Comparing Dependent Undirected Gaussian Networks[*]

Hongmei Zhang[†,‖], Xianzheng Huang[‡], and Hasan Arshad[§,¶]

Abstract. A Bayesian approach is proposed, which unifies network constructions and comparisons between two longitudinal undirected Gaussian networks on their differentiation status (identical or differential) for data collected at two time points. Utilizing the concept of modeling repeated measures, we construct a joint likelihood of networks. The conditional posterior probability mass function for network differentiation is derived and its asymptotic proposition is theoretically assessed. An alternative approach built upon latent rather than manifest data is proposed to significantly reduce computing burden. Simulations are used to demonstrate and compare the two methods and compare them with existing approaches. Based on epigenetic data collected at different ages, the proposed methods are demonstrated on their ability to detect dependent network differentiations. Our theoretical assessment, simulations, and real data applications support the effectiveness of the proposed methods, although the approach relying on latent data is less efficient.

Keywords: Bayesian methods, efficient Bayesian sampling, Gaussian network testing, latent data likelihood, penalized conditional posterior probability.

1 Introduction

Epigenetics regulates gene functions through its modulation of gene expression. One epigenetic mechanism is DNA methylation, where a methyl group is added to the $5'$ position of the cytosine base of cytosine-phosphate-guanine dinucleotide sites (CpG sites or CpGs) in the DNA. Recent studies suggest that joint activities among CpGs may be more biologically informative and important for their associated health conditions compared to individual CpG sites (Hotta et al., 2018). Interplay between genes or CpG sites can be reflected by Gaussian undirected networks assuming data following a multivariate normal distribution. Such networks offer researchers an opportunity to decipher underlying complex mechanisms leading to the interconnections, which is valuable to the detection of gene markers beneficial to disease prediction and consequently prevention. For data following a multivariate normal distribution, zeros in the covariance matrix represent marginal independence between nodes. The precision matrix, on the other

[†]Division of Epidemiology, Biostatistics, and Environmental Health, School of Public Health, University of Memphis, Memphis, Tennessee, U.S.A., hzhang6@memphis.edu

[‡]Department of Statistics, University of South Carolina, Columbia, South Carolina, U.S.A., huang@stat.sc.edu

[§]Allergy and Clinical Immunology, University of Southampton, Southampton, U.K., s.h.arshad@soton.ac.uk

[¶]The David Hide Asthma and Allergy Research Centre, Isle of Wight, U.K.

[‖]Corresponding author.

hand, represents a conditional independence structure of the distribution with a zero denoting independence between nodes conditional on the remaining nodes. Under the context of joint biological activities of genes, modeling a network of genes via precision matrix better reflects such underlying mechanisms.

In genetic or epigenetic studies, often it is of interest to examine whether joint activities among genes change after a certain treatment or after a certain period of time, and such change may reflect modification of underlying biological pathways of disease etiology. Although most studies under frequentist (Friedman et al., 2008; Mazumder and Hastie, 2012) or Bayesian framework (Wang et al., 2015; Li and Zhang, 2017; Bashir et al., 2019; Shaddox et al., 2020; Ni et al., 2021), focus on network constructions, approaches to compare between networks have been proposed. For comparison between undirected networks, Gill et al. (2010) suggested a procedure to globally test differential undirected graphs applied to genes. The approach is based on strength of genetic associations or interaction between genes. Jacob et al. (2012) compares two graphs by comparing multivariate two-sample means on known graphs using Hotelling's T^2-tests. A method estimates the differences in precision matrices between two differential undirected networks was suggested by Zhao et al. (2014), which was further developed with an ability to globally test differentiation of undirected graphs (Xia et al., 2015). The work by Städler et al. (2017) was under a framework similar to Zhao et al. (2014). More recently, He et al. (2019), based on the inference of Gaussian graphic modeling and asymptotic normality of precision matrix components, proposed a test statistic that has the ability to efficiently compare two precision matrices. Approaches comparing between covariance matrices have been proposed and can also be applied to compare agreement between two Gaussian undirected networks based on precision matrices. For instance, Cai et al. (2013) derived test statistics for two-sample covariance matrix testing; and Chang et al. (2017) proposed a computationally efficient procedure to compare two covariance matrices that is expected to be robust against complex structures. However, to our knowledge, existing methods focus on comparing two independent graphs, which is likely to be underpowered if applied to compare two networks that are dependent.

Motivated by the needs in genetic and epigenetic epidemiological and medical studies with interest in longitudinally network differentiations, in this article we propose two methods to compare dependent networks constructed based on data with repeated measures, in which the novelty lies. The proposed approaches unify network construction and comparison. The remaining of the article is organized as follows. We introduce the modeling of two dependent graphs in Section 2. A Bayesian method used to compare the two dependent graphs and the property of a penalty-incorporated posterior probability are discussed in Section 3. Also in this section, we propose another likelihood-based method that is computationally less demanding and incorporates a block sampling approach to achieve efficient sampling. Simulations with different simulation scenarios and comparison with competing methods are discussed in Section 4. To demonstrate the methods, in Section 5, we present real data applications to compare epigenetic networks between pre- and post-adolescence and during young adulthood. We summarize our work in Section 6.

2 Modeling two dependent graphs

Let $\boldsymbol{X}_{it}$ denote a vector of p variables for subject i at time t, $i = 1, \ldots, n, t = 1, \ldots, T$. Without loss of generality, we assume the mean of $\boldsymbol{X}_{it}$ is $\boldsymbol{0}$. Following the concept of linear mixed models, we decompose $\boldsymbol{X}_{it}$ into two components (Xie et al., 2016),

$$\boldsymbol{X}_{it} = \boldsymbol{X}_{it}^* + \boldsymbol{S}_i, \tag{1}$$

where $\boldsymbol{X}_{it}^*$ are the underlying variables after eliminating random subject effects $\boldsymbol{S}_i$ across the p variables shared between T time points. We assume $\boldsymbol{X}_{it}^*$ and $\boldsymbol{S}_i$ are independent and each follows a multivariate Gaussian distribution with mean $\boldsymbol{0}$ and $p \times p$ covariance matrices Σ_t and Σ_0, respectively. Denoting $\Omega_t = \Sigma_t^{-1}$, $\Omega_0 = \Sigma_0^{-1}$, and $\boldsymbol{X}_{i.} = \{\boldsymbol{X}_{i1}', \ldots, \boldsymbol{X}_{iT}'\}'$, a vector of length pT, the joint density of $\boldsymbol{X}_{i.}$ in terms of Ω_t and Ω_0 can be derived,

$$\prod_{i=1}^{n} p(\boldsymbol{X}_{i.}|\boldsymbol{\Omega}) = (2\pi)^{-\frac{npT}{2}} \left[\frac{\det(\Omega_0)}{\det(A_0)}\right]^{\frac{n}{2}} \prod_{t=1}^{T} \det(\Omega_t)^{\frac{n}{2}} \exp\left\{-\frac{n}{2} \operatorname{tr}(\hat{\Sigma}_{X(t,t)}\Omega_t)\right\}$$

$$\times \exp\left\{\frac{n}{2} \sum_{t_1,t_2=1}^{T} \operatorname{tr}(\Omega_{t_1}\hat{\Sigma}_{X(t_1,t_2)}\Omega_{t_2}A_0^{-1})\right\},$$

$$\propto \left[\frac{\det(\Omega_0)}{\det(A_0)}\right]^{\frac{n}{2}} \prod_{t=1}^{T} \det(\Omega_t)^{\frac{n}{2}} \exp\left\{-\frac{n}{2} \operatorname{tr}(\hat{\Sigma}_{X(t,t)}\Omega_t)\right\}$$

$$\times \exp\left\{\frac{n}{2} \sum_{t_1,t_2=1}^{T} \operatorname{tr}(\Omega_{t_1}\hat{\Sigma}_{X(t_1,t_2)}\Omega_{t_2}A_0^{-1})\right\}, \tag{2}$$

with $\boldsymbol{\Omega} = (\Omega_0, \Omega_1, \ldots, \Omega_T)$, $A_0 = \Omega_0 + \Omega_1 + \cdots + \Omega_T$, $\hat{\Sigma}_{X(t,t)}$ being the sample covariance matrix of $\boldsymbol{X}_{.t} = \{\boldsymbol{X}_{1,t}, \ldots, \boldsymbol{X}_{n,t}\}$ at time t, and $\hat{\Sigma}_{X(t_1,t_2)}$ being the sample covariance matrix between t_1 and t_2, $t_1, t_2 = 1, \ldots, T$. The joint density (2) is derived following $\boldsymbol{X}_{i.} \sim N(\boldsymbol{0}, \Sigma_X)$ with $\Sigma_X = \Omega_X^{-1} = \Omega_1^{-1} \oplus \cdots \oplus \Omega_T^{-1} + J \otimes \Omega_0^{-1}$ (Xie et al., 2016). Covariance Σ_X is a $pT \times pT$ positive definite matrix. The first part of Σ_X represents a matrix with Ω_t^{-1} on the diagonal blocks, and the second part is a Kronecker product between J, a $T \times T$ matrix of 1, and Ω_0^{-1}. More specifically, $\Omega_t^{-1} + \Omega_0^{-1}$ is on the diagonal blocks of Σ_X representing the covariance structure at time t, and Ω_0^{-1} is on the off-diagonal blocks of Σ_X for the covariance among nodes between the two time points.

Graph structures corresponding to $\Omega_0, \Omega_1, \ldots, \Omega_T$ are denoted individually by binary adjacency matrices $G_0, G_1, \ldots, G_T$ such that each entry indicates a connected (represented by the number 1) or disconnected (denoted by 0) edge. We assume that each graph does not have any self-loops (an edge that connects a node to itself) and thus all the diagonal entries in the adjacency matrices are zeros. With dependent networks modeled under this framework as in Xie et al. (2016), in the sections below we describe our network comparison approaches.

3 Comparing two dependent graphs

3.1 The probability density function under different graphical conditions

In the following, we focus on longitudinal data at two time points, i.e., $T = 2$, but the idea is ready to be generalized for $T > 2$. In practice, networks under two conditions or between two time points, represented by two graphs, can be differential or identical. To incorporate such graphical conditions into the definition of density function, we introduce an indicator variable η with $\eta = 1$ denoting two graphs identical and $\eta = 0$ differential. The density function defined in (2) can be viewed as the density associated with the observed data conditioning on $\eta = 0$, that is, $p(\boldsymbol{X}_{i.}|\boldsymbol{\Omega}, \eta = 0)$. Imposing the constraint that $\Omega_1 = \Omega_2 = \Omega_c$ in (2) yields the density function of $\boldsymbol{X}_{i.}$ conditional on $\eta = 1$,

$$\prod_{i=1}^{n} p(\boldsymbol{X}_{i.}|\boldsymbol{\Omega}, \eta = 1) \propto \Big[\frac{\det(\Omega_0)}{\det(A_1)}\Big]^{\frac{n}{2}} \det(\Omega_c)^n \exp\Big\{-\frac{n}{2}\Big[\sum_{t=1}^{2} \mathrm{tr}(\hat{\Sigma}_{X(t,t)}\Omega_c)\Big]\Big\}$$

$$\times \exp\Big\{\frac{n}{2}\sum_{t_1,t_2=1}^{2} \mathrm{tr}(\Omega_c\hat{\Sigma}_{X(t_1,t_2)}\Omega_c A_1^{-1})\Big\}, \tag{3}$$

with $A_1 = \Omega_0 + 2\Omega_c$. Correspondingly, the graph structures are $G_1 = G_2 = G_c$. Since Ω_0 is common between different time points, in graph comparisons, the focus is on Ω_t and G_t.

3.2 Inference on graphical conditions

Inferences on the graphs as well as graphical conditions (η) will be carried out through a fully Bayesian approach. For graph structures, we introduce a vector denoting edge connections, $\boldsymbol{g}_t = \{g_{t,m_1m_2}\}, m_1 \leq m_2 = 1,\ldots,p$, with $g_{t,m_1m_2} = 1, m_1 \neq m_2$, if nodes m_1 and m_2 are connected and the corresponding entry of Ω_t is non-zero, and 0 otherwise. When two graphs are identical, $\Omega_t = \Omega_c$ and $G_t = G_c$ and the vector of edge connections is denoted as $\boldsymbol{g}_c$. Analogously, we define $\boldsymbol{g}_0$ in the same way for edge connections of G_0. To incorporate η to the definitions of $\boldsymbol{G}_t$, Ω_t, $\boldsymbol{G}_c$, and Ω_c, we define $\boldsymbol{G}^{(t)} = I_{\eta=0}\boldsymbol{G}_t + I_{\eta=1}\boldsymbol{G}_c$ and $\Omega^{(t)} = I_{\eta=0}\Omega_t + I_{\eta=1}\Omega_c, t-1, 2$. Edge connections for graph $\boldsymbol{G}^{(t)}$ denoted as $\boldsymbol{g}^{(t)}$ and components in $\Omega^{(t)}$ denoted as $\omega^{(t)}$ are defined analogously.

Let $\Theta = \{\eta, \Omega^{(t)}, \Omega_0, \boldsymbol{g}^{(t)}, \boldsymbol{g}_0, t = 1, 2\}$ be a collection of parameters to be inferred. To draw posterior inferences on the parameters, we start from assigning prior distributions to the parameters.

Prior distributions Since our focus is on network comparisons, for network constructions, we apply established prior distributions. Let $\boldsymbol{\omega}^{(t)} = (\omega^{(t)}_{m_1m_2}, m_1, m_2, m_1 \leq m_2)$ denote a collection of unique parameters in $\Omega^{(t)}$ with (m_1, m_2) being an entry of $\Omega^{(t)}$. We assign the spike-and-slab prior distribution to each $\omega^{(t)}_{m_1m_2}$ with $m_1 \neq m_2$,

$$p(\omega^{(t)}_{m_1m_2}|g^{(t)}_{m_1m_2}, \eta) = g^{(t)}_{m_1m_2}N(0, \nu_{1,t}^2) + (1 - g^{(t)}_{m_1m_2})N(0, \nu_{0,t}^2), \tag{4}$$

conditional on $\Omega^{(t)} \in \mathcal{M}^+$ with $\mathcal{M}^+$ denoting a space of positive definite matrices. The spike-and-slab prior was initially proposed by Mitchell and Beauchamp (1988), was further studied in George and McCulloch (1993) and Ishwaran and Rao (2005), and has been widely applied in variable selections. The hyper-parameters $\nu_{1,t}^2$ and $\nu_{0,t}^2$ are the variances in the two respective normal distributions. In this work, we use the same hyper-parameters for the two time points, i.e., $\nu_{1,t}^2 = \nu_1^2$ and $\nu_{0,t}^2 = \nu_0^2$, since the underlying comparison is between two dependent networks and similarity in network sparsity between the two time points can be assumed. The selection of ν_0 and ν_1 is critical and is discussed in detail after we present the full conditional posterior distributions. For parameters on the diagonal of $\Omega^{(t)}$ with $m_1 = m_2 = m$, $p(\omega_{mm}^{(t)})$ is assumed to be an exponential distribution with parameter λ. Following Wang et al. (2015), we set $\lambda = 1$. Our simulation assessment of selecting different values of λ also indicates that posterior inferences on network comparison are insensitive to different choices of λ. Similar definitions are applied to Ω_0 with conditioning graph structure parameters $\boldsymbol{g}_0$ and hyper-parameters of variances ζ_0 and ζ_1.

For each component in $\boldsymbol{g}^{(t)}$ and $\boldsymbol{g}_0$, we take Bernoulli probability mass function with prior probability parameter π denoting our a priori belief of two nodes being connected, e.g., for $\boldsymbol{g}^{(1)}$, $p(g_{m_1 m_2}^{(1)}) = \pi^{g_{m_1 m_2}^{(1)}}(1-\pi)^{1-g_{m_1 m_2}^{(1)}}, m_1 \neq m_2$. We set $\pi = O(2/(p-1))$ based on a common assumption on the expected number of edges, $O(p)$, for sparse graphs (Jones et al., 2005; Wang et al., 2015). Finally, for the prior mass function of η, we assume a Bernoulli distribution with hyper-parameter 0.5.

Joint likelihood and full conditional posterior distribution   Based on (2) and (3), the joint likelihood of Θ is defined as

$$
\begin{aligned}
L(\Theta|\boldsymbol{X}) = \prod_{i=1}^{n} & p(\boldsymbol{X}_{i.}|\boldsymbol{\Omega}, \eta) \\
\propto & \left\{ (1-\eta) \left[\frac{\det(\Omega_0)}{\det(A_0)} \right]^{\frac{n}{2}} \prod_{t=1}^{T} \det(\Omega^{(t)})^{\frac{n}{2}} \exp\left\{ -\frac{n}{2} \operatorname{tr}(\hat{\Sigma}_{X(t,t)} \Omega^{(t)}) \right\} \right. \\
& \times \exp\left\{ \frac{n}{2} \sum_{t_1, t_2=1}^{T} \operatorname{tr}(\Omega_{t_1} \hat{\Sigma}_{X(t_1,t_2)} \Omega_{t_2} A_0^{-1}) \right\} \\
& + \eta \left[\frac{\det(\Omega_0)}{\det(A_1)} \right]^{\frac{n}{2}} \det(\Omega_c)^n \exp\left\{ -\frac{n}{2} \left[\sum_{t=1}^{2} \operatorname{tr}(\hat{\Sigma}_{X(t,t)} \Omega_c) \right] \right\} \\
& \left. \times \exp\left\{ \frac{n}{2} \sum_{t_1, t_2=1}^{2} \operatorname{tr}(\Omega_c \hat{\Sigma}_{X(t_1,t_2)} \Omega_c A_1^{-1}) \right\} \right\},
\end{aligned} \tag{5}
$$

where $\boldsymbol{X} = \{\boldsymbol{X}_{1.}', \dots, \boldsymbol{X}_{n.}'\}'$ and $\boldsymbol{\Omega} = \{\Omega^{(t)}, \Omega_0, t = 1, 2\}$. With the prior distributions specified earlier, the joint posterior distribution of Θ is defined as,

$$
p(\Theta|\boldsymbol{X}) \propto \prod_{i=1}^{n} p(\boldsymbol{X}_{i.}|\boldsymbol{\Omega}, \eta) p(\boldsymbol{\Omega}|\eta, \boldsymbol{g}^{(t)}, \boldsymbol{g_0}) \left\{ p(\boldsymbol{g_1}) p(\boldsymbol{g_2}) \right\}^{I(\eta=0)} \left\{ p(\boldsymbol{g_c}) \right\}^{I(\eta=1)} p(\eta) p(\boldsymbol{g_0}).
\tag{6}
$$

To infer the parameters, we utilize Markov Chain Monte Carlo (MCMC) simulations, in particular, the Gibbs sampler, to sample from full conditional posterior distributions of each parameter, which can be derived straightforwardly based on the joint posterior distribution (6). Posterior samples are then drawn using the Gibbs sampler, based on which we infer graph structures and η. In the following, we present conditional posterior distributions of each parameter in Θ. We use $(\cdot)$ to denote a collection of other parameters which a parameter is conditioned on in its conditional posterior distribution.

We start from the conditional posterior probability of η, the main focus of our work and a critical parameter for subsequent inferences on graph structures. Based on (6), $p(\eta = 1|(\cdot), \boldsymbol{X}) \propto \prod_{i=1}^{n} p(\boldsymbol{X}_{i.}|\boldsymbol{\Omega}_0, \eta = 1, \boldsymbol{g_c})p(\eta = 1)$. We show in Appendix A (Zhang et al., 2022) that $p(\eta = 1|(\cdot), \boldsymbol{X})$ can be approximated by

$$p(\eta = 1|(\cdot), \boldsymbol{X}) \approx \left[1 + \exp\{\log(b) - \log(a) + \lambda(n)\}\right]^{-1} \equiv p^\lambda(\eta = 1|(\cdot)),$$
$$\lambda(n) = (\log n)/2(|E| - |E_1| - |E_2|), \tag{7}$$

where $|E|$ denotes the number of edges in an inferred network when $\eta = 1$ (identical networks), and $|E_t|, t = 1, 2$, the number of edges of an inferred network at time t when $\eta = 0$ (differential networks). The two parameters a and b are defined as follows,

$$a = \prod_{i=1}^{n} p(\boldsymbol{X}_{i.}|\boldsymbol{\Omega}, \eta = 1)$$
$$= C\left[\frac{\det(\Omega_0)}{\det(A_1)}\right]^{\frac{n}{2}} \det(\Omega_c)^n \exp\left\{-\frac{n}{2}\sum_{t=1}^{2} \text{tr}(\hat{\Sigma}_{X(t,t)}\Omega_c)\right\}$$
$$\times \exp\left\{\frac{n}{2}\sum_{t_1,t_2=1}^{2} \text{tr}(\Omega_c\hat{\Sigma}_{X(t_1,t_2)}\Omega_c A_1^{-1})\right\},$$
$$b = \prod_{i=1}^{n} p(\boldsymbol{X}_{i.}|\boldsymbol{\Omega}, \eta = 0)$$
$$= C\left[\frac{\det(\Omega_0)}{\det(A_0)}\right]^{\frac{n}{2}} \prod_{t=1}^{2} \det(\Omega_t)^{\frac{n}{2}} \exp\left\{-\frac{n}{2}\text{tr}(\hat{\Sigma}_{X(t,t)}\Omega_t)\right\}$$
$$\times \exp\left\{\frac{n}{2}\sum_{t_1,t_2=1}^{2} \text{tr}(\Omega_{t_1}\hat{\Sigma}_{X(t_1,t_2)}\Omega_{t_2} A_0^{-1})\right\}, \tag{8}$$

where $C = (2\pi)^{-npT/2}$. This approximation ensures that the estimate of η approaches to its underlying truth with probability 1 under some regulatory conditions, as seen from the following Proposition of $p^\lambda(\eta = 1|(\cdot), \boldsymbol{X})$.

Proposition. Assume 1) sparse networks with $|E|, |E_1|$, and $|E_2|$ in the order of $O(p)$ and 2) $\log n/p \to \infty$ as $n, p \to \infty$. Then $\lim_{n\to\infty} p^\lambda(\eta = 1|(\cdot), \boldsymbol{X}) = 1$ if the underlying $\eta = 1$, and $\lim_{n\to\infty} p^\lambda(\eta = 1|(\cdot), \boldsymbol{X}) = 0$ if the underlying $\eta = 0$.

The proof of the Proposition is included in Appendix B. With $p^\lambda(\eta = 1|(\cdot), \boldsymbol{X})$ defined as in (7), the Proposition ensures the detection of the underlying truth of η

with probability. In addition, under a finite sample situation, the definition of $\lambda(n)$ adds to the preference on parsimonious networks and penalizes large numbers of edges (Appendix B). Parsimonious networks have the potential to avoid false positive edges. In medical studies, such a property will benefit biomarker detections. In the context of network differentiation, the concept of parsimony also includes the situation of two networks being identical rather than being differential.

As seen in (8), the conditional posterior distribution of η involves all the precision matrices, Ω_0 and $\Omega^{(t)}$. Such an involvement requires demanding computing power when p is large, especially in the calculation of the inverse of A_0 and A_1. In the next section, Section 3.3, we propose an alternative approach to reduce the computing burden. A block-wise sampling method suggested by Wang et al. (2015) is applied to further improve the computing speed.

Turning back to conditional posterior distributions, we now move to parameters in $\Omega^{(t)}$,

$$p(\omega_{m_1 m_2}^{(t)}|(\cdot), \boldsymbol{X}) \propto \prod_{i=1}^{n} p(\boldsymbol{X}_{i.}|\boldsymbol{\Omega}, \eta) p(\omega_{m_1 m_2}^{(t)}|\eta, g_{m_1 m_2}^{(t)}),$$

$$p(\omega_{mm}^{(t)}|(\cdot), \boldsymbol{X}) \propto \prod_{i=1}^{n} p(\boldsymbol{X}_{i.}|\boldsymbol{\Omega}, \eta) p(\omega_{mm}^{(t)}|\lambda), \tag{9}$$

subjective to $\Omega^{(t)}$ being positive definite, where $t = 1, 2$, $m = 1, \ldots, p$, $m_1 < m_2$, $m_1, m_2 = 1, \ldots, p$. The conditional posterior distributions of parameters in graph structures $G^{(t)}$,

$$p(g_{m_1 m_2}^{(t)}|(\cdot), \boldsymbol{X}) \propto \prod_{i=1}^{n} p(\boldsymbol{X}_{i.}|\boldsymbol{\Omega}, \eta) p(\omega_{m_1 m_2}^{(t)}|\eta, g_{m_1 m_2}^{(t)}) p(g_{m_1 m_2}^{(t)}|\eta). \tag{10}$$

The conditional posterior distributions of parameters in Ω_0 and G_0 follow similar structures as in (9) and (10), respectively.

As seen from (4) and (10), hyper-parameters ν_0 and ν_1, $\nu_0 < \nu_1$, define a priori the sparsity of a graph. With ν_1 fixed, different values of ν_0 control the exclusion of edges with larger values of ν_0 resulting in a smaller number of edges, and *vice versa*. If the value of ν_0 is too small, an edge between nodes m_1 and m_2 will not be excluded unless extremely strong information on edge exclusion is present in the data leading to a rather small entry of $\omega_{m_1 m_2}^{(t)}$ of $\Omega^{(t)}$. This may result in a graph with a large number of falsely connected edges, causing difficulties to decipher underlying network structures. On the other hand, if we take a relatively large value of ν_0 (relative to ν_1), then we may end up with a graph that is overly spare.

Since G_1 and G_2 are unique graphs after adjusting the shared G_0, it is sensible to assume that G_1 and G_2 are both sparse. We do not have to impose such a stringency on G_0. Under this context, the selection of ν_0 needs to have a potential to control the sparsity of G_1 and G_2. Rather than taking a fixed value of ν_0, we tune ν_0 by restricting the level of sparsity of a graph within a certain range. As noted earlier, for sparse

graphs, the expected number of edges is $O(p)$, based on which the probability of edge-inclusion can be set at $2/(p-1)$ when the expected number being p. This further gives the variance of the number of edges $V^2 = p(p-3)/(p-1)$. In this article, we set the upper bound of the number of edges of G_1 and G_2 at $p + cV$ and the lower bound at $\max(1, p - cV)$ with $c = 10$. In practice, users have the flexibility to set the limits guided by minimum and maximum sparsity. Based on our experience, the initial value of ν_0 needs to be relatively small compared to ν_1. Herein, based on suggestions of Wang et al. (2015) on the selection of ν_0, we take $\nu_0 = 0.02$ as the initial value in both simulations and real world examples. The value of ν_1 is set at $\nu_1 = 4$. For the hyper-parameters ζ_0 and ζ_1 involved in the conditional posterior distributions of parameters in Ω_0 and G_0, if sparsity is not assumed, then we could fix ζ_0 and ζ_1 at $\zeta_0 = 0.02$ and $\zeta_1 = 4$. The algorithm for the whole sampling procedure is included in Appendix C.

3.3 An efficient modeling and sampling approach

As noted in Section 3.2, the method introduced so far for inferring network differentiation status involves computationally burdensome matrix inverse calculations. This section we propose an alternative approach that is computationally efficient. Based on (1),

$$\prod_{i=1}^{n} p(\boldsymbol{X}_{it}^* | \Omega^{(t)}) = (2\pi)^{-\frac{np}{2}} \det(\Omega^{(t)})^{n/2} \exp\left\{ \frac{-n}{2} \operatorname{tr}(\hat{\Sigma}_t \Omega^{(t)}) \right\},$$

$$\prod_{i=1}^{n} p(\boldsymbol{S}_i | \Omega_0) = (2\pi)^{-\frac{np}{2}} \det(\Omega_0)^{n/2} \exp\left\{ \frac{-n}{2} \operatorname{tr}(\hat{\Sigma}_0 \Omega_0) \right\}, \tag{11}$$

where $\hat{\Sigma}_0$ and $\hat{\Sigma}_t$ are the sample covariance matrices of $\boldsymbol{S}_i$ and $\boldsymbol{X}_{it}^*$, respectively, had they been observed. Certainly, both $\boldsymbol{S}_i$ and $\boldsymbol{X}_{it}^*$ are latent. We now propose an estimate of these sample covariance matrices. Here, with some abuse of notation, we still use $\hat{\Sigma}_0$ and $\hat{\Sigma}_t$ to denote the estimated sample covariance matrices.

Since Σ_0 is the covariance matrix shared between the two time points, logically we estimate $\hat{\Sigma}_0$ in (11) with the off-diagonal block of $\hat{\Sigma}_X$, a $pT \times pT$ sample covariance of $\boldsymbol{X}_{i.}$,

$$\hat{\Sigma}_0 = \frac{1}{2n} \sum_{t_1, t_2} \sum_{i=1}^{n} (\boldsymbol{x}_{i,t_1} \boldsymbol{x}_{i,t_2}'),$$

where $t_1, t_2 = 1, 2$ with $t_1 \neq t_2$ and $\boldsymbol{x}_{i,t}, t = 1, 2$ are centered observed data of $\boldsymbol{X}_{it}$. The definition of Σ_X leads to $\Sigma_t = \Sigma_{X,t} - \Sigma_0$ with $\Sigma_{X,t}$ denoting the covariance matrix of X at time t. We then estimate $\hat{\Sigma}_t$ in (11) as $\hat{\Sigma}_t = \hat{\Sigma}_{X,t} - \hat{\Sigma}_0$ with $\hat{\Sigma}_{X,t}$ being the sample covariance of data at time t.

Recall that η denotes the differentiation in networks between two time points determined by $\boldsymbol{X}_{i.}^* = \{X_{i1}^*, X_{i2}^*\}$. Under this context, we formulate the likelihood as $L^*(\Theta | \boldsymbol{X}^*) = \prod_{t=1}^{2} \prod_{i=1}^{n} p(\boldsymbol{X}_{it}^* | \Omega^{(t)}, \eta)$, where $\boldsymbol{X}^* = \{\boldsymbol{X}_{1.}^*, \ldots, \boldsymbol{X}_{n.}^*\}$. Since the distribution of $\boldsymbol{X}_{it}^*$ does not depend on Ω_0, this parameter is excluded from Θ. The prior

distributions of each parameter in Θ are constructed as in Section 3.2 and we do not repeat here.

The joint posterior distribution of Θ in (6) is updated to the following,

$$p(\Theta|\boldsymbol{X}^*) \propto \prod_{t=1}^{2}\prod_{i=1}^{n} p(\boldsymbol{X}_{it}^*|\Omega^{(t)}, \eta)p(\Omega^{(t)}|\eta, \boldsymbol{g^{(t)}})\Big\{p(\boldsymbol{g_1})p(\boldsymbol{g_2})\Big\}^{I(\eta=0)}\Big\{p(\boldsymbol{g_c})\Big\}^{I(\eta=1)}p(\eta).$$

(12)

Note that the joint conditional posterior distribution does not involve any expensive matrix inversions. The conditional posterior distributions of each parameter are derived in the same way as in Section 3.2. In particular, $p(g_{m_1m_2}^{(t)}|(\cdot), \boldsymbol{X}^*)$ is in the same format as in (10) with $\boldsymbol{X}$ replaced by $\boldsymbol{X}^*$. For η, following a similar route of (7) and (8),

$$p(\eta = 1|(\cdot), \boldsymbol{X}^*) \approx \big[1 + \exp\{\log(b^*) - \log(a^*) + \lambda(n)\}\big]^{-1},$$

(13)

with $\lambda(n)$ being the same as in (7) and $a^* = C^* \det(\Omega_c)^n \exp\big\{\frac{-n}{2}\sum_{t=1}^{2}\big[\operatorname{tr}(\hat{\Sigma}_t\Omega_c)\big]\big\}$, $b^* = C^* \prod_{t=1}^{2}\det(\Omega_t)^{n/2}\exp\big\{\frac{-n}{2}\operatorname{tr}(\hat{\Sigma}_t\Omega_t)\big\}$ with $C^* = (2\pi)^{-np}$. The derivation of (13) follows the same way as that for (7) and it shares the same Proposition with (7).

We now present the full conditional posterior distribution of $\Omega^{(t)}$, $t = 1, 2$. From (12),

$$p(\Omega^{(t)}|(\cdot), \boldsymbol{X}^*) \propto \prod_{t=1}^{2}\prod_{i=1}^{n} p(\boldsymbol{X}_{it}^*|\Omega^{(t)}, \eta)p(\Omega^{(t)}|\eta, \boldsymbol{g^{(t)}}).$$

The independence between $\boldsymbol{X}_{i1}^*$ and $\boldsymbol{X}_{i2}^*$ allows us to further improve computing efficiency in the posterior sampling of $\Omega^{(t)}$. Specifically, we apply the block Gibbs sampler proposed by Wang et al. (2015) to sample $\Omega^{(t)}$, $t = 1, 2$. Details of the sampling method are in Wang et al. (2015). Basically, instead of sampling $\Omega^{(t)}$ node-by-node, the block sampler generates Monte Carlo samples by columns, which tremendously reduces computing time. Since this approach relies on an approximation to the latent sample covariance matrix of $\boldsymbol{S}_i$ and that of $\boldsymbol{X}_{it}^*$, we denote this method as a method based on latent-data likelihood or a LDL method. Accordingly, we denote the approach using observed data presented in earlier sections as a method based on manifest-data likelihood or an MDL method. The algorithm for this efficient sample approach is also included in Appendix C.

4 Simulations

4.1 Network structures and settings

We consider the following network structures to simulate a network shared between the two time points (G_0) and networks unique at two time points (G_1 and G_2).

S1. Chain networks at both time points with shared network being a nearest-neighbor network (see the top panel of Figure 1). When two networks are differential, the
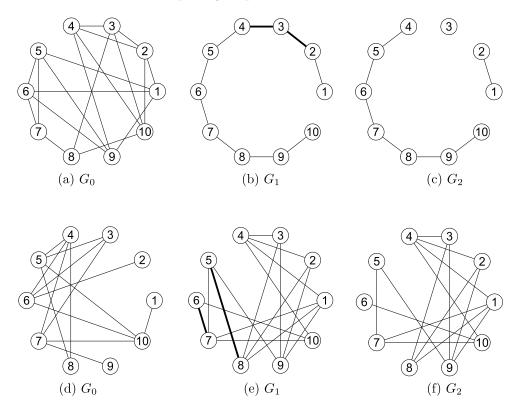
Figure 1: Graph structures for scenario S1 (top panel) and S2 (bottom panel). Figures (a) and (d) are the shared network (G_0), (b) and (e) network unique to time $t = 1$ (G_1), and (c) and (f) network unique to time $t = 2$ (G_2). Thicker lines indicate different connections between G_1 and G_2.

network at each time point is a chain-network but at time $t = 2$ the chain is broken at two consecutive nodes. In this case, the two networks are different in two edges (Figure 1).

The generation of a chain network follows the suggestion of Fan et al. (2009). A covariance matrix is designed following an autoregressive process of order one. In particular, for entry $\{m_1, m_2 = 1, \ldots, p, m_1 \neq m_2\}$, it is defined as $\exp\{-a|s_{m_1} - s_{m_2}|\}$ with $s_{m_1} - s_{m_1-1} \overset{i.i.d}{\sim} \mathrm{Unif}(0.5, 1), m_1 = 2, \ldots, p$, and $s_1 < s_2 < \cdots < s_p$. The diagonal entries are all 1's. Parameter a controls the magnitude of link between nodes, and we set $a = 0.1$. Precision matrices generated under this setting have partial correlations ranged from around -0.4 (low in magnitude) to 0.9 (high in magnitude).

To generate a nearest-neighbor network, the framework given by Li and Gui (2006) is implemented. The structure of a network is generated based on pairwise distances between nodes with each node point generated from a uniform distribution

on a $[0,1] \times [0,1]$ space. For each node, k nearest neighbors are identified using the distances with k pre-specified. In our simulations, we set $k = 4$. Then the algorithm in Li and Gui (2006) is applied to generate entries in the precision matrix for each pair of nodes with an edge. This process yields a precision matrix with all diagonal entries being 1 and off-diagonal elements ranged -0.2 to 0.2 for most entries, indicating overall low partial correlations.

S2. Nearest-neighbor networks at both time points and for the shared network (see the bottom panel of Figure 1). To create differential networks, the network at the first time point (G_1) is a nearest-neighbor network generated as in S1, and randomly taking out two edges from the network produces a network for the second time point (G_2). The shared network G_0 is generated in the same way as in S1.

For each of the two scenarios, we set the number of nodes at $p = 10, 30$ and sample size at $n = 50, 100, 200$, and 500 to demonstrate finite sample properties. In total, 100 Monte Carlo (MC) replicates are generated under each setting to address sampling errors.

In the process of MCMC simulations, two chains and each with 5,000 iterations were applied to multiple randomly selected MC replicates to examine the convergence of the Markov Chains. After observing fast convergence, for each MC replicate, one chain with 500 iterations is adopted, of which 250 iterations are used as the burn-in. The program is written in R and available to users from the first author's website. Four statistics focusing on network comparison and network constructions are used to summarize the results and assess the proposed methods: 1) empirical power of correct detection with respect to network comparison (i.e., $\eta = 1$ [identical] or 0 [differential]), 2) median proportion of true positives for edge connection (TP; sensitivity) in a network, 3) median proportion of true negatives (TN; specificity) of a network, and 4) median proportion of correct connections (CC) of edges. A correct connection between two nodes is defined as an observed connection status between the two nodes is consistent with the underlying truth. A proportion of correct connections combines information on sensitivities and specificities. For all the statistics except for power, we also provide 95% empirical credible intervals. We evaluate the proposed method based on these statistics on the choices of number of nodes, sample sizes and different network structures noted earlier in this section.

4.2 Performance of the two proposed methods

Results under scenario S1 (Table 1) We start from graphs with a smaller number of nodes, $p = 10$. Under scenario S1, when the two networks are truly identical, using the approach based on manifest-data likelihood (the MDL method; Table 1a) provides a power in general comparable to or higher than the power via the method based on latent-data likelihood (the LDL method; Table 1b). For both approaches, the power to detect the underlying truth of identical networks is low when the sample size is small, but quickly increases as the sample size increases.

As for proportions of TP, TN, and CC, their values from the MDL approach are overall higher than the statistics when the LDL method is applied. The statistics from

(a) Results from the MDL approach

n	EP (%)	TP (95% EI)	TN (95% EI)	CC (95% EI)

n	EP	TP	TN	CC
	(%)	(95% EI)	(95% EI)	(95% EI)
Underlying truth: identical networks ($p = 10$ nodes, $|E_{X_1}| = |E_{X_2}| = 10$ edges)				
50	32.0	0.444 (0.111, 0.736)	0.833 (0.722, 0.972)	0.756 (0.631, 0.881)
100	97.2	0.667 (0.386, 1.0)	0.806 (0.722, 0.944)	0.80 (0.667, 0.933)
200	99.6	1.0 (0.778, 1.0)	0.889 (0.778, 0.972)	0.911 (0.778, 0.978)
500	99.6	1.0 (0.999, 1.0)	0.917 (0.833, 0.972)	0.933 (0.867, 0.978)
Underlying truth: differential networks ($p = 10$ nodes, $|E_{X_1}| = 9, |E_{X_2}| = 7$ edges)				
50	99.0	X_1: 0.667 (0.386, 0.889)	0.806 (0.694, 0.903)	0.756 (0.644, 0.901)
		X_2: 0.286 (0.0, 0.714)	0.921 (0.816, 0.988)	0.822 (0.733, 0.911)
100	99.9	X_1: 0.889 (0.556, 1.0)	0.806 (0.694, 0.903)	0.822 (0.689, 0.923)
		X_2: 0.571 (0.286, 1.0)	0.895 (0.816, 0.974)	0.844 (0.756, 0.945)
200	99.9	X_1: 1.0 (0.778, 1.0)	0.806 (0.722, 0.889)	0.822 (0.733, 0.911)
		X_2: 0.999 (0.593, 1.0)	0.868 (0.789, 0.947)	0.867 (0.778, 0.956)
500	99.9	X_1: 1.0 (0.999, 1.0)	0.778 (0.639, 0.944)	0.822 (0.711, 0.956)
		X_2: 0.996 (0.998, 1.0)	0.868 (0.776, 0.935)	0.889 (0.811, 0.945)

(b) Results from the LDL approach

n	EP	TP	TN	CC
	(%)	(95% EI)	(95% EI)	(95% EI)
Underlying truth: identical networks ($p = 10$ nodes, $|E_{X_1}| = |E_{X_2}| = 10$ edges)				
50	48.8	0.444 (0.222, 0.889)	0.806 (0.639, 0.889)	0.733 (0.600, 0.856)
100	71.8	0.667 (0.444, 0.889)	0.694 (0.541, 0.806)	0.689 (0.556, 0.800)
200	94.2	0.889 (0.667, 1.0)	0.639 (0.528, 0.722)	0.689 (0.556, 0.778)
500	100	1 (0.889, 1.0)	0.556 (0.458, 0.639)	0.622 (0.556, 0.701)
Underlying truth: differential networks ($p = 10$ nodes, $|E_{X_1}| = 9, |E_{X_2}| = 7$ edges)				
50	99.9	X_1: 0.222 (0.053, 0.556)	0.847 (0.750, 0.944)	0.733 (0.633, 0.822)
		X_2: 0.143 (0.00, 0.429)	0.947 (0.881, 1.0)	0.822 (0.756, 0.889)
100	99.9	X_1: 0.444 (0.222, 0.778)	0.778 (0.667, 0.861)	0.711 (0.588, 0.844)
		X_2: 0.286 (0.00, 0.571)	0.921 (0.842, 0.974)	0.822 (0.756, 0.901)
200	99.9	X_1: 0.722 (0.444, 0.947)	0.750 (0.652, 0.833)	0.756 (0.611, 0.844)
		X_2: 0.571 (0.211, 0.857)	0.895 (0.763, 0.947)	0.844 (0.711, 0.911)
500	99.9	X_1: 0.889 (0.667, 1.0)	0.667 (0.556, 0.807)	0.711 (0.600, 0.822)
		X_2: 0.714 (0.429, 1.0)	0.816 (0.749, 0.895)	0.822 (0.744, 0.889)

Table 1: Scenario S1. Summary statistics for detecting differential networks, including empirical power (EP) of correct detection on network differentiation, and proportions of true positives for edge connections (TP), true negatives (TN), and correct connections (CC) across 100 MC replicates along with 95% empirical intervals (EI). $|E_{X_1}|$ and $|E_{X_2}|$ denote the number of edges of the underlying networks at time 1 and 2, respectively.

the later approach, especially for the proportion of TP when underlying networks are differential, are low when sample sizes are smaller (i.e., $n = 100$ and lower), but all become acceptable when the sample size is larger. The computing time of the MDL

method is substantially longer than the LDL method. Considering both computing speed and the statistics assessing the performance of the methods, the LDL approach is promising under S1. It has a potential to efficiently work with a large number of nodes and provide satisfactory statistic inferences when sample sizes are relatively large.

With a larger number of nodes, $p = 30$, under scenario S1, the number of edges also increases. Patterns of the statistics with $p = 30$ are comparable to those in Table 1 (Appendix D). Overall, the MDL method handles larger nodes better, but when sample size is large, all the statistics from the LDL approach are significantly improved and closer to results from the MDL method.

Under this scenario, we further assessed the inferences and computing efficiency of the two approaches dealing with even larger numbers of nodes ($p = 50, 70,$ and 90) and results are in Appendix E. Basically, compared to the LDL approach, the MDL method offers better inferences at the expense of high computing cost, especially when $p = 90$.

Results under scenario S2   The overall patterns of the statistics are consistent between different numbers of nodes, although with $p = 30$ the statistics are lower, especially when the sample size is small. In this section, we focus on results from $p = 10$ and results with $p = 30$ are included in Appendix E.

With $p = 10$, when the underlying true network is temporally stable, with a small sample size (e.g., $n = 50$), implementing the MDL method gives a lower power to detect the truth than if the LDL approach is applied. This is potentially due to a larger number of parameters to estimate with the MDL method compared to LDL. It is promising to observe that the MDL approach quickly picks up the power as the sample size increases (Figure 2). When two networks are truly differential, the two approaches both give close to perfect power to detect the underlying truth.
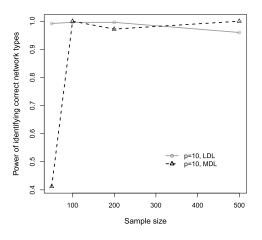

Figure 2: Power for detecting the truth when two networks are identical.

With most partial correlations between -0.2 and 0.2, networks constructed following scenario S2 overall have weak connections between nodes. This is reflected by relatively

low proportions of true positives when sample sizes are small, but as the sample size gets larger, proportion of true positives increases, regardless of the underlying truth about network differentiation status (identical or differential) (Figure 3). A similar pattern is observed for the proportions of correct connections. Proportion of true negatives shows a different pattern; overall, it slightly decreases as the sample size increases. This pattern is also shown in the estimated powers of detecting two identical networks (Figure 2). All these are likely due to the inclusion of edges that are indirectly connected to a node under investigation, a phenomenon discussed in Wasserman and Roeder (2009).

Under scenario S2, although the overall trends of TP, TN, and CC between the two approaches are comparable, with the implementation of MDL, all the statistics are generally better than those with LDL applied, regardless of sample sizes and numbers of nodes (Figure 3 and Appendix E). On the other hand, comparing the statistics from data generated under scenario S2 with those under S1, overall the proportions of TP are lower than those under S1, which is likely due to different sparsity levels between the two types of networks, the same pattern as seen in direct networks (Zhang et al., 2021). Networks generated under S2 are nearest-neighbor networks, which in our simulations have a larger number of edges compared to chain graphs generated under S1 (Figure 1).
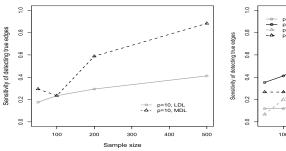
4.3 Inference of G_0

As noted earlier, although the computing burden of the MDL method is heavier than the LDL method, the MDL approach is awarded by its high power in detecting the true status of differentiation along with generally higher statistics in network identifications. Another strength of the MDL method is its ability to infer G_0, which cannot be carried out by the LDL approach. Although not the main focus of our work, inference of G_0 in practice is beneficial to understand underlying network invariant to time.
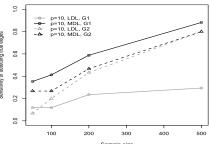
To demonstrate, we use 100 MC replicates generated under scenario S1 with $p = 10$. The underlying truth on the number of edges of G_0 is 19. Graph G_0 is constructed via nearest-neighbors and thus G_0 is expected to have more edges associated with each node than the number of edges in chain graphs of G_1 and G_2 under this scenario. We apply the MDL method to the 100 MC replicates and record TP, TN, and CC statistics.

The patterns of the statistics on network constructions follow our expectation of finite sample properties (Figure 4), i.e., as the sample sizes increases, the statistics are overall all improved. It is noted that proportions of true positive connections are low when the sample sizes are smaller, e.g., $n = 50$, which is expected due to the relatively large number of parameters to be inferred compared to the sample size.
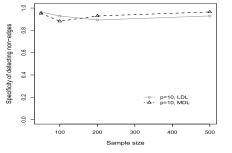
4.4 Comparison with existing approaches

Competing methods Each of the two proposed approaches includes two parts, network construction and comparison. For network constructions, we compare our approach with Xie et al. (2016), since the graphical EM algorithm for shared and specific network estimations is developed based on the same model structure as in our approaches. The graphical LASSO penalty is utilized in Xie et al. (2016) to construct graphs and the
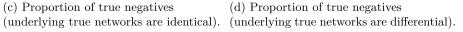
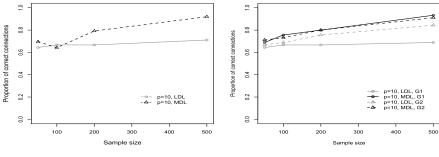(a) Proportion of true positives
(underlying true networks are identical).

(b) Proportion of true positives
(underlying true networks are differential).

(c) Proportion of true negatives
(underlying true networks are identical).

(d) Proportion of true negatives
(underlying true networks are differential).

(e) Proportion of correct connections
(underlying true networks are identical).

(f) Proportion of correct connections
(underlying true networks are differential).

Figure 3: Scenario S2. Proportions of true positives, true negatives, and correct connections, when true underlying networks are identical ((a), (c), (e); 15 edges) or differential ((b), (d), (f); G_1 15 edges and G_2 13 edges).
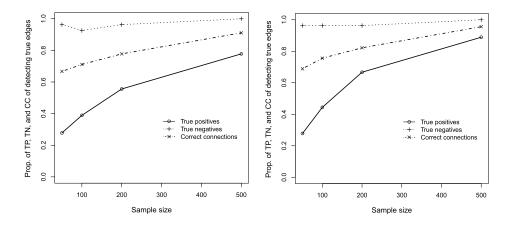
Figure 4: Assessment of inferences of G_0 via the MDL method. Patterns of proportions of true positives (TP), true negatives (TN), and correct connections (CC) (Left: underling true networks are identical; Right: underlying true networks are differential).

tuning parameters in the penalty are selected using the extended Bayesian Information Criterion suggested in that article. For network comparisons, two competing approaches are included with one focusing on comparing networks constructed based on precision matrix (Stäedler and Dondelinger, 2020), implemented in the R package *nethet*, and the other on comparison between the two covariance matrices (Cai et al., 2013), Ω_1^{-1} and Ω_2^{-1}, available in an R package *HDtest*. These two approaches are applied assuming $\boldsymbol{X}_{i1}$ and $\boldsymbol{X}_{i2}$ independent. We focus on the power of detecting the underlying differentiation status of the two networks, and compare the testing results with those from the proposed approach. For the purpose of comparison, we use data simulated under scenario S1 with $p = 10$ and a sample size of 100.

Results of network constructions   Since the focus of this comparison is on network constructions, we use data with underlying networks being differential and assess the quality of network constructions for shared networks as well as time-specific networks. Overall, the MDL performs the best (Figure 5); especially for G_0, MDL clearly outperforms Xie et al. (2016). For time-specific graphs (G_1 and G_2), the method of Xie et al. (2016) tends to choose more edges and thus has higher proportions of TP and lower proportions of TN than those from MDL and LDL. Thus, it has a tendency to produce more false positive edges, compared to the other two approaches. Although LDL offers the lowest statistics in terms of TP and CC, high proportions of TN from this approach indicate that LDL is likely to be more suitable for studies aiming to identify edges with low probability of false positives.

Results of network comparison   We apply the aforementioned two competing methods (Stäedler and Dondelinger, 2020; Cai et al., 2013) to the data generated under S1 with $p = 10, n = 100$. As noted in Table 1, with the MDL approach, the power to
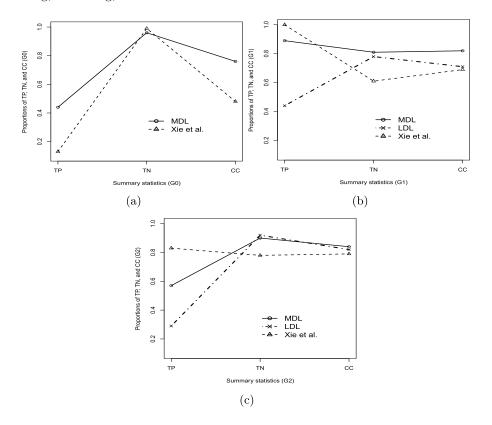
(a)

(b)

(c)

Figure 5: Comparison with results from Xie et al. (2016) on network constructions. Differential networks generated under Scenario S1 ($p = 10$ nodes) are included in the comparison. Proportions of true positives for edge connections (TP), true negatives (TN), and correct connections (CC) across 100 MC replicates for (a) G_0 (no results from LDL is included due to the inestimability of G_0 from LDL), (b) G_1, and (c) G_2. Underlying truth: $|E_{X_1}| = 9, |E_{X_2}| = 7$ edges with $|E_{X_1}|$ and $|E_{X_2}|$ denote the number of edges of the underlying networks at time 1 and 2, respectively.

detect two networks being identical is estimated as 97.2% and the power is 99.9% to identify two networks being truly differential. The LDL approach produces a lower power to detect the truth when two network are truly identical. Both competing methods have a perfect power, higher than the proposed methods, to detect the truth when the two underlying networks are identical. However, when the two networks are truly differential, both approaches lose their power substantially to detect the underlying truth. For the approach by Stäedler and Dondelinger (2020), the estimated power to detect the underlying truth is 23%, and for the method by Cai et al. (2013), the power is 48%. The plummet of the power is expected, since both approaches are designed for independent networks and the dependence between the two time points cannot be incorporated into the process of comparison.

4.5 Detection of potential driving edges/nodes

The utilization of MCMC simulations offers an opportunity to identify edges that potentially drive network differentiation (noted as driving edges). The probability that an edge is a driving edge is estimated using the proportion that an edge is uniquely selected at one time point across a certain number of iterations that are low in autocorrelations. To demonstrate, we again utilize 100 MC replicates generated under scenario S1 with $p = 10$. The cutoff proportion for an edge being a driving edge is set at 0.5. As seen in Figure 1 (b) and (c), the true driving edges are ②—③ and ③—④. Based on findings from the 100 MC replicates, at least one driving edge is detected regardless of the sample size (Table 2). Other detected but pseudo driving edges are all formed by neighbors of the nodes in the true driving edges. These pseudo driving edges are potentially due to indirect connections, but tend to diminish when sample size is large; for instance, when $n = 500$, pseudo driving edges are limited to edges connected among nodes 1 to 4.

Sample size (n)	Detected driving edges (estimated probability)
50	①—② (0.55), **②—③** (0.68)
100	①—② (0.56), **②—③ (0.81)**, **③—④ (0.60)**, ④—⑤ (0.77)
200	①—② (0.63), **②—③ (0.98)**, **③—④ (0.66)**, ④—⑤ (0.72)
500	①—③ (0.88), **②—③ (0.86)**, **③—④ (0.75)**, ②—④ (0.58)

Table 2: Detected driving edges under S1. The probabilities are inferred based on 100 MC replicates.

5 Assessment of epigenetic network differentiation over time

5.1 The data and plan for assessment

To demonstrate the proposed methods, we utilize an epigenetic data set, in particular, DNA methylation at 22 CpGs (Table 3) measured at ages 10, 18 and 26 years, in a population-based birth cohort established on the Isle of Wight in the United Kingdom (Arshad et al., 2018). Each of these 22 CpG sites has shown to be associated with exposure to tobacco smoking in uterus (Joubert et al., 2016), indicating the potential of an underlying epigenetic network among these CpGs, due to biological interaction and dependence within and between genes. Note that at age 10 years, although some girls have started puberty, most children have not experienced any significant pubertal changes, while at age 18 most children have gone through the period of adolescence. We apply the methods to these 22 CpGs at two pairs of ages, 10 and 18 years and 18 and 26 years, as an attempt to answer a real world question, i.e., does the period of adolescence impact the networks of CpGs? It is worth pointing out that the focus here is

Chr	Gene	CpG	CpG Index
1	*GFI1*	cg06338710	1
		cg09662411	2
		cg09935388	3
		cg10399789	4
		cg12876356	5
		cg14179389	6
		cg18146737	7
		cg18316974	8
5	*AHRR*	cg05575921	9
6	*HLA-DPB2*	cg11715943	10
7	*CNTNAP2*	cg25949550	11
	ENSG00000225718	cg04598670	12
	MYO1G	cg04180046	13
		cg12803068	14
		cg19089201	15
8	*EXT1*	cg03346806	16
14	*TTC7B*	cg18655025	17
15	*CYP1A1*	cg05549655	18
		cg11924019	19
		cg18092474	20
		cg22549041	21
21	*RUNX1*	cg12477880	22

Table 3: Relevant information of the 22 CpG sites. "Chr" denotes the chromosome location of a CpG site.

on an examination of temporal differentiation in networks in a general population rather than difference in networks between different exposure statuses; the later comparison is expected to be a straightforward extension of (1) with the inclusion of a component reflecting treatment effect analogous to linear mixed modeling.

We apply both the MDL and LDL approaches to draw inferences. In total, 5,000 iterations are run after the parameter ν_0 is tuned by controlling the number of edges with a lower bound of 10 edges and an upper bound of 31 edges, defined in Section 3.2. Samples from the last 2,500 iterations are used to draw posterior inferences.

5.2   Results

We first assess network differentiation between ages 10 and 18 years ($n = 325$). Results from the two approaches (MDL and LDL) have certain noticeable agreements. The probability that the network at age 18 years is different from that at age 10 years is estimated as 0.999 by both methods, consistent with simulation findings when underlying networks are differential. As for the inferred graphs (Figure 6), at age 18 years, both approaches detected links among nodes 13, 14, and 15 and between nodes 18 and 20. CpGs at nodes 13, 14, and 15 are all on *MYO1G* gene, and nodes 18 and 20 are

CpGs both on gene *CYP1A1*, which supports the identified joint activities among these CpGs.
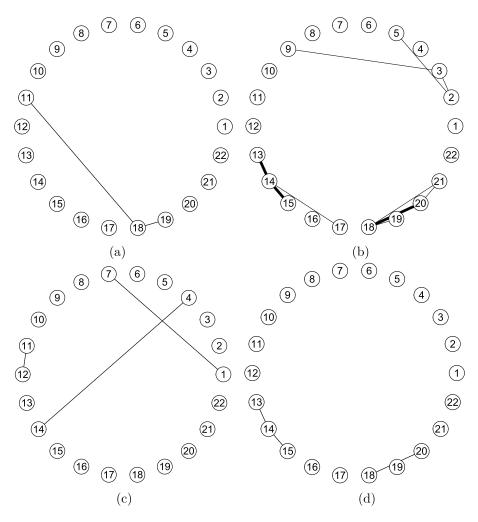

Figure 6: Estimated networks unique to each age ($\hat{G}_1$ and $\hat{G}_2$) based on DNA methylation data at $p = 22$ CpG sites. Graphs in the top panel are inferred based on the manifest-data likelihood (MDL) method and the lower panel is based on the latent-data likelihood (LDL) method. Figures (a) and (c) are for age 10 years, and (b) and (d) are for age 18 years. The thick lines in (b) indicate edges common between the networks identified by the MDL and LDL methods at age 18 years.

At age 18 years, the MDL method identified a network with more edges (10 edges) compared to the LDL approach (3 edges). Based on simulation results, the MDL method tends to have a higher sensitivity, which indicates that the additional edges identified by this approach is likely to be informative, e.g., the subnetwork formed by nodes 18 to 21 with CpGs all on gene *CYP1A1*. In addition, the MDL method identified a unique sub-network formed by nodes 2, 3, 5, and 9. CpGs on nodes 2, 3, and 5 are all on gene *GFI1*, and node 9 has cg05575921 on gene *AHRR*. CpG site cg05575921 has been suggested by multiple studies that it can be used as a marker for smoking exposure, e.g., Bojesen et al. (2017). Note that this node is not connected to any of the remaining nodes at age 10 from both methods. No children in the cohort actively smoked at age 10 years but by age 18 years, some children had smoked for a certain amount of time, which may explain the appearance of this node in the unique network detected at age 18 years by the MDL approach. Further assessment on driving edges (results for "10 and 18 years" in Table 4) indicates that, overall, the subnetwork with node 9 included is not strong enough to cause differentiation in the networks between the two ages, while the subnetwork formed by nodes 18 to 21 plays a more important role in network differentiation. These are indicated by the estimated probabilities for an edge being a driving edge.

Ages	Detected driving edges (Estimated probability)
10 and 18 years	⑤—② (0.55), ⑪—⑱ (0.91), **⑬**—**⑭** (0.67), ⑭—⑮ (0.88)
	⑭—⑰ (0.53), ⑱—⑳ (0.84), ⑱—㉑ (0.52), ⑳—㉑ (0.82)
18 and 26 years	**⑬**—**⑭** (0.57), ⑲—⑳ (0.75), ⑲—㉑ (0.59)

Table 4: Potential driving edges for network differentiation between ages 10 and 18 and between ages 18 and 26 years.

Regardless of age, children are likely to be exposed to passive smoking, and such impact is expected to be captured by the shared network G_0 but not $G^{(t)}$. The G_0 estimated using the MDL approach has in total 42 edges (Figure 7(a)), reflecting the interconnections among CpGs invariant to age. Since our approach for detecting network differentiation takes into account edge connections as well as strength of connections, the detected age-18-year subnetworks noted earlier (one by 13, 14, and 15, and the other in the cluster of nodes 18 to 21, which are both in the inferred G_0) indicate that at these nodes the strength of connection is likely to be different between ages 10 and 18. Finally, at age 10 years, both methods identified a small number of edges but without any agreement. This may reflect a strong uncertainty of unique network at age 10 compared to the shared network G_0.

We further compared network differentiation between ages 18 and 26 years ($n = 244$). Again, both methods estimated the probability of differentiation close to 1 and our analysis on driving edges indicates that such a differentiation is most likely to be caused by the connection between nodes 19 and 20 (probability of 0.75 as a driving
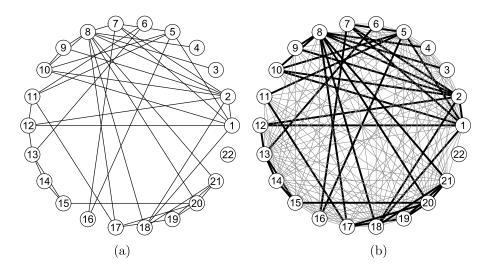
Figure 7: Estimated underlying shared networks ($\hat{G}_0$) based on DNA methylation data at $p = 22$ CpG sites, (a) between ages 10 and 18 years, and (b) between ages 18 and 26 years. Edges in (a) also shown in (b) are indicated by thicker lines.

edge). In terms of the number of edges showing differentiation between 18 and 26 years (Appendix G), results from the LDL approach is overall comparable to that for ages 10 and 18 years. Notably, the MDL approach found a much smaller number of edges showing differentiation between ages 18 and 26 years (5 edges at age 26 years and one edge at age 18 years) compared to ages 10 and 18 years (10 and 2 edges, respectively). Consequently, the number of driving edges for this period (18 to 26 years) is also smaller. It is noteworthy that edge ⑬—⑭ is identified as a driving edge at both periods (10 to 18 years and 18 to 26 years), indicating the strength of connection between these two nodes is likely to change over time. Furthermore, the estimated shared network has 154 edges, which covered most of the 42 edges of the inferred network shared between ages 10 and 18 years (only 4 of the 42 not included; Figure 7(b)). All this seems to indicate that adolescence transition (from 10 to 18 years) may substantially affect the underlying interconnection among epigenetic factors, such influence tends to be stable as children become adults, and connections among certain CpG sites appear during adolescence and become stable after adolescence.

6   Discussion

We proposed a Bayesian approach to compare two dependent networks potentially due to temporal changes or impact from other unknown factors over time. To ensure efficient convergence, we developed a penalty-incorporated conditional posterior probability for η, the indicator of two network being identical or differential, and theoretically investigated the property of this posterior probability on its convergence to the underlying

status of network differentiation. In posterior computing, in addition to the standard sampling approach constructed based on the joint likelihood of manifest data (the MDL method), we defined a joint likelihood utilizing latent data and implemented block sampling to substantially improve computing efficiency (the LDL method). Simulations indicated the efficacy of the proposed approaches; the MDL method overall outperforms the LDL approach, especially in the detection of underlying edges, but at the expense of computing cost. Based on simulations, the MDL approach is able to handle relatively small numbers of nodes (e.g., < 90) without a significant computing burden on a personal computer (e.g., with 3.70 GHz processor and 32.0 GB memory). The LDL approach is computationally scalable and has the ability to deal with a large number of nodes. The relatively low true positive proportions and high true negative proportions indicate that the LDL method is likely to benefit studies aiming to identify strong edges unique to each time point. Findings from our real data applications using DNA methylation data are consistent with simulation results. In particular, our results indicate that epigenetic networks are likely to change over time and the period of adolescence tends to contribute substantially to network differentiation.

The proposed methods are not limited to DNA methylation data and can be directly applied to other data, e.g., gene expression measured at different time points or in different tissues from the same subject. In addition, the proposed approaches can be applied to perform pair-wise comparisons between multiple networks (> 2 networks), in which case adjustment of multiple testing needs to be considered. However, before pair-wise comparisons, it is desirable to examine overall status of network differentiation. Under this context, there is a need to extend the proposed techniques of two dependent network comparisons to fit the situation of multiple dependent networks.

Supplementary Material

Appendix (DOI: 10.1214/22-BA1337SUPP; .pdf).

References

Arshad, S. H., Holloway, J. W., Karmaus, W., Zhang, H., Ewart, S., Mansfield, L., Matthews, S., Hodgekiss, C., Roberts, G., and Kurukulaaratchy, R. (2018). "Cohort profile: The Isle of Wight whole population birth cohort (IOWBC)." *International Journal of Epidemiology*, 47: 1043–1044.   1358

Bashir, A., Carvalho, C. M., Hahn, P. R., Jones, M. B., et al. (2019). "Post-processing posteriors over precision matrices to produce sparse graph estimates." *Bayesian Analysis*, 14(4): 1075–1090. MR4044846. doi: https://doi.org/10.1214/18-BA1139. 1342

Bojesen, S. E., Timpson, N., Relton, C., Smith, G. D., and Nordestgaard, B. G. (2017). "AHRR (cg05575921) hypomethylation marks smoking behaviour, morbidity and mortality." *Thorax*, 72(7): 646–653.   1361

Cai, T., Liu, W., and Xia, Y. (2013). "Two-sample covariance matrix testing and support recovery in high-dimensional and sparse settings." *Journal of the American Statistical Association*, 108(501): 265–277. MR3174618. doi: https://doi.org/10.1080/01621459.2012.758041. 1342, 1356, 1357

Chang, J., Zhou, W., Zhou, W.-X., and Wang, L. (2017). "Comparing large covariance matrices under weak conditions on the dependence structure and its application to gene clustering." *Biometrics*, 73(1): 31–41. MR3632349. doi: https://doi.org/10.1111/biom.12552. 1342

Fan, J., Feng, Y., and Wu, Y. (2009). "Network exploration via the adaptive LASSO and SCAD penalties." *The Annals of Applied Statistics*, 3(2): 521. MR2750671. doi: https://doi.org/10.1214/08-AOAS215. 1350

Friedman, J., Hastie, T., and Tibshirani, R. (2008). "Sparse inverse covariance estimation with the graphical lasso." *Biostatistics*, 9(3): 432–441. 1342

George, E. I. and McCulloch, R. E. (1993). "Variable selection via Gibbs sampling." *Journal of the American Statistical Association*, 88: 881–889. 1345

Gill, R., Datta, S., and Datta, S. (2010). "A statistical framework for differential network analysis from microarray data." *BMC Bioinformatics*, 11(1): 1. 1342

He, H., Cao, S., Zhang, J.-g., Shen, H., Wang, Y.-P., and Deng, H.-w. (2019). "A statistical test for differential network analysis based on inference of Gaussian graphical model." *Scientific Reports*, 9(1): 1–8. 1342

Hotta, K., Kitamoto, A., Kitamoto, T., Ogawa, Y., Honda, Y., Kessoku, T., Yoneda, M., Imajo, K., Tomeno, W., Saito, S., et al. (2018). "Identification of differentially methylated region (DMR) networks associated with progression of nonalcoholic fatty liver disease." *Scientific Reports*, 8(1): 1–11. 1341

Ishwaran, H. and Rao, J. S. (2005). "Spike and slab variable selection: frequentist and Bayesian strategies." *The Annals of Statistics*, 33: 730–773. MR2163158. doi: https://doi.org/10.1214/009053604000001147. 1345

Jacob, L., Neuvial, P., and Dudoit, S. (2012). "More power via graph-structured tests for differential expression of gene networks." *The Annals of Applied Statistics*, 561–600. MR2976483. doi: https://doi.org/10.1214/11-AOAS528. 1342

Jones, B., Carvalho, C., Dobra, A., Hans, C., Carter, C., and West, M. (2005). "Experiments in stochastic computation for high-dimensional graphical models." *Statistical Science*, 388–400. MR2210226. doi: https://doi.org/10.1214/088342305000000304. 1345

Joubert, B. R., Felix, J. F., Yousefi, P., Bakulski, K. M., Just, A. C., Breton, C., Reese, S. E., Markunas, C. A., Richmond, R. C., Xu, C.-J., et al. (2016). "DNA methylation in newborns and maternal smoking in pregnancy: genome-wide consortium meta-analysis." *The American Journal of Human Genetics*, 98(4): 680–696. 1358

Li, F.-q. and Zhang, X.-s. (2017). "Bayesian Lasso with neighborhood regression method for Gaussian graphical model." *Acta Mathematicae Applicatae Sinica, English Series*,

33(2): 485–496. MR3647001. doi: https://doi.org/10.1007/s10255-017-0676-z. 1342

Li, H. and Gui, J. (2006). "Gradient directed regularization for sparse Gaussian concentration graphs, with applications to inference of genetic networks." *Biostatistics*, 7(2): 302–317.   1350, 1351

Mazumder, R. and Hastie, T. (2012). "The graphical lasso: New insights and alternatives." *Electronic Journal of Statistics*, 6: 2125. MR3020259. doi: https://doi.org/10.1214/12-EJS740.   1342

Mitchell, T. J. and Beauchamp, J. J. (1988). "Bayesian variable selection in linear regression." *Journal of the American Statistical Association*, 83: 1023–1032. MR0997578. 1345

Ni, Y., Baladandayuthapani, V., Vannucci, M., and Stingo, F. C. (2021). "Bayesian graphical models for modern biological applications." *Statistical Methods & Applications*, 1–29. MR4426829. doi: https://doi.org/10.1007/s10260-021-00572-8. 1342

Shaddox, E., Peterson, C. B., Stingo, F. C., Hanania, N. A., Cruickshank-Quinn, C., Kechris, K., Bowler, R., and Vannucci, M. (2020). "Bayesian inference of networks across multiple sample groups and data types." *Biostatistics*, 21(3): 561–576. MR4120340. doi: https://doi.org/10.1093/biostatistics/kxy078.   1342

Städler, N., Dondelinger, F., Hill, S. M., Akbani, R., Lu, Y., Mills, G. B., and Mukherjee, S. (2017). "Molecular heterogeneity at the network level: High-dimensional testing, clustering and a TCGA case study." *Bioinformatics*, 33(18): 2890–2896.   1342

Stäedler, N. and Dondelinger, F. (2020). *nethet: A Bioconductor package for high-dimensional exploration of biological network heterogeneity*. R package version 1.20.1. 1356, 1357

Wang, H. et al. (2015). "Scaling it up: Stochastic search structure learning in graphical models." *Bayesian Analysis*, 10(2): 351–377. MR3420886. doi: https://doi.org/10.1214/14-BA916.   1342, 1345, 1347, 1348, 1349

Wasserman, L. and Roeder, K. (2009). "High dimensional variable selection." *Annals of Statistics*, 37(5A): 2178. MR2543689. doi: https://doi.org/10.1214/08-AOS646. 1354

Xia, Y., Cai, T., and Cai, T. T. (2015). "Testing differential networks with applications to the detection of gene-gene interactions." *Biometrika*, asu074. MR3371002. doi: https://doi.org/10.1093/biomet/asu074.   1342

Xie, Y., Liu, Y., and Valdar, W. (2016). "Joint estimation of multiple dependent Gaussian graphical models with applications to mouse genomics." *Biometrika*, 103(3): 493–511. MR3551780. doi: https://doi.org/10.1093/biomet/asw035.   1343, 1354, 1356, 1357

Zhang, H., Huang, X., Han, S., Rezwan, F. I., Karmaus, W., Arshad, H., and Holloway, J. W. (2021). "Gaussian Bayesian network comparisons with graph order-

ing unknown." *Computational Statistics & Data Analysis*, 157: 107156. MR4196481. doi: https://doi.org/10.1016/j.csda.2020.107156.   1354

Zhang, H., Huang, X., and Arshad, H. (2022). "Supplementary Material for "Comparing dependent undirected Gaussian networks"." *Bayesian Analysis*. doi: https://doi.org/10.1214/22-BA1337SUPP.   1346

Zhao, S. D., Cai, T. T., and Li, H. (2014). "Direct estimation of differential networks." *Biometrika*, 101(2): 253–268. MR3215346. doi: https://doi.org/10.1093/biomet/asu009.   1342

Acknowledgments