(6): 1477–1488. 1

Kowal, D. R. and Canale, A. (2020). "Simultaneous transformation and rounding (STAR) models for integer-valued data." *Electronic Journal of Statistics*, 14(1): 1744–1772. MR4083734. doi: `https://doi.org/10.1214/20-EJS1707`. 21

Lenkoski, A. (2013). "A direct sampler for G-Wishart variates." *STAT*, 2: 119–128. MR4027305. doi: `https://doi.org/10.1002/sta4.23`. 16

Li, Z., Mccormick, T., and Clark, S. (2019). "Bayesian joint spike-and-slab graphical LASSO." In *Proceedings of the 36th International Conference on Machine Learning*. 6

McCarter, C. and Kim, S. (2014). "On sparse Gaussian chain graph models." In *Advances in Neural Information Processing Systems*. 4

Meng, X.-L. and Rubin, D. B. (1993). "Maximum likleihood estimation via the ECM algorithm: A general framework." *Biometrika*, 80(2): 267–278. MR1243503. doi: `https://doi.org/10.1093/biomet/80.2.267`. 9

Mitchell, T. J. and Beauchamp, J. J. (1988). "Bayesian variable selection in linear regression." *Journal of the American Statistical Association*, 83(404): 1023–1032. MR0997578. 5

Moran, G. E., Ročková, V., and George, E. I. (2019). "Variance prior forms for high-dimensional Bayesian variable selection." *Bayesian Analysis*, 14(4): 1091–1119. MR4044847. doi: `https://doi.org/10.1214/19-BA1149`. 10, 11

Moran, G. E., Ročková, V., and George, E. I. (2021). "Spike-and-slab LASSO biclustering." *The Annals of Applied Statistics*, 15(1): 148–173. MR4255269. doi: `https://doi.org/10.1214/20-aoas1385`. 6, 10

Newton, M. A., Polson, N. G., and Xu, J. (2021). "Weighted Bayesian bootstrap for scalable posterior distributions." *Canadian Journal of Statistics*, 49(2): 421–437. MR4267927. doi: `https://doi.org/10.1002/cjs.11570`. 3, 11, 21

Ni, Y., Stingo, F. C., and Baladandayuthapani, V. (2019). "Bayesian graphical regression." *Journal of the American Statistical Association*, 114(525): 184–197. MR3941247. doi: `https://doi.org/10.1080/01621459.2017.1389739`. 5

Nie, L. and Ročková, V. (2022). "Bayesian bootstrap spike-and-slab LASSO." *Journal of* *the American Statistical Association*. MR4646623. doi: https://doi.org/10.1080/01621459.2022.2025815. 21

Ning, B., Jeong, S., and Ghosal, S. (2020). "Bayesian linear regression for multivariate responses under group sparsity." *Bernoulli*, 26(3): 2353–2382. MR4091112. doi: https://doi.org/10.3150/20-BEJ1198. 6, 12, 13

R Core Team (2022). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Consulting, Vienna, Austria. 11, 17

Ročková, V. and George, E. I. (2014). "EMVS: The EM approach to Bayesian variable selection." *Journal of the American Statistical Association*, 109(506): 828–846. MR3223753. doi: https://doi.org/10.1080/01621459.2013.869223. 6, 10

Ročková, V. and George, E. I. (2016). "Fast Bayesian factor analysis via automatic rotations to sparsity." *Journal of the American Statistical Association*, 111(516): 1608–1622. MR3601721. doi: https://doi.org/10.1080/01621459.2015.1100620. 6, 10

Ročková, V. and George, E. I. (2018). "The spike-and-slab LASSO." *Journal of the American Statistical Association*, 113(521): 431–444. MR3805383. 3, 6, 7, 10, 11, 13

Roverato, A. (2002). "Hyper Inverse Wishart distribution for non-decomposable graphs and its applicaiton to Bayesian inference for Gaussian graphical models." *Scandinavian Journal of Statistics*, 29(3): 391–411. MR1925566. doi: https://doi.org/10.1111/1467-9469.00297. 16

Sagar, K., Banerjee, S., Datta, J., and Bhadra, A. (2021). "Precision matrix estimation under the horseshoe-like prior-penalty dual." *arXiv preprint arXiv:2104.10750*. MR4693861. doi: https://doi.org/10.1214/23-ejs2196. 13

Schwartz, D. J., Langdon, A. E., and Dantas, G. (2020). "Understanding the impact of antibiotic perturbation on the human microbiome." *Genome Medicine*, 12(1): 1–12. 3

Shen, Y. and Solís-Lemus, C. (2021). "Bayesian conditional auto-regressive LASSO models to learn sparse microbial networks with predictors." *arXiv preprint arXiv:2012.08397*. 4, 15

Shen, Y., Solís-Lemus, C., and Deshpande, S. K. (2024). "Supplement to "Sparse Gaussian chain graph models with the spike-and-slab LASSO"." 7, 8, 9, 10, 11, 12, 17, 19

Shreiner, A. B., Kao, J. Y., and Young, V. B. (2015). "The gut microbiome in health and in disease." *Current Opinion in Gastroenterology*, 31(1): 69. 1

Singh, R. K., Chang, H.-W., Yan, D., Lee, K. M., Ucmak, D., Wong, K., Abrouk, M., Farahnik, B., Nakamura, M., Zhu, T. H., Bhutani, T., and Liao, W. (2017). "Influence of diet on the gut microbiome and implications for human health." *Journal of Translational Medicine*, 15(1): 1–17. 1

Sonntag, D. and Peña, J. M. (2015). "Chain graphs and gene networks." *Foundations of Biomedical Knowledge Representation: Methods and Applications*, 159–178. 4

Takeshita, T., Yasui, M., Tomioka, M., Nakano, Y., Shimazaki, Y., and Yamashita, Y. (2011). "Enteral tube feeding alters the oral indigenous microbiota in elderly adults." *Applied and Environmental Microbiology*, 77(19): 6739–6745. 19

Tang, Z., Shen, Y., Zhang, X., and Yi, N. (2017). "The spike-and-slab LASSO generalized linear models for prediction and associated genes detection." *Genetics*, 205: 77–88. 6

Thorpe, C. M., Kane, A. V., Chang, J., Tai, A., Vickers, R. J., and Snydman, D. R. (2018). "Enhanced preservation of the human intestinal microbiota by ridinilazole, a novel Clostridium difficile-targeting antibacterial, compared to vancomycin." *PLOS ONE*, 13(8): e0199810. 3

Wang, Z., Klipfell, E., Bennett, B. J., Koeth, R., Levison, B. S., DuGar, B., Feldstein, A. E., Britt, E. B., Fu, X., Chung, Y.-M., et al. (2011). "Gut flora metabolism of phosphatidylcholine promotes cardiovascular disease." *Nature*, 472(7341): 57–63. 1

Watts, D. J. and Strogatz, S. H. (1998). "Collective dynamics of 'small world' networks." *Nature*, 3933: 440–442. MR1716136. 16

Yassour, M., Vatanen, T., Siljander, H., Hämäläinen, A.-M., Härkönen, T., Ryhänen, S. J., Franzosa, E. A., Vlamakis, H., Huttenhower, C., Gevers, D., et al. (2016). "Natural history of the infant gut microbiome and impact of antibiotic treatment on bacterial strain diversity and stability." *Science Translational Medicine*, 8(343): 343ra81–343ra81. 3