Bayesian Regression with Undirected Network Predictors with an Application to Brain Connectome Data

Sharmistha Guha and Abel Rodriguez

January 24, 2018

Abstract

This article proposes a Bayesian approach to regression with a continuous scalar response and an undirected network predictor. Undirected network predictors are often expressed in terms of symmetric adjacency matrices, with rows and columns of the matrix representing the nodes, and zero entries signifying no association between two corresponding nodes. Network predictor matrices are typically vectorized prior to any analysis, thus failing to account for the important structural information in the network. This results in poor inferential and predictive performance in presence of small sample sizes. We propose a novel class of network shrinkage priors for the coefficient corresponding to the undirected network predictor. The proposed framework is devised to detect both nodes and edges in the network predictive of the response. Our framework is implemented using an efficient Markov Chain Monte Carlo algorithm. Empirical results in simulation studies illustrate strikingly superior inferential and predictive gains of the proposed framework in comparison with the ordinary high dimensional Bayesian shrinkage priors and penalized optimization schemes. We apply our method to a brain connectome dataset that contains information on brain networks along with a measure of creativity for multiple individuals. Here, interest lies in building a regression model of the creativity measure on the network predictor to identify important regions and connections in the brain strongly associated with creativity. To the best of our knowledge, our approach is the first principled Bayesian method that is able to detect scientifically interpretable regions and connections in the brain actively impacting the continuous response (creativity) in the presence of a small sample size.

Keywords: Brain connectome; Edge selection; High dimensional regression; Network predictors; Network shrinkage prior; Node selection.
1 Introduction

In recent years, network data has become ubiquitous in disciplines as diverse as neuroscience, genetics, finance and economics. Nonetheless, statistical models that involve network data are particularly challenging, not only because they require dimensionality reduction procedures to effectively deal with the large number of pairwise relationships, but also because flexible formulations are needed to account for the topological structure of the network.

The literature has paid heavy attention to models that aim to understand the relationship between node-level covariates and the structure of the network. A number of classic models treat the dyadic observations as the response variable, examples include random graph models (Erdos and Rényi, 1960), exponential random graph models (Frank and Strauss, 1986), social space models Hoff et al. (2002); Hoff (2005, 2009) and stochastic block models (Nowicki and Snijders, 2001). The goal of these models is either link prediction or to investigate homophily, i.e., the process of formation of social ties due to matching individual traits. Alternatively, models that investigate influence or contagion attempt to explain the node-specific covariates as a function of the network structure (e.g., see Christakis and Fowler, 2007; Fowler and Christakis, 2008; Shoham et al., 2015 and references therein). Common methodological approaches in this context include simultaneous autoregressive (SAR) models (e.g., see Lin, 2010) and threshold models (e.g., see Watts and Dodds, 2009). However, ascertaining the direction of a causal relationship between network structure and link or nodal attributes, i.e. whether it pertains to homophily or contagion, is difficult (e.g., see Doreian, 2001 and Shalizi and Thomas, 2011 and references therein). Hence there has been a growing interest in joint models for the coevolution of the network structure and nodal attributes (e.g., see Fosdick and Hoff, 2015; Durante et al., 2017; De la Haye et al., 2010; Niezink and Snijders, 2016; Guhaniyogi and Rodriguez, 2017).

In this paper we investigate Bayesian models for network regression. Unlike the problems discussed above, in network regression we are interested in the relationship between the structure of the network and one or more global attributes of the experimental unit on which the network data is collected. As a motivating example, we consider the problem of predicting the composite creativity index of individuals on the basis of neuroimaging data reassuring the connectivity of different brain regions. The goal of these studies is twofold. First neuroscientists are interested in identifying regions of the brain that are involved in creative thinking. Secondly, it is important to determine which how the strength of connection among these influential regions affects the level of creativity of the individual. More specifically, we construct a novel Bayesian network shrinkage prior that combines ideas from spectral decomposition methods and spike-and-slab priors to generate a model that respects the structure of the predictors. The model produces accurate predictions, allows us to identify both modes and links that have influence on the response, and yield
well-calibrated interval estimates for the model parameters.

A common approach to network regression is to use a few summary measures from the network in the context of a flexible regression or classification approach (see, for example, Bullmore and Sporns, 2009 and references therein). Clearly, the success of this approach is highly dependent on selecting the right summaries to include. Furthermore, it cannot identify the impact of specific nodes on the response, which is of clear interest in our setting. Alternatively, a number of authors have proceeded to vectorize the network predictor (originally obtained in the form of a symmetric matrix). Subsequently, the continuous response would be regressed on the high dimensional collection of edge weights (e.g., see Richiardi et al., 2011 and Craddock et al., 2009). This approach can take advantage of the recent developments in high dimensional regression, consisting of both penalized optimization (Tibshirani, 1996), and Bayesian shrinkage (Park and Casella, 2008; Carvalho et al., 2010; Armagan et al., 2013) perspectives. However, this treat the links of the network as if they were exchangeable, ignoring the fact that coefficients that involving common nodes can be expect to be correlated a priori. Ignoring this correlation often leads to poor predictive performance and can potentially impact model selection.

Recently, Relión et al. (2017) proposed a penalized optimization scheme that not only enables classification of networks, but also identifies important nodes and edges. Although this model seems to perform well for prediction problem, uncertainty quantification is difficult because standard bootstrap methods are not consistent for Lasso-type methods (e.g., see Kyung et al., 2010 and Chatterjee and Lahiri, 2010). Modifications of the bootstrap that produce well-calibrated confidence intervals in the context of standard Lasso regression have been proposed (e.g., see Chatterjee and Lahiri, 2011), but it is not clear whether they extend to the more complex penalties introduced in Relión et al. (2017). Recent developments on tensor regression (e.g., see Zhou et al., 2013; Guhaniyogi et al., 2017) are also relevant to our work. However, these approaches tend to focus mainly on prediction and are not designed to detect important nodes impacting the response.

The rest of the article evolves as follows. Section 2 proposes the novel network shrinkage prior and discusses posterior computation for the proposed model. Empirical investigations with various simulation studies are presented in Section 3 while Section 4 analyzes the brain connectome dataset. We provide results on region of interest (ROI) and edge selection and find them to be scientifically consistent with previous studies. Finally, Section 5 concludes the article with an eye towards future work.
2 Model Formulation

2.1 Notations

Let \( y_i \in \mathbb{R} \) and \( A_i \) represent the observed scalar response and the corresponding weighted undirected network for the \( i \)th sample, \( i = 1, \ldots, n \) respectively. The weights corresponding to the edges belong to \( \mathbb{R} \) and all graphs share the same labels on their nodes. For example, in our brain connectome application discussed subsequently, \( y_i \) corresponds to a phenotype, while \( A_i \) encodes the network connections between different regions of the brain for the \( i \)th individual. Let the network corresponding to any individual consist of \( V \) nodes. Mathematically, it amounts to \( A_i \) being a \( V \times V \) matrix, with the \((k, l)\)th entry of \( A_i \) denoted by \( a_{i,k,l} \in \mathbb{R} \). We focus on networks that contain no self relationship, i.e. \( a_{i,k,k} \equiv 0 \), and are undirected \((a_{i,k,l} = a_{i,l,k})\). The brain connectome application considered here naturally justifies these assumptions. Although we present our model specific to these settings, it will be evident that the proposed model can be easily extended to directed networks with self-relations. Throughout this article, we denote the Frobenius inner product between two \( V \times V \) matrices \( C \) and \( D \) by \( \langle C, D \rangle_F = \text{Trace}(D'C) \). Frobenius inner product is the natural inner product on the space of matrices and is a generalization of the dot product from vector to matrix spaces. The square root of the Frobenius inner product of a matrix with itself yields the Frobenius norm. Mathematically, it is denoted by \( ||C||_F = \sqrt{\langle C, C \rangle_F} \). Additionally, we define \( ||B||_1 \) as the sum of the absolute values of all the elements of matrix \( B \). Also, define \( ||\cdot||_2 \) as the \( l_2 \) norm of a vector and \( B_{(k)} \) as the \( k \)th row of \( B \).

2.2 Bayesian Network Regression Model

We propose the high dimensional regression model of the response \( y_i \) for the \( i \)-th individual on the undirected network predictor \( A_i \) as

\[
y_i = \mu + \langle A_i, B \rangle_F + \epsilon_i, \quad \epsilon_i \sim N(0, \tau^2),
\]

where \( B \) is the network coefficient matrix of dimension \( V \times V \) whose \((k, l)\)th element is given by \( \beta_{k,l} \). Similar to the network predictor, the network coefficient matrix \( B \) is also assumed to be symmetric with zero diagonal entries. \( \tau^2 \) is the variance of the idiosyncratic error.

Since self relationship is absent and \( A_i \) is symmetric, \( \langle A_i, B \rangle_F = 2 \sum_{1 \leq k < l \leq V} a_{i,k,l} \beta_{k,l} \). Then, denoting \( \gamma_{k,l} = 2\beta_{k,l} \), (1) can be rewritten as

\[
y_i = \mu + \sum_{1 \leq k < l \leq V} a_{i,k,l} \gamma_{k,l} + \epsilon_i, \quad \epsilon_i \sim N(0, \tau^2),
\]
Equation (2) connects the network regression model with the linear regression framework with $a_{i,k,l}$’s as predictors and $\gamma_{k,l}$’s as the corresponding coefficients.

While in ordinary linear regression, the predictor coefficients are indexed by the natural numbers $\mathcal{N}$, Model (2) indexes the predictor coefficients by their positions in the matrix $B$. This is done in order to keep a tab not only on the edge itself but also on the nodes connecting the edges. As mentioned earlier, we are interested in identifying nodes and edges which contribute to the regression. Additionally, we would also like to accurately estimate the coefficients $\gamma_{k,l}$ and subsequently make accurate predictions. The next section describes a novel network shrinkage prior to achieve these goals.

2.3 Developing the Network Shrinkage Prior

2.3.1 Vector Shrinkage Prior

High dimensional regression with vector predictors has recently been of interest in Bayesian statistics. An overwhelming literature in Bayesian statistics in the last decade has focused on shrinkage priors which shrink coefficients corresponding to unimportant variables to zero while minimizing the shrinkage of coefficients corresponding to influential variables. Many of these shrinkage prior distributions can be expressed as a scale mixture of normal distributions, commonly referred to as global-local (GL) scale mixtures (Polson and Scott (2010)), that enable fast computation employing simple conjugate Gibbs sampling. More precisely, in the context of model (2), a global-local scale mixture prior would take the form

$$
\gamma_{k,l} \sim N(0, s_{k,l} \tau^2), \quad s_{k,l} \sim g_1, \quad \tau^2 \sim g_2, \quad 1 \leq k < l \leq V,
$$

where $(\gamma_{1,2}, ..., \gamma_{V-1,V})$ are the regression coefficients in (2) with $q = V(V-1)/2$ predictors. Note that $s_{1,2}, ..., s_{V-1,V}$ are local scale parameters controlling the shrinkage of the coefficients, while $\tau^2$ is the global scale parameter. Different choices of $g_1$ and $g_2$ lead to different classes of Bayesian shrinkage priors which have appeared in the literature. For example, the Bayesian Lasso (Park and Casella (2008)) prior takes $g_1$ as exponential and $g_2$ as the Jeffreys prior, the Horseshoe prior (Carvalho et al. (2010)) takes both $g_1$ and $g_2$ as half-Cauchy distributions and the Generalized Double Pareto Shrinkage prior (Armagan et al. (2013)) takes $g_1$ as exponential and $g_2$ as the Gamma distribution.

The direct application of this global-local prior in the context of (2) is unappealing. To elucidate further, note that if node $k$ contributes minimally to the response, one would expect to have smaller estimates for all coefficients $\gamma_{k,l}, l > k$ and $\gamma_{l',k}, l' < k$ corresponding to edges connected to node $k$. The ordinary GL shrinkage prior distribution given as above does not necessarily conform to such an important restriction. In what follows, we build a network
shrinkage prior upon the existing literature that respects this constraint as elucidated in the next section.

2.3.2 Network Shrinkage Prior

We propose a shrinkage prior on the coefficients $\gamma_{k,l}$ and refer to it as the Bayesian Network Shrinkage prior. The prior borrows ideas from low-order spectral representations of matrices. Let $\mathbf{u}_1, ..., \mathbf{u}_V \in \mathbb{R}^R$ be a collection of $R$-dimensional latent variables, one for each node, such that $\mathbf{u}_k$ corresponds to node $k$. We draw each $\gamma_{k,l}$ conditionally independent from a density that can be represented as a location and scale mixture of normals. More precisely,

$$
\gamma_{k,l} | s_{k,l}, \mathbf{u}_k, \mathbf{u}_l, \tau^2 \sim N(u_k' \Lambda u_l, \tau^2 s_{k,l}),
$$

where $s_{k,l}$ is the scale parameter corresponding to each $\gamma_{k,l}$ and $\Lambda = \text{diag}(\lambda_1, ..., \lambda_R)$ is an $R \times R$ diagonal matrix. $\lambda_r \in \{0, 1\}$’s are introduced to assess the effect of the $r$th dimension of $\mathbf{u}_k$ on the mean of $\gamma_{k,l}$. In particular, $\lambda_r = 0$ implies that the $r$th dimension of the latent variable $\mathbf{u}_k$ is not informative for any $k$. Note that if $s_{k,l} = 0$, this prior will imply that $\Gamma = 2B = U' \Lambda U$, where $U$ is an $R \times V$ matrix whose $k$th column corresponds to $\mathbf{u}_k$ and $\Gamma = ((\gamma_{k,l}))_{k,l=1}^V$. Since $R << V$, the mean structure of $\Gamma$ assumes a low-rank matrix decomposition.

In order to learn how many components of $\mathbf{u}_k$ are informative for (3), we assign a hierarchical prior

$$
\lambda_r \sim \text{Ber}(\pi_r), \; \pi_r \sim \text{Beta}(1, r\eta), \; \eta > 1.
$$

The choice of hyper-parameters of the beta distribution is crucial in order to impart increasing shrinkage on $\lambda_r$ as $r$ grows. In particular, $E[\lambda_r] = 1/(1 + r\eta) \to 0$, as $r \to \infty$, so that the prior favors choice of smaller number of active components in $\mathbf{u}_k$’s impacting the response. Additionally, the hyper-parameter of the distribution of $\lambda_r$ safeguards the prior on $\lambda_r$ from flattening out even at large $r$. In particular, $\sum_{r=1}^R E[\lambda_r] = \sum_{r=1}^R 1/(1 + r\eta)$, so that $\sum_{r=1}^R E[\lambda_r]$ converges as $R \to \infty$. Note that $\sum_{r=1}^R \lambda_r$ is the number of dimensions of $\mathbf{u}_k$ contributing to predict the response. We refer to $\sum_{r=1}^R \lambda_r$ as $R_{eff}$, the effective dimensionality of the latent variables.

The mean structure of $\gamma_{k,l}$ is constructed to take into account the interaction between the $k$th and the $l$th nodes. In particular, the node specific latent variables $\mathbf{u}_1, ..., \mathbf{u}_V$ can be thought of as points in an $R$-dimensional sphere. Drawing intuition from Hoff (2005), we might imagine that the interaction between the $k$th and $l$th nodes has a positive, negative or neutral impact on the response depending on whether $\mathbf{u}_k$ and $\mathbf{u}_l$ are in the same direction, opposite direction or orthogonal to each other respectively. In other words, whether the
angle between $\mathbf{u}_k$ and $\mathbf{u}_l$ is acute, obtuse or right, i.e. $\mathbf{u}_k' \Lambda \mathbf{u}_l > 0$, $\mathbf{u}_k' \Lambda \mathbf{u}_l < 0$ or $\mathbf{u}_k' \Lambda \mathbf{u}_l = 0$ respectively. The conditional mean of $\gamma_{k,l}$ in (3) is constructed to capture such latent network information in the predictor.

In order to directly infer on node selection (i.e. to determine if a node is inactive in explaining the response), we assign the spike-and-slab (Ishwaran and Rao (2005)) mixture distribution prior on the latent factor $\mathbf{u}_k$ as below

$$
\mathbf{u}_k \sim \begin{cases} 
N(0, M), & \text{if } \xi_k = 1 \\
\delta_0, & \text{if } \xi_k = 0
\end{cases}, \quad \xi_k \sim \text{Ber}(\Delta), 
$$

(4)

where $\delta_0$ is the Dirac-delta function at 0 and $M$ is a covariance matrix of order $R \times R$. The parameter $\Delta$ corresponds to the probability of the nonzero mixture component. Note that if the $k$th node of the network predictor is inactive in predicting the response, then a-posteriori $\xi_k$ should provide high probability to 0. Thus, based on the posterior probability of $\xi_k$, it will be possible to identify unimportant nodes in the network regression. The rest of the hierarchy is accomplished by assigning prior distributions on the $s_{k,l}$’s, $\tau^2$ and $M$ as follows:

$$
s_{k,l} \sim \text{Exp}(\lambda^2/2), \quad \tau^2 \sim \pi(\tau^2) \propto \frac{1}{\tau^2}, \quad M \sim \text{IW}(S, \nu), 
$$

$$
\lambda^2 \sim \text{Gamma}(r, \delta), \quad \Delta \sim \text{Beta}(a, b),
$$

where $S$ is an $R \times R$ positive definite scale matrix. $\text{IW}(S, \nu)$ denotes an Inverse-Wishart distribution with scale matrix $S$ and degrees of freedom $\nu$. Finally, we choose a non-informative flat prior on $\mu$. Appendix B shows the propriety of the marginal posterior distributions for all parameters.

Note that, if $\mathbf{u}_k' \Lambda \mathbf{u}_l = 0$, the marginal prior distribution of $\gamma_{k,l}$ integrating out all the latent variables turns out to be the double exponential distribution which is connected to the Bayesian Lasso prior. When $\mathbf{u}_k' \Lambda \mathbf{u}_l = 0$, the marginal prior distribution of $\gamma_{k,l}$ appears to be a location mixture of double exponential prior distributions, the mixing distribution being a function of the network. Owing to this fact, we coin the proposed prior distribution as the Bayesian Network Lasso prior.

### 2.4 Posterior Computation

Although summaries of the posterior distribution cannot be computed in closed form, full conditional distributions for all parameters are available and correspond to standard families. Thus posterior computation of parameters proceed through Gibbs sampling. Details of all the full conditionals are presented in Appendix A.

In order to identify whether the $k$th node is important in terms of predicting the response,
we rely on the post burn-in $L$ samples $\xi_k^{(1)}, \ldots, \xi_k^{(L)}$ of $\xi_k$. Node $k$ is recognized to be influential if $\frac{1}{L} \sum_{l=1}^{L} \xi_k^{(l)} > 0.5$. Next, we turn our attention to inferring on the dimension of the latent variable $u_k$. To this end, we empirically estimate the full posterior distribution of $R_{eff}$. In particular, an estimate of $P(R_{eff} = r | Data)$ is given by $\frac{1}{L} \sum_{l=1}^{L} I(\sum_{m=1}^{R} \lambda_m^{(l)} = r)$, where $I(A)$ for an event $A$ is 1 if the event $A$ happens and 0 otherwise and $\lambda_m^{(1)}, \ldots, \lambda_m^{(L)}$ are the $L$ post burn-in MCMC samples of $\lambda_m$. The estimated posterior distribution of $R_{eff}$ is presented in the form of a dot chart.

### 3 Simulation Studies

This section comprehensively contrasts both the inferential and the predictive performances of our proposed approach with a number of competitors in various simulation settings. We refer to our proposed approach as the Bayesian Network Regression (BNR). As competitors, we consider both frequentist penalized optimization methods as well as Bayesian shrinkage priors for high-dimensional regression.

First we use the generic variable selection and shrinkage methods that ignore the relational nature of the predictors. More specifically, we use Lasso (Tibshirani (1996)) as a popular penalized optimization scheme, and the Bayesian Lasso (Park and Casella (2008)) and Horseshoe priors (Carvalho et al. (2010)) as Bayesian shrinkage regression methods. The Horseshoe is considered to be the state-of-the-art Bayesian shrinkage prior and is known to perform well, both in sparse and not-so-sparse regression settings. In order to implement Lasso, Bayesian Lasso (henceforth interchangeably referred to as BLasso) and the Horseshoe (henceforth interchangeably referred to as BHS), we employ the glmnet (Friedman et al. (2010)) and monomvn (Gramacy and Gramacy (2013)) packages in R, respectively. All these methods treat edges between nodes as “bags of predictors” to run high dimensional regression, thereby ignoring the structure of the network predictor. A thorough comparison with these methods will indicate the relative advantage of exploiting the structure of the network predictor.

Additionally, we compare our method to a frequentist approach that develops network regression in the presence of a network predictor and scalar response (Relión et al. (2017)). To be precise, we adapt Relión et al. (2017) to a continuous response context and propose to estimate the network regression coefficient matrix $B$ from

$$
\hat{B} = \arg \min_{B \in \mathbb{R}, B = B', \text{diag}(B) = 0} \left\{ \frac{1}{n} \sum_{i=1}^{n} (y_i - \mu - \langle A_i, B \rangle_F)^2 + \frac{\phi}{2} \|B\|_F^2 + \zeta \left( \sum_{k=1}^{V} \|B_{(k)}\|_2 + \rho \|B\|_1 \right) \right\},
$$

(5)
where $\varphi, \rho, \varsigma$ are tuning parameters. The best possible choice of the tuning parameter triplet $(\varphi, \rho, \varsigma)$ is made using cross validation over a grid of possible values. Relión et al. (2017) argue that the penalty in (5) incorporates the network information of the predictor, thereby yielding superior inference to any ordinary penalized optimization scheme. Hence comparison with (5) will potentially highlight the advantages of a carefully structured Bayesian network shrinkage prior over the penalized optimization scheme incorporating network information. It is also worth mentioning that in the absence of any open source code, we implement the algorithm in Relión et al. (2017). All Bayesian competitors are allowed to draw 50,000 MCMC samples out of which the first 30,000 are discarded as burn-ins. All posterior inference is carried out based on the rest 20,000 MCMC samples after suitably thinning the post burn-in chain.

### 3.1 Predictor and Response Data Generation

In all simulation studies, the undirected symmetric network predictor $A_i$ for the $i$th sample is simulated by drawing $a_{i,k,l} \sim \text{iid } N(0,1)$ for $k < l$ and setting $a_{i,k,l} = a_{i,l,k}, a_{i,k,k} = 0$ for all $k, l \in \{1, ..., V\}$. The response $y_i$ is generated according to the network regression model

$$y_i = \mu_0 + \langle A_i, B_0 \rangle F + \epsilon_i; \epsilon_i \sim N(0, \tau_0^2), \quad (6)$$

with $\tau_0^2$ as the true noise variance. In a similar vein as section 2.2, define $\gamma_{k,l,0} = 2\beta_{k,l,0}$, which will later be used to define the mean squared error. In the interest of better presentation, all of our simulations use $V = 20$ nodes and $n = 70$ samples.

To study all competitors under various data generation schemes, we simulate the true network predictor coefficient $B_0$ under the three simulation scenarios, referred to as *Simulation 1*, *Simulation 2* and *Simulation 3*.

**Simulation 1**

In *Simulation 1*, we draw $V$ latent variables $w_k$, each of dimension $R_{\text{gen}}$, from a mixture distribution given by

$$w_k \sim \pi_w N_{R_{\text{gen}}}(w_{\text{mean}}, w_{\text{sd}}^2) + (1 - \pi_w) \delta_0; \ k \in \{1, ..., V\}, \quad (7)$$

where $\delta_0$ is the Dirac-delta function and $\pi_w$ is the probability of any $w_k$ being nonzero. The $(k,l)$th element (and the $(l,k)$th element) of the network predictor coefficient $B_0$, denoted by $\beta_{k,l,0}$ (and $\beta_{l,k,0}$), is given by $\beta_{k,l,0} = \beta_{l,k,0} = \frac{w_k w_l}{2}$ (Hoff (2005)). Note that if $w_k$ is zero, any edge connecting the $k$th node has no contribution to the regression mean function in (6), i.e. the $k$th node becomes inactive in predicting the response. According to (7), $(1-\pi_w)$ is the probability of a node being inactive. Hereafter, $(1-\pi_w)$ is referred to as the sparsity parameter in the context of the data generation mechanism under *Simulation 1*. $R_{\text{gen}}$ is the
true dimension of $w_k$.

**Simulation 2**

To simulate $B_0$ under Simulation 2, we draw $V$ latent variables $w_k$, each of dimension $R_{gen}$, from a mixture distribution given by

$$w_k \sim \pi w^* N_{R_{gen}}(w_{\text{mean}}, w_{sd}^2) + (1 - \pi w^*)\delta_0; \ k \in \{1, \ldots, V\}. \quad (8)$$

If $w_k$ is simulated as $0$, we set $\beta_{k,l,0} = \beta_{l,k,0} = 0$ for any $l$. Thus if $w_k = 0$, the $k$th node is set to be inactive in predicting the response according to (6). If $w_k$ and $w_l$ are both nonzero, the edge coefficient $\beta_{k,l,0}$ connecting the $k$th and the $l$th nodes ($k < l$) is simulated from $N(0.8, 1)$. Respecting the symmetry condition, we set $\beta_{l,k,0} = \beta_{k,l,0}$. The edge between any two nodes $k$ and $l$ contributes in predicting the response if either $w_k$ or $w_l$ is nonzero. Hence in the context of Simulation 2, $1 - \pi w^*$ is referred to as the sparsity parameter.

**Simulation 3**

To simulate $B_0$ under Simulation 3, we draw $V$ latent variables $w_k$, each of dimension $R_{gen}$, from a mixture distribution given by

$$w_k \sim \pi w^{**} N_{R_{gen}}(w_{\text{mean}}, w_{sd}^2) + (1 - \pi w^{**})\delta_0; \ k \in \{1, \ldots, V\}. \quad (9)$$

If $w_k$ is simulated as $0$, we set $\beta_{k,l,0} = \beta_{l,k,0} = 0$ for any $l$, i.e. the $k$th node is set to be inactive for the purpose of regression. The edge coefficient $\beta_{k,l,0}$ connecting any two nodes $k$ and $l$ with $w_k$ and $w_l$ both nonzero ($k < l$), is simulated from a mixture distribution given by

$$\beta_{k,l,0} \sim \pi w^{***} N_{R_{gen}}(0.8, 1) + (1 - \pi w^{***})\delta_0; \ k, l \in \{1, \ldots, V\}. \quad (10)$$

Again, we set $\beta_{k,l,0} = \beta_{l,k,0}$ respecting the symmetry condition. Contrary to Simulation 2, Simulation 3 allows the possibility of an edge between the $k$th and the $l$th nodes having no impact on the response even when both $w_k$ and $w_l$ are nonzero. In the context of Simulation 3, $(1 - \pi w^{**})$ and $(1 - \pi w^{***})$ are referred to as the node sparsity and edge sparsity parameters.

It is worth mentioning that in Simulation 1, the simulated coefficient $\beta_{k,l,0}$ corresponding to the $(k, l)$th edge $\alpha_{k,l}$ represents a bi-linear interaction between the latent variables corresponding to the $k$th and the $l$th nodes. Thus Simulation 1 generates $B_0$ respecting the network structure in $A_i$ and is the most interesting simulation scenario to investigate.

For a comprehensive picture for Simulation 1, we consider 9 different cases as summarized in Table 1. In each of these cases, the network predictor and the response are generated by changing the sparsity $\pi w$ and the true dimension $R_{gen}$ of the latent variables $w_k$’s. The table also presents the maximum dimension $R$ of the latent variables $u_k$ for the fitted network re-
gression model (2). Note that various cases also allow model mis-specification with unequal choices of $R$ and $R_{gen}$.

Both Simulation 2 and Simulation 3 partially respect the network structure while considering no impact on the regression function in (6) from any edge that is connected to any inactive node. However, both in simulations 2 and 3, the coefficient corresponding to the $(k,l)$th edge $a_{i,k,l}$ is drawn from a normal distribution with no interaction specific to nodes $k$ and $l$. Thus, the data generation schemes for simulations 2 and 3 will presumably offer no favor in terms of predictive ability or estimation of edge specific parameters to the network regression model over the ordinary high dimensional regression models that treat edges as bags of predictors and run a penalized optimization or shrinkage method. We present two cases for both simulations 2 and 3, recorded in Table 2. In all three simulation cases, apart from investigating the model as a tool for inference and prediction, it is of interest to observe the posterior distributions of $\lambda_r$’s to judge if the model effectively learns the dimensionality $R_{gen}$ of the latent variables $w_k$. Noise variance $\tau_0^2$ is fixed at 1 for all scenarios. $w_{mean}$ and $w_{sd}$ are set as $0.8 \times 1_{R_{gen}}$ and $I_{R_{gen} \times R_{gen}}$ for all simulations.

<table>
<thead>
<tr>
<th>Cases</th>
<th>$R_{gen}$</th>
<th>$R$</th>
<th>Sparsity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case - 1</td>
<td>2</td>
<td>2</td>
<td>0.5</td>
</tr>
<tr>
<td>Case - 2</td>
<td>2</td>
<td>3</td>
<td>0.6</td>
</tr>
<tr>
<td>Case - 3</td>
<td>2</td>
<td>5</td>
<td>0.3</td>
</tr>
<tr>
<td>Case - 4</td>
<td>2</td>
<td>5</td>
<td>0.4</td>
</tr>
<tr>
<td>Case - 5</td>
<td>3</td>
<td>5</td>
<td>0.5</td>
</tr>
<tr>
<td>Case - 6</td>
<td>4</td>
<td>5</td>
<td>0.4</td>
</tr>
<tr>
<td>Case - 7</td>
<td>2</td>
<td>5</td>
<td>0.5</td>
</tr>
<tr>
<td>Case - 8</td>
<td>2</td>
<td>4</td>
<td>0.7</td>
</tr>
<tr>
<td>Case - 9</td>
<td>3</td>
<td>5</td>
<td>0.7</td>
</tr>
</tbody>
</table>

Table 1: Table presents different cases under Simulation 1. The true dimension $R_{gen}$ is the dimension of vector object $w_k$ using which data has been generated. The maximum dimension $R$ is the dimension of vector object $u_k$ using which the model has been fit. Sparsity refers to the fraction of generated $w_k = 0$, i.e. $(1 - \pi_w)$.

### 3.2 Results

In all simulation results shown in this section, the BNR model is fitted with the choices of the hyper-parameters given by $S = I$, $\nu = 10$, $a_\Delta = 1$, $b_\Delta = 1$, $r_\lambda = 1$ and $\delta = 1$. Our extensive simulation studies reveal that both inference and prediction are robust with various choices of the hyper-parameters.

**Node Selection**

Figure 1 shows a matrix whose rows correspond to different cases in Simulation 1 and
Table 2: Table presents different cases for Simulations 2 and 3. The true dimension $R_{gen}$ is the dimension of vector object $w_k$ using which data has been generated. The maximum dimension $R$ is the dimension of vector object $u_k$ using which the model has been fit. While Simulation 2 only has a sparsity parameter $\pi_w$, Simulation 3 has a node sparsity and an edge sparsity parameter.

<table>
<thead>
<tr>
<th>Cases</th>
<th>$R_{gen}$</th>
<th>$R$</th>
<th>Sparsity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case - 1</td>
<td>3</td>
<td>5</td>
<td>0.7</td>
</tr>
<tr>
<td>Case - 2</td>
<td>3</td>
<td>5</td>
<td>0.2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cases</th>
<th>$R_{gen}$</th>
<th>$R$</th>
<th>Node Sparsity</th>
<th>Edge Sparsity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case - 1</td>
<td>3</td>
<td>5</td>
<td>0.7</td>
<td>0.5</td>
</tr>
<tr>
<td>Case - 2</td>
<td>3</td>
<td>5</td>
<td>0.2</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Simulation columns correspond to the nodes of the network. The dark and clear cells correspond to the truly active and inactive nodes respectively. The posterior probability of the $k$th node being detected as active, i.e. $P(\xi_k = 1 \mid Data)$ has been overlaid for all $k \in \{1,...,20\}$ in all 9 simulation cases. The plot suggests overwhelmingly accurate detection of nodes influencing the response. We provide similar plots on node detection in Figure 2 for various cases in Simulation 2 and Simulation 3. Both figures show overwhelming detection of active nodes both in simulations 2 and 3 with a very few false positives. We emphasize the fact that BNR is designed to detect important nodes while BLasso, HS or Lasso (or any other ordinary high dimensional regression technique) do not allow node selection in the present context. Relión et al. (2017) performs sub-optimally in terms of node selection and is not shown here.

**Estimating the network coefficient**

Another important aspect of the parametric inference lies in the performance in terms of estimating the network predictor coefficient. Table 3, 4, 5 present the MSE of all the competitors in simulations 1, 2 and 3 respectively. Given that both the fitted network regression coefficient $B$ and the true coefficient $B_0$ are symmetric, the MSE for any competitor is calculated in each dataset as $\frac{1}{q} \sum_{k<l}(\hat{\gamma}_{k,l} - \gamma_{k,l,0})^2$, where $\hat{\gamma}_{k,l}$ is the point estimate of $\gamma_{k,l} = 2\beta_{k,l}$ and $\gamma_{k,l,0}$ is the true value of the coefficient. For Bayesian models (such as the proposed model), $\hat{\gamma}_{k,l}$ is taken to be the posterior mean of $\gamma_{k,l}$.

From Table 3, it is evident that the proposed Bayesian network regression (BNR) significantly outperforms all its Bayesian and frequentist competitors in Simulation 1 when the sparsity parameter is low to moderate (cases 1-7). While BNR is expected to perform much better than BLasso, Horseshoe and Lasso due to incorporation of network information, it is important to note that the carefully chosen local-global shrinkage with a well formulated hierarchical mean structure seems to possess more detection and estimation power than Relión et al. (2017). When the sparsity parameter is high in Simulation 1 (cases 8-9), our simulation scheme sets an overwhelming proportion of $\beta_{k,l,0}$’s as zero. As a result, only a small subnetwork of $A_i$ affects the response $y_i$, and hence BNR’s performance becomes closely
<table>
<thead>
<tr>
<th>Simulation Cases</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>0.105</td>
<td>1</td>
<td>0</td>
<td>0.002</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0.032</td>
<td>0.004</td>
<td>1</td>
<td>1</td>
<td>0.001</td>
<td>1</td>
<td>0.003</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>0.014</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.006</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>0.022</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.001</td>
<td>1</td>
<td>1</td>
<td>0.002</td>
</tr>
<tr>
<td>5</td>
<td>0.066</td>
<td>0.038</td>
<td>1</td>
<td>0.006</td>
<td>1</td>
<td>1</td>
<td>0.067</td>
<td>1</td>
<td>0.002</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.020</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.001</td>
<td>0.994</td>
</tr>
<tr>
<td>7</td>
<td>0.005</td>
<td>0.116</td>
<td>0.006</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.004</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>0.006</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.007</td>
<td>1</td>
<td>0.002</td>
<td>0.009</td>
<td>0.002</td>
</tr>
<tr>
<td>9</td>
<td>0.008</td>
<td>0.042</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.008</td>
<td>0.009</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>0.024</td>
<td>1</td>
<td>1</td>
<td>0.007</td>
<td>1</td>
<td>0.003</td>
<td>1</td>
<td>0.002</td>
</tr>
<tr>
<td>11</td>
<td>1</td>
<td>0.024</td>
<td>1</td>
<td>1</td>
<td>0.006</td>
<td>1</td>
<td>1</td>
<td>0.001</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>1</td>
<td>0.026</td>
<td>1</td>
<td>1</td>
<td>0.075</td>
<td>1</td>
<td>0.002</td>
<td>0.012</td>
<td>1</td>
</tr>
<tr>
<td>13</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.005</td>
<td>1</td>
<td>0.001</td>
<td>1</td>
<td>1</td>
<td>0.002</td>
</tr>
<tr>
<td>14</td>
<td>0.013</td>
<td>1</td>
<td>1</td>
<td>0.059</td>
<td>0</td>
<td>1</td>
<td>0.014</td>
<td>0.001</td>
<td>0.001</td>
</tr>
<tr>
<td>15</td>
<td>0.123</td>
<td>0.028</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0.006</td>
<td>0.004</td>
<td>0.004</td>
<td>0.023</td>
</tr>
<tr>
<td>16</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0.011</td>
<td>1</td>
<td>1</td>
<td>0.053</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>17</td>
<td>0.004</td>
<td>0.037</td>
<td>1</td>
<td>0.029</td>
<td>1</td>
<td>1</td>
<td>0.003</td>
<td>0.025</td>
<td>0.029</td>
</tr>
<tr>
<td>18</td>
<td>0.009</td>
<td>1</td>
<td>0.009</td>
<td>1</td>
<td>0.004</td>
<td>1</td>
<td>0.008</td>
<td>0.008</td>
<td>0.006</td>
</tr>
<tr>
<td>19</td>
<td>0.006</td>
<td>0.043</td>
<td>1</td>
<td>1</td>
<td>0.001</td>
<td>1</td>
<td>0.006</td>
<td>0.001</td>
<td>1</td>
</tr>
<tr>
<td>20</td>
<td>0.009</td>
<td>0.034</td>
<td>1</td>
<td>0.009</td>
<td>0.003</td>
<td>1</td>
<td>0.006</td>
<td>0.002</td>
<td>0.003</td>
</tr>
</tbody>
</table>

Figure 1: True activity status of a network node (clear background denotes *inactive* and dark background denotes *active*). Note that there are 20 rows (corresponding to 20 nodes) and 9 columns (corresponding to 9 different cases. The model-detected posterior probability of being active has been super-imposed onto the corresponding node.

comparable to Horseshoe and BLasso in terms of estimating $B_0$ from (6). As argued earlier, generation of $\beta_{k,l,0}$ does not fully incorporate the interaction between the $k$th and the $l$th nodes in Simulation 2 and Simulation 3. Thus in Simulation 2 and Simulation 3, BNR is not expected to perform significantly better than its competitors. Indeed, Table 4 and 5 demonstrates comparable performance of all competitors with BNR slightly outperforming in the low-sparsity cases which are less conducive to ordinary high dimensional regression methods.

**Inference on the effective dimensionality**

Next, the attention turns to inferring on the posterior expected value of the effective dimensionality of $u_k$. Figure 3 presents posterior probabilities of effective dimensionality in all 9 cases in Simulation 1. The filled bullets indicate the true value of the effective dimensionality. All 9 figures indicate that the true dimensionality of the latent variable $u_k$ is effectively captured by the models. Mostly, the performance seems to have been impacted very negligibly by the change in sparsity or discrepancy between $R$ and $R_{gen}$. Only in cases 8 and 9, in presence of less network information, the posterior distribution of $R_{eff}$ appears to be bimodal. We also investigate similar figures for simulations 2 and 3 and arrive at an identical conclusion. In the interest of space, we omit these figures from the main text.
Figure 2: True activity status of a network node (clear black background denotes inactive and dark background denotes active). Note that there are 20 rows (corresponding to 20 nodes) and 2 columns corresponding to 2 different cases both in Simulations 1 and 2. The model-detected posterior probability of being active has been super-imposed onto the corresponding node.

<table>
<thead>
<tr>
<th>Cases</th>
<th>$R_{gen}$</th>
<th>$R$</th>
<th>Sparsity</th>
<th>MSE BNR</th>
<th>Lasso</th>
<th>Relión(2017)</th>
<th>BLasso</th>
<th>Horseshoe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case - 1</td>
<td>2</td>
<td>2</td>
<td>0.5</td>
<td><strong>0.009</strong></td>
<td>0.438</td>
<td>0.524</td>
<td>0.472</td>
<td>0.395</td>
</tr>
<tr>
<td>Case - 2</td>
<td>2</td>
<td>3</td>
<td>0.6</td>
<td><strong>0.007</strong></td>
<td>0.660</td>
<td>0.929</td>
<td>0.863</td>
<td>0.012</td>
</tr>
<tr>
<td>Case - 3</td>
<td>2</td>
<td>5</td>
<td>0.3</td>
<td><strong>0.006</strong></td>
<td>1.295</td>
<td>1.117</td>
<td>1.060</td>
<td>1.070</td>
</tr>
<tr>
<td>Case - 4</td>
<td>2</td>
<td>5</td>
<td>0.4</td>
<td><strong>0.006</strong></td>
<td>0.371</td>
<td>0.493</td>
<td>0.699</td>
<td>0.298</td>
</tr>
<tr>
<td>Case - 5</td>
<td>3</td>
<td>5</td>
<td>0.5</td>
<td><strong>0.009</strong></td>
<td>1.344</td>
<td>1.629</td>
<td>1.638</td>
<td>1.381</td>
</tr>
<tr>
<td>Case - 6</td>
<td>4</td>
<td>5</td>
<td>0.4</td>
<td><strong>0.006</strong></td>
<td>3.054</td>
<td>2.601</td>
<td>2.680</td>
<td>3.284</td>
</tr>
<tr>
<td>Case - 7</td>
<td>2</td>
<td>4</td>
<td>0.5</td>
<td><strong>0.009</strong></td>
<td>0.438</td>
<td>0.524</td>
<td>0.472</td>
<td>0.395</td>
</tr>
<tr>
<td>Case - 8</td>
<td>2</td>
<td>4</td>
<td>0.7</td>
<td><strong>0.005</strong></td>
<td>0.015</td>
<td>0.251</td>
<td>0.007</td>
<td>0.008</td>
</tr>
<tr>
<td>Case - 9</td>
<td>3</td>
<td>5</td>
<td>0.7</td>
<td><strong>0.004</strong></td>
<td>0.029</td>
<td>0.071</td>
<td>0.019</td>
<td>0.007</td>
</tr>
</tbody>
</table>

Table 3: Performance of Bayesian Network Regression (BNR) vis-a-vis competitors for cases in Simulation 1. Parametric inference in terms of point estimation of edge coefficients has been captured through the Mean Squared Error (MSE). The minimum MSE among competitors for any case is made bold.
Figure 3: Plots showing posterior probability distribution of effective dimensionality in all 9 cases in Simulation 1. Filled bullets indicate the true value of effective dimensionality.
Table 4: Performance of Bayesian Network Regression (BNR) vis-a-vis competitors for cases in Simulation 2. Parametric inference in terms of point estimation of edge coefficients has been captured through the Mean Squared Error (MSE). The minimum MSE among competitors for any case is made bold.

<table>
<thead>
<tr>
<th>Cases</th>
<th>$R_{gen}$</th>
<th>$R$</th>
<th>Sparsity</th>
<th>BNR</th>
<th>Lasso</th>
<th>Relión(2017)</th>
<th>BLasso</th>
<th>Horseshoe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case - 1</td>
<td>3</td>
<td>5</td>
<td>0.7</td>
<td>0.011</td>
<td>0.013</td>
<td>0.036</td>
<td>0.010</td>
<td><strong>0.008</strong></td>
</tr>
<tr>
<td>Case - 2</td>
<td>3</td>
<td>5</td>
<td>0.2</td>
<td><strong>0.629</strong></td>
<td>0.843</td>
<td>0.859</td>
<td>0.836</td>
<td>0.948</td>
</tr>
</tbody>
</table>

Table 5: Performance of Bayesian Network Regression (BNR) vis-a-vis competitors for cases in Simulation 3. Parametric inference in terms of point estimation of edge coefficients has been captured through the Mean Squared Error (MSE). The minimum MSE among competitors for any case is made bold.

<table>
<thead>
<tr>
<th>Cases</th>
<th>$R_{gen}$</th>
<th>$R$</th>
<th>Node Sparsity</th>
<th>Edge Sparsity</th>
<th>BNR</th>
<th>Lasso</th>
<th>Relión(2017)</th>
<th>BLasso</th>
<th>Horseshoe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case - 1</td>
<td>3</td>
<td>5</td>
<td>0.7</td>
<td>0.5</td>
<td>0.004</td>
<td>0.006</td>
<td>0.017</td>
<td>0.004</td>
<td><strong>0.003</strong></td>
</tr>
<tr>
<td>Case - 2</td>
<td>3</td>
<td>5</td>
<td>0.2</td>
<td>0.5</td>
<td><strong>0.457</strong></td>
<td>0.636</td>
<td>0.617</td>
<td>0.659</td>
<td>0.629</td>
</tr>
</tbody>
</table>

3.3 Predictive Inference

For the purpose of assessing predictive inference of the competitors, $n_{pred} = 30$ samples are generated from (6). We compare the predictive ability of competitors based on the point prediction and characterization of predictive uncertainties. To assess point prediction, we employ the mean squared prediction error (MSPE) which is obtained as the average squared distance between the point prediction and the true responses for all the competitors. As measures of predictive uncertainty, we provide coverage and length of 95% predictive intervals. For frequentist competitors, 95% predictive intervals are obtained by using predictive point estimates plus and minus 1.96 times standard errors. Figure 4 provides all three measures for all competitors in the 9 cases for Simulation 1.

It is quite evident from Figure 4(a) that BNR remarkably outperforms other competitors in terms of point prediction. Among the competitors, Horseshoe does a reasonably good job in cases with a higher degree of sparsity. This can be explained by taking into account the fact that the data generation procedure in Simulation 1 ensures predictor coefficients to take reasonably large positive and negative values, along with an overwhelming number set to zero. Clearly, with a small sample size, all local and global parameters for the ordinary vector shrinkage priors are shrunk to zero to provide a good estimation of zero coefficients. While doing so, they heavily miss out on estimating coefficients which significantly deviate from
zero. Thus, the ordinary vector shrinkage priors do a modest job in terms of prediction. Lasso also fails to provide satisfactory performance for similar reasons. It is important to note that all competitors provide close to nominal coverage. However, BNR yields so with predictive intervals $8 - 10$ times narrower than other competitors. The precise characterization of uncertainty in the BNR is due to fact that the relational nature of the predictor is properly accounted for. On the contrary, the ordinary high dimensional regression models do not account for the relational nature of the predictor and thus inflate the predictive interval. Overall, results from predictive inference from Simulation 1 demonstrate strikingly superior performance of BNR.

The predictive performance of all competitors corresponding to simulations 2 and 3 are given in Table 6 and 7 respectively. Since the data generation scheme in simulations 2 and 3 minimally incorporates network structure, BNR does not shine much over its competitors in the high sparsity case. As expected, in presence of low node sparsity, performance of Lasso, BLasso and HS deteriorate and BNR turn out to be the best performer. Thus, when the data generation procedure incorporates minimal network structure, BNR shows comparable or little better performance than its competitors. Substantial gain using BNR is evident when the network structure is prominent as in Simulation 1.

### 4 Application to Human Brain Network Data

This section illustrates the inferential and predictive ability of Bayesian network regression in the context of a diffusion tensor imaging (DTI) dataset. Along with the brain network data, the dataset of interest contains a measure of creativity (Kiar et al. (2016); Kiar et al. (2017a);
### Table 6: MSPE, coverage and length of 95% predictive intervals (PIs) of Bayesian Network Regression (BNR) vis-a-vis competitors for cases in Simulation 2. Lowest MSPE for any case is made bold.

<table>
<thead>
<tr>
<th>Cases</th>
<th>$R_{gen}$</th>
<th>$R$</th>
<th>Sparsity</th>
<th>BNR</th>
<th>Lasso</th>
<th>Reli´on(2017)</th>
<th>Blasso</th>
<th>Horseshoe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case - 1</td>
<td>3</td>
<td>5</td>
<td>0.7</td>
<td>0.079</td>
<td>0.100</td>
<td>0.371</td>
<td>0.076</td>
<td>0.061</td>
</tr>
<tr>
<td>Case - 2</td>
<td>3</td>
<td>5</td>
<td>0.2</td>
<td><strong>0.432</strong></td>
<td>0.726</td>
<td>0.859</td>
<td>0.629</td>
<td>0.725</td>
</tr>
</tbody>
</table>

| Coverage of 95% PI |
|-------------------|-----------|-------|----------|-----------|-------|---------------|--------|
| Case - 1           | 3         | 5     | 0.7      | 1.00      | 1.00  | 0.867         | 1.00   | 0.967     |
| Case - 2           | 3         | 5     | 0.2      | 0.94      | 0.73  | 0.56          | 0.96   | 0.87      |

| Length of 95% PI  |
|-------------------|-----------|-------|----------|-----------|-------|---------------|--------|
| Case - 1           | 3         | 5     | 0.7      | 8.97      | 18.70 | 10.23         | 8.25   | 6.40      |
| Case - 2           | 3         | 5     | 0.2      | 42.18     | 32.81 | 23.54         | 45.69  | 42.91     |

### Table 7: MSPE, coverage and length of 95% predictive intervals (PIs) of Bayesian Network Regression (BNR) vis-a-vis competitors for cases in Simulation 3. Lowest MSPE for any case is made bold.

<table>
<thead>
<tr>
<th>Cases</th>
<th>$R_{gen}$</th>
<th>$R$</th>
<th>Node Sparsity</th>
<th>Edge Sparsity</th>
<th>BNR</th>
<th>Lasso</th>
<th>Reli´on(2017)</th>
<th>Blasso</th>
<th>Horseshoe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case - 1</td>
<td>3</td>
<td>5</td>
<td>0.7</td>
<td>0.5</td>
<td>0.119</td>
<td>0.151</td>
<td>0.425</td>
<td>0.125</td>
<td><strong>0.096</strong></td>
</tr>
<tr>
<td>Case - 2</td>
<td>3</td>
<td>5</td>
<td>0.2</td>
<td>0.5</td>
<td><strong>0.451</strong></td>
<td>0.549</td>
<td>0.699</td>
<td>0.692</td>
<td>0.566</td>
</tr>
</tbody>
</table>

| Coverage of 95% PI |
|-------------------|-----------|-------|---------------|-----------|-------|---------------|--------|
| Case - 1           | 3         | 5     | 0.7           | 0.5       | 0.93   | 1.00          | 0.86   | 0.96      | 0.96    |
| Case - 2           | 3         | 5     | 0.2           | 0.5       | 1.00   | 0.83          | 0.70   | 1.00      | 1.00    |

| Length of 95% PI  |
|-------------------|-----------|-------|---------------|-----------|-------|---------------|--------|
| Case - 1           | 3         | 5     | 0.7           | 0.5       | 6.18   | 14.19         | 8.35   | 6.44      | 5.91    |
| Case - 2           | 3         | 5     | 0.2           | 0.5       | 41.69  | 27.98         | 17.84  | 51.70     | 49.12   |

Kiar et al. (2017b)) for several subjects, known as the Composite Creativity Index (CCI). The scientific goal in this setting pertains to understanding the relationship between brain connectivity and the composite creativity index (CCI). Specifically, our main interest lies in proposing a regression relationship between the brain network and CCI, and subsequently predicting the CCI of a subject from his/her brain network, which serves as the predictor. Additionally, it is desirable to identify brain regions (nodes in the brain network) that are involved with creativity, as well as significant connections between different brain regions. Our analysis involves 37 subjects.

Human creativity has been at the crux of the evolution of the human civilization, and has been the topic of research in several disciplines including, of course, neuroscience. Though
creativity can be defined in numerous ways, one could envision a creative idea as one that is unusual as well as effective in a given social context (Flaherty (2005)). Neuroscientists generally concur that a coalescence of several cognitive processes determines the creative process, which often involves a divergence of ideas to conceivable solutions for a given problem. To measure the creativity of an individual, Jung et al. (2010) propose the CCI, which is formulated by linking measures of divergent thinking and creative achievement to cortical thickness of young (23.7 ± 4.2 years), healthy subjects. Three independent judges grade the creative products of a subject from which the “composite creativity index” (CCI) is derived. CCI serves as the response in our study.

The brain network information for subjects is gathered using the diffusion tensor imaging technique. Diffusion tensor imaging (DTI) is a magnetic resonance imaging technique that enables measurement of the restricted diffusion of water in tissue in order to produce neural tract images. In the context of DTI, the human brain is divided according to the Desikan atlas (Desikan et al. (2006)) that identifies 34 cortical regions of interest (ROIs) both in the left and right hemispheres of the human brain, implying 68 cortical ROIs in all. A ‘brain network’ for each subject is represented by a symmetric adjacency matrix whose rows and columns correspond to different ROIs and entries correspond to estimates of the number of ‘fibers’ connecting pairs of brain regions. Thus, for each individual, representing the brain network, is a weighted adjacency matrix of dimension 68 × 68, with the (k,l)th off-diagonal entry in the adjacency matrix being the estimated number of fibers connecting the kth and the lth brain regions. Figure 5 shows maps of the brain network for two representative individuals.

4.1 Findings from BNR

We focus on identifying influential ROIs in the brain network using the node selection strategy described in the simulation studies. For the purpose of this data analysis, the Bayesian network regression model is fitted with $R = 5$ which is found to be sufficient for this study. Recall that the kth node is identified as active if $P(\xi_k = 1 \mid Data)$ exceeds 0.5. This principle, when applied to the real data discussed above, identifies 36 ROIs out of 68 as active. Of these 36 ROIs, 16 belong to the left portion of the brain (or the left hemisphere) and 20 belong to the right hemisphere. Table 9 shows the brain regions of interest (ROIs) in the Desikan atlas detected as being actively associated with the CCI.

A large number of the 36 active nodes detected by our method are part of the frontal (15) and temporal (8) cortices in both hemispheres. The frontal cortex has been scientifically associated with divergent thinking and problem solving ability, in addition to motor function, spontaneity, memory, language, initiation, judgement, impulse control, and social behavior (Stuss et al. (1985)). Some of the other functions directly related to the frontal cortex
Figure 5: Figure shows maps of the brain network (weighted adjacency matrices) for two representative individuals in the sample. Since the $(k, l)$th off-diagonal entry in any adjacency matrix corresponds to the number of fibers connecting the $k$th and the $l$th ROIs, the adjacency matrices are symmetric. Hence the figure only shows the upper triangular portion.

seem to be “behavioral spontaneity”, interpreting environmental feedback and risk taking (Razumnikova (2007); Miller and Milner (1985); Kolb and Milner (1981)). On the other hand, Finkelstein et al. (1991) report de novo artistic expression to be associated with the temporal and frontal regions. Our method also finds a strong relationship between creativity and the right parahippocampal gyrus and right inferior parietal lobule, regions found active by a few earlier scientific studies, see e.g., Chavez et al. (2004).

As a reference point to our analysis, we compare our findings with Jung et al. (2010). It is worth mentioning that Jung et al. (2010) does not employ a sophisticated model based Bayesian analysis for identifying important nodes with uncertainties. Hence, we do not expect a complete overlap with their results. Our analysis finds a number of overlaps with the regions that Jung et al. (2010) identify as significantly associated with the creativity process, namely the middle frontal gyrus, the left cingulate cortex, the left orbitofrontal region, the left lingual region, the right fusiform, the left cuneus and the right superior parietal lobule. They also find the inferior parietal, superior parietal lobules and the right posterior cingulate regions to be very significantly associated with creativity. Our model detects these nodes as active. Although there is significant intersection between the findings of Jung et al. (2010) and our method, there are a couple of regions that we detect as active and they do not, and vice versa. For e.g., our model detects the precuneus and the supramarginal regions in both the hemispheres to be significantly related to CCI, while they do not. On the other hand,
they identify the right angular region to be significant while we do not.

Along with the influential ROIs, another important scientific question remains identifying the statistically significant edges or connections between the 68 ROIs. We consider the edge between two ROIs \(k\) and \(l\) to have a statistically significant impact on CCI if the 95\% credible interval of the posterior distribution of its corresponding coefficient \(\gamma_{k,l}\) does not contain 0. Under this measure, our model identifies 576 significant \(\gamma_{k,l}\)'s connecting 30 ROIs. Figure 6 plots significant inter-connections detected among brain regions of interest (ROIs), where the brain can be viewed from different angles. Red dots show the active ROIs and blue lines show significant connections between them. It is to be noted that all 30 of the aforementioned nodes are a subset of the group of 36 nodes that have been detected as active. This hints at effective detection. Figure 7 plots these influential interconnections, where a white cell represents an edge predictive of the response with the corresponding row ROI and column ROI. Since this is an undirected network, the matrix is symmetric and we only show connections in the upper triangular region.

Finally, our interest turns to the predictive ability of the Bayesian network regression model. To this end, Table 8 reports the mean squared prediction error (MSPE) between observed and predicted responses, length and coverage of 95\% predictive intervals. Here, the average is computed over 10 cross-validated folds. As reference, we also present MSPE, length and coverage values for Lasso, BLasso and Relión et al. (2017). Note that Lasso and BLasso completely ignore the network structure in the predictor. Thus the edge predictors become highly correlated, with the correlation structure governed by the network structure in the predictor. It is well known that high dimensional regression models perform poorly in terms of predictive performance when the sample size is small and predictors are highly correlated. In the present context, the poor MSPE of Lasso and BLasso can be attributed to this phenomenon. The table clearly shows excellent point prediction of our proposed approach even under small sample size and low signal to noise ratio. Additionally, BNR provides close to nominal coverage for 95\% predictive intervals with a much narrower predictive interval. Hence, uncertainty quantification from BNR also turns out to be precise.

<table>
<thead>
<tr>
<th></th>
<th>BNR</th>
<th>Lasso</th>
<th>BLasso</th>
<th>Relión (2017)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSPE</td>
<td>0.69</td>
<td>0.98</td>
<td>1.84</td>
<td>0.98</td>
</tr>
<tr>
<td>Coverage of 95% PI</td>
<td>0.93</td>
<td>0.97</td>
<td>0.97</td>
<td>0.97</td>
</tr>
<tr>
<td>Length of 95% PI</td>
<td>2.72</td>
<td>3.88</td>
<td>3.40</td>
<td>3.89</td>
</tr>
</tbody>
</table>

Table 8: Predictive performance of competitors in terms of mean squared prediction error (MSPE), coverage and length of 95\% predictive intervals, obtained through 10-Fold Cross Validation in the context of real data. Note that since the response has been standardized, an MSPE value greater than or around 1 will denote an inconsequential analysis.
Figure 6: Significant inter-connections detected among brain regions of interest (ROIs) in the Desikan atlas. Red dots show the active ROIs and blue lines show significant connections between them.

5 Conclusion and Future Work

This article proposes a novel and pioneering Bayesian framework to address a regression problem with a continuous response and network-valued predictors, respecting the underlying network structure. Our contribution lies in carefully constructing a novel class of network shrinkage priors corresponding to the network predictor, simultaneously recognizing the latent network structure in the predictor variable. Empirical results from simulation studies display remarkably superior performance of our method, both in terms of inference as well as prediction. Our framework is employed to analyze a brain connectome data on composite creativity index along with the brain network of multiple individuals. It is able to identify important regions in the brain and important brain connectivity patterns which have profound influence on the creativity of a person.

A number of future directions emerge from this work. First, our framework finds natural extension to regression problems with a binary response and any network predictor, whether binary or weighted. Such a framework would be useful in various classification problems involving network predictors, e.g. in classifying diseased patients from normal people in neuroimaging studies. Another important direction appears to be the development of a regression framework with the network as the response regressed on a few scalar/vector predictors. Some of these constitute our current work.
Table 9: Brain regions (ROIs) detected as actively associated with the composite creativity index by BNR.

### Appendix A

This section provides details of posterior computation for all the parameters in the Bayesian network regression with a continuous response.

Let \( \mathbf{x}_i = (a_{i,1,2}, \ldots, a_{i,1,3}, \ldots, a_{i,1,V}, a_{i,2,1}, \ldots, a_{i,2,3}, \ldots, a_{i,V-1,1}, \ldots, a_{i,V-1,V})' \) be of dimension \( q \times 1 \), where \( q = \frac{V \times (V-1)}{2} \). Assume \( \mathbf{y} = (y_1, \ldots, y_n)' \in \mathbb{R}^n \) and \( \mathbf{X} = (\mathbf{x}_1 : \cdots : \mathbf{x}_n)' \) is an \( n \times q \) matrix. Further, assume \( \mathbf{W} = (\mathbf{u}_1' \mathbf{A} \mathbf{u}_2, \ldots, \mathbf{u}_1' \mathbf{A} \mathbf{u}_V, \ldots, \mathbf{u}_{V-1}' \mathbf{A} \mathbf{u}_V)' \), \( \mathbf{D} = diag(s_{1,2}, \ldots, s_{V-1,V}) \) and \( \gamma = (\gamma_{1,2}, \ldots, \gamma_{V-1,V})' \). Thus, with \( n \) data points, the hierarchical model with the Bayesian Network Lasso prior can be written as

\[
\begin{align*}
\mathbf{y} & \sim \mathcal{N}(\mu + \mathbf{X}\gamma, \tau^2 \mathbf{I}) \\
\gamma & \sim \mathcal{N}(\mathbf{W}, \tau^2 \mathbf{D}) \quad \log(\tau^2) \sim \text{flat}() \\
\mathbf{u}_k | \xi_k = 1 & \sim \mathcal{N}(\mathbf{u}_k | \mathbf{0}, \mathbf{M}) \\
\mathbf{u}_k | \xi_k = 0 & \sim \delta_0 \\
\mu & \sim \text{flat}() \\
\mathbf{s}_{k,l} & \sim \text{Exp}(\lambda^2 / 2) \\
\lambda & \sim \text{Gamma}(r_\lambda, \delta) \\
\mathbf{M} & \sim \text{IW}(\mathbf{S}, \nu) \\
\Delta & \sim \text{Beta}(a_\Delta, b_\Delta) \\
\xi_k & \sim \text{Ber}(\Delta) \\
\lambda_r & \sim \text{Ber}(\pi_r) \\
\pi_r & \sim \text{Beta}(1, 1/\eta) \quad \eta > 1.
\end{align*}
\]

The hierarchical model specified above leads to straightforward Gibbs sampling with full conditionals obtained as following:
Figure 7: Significant inter-connections detected among brain regions of interest (ROIs) in the Desikan atlas. White cells show significant nodal associations among ROIs denoted by the corresponding rows and columns. Prefix ‘lh-’ and ‘rh-’ in the ROI names denote their positions in the left and right hemispheres of the brain respectively. For full names of the ROIs specified on the axes, please consult the widely available Desikan Atlas.

\[ \mu | \sim N \left( \frac{1}{n} (y - X\gamma), \frac{\tau^2}{n} \right) \]

\[ \gamma | \sim N(\mu_{\gamma | \cdot}, \Sigma_{\gamma | \cdot}), \text{ where } \mu_{\gamma | \cdot} = (X'X + D^{-1})^{-1} (X'(y - \mu 1) + D^{-1}W) \text{ and } \Sigma_{\gamma | \cdot} = \tau^2 (X'X + D^{-1})^{-1} \]

\[ \tau^2 | \sim IG \left[ \frac{n}{2} + \frac{1}{4} (V - 1), \frac{(y - \mu 1 - X\gamma)'(y - X\gamma) + (\gamma - W)'D^{-1}(\gamma - W)}{2} \right] \]
• $s_{k,l} \mid \sim GIG \left[ \frac{1}{2}, \frac{(\gamma_{k,l} - u_k^\prime \Lambda u_k)^2}{\tau^2}, \lambda^2 \right]$, where GIG denotes the generalized inverse Gaussian distribution.

• $\lambda^2 \mid \sim Gamma \left[ \left( r_\lambda + \frac{V(V-1)}{2} \right), \left( \delta + \sum_{k<l} s_{k,l} \right) \right]

• $u_k \mid \sim w_{u_k} \delta_0(u_k) + (1 - w_{u_k}) N(u_k \mid m_{u_k}, \Sigma_{u_k})$, where $U^*_k = (u_1 : \ldots : u_{k-1} : u_{k+1} : \ldots : u_V)' \Lambda, H_k = diag(s_{1,k}, ..., s_{k-1,k}, s_{k+1,k}, ..., s_{k,V}), \gamma_k = (\gamma_{1,k}, ..., \gamma_{k-1,k}, \gamma_{k+1,k}, ..., \gamma_{k,V})$, and

$$\Sigma_{u_k} = \left( U_h^\prime H_k^{-1} U_k^\prime / \tau^2 + M^{-1} \right)^{-1}, \quad m_{u_k} = \Sigma_{u_k} U^\prime_k H_k^{-1} \gamma_k / \tau^2$$

$$w_{u_k} = \frac{(1 - \pi) N(\gamma_k \mid 0, \tau^2 H_k)}{(1 - \pi) N(\gamma_k \mid 0, \tau^2 H_k) + \pi N(\gamma_k \mid 0, \tau^2 H_k + U_k^\prime M U_k^\prime)}$$

• $\xi_k \mid \sim Ber(1 - w_{u_k})$

• $\Delta \mid \sim Beta \left[ (a_\Delta + \sum_{k=1}^V \xi_k), (b_\Delta + \sum_{k=1}^V (1 - \xi_k)) \right]$

• $M \mid \sim IW(\{S + \sum_{k:u_k \neq 0} u_k \Lambda u_k^\prime\}, (\nu + \{k : u_k \neq 0\}))$

• $\lambda_r \mid \sim Ber(p_{\lambda_r})$, where $p_{\lambda_r} = \frac{\pi_r N(\gamma \mid W_1, \tau^2 D)}{\pi_r N(\gamma \mid W_1, \tau^2 D) + (1 - \pi_r) N(\gamma \mid W_0, \tau^2 D)}$. Here

$W_1 = (u_1^\prime \Lambda_1 u_2, ..., u_1^\prime \Lambda_1 u_V, ..., u_{V-1}^\prime \Lambda_1 u_V)'$, $W_0 = (u_1^\prime \Lambda_0 u_2, ..., u_1^\prime \Lambda_0 u_V, ..., u_{V-1}^\prime \Lambda_0 u_V)'$,

$\Lambda_1 = diag(\lambda_1, ..., \lambda_{r-1}, 1, \lambda_{r+1}, ..., \lambda_R)$, $\Lambda_0 = diag(\lambda_1, ..., \lambda_{r-1}, 0, \lambda_{r+1}, ..., \lambda_R)$, for $r = 1, ..., R$.

• $\pi_r \mid \sim Beta(\lambda_r + 1, 1 - \lambda_r + r^g)$, for $r = 1, ..., R$.

Appendix B

This section shows the posterior propriety of the parameters in the BNR model. Without loss of generality, we set $\mu = 0$ while proving the posterior propriety. To begin with, we state a number of useful lemmas.

**Lemma 5.1** If $C$ is an $h \times h$ non-negative definite matrix, then $|C + I| \geq 1$.

**Proof** The eigenvalues of $(C + I)$ are given by $\varphi_1 + 1, ..., \varphi_m + 1$, where $\varphi_1, ..., \varphi_m$ are eigenvalues of $C$. Since $C$ is non-negative definite, $\varphi_1 \geq 0, ..., \varphi_m \geq 0$. The result follows from the fact that $|C + I| = \prod_{m=1}^h (\varphi_m + 1)$ is the product of eigenvalues.

**Lemma 5.2** If $C$ is an $h \times h$ diagonal matrix with diagonal entries $c_1, ..., c_h$ all greater than 0. Suppose $A$ is an $n \times h$ matrix with the largest eigenvalue of $AA'$ is given by $\mu_{AA'}$. Then $ACA' + I \leq \left( \mu_{AA'} \sum_{l=1}^h c_l + 1 \right) I$, where $H_1 \leq H_2$ implies $H_2 - H_1$ is a positive definite matrix.
Proof Since \( c_1, ..., c_h > 0 \), \( AC A' \leq (\sum_{i=1}^{h} c_i) AA' \). Consider the spectral decomposition of the matrix \( AA' \). Let the eigen-decomposition of \( AA' = \Lambda H A' \), where \( \Lambda \) is the matrix of eigenvectors and \( H \) is a diagonal matrix with diagonal entries \( \mu_1, ..., \mu_n \). Since each \( \mu_i \leq \mu_{AA'}, \ AA' \leq \mu_{AA'} \AA' = \mu_{AA'} I \). Thus, \( AC A' \leq (\sum_{i=1}^{h} c_i) \mu_{AA'} I \). Hence \( AC A' + I \leq \left( \mu_{AA'} \sum_{i=1}^{h} c_i + 1 \right) I \).

Lemma 5.3 Suppose \( z \) is an \( h \times 1 \) vector and \( A \) is an \( h \times h \) positive definite matrix. Let \( B \) be another \( h \times h \) positive definite matrix such that \( A \geq B \) (where \( A \geq B \) implies \( A - B \) is non-negative definite). Then \( z' A^{-1} z \leq z' B^{-1} z \).

Proof \( A \geq B \) implies \( A^{-1} \leq B^{-1} \). Then \( z' A^{-1} z \leq z' B^{-1} z \).

Note that the posterior distribution of the parameters is given by:
\[
p(\gamma, \Lambda, \tau^2, \theta, u_1, ..., u_v, \xi_1, ..., \xi_v, \lambda^2, \Delta, \{s_{k,l}\}_{k<l}, \pi_1, ..., \pi_R, M \mid y, X) \propto N(y \mid X \gamma, \tau^2 I) \times N(\gamma \mid W, \tau^2 D) \times
\frac{1}{\tau^2} \times \prod_{k=1}^{v} [\Delta N(u_k \mid 0, M) + (1 - \Delta)\delta_0] \times \prod_{k<l} Exp(s_{k,l} \mid \lambda^2 / 2) \times \text{Gamma}(\lambda^2 \mid r_\lambda, \delta) \times
IW(M \mid S, \nu) \times \text{Beta}(\Delta \mid a_\Delta, b_\Delta) \times \text{Ber}(\lambda_r \mid \pi_r) \times \text{Beta}(\pi_r \mid 1, 1) \times \prod_{k=1}^{v} \text{Ber}(\xi_k \mid \Delta).
\]

Integrating over \( \xi_1, ..., \xi_v \),
\[
p(\gamma, \Lambda, \tau^2, \theta, u_1, ..., u_v, \lambda^2, \Delta, \{s_{k,l}\}_{k<l}, \pi_1, ..., \pi_R, M \mid y, X) \propto N(y \mid X \gamma, \tau^2 I) \times N(\gamma \mid W, \tau^2 D) \times
\frac{1}{\tau^2} \times \prod_{k=1}^{v} [\Delta N(u_k \mid 0, M) + (1 - \Delta)\delta_0] \times \prod_{k<l} Exp(s_{k,l} \mid \lambda^2 / 2) \times \text{Gamma}(\lambda^2 \mid r_\lambda, \delta) \times
IW(M \mid S, \nu) \times \text{Beta}(\Delta \mid a_\Delta, b_\Delta) \times \text{Ber}(\lambda_r \mid \pi_r) \times \text{Beta}(\pi_r \mid 1, 1) \times \prod_{k=1}^{v} \text{Ber}(\xi_k \mid \Delta).
\]

The prior specifications on \( \lambda_r, \pi_r, \Delta \) enable these quantities to be bounded within a finite interval. Thus in showing the posterior propriety of parameters with unbounded range, it is enough to treat these quantities as constant. We treat them as fixed henceforth.

Integrating over \( \gamma \), we obtain,
\[
p(u_1, ..., u_v, \tau^2, \lambda^2, \{s_{k,l}\}_{k<l}, M \mid y, X) \propto \frac{1}{(\tau^2)^{n/2} [XDX' + I]^{1/2}} \times
\exp \left\{ -\frac{(y - WX)'(XDX' + I)^{-1}(y - WX)}{2\tau^2} \right\} \times \prod_{k=1}^{v} [\Delta N(u_k \mid 0, M) + (1 - \Delta)\delta_0] \times
\prod_{k<l} Exp(s_{k,l} \mid \lambda^2 / 2) \times \text{Gamma}(\lambda^2 \mid r_\lambda, \delta) \times IW(M \mid S, \nu).
\]

26
Next, we integrate w.r.t. $\lambda^2$ to obtain

$$
p(\mathbf{u}_1, ..., \mathbf{u}_V, \tau^2, \{s_{k,l}\}_{k<l}, \mathbf{M} \mid \mathbf{y}, \mathbf{X}) \propto \frac{1}{(\tau^2)^{n/2+1}|\mathbf{XD} \mathbf{X}' + \mathbf{I}|^{1/2}} \times 
\exp \left\{ -\frac{(\mathbf{y} - \mathbf{XW})' (\mathbf{XD} \mathbf{X}' + \mathbf{I})^{-1} (\mathbf{y} - \mathbf{XW})}{2\tau^2} \right\} \times 
\prod_{k=1}^{V} |\Delta N(\mathbf{u}_k \mid 0, \mathbf{M}) + (1 - \Delta)\delta_0| \times 
\frac{1}{(\delta + \sum_{k<l} s_{k,l})^{q+r_n}} \times IW(\mathbf{M} \mid \mathbf{S}, \nu). \tag{11}
$$

Note the fact that $\mathbf{D}$ is a diagonal matrix with all positive diagonal entries. Thus $\mathbf{XD} \mathbf{X}'$ is non-negative definite and by using Lemma 5.1, $\frac{1}{|\mathbf{XD} \mathbf{X}' + \mathbf{I}|^{1/2}} < 1$. Further, using Lemma 5.2

$$
\mathbf{XD} \mathbf{X}' + \mathbf{I} \leq \mathbf{XX}' \sum_{k<l} s_{k,l} + \mathbf{I} \leq \left( \mu_{\mathbf{XX}'} \sum_{k<l} s_{k,l} + 1 \right) \mathbf{I},
$$

where $\mathbf{A} \leq \mathbf{B}$ implies $\mathbf{A} - \mathbf{B}$ is a non-negative definite matrix and $\mu_{\mathbf{XX}'}$ is the largest eigenvalue of $\mathbf{XX}'$. Using Lemma 5.3, the above inequality implies

$$(\mathbf{y} - \mathbf{XW})'(\mathbf{XD} \mathbf{X}' + \mathbf{I})^{-1}(\mathbf{y} - \mathbf{XW}) \geq \frac{||\mathbf{y} - \mathbf{XW}||^2}{\mu_{\mathbf{XX}'} \sum_{k<l} s_{k,l} + 1}.$$

Let

$$
\tilde{p}(\mathbf{u}_1, ..., \mathbf{u}_V, \tau^2, \{s_{k,l}\}_{k<l}, \mathbf{M}) = \frac{1}{(\tau^2)^{n/2+1}|\mathbf{XD} \mathbf{X}' + \mathbf{I}|^{1/2}} \times 
\exp \left\{ -\frac{(\mathbf{y} - \mathbf{XW})' (\mathbf{XD} \mathbf{X}' + \mathbf{I})^{-1} (\mathbf{y} - \mathbf{XW})}{2\tau^2} \right\} \times 
\prod_{k=1}^{V} N(\mathbf{u}_k \mid 0, \mathbf{M}) \times 
\frac{1}{(\delta + \sum_{k<l} s_{k,l})^{q+r_n}} \times IW(\mathbf{M} \mid \mathbf{S}, \nu). \tag{12}
$$

With little algebra it can be shown that

$$
p(\mathbf{u}_1, ..., \mathbf{u}_V, \tau^2, \{s_{k,l}\}_{k<l}, \mathbf{M} \mid \mathbf{y}, \mathbf{X}) = \text{constant} \times \sum_{1 \leq j_1, ..., j_l \leq V, 0 \leq l \leq V} \Delta^l (1 - \Delta)^{V-l} \tilde{p}(\mathbf{u}_{j_1}, ..., \mathbf{u}_{j_l}, \mathbf{u}_{j_{l+1}} = 0, ..., \mathbf{u}_{j_V} = 0, \tau^2, \{s_{k,l}\}_{k<l}, \mathbf{M}).
$$

Therefore, the integral of (11) w.r.t all parameters is finite if and only if $\int \tilde{p}(\mathbf{u}_1, ..., \mathbf{u}_V, \tau^2, \{s_{k,l}\}_{k<l}, \mathbf{M}) < \infty$. Henceforth, we will proceed to show that this integral is finite.

With little algebra, we have that $\int IW(\mathbf{M} \mid \nu, \mathbf{S}) \prod_{k=1}^{V} N(\mathbf{u}_k \mid 0, \mathbf{M}) d\mathbf{M} \propto \frac{1}{|\mathbf{S} + \sum_{k=1}^{V} \mathbf{u}_k \mathbf{u}_k' + (\nu + V)/2|}.$
Hence,

\[
\hat{p}(u_1, \ldots, u_V, \tau^2, \{s_{k,l}\}_{k<l}) \leq \text{constant} \times \frac{1}{|S + \sum_{k=1}^{V} u_k u'_k|^{|\nu + V|/2} (\tau^2)^{n/2+1}} \times \frac{1}{(\delta + \sum_{k<l} s_{k,l})^{q+r\lambda}} \times \frac{1}{|XDX' + I|^{1/2}}.
\]

Define \(A = \{(u_1, \ldots, u_V) : ||y - XW||^2 > 1\}\). Then

\[
\int_A \hat{p}(u_1, \ldots, u_V, \tau^2, \{s_{k,l}\}_{k<l}) du_1 \ldots du_V d\tau^2 d\prod_{k<l} s_{k,l} = \int_A \hat{p}(u_1, \ldots, u_V, \tau^2, \{s_{k,l}\}_{k<l}) du_1 \ldots du_V d\tau^2 d\prod_{k<l} s_{k,l} + \int_{A^c} \hat{p}(u_1, \ldots, u_V, \tau^2, \{s_{k,l}\}_{k<l}) du_1 \ldots du_V d\tau^2 d\prod_{k<l} s_{k,l}.
\]

Now,

\[
\int_A \hat{p}(u_1, \ldots, u_V, \tau^2, \{s_{k,l}\}_{k<l}) d\tau^2 d\prod_{k<l} s_{k,l} du_1 \ldots du_V \leq \text{constant} \int \frac{1}{|S + \sum_{k=1}^{V} u_k u'_k|^{|\nu + V|/2} |XDX' + I|^{1/2}} \times \frac{1}{(\delta + \sum_{k<l} s_{k,l})^{q+r\lambda}} \times \frac{1}{(\tau^2)^{n/2+1}} \times \frac{1}{\left(\delta + \sum_{k<l} s_{k,l}\right)^{q+r\lambda}} \times \left\{ \int \frac{1}{|S + \sum_{k=1}^{V} u_k u'_k|^{|\nu + V|/2}} du_1 \ldots du_V \right\} \times \left\{ \int \frac{1}{|XDX' + I|^{1/2}} du_1 \ldots du_V \right\}.
\]

Use the fact that \(q > n\) in all our applications to conclude that \(\int \frac{1}{(\delta + \sum_{k<l} s_{k,l})^{q+r\lambda}} d\prod_{k<l} s_{k,l} = c < \infty\). Similarly,

\[
\int \frac{1}{|S + \sum_{k=1}^{V} u_k u'_k|^{|\nu + V|/2}} du_1 \ldots du_V \leq \int \frac{1}{\prod_{k=1}^{V} |S + u_k u'_k|^{|\nu + V|/2}} du_1 \ldots du_V.
\]

where the first inequality follows from the fact that \(|S + \sum_{k=1}^{V} u_k u'_k| \geq |S + u_k u'_k|\) for all \(k\). The second inequality is a direct application of the Cauchy-Schwartz inequality. By the ratio test of integrals, this integral is finite if \(\int \frac{1}{[(1+u_{k,1})^2 \ldots (1+u_{k,R})^2]^{|\nu + V|/2}} du_k\) is finite. Now use the fact that \(\int \frac{1}{x^{1+\epsilon}} dx < \infty\) for any \(c > 0\) to argue that \(\int \frac{1}{[(1+u_{k,1})^2 \ldots (1+u_{k,R})^2]^2|\nu + V|/2} du_k\) is finite. Hence, \(\int_A \hat{p}(u_1, \ldots, u_V, \tau^2, \{s_{k,l}\}_{k<l}) du_1 \ldots du_V d\tau^2 d\prod_{k<l} s_{k,l} \leq \infty\).
Now consider the expression \( \int_{\mathcal{A}} \tilde{p}(u_1, \ldots, u_V, r^2, \{ s_{k,l} \}_{k<l}) \). It is easy to see that \( \{(u_1, \ldots, u_V) : ||y - XW||^2 \leq 1\} \) is a bounded set, so that the bounded function \( \exp \left\{ -\frac{||y - XW||^2}{2r^2(\mu x' x \sum_{k<l} s_{k,l} + 1)} \right\} \) achieves the maximum value at \( W = W^* \). Thus,

\[
\int_{\mathcal{A}} \tilde{p}(u_1, \ldots, u_V, r^2, \{ s_{k,l} \}_{k<l}) \leq \text{constant} \int \frac{1}{|S + \sum_{k=1}^V u_k u'_k|^{(\nu+V)/2} (\tau^2)^{n/2+1}} \times \\
\exp \left\{ -\frac{||y - XW^*||^2}{2r^2(\mu x' x \sum_{k<l} s_{k,l} + 1)} \right\} \times \frac{1}{(\delta + \sum_{k<l} s_{k,l})^q + r \lambda} |Xdx' + I|^{1/2} \times \\
\left\{ \int \frac{(\mu x' x \sum_{k<l} s_{k,l} + 1)^{n/2}}{|Xdx' + I|^{1/2}(\delta + \sum_{k<l} s_{k,l})^{q + r \lambda}} d\prod_{k<l} s_{k,l} \right\} \times \\
< \infty,
\]

where the last step follows from earlier discussions.

References


