Journal of Machine Learning Research 24 (2023) 1-35 Submitted 10/20; Revised 11/22; Published 1/23

Bayesian Spiked Laplacian Graphs

Leo L Duan LI.DUAN@UFL.EDU
Department of Statistics, University of Florida

George Michailidis GMICHAIL@UFL.EDU
Department of Statistics, University of Florida

Mingzhou Ding MDING@BME.UFL.EDU
Department of Biomedical Engineering, University of Florida

Editor: Edo Airoldi

Abstract

In network analysis, it is common to work with a collection of graphs that exhibit heterogeneity. For example, neuroimaging data from patient cohorts are increasingly available. A critical analytical task is to identify communities, and graph Laplacian-based methods are routinely used. However, these methods are currently limited to a single network and also do not provide measures of uncertainty on the community assignment. In this work, we first propose a probabilistic network model called the "Spiked Laplacian Graph" that considers an observed network as a transform of the Laplacian and degree matrices of the network generating process, with the Laplacian eigenvalues modeled by a modified spiked structure. This effectively reduces the number of parameters in the eigenvectors, and their sign patterns allow efficient estimation of the underlying community structure. Further, the posterior distribution of the eigenvectors provides uncertainty quantification for the community estimates. Second, we introduce a Bayesian non-parametric approach to address the issue of heterogeneity in a collection of graphs. Theoretical results are established on the posterior consistency of the procedure and provide insights on the trade-off between model resolution and accuracy. We illustrate the performance of the methodology on synthetic data sets, as well as a neuroscience study related to brain activity in working memory.

Keywords: Isoperimetric Constant, Mixed-Effect Eigendecomposition, Normalized Graph Cut, Stiefel Manifold.

1. Introduction

In recent years, there has been a strong interest in modeling network data due to their increased availability in social sciences (Aggarwal, 2011), biology (Minch et al., 2015) and engineering (Zhang et al., 2008). A popular generative model, suitable for social network analysis has been the stochastic block model (Nowicki and Snijders, 2001; Karrer and Newman, 2011) and its variant, the mixed membership stochastic block model (Airoldi et al., 2008). Their popularity stems from the fact that these models tend to produce networks organized in communities; subsets of vertices connected with one another with particular edge densities. For example, edge density may be higher within communities than be-

©2023 Leo L Duan, George Michailidis and Mingzhou Ding.

License: CC-BY 4.0, see https://creativecommons.org/licenses/by/4.0/. Attribution requirements are provided at http://jmlr.org/papers/v24/20-1206.html.

tween communities. A key analytical task is that of community detection and a plethora of algorithms have been proposed in the literature [Leighton and Rao (1999); Khandekar et al. (2009); Arora et al. (2009); Mucha et al. (2010); Fortunato (2010); Papadopoulos et al. (2012); for recent reviews, see Abbe (2017); Javed et al. (2018)]. Further, consistency results when the number of vertices grows to infinity have also been provided for certain community detection algorithms, with spectral clustering being the most prominent among them (Rohe et al., 2011; Amini et al., 2013).

However, when the network is of small to moderate size, such consistency results are not directly applicable, which motivated various Bayesian approaches. Many of them can be viewed as variants of the latent space model (Hoff et al., 2002), wherein the key idea is to assume a latent coordinate for each vertex, and the pairwise interaction of two coordinates (e.g., inner product, distance) determines the probability of whether an edge should form. Such an example is the Bayesian stochastic block model [see, e.g., McDaid et al. (2013); van der Pas and van der Vaart (2018); Geng et al. (2019)] that characterizes the randomness in the community labels. Some other approaches consider edge formation as the outcome of a stochastic mechanism that can lead to a power-law degree distribution (Cai et al., 2016), or to sparse networks (Caron and Fox, 2017) and can aid in link prediction tasks (Williamson, 2016).

In many scientific areas, it is becoming common to have access to a *collection* of networks, that usually exhibit a certain degree of heterogeneity. For example, neuroscientists collect brain signals from EEG/MEG technologies for cohorts of patients that give rise to networks capturing brain activity between regions of interest (ROIs) (Shen et al., 2013). The networks in the collection share common features (e.g., community structure, since they are derived from subjects either responding to the same stimulus in designed experimental studies or having the same disease condition in observational studies), but also exhibit heterogeneity. Analysis of such collections could proceed by applying current approaches to each network and then devising methods for aggregating the results, which could prove challenging, since the possible significant variation from one network to another renders pooling information error-prone (e.g., by assuming a shared latent space). This issue was recognized by Durante et al. (2017) that proposed to use multiple sets of coordinates, modeled by a non-parametric mixture distribution. The latter approach fits better the underlying data, vis-a-vis a naive averaging across multiple networks. Similarly, Mukherjee et al. (2017) proposed an approach to directly cluster the networks, which reduces the heterogeneity for downstream analysis.

Another important factor to consider for multiple networks is the risk of model misspecification. Unfortunately, a fully Bayesian network model is sensitive to this issue, since it needs to impose a parametric distribution on the edge generating mechanism, often using a Bernoulli distribution. In contrast, in real world applications, the available network data are in fact produced by various processing algorithms, such as thresholding the correlation matrix from the multivariate time series (Sojoudi, 2016) (hence not Bernoulli). Although one could trace back the data processing steps and develop a corresponding generative model, often this is impractical due to the complexity and use of heuristics in those steps.

Rather, it is more useful to consider a modeling approach, that is probabilistic, but based on more relaxed assumptions.

The above two factors lead us to consider the graph Laplacian (Chung and Graham, 1997), which lies at the heart of spectral clustering algorithms that use a set of eigenvectors corresponding to the smallest non-trivial eigenvalues. It is a simple transformation of the adjacency matrix, and its smallest non-trivial eigenvalues provide information on the minimum edge loss when partitioning the network into multiple communities. However, spectral clustering-based approaches are primarily algorithmic in nature, involving a multi-stage procedure starting by normalizing the graph Laplacian, followed by a singular value decomposition and selection of the appropriate number of eigenvectors to use (based mostly on empirical inspection) and then finally a post-processing of the eigenvectors through an application of the K-means algorithm (Ng et al., 2002). Further, performance guarantees for such approaches are asymptotic in nature and take the form of high probability error bounds on the number of communities selected and the misclassification error rate (Hein et al., 2007; Von Luxburg et al., 2008; Rohe et al., 2011). Since there is no likelihood function involved, measures of uncertainty for community assignments are difficult to obtain, and further, it becomes challenging to accommodate heterogeneity across networks.

To overcome these challenges, we consider a probabilistic model for the graph Laplacian, leveraging its spectral properties and subsequently introducing a non-parametric Bayes approach on a population of networks/graphs. The crux of the problem is how to parameterize a valid Laplacian matrix by only focusing on a small set of eigenvectors (rank) that captures the underlying community structure. We leverage ideas from the spiked covariance model (Donoho et al., 2018), and adding a new transformation that focuses on the smallest eigenvalues (as opposed to the largest ones in covariance modeling). We then show that the associated eigenvectors contain useful information for a hierarchical partitioning of the graph, which leads to an almost instantaneous estimation of the underlying communities, with no need for post-processing of the results from spectral clustering with iterative algorithms such as K-means. Due to the Bayesian nature of the model, the estimated community labels have a posterior distribution, which quantifies their uncertainty. To the best of our knowledge, existing Bayesian models involving the graph Laplacian mostly use it as a tool to construct a regularizing prior distribution for different types of problems, such as variable selection in regression (Liu et al., 2014), function estimation on a graph (Kirichenko et al., 2017) and covariance specification for Gaussian processes (Dunson et al.). In contrast, we use the Laplacian as a transformation for network data and propose a new likelihood with the goal of carrying out near-optimal community detection, hence our focus is different and novel.

The remainder of this paper is organized as follows: Section 2 introduces the construction of the spiked graph Laplacian, and the non-parametric Bayesian model that accommodates heterogeneity in a collection of graphs; Section 3 introduces the estimation of the communities based on the posterior distribution; Section 4 establishes theoretical properties for the proposed model. Section 5 evaluates the model performance based on synthetic data, while Section 6 illustrates the modeling approach in a data application aiming to charac-

terize the heterogeneity in brain scans in a human working memory study. The software implementation can be found on https://github.com/leoduan/BayesSpikedLaplacian.

2. The Spiked Graph Laplacian Model

Suppose S graphs/networks are observed, each denoted by $G^{(s)} = \{V^{(s)}, E^{(s)}\},\ s = 1, \ldots, S$, with corresponding vertex set $V^{(s)} = \{1, \ldots, n\}$ and edge set $E^{(s)} = \{e^{(s)}_{i,j}\}_{i,j}$. For ease of presentation, we focus on undirected, weighted graphs, whose adjacency matrix is given by $A^{(s)} = \{A^{(s)}_{i,j}\}_{i,j}$ with entries satisfying $A^{(s)}_{i,j} \geq 0$, $A^{(s)}_{j,i} = A^{(s)}_{i,j}$ and $A^{(s)}_{i,i} = 0$. Extension to a binary $A^{(s)}_{i,j}$ is discussed at the end of the paper.

For notational convenience, the graph index (s) is omitted in the sequel. The observed normalized Laplacian is a transformation of the adjacency matrix given by

$$L = D^{-1/2}(D - A)D^{-1/2}, \tag{1}$$

where $D = \text{diag}\{d_i\}_{i=1}^n$ is the observed degree matrix, with $d_i = \sum_{i=1}^n A_{i,j}$.

Theorem 1 *For any adjacency matrix A with $d_i > 0$ for $i = 1, \ldots, n$, the normalized Laplacian $L = D^{-1/2}(D - A)D^{-1/2}$ has all eigenvalues $\lambda_k \in [0, 2]$ (Chung and Graham, 1997). The smallest eigenvalue $\lambda_1 = 0$ (index 1 denotes the smallest one) and $L\vec{d}^{1/2} = \vec{0}$.*

The above theorem shows that D is dependent on L, therefore, to build a probabilistic model on the adjacency matrix, we consider the following factorized model:

$$\Pi(A) = \Pi(L)\Pi(D \mid L).$$

where we use $\Pi(\cdot)$ to denote a density.

To fully characterize the level of dependency of D on L, we make the following observations. (i) If the multiplicity of zero eigenvalue from L is one, then the corresponding unit-norm eigenvector (subject to sign change) must be $\vec{\phi}_1 = (\sqrt{d_i}/\tilde{z}_1)_{i=1\ldots n}$, with $\tilde{z}_1 = \sum_{i=1}^k d_i$. This means that given L, all d_i's are almost known except for a scalar $\tilde{z}_1$. (ii) If the multiplicity of zero eigenvalues of L is $K > 1$, then it means there are K disjoint component subgraphs (Chung and Graham, 1997), we can see that the corresponding eigenvectors are in the form of an $n \times K$ matrix $[\vec{\phi}_1 \ldots \vec{\phi}_K]O$, with $O \in \mathbb{R}^{K\times K}$ a rotation matrix, $\vec{\phi}_k = [\sqrt{d_i}1(c_i = k)/\tilde{z}_k]_{i=1\ldots n}$, $\tilde{z}_k = \sqrt{\sum_{i=1}^n d_i 1(c_i = k)}$ and $1(c_i = k)$ the indicator function representing if the ith vertex is in the kth component subgraph. Since any rotation will not change the 2-norm of each row vector, it is not hard to see that the 2-norm of the each row is still $\sqrt{d_i}1(c_i = k)/\tilde{z}_k$; hence all d_i's are almost known except for $(\tilde{z}_1, \ldots, \tilde{z}_K)$. Summarizing these two cases, we can see that $\Pi(D \mid L)$ is a distribution that only describes the total scale of degrees in each component.

Remark 2 *Based on the above discussion, we can see that once L is known, $\Pi(D \mid L)$ contains very little additional information about the pairwise relationship in A. Therefore, we choose to focus on $\Pi(L)$ from now on — to be exact, if we have parameter $\beta = (\beta_D, \beta_L)$ that enters the likelihood as $\Pi(L; \beta_L)\Pi(D \mid L, \beta_D)$, as our parameter of interest is β_L, we only need to handle $\Pi(L; \beta_L)$ in model specification and posterior inference.*

To specify $\Pi(L;\beta_L)$, we use the following signal-plus-noise matrix model:

$$L = \mu_L + \mathcal{E}, \qquad \mu_L = \sum_{k=1}^{T} \lambda_k \vec{q}_k \vec{q}_k' + \sum_{l=T+1}^{n} \theta \vec{q}_l \vec{q}_l', \tag{2}$$

where $\mathcal{E}$ is a symmetric matrix capturing random variation with $\mathcal{E} = \{e_{i,j}\}_{i,j}, e_{i,j} \sim N(0, \sigma_e^2)$ for $i < j$. The matrix μ_L is symmetric with eigenvalues $\lambda_1, \lambda_2 \ldots, \lambda_T, \underbrace{\theta, \ldots, \theta}_{(n-T)}$ and corresponding eigenvectors $\vec{q}_1, \ldots, \vec{q}_n$. We further require $q_1(i) > 0$ for all i. We do not impose monotonicity constraint for those λ_k except having $\lambda_1 = 0$; later, we may re-arrange them in non-descending order and will index them by subscript $\cdot_{(k)}$.

Collecting the eigenvectors and eigenvalues in matrices $Q = (\vec{q}_1, \ldots, \vec{q}_T)$ and $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_T)$, and after re-arranging terms in (2) we obtain

$$\begin{aligned} \mu_L &= \sum_{k=1}^{T} (\lambda_k - \theta) \vec{q}_k \vec{q}_k' + \sum_{l=1}^{n} \theta \vec{q}_l \vec{q}_l' \\ &= Q(\Lambda - I_T \theta) Q' + I_n \theta, \end{aligned} \tag{3}$$

where $\vec{q}_{T+1}, \ldots, \vec{q}_n$ are canceled due to orthonormality, $\sum_{l=1}^{n} \vec{q}_l \vec{q}_l' = I_n$. Therefore, our parameter of interest is $\beta_L = (Q, \Lambda, \theta, \sigma_e^2)$, which has $\mathcal{O}(nT)$ many elements.

Remark 3 *This model shares similarities with the spiked covariance model (Donoho et al., 2018), except that the "spikes" $\lambda_2, \ldots, \lambda_T$ are associated with the* smallest *eigenvalues, that as shown later, drive the partitioning of the graph into communities. For this reason, we coin the term "spiked graph Laplacian" for μ_L.*

2.1 A Non-parametric Bayesian model for Heterogeneous Spiked Graph Laplacians

A key benefit of the probabilistic model introduced for the spiked graph Laplacian is that it enables us to naturally capture heterogeneity in a collection of graphs $G^{(s)}, s = 1, \cdots, S$, with associated Laplacians and their decompositions $(\mu_L^{(s)}, Q^{(s)}, \theta^{(s)}, \Lambda^{(s)})$. Note that in (2), each $\vec{q}_k^{(s)}$ forms a factor matrix $\vec{q}_k^{(s)} \vec{q}_k^{(s)\prime}$ encoding the pairwise interactions of the vertices, while each $\lambda_k^{(s)}$ modulates the magnitude of the interactions.

Given such a heterogeneous collection of graphs/networks, in order to learn both the shared community structure across them, and also capture their heterogeneity as reflected in their edge density, we use a two-pronged approach: (i) a non-parametric Bayesian model is used for estimating a common dictionary of factors, and (ii) a random-effects model controls the number of spikes for each graph Laplacian $L^{(s)}$.

The matrix of eigenvectors $Q^{(s)}$ is modeled based on a Dirichlet process mixture,

$$\begin{aligned} &Q^{(s)} \sim \sum_{l=1}^{\infty} \pi_l \delta_{U^{(l)}}(.), \qquad \Pi(U^{(l)}) \propto \exp\left\{\mathrm{tr}\left[\Omega M' U^{(l)}\right]\right\} \mathbf{I}[u_1^{(l)}(i) > 0 \text{ for } i = 1, \ldots, n], \\ &\pi_1 = \nu_1, \quad \pi_l = \nu_l \prod_{l'<l} (1 - \nu_{l'}) \text{ for } l > 1, \\ &\nu_l \sim \mathrm{Beta}(1, \alpha_0), \end{aligned} \tag{4}$$

whose base measure is a constrained matrix Langevin distribution, on a Stiefel sub-manifold with the elements in the first column being all positive, $\mathcal{V}_*^{T,n} = \{Q \in \mathbb{R}^{n \times T} : Q'Q = I_T, q_1(i) > 0, i = 1 \ldots n\}$; Ω a diagonal $T \times T$ matrix; the concentration parameter $\alpha_0 > 0$; $\delta_a(.)$ a point mass at a; $\mathbf{I}(E)$ takes the value 1 if E holds, and 0 otherwise.

An important property of the Dirichlet process mixture is that the posterior distribution is discrete almost surely. Therefore, using this non-parametric prior distribution allows us to obtain a discrete distribution for $Q^{(s)}$, where $Q^{(1)} \ldots, Q^{(S)}$ have only a few unique values that are significantly less than S. That is, we learn a "dictionary" of the eigenmatrices.

Remark 4 *Since $L^{(s)} = \mu_{L^{(s)}} + \mathcal{E}^{(s)}$, the term $\mathcal{E}^{(s)}$ gives a perturbation to the eigenvectors of $L^{(s)}$, so that they become unique for each subject $s = 1, \ldots, S$. Therefore, even though we use a Dirichlet process prior that makes the distribution of $Q^{(s)}$ (eigenvectors of $\mu_{L^{(s)}}$) discrete — that is, we allow the possibility of having $Q^{(s)} = Q^{(s')}$ for two subjects s and s', the corresponding observed Laplacian matrices $L^{(s)}$ and $L^{(s')}$ do not have the same eigenvectors (otherwise, the model would be too restrictive).*

The eigenvalues $\lambda_k^{(s)}$ and $\theta^{(s)}, s = 1, \ldots, S$ are assumed independently and identically distributed according to the following prior distribution:

$$\begin{aligned} &\eta_k^{(s)} \sim \mathrm{Bernoulli}(w), \\ &\lambda_k^{(s)} \mid \eta_k^{(s)} = 1 \sim N_{(0,2)}(0, \sigma_{\lambda,1}^2), \quad \lambda_k^{(s)} \mid \eta_k^{(s)} = 0 \sim N_{(0,2)}(\mu_\theta, \sigma_{\lambda,0}^2), \\ &\theta^{(s)} \sim N_{(0,2)}(\mu_\theta, \sigma_\theta^2), \end{aligned} \tag{5}$$

for $k = 2, \ldots, T$, with $N_{(0,2)}$ denoting a Gaussian distribution truncated to the $(0, 2)$ interval. Further, since $\lambda_1^{(s)} = 0$, we assign $\eta_1^{(s)} = 1$. When marginalizing over $\eta_k^{(s)}$, each $\lambda_k^{(s)}$ follows a two-component mixture, with the first component capturing small spikes, and the second component capturing those large ones close to θ. This enables a constant dimension T for all $L^{(s)}$, while retaining adaptiveness to have the effective number of small spikes:

$$\kappa^{(s)} = \sum_{k=1}^{T} \eta_k^{(s)}, \tag{6}$$

as shown later, equivalent to $\kappa^{(s)}$ communities.

Remark 5 *An alternative parameterization would be using $T^{(s)}$ that varies directly with each graph; however, this would lead to an inefficient discrete search when estimating the*

posterior distribution. In theory, one could also consider fixing $T = n$; nevertheless, when setting $T \ll n$, we only need to estimate the first T eigenvectors [due to (3)], and the algorithm is computationally more efficient than when $T = n$.

Our parameterization of the mixture prior in (5) is motivated by the observation that those large eigenvalues of Laplacian have a concentration at a value away from 0 (see Figure 2). Therefore, we make the second-component location μ_θ non-zero, along with a small scale $\sigma^2_{\lambda,0}$. Another possibility is the continuous spike-and-slab prior (George and McCulloch, 1993) with the second location $\mu_\theta = 0$, a large scale $\sigma^2_{\lambda,0}$, and the truncated support on $(0, 2)$; nevertheless, under that prior, those small but not close-to-zero $\lambda_k^{(s)}$'s will be more likely to be assigned to the component with $\eta_k^{(s)} = 0$ (as in the "slab" group), which is not ideal for our modeling purpose.

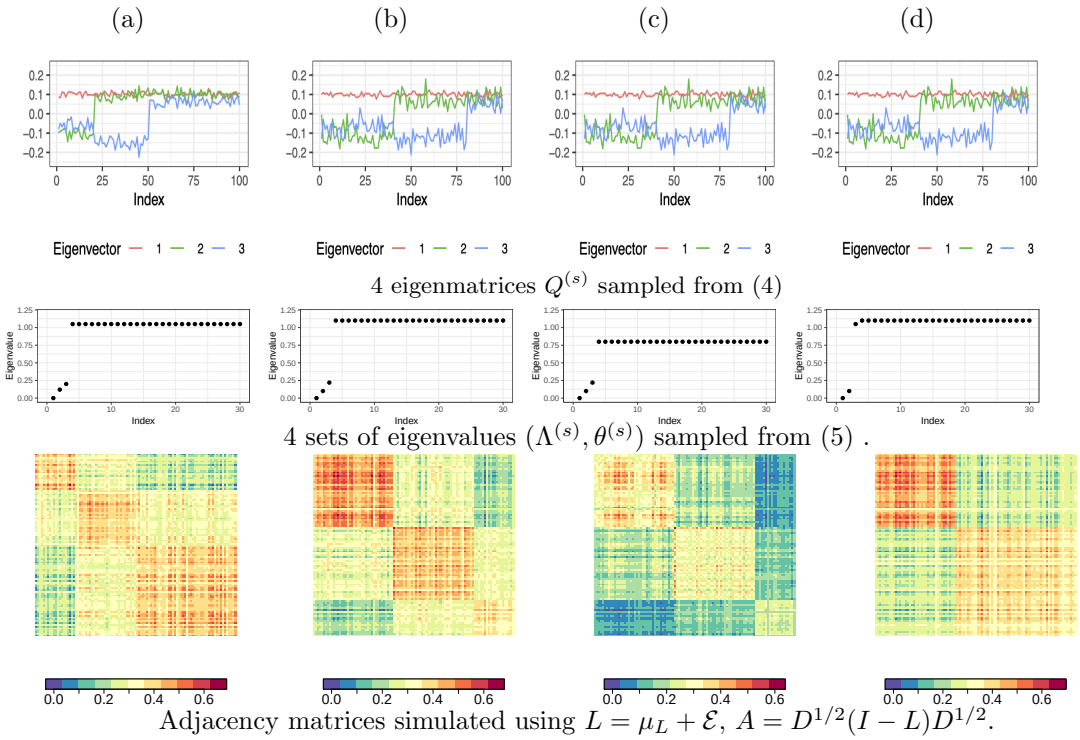


Figure 1: Simulated adjacency matrices illustrating how the non-parametric spiked Laplacian model captures graph heterogeneity: Graphs (b), (c), (d) use the same eigenmatrix $Q^{(s)}$ drawn from the Dirichlet process, creating similar community structure; the independent eigenvalues $\Lambda^{(s)}$ lead to varying degree of sparsity ((b) vs (c)) and also dictate whether a community can be further divided into two smaller communities ((b) vs (d)). Graph a take a different value for $Q^{(s)}$; hence its community structure is completely different from (b), (c), (d).

Next, we illustrate the high flexibility of the proposed modeling framework based on synthetic data. We draw four eigenmatrices from (4) and four sets of eigenvalues from (5), and obtain the adjacency matrix using (2) with $\sigma_e^2 = 10^{-2}$. As shown in Figure 1: (i) Graphs (b), (c), (d) have the same values in the eigenmatrix $Q^{(s)}$, therefore they share a similar community structure and appeare quite different from graph (a); (ii) among those three,

the independent eigenvalues $\Lambda^{(s)}$ create varying edge, thus leading to different strengths in connectivity between graphs (b) and (c), and also dictate whether a community can be further divided into two smaller communities [(b) vs (d)].

2.2 Specification of the Prior Distribution

For the variance parameters σ_θ^2, $\sigma_{\lambda,0}^2$ and $\sigma_{\lambda,1}^2$, we assign proper Inverse-Gamma$(2, 0.1)$ distributions with a weakly informative prior mean of 0.1. To choose the mean parameter μ_θ of those larger eigenvalues, we consider the idealized case of having a graph G^* consisting of K disjoint complete subgraphs — the kth subgraph has $n_k > 1$ vertices, among which each pair of vertices are connected with an equal edge weight $A_{i,j} = a > 0$. Then for this graph G^*, we have that the eigenvalues of its normalized Laplacian are given by:

$$\Bigg(\underbrace{0,\ldots,0}_{K},\underbrace{\frac{n_1}{n_1-1},\ldots,\frac{n_1}{n_1-1}}_{(n_1-1)},\ldots,\underbrace{\frac{n_K}{n_K-1},\ldots,\frac{n_K}{n_K-1}}_{(n_K-1)}\Bigg), \tag{7}$$

which can be derived as a direct extension of (6) from Banerjee and Jost (2008). Therefore, with most of $n_k/(n_k-1) \approx 1$ and viewing the observed graph as some deviation from G^*, we set the prior mean $\mu_\theta = 1$ in this article. As a flexible alternative, one could further use a hierarchical prior $\mu_\theta \sim N(1, \sigma_{\mu 1}^2)$, so that μ_θ can be adaptive to the data.

For w, we assign a non-informative prior Beta$(1, 1)$ distribution. For the noise variance σ_e^2, we set a diffuse prior Inverse-Gamma$(0.01, 0.01)$ distribution. For the base measure of the Dirichlet process (4), we choose the non-informative $\Omega = \text{diag}(0, \cdots, 0)$, making it a uniform prior measure over $\mathcal{V}_*^{T,n}$ and eliminating the need to estimate M or any intractable normalizing constant. We choose a small concentration $\alpha_0 = 0.1$ to induce sparsity in the mixture weights, which lead to fewer unique values in $Q^{(s)}$, thus aiding interpretation by having few communities. Note that one can always select a larger α_0 value, if more communities with finer-scale differences is desired.

In numerical experiments, this prior specification shows good empirical performance in recovering the ground truth and is robust to a wide range of values of n, S and noise levels without the need for tuning.

2.3 Estimation of the Posterior Distribution

We use Gibbs sampling to estimate the posterior distribution. Since an infinite mixture distribution is involved, we use a latent assignment $z_s \in \{1, 2, \ldots\}$ for each graph, such that $Q^{(s)} = U^{(l)}$ if $z_s = l$. Then, the likelihood given $\{z_s\}$ becomes

$$\begin{aligned}\prod_{s=1}^{S} \Pi(L^{(s)};\quad & \sigma_e^2, \Lambda^{(s)}, Q^{(s)}, \theta^{(s)}, z^{(s)}) \\ & \propto (\sigma_e^2)^{-\frac{Sn(n+1)}{4}} \exp\Bigg(-\sum_{s=1}^{S} \frac{1}{4\sigma_e^2}\Big\{ \text{tr}\big[(\Lambda^{(s)} - \theta^{(s)} I_T)^2\big] + \|L - \theta^{(s)} I_n\|_F^2 \Big\} \\ & + \sum_{l=0}^{\infty} \sum_{s:z_s=l} \frac{1}{2\sigma_e^2} \text{tr}\big[(\theta^{(s)} I_n - L^{(s)}) U^{(l)} (\theta^{(s)} I_T - \Lambda^{(s)}) U^{(l)\prime}\big] \Bigg).\end{aligned} \tag{8}$$

In the above, we replaced the fixed diagonal elements $L^{(s)}_{i,i} = 1$ with an augmented random variant $L^{(s)}_{i,i} = N(\mu_{L,i,i}, 2\sigma_e^2)$, for easier matrix-based computation as in Hoff (2009).

A Gaussian Integral Trick for the Product-Matrix-Bingham Distribution: One immediate challenge of sampling $U^{(l)}$ from (8) is the exponential-quadratic in the full conditional distribution:

$$\Pi(U^{(l)} \mid .) \propto \exp\left\{\frac{1}{2\sigma_e^2}\sum_{s:z_s=l}\mathrm{tr}(F_s U^{(l)} G_s U^{(l)\prime})\right\}\mathrm{etr}(\Omega M' U^{(l)}), \tag{9}$$

where $F_s = \theta^{(s)} I_n - L^{(s)}$ and $G_s = \theta^{(s)} I_T - \Lambda^{(s)}$. This corresponds to the product of a matrix Bingham-$\{F_s/(2\sigma_e^2), G_s\}$ distribution, for which a closed form is unavailable for sampling purposes.

To address this problem, we propose a new data augmentation for the product-matrix-Bingham distribution, which extends the Gaussian integral trick (Zhang et al., 2012) on the Stiefel manifold. Consider an augmented random matrix $R_s \in \mathbb{R}^{T\times n}$ from the matrix Gaussian Mat-No$(G_s U' F_s, G_s\sigma_e^2, F_s)$:

$$\Pi(R_s \mid U^{(l)}, .) \propto |F_s|^{-T/2}|G_s|^{-n/2}\mathrm{etr}\left\{-\frac{1}{2\sigma_e^2}F_s^{-1}(R_s - G_s U^{(l)\prime}F_s)'G_s^{-1}(R_s - G_s U^{(l)\prime}F_s)\right\}, \tag{10}$$

The joint distribution becomes:

$$\begin{aligned}&\Pi(\{R_s\}_{s:z_s=l}, U^{(l)} \mid .)\\ &\propto \exp\left\{-\frac{1}{2\sigma_e^2}\sum_{s:z_s=l}\left[\mathrm{tr}(F_s^{-1}R_s'G_s^{-1}R_s) - 2\mathrm{tr}(R_s' U^{(l)\prime})\right]\right\}\mathrm{etr}\left[\Omega M' U^{(l)}\right],\end{aligned}$$

where all the quadratic terms in (9) are canceled, leading to full conditional

$$\Pi(U^{(l)} \mid \{R_s\}_{s:z_s=l}, .) \propto \mathrm{etr}\left(\frac{1}{\sigma_e^2}\sum_{s:z_s=l}R_s U^{(l)} + \Omega M' U^{(l)}\right). \tag{11}$$

Therefore, we can sample (10) and (11) alternatively in closed form; note that the latter is a matrix Langevin distribution amenable to the sampling algorithm in Hoff (2009).

Sampling Algorithm: To simplify computations, we approximate the Dirichlet process mixture model with a truncated version, setting the number of mixture components to g and using $\mathrm{Dir}(\alpha_0/g, \ldots, \alpha_0/g)$ (in this paper, we use $g = 30$). The detailed steps of the algorithm are given in the Appendix.

3. Community Detection based on the Posterior Distribution

Next, we focus on the community assignment labels $c_i^{(s)} \in \mathbb{N}$ for each vertex i in graph s, using the obtained posterior sample of $Q^{(s)}$ and $\Lambda^{(s)}$. Specifically, we obtain $\{c_i^{(s)}\}_{i=1}^n$ via a fast and deterministic transformation of $(Q^{(s)}, \Lambda^{(s)})$, in which the algorithm aims to

optimize the partitioning of each graph. Note that this is a measurable transformation, we quantify the uncertainty via the induced distribution of $c_i^{(s)}$. Since the discussion pertains to each graph, we omit superscript (s) for ease of presentation.

Optimal Graph Cut

We first introduce the concept of "optimal graph cut"'. In the simplest possible case, suppose we want to bi-partition (or, "cut") a graph $G = (V, E)$ into two sub-graphs $G_1 = \big(V_1, E(V_1, V_1)\big)$ and $G_2 = \big(V_2, E(V_2, V_2)\big)$, with $V_1 \cup V_2 = V$ and $V_1 \cap V_2 = \varnothing$, and $E(V_j, V_j)$ the edges formed among the vertices within V_j. An intuitive cut criterion corresponds to minimizing the loss of edge weights between two sub-graphs: $\sum_{i\in V_1, j\in V_2} A_{i,j}$.

On the other hand, we want to prevent trivial cuts, where one of the partition vertex sets $V_j, j = 1, 2$ comprises of few or even a single vertex. To that end, Shi and Malik (2000) introduced the minimal *normalized cut* loss defined as

$$h_2(G) = \min_{(V_1,V_2)} \frac{\sum_{i\in V_1, j\in V_2} A_{i,j}}{\min_{l=1,2} \sum_{i,j\in V_l} A_{i,j}},$$

where the denominator is the sum of the vertex degrees in one of two subgraphs. Initially, $h(G)$ was proposed for a binary adjacency matrix A, and is also known as the Cheeger or isoperimetric constant (Mohar, 1989), representing the bottleneck of the flow across the edges connecting the two partitioned vertex sets; later on, this loss was extended to weighted graphs (Friedland and Nabben, 2002).

Louis et al. (2011) extends it to κ-partitioning of a weighted graph, with the corresponding loss function known as the "sparsest κ-cut":

$$h_\kappa(G) = \min_{(V_1,\ldots,V_\kappa)} \frac{\sum_{m<l} \sum_{i\in V_m, j\in V_l} A_{i,j}}{\min_{l=1,\ldots,\kappa} \sum_{i,j\in V\setminus V_l} A_{i,j}},$$

where $(V_1, \ldots, V_\kappa)$ is a partitioning of V.

Interestingly, the optimal values of these losses are upper-bounded by the eigenvalues of the graph Laplacian. Consider the graph associated with μ_L; we then have

$$h_2(G) \le \sqrt{2\lambda_{(2)}}, \qquad h_\kappa(G) \le (8 \log \kappa)\sqrt{\lambda_{(\kappa)}} \text{ for } \kappa \ge 3, \tag{12}$$

where the former is due to Friedland and Nabben (2002), and the latter due to Louis et al. (2011), with $\lambda_{(k)}$ denoting the k-th smallest eigenvalue in $\{\lambda_1, \ldots \lambda_T\}$.

Recall that in the spiked graph Laplacian model, there are κ small spikes; see, (6). Hence, since $\lambda_{(1)}, \ldots, \lambda_{(\kappa)} \approx 0$, κ communities can be extracted with negligible graph-cut loss.

3.1 Sign-based Partitioning

Finding the best κ-partition is a challenging problem computationally, due to the combinatorial search required. However, there are numerous algorithms in the literature that approximate the optimal cut. Examples include the spectral clustering (Ng et al., 2002) and the random search algorithm (Louis et al., 2011). In particular, the latter one is shown to

achieve a loss smaller than $(8 \log \kappa)\sqrt{\lambda_{(\kappa)}}$ for any $\lambda_{(\kappa)}$, although the computations involved can be intensive.

Algorithm 1 Sign-based κ-partitioning.

Initialize: $V_{[1]1} = \{1, \ldots, n\}$, re-order $\{\vec{q}_k\}_{k=1}^T$ according to non-descending order of λ_k, denoted by $\{\vec{q}_{(k)}\}_{k=1}^T$.
for $k = 1$ **to** $(\kappa - 1)$ **do**

1. Compute the sign-based partitioning loss when dividing the $[k]l$th existing partition, for $l = 1, \ldots, k$:

$$loss_{[k]l} = \sum_{i,j \in V_{[k]l}} \big[q_{(k)}(i) q_{(k)}(j)\big] 1\big[q_{(k)}(i) q_{(k)}(j) < 0\big].$$

2. Find $l^* = \underset{l \in \{1 \ldots, k\}}{\arg\min}\, loss_{[k]l}$, add one partition by setting:

$$\begin{aligned} V_{[k+1]l^*} &:= \{i \in V_{[k]l^*} q_k(i) \geq 0\}, \\ V_{[k+1](k+1)} &:= \{i \in V_{[k]l^*} : q_k(i) < 0\}, \\ V_{[k+1]l} &:= V_{[k]l} \text{ for } l \neq l^*, l \leq k. \end{aligned}$$

end for
Use $\{V_{[\kappa]l}\}_{l=1}^{\kappa}$ as the κ-partition; record $c_i = l$ for $i \in V_{[\kappa]l}$.

Inspired by the famous Fiedler vector (Fiedler, 1989), we propose a more efficient algorithm using the signs of the eigenvectors (see Algorithm 1).

The justification for the key steps in the proposed algorithm is as follows. Examine the off-diagonal elements of the smoothed adjacency matrix $A_* = D^{1/2}(I - \mu_L)D^{1/2}$

$$A_{*,i,j} = d_i d_j \sum_{k=1}^{T} (\theta - \lambda_{(k)}) q_{(k)}(i) q_{(k)}(j), \qquad i \neq j, \tag{13}$$

for $d_i > 0, d_j > 0$ and $(\theta - \lambda_{(k)}) > 0$ for small $\lambda_{(k)}$. If $q_k(i)$ and $q_k(j)$ have the same sign, they contribute positively to $A_{*,i,j}$. Therefore, to minimize the loss due to a graph cut, a locally optimal cut is simply dividing the set into two subsets — the one with $q_{(k)} \geq 0$ and the one with $q_{(k)} < 0$. In the simplest case with $\kappa = 2$, this is exactly the Fiedler vector partitioning (Fiedler, 1989). We do this recursively until obtaining κ subsets.

Due to the orthonormality of the eigenvectors, the following holds for $k \geq 2$,

$$\|\vec{q}_{(k)}\| = 1, \qquad \sum_{i=1}^{n} q_{(1)}(i) q_{(k)}(i) = 0, \qquad q_{(1)}(i) > 0.$$

To satisfy these constraints, each vector $q_{(k)}$ must contain both plus and minus signs; hence, we can always use the sign-based partitioning. This algorithm can run very fast, since it only takes one scan from 1 to κ.

3.2 Obtaining Point Estimates from Posterior Samples

Based on the posterior samples, it is of interests to obtain point estimates for both interpration and benchmarking purposes.

On estimating the effective number of patterns, as $z_s = l$ represents assigning $L^{(s)}$ to the l-th group, for each posterior sample, we record the number of unique values in $\{z_s\}_{s=1,\cdots,S}$, denoted by b_z. Using the posterior samples, we obtain a discrete distribution for $b_z = 1, 2, \cdots$, and we take the one with the largest probability as a point estimate $\hat{b}_z$.

To estimate the number of communities for each subject, for each posterior sample of $\Lambda^{(s)}$, we obtain a $\kappa^{(s)}$ from (6) as the number of communities. Similarly based on the posterior samples, we obtain a discrete distribution for $\kappa^{(s)} = 1, 2, \ldots$ and pick the one with the largest probability as a point estimate for $\hat{\kappa}^{(s)}$.

To obtain a point estimate of community assignments for the s-th subject, conditioned on our point estimate on the number of communities, we take those samples $Q^{(s)} : \kappa^{(s)} = \hat{\kappa}^{(s)}$, and find one that has the largest posterior density, and run Algorithm 1 to obtain assignment labels $\hat{c}_i^{(s)}$'s to $\hat{\kappa}^{(s)}$ communities.

4. Theoretical Results

Next, we establish several properties of the proposed methodology. We first show that adapting $\kappa^{(s)}$ for the graph involves a trade-off between the number of eigenvectors to be estimate and their estimation accuracy compared to the ground truth under noise perturbations; hence, this is a trade-off between the number of communities $\kappa^{(s)}$ one attempts to identify and the uncertainty/error on the estimated community membership $c_i^{(s)}$.

Assume L is a noisy version of a true L_0 (not necessarily having a spiked structure), with Q_0 containing its eigenvectors. The spiked graph Laplacian model produces an estimated $\hat{L} = \hat{Q}(\lambda - I_T\theta)\hat{Q}' + I_n\theta$ based on the posterior distribution. We can quantify the distance between the sub-matrices of $\hat{Q}$ and Q_0.

Theorem 6 (Trade-off between resolution and estimating accuracy) *For any given posterior sample from the spiked graph Laplacian model, let the eigen-vectors/values in the spiked graph Laplacian estimate be ordered such that* $\lambda_{(1)} < \lambda_{(2)} \leq \lambda_{(3)} \ldots \leq \lambda_{(T)} < \lambda_{(T+1)} = \ldots = \lambda_{(n)} = \theta$. *Further, assume each element of* $(\hat{L} - L_0)$ *is* σ_e*-sub-Gaussian, due to both* L_0 *and* $\hat{L}$ *being normalized Laplacians and thus all their elements are in the* $[-1, 0]$ *interval. Denote the sub-matrices formed by the first* k *columns of the* $\hat{Q}$ *and* Q_0 *matrices as* $\hat{Q}_{1:k}$ *and* $Q_{0,1:k}$*, respectively; then, for any* $k \in [2, T-1]$*, there exists an orthonormal matrix* O*, for any* $t > 0$*:*

$$\Pr\left(\|\hat{Q}_{1:k}O - Q_{0,1:k}\|_F \leq \frac{\sqrt{kn}2^{3/2}\sigma_e}{\lambda_{(k+1)} - \lambda_{(k)}}t\right) \geq 1 - \delta_t,$$

where $\delta_t = \exp[-\{t^2/64 - \log(5\sqrt{2})\}n]$.

Next, note that the likelihood function can be re-written as,

$$\Pi(L;\sigma_e^2,\Lambda,Q,\theta)\propto(\sigma_e^2)^{-n(n+1)/4}\exp\Big(-\frac{1}{4\sigma_e^2}\big[\mathrm{tr}(\Lambda\Lambda)-2\mathrm{tr}(\Lambda Q'LQ)\big]\Big)$$
$$\exp\Big(-\frac{1}{4\sigma_e^2}\big[\theta^2(n-T)-2\theta\mathrm{tr}\{L(I-QQ')\}\big]\Big)\exp\big\{-\frac{1}{4\sigma_e^2}\mathrm{tr}(LL)\big\},$$

where Λ and θ are conditionally independent. Integrating out Λ and θ, we obtain the marginal likelihood of Q:

$$\Pi(L;\sigma_e^2,Q)\propto\exp\left\{\frac{(\sum_{k=T+1}^n q_k'Lq_k)^2}{4\sigma_e^2(n-T)}\right\}\exp\left\{\frac{\sum_{k=1}^T(q_k'Lq_k)^2}{4\sigma_e^2}\right\}\zeta,$$

with Φ denoting the cumulative distribution function of the normal distribution and

$$\zeta=(\sigma_e^2)^{-n(n+1)/4+(T+1)/2}\prod_{k=2}^T\{\Phi(\frac{2-q_k'Lq_k}{\sqrt{2\sigma_e^2}})-\Phi(\frac{-q_k'Lq_k}{\sqrt{2\sigma_e^2}})\}$$
$$\times[\Phi\{\frac{2-(\sum_{k=T+1}^n q_k'Lq_k)/(n-T)}{\sqrt{2\sigma_e^2/(n-T)}}\}-\Phi\{\frac{-(\sum_{k=T+1}^n q_k'Lq_k)/(n-T)}{\sqrt{2\sigma_e^2/(n-T)}}\}].$$

Remark 7 *To see the intuition regarding the marginal likelihood, consider* $-\log\Pi(L;\sigma_e^2,Q)$ *as a loss function over* Q*, while ignoring the normalizing constant* ζ*,*

$$-\sum_{k=1}^T(q_k'Lq_k)^2-\frac{1}{n-T}(\sum_{k=T+1}^n q_k'Lq_k)^2=-\sum_{k=1}^T(q_k'Lq_k)^2-\frac{m}{n-T}\sum_{k=T+1}^n(q_k'Lq_k)^2,$$

where $m\in[1,2]$ *due to* $\sum x_k^2\le(\sum x_k)^2\le 2\sum x_k^2$*, with* $x_k=q_k'Lq_k\ge 0$ *with* L *being positive semi-definite. Therefore, the first* T *factors* $(q_k'Lq_k)^2$ *have a substantially higher contribution compared to the remaining ones, which is consistent with our modeling focus on the first* T *eigenvectors.*

Lastly, we show that the proposed non-parametric model of the matrix containing the eigenvectors is posterior consistent. There has been theoretical work on community detection and eigenvector estimation for single graphs, assuming that the number of vertices n goes to infinity. A fundamental difference here is that we have fixed and bounded n in each graph, but the number of graphs S grows. Hence, a new theoretical approach is required.

In order to avoid a potential discrepancy between the number of spikes in the true and prescribed models, we use the *full* eigen-decomposition for the raw observed $L^{(s)}=W^{(s)}\Omega^{(s)}W^{(s)\prime}$, where $W^{(s)}$ is an orthonormal matrix and $\Omega^{(s)}$ diagonal. Note that $W^{(s)}$ belong to a a Stiefel sub-manifold $\mathcal{V}^*\subseteq\mathcal{V}^{n,n}$, with the first column elements being all positive. Similarly, for the spiked graph Laplacian we have $\mu_L=Q^\dagger\Lambda^\dagger Q^{\dagger\prime}$, where $Q^\dagger\in\mathcal{V}^*$ and the first T columns equal to parameter Q, $\Lambda^\dagger=\mathrm{diag}\{\lambda_1,\ldots,\lambda_T,\theta,\ldots,\theta\}$.

Using f to denote the likelihood, each observed $L^{(s)}=W^{(s)}\Omega^{(s)}W^{(s)\prime}$ can be generated from

$$f(W^{(s)},\Omega^{(s)}\mid Q^\dagger,\Lambda^\dagger)\propto\underbrace{\mathrm{etr}\{\frac{1}{2\sigma_e^2}Q^\dagger\Lambda^\dagger Q^{\dagger\prime}W^{(s)}\Omega^{(s)}W^{(s)\prime}\}}_{f(W^{(s)}|\Omega^{(s)},Q^\dagger,\Lambda^\dagger)}\underbrace{\mathrm{etr}\{-\frac{1}{4\sigma_e^2}\big[\Omega^{(s)}\Omega^{(s)}+\Lambda^\dagger\Lambda^\dagger\big]\}}_{f(\Omega^{(s)}|\Lambda^\dagger)}.$$

The former corresponds to $W^{(s)} \sim \text{Matrix-Bingham}\big[\Omega^{(s)}, (2\sigma_e^2)^{-1}Q^\dagger\Lambda^\dagger Q^{\dagger\prime}\big]$, in which $Q^\dagger$ serves as the location parameter.

Therefore, based on a non-parametric mixture priordistribution for $Q^\dagger$, our task is equivalent to showing the consistency of estimating $Q^\dagger \in \mathcal{V}^*$ under the Matrix-Bingham likelihood. Using the $Q^\dagger$-marginal density $f_{Q^\dagger}(W^{(s)}) = \int\int f(W^{(s)}, \Omega^{(s)} \mid Q^\dagger, \Lambda^\dagger) P(\mathrm{d}\Lambda^\dagger, \mathrm{d}\Omega^{(s)})$, where $P(.)$ denotes the appropriate measure, consider a neighborhood of the true density $f_{Q^\dagger,0}$ on the manifold $\mathcal{V}^*$ as

$$B_\epsilon(f_{Q^\dagger,0}) = \left\{ f_{Q^\dagger} : \left| \int g f_{Q^\dagger} \mu(\mathrm{d}W) - g f_{Q^\dagger,0} \mu(\mathrm{d}W) \right| \leq \epsilon, \quad \forall g \in C_b(\mathcal{V}^*) \right\},$$

with C_b denoting the class of continuous and bounded functions, and $\mu(.)$ the Haar measure on $\mathcal{V}^*$. Next, we establish that the probability for the posterior density falling into $B_\epsilon(f_{Q^\dagger,0})$ goes to 1 as $S \to \infty$.

Theorem 8 (Consistent density estimation for the eigenmatrix) *Let $W^{(1)} \dots W^{(S)}$ be matrices of eigenvectors, whose elements are independently and identically distributed from a distribution with density $f_{Q^\dagger,0}$. Then, for all $\epsilon > 0$, as $S \to \infty$,*

$$\Pi\big\{B_\epsilon(f_{Q^\dagger,0}) \mid W^{(1)}, \dots, W^{(S)}\big\} = \frac{\int_{B_\epsilon(f_{Q^\dagger,0})} \prod_{s=1}^S f_{Q^\dagger}(W^{(s)}) \Pi(\mathrm{d}f)}{\int \prod_{s=1}^S f_{Q^\dagger}(W^{(s)}) \Pi(\mathrm{d}f)} \to 1 \; a.s. P f^\infty_{Q^\dagger,0},$$

with $P f^\infty_{Q^\dagger,0}$ the true probability measure for $(W^{(1)}, W^{(2)}, \dots)$.

Remark 9 *Although the space of eigenmatrix is a compact domain, the primary challenge is that there are more than one (up to infinite) different data-generating eigenmatrices, due to the presence of heterogeneity. Therefore, this theorem shows that we can obtain consistency in the sense of density estimation of the population of those eigenmatrices, as opposed to having the posterior converge to any fixed eigenmatrix.*

The consistency in density estimation shows that we can accurately estimate the true data generating distribution for the population of networks.

On clustering the subjects, suppose we have K local maxima in the density $f(W^{(s)})$ ($W^{(s)}$ as all the n eigenvectors of $L^{(s)}$), a consistently estimated density will ensure: (i) if an observation $L^{(s)}$ has its $W^{(s)}$ sufficently close to the kth local maximum, then a classifier that maximizes the density:

$$\arg\max_{l=1\dots K} f(W^{(s)} \mid Q^\dagger = U^{(l)}, \Lambda^\dagger, \Omega^{(s)})$$

will correctly assign subject s to the kth group (provided neither $\Lambda^\dagger$ and $\Omega^{(s)}$ is a zero-valued matrix); (ii) on the other hand, if an observation $L^{(s)}$ has its $W^{(s)}$ almost equally distant away from several local maxima, then we may not perfectly cluster subject s; nevertheless, we can quantify the uncertainty via $\Pr(z_s = k \mid .) = f(W^{(s)} \mid Q^\dagger = U^{(k)}, \Lambda^\dagger, \Omega^{(s)}) / [\sum_{l=1}^K f(W^{(s)} \mid Q^\dagger = U^{(l)}, \Lambda^\dagger, \Omega^{(s)})]$. That is, we will have the mis-clustering error converge asymptotically to the Bayes error rate (as an irreducible error).

Regarding the theoretical guarantees of finding communities for each subject, we want to clarify that the obtained results are based on a *fixed* (and small) n (number of vertices)

and growing S (number of subjects) setting. We purposely let the accuracy of community detection vary from one subject to another — as one can imagine, due to the heterogeneity of data, it is likely that two networks may share the same community-partition pattern (that is, we may have $z_s = z_{s'}$), but network s may be "noisier" as having much more between-community connections than network s'. Mathematically, this is reflected in the eigenvalues with $\lambda_k^{(s)} \gg 0$, but $\lambda_k^{(s')} \approx 0$. As we consider $\{\lambda_k^{(s)}\}_{k=1...T}$ to be *random effects* that vary over s, we do not aim for obtaining a convergence result for any *individual* subject. Rather, we aim for discovering a few shared community-partition patterns, each represented by $U^{(l)}$ in Theorem 8. On the other hand, if one wants to obtain some asymptotic guarantee on the community detection error rate for a *specific* subject, it is necessary to consider a very different scenario with $n \to \infty$, and impose some stronger assumption on the eigenvalues. Subsequently, one can show that the first few eigenvectors will converge to the corresponding population eigenfunction (Von Luxburg et al., 2008).

5. Performance Evaluation based on Synthetic Data

5.1 Impact of Different Noise Levels on a Single Graph

We first examine the effects of noise on estimating the communities in a single graph. We generate a weighted graph comprising of 60 vertices and three communities of size 10, 20 and 30 vertices, respectively. To avoid directly using the proposed model to generate data, we simulate each edge within the communities as a Bernoulli event with probability 0.5, and then add Gaussian noise $\mathrm{No}(0, \xi^2)$ to the adjacency matrix with varying ξ^2.

As shown in Figure 2, the 3-community structure can be visualized by the spectral gap between the third and fourth eigenvalues. As the noise increases, the gap diminishes, making it more difficult to separate the communities.

Observed Spectral Gap $(\hat{\lambda}_4 - \hat{\lambda}_3)$	0.6	0.3	0.1	0.05	0.01
Spiked Laplacian	(1 ± 0)	(0.95 ± 0.05)	(0.88 ± 0.09)	(0.58 ± 0.20)	(0.40 ± 0.14)
Observed Laplacian	(1 ± 0)	(0.91 ± 0.07)	(0.78 ± 0.05)	(0.42 ± 0.20)	(0.36 ± 0.25)

Table 1: The spiked Laplacian model has higher accuracy in recovering community labels, comparing to direct decomposition of the observed Laplacian. The results are calculated by clustering the second and third eigenvectors into three groups (patterns), then comparing with the ground truth labels to compute normalized mutual information (NMI). Mean $\pm$ standard deviation is reported based on 50 times of experiments. The higher the NMI, the higher the accuracy.

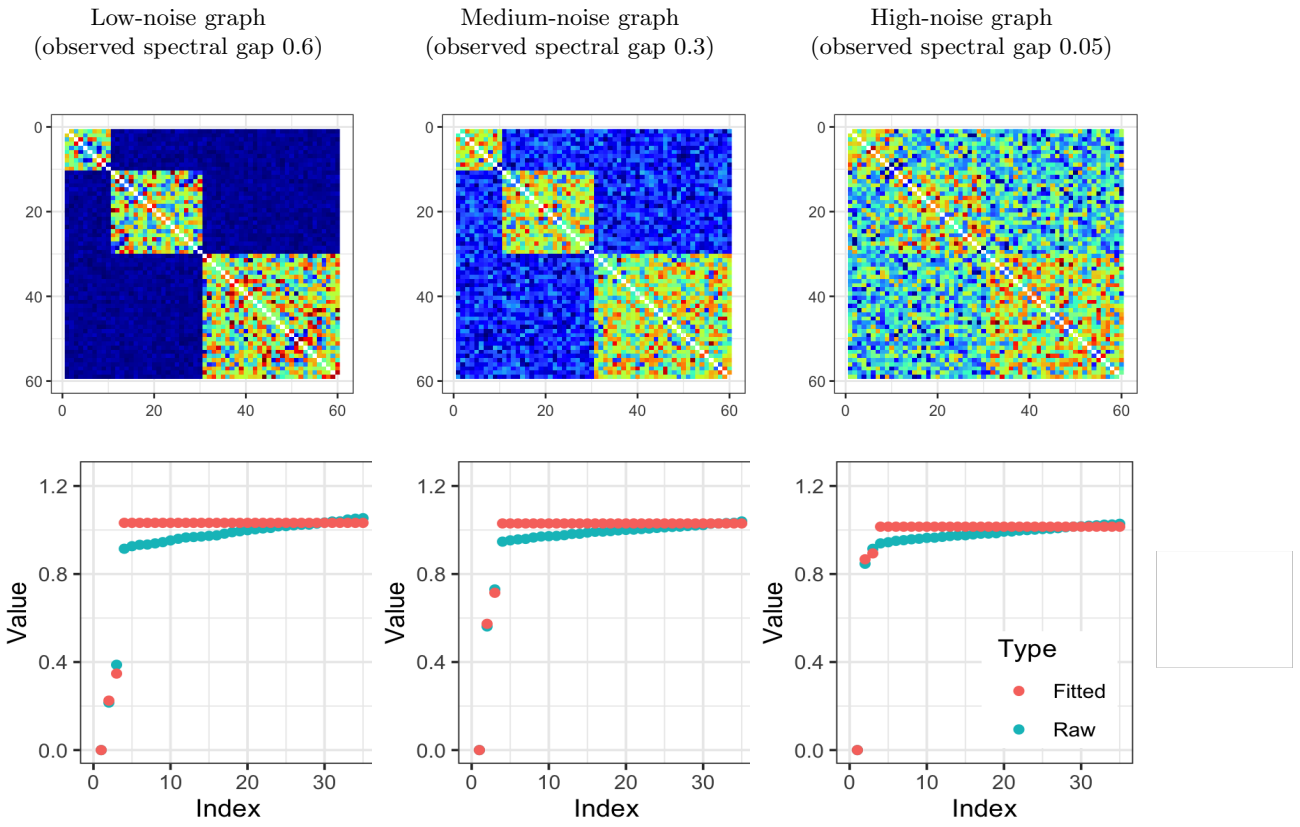


Figure 2: Three simulated graphs with different degree of noise, corresponding to different spectral gaps in the eigenvalues (for clarity, we show the first 35 eigenvalues out of 60). Comparing the eigenvalues produced by the direct decomposition of the raw Laplacian (cyan), and the ones by spiked Laplacian model (red), the latter has a clearly larger spectral gap between the third and fourth, corresponding to better separation between signal and noise.

The spiked Laplacian model has a "lifting effect" on the fourth eigenvalue (shown in red in Figure 2). This is due to the flat structure imposed, effectively replacing the fourth eigenvalue $\hat{\lambda}_4$ by $\theta \approx (\sum_{k=4}^{60} \hat{\lambda}_k)/57$ with $\hat{\lambda}_k > \hat{\lambda}_4$ for $k > 4$. Consequently, it leads to an increase in the spectral gap, compared to a direct eigendecomposition of the graph Laplacian (shown in cyan). Practically, this leads to improved accuracy in finding the community labels, as shown in Table 1. This phenomenon can be viewed as a result of rank regularization on the Laplacian matrix; Le et al. (2018) discussed similar effects under a slightly different regularization in spectral clustering.

Further, we would like to discuss the effect of sparsity on the eigenvalues of L. On the one hand, since L is the normalized Laplacian, it is scale-invariant in the magnitude of A; hence the above result will remain the same, even if A is multiplied to a small positive number. On the other hand, if the noise-free (unobserved) graph A_* becomes sparser, but the noise magnitude does not scale down with A_*, then it would impact the accuracy of

signal recovery, with a similar effect as in Subfigure 3 of Figure 2 (high-noise graph) — this is as expected since the signal-to-noise ratio decreases as the signal becomes sparse. To provide some numerical illustration, we simulate additional graphs in the appendix, except with the Bernoulli probability reduced from 0.5 to 0.3, 0.2 and 0.1, respectively.

5.2 Impact of Graph Size on Community Detection

n	100	300	500	1000
Spiked Laplacian Model	(0.84 ± 0.15)	(0.90 ± 0.04)	(0.86 ± 0.04)	(1 ± 0)
Stochastic Block Model	(0.65 ± 0.23)	(0.84 ± 0.09)	(0.87 ± 0.04)	(1 ± 0)
Bayesian SBM	(0.70 ± 0.14)	(0.83 ± 0.08)	(0.88 ± 0.05)	(1 ± 0)

Table 2: At small n, the spiked Laplacian model has higher accuracy in estimating the community labels. Mean $\pm$ standard deviation is reported based on 50 times of experiments. The higher the NMI, the higher the accuracy.

Next, we evaluate the effects of varying the number of vertices n on community detection. We adopt a similar 3-community setting as in the previous subsection, retaining the community size ratio as 1:2:3, and increase the total number of vertices. We calculate the normalized mutual information that compares the estimated community labels and the ground truth (details provided in the Appendix), using estimates produced by the spiked graph Laplacian model, the stochastic block model using the spectral clustering algorithm (Ng et al., 2002) and the Bayesian stochastic block model using a Gibbs sampler based on the model in van der Pas and van der Vaart (2018).

As shown in Table 2, for large $n \geq 500$, there are almost no differences in terms of the point estimate accuracy. However, at smaller vertex sizes n, the spiked graph Laplacian model exhibits clearly superior performance. Figure 3 shows the posterior distribution on the effective number of communities in one experiment at $n = 100$. It is evident in this experiment, that the point estimate $\hat{\kappa}$ matches the ground truth of 3 communities. We find similar results for larger n's as well. In 50 times of repeated experiments, $\hat{\kappa}$ matches with the ground truth for 98% of time.

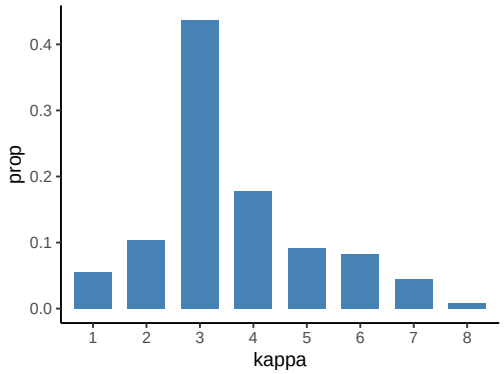
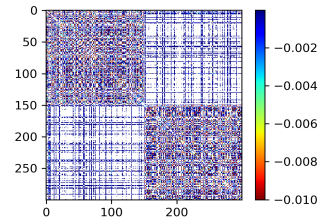
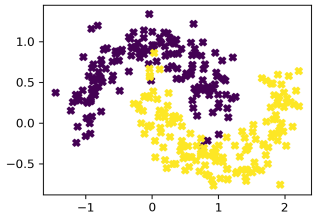


Figure 3: Posterior distribution of κ as the effective number of communities at $n = 100$.

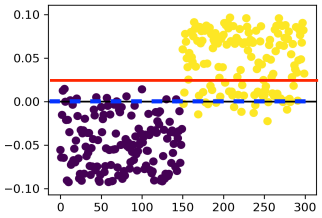
Next, we empirically show that the advantage for small n is attributable to more accurate uncertainty quantification. For a more intuitive illustration of this issue, we generate a 2-community graph using a latent position model — we first sample latent y_i's near two manifolds [Figure 4, Panel (b)], then compute the pairwise similarity between latent positions [$A_{i,j} = \exp(-10\|y_i - y_j\|_2)$], and use it as the edge weight. Clearly, most of the uncertainty is located in the center of the adjacency matrix, where the manifolds get close.



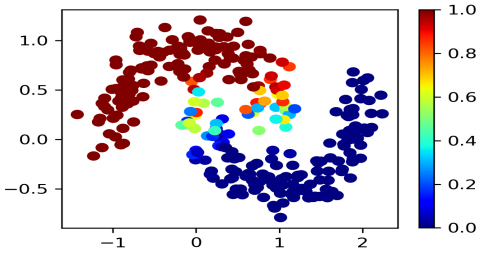
(a) Observed Laplacian L based on the simulated graph.



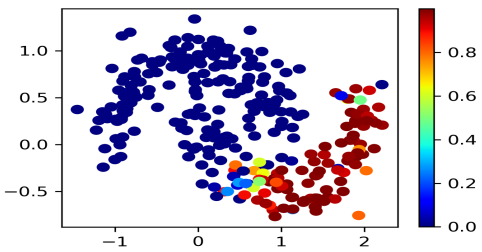
(b) The latent positions y_i used to generate the graph, colored by the true labels.



(c) $q_2(i)$ vs vertex index. Two communities are classified by the K-means (red) or sign-based algorithm (blue).



(d) Spiked Laplacian model correctly estimated the uncertainty $\text{pr}(c_i = 1)$, based on sign-partitioning of *each* posterior sample q_2.



(e) Applying Gaussian mixture model on *only one* sample of $q_2(i)$ (as in the stochastic block model) under-estimates the uncertainty $\text{pr}(c_i = 1)$.

Figure 4: Illustration of uncertainty quantification by the spiked graph Laplacian model.

Panel (c) plots one sample of $\vec{q}_2$. The sign-based partition used by the spiked graph Laplacian model has a default decision boundary at the zero line (in blue). Applying this bi-partitioning on each posterior sample of $\vec{q}_2$, it leads to an accurate uncertainty quantification [Panel (d)]. Comparatively, in the estimation of stochastic block model, one applies K-means or a Gaussian mixture model on *one* sample of $\vec{q}_2$ (based on the direct eigendecomposition of L), which could result in a severe underestimation of the uncertainty, as shown in Panel (e).

5.3 Accommodating Heterogeneity in a Collection of Graphs

In this experiment, we deal with multiple graphs comprising of 300 vertices each, whose adjacency matrices exhibit heterogeneity. We first generate a set of five possible community-partition patterns, each represented by a binary matrix (denoted by $W^{(l)}$) of size 300×6 for

$l = 1, \ldots, 5$; each row has one 1 and five 0's, encoding the ground truth of the community labels in $1, \cdots, 6$. To generate a graph, we randomly draw one of five patterns as $\tilde{W}^{(s)}$ and a non-negative random vector $\tilde{\Lambda}$, producing its adjacency matrix by $A^{(s)} = \tilde{W}^{(s)}\tilde{\Lambda}\tilde{W}^{(s)\prime} + \tilde{\mathcal{E}}^{(s)}$, with $\tilde{\mathcal{E}}^{(s)}$ being a Gaussian noise matrix and $\tilde{e}^{(s)}_{i,j} = \tilde{e}^{(s)}_{j,i} \sim N(0,1)$, for $s = 1, \ldots, 500$.

We compare the performance of the proposed model against several popular alternatives: (1) simple averaging of all graphs followed by the use of a stochastic block model, (2) co-regularized stochastic block model/spectral clustering (Kumar et al., 2011), (3) clustering the graphs into five groups (patterns), and applying the stochastic block model in each group, (4) independent stochastic block model for each graph. The first two competitors produce only one partitioning, while the latter two accommodate the heterogeneity.

We compute two benchmark scores: the normalized mutual information (NMI), reflecting the similarity between the estimated community labels to the ground truth in each graph; and the Root Mean Squared Error between the individual $L^{(s)}$ and the smoothed $\hat{L}^{(s)}$, as the goodness of fit criterion.

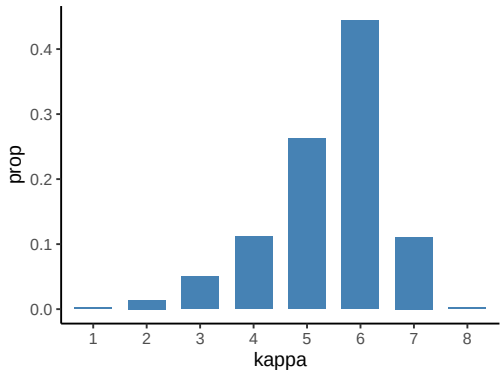
Benchmark Scores	NMI (higher is better)	RMSE ($\times 10^{-3}$, lower is better)
Spiked Laplacian Graphs	0.85 ± 0.04	1.9 ± 0.2
Average+SBM	0.21 ± 0.15	9.2 ± 2.5
Co-regularized SBM	0.25 ± 0.11	10.2 ± 4.5
Clustering Graphs + SBMs	0.67 ± 0.24	5.5 ± 1.5
Individual SBMs	0.45 ± 0.13	1.2 ± 0.2

Table 3: Benchmark of the fitting models to a population of heterogeneous graphs. Mean $\pm$ standard deviation is reported based on 50 times of experiments. When computing the RMSE, for the Spiked Laplacian Graphs, we obtain $\hat{L}^{(s)}$ from the spiked representation taking individual $\kappa^{(s)}$ as the truncated dimension, averaging over the posterior sample; for the other four, we define $\hat{L}^{(s)}$ as the truncated spectral representation $\hat{Q}\hat{\Lambda}\hat{Q}$ with $(\hat{Q}, \hat{\Lambda})$ corresponding to the top 6 dimensions (as the ground truth dimension for data generation).

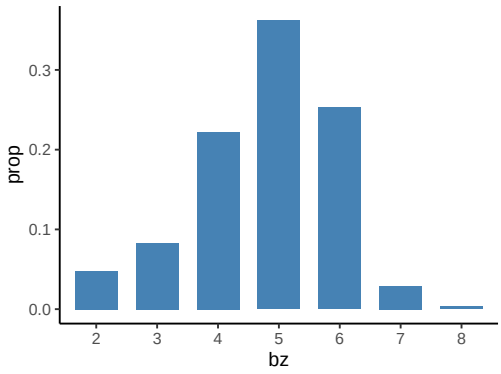
As shown in Table 3, our proposed model has the highest accuracy in estimating the community labels, followed by the two-stage estimator that clusters the graphs first and then partitions the vertices via the stochastic block model. The performance of individual stochastic block models is significantly inferior, likely due to the fact that they do not borrow information among graphs, and the number of vertices per graph is relatively small. For the goodness-of-fit measure, the individual stochastic block models achieve the best score due to their higher flexibility. The proposed model xhibits a slightly larger error, but is significantly lower than the remaining competitors.

Figure 5 shows the posterior distributions on the effective numbers of communities ($\kappa^{(s)}$) in one graph and the numbers of distinct patterns (b_z) in one experiment. In this experiment, both the point estimates $\hat{\kappa}$ and $\hat{b}_z$ match the ground truth. In 50 times of repeated experiments, $\hat{\kappa}$ matches the ground truth for 92% of time, $\hat{b}_z$ for 72% of time.

Despite the very good empirical results, we want to caution that one should be careful on choosing those two dimensions, and we provide further guidance at the end of this article.



(a) Posterior distribution of $\kappa^{(s)}$ as the effective number of communities for one graph.

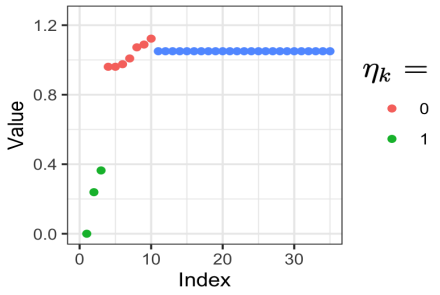


(b) Posterior distribution of b_z as the effective number of unique values in z_s (distinct patterns).

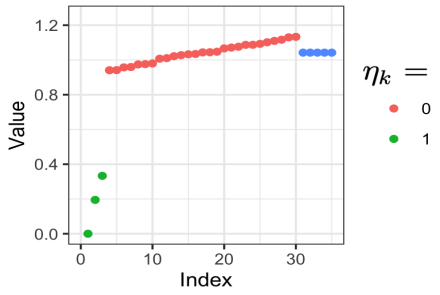
Figure 5: Posterior distributions on the numbers of communities and distinct patterns in one experiment.

5.4 Robustness to Over-specified T

Lastly, we examine if the proposed model can handle an over-specified T, when it is larger than necessary. We focus on the following two issues: (i) whether the posterior sample of η can successfully identify redundant λ_k's; (ii) whether a misspecified T affects the estimation of the first few eigenvalues.



(a) Eigenvalues estimated with $T = 10$.



(b) Eigenvalues estimated with $T = 30$.

Figure 6: Simulation showing the first few small eigenvalues are almost unaffected by an overly large T, and the variable η_k successfully identifies the redundant λ_k.

We use the same single graph setup to generate graphs with three communities, except we now set $T = 10$ and $T = 30$. As shown in Figure 6, the posterior distribution of η_k successfully finds all unnecessary λ_k's, as indicated by $\eta_k = 0$. Further, there is almost

no difference in the estimates of the first few eigenvalues $\lambda_1, \lambda_2, \lambda_3$. On the other hand, we should clarify that if the spectral gap is small, η_k will more likely be assigned to 0; this is an expected behavior indicating there is a large loss if we still want to partition the graph.

6. Data Application: Characterizing Heterogeneity in a Human Working Memory Study

We employ the proposed spiked graph Laplacian model on data obtained from a neuroscience study on working memory, focusing on human brain functional connectivity (Hu et al., 2019). The study involved 1,329 brain scans, wherein each subject in the study was asked to do the Sternberg verbal working memory task, which involved memorizing a list of six numbers, followed by a memory retrieval task that requires the subject to answer if a number was among the six shown earlier. Electroencephalogram (EEG) signals were obtained from 128 electrode channels placed over each subject's head, and subsequently, a 128×128 connectivity network is estimated during the retrieval task period using absolute Pearson correlation. Each network has weighted edges taking values in the $[0, 1]$ interval.

Figure 7 depicts the adjacency matrices of three subjects for the memory retrieval task, and the presence of heterogeneity is apparent. It can be seen that memory-related connectivity can exhibit different levels of concentration in the front or back of the head [Panels (c) or (d), with spatial coordinates, plotted in Figure 9 (a)], or, they are more localized in smaller regions [Panel (e)].

We apply the spiked graph Laplacian model on this data set and the results obtained are based on an MCMC run of $30,000$ steps, with the first $10,000$ used as the burn-in period. The majority of the samples from the posterior distribution contain six distinct $U^{(l)}$'s in the clustered eigenmatrix values. Figure 8 depicts the three corresponding to the raw $A^{(s)}$ shown in the previous Figure, obtained from the fitted Laplacian matrices. The remaining three seem to correspond to smaller variations and are shown in the Appendix. The proportions for these six patterns are $25.6\%, 24.1\%, 16.1\%, 14.7\%, 15.2\%$ and 4.3%, as estimated in the posterior mean of allocation z_s.

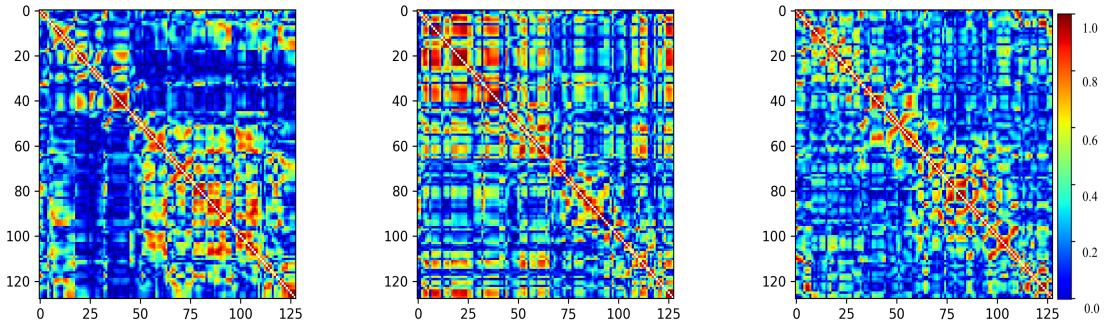


Figure 7: Brain functional connectivity adjacency matrices of three individuals undertaking the memory retrieval task. A significant level of heterogeneity can be observed.

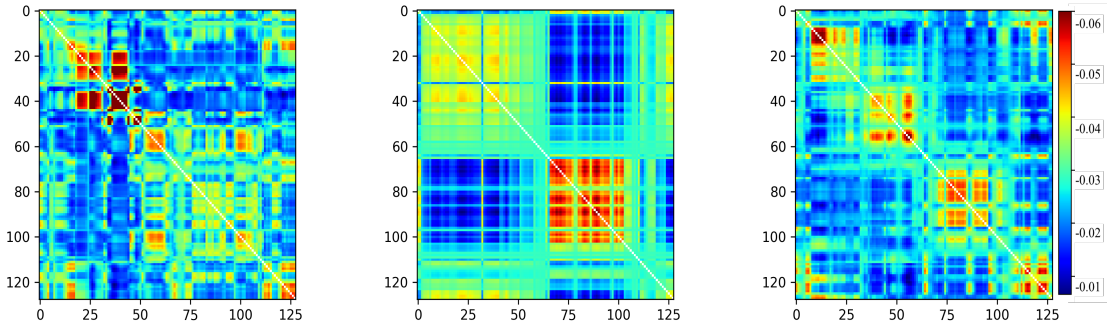
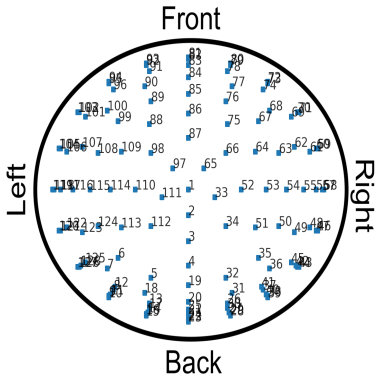
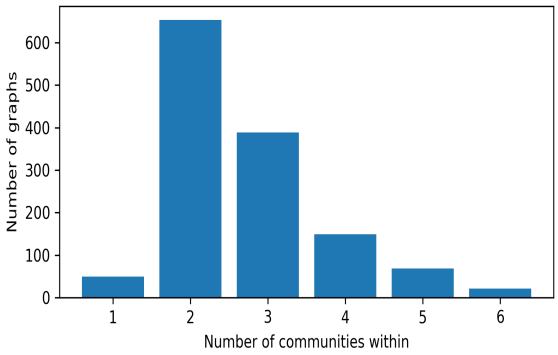


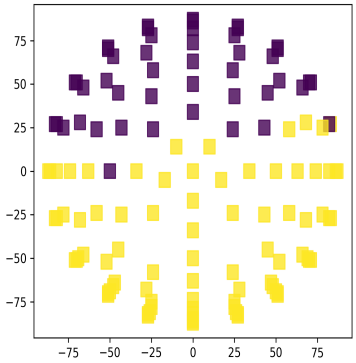
Figure 8: Fitted Laplacian shows the structure underneath each raw connectivity matrix.



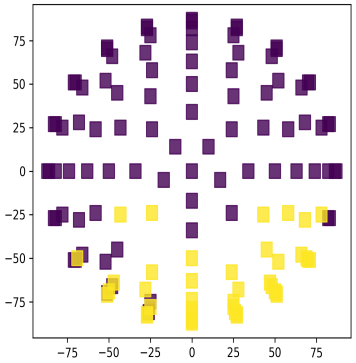
(a) Coordinates of the EEG sensors, viewed from the top of the head.



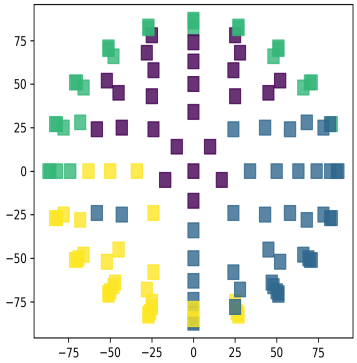
(b) Histogram of the number of communities in all subjects.



(c) The subject has two communities, with the larger one near the back of the head.



(d) The subject has two communities, with the larger one near the front of the head.



(e) The subject has four communities: outer-front, mid-front, left-back, right-back.

Figure 9: Community structure for each brain scan from multiple subjects in the working memory study.

We then evaluate the community structures in each network. As shown in Figure 9(a), the model discovers $1 \sim 6$ communities from these graphs, as estimated by $\kappa^{(s)}$. To gain insight into the scientific implications, we plot the community labels mapped to the spatial coordinates. Panel (c) and (d) show that most of the networks contain only two distinct communities, although the division can be quite different in the dominating area either in the front or in the back of the head. Panel (e) shows a very distinct pattern with four communities, partitioned as the outer-front, mid-front, left-back, right-back regions of the head.

7. Discussion

In this paper, we propose a probabilistic graph model based on the Laplacian, allowing us to exploit concepts and results rom spectral graph theory to conduct flexible community detection in a population of heterogeneous graphs. Our model can be considered as a

general method to introduce Bayesian tools into the spectral graph framework. There are several extensions worth exploring in future work. First, if the goal is to generate a new graph with binary $A_{i,j}$, such as in link prediction, then it could adopt a Bernoulli distribution associated with a canonical link. Second, if those graphs have some known covariance structure, such as is the case of repeated measurements or temporal effects, then it could take an alternative distribution on the eigenmatrix or eigenvalues to incorporate those structures. Third, for large graphs, it is of interest to consider θ not as a single constant, but as a step function.

Lastly, a recent discovery is that the Dirichlet process mixture model, although it enjoys posterior consistency in density estimation (Ghosal et al., 1999), can lead to inconsistent estimates of the number of clusters (Miller and Harrison, 2013, 2014). Therefore, although we did obtain interpretable results of finding 6 patterns and small numbers of communities in our application, to be rigorous, we cautiously believe recovering a "ground-truth" number of clusters/patterns is still a non-trivial task. As alternatives, one may replace the Dirichlet process mixture prior with a mixture of finite mixtures prior distributions (Miller and Harrison, 2018), or an infinite mixture of quasi-Bernoulli stick-breaking prior distributions (Zeng et al., 2022), both of which have shown correct asymptotic behavior in simpler cases. However, for the task of clustering eigenvectors, significant work is needed to verify if consistency holds on the number of clusters, as it requires checking for a completely correct model specification and identifiability of the parameters.

References

Emmanuel Abbe. Community Detection and Stochastic Block Models: Recent Developments. *Journal of Machine Learning Research*, 18(1):6446–6531, 2017.

Charu C Aggarwal. An Introduction to Social Network Data Analytics. In *Social Network Data Analytics*, pages 1–15. Springer, 2011.

Edoardo M Airoldi, David M Blei, Stephen E Fienberg, and Eric P Xing. Mixed Membership Stochastic Blockmodels. *Journal of Machine Learning Research*, 9:1981–2014, 2008.

Arash A Amini, Aiyou Chen, Peter J Bickel, and Elizaveta Levina. Pseudo-Likelihood Methods for Community Detection in Large Sparse Networks. *The Annals of Statistics*, 41(4):2097–2122, 2013.

Sanjeev Arora, Satish Rao, and Umesh Vazirani. Expander Flows, Geometric Embeddings and Graph Partitioning. *Journal of the Association for Computing Machinery*, 56(2): 1–37, 2009.

Anirban Banerjee and Jürgen Jost. On the Spectrum of the Normalized Graph Laplacian. *Linear Algebra and Its Applications*, 428(11-12):3015–3022, 2008.

Abhishek Bhattacharya and David B Dunson. Nonparametric Bayesian Density Estimation on Manifolds With Applications to Planar Shapes. *Biometrika*, 97(4):851–865, 2010.

Diana Cai, Trevor Campbell, and Tamara Broderick. Edge-Exchangeable Graphs and Sparsity. In *Advances in Neural Information Processing Systems*, pages 4249–4257, 2016.

François Caron and Emily B Fox. Sparse Graphs Using Exchangeable Random Measures. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(5):1295–1366, 2017.

Fan RK Chung and Fan Chung Graham. *Spectral Graph Theory*. Number 92. American Mathematical Soc., 1997.

David L Donoho, Matan Gavish, and Iain M Johnstone. Optimal Shrinkage of Eigenvalues in the Spiked Covariance Model. *The Annals of Statistics*, 46(4):1742, 2018.

David B. Dunson, Hau-Tieng Wu, and Nan Wu. Graph based gaussian processes on restricted domains. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 84(2):414–439. doi: https://doi.org/10.1111/rssb.12486.

Daniele Durante, David B Dunson, and Joshua T Vogelstein. Nonparametric Bayes Modeling of Populations of Networks. *Journal of the American Statistical Association*, 112 (520):1516–1530, 2017.

Miroslav Fiedler. Laplacian of Graphs and Algebraic Connectivity. *Banach Center Publications*, 25(1):57–70, 1989.

Santo Fortunato. Community Detection in Graphs. *Physics Reports*, 486(3-5):75–174, 2010.

Shmuel Friedland and Reinhard Nabben. On Cheeger-Type Inequalities for Weighted Graphs. *Journal of Graph Theory*, 41(1):1–17, 2002.

Junxian Geng, Anirban Bhattacharya, and Debdeep Pati. Probabilistic Community Detection With Unknown Number of Communities. *Journal of the American Statistical Association*, 114(526):893–905, 2019.

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

Subhashis Ghosal, Jayanta K Ghosh, and RV Ramamoorthi. Posterior Consistency of Dirichlet Mixtures in Density Estimation. *The Annals of Statistics*, 27(1):143–158, 1999.

Matthias Hein, Jean-Yves Audibert, and Ulrike von Luxburg. Graph Laplacians and Their Convergence on Random Neighborhood Graphs. *Journal of Machine Learning Research*, 8:1325–1368, 2007.

Peter D Hoff. Simulation of the matrix Bingham–von Mises–Fisher Distribution, With Applications to Multivariate and Relational Data. *Journal of Computational and Graphical Statistics*, 18(2):438–456, January 2009.

Peter D Hoff, Adrian E Raftery, and Mark S Handcock. Latent Space Approaches to Social Network Analysis. *Journal of the American Statistical Association*, 97(460):1090–1098, 2002.

Zhenhong Hu, Christopher M Barkley, Susan E Marino, Chao Wang, Abhijit Rajan, Ke Bo, Immanuel Babu Henry Samuel, and Mingzhou Ding. Working Memory Capacity Is Negatively Associated With Memory Load Modulation of Alpha Oscillations in Retention of Verbal Working Memory. *Journal of Cognitive Neuroscience*, pages 1–13, 2019.

Muhammad Aqib Javed, Muhammad Shahzad Younis, Siddique Latif, Junaid Qadir, and Adeel Baig. Community Detection in Networks: A Multidisciplinary Review. *Journal of Network and Computer Applications*, 108:87–111, 2018.

Brian Karrer and Mark EJ Newman. Stochastic Blockmodels and Community Structure in Networks. *Physical Review E*, 83(1):016107, 2011.

Rohit Khandekar, Satish Rao, and Umesh Vazirani. Graph Partitioning Using Single Commodity Flows. *Journal of the Association for Computing Machinery*, 56(4):1–15, 2009.

Alisa Kirichenko, Harry van Zanten, et al. Estimating a Smooth Function on a Large Graph by Bayesian Laplacian Regularisation. *Electronic Journal of Statistics*, 11(1): 891–915, 2017.

Abhishek Kumar, Piyush Rai, and Hal Daume. Co-Regularized Multi-View Spectral Clustering. In *Advances in Neural Information Processing Systems*, pages 1413–1421, 2011.

Can M Le, Elizaveta Levina, and Roman Vershynin. Concentration of Random Graphs and Application to Community Detection. In *Proceedings of the International Congress of Mathematicians: Rio de Janeiro 2018*, pages 2925–2943. World Scientific, 2018.

Tom Leighton and Satish Rao. Multicommodity Max-Flow Min-Cut Theorems and Their Use in Designing Approximation Algorithms. *Journal of the Association for Computing Machinery*, 46(6):787–832, 1999.

Lizhen Lin, Vinayak Rao, and David Dunson. Bayesian Nonparametric Inference on the Stiefel Manifold. *Statistica Sinica*, pages 535–553, 2017.

Fei Liu, Sounak Chakraborty, Fan Li, Yan Liu, Aurelie C Lozano, et al. Bayesian Regularization via Graph Laplacian. *Bayesian Analysis*, 9(2):449–474, 2014.

Anand Louis, Prasad Raghavendra, Prasad Tetali, and Santosh Vempala. Algorithmic Extensions of Cheeger's Inequality to Higher Eigenvalues and Partitions. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 315–326. Springer, 2011.

Aaron F McDaid, Thomas Brendan Murphy, Nial Friel, and Neil J Hurley. Improved Bayesian Inference for the Stochastic Block Model With Application to Large Networks. *Computational Statistics & Data Analysis*, 60:12–31, 2013.

Jeffrey W Miller and Matthew T Harrison. A Simple Example of Dirichlet Process Mixture Inconsistency for the Number of Components. *Advances in Neural Information Processing Systems*, 26, 2013.

Jeffrey W Miller and Matthew T Harrison. Inconsistency of Pitman-Yor Process Mixtures for the Number of Components. *Journal of Machine Learning Research*, 15(1):3333–3370, 2014.

Jeffrey W Miller and Matthew T Harrison. Mixture Models With a Prior on the Number of Components. *Journal of the American Statistical Association*, 113(521):340–356, 2018.

Kyle J Minch, Tige R Rustad, Eliza JR Peterson, Jessica Winkler, David J Reiss, Shuyi Ma, Mark Hickey, William Brabant, Bob Morrison, and Serdar Turkarslan. The DNA-binding Network of Mycobacterium tuberculosis. *Nature Communications*, 6:5829, 2015.

Bojan Mohar. Isoperimetric Numbers of Graphs. *Journal of Combinatorial Theory, Series B*, 47(3):274–291, 1989.

Peter J Mucha, Thomas Richardson, Kevin Macon, Mason A Porter, and Jukka-Pekka Onnela. Community Structure in Time-Dependent, Multiscale, and Multiplex Networks. *Science*, 328(5980):876–878, 2010.

Soumendu Sundar Mukherjee, Purnamrita Sarkar, and Lizhen Lin. On Clustering Network-Valued Data. In *Advances in Neural Information Processing Systems*, pages 7071–7081, 2017.

Andrew Y Ng, Michael I Jordan, and Yair Weiss. On Spectral Clustering: Analysis and an Algorithm. In *Advances in Neural Information Processing Systems*, pages 849–856, 2002.

Krzysztof Nowicki and Tom A B Snijders. Estimation and Prediction for Stochastic Blockstructures. *Journal of the American Statistical Association*, 96(455):1077–1087, 2001.

Symeon Papadopoulos, Yiannis Kompatsiaris, Athena Vakali, and Ploutarchos Spyridonos. Community Detection in Social Media. *Data Mining and Knowledge Discovery*, 24(3): 515–554, 2012.

Karl Rohe, Sourav Chatterjee, and Bin Yu. Spectral Clustering and the High-Dimensional Stochastic Blockmodel. *The Annals of Statistics*, 39(4):1878–1915, 2011.

Xilin Shen, Fuyuze Tokoglu, Xenios Papademetris, and R Todd Constable. Groupwise Whole-Brain Parcellation From Resting-State fMRI Data for Network Node Identification. *Neuroimage*, 82:403–415, 2013.

Jianbo Shi and Jitendra Malik. Normalized Cuts and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

Somayeh Sojoudi. Equivalence of Graphical Lasso and Thresholding for Sparse Graphs. *Journal of Machine Learning Research*, 17(1):3943–3963, 2016.

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

SL van der Pas and AW van der Vaart. Bayesian Community Detection. *Bayesian Analysis*, 13(3):767–796, 2018.

Ulrike Von Luxburg, Mikhail Belkin, and Olivier Bousquet. Consistency of Spectral Clustering. *The Annals of Statistics*, pages 555–586, 2008.

Martin J Wainwright. *High-Dimensional Statistics: a Non-asymptotic Viewpoint*, volume 48. Cambridge University Press, 2019.

Sinead A Williamson. Nonparametric Network Models for Link Prediction. *Journal of Machine Learning Research*, 17(1):7102–7121, 2016.

Cheng Zeng, Jeffrey W Miller, and Leo L Duan. Consistent Model-based Clustering: using the Quasi-Bernoulli Stick-Breaking Process. *arXiv preprint arXiv:2008.09938*, 2022.

Li Zhang, Yuan Li, and Ramakant Nevatia. Global Data Association for Multi-Object Tracking Using Network Flows. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

Yichuan Zhang, Zoubin Ghahramani, Amos J Storkey, and Charles A Sutton. Continuous Relaxations for Discrete Hamiltonian Monte Carlo. In *Advances in Neural Information Processing Systems*, pages 3194–3202, 2012.

Appendix:

Proof of Theorem 1

Proof The bounds on the eigenvalues of the Laplacian are given and discussed in Chung and Graham (1997). For the first eigenvector we have

$$\mu_L \vec{d}_*^{1/2} = D_*^{-1/2}(D_* - A_*)D_*^{-1/2}\vec{d}_*^{1/2} = D_*^{-1/2}(D_* - A_*)\vec{1} = \vec{0}.$$

■

Proof of Theorem 6

Proof For simplicity, we omit $.^{(s)}$ in the proof and use σ_e for σ_{e0}. The proof consists of the following four parts:

1. An application of the Davis-Kahan Theorem

Let $E = \tilde{L} - L$, using Theorem 2 in citepyu2014useful with $r = 1$ and $s = k$, we obtain

$$\begin{aligned}\|Q_0 - QO\|_F &\leq \frac{2^{3/2}\min(k^{1/2}\|E\|_{op}, \|E\|_F)}{\lambda_{k+1} - \lambda_k} \\ &\leq \frac{2^{3/2}(k^{1/2}\|E\|_{op})}{\lambda_{k+1} - \lambda_k}\end{aligned}$$

where $\|E\|_{op}$ denotes the operator norm ($\|E\|_{op} = \sup_{\|x\|=1}\|Ex\|$).

2. Discretizing $\mathbb{S}^{n-1} = \{x : \|x\| = 1\}$ using a maximal ϵ-net:

Following Tao (2012), let $N_\epsilon \subset \mathbb{S}^{n-1}$ be an ϵ-net with $\epsilon \in (0, 1)$, such that for any two $x \in \mathcal{N}_\epsilon, x' \in \mathcal{N}_\epsilon, \|x - x'\| \geq \epsilon$. Maximizing over the number of included points in $\mathbb{S}^{n-1}$, we obtain a maximal ϵ-net N_ϵ^0. Clearly, the balls with centers $x \in N_\epsilon^0$ and radius $\epsilon/2$ are disjoint, and all covered by a large ball centered at the origin with radius $1 + \epsilon/2$, hence

$$|\mathcal{N}_\epsilon^0| \leq (\frac{\epsilon/2 + 1}{\epsilon/2})^n = (\frac{\epsilon + 2}{\epsilon})^n.$$

On the other hand, for any $y \in \mathbb{S}^n$, there is at least one $x \in \mathcal{N}_\epsilon^0 : \|x - y\| \leq \epsilon$, otherwise y can be added to the net, contradicting the maximal condition.

Choosing $y \in \mathbb{S}^n$ that attains $\|Ey\| = \|E\|_{op}$, and its associated $x \in \mathcal{N}_\epsilon^0 : \|x - y\| \leq \epsilon$

$$\|E\|_{op} - \|Ex\| = \|Ey\| - \|Ex\| \leq \|E(y - x)\| \leq \|E\|_{op}\epsilon,$$

by an application of the triangle inequality and $f(x) = \|Ex\|$ is $\|E\|_{op}$-Lipschitz.

Therefore, $\|E\|_{op} \geq t$ implies at least one $x \in \mathcal{N}_\epsilon^0 : \|Ex\| \geq (1 - \epsilon)t$.

$$\begin{aligned}\mathrm{pr}(\|E\|_{op} \geq t) &\leq \mathrm{pr}\Big(\bigcup_{x \in \mathcal{N}_\epsilon^0} \|Ex\| \geq (1 - \epsilon)t\Big) \\ &\leq |\mathcal{N}_\epsilon^0| \mathrm{pr}\Big(\|Ex\| \geq (1 - \epsilon)t, \text{ where } x \in \mathbb{S}^n\Big)\end{aligned}$$

where the last inequality follows from the union bound.

3. Concentration inequality for $\|Ex\|$

Since E is symmetric, let $E = E_U + E_L$, with E_U being the upper triangular portion including the diagonal and E_L the lower triangular portion. We first use B to represent either E_U or E_L. Let B be an $n \times n$ matrix comprising of $b_{i,j}$ independent and σ_e^2-sub-Gaussian elements. Then, for each element Bx

$$\begin{aligned}\mathbb{E} \exp\{tB_j'x\} &= \mathbb{E} \exp\{t \sum_{k=1}^n x_k b_{j,k}\} \\ &= \prod_{k=1}^n \mathbb{E} \exp\{t x_k b_{j,k}\} \\ &\leq \prod_{k=1}^n \exp\{t^2 \sigma_e^2 x_k^2/2\} \\ &= \exp\{t^2 \sigma_e^2/2\}\end{aligned}$$

where the inequality is due to the sub-Gaussian assumption, and the last equality due to $\|x\| = 1$. Therefore, each $Z_j = B_j x$ is sub-Gaussian as well. By a result in Wainwright (2019), this is equivalent to

$$\mathbb{E} \exp(\frac{\kappa Z_j^2}{2\sigma_e^2}) \leq (1 - \kappa)^{-1/2} \tag{14}$$

for all $\kappa \in (0, 1)$.

We have

$$\|Ex\|^2 = \|E_U x + E_L x\|^2 \leq (\|E_U x\| + \|E_L x\|)^2 \leq 2(\|E_U x\|^2 + \|E_L x\|^2)$$

By the Cauchy–Schwarz inequality, we obtain

$$\begin{aligned}\mathbb{E} \exp(\frac{\kappa \|Ex\|^2}{2\sigma_e^2}) &\leq \mathbb{E} \exp(\frac{2\kappa[\|E_U x\|^2 + \|E_L x\|^2]}{2\sigma_e^2}) \\ &\leq \sqrt{\mathbb{E} \exp(\frac{4\kappa \|E_U x\|^2}{2\sigma_e^2}) \mathbb{E} \exp(\frac{4\kappa \|E_L x\|^2}{2\sigma_e^2})}\end{aligned}$$

Since E_U and E_L comprise of sub-Gaussian elements and zeros, they are also sub-Gaussian with σ_e^2; then, multiplying (14) over $j = 1, \ldots, n$ for each matrix, we get

$$\mathbb{E} \exp(\frac{\kappa \|Ex\|^2}{2\sigma_e^2}) \leq \sqrt{(1 - 4\kappa)^{-n/2}(1 - 4\kappa)^{-n/2}} = (1 - 4\kappa)^{-n/2}$$

where $\kappa \in (0, 1/4)$. Using Markov's inequality

$$\mathrm{pr}(\|Ex\| \geq t) = \mathrm{pr}\Big(\exp(\frac{\kappa \|Ex\|^2}{2\sigma_e^2}) \geq \exp(\frac{\kappa t^2}{2\sigma_e^2})\Big) \leq (1 - 4\kappa)^{-n/2} \exp(-\frac{\kappa t^2}{2\sigma_e^2}).$$

4. Combining results to obtain a concentration inequality

Therefore,

$$\text{pr}(\|E\|_{op} \geq t) \leq (\frac{\epsilon+2}{\epsilon})^n (1-4\kappa)^{-n/2} \exp(-\frac{\kappa(1-\epsilon)^2 t^2}{2\sigma_e^2})$$

Letting $t = c_1\sqrt{n}\sigma_e$, $\kappa = 1/8$ and $\epsilon = 1/2$, we have

$$\text{pr}(\|E\|_{op} \geq c_1\sqrt{n}\sigma_e) \leq \exp[-\{c_1^2/64 - \log(5\sqrt{2})\}n] \equiv \delta$$

Therefore,

$$\|Q - \hat{Q}\hat{O}\|_F \leq \frac{2^{3/2}k^{1/2}c_1\sqrt{n}\sigma_e}{\lambda_{k+1} - \lambda_k}$$

with probability greater than $1-\delta$.

■

Proof of Theorem 8

Proof

For simplicity, we omit $.^{(s)}$ for now and let $D = \Lambda^\dagger$ and $B = \Omega$. Without loss of generality, we assume the diagonal of B are ordered $0 = b_1 < b_2 \leq \ldots \leq b_n$; and we have fixed $d_1 = 0$ and $d_2, \ldots, d_n > 0$. The parameter $Q^\dagger$ follows a matrix Bingham distribution truncated to $\mathcal{V}^*$

$$\tilde{g}(Q^\dagger; W, D, B, \sigma_e^2)\Pi(dQ^\dagger) = Z^{-1}(\sigma_e^2, D, B)\text{etr}\Big\{\frac{1}{2\sigma_e^2}DQ^{\dagger\prime}WBW'Q^\dagger\Big\}\Pi(\text{d}Q^\dagger)$$

where Z is a normalizing constant.

We utilize the result of Bhattacharya and Dunson (2010) to establish weak consistency of the posterior density estimation. There are three sufficient conditions to check:

(1) The kernel $\tilde{g}(.)$ is continuous in all of its arguments.

(2) The set $\{F_0\} \times D_\epsilon^0$ intersects the parameter support of $Q^\dagger$ and σ_e^2, where $\mathcal{D}_\epsilon^0$ is the interior of $\mathcal{D}_\epsilon$, a compact neighborhood for σ_e^2.

(3) For any continuous f, there is a $\mathcal{D}_\epsilon$ for σ_e^2, such that

$$\Delta = \sup_{W\in\mathcal{V}^*, \sigma_e^2\in\mathcal{D}_\epsilon} \Big\| f(W) - \int \tilde{g}(Q^\dagger; W, D, B, \sigma_e^2) f(Q^\dagger)\Pi(\text{d}Q^\dagger)\Big\| \leq \epsilon.$$

The first two conditions are straightforward to check (see Lin et al. (2017) for similar derivation). We will focus on verifying (3). Note the Frobenius distance between two orthonormal matrices

$$\text{dist}(W, Q^\dagger)^2 = 2n - 2\text{tr}(W'Q^\dagger) = 2\sum_{j=1}^n (1 - g_{j,j}),$$

where $g_{i,j}$ is the element of $G = W'Q^\dagger$, where $|g_{j,j}| \leq 1$ due to orthonormality of G. Let $(1-g_{j,j}) = s_{j,j}\sigma_e$, with $s_{j,j} \in [0, 2/\sigma_e]$, then $\sum_{j=1}^n (1-g_{j,j}) = \sum_{j=1}^n s_{j,j}\sigma_e$. As $\sigma_e \to 0$, $\text{dist}(W, Q^\dagger) \to 0$ for any fixed $(s_{1,1}, \ldots, s_{n,n})$. By the continuity of f and compactness of Stiefel manifold, as $\sigma_e \to 0$

$$\sup_{W\in\mathcal{V}^*} \Big\| f(W) - f(Q^\dagger)\Big\| \to 0. \tag{15}$$

Now

$$\Delta \leq Z^{-1}(\sigma_e^2, D, B) \int \sup_{W \in \mathcal{V}^*} \left\| f(W) - f(Q^\dagger) \right\| \mathrm{etr}\left\{ \frac{1}{2\sigma_e^2} DQ^{*\prime}[WBW^\prime]Q^\dagger \right\} \Pi(\mathrm{d}Q^\dagger)$$
$$= Z^{-1}(\sigma_e^2, D, B) \int \sup_{W \in \mathcal{V}^*} \left\| f(W) - f(WG) \right\| \mathrm{etr}\left\{ \frac{1}{2\sigma_e^2} DG^\prime BG \right\} \Pi(\mathrm{d}G)$$

where the second line is due to the invariant volume of rotation via W. It can be verified that

$$\begin{aligned}
\mathrm{tr}(DG^\prime BG) &= \sum_{i=1}^n \sum_{j=1}^n b_i d_j g_{i,j}^2 \\
&= \sum_{j=1}^n b_j d_j - \sum_{j=1}^n b_j d_j (1 - g_{j,j}^2) + \sum_{j=1}^n \sum_{i \neq j} b_i d_j g_{i,j}^2 \\
&\leq \sum_{j=1}^n b_j d_j - \sum_{j=1}^n b_j d_j (1 - g_{j,j}^2) + \sum_{j=1}^n d_j b_n \sum_{i \neq j} g_{i,j}^2 \\
&= \sum_{j=1}^n b_j d_j - \sum_{j=1}^n b_j d_j (1 - g_{j,j}^2) + \sum_{j=1}^n d_j b_n (1 - g_{j,j}^2) \\
&= \sum_{j=1}^n b_j d_j + \sum_{j=1}^n d_j (b_n - b_j)(1 - g_{j,j}^2) \\
&= \sum_{j=1}^n b_n d_j - \sum_{j=1}^n d_j (b_n - b_j) g_{j,j}^2,
\end{aligned}$$

where the first inequality is due to $d_j \geq 0$ and $b_n \geq b_i$ for all i; the fourth line is due to the 1 unit norm for each column of G.

Applying one-to-one transformation $T : \mathcal{V}^* \to \mathcal{S}$, $T(G) = \{s_{i,j} = g_{i,j} \text{ for } i \neq j, s_{j,j} = (1 - g_{j,j})/\sigma_e\}_{i,j}$, denote the transformed G matrix by G_S. We have

$$\begin{aligned}
\Pi(dG) &= \phi^*(G) \mathrm{d}g_{1,1} \wedge \mathrm{d}g_{1,2} \wedge \ldots \wedge \mathrm{d}g_{n,n} \\
&= \frac{\phi^*(G)}{\tilde{\phi}^*(G_S)} \sigma_e^n \tilde{\phi}^*(G_S) \mathrm{d}s_{1,1} \wedge \mathrm{d}s_{1,2} \wedge \ldots \wedge \mathrm{d}s_{n,n} \\
&= \frac{\phi^*(G)}{\tilde{\phi}^*(G_S)} \sigma_e^n \Pi(\mathrm{d}G_S),
\end{aligned}$$

where ϕ^* and $\tilde{\phi}^*$ are some functions of G and G_s, respectively.

Since $s_{j,j} \leq 2/\sigma_e$, we have $-(1 - s_{j,j}\sigma_e)^2 = -1 + 2s_{j,j}\sigma_e - s_{j,j}^2\sigma_e^2 \leq 3 - s_{j,j}^2\sigma_e^2$. Continuing from above,

$$\begin{aligned}
&\sum_{j=1}^n b_n d_j - \sum_{j=1}^n d_j (b_n - b_j)(1 - s_{j,j}\sigma_e)^2 \\
&\leq \sum_{j=1}^n b_n d_j + \sum_{j=1}^n d_j (b_n - b_j)(3 - s_{j,j}^2 \sigma_e^2) \\
&= \sum_{j=1}^n 4 b_n d_j - \sum_{j=1}^n 3 d_j b_j - \sum_{j=1}^n d_j (b_n - b_j) s_{j,j}^2 \sigma_e^2 \\
&\leq \sum_{j=1}^n 4 b_n d_j - \sum_{j=1}^n d_j (b_n - b_j) s_{j,j}^2 \sigma_e^2
\end{aligned}$$

Combining the above,

$$\begin{aligned}\Delta \leq Z^{-1}(\sigma_e^2, D, B) \exp\left[\frac{1}{2\sigma_e^2}(\sum_{j=1}^{n} 4b_n d_j - \sum_{j=1}^{n} 3d_j b_j)\right]\sigma_e^n \int_{\mathcal{S}} \sup_{W\in\mathcal{V}^*} \Big\| f(W) - f(WG_S)\Big\| \\ \times \exp\Big[-\frac{1}{2}\sum_{j=1}^{n} d_j(b_n - b_j)s_{j,j}^2\Big]\frac{\phi^*(G)}{\tilde{\phi}^*(G_S)}\Pi(\mathrm{d}G_S).\end{aligned} \tag{16}$$

Note that $\sup_{G_S\in\mathcal{V}^*}\sup_{W\in\mathcal{V}^*}\Big\|f(W) - f(WG_S)\Big\| \leq M$ due to the compactness of $\mathcal{V}^*$ and continuity of f. And clearly,

$$\int_{\mathcal{S}} M\exp\Big[-\frac{1}{2}\sum_{j=1}^{n} d_j(b_n - b_j)s_{j,j}^2\Big]\frac{\phi^*(G)}{\tilde{\phi}^*(G_S)}\Pi(\mathrm{d}G_S) < \infty.$$

Using dominated convergence theorem, when $\sigma_e \to 0$, the integral in (16) goes to zero.

Our remaining task is to verify the constant before the integral is finite as $\sigma_e \to 0$. Note the inverse of the constant in (16)

$$\begin{aligned}&\sigma_e^{-n} Z(\sigma_e^2, D, B)\exp\Big[-\frac{1}{2\sigma_e^2}\sum_{j=1}^{n} 4b_n d_j\Big] \\ &= \sigma_e^{-n}\exp\Big[-\frac{1}{2\sigma_e^2}\sum_{j=1}^{n} 4b_n d_j\Big]\int_{\mathcal{V}^*}\mathrm{etr}\Big\{\frac{1}{2\sigma_e^2}DU'BU\Big\}\Pi(\mathrm{d}U) \\ &= \sigma_e^{-n}\int_{\mathcal{V}^*}\exp\Big\{\frac{1}{2\sigma_e^2}(\sum_{i=1}^{n}\sum_{j=1}^{n} b_i d_j u_{i,j}^2 - \sum_{j=1}^{n} 4b_n d_j\sum_{i=1}^{n} u_{i,j}^2)\Big\}\Pi(\mathrm{d}U) \\ &= \sigma_e^{-n}\int_{\mathcal{V}^*}\exp\Big\{\frac{1}{2\sigma_e^2}\sum_{i=1}^{n}\sum_{j=1}^{n}(b_i - 4b_n)d_j u_{i,j}^2\Big\}\Pi(\mathrm{d}U) \\ &= \sigma_e^{-n}\int_{\mathcal{V}^*}\exp\Big\{\frac{1}{2\sigma_e^2}\sum_{i=1}^{n}\sum_{j=2}^{n}(b_i - 4b_n)d_j u_{i,j}^2\Big\}\Pi(\mathrm{d}U) \\ &\geq \sigma_e^{-n}\int_{\mathcal{V}^*}\exp\Big\{\frac{1}{2\sigma_e^2}\sum_{i=1}^{n}\sum_{j=2}^{n}(b_i - 4b_n)d_n u_{i,j}^2\Big\}\Pi(\mathrm{d}U) \\ &= \sigma_e^{-n}\int_{\mathcal{V}^*}\exp\Big\{\frac{1}{2\sigma_e^2}\sum_{i=1}^{n}(b_i - 4b_n)d_n(1 - u_{i,1}^2)\Big\}\Pi(\mathrm{d}U),\end{aligned} \tag{17}$$

where we use $d_1 = 0$, $(b_i - 4b_n) \leq 0$ and $d_j \leq d_n$ in the inequality. Since the last line does not depend on $U_{2:n}$, we denote the null space of u_1 by $\mathcal{K}(u_1) = \{U_{2:n} \in \mathcal{V}^{n-1,n} : u_k' u_1 = 0, k > 1\}$. It is not hard to see that the volume $\mathcal{K}(u_1)$ is a constant invariant to u_1, we denote it by $vol(\mathcal{K})$. The above is then,

$$\begin{aligned}&\sigma_e^{-n} vol(\mathcal{K})\int_{\mathbb{S}_+}\exp\Big\{-\frac{1}{2\sigma_e^2}\sum_{i=1}^{n}(4b_n - b_i)d_n(1 - u_{i,1}^2)\Big\}\Pi(\mathrm{d}U_1) \\ &\geq \sigma_e^{-n} vol(\mathcal{K})\int_{\mathbb{S}_+}\exp\Big\{-\frac{1}{2\sigma_e^2}\sum_{i=1}^{n}(4b_n - b_i)d_n(1 + u_{i,1})^2\Big\}\Pi(\mathrm{d}U_1),\end{aligned}$$

where $\mathbb{S}_+$ is the unit-norm space constrained to all elements positive; and the inequality due to $-(1-u^2) = -(1-u)(1+u) \geq -(1+u)^2$ for $u \geq 0$.

Let $t_i = (1+u_{i,1})/\sigma_e$. We have

$$\begin{aligned}\Pi(\mathrm{d}U_1) &= \psi(U_1)\mathrm{d}u_{1,1}\wedge\mathrm{d}u_{2,1}\wedge\ldots\wedge\mathrm{d}u_{n,1}\\ &= \frac{\psi(U_1)}{\tilde{\psi}(T)}\sigma_e^n\tilde{\psi}(T)\mathrm{d}t_1\wedge\mathrm{d}t_2\wedge\ldots\wedge\mathrm{d}t_n\\ &= \frac{\psi(U_1)}{\tilde{\psi}(T)}\sigma_e^n\Pi(\mathrm{d}T),\end{aligned}$$

where ψ and $\tilde{\psi}$ are some functions of U_1 and T, respectively.

The above is then

$$vol(\mathcal{K})\int_{\mathcal{T}}\exp\Big\{-\frac{1}{2}\sum_{i=1}^{n}(4b_n-b_i)d_nt_i^2\Big\}\frac{\psi(U_1)}{\tilde{\psi}(T)}\Pi(\mathrm{d}T),$$

which is bounded away from 0. Therefore, the constant in (16) is finite as $\sigma_e^2\to 0$.

The limit result means that for any $\epsilon>0$, we have a neighborhood $\mathcal{D}_\epsilon=\{\sigma_e^2 : 1/\sigma_e^2 > N_\epsilon\}$, so that $\Delta<\epsilon$.

■

Details of the Gibbs Sampling Algorithm

The posterior sampling proceeds according to the following steps:

1. Sample R_s from (11) in the main article.
2. Sample $U^{(l)}$ from (12) in the main article.
3. Sample from the categorical distribution

$$\begin{aligned}z_s\sim\Pi(z_s\mid .)\propto\pi_l 1(z_s=l)\exp\Big\{&\frac{1}{2}(\frac{n-T}{2\sigma_e^2}+\frac{1}{\sigma_\theta^2})^{-1}\Big[\frac{1}{2\sigma_e^2}\mathrm{tr}(\big[L^{(s)}(I_n-U^{(l)}U^{(l)T})\big])+\frac{\mu_\theta}{\sigma_\theta^2}\Big]^2\\ &+\frac{1}{2}(\frac{1}{\sigma^2_{\lambda,\eta_k^{(s)}}}+\frac{1}{2\sigma_e^2})^{-1}\sum_{k=1}^{T}\Big[\frac{u_k^{(l)\prime}L^{(s)}u_k^{(l)}}{2\sigma_e^2}+\frac{(1-\eta_k^{(s)})\mu_\theta}{\sigma^2_{\lambda,\eta_k^{(s)}}}\Big]^2\Big\},\end{aligned}$$

   with $1(.)$ the indicator function, update $Q^{(s)}=U^{(z_s)}$.
4. Sample $(\pi_1,\pi_2,\ldots,\pi_g)\sim\mathrm{Dir}(\alpha_0/g+\sum 1(z_s=1),\alpha_0/g+\sum 1(z_s=2),\ldots,\alpha_0/g+\sum 1(z_s=1))$.
5. Sample for $k=2,\ldots,T$

$$\lambda_k^{(s)}\sim N_{(0,2)}\Big\{(\frac{1}{\sigma^2_{\lambda,\eta_k^{(s)}}}+\frac{1}{2\sigma_e^2})^{-1}\Big[\frac{q_k^{(s)\prime}L^{(s)}q_k^{(s)}}{2\sigma_e^2}+\frac{(1-\eta_k^{(s)})\mu_\theta}{\sigma^2_{\lambda,\eta_k^{(s)}}}\Big],(\frac{1}{\sigma^2_{\lambda,\eta_k^{(s)}}}+\frac{1}{2\sigma_e^2})^{-1}\Big\}.$$

6. Sample from the Bernoulli for $k=2,\ldots,T$,

$$\eta_k^{(s)}\sim 1(\eta_k^{(s)}=1)wN_{(0,2)}(\lambda_k^{(s)};0,\sigma^2_{\lambda,1})+1(\eta_k^{(s)}=0)(1-w)N_{(0,2)}(\lambda_k^{(s)};\mu_\theta,\sigma^2_{\lambda,0}),$$

   where $N_{(0,2)}(x;a,b)$ denotes the density of the truncated normal.
7. Sample

$$\begin{aligned}\theta^{(s)}\sim N_{(0,2)}\Big\{&(\frac{n-T}{2\sigma_e^2}+\frac{1}{\sigma_\theta^2})^{-1}\Big[\frac{1}{2\sigma_e^2}(\sum_i L^{(s)}(i,i)-\sum_k q_k^{(s)\mathrm{T}}L^{(s)}q_k^{(s)})+\frac{\mu_\theta}{\sigma_\theta^2}\Big],\\ &(\frac{n-T}{2\sigma_e^2}+\frac{1}{\sigma_\theta^2})^{-1}\Big\}.\end{aligned}$$

8. Sample for $i=1,\ldots,n$

$$L_{i,i}^{(s)}\sim N\Big\{\big[Q^{(s)}(\Lambda^{(s)}-\theta^{(s)}I_T)Q^{(s)\prime}\big]_{(i,i)}+\theta^{(s)},2\sigma_e^2\Big\}.$$

9. Sample

$$\sigma_e^2 \sim \text{Inv-Gamma}\left\{\frac{n^2 S}{2}, \frac{1}{4}\sum_{s=1}^{S}\|L^{(s)} - \theta I_n - Q_*^{(l)}(\Lambda^{(s)} - \theta^{(s)} I_T)Q_*^{(l)\prime}\|_F^2\right\}.$$

The Laplacian Eigenvalues of Sparse Graphs under High Noise Level

To provide some numerical illustration, we simulated additional graphs as in Section 5.1, except with the Bernoulli probability reduced to 0.3, 0.2 and 0.1, respectively.

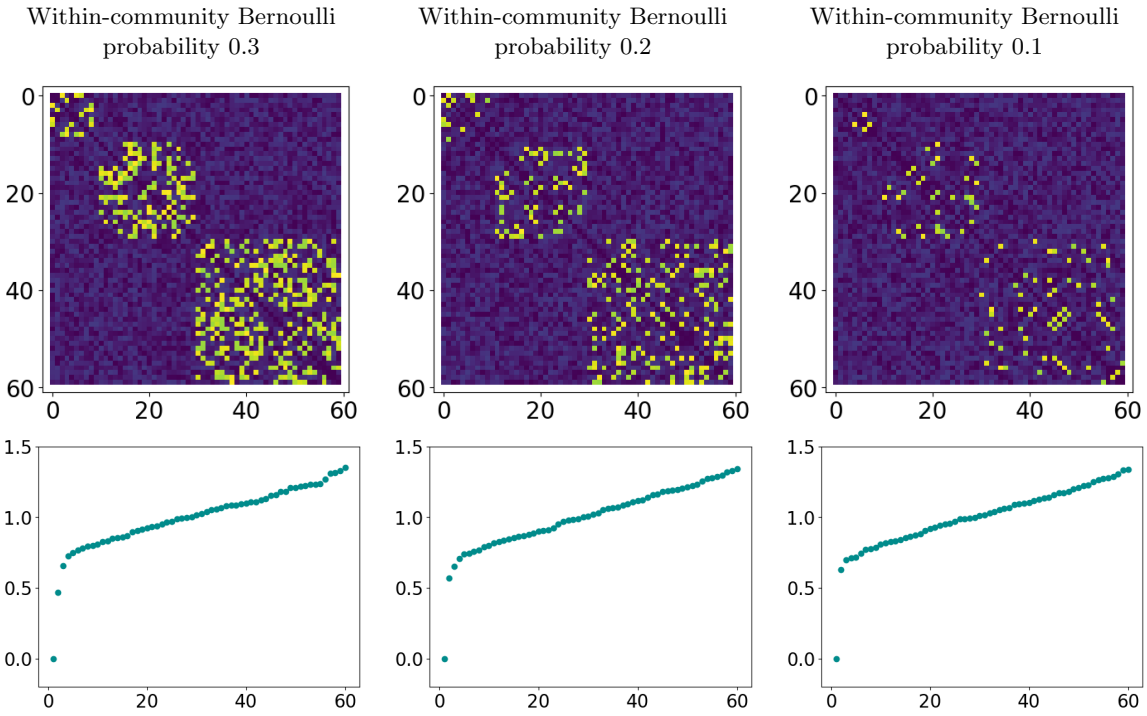


Figure 10: When the graph sparsity increases, but the noise level remains relatively high, it becomes more difficult to distinguish the first few eigenvalues of the Laplacian from the remaining larger ones.

Additional Components in Working Memory Data Analysis

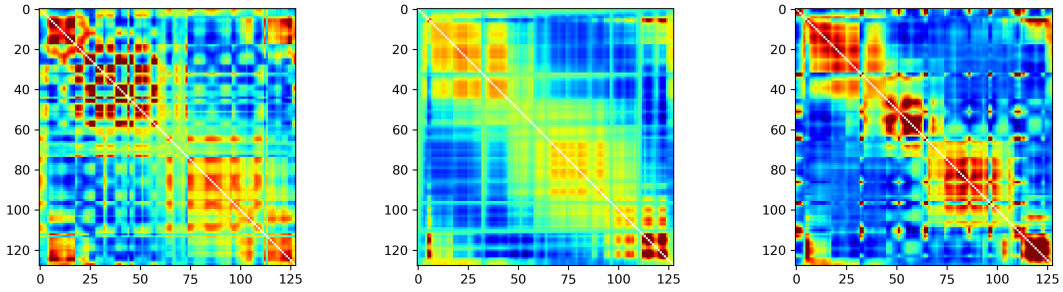


Figure 11: Fitted Laplacian shows the structure underneath the raw connectivity matrix.

Additional Details on Normalized Mutual Information

On quantifying the accuracy of $\hat{c}_i^{(s)}$, we use the normalized mutual information, as a measure of similarity that is invariant to label switching. To provide some more details, consider discrete x and y as in two vectors of equal lengths, using $P(\cdot)$ to denote a proportion, we have

$$\begin{aligned}
\mathrm{I}(x, y) &= \sum_{i,j} P(x = i, y = j) \log \left(\frac{P(x = i, y = j)}{P(x = i)P(y = j)} \right), \\
\mathrm{H}(x) &= -\sum_{i} P(x = i) \log P(x = i), \\
\mathrm{NMI}(x, y) &= \frac{2I(x, y)}{H(x) + H(y)}.
\end{aligned}$$