Latent Network Estimation and Variable Selection for Compositional Data via Variational EM

Nathan Osborne[1], Christine B. Peterson[2], and Marina Vannucci[1]

[1]Department of Statistics, Rice University, Houston, TX

[2]Department of Biostatistics, The University of Texas MD Anderson Cancer Center, Houston, TX

May 17, 2022

Abstract

Network estimation and variable selection have been extensively studied in the statistical literature, but only recently have those two challenges been addressed simultaneously. In this paper, we seek to develop a novel method to simultaneously estimate network interactions and associations to relevant covariates for count data, and specifically for compositional data, which have a fixed sum constraint. We use a hierarchical Bayesian model with latent layers and employ spike-and-slab priors for both edge and covariate selection. For posterior inference, we develop a novel variational inference scheme with an expectation maximization step, to enable efficient estimation. Through simulation studies, we demonstrate that the proposed model outperforms existing methods in its accuracy of network recovery. We show the practical utility of our model via an application to microbiome data. The human microbiome has been shown to contribute to many of the functions of the human body, and also to be linked with a number of diseases. In our application, we seek to better understand the interaction between microbes and relevant covariates, as well as the interaction of microbes with each other. We call our algorithm SINC (Simultaneous Inference for Networks and Covariates) and provide a Python implementation, which is available online.

Keywords: graphical model, variational inference, EM algorithm, count data, Bayesian hierarchical model, microbiome data

1 Introduction

Variable selection, also known as feature selection, is a well-studied subject in the statistical literature, particularly in the context of regression models, where many approaches have been proposed. Feature selection offers an opportunity to both improve model predictions, by avoiding the inclusion of noisy or

irrelevant predictors, and to identify interpretable parsimonious models. Frequentist approaches often use a penalized likelihood to obtain sparse estimates of the regression coefficients, and include methods such as LASSO (Tibshirani, 1996), adaptive LASSO (Zou, 2006), and SCAD (Fan and Li, 2001). Alternatively, Bayesian approaches employ carefully constructed priors on the regression coefficients to identify the relevant variables. Spike-and-slab priors, first proposed by Mitchell and Beauchamp (1988), are a popular class of priors that use a latent indicator to represent variable inclusion. Conditional on the indicators, the regression coefficients are assumed to come from a mixture prior representing important vs negligible effects (George and McCulloch, 1997; Brown et al., 1998). In addition to sparse estimation of the coefficients, these priors produce posterior probabilities of inclusion (PPIs) for each covariate that capture the uncertainty in the selection. Spike-and-slab priors have been extended to regression models for non-Gaussian data, including binary, multinomial and count responses (Raftery, 1996; Ntzoufras et al., 2003; Sha et al., 2004; Wadsworth et al., 2017; Koslovsky and Vannucci, 2020).

A parallel development has happened in the graphical model literature: in this framework, nodes correspond to variables, and edges connecting these nodes represent conditional dependence relations. In the Gaussian setting, the problem of selecting edges in the graph reduces to the estimation of a sparse inverse covariance matrix, since exact zeros in this matrix, which is also known as the precision matrix, correspond to conditional independence relations (Dempster, 1972). In frequentist settings, penalized likelihood methods, such as neighborhood selection (Meinshausen and Bühlmann, 2006) and the graphical LASSO (Yuan and Lin, 2007; Friedman et al., 2008), have been proposed. These methods have been extended to count data by using data transformations (Kurtz et al., 2015) or penalized log-likelihood methods (Fang et al., 2017). In Bayesian inference, the G-Wishart prior (Roverato, 2002), which is the conjugate prior that imposes exact zeros in the precision matrix, has been explored by several authors for inference in Gaussian graphical models, but poses significant computational challenges (Lenkoski and Dobra, 2011). As a result, this prior is not easily scalable. Alternative shrinkage constructions that employ continuous priors on the off-diagonal elements of the precision matrix have been proposed, such as the Bayesian graphical lasso (Wang, 2012), which relies on double exponential priors, and mixture priors (Wang, 2015), inspired by the spike-and-slab priors in the regression framework discussed above. To enable estimation of the spike-and-slab model of Wang (2015) in high-dimensional settings, Li and McCormick (2019) recently proposed an efficient expectation conditional maximization method, which offers an attractive alternative to stochastic search approaches.

In this paper, we propose a novel Bayesian hierarchical model for count data that allows for simultaneous estimation of covariate dependence and network interactions. Methods for simultaneous estimation are gaining popularity, with approaches including penalized likelihood methods (Rothman et al., 2010; Yang et al., 2017), and, most recently, spike-and-slab lasso prior models (Deshpande et al., 2019). By accounting for covariate selection, simultaneous estimation methods are able to control for those variables, which ultimately leads to more accurate network estimation. Moreover, simultaneous estimation can improve the detection of covariate effects, as noted by Deshpande et al. (2019). However, with the exception of Yang et al. (2017),

these methods are not suitable for count data. In our approach, we consider multivariate count data, and specifically compositional data that have a fixed sum constraint. We model the data using a Dirichlet-multinomial likelihood and then introduce a latent layer by modeling the log concentration parameters via a Gaussian distribution. We account for covariates through the mean function of the latent layer and employ multivariate variable selection spike-and-slab priors that allow each covariate to be relevant for individual response variables (Richardson et al., 2010; Stingo et al., 2010). We also capture a network of latent dependence relationships by estimating the inverse covariance matrix via the mixture prior of Wang (2015). For posterior inference, we implement a novel variational Bayes approach that includes an expectation-minimization (EM) step to estimate the model. This allows us to gain flexibility by using a Bayesian model, while still remaining computationally efficient. Additionally, the algorithm is developed so that multiple steps can be run in parallel, achieving larger computational gains. We show through simulations that our method outperforms the LASSO-based approach of Yang et al. (2017). We refer to our model as SINC (Simultaneous Inference of Networks and Covariates).

Compositional data are often collected in chemistry, geology, and biology applications. In biomedicine, modern genomic sequencing technologies have allowed investigators to collect samples on the human microbiome. Microbes associated with the human body include eukaryotes, archaea, bacteria, and viruses, which have been shown to contribute to important bodily functions including food digestion and energy supply. The human microbiome has also been implicated in many diseases including colorectal cancer, inflammatory bowel disease, and immunologically mediated skin diseases. The observed data from a microbiome study are typically short reads of DNA sequences, which are clustered to create operational taxonomic units (OTUs). The abundances across samples of these OTUs, which represent genetically close groups of microbes assumed to have similar functions, are taken as input to downstream analysis. A challenge to modeling these data is that the number of counts for a particular OTU depends on the number of sequences collected for that sample, meaning that the observed counts are dependent on each other, as they constitute proportions of a whole. This results in data that are compositional. For these reasons, Dirichlet-multinomial distributions are particularly appropriate to model microbiome data, as demonstrated by several authors (Chen and Li, 2013; Tang et al., 2018; Wadsworth et al., 2017). In the application of this paper we focus on two questions of interest in the understanding of the microbiome: which variables influence the microbial abundances, and what are the dependence relationships among microbes. The abundance of microbes or groups of microbes is dependent on many factors. Microbial abundance may be related to external covariates, such as diet, cytokines, or use of medication.These factors influence the microbiome by introducing new organisms, changing the abundance of metabolites, or altering the pH of their environment. For example, consumption of an animal-based diet high in meat has been shown to increase production of bile acid, which inhibits growth of bacteria belonging to the Bacteroidetes and Firmicutes phyla (David et al., 2014). Antibiotics can alter the microbiome substantially, by killing off components of the microbiome in addition to the bacteria triggering the infection (Edwards et al., 2019). As we understand more about the importance of the

microbiome, it is also critical to understand what factors lead to the prevalence of different microbes. Here we apply the proposed method to real data from the Multi-Omic Microbiome Study: Pregnancy Initiative (MOMS-PI) study, to estimate the interaction between microbes in the vagina, as well as the interplay between vaginal cytokines and microbial abundances, providing insight into mechanisms of host-microbial interaction during pregnancy.

The paper is outlined as follows: in Section 2, we describe the proposed hierarchical model, followed by the variational EM estimation method in Section 3. We provide a simulation study in section 4, and then showcase the proposed model in an application to multi-omic data from a study on the role of the microbiome in pregnancy in Section 5. Finally, we discuss the advantages of the proposed model in Section 6.

2 Proposed Model

Suppose we have observed multivariate counts arranged in an $n \times p$ matrix, $\mathbf{X}$, where p is the number of observed variables measured across n samples. We then let the p-vector $\mathbf{X}_i$ correspond to the measurements for observation i, and the matrix entry $x_{i,j}$ correspond to the j^{th} variable measurement for the i^{th} observation. We also observe q covariates for each of the n observations, with these q additional factors possibly influencing the measured counts for each observation. We arrange the covariate data in an $n \times q$ matrix, $\mathbf{M}$.

We are interested in understanding the conditional dependence relationships among the p variables while simultaneously selecting the relevant covariates. We adopt a hierarchical model formulation with a latent Gaussian layer, similarly to Yang et al. (2017), as

$$\begin{aligned} \mathbf{Z}_i \mid \mathbf{B}_0, \mathbf{M}_i, \mathbf{B}, \mathbf{\Omega} &\sim \text{MVNorm}(\mathbf{B}_0 + \mathbf{M}_i\mathbf{B}, \mathbf{\Omega}^{-1}) \\ \boldsymbol{\alpha}_i &= \exp\{\mathbf{Z}_i\} \\ \mathbf{h}_i \mid \boldsymbol{\alpha}_i &\sim \text{Dirichlet}(\boldsymbol{\alpha}_i) \\ \mathbf{X}_i \mid \mathbf{h}_i &\sim \text{Multinomial}(\mathbf{h}_i, N_i). \end{aligned} \tag{1}$$

In this hierarchical formulation, we introduce a latent normal variable $\mathbf{Z}_i$, which is a direct transformation of the concentration parameter $\boldsymbol{\alpha}_i$ and therefore controls the observed counts $\mathbf{X}_i$. This model has several important features: the Dirichlet-multinomial likelihood for count data, $\mathbf{X}_i$, allows us to account for overdispersion as well as the compositional nature of the data. The dependence on covariates is incorporated through the mean of the multivariate normal, where the observed covariates $\mathbf{M}_i$ have effects $\mathbf{B}$. The dependence among the $\mathbf{Z}_i$ is captured by the inverse covariance matrix, also known as the precision matrix, $\mathbf{\Omega}$. The $1 \times p$ vector $\mathbf{B}_0$ accounts for the mean of each column of the latent matrix $\mathbf{Z}$. In our modeling approach, careful consideration of the priors on the covariate effects $\mathbf{B}$, the intercepts $\mathbf{B}_0$ and the precision matrix $\mathbf{\Omega}$ allows us to construct a directed graph between covariates $\mathbf{M}$ and latent variables $\mathbf{Z}$, as well as an undirected graph between the columns of $\mathbf{Z}$.

For microbiome studies, Gloor et al. (2017) noted that the observed compositional data have a different correlation structure than the true underlying abundances. More specifically, due to the fixed sum constraint,

compositional data tend to exhibit negative correlations. In model formulation (1), we interpret the latent layer $\mathbf{h}_i$ to be the relative abundances, and $\boldsymbol{\alpha}_i$ to be the absolute abundances (Yang et al., 2017). By estimating a network on the latent $\mathbf{Z}$, we capture the network of the underlying, absolute abundances through the precision matrix $\boldsymbol{\Omega}$. Therefore, even though the latent Gaussian layer does not allow us to recover relationships directly among the observed counts, the inferred dependences do provide some insights into the relationships among the underlying processes. Latent graphical models for Poisson-distributed count data that use Gaussian layers were used by Vinci et al. (2018), for spike-count data. See also Talhouk et al. (2012) and Li et al. (2020) for latent graphical model constructions for binary data.

2.1 Prior on covariate effects B

Here we describe the prior on the covariate effects, which enables selection of the important associations between $\mathbf{X}$ and other potentially related factors $\mathbf{M}$. We consider the effects of the covariates $\mathbf{M}$ on each column of $\mathbf{Z}$ separately, which means that we will be able to update the columns of $\mathbf{B}$ independently of each other. Here $\mathbf{B}$ is a $q \times p$ matrix, where each column of $\mathbf{B}$ represents the vector of regression coefficients for the q covariates of $\mathbf{M}$ on the j^{th} column of $\mathbf{Z}$. We use a spike-and-slab prior on each element of the matrix $\mathbf{B}$, which shrinks features that do not influence $\mathbf{Z}$ to zero. Remember that we are looking at the columns of $\mathbf{Z}$ one at a time, and can thus say that any entry from the j^{th} column, $Z_{i,j}$, comes from a Normal($\mathbf{B}_{0j} + \mathbf{M}_i\mathbf{B}_j, \sigma_j^*$), where σ_j^* is the standard deviation of the j^{th} column of $\mathbf{Z}$, found by using the properties of the multivariate normal distribution shown in equation (1). The prior on $\mathbf{B}$ is as follows:

$$\begin{aligned} B_{k,j} \mid \gamma_{k,j}, \nu_B^2 &\sim \gamma_{k,j}\text{Normal}(0, \nu_B^2) + (1 - \gamma_{k,j})\delta_0 \\ \gamma_{k,j} \mid \theta_{\gamma_j} &\sim \text{Bernoulli}(\theta_{\gamma_j}) \\ \theta_{\gamma_j} \mid a_\gamma, b_\gamma &\sim \text{Beta}(a_\gamma, b_\gamma), \end{aligned} \tag{2}$$

for $j = 1, \ldots, p$ and $k = 1, \ldots, q$, and with δ_0 a point mass at 0, indicating that when $\gamma_{k,j}$ is 0, $B_{k,j}$ is exactly 0. Here, θ_{γ_j} is the probability of a variable being relevant in $\mathbf{B}_j$. Notice that the mixture prior (2) allows each variable to be relevant for individual responses (Richardson et al., 2010; Stingo et al., 2010), as opposed to spike-and-slab constructions that select variables as relevant to either all or none of the responses (Brown et al., 1998). We also put a non-informative prior on each element of $\mathbf{B}_0$, i.e. $B_{0j} \propto 1$.

2.2 Prior on precision matrix Ω

Next we introduce the prior on the precision matrix $\boldsymbol{\Omega}$, which allows us to learn a sparse association network. We consider the prior of Wang (2015) in the formulation proposed by Li and McCormick (2019):

$$\begin{aligned} \pi(\boldsymbol{\Omega} \mid \boldsymbol{\delta}, \nu_1, \nu_0, \lambda, \tau) \propto &\prod_{i<j} \left\{(1 - \delta_{i,j})\text{Normal}(\omega_{i,j} \mid 0, \frac{\nu_0^2}{\tau}) + \delta_{i,j}\text{Normal}(\omega_{i,j} \mid 0, \frac{\nu_1^2}{\tau})\right\} \cdot \\ &\prod_i \text{Exp}(\omega_{i,i} \mid \lambda/2)\mathbf{1}_{\boldsymbol{\Omega} \in M^+}, \end{aligned} \tag{3}$$

where ν_0 and ν_1 are fixed standard deviations, that assume small and large values respectively, $\delta_{i,j}$ is a latent variable indicating whether or not an edge is present between nodes i and j, and τ is a scaling parameter, with a hyperprior Gamma(a_τ, b_τ) that allows to adaptively learn the standard deviations. The original prior of Wang (2015) is obtained by setting $\tau = 1$. Additional complexity can be added to the prior on τ to include existing knowledge about variable associations, as shown in Li and McCormick (2019). The mixture of normals on the off-diagonal precision matrix entries enables the selection of interactions, represented by edges in a network, since non-zero precision matrix entries reflect conditional dependence relationships (Dempster, 1972). Here, entries reflecting conditional independence relations do not equal exactly zero, but get shrunk to close to zero. The diagonal entries are drawn from a common exponential prior. The final term in equation (3) expresses a constraint to the space of positive definite matrices M^+. This prior is particularly advantageous in our model, as it allows for efficient estimation via the EM algorithm and leads to less bias in estimation of the off-diagonal precision matrix elements than the graphical LASSO, as shown by Li and McCormick (2019).

We complete the modeling of the precision matrix $\mathbf{\Omega}$ by setting the prior on the graph structure, assuming independent Bernoulli distributions on the inclusion of each edge as follows:

$$\begin{aligned} \delta_{i,j} \mid \pi &\sim \text{Bernoulli}(\pi) \\ \pi \mid a_\pi, b_\pi &\sim Beta(a_\pi, b_\pi). \end{aligned} \tag{4}$$

3 Posterior Inference

We now discuss how to obtain posterior estimates of the parameters in the model outlined in Section 2. Instead of a traditional Markov chain Monte Carlo (MCMC) sampler, which can be computationally quite expensive, we rely on a Variational Inference (VI) approach, which aims to find an approximation of the posterior using optimization methods. VI works by specifying a family of approximate distributions $\mathcal{Q}$, which are densities over latent variables W that are dependent on free parameters $\boldsymbol{\xi}$, and then seeking to find the values of $\boldsymbol{\xi}$ that minimize the Kullback-Leibler (KL) divergence between the approximate distribution and the true posterior. As discussed in Blei et al. (2017), minimizing the KL divergence is equivalent to maximizing the Evidence Lower BOund (ELBO), which is defined as:

$$\text{ELBO} = \mathbb{E}_\xi[\log\, p(X, W)] - \mathbb{E}_\xi[\log\, q(W)], \tag{5}$$

with $p(X, W)$ as the joint distribution of the observed data and the latent variables, and $q(W)$ the variational distributions of the latent variables.

The most common approach to obtain an approximating distribution when applying a variational Bayes approach is mean field approximation, where the approximating distribution is assumed to factorize over some partition of the parameters. This is the approach that we adopt for the coefficient vector $\mathbf{B}$. However, a mean field approach for the elements of the precision matrix $\mathbf{\Omega}$ is not appropriate, due to the dependence

among the parameters induced by the fact that this matrix is constrained to be symmetric and positive semi-definite. For this reason, the choice of an appropriate approximating distribution for the precision matrix is an open research question. To circumvent this issue, similar to Miao et al. (2020), we adopt a hybrid VI algorithm, with an Expectation-Minimization (EM) step to estimate $\mathbf{\Omega}$ and $\boldsymbol{\delta}$.

Specifically, for $\mathbf{B}$ we use the mean field variational distributions $q(\mathbf{B}, \boldsymbol{\gamma}) = \prod_{j=1}^{p} \prod_{k=1}^{q} q(B_{k,j}, \gamma_{k,j} \mid \phi_{k,j}, \mu_{k,j}, \sigma_{k,j})$, where

$$q(B_{k,j}, \gamma_{k,j} \mid \phi_{k,j}, \mu_{k,j}, \sigma_{k,j}) = \begin{cases} \phi_{k,j} \text{Normal}(B_{k,j} \mid \mu_{k,j}, \sigma_{k,j}) \text{ if } \gamma_{k,j} = 1, \\ (1 - \phi_{k,j}) \delta_0(B_{k,j}) \qquad\qquad\qquad \text{otherwise,} \end{cases}$$

with free parameters $\boldsymbol{\xi} = \{\phi_{k,j}, \mu_{k,j}, \sigma_{k,j}\}$. We then define the ELBO as

$$\begin{aligned} \text{ELBO} = \mathbb{E}_{\xi} \Bigg[\log \prod_{j=1}^{p} \Bigg(p(\mathbf{X}_i \mid \mathbf{Z}_i) p(\mathbf{Z}_i \mid \mathbf{B}_0, \mathbf{M}_i, \mathbf{B}, \mathbf{\Omega}) \prod_{k=1}^{q} \Big(p(B_{k,j} \mid \gamma_{k,j}, \nu_B^2) p(\gamma_{k,j} \mid \theta_{\gamma_j}) p(\theta_{\gamma_j} \mid a_\gamma, b_\gamma) \Big) \Bigg) \\ p(\mathbf{\Omega} \mid \boldsymbol{\delta}, \nu_1, \nu_0, \lambda, \tau) p(\delta_{i,j} \mid \pi) p(\pi \mid a_\pi, b_\pi) \Bigg] - \mathbb{E}_{\xi} \left[\log \prod_{j=1}^{p} \prod_{k=1}^{q} q(B_{k,j}, \gamma_{k,j} \mid \phi_{k,j}, \mu_{k,j}, \sigma_{k,j}) \right], \end{aligned}$$

where the first expectation is equivalent to $\mathbb{E}_{\xi}[\log p(X, W)]$ and the second expectation is $\mathbb{E}_{\xi}[\log q(W)]$ of equation (5), for $W = \{\mathbf{Z}, \mathbf{B}_0, \mathbf{B}, \mathbf{\Omega}, \boldsymbol{\delta}, \pi, \boldsymbol{\gamma}, \boldsymbol{\theta}_\gamma\}$.

The hybrid scheme we use to maximize the ELBO, where the first part is a VI step and the second part is an EM step, is described in detail in the following subsections. In the VI step we update the free parameters, $\boldsymbol{\xi}$, by setting the partial derivative of the ELBO with respect to the desired parameters equal to zero. This maximizes the ELBO with respect to $\boldsymbol{\xi}$. We then further maximize the ELBO by finding the optimal values for the remainder of the latent parameters. For this, we rely on an EM step, by treating $\boldsymbol{\delta}$ as latent parameters and taking the expectation of the ELBO with respect to $\boldsymbol{\delta}$, or equivalently setting

$$Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)}, \boldsymbol{\xi}^{(t)}) = \mathbb{E}_{\boldsymbol{\delta}}[ELBO],$$

and optimizing $Q(\boldsymbol{\theta} \mid \boldsymbol{\theta}^{(t)}, \boldsymbol{\xi}^{(t)})$ by finding the maximum a posteriori (MAP) estimate of the remaining parameters $\boldsymbol{\theta} = \{\mathbf{Z}, \mathbf{B}_0, \mathbf{\Omega}, \pi, \boldsymbol{\theta}_\gamma\}$. The resulting algorithm, which we call SINC, is described in Algorithm 1. As with traditional EM and VI schemes, parameter updates at each iteration are made with the most current estimates of all other parameters. The algorithm results in MAP estimates for the parameters in $\boldsymbol{\theta}$. Additionally, since no uncertainty about these parameters is used in the updates of the other parameters, the proposed algorithm is only suitable for point estimation.

Our proposed hybrid algorithm builds upon the similarities between the VI and EM algorithms. As noted in Blei et al. (2017), the first term of equation (5) is the expected complete log likelihood, which is optimized by the EM algorithm. Since no variational distributions are proposed for the parameters in $\boldsymbol{\theta}$, updating those parameters is achieved by optimizing $\log p(W, Y)$ in equation (5). As an alternative perspective to highlight the similarity, we could say that we have assigned a point mass as our variational distribution for

Algorithm 1 SINC Algorithm

procedure INITIALIZE($\mathbf{Z}, \mathbf{\Sigma}, \mathbf{\Omega}$)
  Set $\mathbf{Z} = \log(\boldsymbol{\alpha} + 1)$
  $\mathbf{\Sigma} = \tilde{\mathbf{Z}}^T\tilde{\mathbf{Z}}/\mathrm{N}$ ▷ $\tilde{\mathbf{Z}}$ is $\mathbf{Z}$, column centered
  $\mathbf{\Omega} = \mathbf{\Sigma}^{-1}$
while ELBO has not converged **do**
  procedure VI-STEP($\mathbf{B}$,$\boldsymbol{\phi}$)
    for j in $1:p$ **do**
      while $\boldsymbol{\mu}_j^{(t)}$, $\boldsymbol{\sigma}_j^{(t)}$, $\boldsymbol{\phi}_j^{(t)}$ not converged **do**
        Update $\boldsymbol{\mu}_j$ ▷ Equation (6)
        Update $\boldsymbol{\sigma}_j$ ▷ Equation (7)
        Update $\boldsymbol{\phi}_j$ ▷ Equation (8)
    Set $B_{k,j} = \mu_{k,j}\phi_{k,j}$
  procedure E-STEP($\boldsymbol{\delta}$)
    Update $\mathbf{p}^*$ ▷ Equation (9)
    Update $\mathbf{d}^*$ ▷ Equation (10)
  procedure M-STEP($\mathbf{B}_0$,$\mathbf{\Omega}, \theta_\gamma, \pi, \mathbf{Z}$)
    for j in $1:p$ **do**
      Update B_{0j} ▷ Equation (11)
    while $\mathbf{\Omega}^{(t)}$ not converged **do**
      Column-wise update of $\mathbf{\Omega}$ ▷ Equations (12,13)
    for j in $1:p$ **do**
      Update $\boldsymbol{\theta}_{\gamma_j}$ ▷ Equation (14)
    Update π ▷ Equation (15)
    Update τ, if applicable ▷ Equation (16)
    Update $\mathbf{Z}$ ▷ Maximize Equation (17) using L-BFGS

these latent parameters. Optimizing the ELBO would then lead to the same result, since taking the partial derivative of the ELBO with respect to the variables with point mass variational distributions would result in optimizing $\mathbb{E}_\xi \log[p(W, Y)]$. By stating the algorithm in this way, we can interpret our approach as a proper VI scheme, solved via an EM step similar to Titsias and Lázaro-Gredilla (2011), which affords us the confidence of VI guarantees of previous literature (Blei et al., 2017).

3.1 VI Step

Here we use a Variational Inference step to estimate the regression coefficients $\mathbf{B}$ by updating the free parameters $\mu_{k,j}$, $\sigma_{k,j}$, and $\phi_{k,j}$, where $\mu_{k,j}$ and $\sigma_{k,j}$ are the mean and variance, respectively, of $B_{k,j}$ when $\gamma_{k,j} = 1$, and $\phi_{k,j}$ is the probability $\gamma_{k,j} = 1$ resulting in $B_{k,j} \neq 0$. Following the work of Titsias and Lázaro-Gredilla (2011) and Carbonetto and Stephens (2012) the free parameters can then be updated as

$$\mu_{k,j} = \frac{\sigma_{k,j}}{\sigma_j^*}\Big[\big\{\mathbf{M}^T(\mathbf{Z}_j - B_{0j})\big\}_k - \sum_{l\neq k}(\mathbf{M}^T\mathbf{M})_{lk}\phi_{l,k}\mu_{l,k}\Big], \tag{6}$$

$$\sigma_{k,j} = \frac{\sigma_j^*}{(\mathbf{M}^T\mathbf{M})_{k,k} + 1/\nu_B}, \tag{7}$$

$$\phi_{k,j} = \mathrm{Logit}^{-1}\big(\log\frac{\theta_{\gamma_j}}{1-\theta_{\gamma_j}} - \frac{1}{2}\log\frac{\sigma_{j,k}}{\nu_B\sigma_j^*} + \frac{\mu_{i,j}^2}{2\sigma_{j,k}^2}\big). \tag{8}$$

Updating each column of $\mathbf{B}$ can then be done independently and, when resources are available, these updates can be done in parallel. While updating a column of $\mathbf{B}_j$, each component of $\boldsymbol{\mu}_j$, $\boldsymbol{\sigma}_j$, $\boldsymbol{\phi}_j$ is updated given all other components. This component-wise update of $\boldsymbol{\mu}_j$, $\boldsymbol{\sigma}_j$, $\boldsymbol{\phi}_j$ is repeated until ELBO$(\boldsymbol{\mu}_j, \boldsymbol{\sigma}_j, \boldsymbol{\phi}_j)$ has converged. Once all $\boldsymbol{\mu}$ and $\boldsymbol{\phi}$ have been updated, the individual elements of $\mathbf{B}$ are assigned $\mathbb{E}(B_{k,j}) = \mu_{k,j}\phi_{k,j}$. Once the SINC algorithm has converged, as common in variational spike-and-slab literature (Huang et al., 2016; Miao et al., 2020), we set $B_{k,j} = \mu_{k,j}$ if $\phi_{k,j} > 0.5$ and $B_{k,j} = 0$ if $\phi_{k,j} \leq 0.5$. The threshold of 0.5 is equivalent to selecting the median model of Barbieri and Berger (2004) and can be adjusted to include or exclude more covariates, but the threshold of 0.5 is the most commonly used.

3.2 E Step

In this step, we focus on updates to the edge inclusion parameter $\delta_{i,j}$. For the first step we take the expectation of the posterior distribution, treating $\boldsymbol{\delta}$ as the latent variable. We define $Q(\boldsymbol{\Theta} \mid \boldsymbol{\Theta}^{(t)}, \boldsymbol{\xi}^{(t)})$ as $E_{\boldsymbol{\delta}|\boldsymbol{\Omega}^{(t)},\pi,\mathbf{X}}\big(\log p(\mathbf{Z}, \boldsymbol{\Omega}, \mathbf{B}, \mathbf{B}_0, \pi \mid \mathbf{X}, \mathbf{M}) \mid \boldsymbol{\Omega}^{(t)}, \pi^{(t)}, \mathbf{X}\big)$. Following the results shown in Li and McCormick (2019), the E step can be broken into two steps:

$$E_{\boldsymbol{\delta}|\boldsymbol{\Omega}^{(t)},\pi,\mathbf{X}}[\delta_{i,j}] = p^*_{ij} \equiv \frac{a_{ij}}{a_{ij} + b_{ij}} \tag{9}$$

$$E_{\boldsymbol{\delta}|\boldsymbol{\Omega}^{(t)},\pi,\mathbf{X}}\Big[\frac{1}{\nu_0^2\tau^{-1}(1-\delta_{i,j}) + \nu_1^2\tau^{-1}\delta_{i,j}}\Big] = d^*_{ij} \equiv \left(\frac{1-p^*_{ij}}{\nu_0^2} + \frac{p^*_{ij}}{\nu_1^2}\right)\tau^{-1}, \tag{10}$$

where $a_{ij} = p(\omega_{i,j} \mid \delta_{i,j} = 1)\pi$ and $b_{ij} = p(\omega_{i,j} \mid \delta_{i,j} = 0)(1-\pi)$, and (i,j) is the $(i,j)^{th}$ entry of the precision matrix, where i and $j \in \{1, \ldots, P\}$.

3.3 M Step

The remainder of the unknown parameters can be found by maximizing the posterior distribution with regards to each of the parameters we are interested in. Here, we first update the column-wise centering parameters B_{0j} independently as

$$B_{0j} = \frac{\sum_{i=1}^{N} Z_{i,j} - \mathbf{M}_i\mathbf{B}_j}{n}. \tag{11}$$

Next, we update the precision matrix, $\boldsymbol{\Omega}$. Following Li and McCormick (2019) and Wang (2015) the conditional distribution of each column of $\boldsymbol{\Omega}$ can be found in closed form. For this, let

$$\boldsymbol{\Omega} = \begin{pmatrix} \boldsymbol{\Omega}_{11} & \boldsymbol{\omega}_{12} \\ \boldsymbol{\omega}_{21}^T & \omega_{22} \end{pmatrix}, (\mathbf{Z} - (\mathbf{MB} + \mathbf{B}_0))^T(\mathbf{Z} - (\mathbf{MB} + \mathbf{B}_0)) = \begin{pmatrix} \mathbf{S}_{11} & \mathbf{s}_{12} \\ \mathbf{s}_{21} & s_{22} \end{pmatrix}.$$

Then, the conditional distributions are

$$\begin{aligned} \boldsymbol{\omega}_{12} &\sim \text{Normal}(-\mathbf{C}\mathbf{s}_{12}, \mathbf{C}), \\ \omega_{22} - \omega_{12}\boldsymbol{\Omega}_{11}\omega_{12} &\sim \text{Gamma}(1 + \frac{n}{2}, \frac{\lambda + s_{22}}{2}), \\ \mathbf{C} &= ((s_{22} + \lambda)\boldsymbol{\Omega}^{-1} + \text{diag}(\nu_{\boldsymbol{\delta}_{12}}))^{-1}. \end{aligned} \tag{12}$$

We can then do a column-by-column update as

$$\begin{aligned}\boldsymbol{\omega}_{12} &= -((s_{22}+\lambda)(\boldsymbol{\Omega})^{-1} + \text{diag}(d^*_{ij}))^{-1})\mathbf{s}_{12}, \\ \omega_{22} &= \boldsymbol{\omega}_{12}\boldsymbol{\Omega}_{11}\boldsymbol{\omega}_{12} + \frac{n}{\lambda + s_{22}}.\end{aligned} \tag{13}$$

The point estimates of θ_γ and π are also updated as

$$\theta_\gamma = \frac{\sum \phi_{i,j} + a_\gamma - 1}{p + a_\gamma + b_\gamma - 2}, \tag{14}$$

$$\pi = (a_\pi + \sum_{i<j} \delta_{ij} - 1)/(a_\pi + b_\pi + \frac{p(p-1)}{2} - 2). \tag{15}$$

If using the adaptive scale parameter, τ, an additional update is done by setting

$$\tau = \frac{a_\tau - 1 + 0.5(p \times (p-1))}{b_\tau - 2 + 0.5\sum_{i<j} \omega^2_{i,j} d^*_{i,j}}. \tag{16}$$

Finally, the matrix of latent variables can be estimated by finding a point estimate for each entry of the matrix. This is done by updating each row of the matrix independent of the others. As shown in Yang et al. (2017), the objective function to optimize with respect to $\mathbf{Z}$ is

$$\begin{aligned}\log\, P(\mathbf{Z} \mid \mathbf{X}, \mathbf{M}, \mathbf{B}_0, \mathbf{B}, \boldsymbol{\Omega}) = -\frac{1}{n}\sum_{i=1}^{n}\left[\sum_{j=1}^{p}\tilde{\Gamma}(\alpha_{ij} + x_{ij}) - \tilde{\Gamma}(s(\boldsymbol{\alpha}_i) + s(\mathbf{X}_i)) - \sum_{j=1}^{p}\tilde{\Gamma}(\alpha_{ij}) + \tilde{\Gamma}(s(\boldsymbol{\alpha}_i))\right] - \\ \frac{1}{2}\log \mid \boldsymbol{\Omega} \mid + \frac{1}{2n}\sum_{i=1}^{n}(\mathbf{Z}_i - (\mathbf{B}_0 + \mathbf{M}_i\mathbf{B}))\boldsymbol{\Omega}(\mathbf{Z}_i - (\mathbf{B}_0 + \mathbf{M}_i\mathbf{B})),\end{aligned} \tag{17}$$

where $\tilde{\Gamma}$ is the log-gamma function, and $s(x_i) = \sum_{j=1}^{p} x_{ij}$. To accomplish optimization of each $\mathbf{Z}_i$ we use the limited-memory quasi-Newton (L-BFGS) algorithm, which is a quasi-newton gradient descent method that makes use of the inverse gradient to direct where to search through the variable space.

For posterior inference, we iterate through the VI step, which iterates between updating $\phi_{k,j}$, $\mu_{k,j}$, and $\sigma_{k,j}$, and the E and M steps, which update $\boldsymbol{\Omega}$ one column at a time, until the algorithm has converged. For both the VI and the M steps, we run each of those steps until the respective parameter estimates have converged. We determine the algorithm to have converged if the ELBO changes by less than a predefined tolerance from one iteration to the next. To obtain a selected network and set of covariates based on these posterior estimates, we select edges i, j with $p^*_{i,j}$ in equation (9) ≥ 0.50, and covariate associations with $\phi^*_{k,j}$ in equation (8) ≥ 0.50. In practice, both the $p^*_{j,k}$ and $\phi_{k,j}$ values, which reflect the posterior probabilities for the selection of edges and covariates, tend to converge to values close to 0 or close to 1. A similar trend has been noted by Kook et al. (2020), who reported that the variational parameters for the marginal posterior probabilities of inclusion tended to become more widely separated as the algorithm converges.

4 Simulation Study

We now compare the performance of our method to existing approaches in a simulation setting designed to mimic the application to microbiome data described later in the paper.

4.1 Simulation Setup

Simulated data were generated with the following steps. First the covariates, $\mathbf{M}$, were generated from a normal distribution MVNorm$(\mathbf{0}, \mathbf{I}_p)$, and subsequently scaled. The values of the regression coefficients $\mathbf{B}$, related to $\mathbf{M}$, were then sampled. Each element of $\mathbf{B}$, $B_{k,j}$, was assigned either a random value between $[-1, -0.5]$ with probability 0.1, a value in $[0.5, 1]$ with probability 0.1, or else 0. Each B_{0j} was then sampled from the interval $[6, 8]$ with probability 0.2, and from $[2, 4]$ with probability 0.8. This allowed for some variables to have larger counts and others to be sparser, as common in microbiome data. Simulated counts were then sampled by first drawing $\mathbf{Z}$ from MVNorm$(\mathbf{MB} + \mathbf{B}_0, \mathbf{\Omega})$, with $\mathbf{\Omega}$ as described below, and then assigning $\boldsymbol{\alpha}$ as $\exp(\mathbf{Z})$. Finally, $\mathbf{h}_i$ was sampled as a random draw from Dirichlet$(\boldsymbol{\alpha}_i)$, and $\mathbf{X}_i$ drawn from a Multinomial$(\mathbf{h}_i, \text{nint}(n_i))$, with n_i generated from a Normal$(3000, 250)$, allowing for samples to have different numbers of total counts, where nint() represents the nearest integer function. We set $p = 100$, $q = 50$ and $n = 300$.

To explore performance for a range of possible network structures, we simulated a variety of configurations. These networks were created using the R package `huge` (Jiang et al., 2019). For this simulation, we used a band, cluster, hub, and random graph structure. An example of what these networks look like can be seen in Figure 1. Band graphs and random graphs are common test cases for network learning, while the hub and cluster graphs capture some aspects of biological networks, such as highly connected nodes and community structure (Girvan and Newman, 2002). The probability of an edge in the network was set to 0.025 for the random graph and 0.30 for each cluster in the cluster graph. The bandwidth in the graph was set to 3 and the number of hubs in the hub graph was set to 3. The precision matrix $\mathbf{\Omega}$ used to generate the simulated data was also constructed using the function `huge.generate` of the R package `huge` Jiang et al. (2019). Parameters `v` and `u` of `huge.generate`, which control the off-diagonal elements of the precision matrix and magnitude of the partial correlations, were set to 1 and 0.0001, respectively.

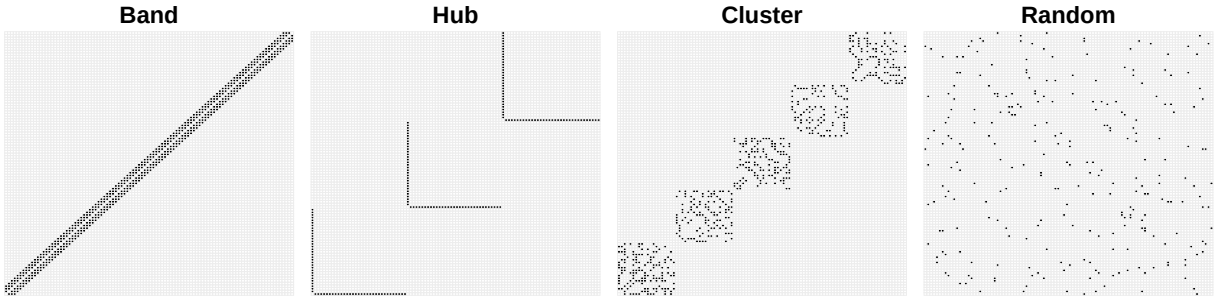


Figure 1: Simulation study: Example networks used in the simulation studies. Black boxes represent true edges, and light gray boxes correspond to no edge. For the cluster and random graphs, the actual networks that generated the data were different for each simulated data set, but each simulation kept the blocked shape or random network, respectively.

The results we report below for our proposed model were obtained with the following hyperparameter

settings. Fairly non-informative priors were set by choosing $a_\gamma = b_\gamma = 2$ in the prior probability of inclusion (2) of each covariate. The same setting was used for the hyperparameters a_π and b_π in Equation (4), which determines the prior probability of an edge being included. The standard deviation of the prior on the selected regression coefficients, ν_B in equation (2), was set to 1. Following guidelines given by Wang (2015) and Li and McCormick (2019), we set $\nu_1 = 10$, the standard deviation in prior (3) on the off-diagonal precision matrix entries corresponding to selected edges, and fit the model across a grid of values, ranging from 0.0001 to 0.1, of ν_0, the prior standard deviation of off-diagonal elements of the precision matrix corresponding to non selected edges. We then chose the final model by using the ν_0 value that gave sparsity closest to 0.10. This sparsity level was selected arbitrarily, and did not result in any specific advantage for our method, as none of the simulation networks had sparsity equal to 0.10. Finally, the rate parameter λ of equation (3), which appears in the prior on diagonal elements of the precision matrix, was set to 150. We also compare the model when the scaling parameter, τ, is learned, and use a Gamma(2,2) prior to do so. We comment on the sensitivity of the results to parameter choices in Section 4.3 below.

4.2 Simulation Results

We compare the performance of our method to several existing alternative approaches in terms of accuracy in network estimation and covariate selection. For comparison, we used mLDM (Yang et al., 2017), which is specifically designed for estimating networks of compositional data while controlling for covariates, and mSSL-DPE (Deshpande et al., 2019), which we applied to the centered log ratio (CLR) transformed version of the simulated count data, a common method to account for the compositionality (Fang et al., 2017; Kurtz et al., 2015). For network estimation, we also considered SpiecEasi (Kurtz et al., 2015), which applies graphical LASSO to the CLR-transformed data. These methods were applied by using the default selection criteria in their respective R packages. To more precisely characterize factors contributing to the network estimation performance of the SINC method, we apply a version of SINC with the covariate effects $\mathbf{B}$ constrained to be $\mathbf{0}$. A comparison of the results from this constrained version of SINC to those of SpiecEasi reflects the performance advantages arising from differences in the network estimation procedure, while comparison to the full unconstrained SINC method provides quantitative insight on the benefit of simultaneous estimation of covariate effects on network recovery. Similarly, for the comparison of variable selection accuracy, we apply a constrained version of SINC with $\mathbf{\Omega}$ fixed to the identity matrix. Comparison of the results from this approach to those of the full unconstrained SINC method illustrates the added value of accounting for the residual covariance in estimation of $\mathbf{B}$.

We report results in terms of true positive rate (TPR), false positive rate (FPR), F1 score, and Matthew's correlation coefficient (MCC). For edge selection, we also report the area under the curve (AUC). This was calculated, for the SINC method, over a grid of ν_0 values, and for the mLDM and mSSL-DPE methods by using the LASSO penalization parameter for the coefficients associated with the best selected graph, and then varying the graph penalization parameter over a grid of values. The AUC for SpiecEasi was calculated

Network estimation accuracy

	TPR	FPR	F1	MCC	AUC	TPR	FPR	F1	MCC	AUC
	Random					Hub				
mSSL-DPE	0.000	0.015	0.000	-0.019	0.499	0.001	0.024	0.001	-0.021	0.450
SpiecEasi	0.013	**0.009**	0.018	0.005	0.563	0.011	**0.009**	0.015	0.003	0.528
mLDM	0.277	0.181	0.067	0.039	0.560	**0.440**	0.248	0.080	0.069	0.602
SINC ($\mathbf{B} = 0$)	0.276	0.225	0.056	0.020	0.542	0.253	0.241	0.038	0.004	0.507
SINC ($\tau = 1$)	0.420	0.093	0.167	0.169	0.750	0.175	0.096	0.059	0.037	0.613
SINC (τ learned)	**0.598**	0.091	**0.237**	**0.263**	**0.838**	0.294	0.094	**0.098**	**0.094**	**0.689**
	Cluster					Band				
mSSL-DPE	0.005	0.0181	0.008	-0.0230	0.446	0.003	0.022	0.004	-0.032	0.468
SpiecEasi	0.013	**0.010**	0.0182	0.005	0.563	0.012	**0.009**	0.020	0.007	0.544
mLDM	0.232	0.155	0.126	0.051	0.544	0.440	0.248	0.080	0.069	0.602
SINC ($\mathbf{B} = 0$)	0.272	0.229	0.110	0.024	0.534	0.294	0.242	0.114	0.028	0.533
SINC ($\tau = 1$)	0.294	0.084	0.223	0.169	0.678	0.311	0.088	0.230	0.175	0.685
SINC (τ learned)	**0.411**	0.080	**0.306**	**0.261**	**0.741**	**0.446**	0.089	**0.312**	**0.269**	**0.737**

Table 1: Simulation results for network selection. mSSL-DPE refers to the method of Deshpande et al. (2019), SpiecEasi to the method of Kurtz et al. (2015), mLDM to Yang et al. (2017), SINC ($\mathbf{B} = 0$) to the modified version of the proposed model with the covariate estimates fixed, and SINC to the proposed model. Random, Hub, Cluster, and Band refer to the underlying shape of the network, as illustrated in Figure 1

Variable selection accuracy

	TPR	FPR	F1	MCC	TPR	FPR	F1	MCC
	Random				Hub			
mSSL-DPE	0.840	**0.000**	0.912	0.898	0.868	**0.000**	0.929	0.915
mLDM	0.609	0.003	0.751	0.734	0.612	0.003	0.754	0.738
SINC ($\mathbf{\Omega} = \mathbf{I}$)	0.808	0.003	0.871	0.889	0.813	0.001	0.887	0.894
SINC ($\tau = 1$)	0.914	0.001	0.943	0.953	0.925	0.000	**0.952**	**0.960**
SINC (τ learned)	**0.917**	0.000	**0.947**	**0.956**	**0.926**	0.001	**0.952**	**0.960**
	Cluster				Band			
mSSL-DPE	0.837	**0.000**	0.910	0.900	0.853	**0.000**	0.920	0.906
mLDM	0.605	0.003	0.747	0.730	0.597	0.003	0.741	0.725
SINC ($\mathbf{\Omega} = \mathbf{I}$)	0.791	0.006	0.854	0.873	0.808	0.000	0.887	0.893
SINC ($\tau = 1$)	0.908	0.000	0.941	0.951	0.921	0.001	0.947	0.957
SINC (τ learned)	**0.910**	0.000	**0.942**	**0.952**	**0.922**	0.000	**0.950**	**0.959**

Table 2: Simulation results for covariate selection. mSSL-DPE refers to the method of Deshpande et al. (2019), mLDM to the method of Yang et al. (2017), SINC ($\mathbf{\Omega} = \mathbf{I}$) to the modified version of the proposed model with the precision matrix fixed, and SINC to the proposed model. SpiecEasi is omitted from the comparison, as it does not perform selection or adjustment for covariates. Random, Hub, Cluster, and Band refer to the underlying shape of the network, as illustrated in Figure 1

by varying the penalization parameter. Tables 1 and 2 show the results for network estimation and variable selection, respectively. From Table 1, we can see that mSSL-DPE and SpiecEasi are generally not competitive in terms of their performance, with low F1, MCC, and AUC values. This is likely because mSSL-DPE was not designed for compositional data, and SpiecEasi is not able to account for the effects of the covariates on the counts. We also see that mLDM performs better than the other two methods but is still outperformed by the proposed model, which does better in all F1, MCC, and AUC scores across all of the network structures except the hub structure. Finally, we see that when the proposed model does not control for additional covariates, the network estimation scores decrease and are comparable to the other methods. Across all methods, SINC while learning τ performed best in all network structures in terms of F1, MCC, and AUC. The performance metrics for SINC are pretty similar across all network types, though the best performance is achieved on the random graph, while the hub and cluster settings are more challenging. From Table 2 we can see that mSSL-DPE performs quite well in selecting the covariates of interest. In fact, its performance is very close to the proposed model, SINC, on all metrics. Similarly, the proposed model while holding the estimated network and precision matrix fixed performs well for coefficient estimation, but does not do as well as mSSL-DPE and the full version of the proposed model. Additionally, we did not see any significant difference in performance in the full SINC models when τ is fixed or learned. SpiecEasi is not included in Table 2, as it is not able to select or adjust for relevant covariates, which is a limitation of the method.

We did not compare our model to MCMC approaches because of the computational complexity resulting from a lack of conjugacy. We did, however, experiment by using a Monte Carlo draw to update $\mathbf{\Omega}$ at each iteration of SINC, and found that the point estimates of SINC without a Monte Carlo step were very close to the mean of the Monte Carlo draws.

4.3 Influence of parameters

An advantage of using spike-and-slab priors for covariate and network edge selection, over penalized methods, is given by the flexible level of sparsity induced on the regression coefficients and the precision matrix entries. For example, Li and McCormick (2019) show that $d^*_{i,j}$ from equation (10) is comparable to the penalty parameter, λ, in the graphical LASSO (Dempster, 1972). However, $d^*_{i,j}$ is unique to each edge and is adaptively learned from the data. In Figure 2 we show this advantage over penalized methods by plotting the estimated coefficients and precision matrix values, for a smaller simulation scenario with $p = 10$, $q = 15$, and $n = 5000$, using SINC (with τ fixed at 1) and the penalization based method mLDM, while varying the sparsity-inducing parameters. The top-left plot shows the estimated $\mathbf{B}$ coefficients by the proposed model when increasing the variance parameter ν_B of the spike-and-slab prior in equation (2). The top-right plot shows the estimated $\mathbf{B}$ coefficients via mLDM when increasing the LASSO penalty parameter. The bottom-left plot shows the estimated off-diagonal values of the precision matrix $\mathbf{\Omega}$ when increasing the variance parameter ν_0 in equation (3) and the bottom-right plot shows the estimated off-diagonal estimates of the precision matrix when increasing the graphical LASSO penalty parameter. In all plots, red lines

correspond to true associations in the simulated data, and black lines correspond to coefficients representing no underlying association. The flat trend in the red lines of the plots related to SINC shows that the estimated covariate effects and precision matrix entries corresponding to true associations are stable, while for mLDM, depicted at bottom, they get shrunken to zero as the penalty parameters increase.

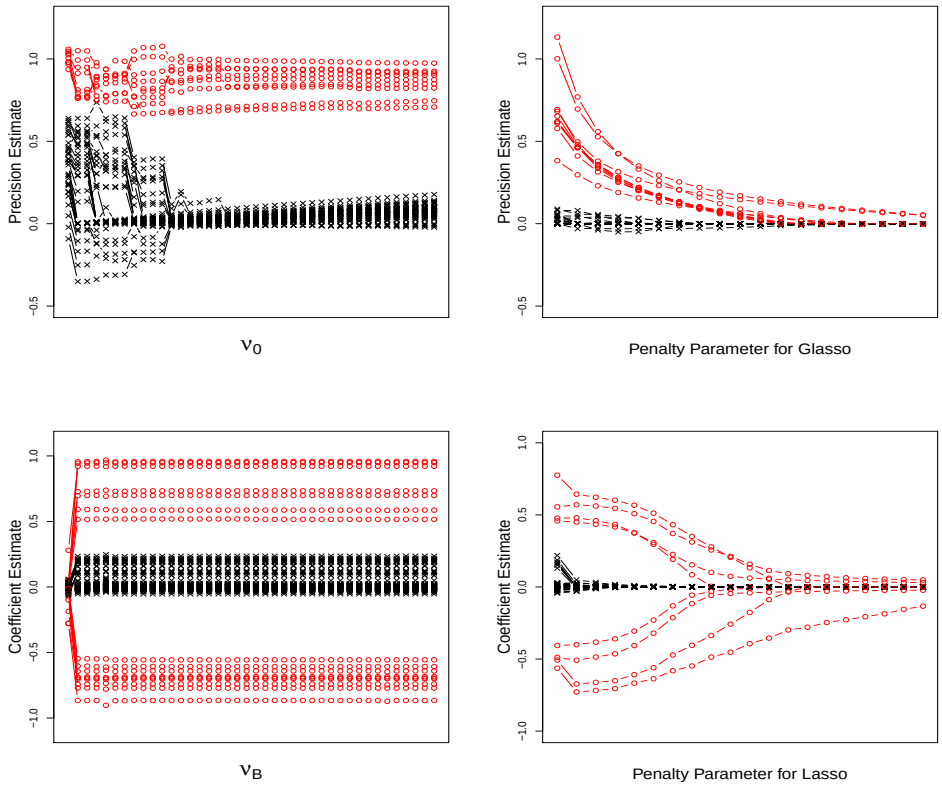


Figure 2: Simulation study: The top-left plot shows the estimated $\mathbf{B}$ coefficients by the proposed model when increasing the variance parameter ν_B of the spike-and-slab prior (2). The top-right plot shows the estimated $\mathbf{B}$ coefficients via mLDM when increasing the LASSO penalty parameter. The bottom-left plot shows the estimated off-diagonal values of the precision matrix $\mathbf{\Omega}$ when increasing the variance parameter ν_0 in (3) and the bottom-right plot shows the estimated off-diagonal estimates of the precision matrix when increasing the graphical LASSO penalty parameter. In all plots, `-o-` lines correspond to true associations in the simulated data, and `-x-` lines correspond to coefficients representing no underlying association.

We conclude this section by providing some comments on the sensitivity of the results to the choice of the hyperparameters. As shown in Figure 2, with sufficient data, the estimates of $\mathbf{\Omega}$ and $\mathbf{B}$ are stable for increasing values of ν_0, in the prior of equation (3), and ν_B, in the prior of equation (2), respectively. These parameters, however, affect the sparsity of the selection. In particular, as ν_0 increases, holding all other parameters constant, the selected network becomes sparser. Similarly, as ν_1, which appears in the prior of equation (3), increases, holding ν_0 constant, the network sparsity increases. In recent work using this type of

mixture prior, Li and McCormick (2019) and Ročková and George (2014) have suggested holding ν_1 constant while varying ν_0. It should be apparent, then, that increasing ν_B while holding all other parameters constant decreases the number of selected coefficients, as increasing ν_B is analogous to increasing ν_1. The remainder of the hyperparameters influence sparsity as well, but to a lesser extent. For example, changing a_π and b_π in the prior given in equation (4) to put more weight on larger values of π results in sparser networks. Similarly, selecting a_γ and b_γ in the prior of equation (2) to reflect a stronger prior belief in larger $\boldsymbol{\theta}_\gamma$ values results in an increase in the number of selected coefficients. Since π and $\boldsymbol{\theta}_\gamma$ are both updated at each iteration of the SINC algorithm, selecting relatively non-informative priors, such as the ones used in the simulations of $a_\gamma = b_\gamma = 2$, allows the sparsity levels to be primarily controlled by ν_0 and ν_B. Alternative choices that would also be appropriate include $a_\gamma = b_\gamma = 1$, a more non-informative setting corresponding to a uniform prior on the unit interval, or $a_\gamma = 1$ and $b_\gamma = p$, which would more strongly favor sparsity, as discussed in Ročková and George (2014). We found that our variable selection results were not overly sensitive to the choice of these parameters. Increasing λ also increases the network sparsity because it changes the scale of the estimated $\boldsymbol{\Omega}$ values by making them smaller. Appropriate λ values need to be selected based on the scale of the data that is being used.

5 Application to a study of the vaginal microbiome in pregnancy

In this section, we apply our proposed method to data from the Multi'Omic Microbiome Study - Pregnancy Initiative (MOMS-PI), an NIH-funded study aimed at characterizing the microbiome and its role in shaping maternal and infant health. Previous research has demonstrated that immune and metabolic changes during pregnancy reshape the microbiome, which undergoes large shifts during the course of pregnancy. The vaginal microbiome in particular has been shown to change early in pregnancy (Serrano et al., 2019) and be predictive of pregnancy outcomes such as preterm birth (Fettweis et al., 2019).

The MOMS-PI study involved following pregnant women throughout pregnancy and for a short term after childbirth. Participants in the MOMS-PI study were asked to provide samples from the mouth, skin, vagina and rectum. Multiple omic technologies were used to process the collected samples including microbiome profiling, metabolomics, and quantification of cytokine abundances via immunoproteomics. Cytokines, in particular, are one mechanism by which the host regulates the composition of the vaginal microbiome. The data was obtained from the R package `HMP2Data` and consists of 596 subjects that were sampled across multiple visits. For our analysis, we focus on samples collected at the first baseline visit. Of the 596 subjects, 225 subjects had both the microbiome and cytokine profiling of the vagina available at this time point. We consider these 225 subjects in the analysis. To avoid the inclusion of very rare taxa, the OTUs were filtered for inclusion in the analysis using the following rule: the absolute abundance of an OTU had to be greater than 1 for at least 10 percent of the subjects, resulting in 90 OTUs. All 29 cytokines profiled were included as covariates. For the analysis, the cytokine data was transformed to the log scale and centered.

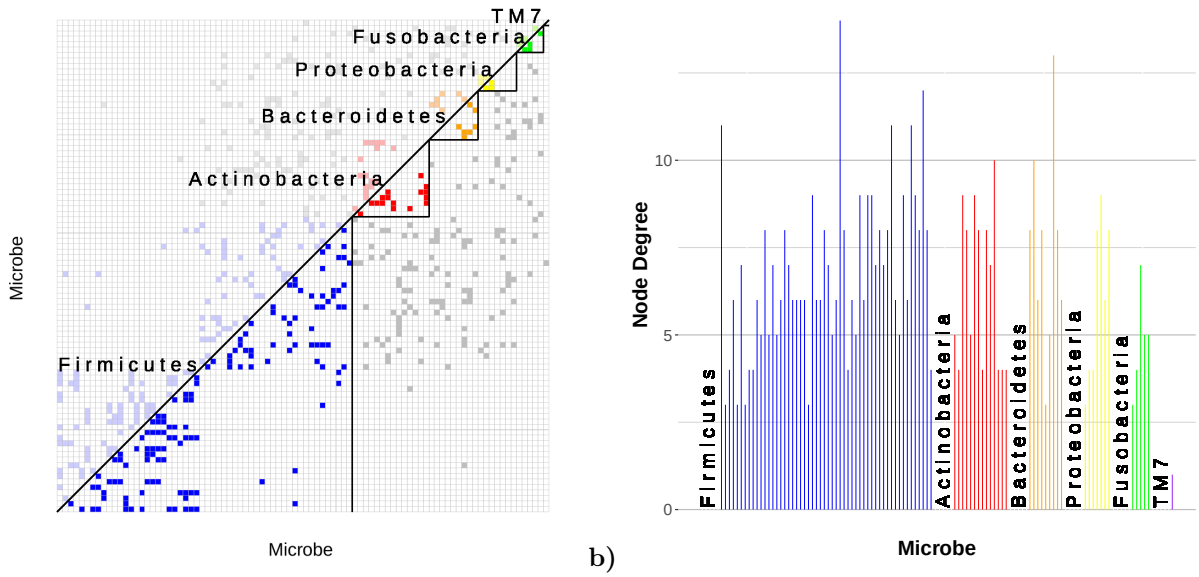
We applied the SINC method to estimate the interaction between vaginal microbes, as well as the interplay between vaginal cytokines and microbial abundances. We used the same hyperparameter settings as in the simulation: $\nu_B = 1$, $a_\gamma = 2$, $b_\gamma = 2$, $\nu_1 = 10$, $\lambda = 150$, $a_\pi = 2$, $b_\pi = 2$, set $\nu_0 = 0.01$, a value that, in the simulations, achieved a sparsity level of 0.10, and fixed τ to 1. Since we are controlling for cytokine counts when estimating the microbiome network, we are more confident in the selection as we do not expect to select an edge between two microbes that may be related only via their common dependence on a cytokine.

5.1 MOMS-PI Results

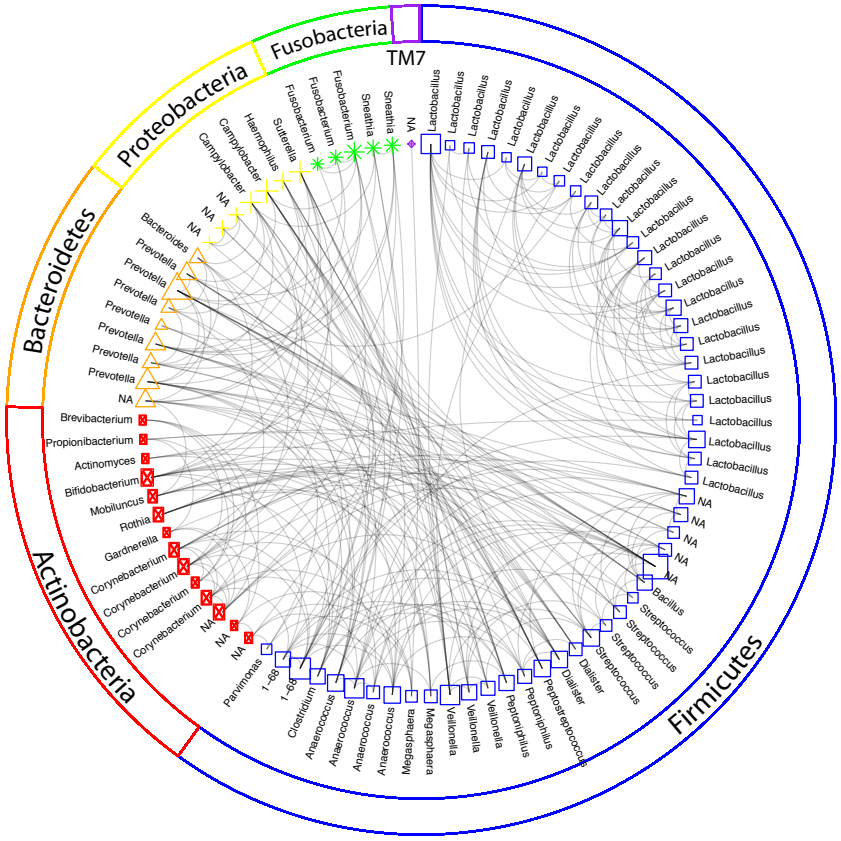
Figure 3a shows the adjacency matrix of the microbial network inferred from the MOMS-PI data, with filled boxes representing selected edges, together with a plot of the number of edges for each OTU in 3b. A network diagram of the inferred microbial network is shown in in Figure 3c, with node sizes representing the degree of the nodes, so the larger a node, the more edges that node has with other nodes. In these plots, OTUs are grouped based on their phylum (Firmicutes, Actinobacteria, Bacteroidetes, Proteobacteria, Fusobacteria, and TM7). Looking at the adjacency matrix and network representation, we notice that Actinobacteria have few shared edges with Bacteroidetes, Proteobacteria and Fusobacteria, instead sharing the majority of their inter-phylum edges with Firmicutes, while the other phyla (Firmicutes, Bacteroidetes, Proteobacteria and Fusobacteria) show no trend in inter-phylum edges. We also notice that within the Firmicutes subnetwork, OTUs of the genus *Lactobacillus* (OTU 1 through 26 in the adjacency matrix) form their own subnetwork with very few inter-genus connections. Also, from the node degrees plot we can see that many Firmicutes have large numbers of edges. Indeed, when looking at the most connected nodes of the inferred microbial network, we found that 5 of the 6 most connected OTUs belong to the Firmicutes phylum.

Next, we show the inference on the microbe-cytokine association network. Figure 4 shows the adjacency matrix of the selected microbe-cytokine associations, with microbes colored based on phylum. We observe clear patterns of association, with both cytokines that show relationships to many OTUs, and OTUs that show relationships with several cytokines. In particular, we see that the cytokines IP-10, IL-1b, IL-17A, FGF basic, and IL-8 have the most associations with OTU abundances. When looking at which OTUs have the most associations with cytokines, we found that 6 of the 10 most connected are *Lactobacillus*. This is also seen in Figure 4, where many of the first 26 microbes (columns) have several cytokine associations. *Lactobacillus* has previously been shown to be largely influenced by cytokines (Valenti et al., 2018).

We also compare the results from SINC to those from SpiecEasi (Kurtz et al., 2015) and the **B**-constrained version of SINC, and found that the two methods that do not control for covariates shared 22 edges that were not selected by the proposed model. Two of these edges can be seen in Figure 5, which shows the network of a subset of three microbes and three cytokines estimated by SINC, as well as the network of the same subset of microbes estimated by SpiecEasi and a variant of SINC with the **B** coefficients not estimated. We hypothesize that SINC did not select the same edges as the other two methods, i.e. the edge between OTUs 14 and 19 and the edge between OTUs 19 and 20, because edges selected by methods that do not control for



a) b)



c)

Figure 3: Case study: Plot *a)*: Adjacency matrix of the microbial network inferred from the MOMS-PI data, with filled boxes representing selected edges. Plot *b)*: number of edges for each OTU, listed in the same order as in the adjacency matrix. Plot *c)*: Network diagram of the inferred microbial network, with node sizes representing the degree of the nodes.

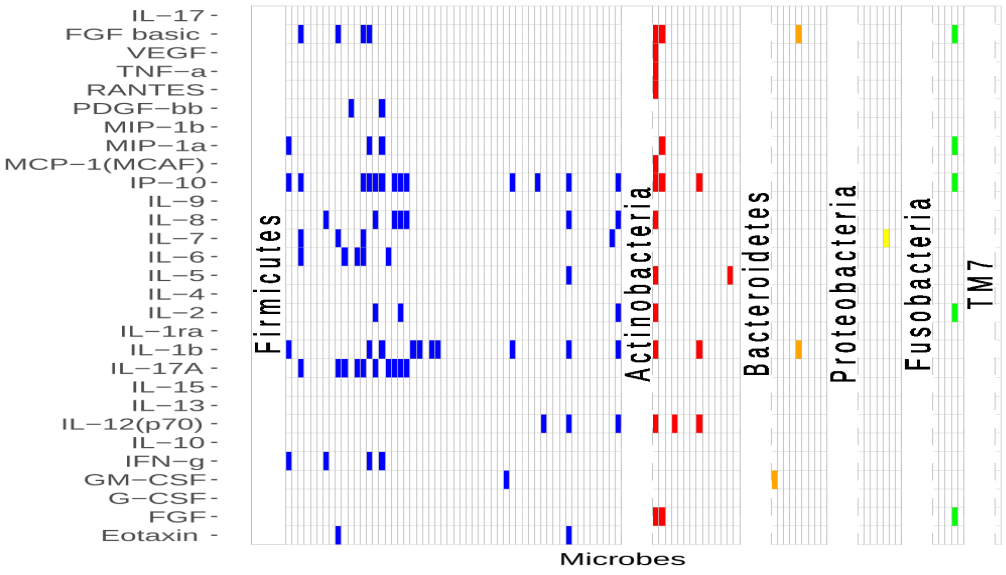


Figure 4: Case study: Adjacency matrix of the cytokine and microbial associations inferred from the MOMS-PI data. Filled boxes indicate that there is an association between a cytokine (plotted on the y axis) and an OTU (plotted on the x axis).

cytokines may incorrectly determine an edge when microbe pairs have a mutual association with a cytokine. This can be seen in Figure 5, where OTUs 14,19, and 20 all have an association with IP-10. This illustrates the ability of our model to discover covariate effects and a sparse network accounting for these effects.

6 Discussion

In this paper, we have introduced a novel Bayesian hierarchical model for count data that allows for simultaneous estimation of covariate dependence and network interactions. By accounting for covariate selection, simultaneous estimation methods are able to control for those variables, which ultimately leads to more accurate network estimation. We have considered multivariate count data, and specifically compositional data that have a fixed sum constraint, and have modeled the data using a Dirichlet-multinomial likelihood. We have accounted for covariates by modeling the log concentration parameters via a Gaussian distribution, and achieved simultaneous covariate and edge selection via spike-and-slab priors. For posterior inference, we have implemented a variational Bayes approach that includes an expectation-minimization step to enable efficient estimation. We have shown through simulations that the proposed model outperforms existing methods in its accuracy of network recovery. This is due, in part, to the flexibility of the hierarchical model, as discussed in Section 4.3, which avoids some of the over-shrinkage typical of penalized approaches, as well as the added accuracy from doing simultaneous covariate and network selection. Finally, we have applied the proposed method to data from the MOMS-PI study, to estimate the microbial interactions in the vagina, as

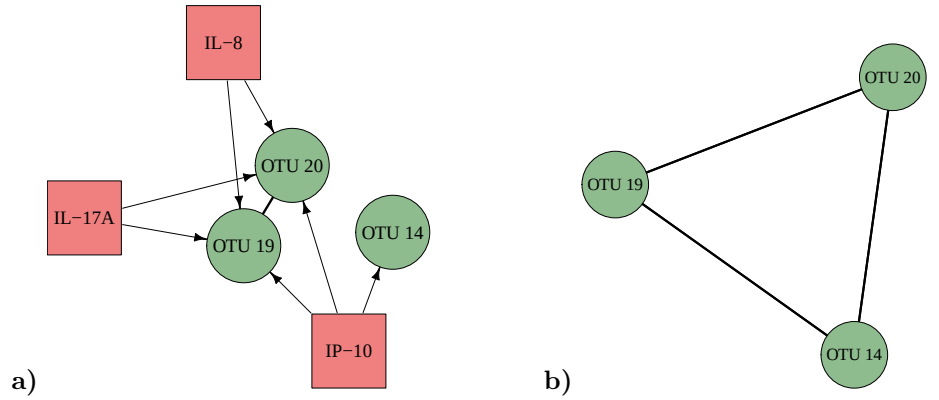


Figure 5: Case study: Plot *a)* Subnetwork of selected edges in a microbe-microbe network and selected associations between microbes and cytokines, found when using SINC. Plot *b)* Subnetwork of selected edges in a microbe-microbe network, found when using both SpeicEasi and a variation of SINC with the $\mathbf{B}$ fixed to 0. Cytokines are not included in the subnetwork of plot *b)*, as neither model accounts for cytokines.

well as the interplay between vaginal cytokines and microbial abundances, providing insight into mechanisms of host-microbial interaction during pregnancy.

Although other estimation methods, such as EM, could potentially be applied, we found that our unique hybrid algorithm offers several advantages. As we noted in Section 3, the VI and EM approaches are similar in many ways. VI, however, can enable additional insight on the uncertainty of the parameter estimates, as one can think of the EM as a special case of the VI algorithm when the variational distributions are point estimates. Although this is one potential advantage of the VI estimation of $\mathbf{B}$, we primarily prefer the VI approach for pragmatic reasons regarding performance, since we found in practice that the VI algorithm for variable selection is less sensitive to the choice of the hyperparameters (specifically, the standard deviations of the spike and slab distributions) than alternative approaches for variable selection in the EM framework. This makes the application of our approach simpler, since we can focus on tuning the parameters for the EMGS portion of the algorithm. Moreover, Ray and Szabo (2019) recently demonstrated that the VI algorithm of Carbonetto and Stephens (2012) generally outperforms the EMVS algorithm of Ročková and George (2014) across various simulation settings, suggesting we may be able to obtain more accurate estimates of $\mathbf{B}$ under this approach. Finally, VI methods can be used with discrete spike and slab priors, whereas continuous spike and slab priors are used with EM methods.

Although our VI scheme is more computationally efficient than MCMC sampling, estimating the latent variable matrix $\mathbf{Z}$ is still computationally expensive and a bottleneck to this problem. For the case study of this paper, the model ran on a cluster using 25 cores and took 16 minutes (approximately 6.2 CPU hours). Using the case study data and the default R package settings, mSSL-DPE took 49.4 minutes, and SpiecEasi took 57 seconds. Even though spiecEasi and mSSL-DPE were much faster, they resulted in less accurate predictions. Finally, mLDM was much slower and took over 120 hours per simulation. The dramatic

difference in time between mLDM and SINC is due, in part, to SINC being calibrated for parallel computing. Not only does the computational complexity of our model scale in p, but because there are $p \times n$ latent variables in $\mathbf{Z}$, the speed of our algorithm also scales with n. Avoiding estimation of these latent variables, or finding computationally more efficient estimates, would allow for further scalability of the implementation.

Python code implementing the SINC method is available at https://github.com/Nathan-Osborne/SINC/.

Acknowledgements

Research supported by NSF/DMS 1811568/1811445.

References

Barbieri, M. M. and Berger, J. O. (2004). Optimal predictive model selection. *The Annals of Statistics*, 32(3):870–897.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017). Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877.

Brown, P. J., Vannucci, M., and Fearn, T. (1998). Multivariate Bayesian variable selection and prediction. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 60(3):627–641.

Carbonetto, P. and Stephens, M. (2012). Scalable variational inference for Bayesian variable selection in regression, and its accuracy in genetic association studies. *Bayesian Analysis*, 7(1):73–108.

Chen, J. and Li, H. (2013). Variable selection for sparse Dirichlet-multinomial regression with an application to microbiome data analysis. *The annals of applied statistics*, 7(1):418–42.

David, L. A., Maurice, C. F., Carmody, R. N., Gootenberg, D. B., Button, J. E., Wolfe, B. E., Ling, A. V., Devlin, A. S., Varma, Y., Fischbach, M. A., et al. (2014). Diet rapidly and reproducibly alters the human gut microbiome. *Nature*, 505(7484):559–563.

Dempster, A. (1972). Covariance selection. *Biometrics*, 28:157–175.

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*, 0(0):1–11.

Edwards, V. L., Smith, S. B., McComb, E. J., Tamarelle, J., Ma, B., Humphrys, M. S., Gajer, P., Gwilliam, K., Schaefer, A. M., Lai, S. K., et al. (2019). The cervicovaginal microbiota-host interaction modulates *Chlamydia trachomatis* infection. *MBio*, 10(4):e01548–19.

Fan, J. and Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456):1348–1360.

Fang, H., Huang, C., Zhao, H., and Deng, M. (2017). gCoda: Conditional dependence network inference for compositional data. *Journal of Computational Biology*, 24(7):699–708.

Fettweis, J. M., Serrano, M. G., Brooks, J. P., Edwards, D. J., Girerd, P. H., Parikh, H. I., et al. (2019). The vaginal microbiome and preterm birth. *Nature Medicine*, 25(6):1012–1021.

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

George, E. I. and McCulloch, R. E. (1997). Approaches for Bayesian variable selection. *Statistica Sinica*, 7(2):339–373.

Girvan, M. and Newman, M. E. (2002). Community structure in social and biological networks. *Proceedings of the national academy of sciences*, 99(12):7821–7826.

Gloor, G. B., Macklaim, J. M., Pawlowsky-Glahn, V., and Egozcue, J. J. (2017). Microbiome datasets are compositional: And this is not optional. *Frontiers in Microbiology*, 8:22–24.

Huang, X., Wang, J., and Liang, F. (2016). A variational algorithm for Bayesian variable selection.

Jiang, H., Fei, X., Liu, H., Roeder, K., Lafferty, J., Wasserman, L., Li, X., and Zhao, T. (2019). *huge: High-Dimensional Undirected Graph Estimation*. R package version 1.3.3.

Kook, J. H., Vaughn, K. A., DeMaster, D. M., Ewing-Cobbs, L., and Vannucci, M. (2020). BVAR-Connect: A variational Bayes approach to multi-subject vector autoregressive models for inference on brain connectivity networks. *arXiv preprint arXiv:2006.04608*.

Koslovsky, M. and Vannucci, M. (2020). MicroBVS: Dirichlet-tree multinomial regression models with Bayesian variable selection – an R package. *BMC Bioinformatics*, 21:301.

Kurtz, Z. D., Müller, C. L., Miraldi, E. R., Littman, D. R., Blaser, M. J., and Bonneau, R. A. (2015). Sparse and compositionally robust inference of microbial ecological networks. *PLOS Computational Biology*, 11:1–25.

Lenkoski, A. and Dobra, A. (2011). Computational aspects related to inference in Gaussian graphical models with the G-Wishart prior. *J. Comput. Graph. Stat.*, 20(1):140–157.

Li, Z. R., McComick, T. H., and Clark, S. J. (2020). Using Bayesian latent Gaussian graphical models to infer symptom associations in verbal autopsies. *Bayesian Analysis*, to appear.

Li, Z. R. and McCormick, T. H. (2019). An expectation conditional maximization approach for Gaussian graphical models. *Journal of Computational and Graphical Statistics*, To appear.

Meinshausen, N. and Bühlmann, P. (2006). High-dimensional graphs and variable selection with the lasso. *Ann. Statist.*, 34(3):1436–1462.

Miao, Y., Kook, J. H., Lu, Y., Guindani, M., and Vannucci, M. (2020). Scalable Bayesian variable selection regression models for count data. In *Flexible Bayesian Regression Modelling*, pages 187–219. Elsevier.

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

Ntzoufras, I., Dellaportas, P., and Forster, J. J. (2003). Bayesian variable and link determination for generalised linear models. *Journal of Statistical Planning and Inference*, 111(1-2):165–180.

Raftery, A. E. (1996). Approximate Bayes factors and accounting for model uncertainty in generalised linear models. *Biometrika*, 83(2):251–266.

Ray, K. and Szabo, B. (2019). Variational bayes for high-dimensional linear regression with sparse priors. *arXiv preprint arXiv:1904.07150*.

Richardson, S., Bottolo, L., and Rosenthal (2010). Bayesian models for sparse regression analysis of high dimensional data. In *Bayesian Statistics 9*, pages 539–569.

Rothman, A. J., Levina, E., and Zhu, J. (2010). Sparse multivariate regression with covariance estimation. *Journal of Computational and Graphical Statistics*, 19(4):947–962. PMID: 24963268.

Roverato, A. (2002). Hyper inverse Wishart distribution for non-decomposable graphs and its application to Bayesian inference for Gaussian graphical models. *Scandinavian Journal of Statistics*, 29(3):391–411.

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.

Serrano, M. G., Parikh, H. I., Brooks, J. P., Edwards, D. J., Arodz, T. J., Edupuganti, L., Huang, B., Girerd, P. H., Bokhari, Y. A., Bradley, S. P., et al. (2019). Racioethnic diversity in the dynamics of the vaginal microbiome during pregnancy. *Nature Medicine*, page 1.

Sha, N., Vannucci, M., Tadesse, M. G., Brown, P. J., Dragoni, I., Davies, N., et al. (2004). Bayesian variable selection in multinomial probit models to identify molecular signatures of disease stage. *Biometrics*, 60(3):812–819.

Stingo, F., Chen, Y., Vannucci, M., Barrier, M., and Mirkes, P. (2010). A Bayesian graphical modeling approach to microRNA regulatory network inference. *Ann. Appl. Stat.*, 4(4):2024–2048.

Talhouk, A., Doucet, A., and Murphy, K. (2012). Efficient Bayesian inference for multivariate probit models with sparse inverse correlation matrices. *Journal of Computational and Graphical Statistics*, 21(3):739–757.

Tang, Y., Ma, L., and Nicolae, D. (2018). A phylogenetic scan test on Dirichlet-tree multinomial model for microbiome data. *Annals of Applied Statistics*, 12(1):1–26.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, 58(1):267–288.

Titsias, M. K. and Lázaro-Gredilla, M. (2011). Spike and slab variational inference for multi-task and multiple kernel learning. In *Advances in Neural Information Processing Systems*, pages 2339–2347.

Valenti, P., Rosa, L., Capobianco, D., Lepanto, M. S., Schiavi, E., Cutone, A., Paesano, R., and Mastromarino, P. (2018). Role of lactobacilli and lactoferrin in the mucosal cervicovaginal defense. *Frontiers in Immunology*, 9:376.

Vinci, G., Ventura, V., Smith, M., and Kass, R. (2018). Adjusted regularization in latent graphical models: Application to multiple-neuron spike count data. *Annals of Applied Statistics*, 12(2):1068–1095.

Wadsworth, W. D., Argiento, R., Guindani, M., Galloway-Pena, J., Shelburne, S. A., and Vannucci, M. (2017). An integrative Bayesian Dirichlet-multinomial regression model for the analysis of taxonomic abundances in microbiome data. *BMC Bioinformatics*, 18(1):94.

Wang, H. (2012). Bayesian graphical lasso models and efficient posterior computation. *Bayesian Analysis*, 7(2):771–790.

Wang, H. (2015). Scaling it up: Stochastic search structure learning in graphical models. *Bayesian Analysis*, 10(2):351–377.

Yang, Y., Chen, N., and Chen, T. (2017). Inference of environmental factor-microbe and microbe-microbe associations from metagenomic data using a hierarchical Bayesian statistical model. *Cell Systems*, 4(1):129 – 137.

Yuan, M. and Lin, Y. (2007). Model selection and estimation in the Gaussian graphical model. *Biometrika*, 94(1):19–35.

Zou, H. (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476):1418–1429.