On Spectral Algorithms for Community Detection in Stochastic Blockmodel Graphs With Vertex Covariates

Cong Mu, Angelo Mele, Lingxin Hao, Joshua Cape, Avanti Athreya, and Carey E. Priebe, *Senior Member, IEEE*

***Abstract*—In network inference applications, it is often desirable to detect community structure. Beyond mere adjacency matrices, many real-world networks also involve vertex covariates that carry key information about underlying block structure in graphs. To assess the effects of such covariates on block recovery, we present a comparative analysis of two model-based spectral algorithms for clustering vertices in stochastic blockmodel graphs with vertex covariates. The first algorithm uses only the adjacency matrix, and directly estimates the block assignments. The second algorithm incorporates both the adjacency matrix and the vertex covariates into the estimation of block assignments, and moreover quantifies the explicit impact of the vertex covariates on the resulting estimate of the block assignments. We employ Chernoff information to analytically compare the algorithms' performance and derive the information-theoretic Chernoff ratio for certain models of interest. Analytic results and simulations suggest that the second algorithm is often preferred: one can better estimate the induced block assignments by first estimating the effect of vertex covariates. In addition, real data examples also indicate that the second algorithm has the advantage of revealing underlying block structure while considering observed vertex heterogeneity in real applications.**

***Index Terms*—Spectral graph inference, community detection, stochastic blockmodel, vertex covariate, chernoff ratio.**

I. Introduction

NETWORK data, which encodes interactions or relationships across entities, often involves more than mere links or connections across network vertices. For example, a network dataset may include both an adjacency matrix, which consolidates information about vertices in the network and the edges between them, as well as additional vertex covariates. For example, in diffusion MRI connectome datasets [1], vertices represent sub-regions of the brain defined via spatial proximity, and edges represent tensor-based fiber streamlines connecting these sub-regions; such graphs can also have brain hemisphere and tissue labels for each vertex. Social network datasets [2]–[4], in which vertices can represent users or web pages and edges can represent followers or relationships, may come with ancillary demographic information for each vertex. Since accurate inference on random networks depends on exploiting all available signal, scalable algorithms that can incorporate both network connectivity data and any additional insight from vertex covariates are desirable. For instance, in the well-known k-block stochastic blockmodel (SBM) [5], network vertices belong to k distinct groups, or communities, called *blocks*, and the probabilities of connection across vertices depend on their block memberships. That is, if τ_i represents the block associated to vertex i, the connection probability between vertex i and j is a function of τ_i and τ_j. Typically, a vertex's block membership depends on inherent but unobserved (latent) vertex properties. Thus, a classic inference task is to estimate block memberships from a realization of the resulting network. If, however, we observe both adjacency matrices and vertex covariates, and if both can contain information about the latent communities or blocks, we need models and scalable algorithms that can effectively incorporate adjacency structure *and* covariate data and account for their potentially disparate effects.

In fact, vertex covariates can influence the very number of communities that are detected in a blockmodel: for example, a two-block SBM might bifurcate further into a four-block SBM because of the impact of a binary covariate (with each block splitting according to the covariate). Standard estimation on a graph, then, may yield a four-block assignment, but understanding the underlying two-block SBM is important in inference applications, as we show. To get to an estimate of the underlying two-block assignment, we need to understand the role of the covariates.

Moreover, a problem of interest in network hypothesis testing is to assess the influence of latent communities on downstream or outcome variables, controlling for vertex covariate effects [6], [7]. For instance, assume $\mathbf{y}_i$ represents some outcome variable associated to vertex i in a K-block SBM (for example, in a demographic data set, $\mathbf{y}_i$ might represent an individual's educational attainment or earnings). Suppose this

Manuscript received February 23, 2021; revised August 3, 2021; accepted May 22, 2022. Date of publication May 26, 2022; date of current version September 9, 2022. This work was supported in part by US National Science Foundation under Grant SES-1951005 and in part by US Defense Advanced Research Projects Agency under the D3M program administered through contract FA8750-17-2-0112. Recommended for acceptance by Prof. Xianbin Cao. *(Corresponding author: Cong Mu.)*

Cong Mu, Avanti Athreya, and Carey E. Priebe are with the Department of Applied Mathematics and Statistics, Johns Hopkins University, Baltimore, MD 21218 USA (e-mail: cmu2@jhu.edu; dathrey1@jhu.edu; cep@jhu.edu).

Angelo Mele is with the Carey Business School, Johns Hopkins University, Baltimore, MD 21202 USA (e-mail: angelo.mele@jhu.edu).

Lingxin Hao is with the Department of Sociology, Johns Hopkins University, Baltimore, MD 21218 USA (e-mail: hao@jhu.edu).

Joshua Cape is with the Department of Statistics, University of Pittsburgh, Pittsburgh, PA 15260 USA (e-mail: joshua.cape@pitt.edu).

Digital Object Identifier 10.1109/TNSE.2022.3177708

outcome variable's distribution depends on the vertex's block assignment within the network, so if $\tau_i = k$, then $\mathbf{y}_i$ follows some distribution F_k that can depend on this block. We describe this scenario by writing $\mathbf{y}_i|(\tau_i = k) \sim F_k$. A natural question to ask is whether the distributions of the outcome variables are the same for different blocks, i.e., to test whether $F_k = F$ for $k \in \{1, \ldots, K\}$. To achieve this goal when we have both adjacency and covariate information, it is crucial to estimate the underlying block structure $\widehat{\boldsymbol{\tau}}$—namely, to obtain an estimate of the block structure *after* accounting for, and effectively "netting out" the vertex covariate effect. Here we write "induced block assignment" to refer to the block assignment *after* accounting for the vertex covariates.

As a special case of random graph models, SBMs are popular in the literature for community detection [5], [8], [9]. Many classical methods consider the adjacency or Laplacian matrices for community detection; see [10] for an overview. However, these methods are typically not designed to distinguishing the impact of covariates from the mechanism of network generation itself—that is, delineating in the observed data what may be underlying, or fundamental, *network* effects from characteristics that are more properly functions of the covariates. By contrast, covariate-aware inference in SBMs often relies on either variational methods [11]–[13] or spectral approaches [14]–[16]. For example, [14] proposed covariate-assisted spectral clustering (CASC) where the covariates are first parameterized as in linear regression, i.e., categorical covariates are represented with dummy variables and continuous covariates can go through standardization, and then combined with the graph for subsequent spectral clustering. The pairwise covariates-adjusted stochastic blockmodel (PCABM), in which pairwise covariate information is incorporated with the classical SBM, was introduced in [15]. There, model parameters can be solved via maximum likelihood estimation (MLE) or spectral clustering with adjustment (SCWA).

Spectral methods [17] that promise applicability to large graphs have been widely used in random graph models for a variety of subsequent inference tasks such as community detection [18]–[21], vertex nomination [22], nonparametric hypothesis testing [23], and multiple graph inference [24]. Two particular spectral embedding methods, adjacency spectral embedding (ASE) and Laplacian spectral embedding (LSE), which are spectral decompositions of the graph adjacency and graph Laplacian matrices, respectively, are popular, since they provide consistent [25] and asymptotically normal [26], [27] estimates of underlying graph parameters, such as block memberships. To compare the performance of these two embedding methods, the concept of Chernoff information is employed for SBMs [8], [27] and then extended to consider the underlying graph structure [28]. The Chernoff information between two distributions F_1 and F_2 is related to the exponential rate of decay of the Bayes risk in the simple hypothesis test comparing F_1 against F_2, as the sample size increases. As such, because of asymptotic normality of the adjacency and Laplacian spectral embedding for stochastic blockmodels, the Chernoff information between two normal distributions (with different mean vectors and covariance matrices) can be adopted to derive the large sample optimal error rate for recovering block assignments in a SBM.

In this work, we investigate two spectral algorithms for clustering vertices in stochastic blockmodel graphs with vertex covariates. Analytically, we compare the algorithms' performance via Chernoff information and derive the Chernoff ratio for certain models of interest. The notion of Chernoff information for comparing algorithms will be addressed in detail in Section IV. Practically, we compare the algorithms' empirical clustering performance by simulations and real data examples on diffusion MRI connectome and social networks.

The structure of this article is as follows. Section II reviews relevant models for random graphs and the basic idea of spectral methods. Section III introduces our spectral algorithms for clustering vertices in stochastic blockmodel graphs with vertex covariates. Section IV analytically compares the algorithms' performance via Chernoff information and derives the Chernoff ratio expression for certain models of interest. Section V provides simulations and real data examples on diffusion MRI connectome and social networks to compare the algorithms' performance. Section VI discusses the findings and raises questions for further investigation. Appendices provide technical details for latent position geometry and analytic derivations of the Chernoff ratio as well as the details of simulations. The implementation of our algorithms can be found at https://github.com/CongM/sbm-cov.

II. Models and Spectral Methods

To ground our analysis and results, we begin with a particular class of random network models known as latent position models [29], [30] for edge-independent random graphs. In these models, each network vertex i is associated with a latent position $\mathbf{X}_i \in \mathcal{X}$ where $\mathcal{X}$ is some latent space such as $\mathbb{R}^d$, and edges between vertices arise independently with probability $\mathbf{P}_{ij} = \kappa(\mathbf{X}_i, \mathbf{X}_j)$ for some kernel function $\kappa : \mathcal{X} \times \mathcal{X} \to [0, 1]$. This is an appealing model to consider not only because of its wide applicability—after all, the kernel can be any reasonable regular function—but because it is easily interpretable. For example, social network connections are often a function of individual participants' (potentially unobserved) interests in a core set of topics or hobbies, and levels of interest can be easily encoded in a low-dimensional space. Moreover, the kernel and this lower-dimensional space can possess intuitive geometry, wherein collinearity or other "closeness" of latent positions increases the probability of a connection between the associated vertices. The core model we focus on here, the generalized random dot product graph (GRDPG), has precisely such a property: the kernel function is taken to be the (indefinite) inner product. As the name suggests, this model generalizes the *random dot product graph* (RDPG) by relaxing the restriction that the kernel function be the inner product, and this relaxation permits SBM with dissassortative structure, and in fact subsumes all SBMs as special cases.

Definition 1 (Generalized Random Dot Product Graph [31]): Let $d = d_+ + d_-$ with $d_+ \geq 1$ and $d_- \geq 0$. Let $\mathbf{I}_{d_+d_-} = \mathrm{diag}(1, \ldots, 1, -1, \ldots, -1)$, i.e., a $d \times d$ diagonal matrix with

1 in first d_+ entries and -1 in the next d_- entries. Let $\mathbf{A} \in \{0,1\}^{n\times n}$ be an adjacency matrix and $\mathbf{X} = [\mathbf{X}_1, \ldots, \mathbf{X}_n]^\top \in \mathbb{R}^{n\times d}$ where each $\mathbf{X}_i \in \mathbb{R}^d$ denotes the latent position for vertex i satisfying $\mathbf{X}_i^\top \mathbf{I}_{d_+d_-}\mathbf{X}_j \in [0,1]$ for all $i,j \in \{1,\ldots,n\}$. Then we say $(\mathbf{A},\mathbf{X}) \sim \mathrm{GRDPG}(n,d_+,d_-)$ if for any $i,j \in \{1,\ldots,n\}$

$$\begin{aligned}\mathbf{A}_{ij} &\sim \mathrm{Bernoulli}(\mathbf{P}_{ij}),\\ \mathbf{P}_{ij} &= \mathbf{X}_i^\top \mathbf{I}_{d_+d_-}\mathbf{X}_j.\end{aligned} \tag{1}$$

As mentioned above, the SBM, which encapsulates block structure in independent-edge networks, is a special case of the GRDPG.

Definition 2 (K-block Stochastic Blockmodel Graph [5]): The K-block stochastic blockmodel (SBM) graph is an independent-edge random graph with each vertex belonging to one of K blocks. It can be parameterized by a block connectivity probability matrix $\mathbf{B} \in [0,1]^{K\times K}$ and a nonnegative vector of block assignment probabilities $\boldsymbol{\pi} \in [0,1]^K$ summing to unity. Let $\mathbf{A} \in \{0,1\}^{n\times n}$ be an adjacency matrix and $\boldsymbol{\tau} \in \{1,\ldots,K\}^n$ be a vector of block assignments with $\tau_i = k$ if vertex i is in block k (occurring with probability π_k). We say $(\mathbf{A},\boldsymbol{\tau}) \sim \mathrm{SBM}(n,\mathbf{B},\boldsymbol{\pi})$ if for any $i,j \in \{1,\ldots,n\}$

$$\begin{aligned}\mathbf{A}_{ij} &\sim \mathrm{Bernoulli}(\mathbf{P}_{ij}),\\ \mathbf{P}_{ij} &= \mathbf{B}_{\tau_i\tau_j}.\end{aligned} \tag{2}$$

The SBM can be thought of as the GRDPG with locations fixed in each block. Formally, let $(\mathbf{A},\boldsymbol{\tau}) \sim \mathrm{SBM}(n,\mathbf{B},\boldsymbol{\pi})$ as in Definition 2 where $\mathbf{B} \in [0,1]^{K\times K}$ with d_+ strictly positive eigenvalues and d_- strictly negative eigenvalues. To represent this SBM in the GRDPG model, we can choose $\boldsymbol{\nu}_1,\ldots,\boldsymbol{\nu}_K \in \mathbb{R}^d$ where $d = d_+ + d_-$ such that $\boldsymbol{\nu}_k^\top \mathbf{I}_{d_+d_-}\boldsymbol{\nu}_\ell = \mathbf{B}_{k\ell}$ for all $k,\ell \in \{1,\ldots,K\}$. For example, we can take $\boldsymbol{\nu} = \mathbf{U}|\boldsymbol{\Lambda}|^{1/2}$ where $\mathbf{B} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top$ is the eigendecomposition of $\mathbf{B}$ after re-ordering. Then we have the latent position of vertex i as $\mathbf{X}_i = \boldsymbol{\nu}_k$ if $\tau_i = k$.

Example 1 (Two-block Rank One Model): As an illustration, consider the prototypical two-block SBM with rank one block connectivity probability matrix $\mathbf{B}$ where $\mathbf{B}_{11} = p^2, \mathbf{B}_{22} = q^2, \mathbf{B}_{12} = \mathbf{B}_{21} = pq$ with $0 < p < q < 1$. Let $\mathbf{X}_i$ be the latent position of vertex i where $\mathbf{X}_i = \boldsymbol{\nu}_1 = p$ if $\tau_i = 1$ and $\mathbf{X}_i = \boldsymbol{\nu}_2 = q$ if $\tau_i = 2$. Then we can represent this SBM in the GRDPG model with latent positions $\boldsymbol{\nu} = [\,p \quad q\,]^\top$ as

$$\mathbf{B} = \boldsymbol{\nu}\boldsymbol{\nu}^\top = \begin{bmatrix} p^2 & pq \\ pq & q^2 \end{bmatrix}. \tag{3}$$

Since our goal is to examine the impact of covariates on network inference, we next extend the GRDPG to permit vertex covariates, as follows.

Definition 3 (GRDPG with Vertex Covariates [16]): Consider GRDPG as in Definition 1. Let $\mathbf{Z}$ denote the observed vertex covariate and β denote the effect of the vertex covariate. Then we say $(\mathbf{A},\mathbf{X},\mathbf{Z},\boldsymbol{\beta}) \sim \mathrm{GRDPG\text{-}Cov}(n,d_+,d_-)$ for any $i,j \in \{1,\ldots,n\}$

$$\begin{aligned}\mathbf{A}_{ij} &\sim \mathrm{Bernoulli}(\mathbf{P}_{ij}),\\ \mathbf{P}_{ij} &= \mathbf{X}_i^\top \mathbf{I}_{d_+d_-}\mathbf{X}_j + \beta\mathbf{1}\{\mathbf{Z}_i = \mathbf{Z}_j\}.\end{aligned} \tag{4}$$

Remark 1: In the case of an SBM, we have

$$\mathbf{P}_{ij} = \mathbf{B}_{\tau_i\tau_j} + \beta\mathbf{1}\{\mathbf{Z}_i = \mathbf{Z}_j\}. \tag{5}$$

Example 2 (Two-block Rank One Model with One Binary Covariate): As an illustration, consider the rank one matrix $\mathbf{B}$ in Eq. (3) and the SBM model in Remark 1. Let $\mathbf{Z} \in \{1,2\}^n$ denote the observed binary covariate. Assume $0 < \beta < 1$ with $p^2+\beta, q^2+\beta, pq+\beta \in [0,1]$. Then we have the block connectivity probability matrix with the vertex covariate effect as

$$\mathbf{B}_{\mathbf{Z}} = \begin{bmatrix} p^2+\beta & p^2 & pq+\beta & pq \\ p^2 & p^2+\beta & pq & pq+\beta \\ pq+\beta & pq & q^2+\beta & q^2 \\ pq & pq+\beta & q^2 & q^2+\beta \end{bmatrix}. \tag{6}$$

Example 3 (Two-block Homogeneous Model with One Binary Covariate): As a second illustration, consider the rank two matrix $\mathbf{B}$ where $\mathbf{B}_{11} = \mathbf{B}_{22} = a, \mathbf{B}_{12} = \mathbf{B}_{21} = b$ with $0 < b < a < 1$. Assume $0 < \beta < 1$ with $a+\beta, b+\beta \in [0,1]$. We then have the block connectivity probability matrix with the vertex covariate effect as

$$\mathbf{B}_{\mathbf{Z}} = \begin{bmatrix} a+\beta & a & b+\beta & b \\ a & a+\beta & b & b+\beta \\ b+\beta & b & a+\beta & a \\ b & b+\beta & a & a+\beta \end{bmatrix}. \tag{7}$$

Remark 2: The SBMs parameterized by $\mathbf{B}$ in Example 3 lead to the notion of the homogeneous model [8], [28]. For K-block homogeneous model, we have $\mathbf{B}_{k\ell} = a$ for $k = \ell$ and $\mathbf{B}_{k\ell} = b$ for $k \neq \ell$.

In Examples 2 and 3, an induced two-block SBM becomes a four-block SBM via the effect of a binary vertex covariate. The goal is to cluster each vertex into one of the two induced blocks after accounting for the vertex covariate effect. To this end, we need to recover the latent positions of the underlying GRDPG, using the adjacency spectral embedding.

Definition 4 (Adjacency Spectral Embedding): Let $\mathbf{A} \in \{0,1\}^{n\times n}$ be an adjacency matrix with eigendecomposition $\mathbf{A} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^\top$. Given the embedding dimension $d < n$, the adjacency spectral embedding (ASE) of $\mathbf{A}$ into $\mathbb{R}^d$ is the $n \times d$ matrix $\widehat{\mathbf{X}} = \widehat{\mathbf{U}}_d|\widehat{\boldsymbol{\Lambda}}_d|^{1/2}$ where $\widehat{\boldsymbol{\Lambda}}_d$ is a diagonal matrix with the d largest eigenvalues in magnitudes and $\widehat{\mathbf{U}}_d$ contains the associated eigenvectors. Here hat notation suggests these terms estimate the eigenvectors and eigenvalues of the matrix $\mathbf{P}$ as in Eq. (1).

Remark 3: There are different methods for choosing the embedding dimension [32], [33]; we adopt the well-established and computationally efficient profile likelihood method [34] to automatically identify an elbow in the scree plot to select embedding dimension $\widehat{d}$.

Algorithm 1: Estimation of induced block assignment using only the adjacency matrix.

Input: Adjacency matrix $\mathbf{A} \in \{0,1\}^{n\times n}$
Output: Induced block assignments as $\widehat{\boldsymbol{\tau}}$.

1: Estimate latent positions under the effects of both observed covariates and unobserved heterogeneity of vertices as $\widehat{\mathbf{Y}} \in \mathbb{R}^{n\times \widehat{d}}$ using ASE of $\mathbf{A}$ where $\widehat{d}$ is chosen as in Remark 3.
2: Cluster $\widehat{\mathbf{Y}}$ using Gaussian mixture modeling (GMM) to estimate the block assignments under the effects of both observed covariates and unobserved heterogeneity of vertices as $\widehat{\boldsymbol{\xi}} \in \{1,\dots,\widehat{K}\}^n$ where $\widehat{K}$ is chosen via Bayesian Information Criterion (BIC).
3: Compute the estimated block connectivity probability matrix including the vertex covariate effect as

$$\widehat{\mathbf{B}}_Z = \widehat{\boldsymbol{\mu}}\mathbf{I}_{d_+ d_-}\widehat{\boldsymbol{\mu}}^\top \in [0,1]^{\widehat{K}\times\widehat{K}},$$

where $\widehat{\boldsymbol{\mu}} \in \mathbb{R}^{\widehat{K}\times\widehat{d}}$ is the matrix of estimated means of all clusters.
4: Cluster the diagonal of $\widehat{\mathbf{B}}_Z$ using GMM to estimate the cluster assignments of the diagonal as $\widehat{\boldsymbol{\phi}} \in \{1,\dots,\frac{\widehat{K}}{2}\}^{\widehat{K}}$.
5: Estimate the induced block assignments as $\widehat{\boldsymbol{\tau}}$ by $\widehat{\boldsymbol{\tau}}_k = c$ for $k \in \{i \mid \widehat{\boldsymbol{\xi}}_i = t \text{ for } t \in \{j \mid \widehat{\boldsymbol{\phi}}_j = c\}\}$ and $c = 1,\dots,\frac{\widehat{K}}{2}$.

III. Model-Based Spectral Inference

We are interested in estimating the induced block assignments (clustering vertices) in a SBM with vertex covariates. To that end, we also consider algorithms for estimating the vertex covariate effect β, which can be further used to estimate the induced block assignments. Our model-based spectral algorithms take observed adjacency matrices (and vertex covariates) as inputs and estimated block assignments for each vertex as outputs.

In Algorithm 1, the estimation of the induced block assignments, i.e., $\widehat{\boldsymbol{\tau}}$, depends on the estimated block connectivity probability matrix $\widehat{\mathbf{B}}_Z$ (see Step 4 of Algorithm 1 for details). This suggests that we may not obtain an accurate estimate of the induced block assignments if the diagonal of $\widehat{\mathbf{B}}_Z$ does not contain enough information to distinguish the induced block structure. To address this uncertainty, we consider a modified algorithm that uses the information from vertex covariates to estimate the induced block assignments along with vertex covariate effect β.

As an illustration of estimating β (Step 2 in Algorithm 2), consider the block connectivity probability matrix $\mathbf{B}_Z$ as in Eq. (7). To get β, we can take the difference between two specific entries of $\mathbf{B}_Z$. For example,

$$\begin{aligned}\mathbf{B}_{Z,11} - \mathbf{B}_{Z,12} &= (a+\beta) - a = \beta,\\ \mathbf{B}_{Z,13} - \mathbf{B}_{Z,14} &= (b+\beta) - b = \beta.\end{aligned} \tag{8}$$

We can then obtain $\widehat{\beta}$ by subtracting two specific entries of $\widehat{\mathbf{B}}_Z$. However, the ASE and GMM under GRDPG model can lead to the re-ordering of $\widehat{\mathbf{B}}_Z$. Thus we need to identify pairs first so that we subtract the correct entries. Two alternative ways to achieve this are described in Step 2(a) and 2(b).

Algorithm 2: Estimation of induced block assignment incorporating both the adjacency matrix and the vertex covariates.

Input: Adjacency matrix $\mathbf{A} \in \{0,1\}^{n\times n}$; observed vertex covariates $\mathbf{Z} \in \{1,2\}^n$
Output: Estimated vertex covariate effect as $\widehat{\beta}$; induced block assignments as $\widetilde{\boldsymbol{\tau}}$.

1: Steps 1 - 4 in Algorithm 1.
2: Estimate the vertex covariate effect as $\widehat{\beta}$ using one of the following procedures [16].
(a) Assign the block covariates as $\mathbf{Z}_B \in \{-1,1\}^{\widehat{K}}$ for each block using the mode, i.e.,

$$\mathbf{Z}_{B,k} = \begin{cases} -1 & \text{if } n_{-1,k} \geq n_{1,k},\\ 1 & \text{if } n_{-1,k} < n_{1,k},\end{cases}$$

where

$$n_{z,k} = \sum_{i:\widehat{\boldsymbol{\xi}}_i = k} \mathbf{1}\{\mathbf{Z}_i = z\}.$$

Construct pair set $S = \{(k\ell, k\ell'), k,\ell,\ell' \in \{1,\dots,\widehat{K}\} \mid \widehat{\boldsymbol{\phi}}_\ell = \widehat{\boldsymbol{\phi}}_{\ell'}, \mathbf{Z}_{B,k} = \mathbf{Z}_{B,\ell}, \mathbf{Z}_{B,k} \neq \mathbf{Z}_{B,\ell'}\}$. Estimate the vertex covariate effect as

$$\widehat{\beta}_{\mathrm{SA}} = \frac{1}{|S|}\sum_{(k\ell,k\ell')\in S} \widehat{\mathbf{B}}_{Z,k\ell} - \widehat{\mathbf{B}}_{Z,k\ell'}.$$

(b) Compute the probability that two entries from $\widehat{\mathbf{B}}_Z$ form a pair as

$$p_{k\ell,k\ell'} = \frac{n_{-1,k}n_{-1,\ell}n_{1,\ell'} + n_{1,k}n_{1,\ell}n_{-1,\ell'}}{n_k n_\ell n_{\ell'}},$$

where

$$n_k = \sum_{i=1}^{n} \mathbf{1}\{\widehat{\boldsymbol{\xi}}_i = k\}.$$

Construct pair set $W = \{(\ell,\ell'), \ell,\ell' \in \{1,\dots,\widehat{K}\} \mid \widehat{\boldsymbol{\phi}}_\ell = \widehat{\boldsymbol{\phi}}_{\ell'}\}$. Estimate the vertex covariate effect as

$$\widehat{\beta}_{\mathrm{WA}} = \frac{1}{\widehat{K}|W|}\sum_{k=1}^{\widehat{K}}\sum_{(\ell,\ell')\in W} p_{k\ell,k\ell'}\left(\widehat{\mathbf{B}}_{Z,k\ell} - \widehat{\mathbf{B}}_{Z,k\ell'}\right).$$

3: Account for the vertex covariate effect by

$$\widetilde{\mathbf{A}}_{ij} = \mathbf{A}_{ij} - \widehat{\beta}\mathbf{1}\{\mathbf{Z}_i = \mathbf{Z}_j\},$$

where $\widehat{\beta}$ is either $\widehat{\beta}_{\mathrm{SA}}$ or $\widehat{\beta}_{\mathrm{WA}}$.
4: Estimate latent positions after accounting for the vertex covariate effect as $\widetilde{\mathbf{Y}} \in \mathbb{R}^{n\times\widetilde{d}}$ using ASE of $\widetilde{\mathbf{A}}$ where $\widetilde{d}$ is chosen as in Remark 3.
5: Cluster $\widetilde{\mathbf{Y}}$ using GMM to estimate the induced block assignments as $\widetilde{\boldsymbol{\tau}} \in \{1,\dots,\frac{\widehat{K}}{2}\}^n$.

In Step 2(a), we find pairs in $\widehat{\mathbf{B}}_Z$ by first assigning each block common covariates using the mode. However, it is possible that we can not find any pairs using this approach, especially in the unbalanced case where the size of each block is

different and/or the distribution of the vertex covariate is different. For example, one block size is much larger than the others and/or vertex covariates are all the same within one block.

In Step 2(b), instead of first finding pairs using the mode, we only compute the probability that two entries of $\widehat{\mathbf{B}}_Z$ form a pair. This will make the estimation more robust to extreme cases or special structure by giving different weights to pairs [16].

IV. Spectral Inference Performance

A. Chernoff Ratio

1) Main Idea: We employ Chernoff information to compare the performance of Algorithms 1 and 2 for estimating the induced block assignments in SBMs with vertex covariates. There are other metrics for comparing spectral inference performance such as within-class covariance. The advantages of Chernoff information are that it is independent of the clustering procedure, i.e., it can be derived no matter which clustering methods are used, and it is intrinsically related to the Bayes risk [9], [27], [35]. In short, there will be a quantity associated with each algorithm, say ρ_1^* and ρ_2^* are associated with the Algorithms 1 and 2 respectively. The comparison is based on the ratio $\rho^* = \rho_1^*/\rho_2^*$. If $\rho^* > 1$, then Algorithm 1 is preferred, otherwise Algorithm 2 is preferred. The following sections provide the mathematical details of Chernoff information and derive ρ^* for certain model of interest.

2) Mathematical Details: Let F_1 and F_2 be two continuous multivariate distributions on $\mathbb{R}^d$ with density functions f_1 and f_2. The Chernoff information [36], [37] is defined as

$$\begin{aligned} C(F_1, F_2) &= -\log\left[\inf_{t\in(0,1)} \int_{\mathbb{R}^d} f_1^t(\mathbf{x}) f_2^{1-t}(\mathbf{x}) d\mathbf{x}\right] \\ &= \sup_{t\in(0,1)}\left[-\log \int_{\mathbb{R}^d} f_1^t(\mathbf{x}) f_2^{1-t}(\mathbf{x}) d\mathbf{x}\right]. \end{aligned} \tag{9}$$

Consider the special case where we take $F_1 = \mathcal{N}(\boldsymbol{\mu}_1, \Sigma_1)$ and $F_2 = \mathcal{N}(\boldsymbol{\mu}_2, \Sigma_2)$; then the corresponding Chernoff information is

$$\begin{aligned} C(F_1, F_2) = \sup_{t\in(0,1)}\Bigg[&\frac{1}{2} t(1-t)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^\top \Sigma_t^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2) \\ &+ \frac{1}{2}\log\frac{|\Sigma_t|}{|\Sigma_1|^t |\Sigma_2|^{1-t}}\Bigg], \end{aligned}$$

where $\Sigma_t = t\Sigma_1 + (1-t)\Sigma_2$. For a given embedding method such as ASE in Algorithms 1 and 2, comparison via Chernoff information is based on the statistical information between the limiting distributions of the blocks and smaller statistical information implies less information to discriminate between different blocks of the SBM. To that end, we also review the limiting results of ASE for SBM, essential for investigating Chernoff information.

Theorem 1 (CLT of ASE for SBM [31]): Let $(\mathbf{A}^{(n)}, \mathbf{X}^{(n)}) \sim \mathrm{GRDPG}(n, d_+, d_-)$ be a sequence of adjacency matrices and associated latent positions of a d-dimensional GRDPG as in Definition 1 from an inner product distribution F where F is a mixture of K point masses in $\mathbb{R}^d$, i.e.,

$$F = \sum_{k=1}^K \pi_k \delta_{\boldsymbol{\nu}_k} \quad \text{with} \quad \forall k,\ \pi_k > 0 \ \text{ and } \ \sum_{k=1}^K \pi_k = 1, \tag{11}$$

where $\delta_{\boldsymbol{\nu}_k}$ is the Dirac delta measure at $\boldsymbol{\nu}_k$. Let $\Phi(\mathbf{z}, \Sigma)$ denote the cumulative distribution function (CDF) of a multivariate Gaussian distribution with mean $\mathbf{0}$ and covariance matrix Σ, evaluated at $\mathbf{z} \in \mathbb{R}^d$. Let $\widehat{\mathbf{X}}^{(n)}$ be the ASE of $\mathbf{A}^{(n)}$ with $\widehat{\mathbf{X}}_i^{(n)}$ as the i-th row (same for $\mathbf{X}_i^{(n)}$). Then there exists a sequence of matrices $\mathbf{M}_n \in \mathbb{R}^{d\times d}$ satisfying $\mathbf{M}_n \mathbf{I}_{d_+d_-} \mathbf{M}_n^\top = \mathbf{I}_{d_+d_-}$ such that for all $\mathbf{z} \in \mathbb{R}^d$ and fixed index i,

$$\mathbb{P}\Big\{\sqrt{n}\Big(\mathbf{M}_n \widehat{\mathbf{X}}_i^{(n)} - \mathbf{X}_i^{(n)}\Big) \le \mathbf{z} \mid \mathbf{X}_i^{(n)} = \boldsymbol{\nu}_k\Big\} \to \Phi(\mathbf{z}, \Sigma_k), \tag{12}$$

where for $\boldsymbol{\nu} \sim F$

$$\begin{aligned} \boldsymbol{\Delta} &= \mathbb{E}\big[\boldsymbol{\nu}\boldsymbol{\nu}^\top\big], \\ \boldsymbol{\Gamma}_k &= \mathbb{E}\big[\big(\boldsymbol{\nu}_k^\top \mathbf{I}_{d_+d_-} \boldsymbol{\nu}\big)\big(1 - \boldsymbol{\nu}_k^\top \mathbf{I}_{d_+d_-} \boldsymbol{\nu}\big)\boldsymbol{\nu}\boldsymbol{\nu}^\top\big], \\ \Sigma_k &= \mathbf{I}_{d_+d_-} \boldsymbol{\Delta}^{-1} \boldsymbol{\Gamma}_k \boldsymbol{\Delta}^{-1} \mathbf{I}_{d_+d_-}. \end{aligned} \tag{13}$$

Remark 4: If the adjacency matrix $\mathbf{A}$ is sampled from an SBM parameterized by the block connectivity probability matrix $\mathbf{B}$ in Eq. (3) and block assignment probabilities $\boldsymbol{\pi} = (\pi_1, \pi_2)$ with $\pi_1 + \pi_2 = 1$, then as a special case for Theorem 1 [27], [35], we have for each fixed index i,

$$\begin{aligned} \sqrt{n}\Big(\widehat{X}_i - p\Big) &\xrightarrow{d} \mathcal{N}\Big(0, \sigma_p^2\Big) \qquad \text{if } \ X_i = p, \\ \sqrt{n}\Big(\widehat{X}_i - q\Big) &\xrightarrow{d} \mathcal{N}\Big(0, \sigma_q^2\Big) \qquad \text{if } \ X_i = q. \end{aligned} \tag{14}$$

where

$$\begin{aligned} \sigma_p^2 &= \frac{\pi_1 p^4(1-p^2) + \pi_2 pq^3(1-pq)}{\left[\pi_1 p^2 + \pi_2 q^2\right]^2}, \\ \sigma_q^2 &= \frac{\pi_1 p^3 q(1-pq) + \pi_2 q^4(1-q^2)}{\left[\pi_1 p^2 + \pi_2 q^2\right]^2}. \end{aligned} \tag{15}$$

Now for a K-block SBM, let $\mathbf{B} \in [0,1]^{K\times K}$ be the block connectivity probability matrix and $\boldsymbol{\pi} \in [0,1]^K$ be the vector of block assignment probabilities. Given an n vertex instantiation of the SBM parameterized by $\mathbf{B}$ and $\boldsymbol{\pi}$, for sufficiently large n, the large sample optimal error rate for estimating the block assignments using ASE can be measured via Chernoff information as [27], [35]

$$\begin{aligned} \rho = \min_{k\neq l} \sup_{t\in(0,1)}\Bigg[&\frac{1}{2} nt(1-t)(\boldsymbol{\nu}_k - \boldsymbol{\nu}_\ell)^\top \Sigma_{k\ell}^{-1}(t)(\boldsymbol{\nu}_k - \boldsymbol{\nu}_\ell) \\ &+ \frac{1}{2}\log\frac{|\Sigma_{k\ell}(t)|}{|\Sigma_k|^t |\Sigma_\ell|^{1-t}}\Bigg] \end{aligned} \tag{16}$$

where $\Sigma_{k\ell}(t) = t\Sigma_k + (1-t)\Sigma_\ell$, Σ_k and Σ_ℓ are defined as in Eq. (13). Also note that as $n \to \infty$, the logarithm term in Eq. (16) will be dominated by the other term. Then we have the Chernoff ratio as

$$\rho^* = \frac{\rho_1^*}{\rho_2^*} \to \frac{\min\limits_{k\neq\ell}\sup\limits_{t\in(0,1)}\left[t(1-t)(\boldsymbol{\nu}_{1,k}-\boldsymbol{\nu}_{1,\ell})^\top \boldsymbol{\Sigma}_{1,k\ell}^{-1}(t)(\boldsymbol{\nu}_{1,k}-\boldsymbol{\nu}_{1,\ell})\right]}{\min\limits_{k\neq\ell}\sup\limits_{t\in(0,1)}\left[t(1-t)(\boldsymbol{\nu}_{2,k}-\boldsymbol{\nu}_{2,\ell})^\top \boldsymbol{\Sigma}_{2,k\ell}^{-1}(t)(\boldsymbol{\nu}_{2,k}-\boldsymbol{\nu}_{2,\ell})\right]}. \quad (17)$$

Here ρ_1^* and ρ_2^* are associated with the Algorithms 1 and 2 respectively. If $\rho^* > 1$, then Algorithm 1 is preferred, otherwise Algorithm 2 is preferred.

B. Two-Block Rank One Model With One Binary Covariate

As an illustration of using Chernoff ratio in Eq. (17) to compare the performance of Algorithms 1 and 2 for estimating the induced block assignments, we consider the two-block SBM with one binary covariate as in Example 2.

Proposition 1: For two-block rank one model with one binary covariate as in Example 2 with the assumption that $n_i = n\pi_i$ and $n_{Z,j} = n\pi_{Z,j}$ for $i \in \{1,2\}$ and $j \in \{1,2,3,4\}$ where $\boldsymbol{\pi} = (\frac{1}{2},\frac{1}{2})$ and $\boldsymbol{\pi}_Z = (\frac{1}{4},\frac{1}{4},\frac{1}{4},\frac{1}{4})$, there is no tractable closed-form for Chernoff ratio as in Eq. (17) but numerical experiments can be used to obtain ρ_1^* and ρ_2^* can be derived analytically as

$$\rho_2^* = \frac{(p-q)^2(p^2+q^2)^2}{2\left[\sqrt{p^2\phi_p + q^2\phi_{pq}} + \sqrt{q^2\phi_q + p^2\phi_{pq}}\right]^2}, \quad (18)$$

where σ_p^2, σ_q^2 are defined as in Eq. (15) and

$$\begin{aligned}\phi_p &= p^2(1-p^2),\\ \phi_q &= q^2(1-q^2),\\ \phi_{pq} &= pq(1-pq).\end{aligned} \quad (19)$$

Technical details of Proposition 1 can be found in the appendices. Figure 1 shows the Chernoff ratio when we fix $p = 0.3$ and take $q \in (0.3, 0.7), \beta \in (0.1, 0.5)$ in the two-block rank one models with one binary covariate. $\rho^* < 1$ for most of the region while $\rho^* > 1$ only when q and β are relatively large. Recall that the performance of Algorithm 1 highly depends on the estimated block connectivity probability matrix $\widehat{\mathbf{B}}_Z$. Large q and β lead to a relatively well-structured $\widehat{\mathbf{B}}_Z$ and thus Algorithm 1 can have better performance in this region.

C. Two-Block Homogeneous Model With One Binary Covariate

Now we consider the two-block SBM with one binary covariate parameterized by the block connectivity probability matrix $\mathbf{B}_Z$ as in Eq. (7).

Corollary 1: For two-block homogeneous model with one binary covariate as in Example 3 with the assumption that $n_i = n\pi_i$ and $n_{Z,j} = n\pi_{Z,j}$ for $i \in \{1,2\}$ and $j \in \{1,2,3,4\}$ where $\boldsymbol{\pi} = (\frac{1}{2},\frac{1}{2})$ and $\boldsymbol{\pi}_Z = (\frac{1}{4},\frac{1}{4},\frac{1}{4},\frac{1}{4})$. The Chernoff ratio as in Eq. (17) can be derived analytically as

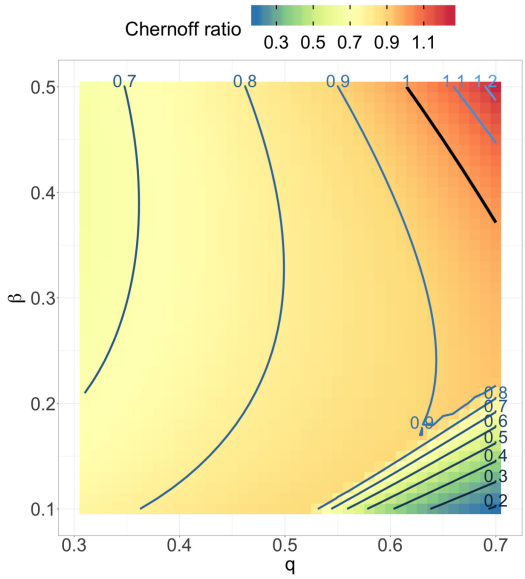


Fig. 1. Chernoff ratio as in Eq. (17) for two-block rank one model, $p = 0.3$, $q \in (0.3, 0.7), \beta \in (0.1, 0.5), \boldsymbol{\pi} = (\frac{1}{2},\frac{1}{2}), \boldsymbol{\pi}_Z = (\frac{1}{4},\frac{1}{4},\frac{1}{4},\frac{1}{4})$.

$$\rho^* = \frac{\rho_1^*}{\rho_2^*} \to \begin{cases} \frac{\beta^2(\phi_a+\phi_b)}{(a-b)^2(\phi_a+\phi_b+\phi_\beta)} & \text{if } \beta \le a-b \\ \frac{\phi_a+\phi_b}{\phi_a+\phi_b+\phi_\beta} & \text{if } \beta > a-b \end{cases}, \quad (20)$$

where

$$\begin{aligned}\phi_a &= a(1-a),\\ \phi_b &= b(1-b),\\ \phi_\beta &= \beta(1-a-b-\beta).\end{aligned} \quad (21)$$

Technical details of Corollary 1 can be found in appendices. Figure 2 shows Chernoff ratio when we fix $b = 0.1$ and take $a \in (0.1, 0.5), \beta \in (0.1, 0.5)$ in the two-block homogeneous models with one binary covariate. Again $\rho^* < 1$ for most of the region while $\rho^* > 1$ only when a and β are relatively large, which agrees with the general expression for Chernoff ratio as in Corollary 1. According to Eq. (20), we can have $\rho^* > 1$ only when $\phi_\beta < 0$ and this can happen only when a and β are relatively large. This implies that Algorithm 2 is often preferred for estimating the induced block assignments.

D. K-Block Homogeneous Model With One Binary Covariate

We extend the discussion from the two-block homogeneous model to the K-block homogeneous model with one binary covariate.

Theorem 2: For the K-block homogeneous balanced model with one binary covariate as in Remark 2 with the assumption that that $n_i = n\pi_i$ and $n_{Z,j} = n\pi_{Z,j}$ for $i \in \{1,\ldots,K\}$ and $j \in \{1,\ldots,2\,K\}$ where $\boldsymbol{\pi} = (\frac{1}{K},\ldots,\frac{1}{K})$ and $\boldsymbol{\pi}_Z = (\frac{1}{2\,K},\ldots,\frac{1}{2\,K})$. The Chernoff ratio as in (17) can be derived analytically as

$$\rho^* = \frac{\rho_1^*}{\rho_2^*} \to \begin{cases} \frac{K^2\beta^2(\phi_a+\phi_b)}{2(a-b)^2 D_4} & \text{if } \delta \le 0 \\ \frac{\phi_a+\phi_b}{\phi_a+\phi_b+\phi_\beta} & \text{if } \delta > 0 \end{cases}, \quad (22)$$

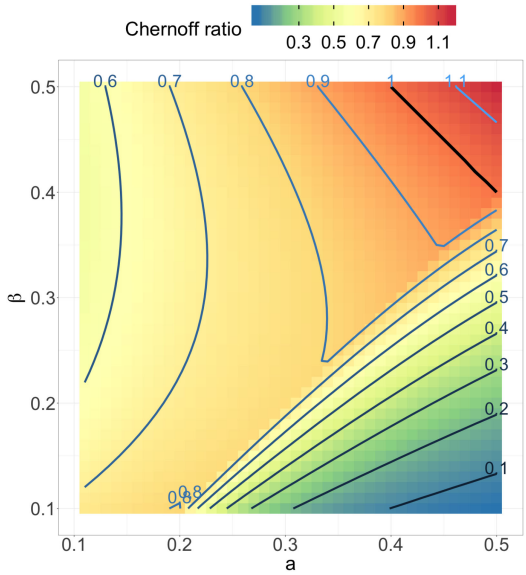


Fig. 2. Chernoff ratio as in Eq. (17) for two-block homogeneous models. $b = 0.1, a \in (0.1, 0.5), \beta \in (0.1, 0.5), \boldsymbol{\pi} = (\frac{1}{2}, \frac{1}{2}), \boldsymbol{\pi}_Z = (\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4})$.

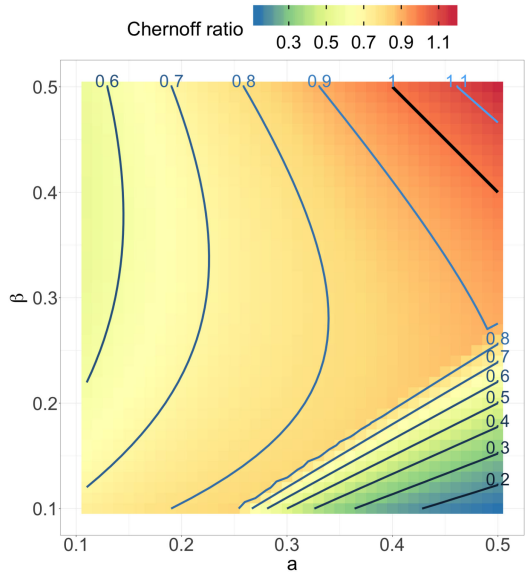


Fig. 3. Chernoff ratio as in Eq. (17) for four-block homogeneous models. $b = 0.1, a \in (0.1, 0.5), \beta \in (0.1, 0.5), \boldsymbol{\pi} = (\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}), \boldsymbol{\pi}_Z = (\frac{1}{8}, \ldots, \frac{1}{8})$.

where $\phi_a, \phi_b, \phi_\beta$ are defined as in (21) and

$$\begin{aligned} D_3 &= K - 2a - 2(K-1)b - K\beta, \\ D_4 &= 2\phi_a + 2(K-1)\phi_b + \beta D_3, \\ \delta &= K^2\beta^2(\phi_a + \phi_b + \phi_\beta) - 2(a-b)^2 D_4. \end{aligned} \tag{23}$$

Remark 5: Theorem 2 generalizes Corollary 1 beyond $K = 2$.

Technical details of Theorem 2 can be found in the appendices. Fig. 3 shows Chernoff ratio when we fix $b = 0.1$ and take $a \in (0.1, 0.5), \beta \in (0.1, 0.5)$ in the four-block homogeneous models with one binary covariate. $\rho^* < 1$ for most of the region while $\rho^* > 1$ only when a and β are relatively large. This implies again that Algorithm 2 is often preferred for estimating the induced block assignments.

V. Simulations and Real Data Examples

In addition to comparing the two algorithms' performance analytically via Chernoff ratio, we also compare Algorithms 1 and 2 by empirical clustering results. Recall that the analytic comparison via Chernoff ratio is based on the limiting results of ASE for SBM when the number of vertices $n \to \infty$. The comparison via empirical clustering results can measure the performance of these two algorithms for finite n. The implementation of Algorithms 1 and 2 can be found at https://github.com/CongM/sbm-cov. Details about the experiments can be found in appendices.

As an illustration of this correspondence, we start with the setting related to "A" ($p = 0.3, q = 0.668, \beta = 0.49$ with $\rho^* = 1.1 > 1$) and "B" ($p = 0.3, q = 0.564, \beta = 0.49$ with $\rho^* = 0.91 < 1$) in left panel of Figure 4 for two-block rank one model with one binary covariate $\mathbf{Z} \in \{1, 2\}^n$. We consider the balanced case where $n_1 = n_2 = \frac{n}{2}$ and $n_{Z,1} = n_{Z,2} = n_{Z,3} = n_{Z,4} = \frac{n}{4}$. For each $n \in \{100, 140, 180, 220, 260\}$, we simulate 100 adjacency matrices with $\frac{n}{2}$ vertices in each block and generate binary covariate with $\frac{n}{4}$ vertices having each value of $\mathbf{Z}$ within each block. We then apply Algorithms 1 and 2 (with β and $\widehat{\beta}$ in Step 3 respectively) using embedding dimension $\widehat{d} = 3$ to estimate the induced block assignments where adjusted Rand index (ARI) [38] is used to measure the performance (ARI can take values from -1 to 1 where larger value indicates a better alignment of the empirical clustering and the "truth"). The upper right panel in Figure 4 shows that although $\rho^* > 1$ and Algorithm 1 should be preferred in terms of Chernoff ratio, the ARI suggests that Algorithm 2 is preferred. While the Chernoff ratio is, in fact, a limit (computed as the sample size n increases to infinity), the region for which $\rho^* > 1$ is so easy for clustering—e.g., $q - p$ is large for "A"—that both algorithms are essentially perfect even for small n. The lower right panel in Fig. 4 shows that Algorithm 2 tends to have better performance than Algorithm 1, which agrees with the Chernoff ratio as in left figure where $\rho^* < 1$ and Algorithm 2 is preferred.

To further investigate the flexibility of our models and algorithms, we also discuss categorical vertex covariate.

A. Two-Block Rank One Model With One Five-Categorical Covariate

Consider the two-block rank one model with one five-categorical covariate $\mathbf{Z} \in \{1, 2, 3, 4, 5\}^n$, i.e., we have the block connectivity probability matrix $\mathbf{B}_Z \in [0, 1]^{10 \times 10}$ with similar structure as in (6).

We first fix $p = 0.3, \beta = 0.4$ and consider $q \in \{0.35, 0.375, 0.4, 0.425, 0.45\}$. For each q, we simulate 100 adjacency matrices with 1000 vertices in each block and generate five-categorical covariate with 200 vertices having each value of $\mathbf{Z}$ within each block. We then apply Algorithms 1 and 2 (with β and $\widehat{\beta}$ in Step 3 respectively) using embedding dimension $\widehat{d} = 6$ to estimate the induced block assignments. Fig. 5(a) shows that both algorithms estimate more accurate induced block

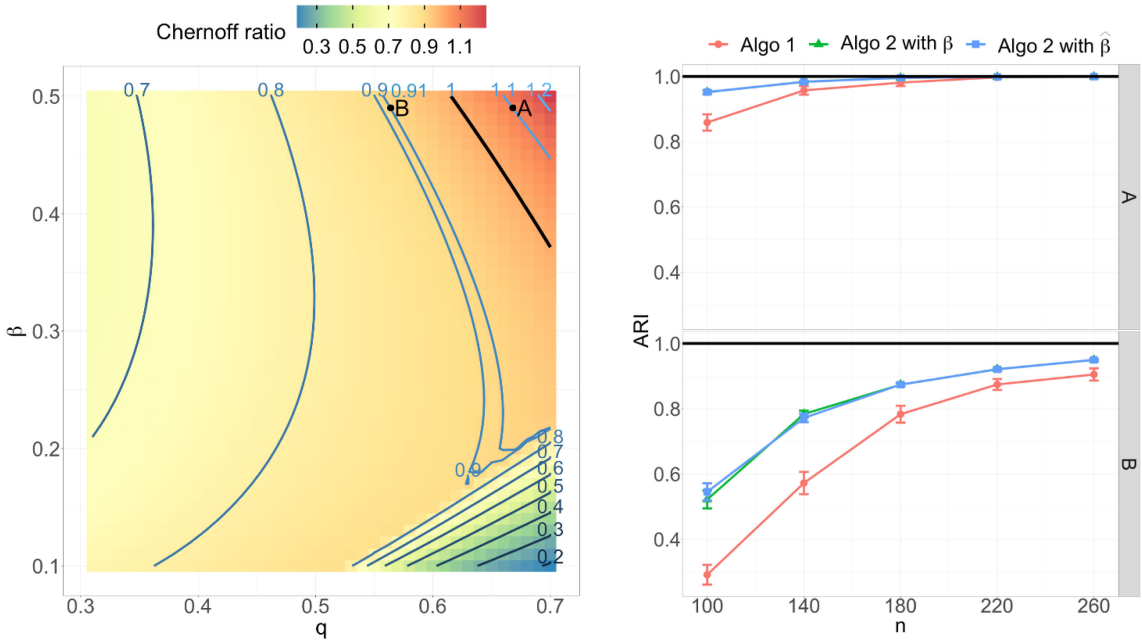
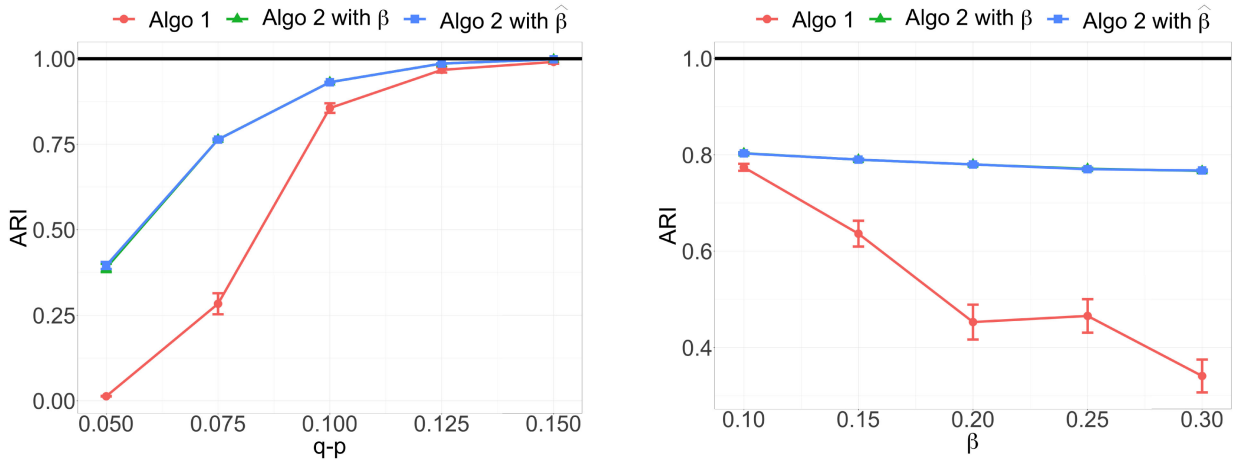


Fig. 4. Correspondence between Chernoff analysis and simulations.



(a) ARI as latent positions of two induced blocks move away from each other with $\beta = 0.4$.

(b) ARI as β increases with $p = 0.3, q = 0.375$.

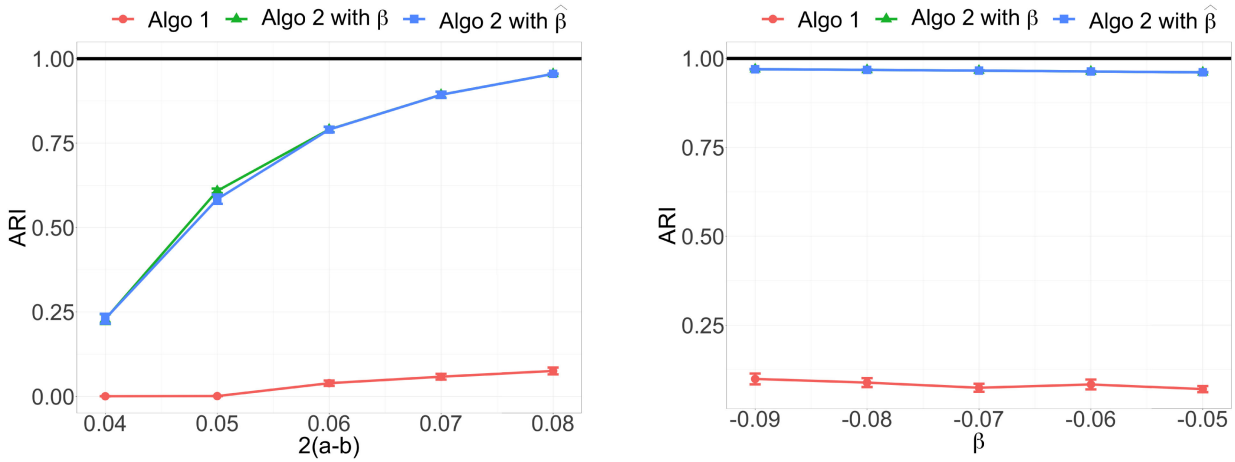
Fig. 5. Simulations for two-block rank one model with one five-categorical covariate, balanced case.

assignments as the latent positions of two induced block move away from each other, i.e., two induced blocks tend to be more separate, and Algorithm 2 can have better performance than Algorithm 1.

Next we fix $p = 0.3, q = 0.375$ and consider $\beta \in \{0.1, 0.15, 0.2, 0.25, 0.3\}$. For each β, we simulate 100 adjacency matrices with 1000 vertices in each block and generate five-categorical covariate with 200 vertices having each value of $\mathbf{Z}$ within each block. We then apply both algorithms (with β and $\widehat{\beta}$ in Step 3 of Algorithm 2 respectively) using embedding dimension $\widehat{d} = 6$ to estimate the induced block assignments. Fig. 5(b) shows Algorithm 1 can only estimate accurate induced block assignments when β is relatively small while Algorithm 2 can estimate accurate induced block assignments no matter β is small or large. Intuitively, as Algorithm 1 directly estimates the induced block assignments, when β is relatively large, i.e., vertex covariates can affect block structure significantly, it lacks the ability to distinguish this effect. However, Algorithm 2 can use additional information from vertex covariates to estimate β, taking this effect into consideration when estimating the induced block assignments. Again, the overall performance of Algorithm 2 is better than that of Algorithm 1.

B. Two-Block Homogeneous Model With One Five-Categorical Covariate

We now consider the two-block homogeneous model with one five-categorical covariate $\mathbf{Z} \in \{1, 2, 3, 4, 5\}^n$, i.e., we have the block connectivity probability matrix $\mathbf{B}_Z \in [0, 1]^{10\times 10}$ with



(a) ARI as latent positions of two induced blocks move away from each other with $\beta = 0.2$.

(b) ARI as β increases with $a = 0.135, b = 0.1$.

Fig. 6. Simulations for two-block homogeneous model with one five-categorical covariate, balanced case.

the similar structure as in (7). Note that we can re-write $\mathbf{B}$ like (3) as

$$\mathbf{B} = \boldsymbol{\nu}\boldsymbol{\nu}^\top = \begin{bmatrix} a & b \\ b & a \end{bmatrix} \text{ with } \boldsymbol{\nu} = \begin{bmatrix} \sqrt{a} & 0 \\ \frac{b}{\sqrt{a}} & \sqrt{\frac{(a-b)(a+b)}{a}} \end{bmatrix}. \quad (24)$$

With these canonical latent positions, the distance between two induced blocks can be measured by

$$\left(\sqrt{a} - \frac{b}{\sqrt{a}}\right)^2 + \left(0 - \sqrt{\frac{(a-b)(a+b)}{a}}\right)^2 = 2(a-b). \quad (25)$$

We first fix $b = 0.1, \beta = 0.2$ and consider $a \in \{0.12, 0.125, 0.13, 0.135, 0.14\}$. For each a, we simulate 100 adjacency matrices with 1000 vertices in each block and generate five-categorical covariate with 200 vertices having each value of $\mathbf{Z}$ within each block. We then apply both algorithms (with β and $\widehat{\beta}$ in Step 3 of Algorithm 2 respectively) using embedding dimension $\widehat{d} = 6$ to estimate the induced block assignments. Fig. 6(a) shows that both algorithms estimate more accurate induced block assignments as the latent positions of two induced block move away from each other, i.e., two induced blocks tend to be more separate as measured by (25), and Algorithm 2 can have much better performance. Recall that Algorithm 1 tries to estimate the induced block assignments by clustering the diagonal of $\widehat{\mathbf{B}}_Z$ and re-assigning the block assignments including the vertex covariate effect. For the homogeneous model, the diagonal of $\mathbf{B}_Z$ are all the same, which can make it hard for Algorithm 1 to accurately estimate the induced block assignments. But Algorithm 2 is not affected by the homogeneous structure since it estiamtes the vertex covariate effect first and then estimates the induced block assignments by clustering the estimated latent positions like the canonical ones in (24).

Next we fix $a = 0.135, b = 0.1$ and consider $\beta \in \{-0.09, -0.08, -0.07, -0.06, -0.05\}$. For each β, we also simulate 100 adjacency matrices with 1000 vertices in each block and generate five-categorical covariate with 200 vertices having each value of $\mathbf{Z}$ within each block. We then apply both algorithms (with β and $\widehat{\beta}$ in Step 3 of Algorithm 2 respectively) using embedding dimension $\widehat{d} = 6$ to estimate the induced block assignments. Fig. 6(b) shows that both algorithms are relative stable for this homogeneous model if we fix a and b, due to the special structure. Still, Algorithm 2 can have much better performance than Algorithm 1.

C. Connectome Data

We consider a real data example on diffusion MRI connectome datasets [1]. There are 114 graphs (connectomes) estimated by the NDMG pipeline [39] in this data set where vertices represent brain sub-regions defined via spatial proximity and edges represent tensor-based fiber streamlines connecting these sub-regions. Each vertex in these graphs also has a {Left, Right} hemisphere label and a {Gray, White} tissue label. We treat one label as the induced block and the other one as the vertex covariate.

Each of the 114 connectomes (the number of vertices n varies from 23728 to 42022) is represented by a point in Fig. 7 with $x = \text{ARI}(\text{Algo2, LR}) - \text{ARI}(\text{Algo1, LR})$ and $y = \text{ARI}(\text{Algo2, GW}) - \text{ARI}(\text{Algo1, GW})$ where ARI(Algo1, LR) denotes the ARI when we apply Algorithm 1 and treat {Left, Right} as the induced block (with analogous notation for the rest). We see that most of the points lie in the (+,+) quadrant, indicating $\text{ARI}(\text{Algo2, LR}) > \text{ARI}(\text{Algo1, LR})$ and $\text{ARI}(\text{Algo2, GW}) > \text{ARI}(\text{Algo1, GW})$. That is, Algorithm 2 is better at estimating the induced block assignments for this real application. Note that this claim holds no matter which label is treated as the induced block. This again emphasizes the importance of distinguishing different factors that can affect block structure in graphs. Algorithm 2 is able to identify particular block structure by using the observed vertex covariate information. That is, it is more likely to discover the {Left, Right} structure after accounting for the effect of {Gray, White} label and more likely to discover the {Gray, White} structure after accounting for the effect of {Left, Right} label.

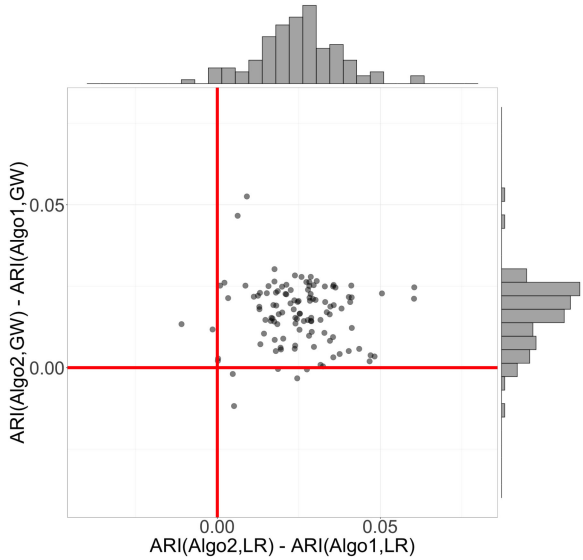


Fig. 7. Algorithms' comparative performance on connectome data.

D. Social Network Data

We also utilize three social network datasets to compare our methods with several existing methods that also incorporate vertex covariates and can be scaled to deal with relatively large networks. Specifically, we compare with spectral clustering with adjacency matrix only (SCA) and covariates only (SCC) [17], pairwise covariates-adjusted stochastic blockmodel via maximum likelihood estimation (PCABM.MLE) and spectral clustering with adjustment (PCABM.SCWA) [15], covariate-assisted spectral clustering (CASC) [14].

- LastFM asia social network dataset [2], [4]: there are 7624 vertices that represent LastFM users from asian countries and 27806 edges that represent mutual follower relationships. We treat the location of users, which are derived from the country field for each user, as the induced block. For the vertex covariate, we focus on the number of artists liked by users, which is discretized into four categories {0–200, 200–400, 400–600, 600+}.
- Facebook large page-page network dataset [2], [3]: there are 22470 vertices that represent official Facebook pages and 171002 edges that represent mutual likes. We treat four page types {Politician, Governmental Organization, Television Show, Company}, which are defined by Facebook, as the induced block. For the vertex covariate, we focus on the number of descriptions created by page owners to summarize the purpose of the site, which is discretized into two categories {0–15, 15+}.
- GitHub social network dataset [2], [3]: there are 37700 vertices that represent GitHub developers and 289003 edges that represent mutual follower relationships. We treat two developer types {Web, Machine Learning}, which are derived from the job title of each developer, as the induced block. For the vertex covariate, we focus on the number of repositories starred by developers, which is discretized into two categories {0–18, 18+}.

TABLE I
ALGORITHMS' PERFORMANCE ON SOCIAL NETWORK DATA VIA ARI

	LastFM	Facebook	GitHub
SCA [17]	0.229	0.050	0.000
SCC [17]	0.012	0.038	0.001
PCABM.SCWA [15]	0.008	-0.002	0.000
PCABM.MLE [15]	0.000	0.004	-0.002
CASC [14]	0.020	0.053	-0.043
Algo 1 (ours)	0.090	0.036	0.001
Algo 2 (ours)	**0.297**	**0.076**	**0.013**

Table I summarizes the algorithms' comparative performances. Algorithm 2 is better at estimating the induced block assignments for all 3 datasets. This again suggests that we can better detect the block structure after accounting for the information contained in vertex covariates with our methods.

In real data, we may not have ground truth for the block structure. Our findings suggest that we are able to discover block structure by using observed vertex covariates, which can lead to meaningful insights in widely varying applications. That is, we can better reveal underlying block structure and thus better understand the data by accounting for the vertex covariate effect.

VI. DISCUSSION

We study the problem of community detection for SBMs with vertex covariates. Specifically, we consider two model-based spectral algorithms to assess the effect of observed and unobserved vertex heterogeneity on block structure in graphs. The main difference of these two algorithms in estimating the induced block assignments is whether we estimate the vertex covariate effect using the observed covariate information. To analyze the algorithms' performance, we employ Chernoff information and derive the Chernoff ratio expression for homogeneous balanced model. We also simulate multiple adjacency matrices with varied type of covariates to compare the algorithms' performance via empirical clustering accuracy measured by ARI. In addition, we conduct real data analysis on diffusion MRI connectome datasets and social network datasets. Analytic results, simulations, and real data examples suggest that the second algorithm is often preferred: we can better estimate the induced block assignments and reveal underlying block structure by using additional information contained in vertex covariates. Our findings also emphasize the importance of distinguishing between observed and unobserved factors that can affect block structure in graphs.

We focus on the model specified as in Definition 3 and Remark 1 where indicator function is used to measure the vertex covariate effect and identity function is used as the link between edge probabilities and latent positions. We also investigate the flexibility of our models and algorithms by considering categorical vertex covariates. The extension from discrete vertex covariates to continuous vertex covariates is under investigation, for instance, via latent structure models [40]. The indicator function is used to measure the vertex covariate

effect for binary and generally categorical vertex covariates under the intuition that vertices having the same covariates are more likely to form an edge between them and different functions can be adopted for the continuous vertex covariates following the similar intuition. For example, similarity and distance functions can be chosen according to the nature of different vertex covariates to measure how they can influence graph structure. One other extension is to replace the identity link with, say, the logit link function. The idea of using Chernoff information to compare algorithms' performance can be adopted for all the above generalizations and numerical evaluations can be obtained in the absence of closed-form expressions, which in turn can reveal how graph structure will affect our algorithms and provide guidelines for real application.

REFERENCES

[1] C. E. Priebe *et al.*, "On a two-truths phenomenon in spectral graph clustering," *Proc. Nat. Acad. Sci. USA*, vol. 116, no. 13, pp. 5995–6000, 2019.

[2] J. Leskovec and A. Krevl, "SNAP datasets: Stanford large network dataset collection," Stanford, CA, USA, Jun. 2014. [Online]. Available: http://snap.stanford.edu/data

[3] B. Rozemberczki, C. Allen, and R. Sarkar, "Multi-scale attributed node embedding," *J. Complex Netw.*, vol. 9, no. 2, p. cnab014, 2019.

[4] B. Rozemberczki and R. Sarkar, "Characteristic functions on graphs: Birds of a feather, from statistical descriptors to parametric models," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Manage*. ACM, 2020, pp. 1325–1334.

[5] P. W. Holland, K. B. Laskey, and S. Leinhardt, "Stochastic blockmodels: First steps," *Social Netw.*, vol. 5, no. 2, pp. 109–137, 1983.

[6] L. Hao, A. Mele, J. Cape, A. Athreya, C. Mu, and C. E. Priebe, "Latent communities in employment relation and wage distribution: A network approach," submitted, 2020.

[7] C. R. Shalizi and E. McFowland III, "Estimating causal peer influence in homophilous social networks by inferring latent locations," 2016, *arXiv:1607.06565*.

[8] E. Abbe, "Community detection and stochastic block models: Recent developments," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 6446–6531, 2017.

[9] B. Karrer and M. E. Newman, "Stochastic blockmodels and community structure in networks," *Phys. Rev. E*, vol. 83, no. 1, 2011, Art. no. 016107.

[10] S. Fortunato and D. Hric, "Community detection in networks: A user guide," *Phys. Rep.*, vol. 659, pp. 1–44, 2016.

[11] D. S. Choi, P. J. Wolfe, and E. M. Airoldi, "Stochastic blockmodels with a growing number of classes," *Biometrika*, vol. 99, no. 2, pp. 273–284, 2012.

[12] S. Roy, Y. Atchadé, and G. Michailidis, "Likelihood inference for large scale stochastic blockmodels with covariates based on a divide-and-conquer parallelizable algorithm with communication," *J. Comput. Graphical Statist.*, vol. 28, no. 3, pp. 609–619, 2019.

[13] T. M. Sweet, "Incorporating covariates into stochastic blockmodels," *J. Educ. Behav. Statist.*, vol. 40, no. 6, pp. 635–664, 2015.

[14] N. Binkiewicz, J. T. Vogelstein, and K. Rohe, "Covariate-assisted spectral clustering," *Biometrika*, vol. 104, no. 2, pp. 361–377, 2017.

[15] S. Huang and Y. Feng, "Pairwise covariates-adjusted block model for community detection," 2018, *arXiv:1807.03469*.

[16] A. Mele, L. Hao, J. Cape, and C. E. Priebe, "Spectral inference for large stochastic blockmodels with nodal covariates," 2019, *arXiv:1908.06438*.

[17] U. V. Luxburg, "A tutorial on spectral clustering," *Statist. Comput.*, vol. 17, no. 4, pp. 395–416, 2007.

[18] V. Lyzinski, D. L. Sussman, M. Tang, A. Athreya, and C. E. Priebe, "Perfect clustering for stochastic blockmodel graphs via adjacency spectral embedding," *Electron. J. Statist.*, vol. 8, no. 2, pp. 2905–2922, 2014.

[19] V. Lyzinski, M. Tang, A. Athreya, Y. Park, and C. E. Priebe, "Community detection and classification in hierarchical stochastic blockmodels," *IEEE Trans. Netw. Sci. Eng.*, vol. 4, no. 1, pp. 13–26, Jan.–Mar. 2016.

[20] F. McSherry, "Spectral partitioning of random graphs," in *Proc. 42nd IEEE Symp. Found. Comput. Sci.*, 2001, pp. 529–537.

[21] K. Rohe, S. Chatterjee, and B. Yu, "Spectral clustering and the high-dimensional stochastic blockmodel," *Ann. Statist.*, vol. 39, no. 4, pp. 1878–1915, 2011.

[22] V. Lyzinski, K. Levin, and C. E. Priebe, "On consistent vertex nomination schemes," *J. Mach. Learn. Res.*, vol. 20, no. 69, pp. 1–39, 2019.

[23] M. Tang, A. Athreya, D. L. Sussman, V. Lyzinski, and C. E. Priebe, "A nonparametric two-sample hypothesis testing problem for random graphs," *Bernoulli*, vol. 23, no. 3, pp. 1599–1630, 2017.

[24] S. Wang, J. Arroyo, J. T. Vogelstein, and C. E. Priebe, "Joint embedding of graphs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 4, pp. 1324–1336, Apr. 2021.

[25] D. L. Sussman, M. Tang, D. E. Fishkind, and C. E. Priebe, "A consistent adjacency spectral embedding for stochastic blockmodel graphs," *J. Amer. Stat. Assoc.*, vol. 107, no. 499, pp. 1119–1128, 2012.

[26] A. Athreya, C. E. Priebe, M. Tang, V. Lyzinski, D. J. Marchette, and D. L. Sussman, "A limit theorem for scaled eigenvectors of random dot product graphs," *Sankhya A*, vol. 78, no. 1, pp. 1–18, 2016.

[27] M. Tang and C. E. Priebe, "Limit theorems for eigenvectors of the normalized laplacian for random graphs," *Ann. Statist.*, vol. 46, no. 5, pp. 2360–2415, 2018.

[28] J. Cape, M. Tang, and C. E. Priebe, "On spectral embedding performance and elucidating network structure in stochastic blockmodel graphs," *Netw. Sci.*, vol. 7, no. 3, pp. 269–291, 2019.

[29] P. D. Hoff, A. E. Raftery, and M. S. Handcock, "Latent space approaches to social network analysis," *J. Amer. Stat. Assoc.*, vol. 97, no. 460, pp. 1090–1098, 2002.

[30] M. S. Handcock, A. E. Raftery, and J. M. Tantrum, "Model-based clustering for social networks," *J. Roy. Stat. Society: Ser. A (Statistics Society)*, vol. 170, no. 2, pp. 301–354, 2007.

[31] P. Rubin-Delanchy, C. E. Priebe, M. Tang, and J. Cape, "A statistical interpretation of spectral embedding: The generalised random dot product graph," 2017, *arXiv:1709.05506*.

[32] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Berlin, Germany: Springer, 2009.

[33] I. T. Jolliffe and J. Cadima, "Principal component analysis: A review and recent developments," *Philos. Trans. Roy. Soc. A, Mathematical, Phys. Eng. Sci.*, vol. 374, no. 2065, 2016, Art. no. 20150202.

[34] M. Zhu and A. Ghodsi, "Automatic dimensionality selection from the scree plot via the use of profile likelihood," *Comput. Statist. Data Anal.*, vol. 51, no. 2, pp. 918–930, 2006.

[35] A. Athreya *et al.*, "Statistical inference on random dot product graphs: A survey," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 8393–8484, 2017.

[36] H. Chernoff, "A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations," *Ann. Math. Statist.*, vol. 23, no. 4, pp. 493–507, 1952.

[37] H. Chernoff, "Large-sample theory: Parametric case," *Ann. Math. Statist.*, vol. 27, no. 1, pp. 1–22, 1956.

[38] L. Hubert and P. Arabie, "Comparing partitions," *J. Classification*, vol. 2, no. 1, pp. 193–218, 1985.

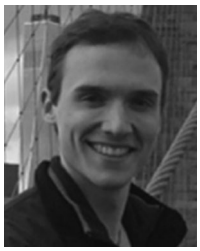
[39] G. Kiar *et al.*, "A high-throughput pipeline identifies robust connectomes but troublesome variability," 2018, Art. no. 188706.

[40] A. Athreya, M. Tang, Y. Park, and C. E. Priebe, "On estimation and inference in latent structure random graphs," *Stat. Sci.*, vol. 36, no. 1, pp. 68–88 2021.

Cong Mu received the B.S. degree in statistics from Sun Yat-Sen University, Guangzhou, China, in 2017, the M.S.E. degree in applied mathematics and statistics in 2019 from Johns Hopkins University, Baltimore, MD, USA, where he is currently working toward the Ph.D. degree in applied mathematics and statistics. His research interests include high-dimensional and graph inference, and computer vision.

Angelo Mele received the B.A. (Laurea) degree in economics from Bocconi University, Milan, Italy in 2002, the M.A. in economics from New York University, New York, NY, USA, in 2005 and the Ph.D. degree in economics from the University of Illinois Urbana-Champaign, Champaign, IL, USA, in 2011. He is currently an associate professor of economics with Carey Business School, Johns Hopkins University, Baltimore, MD, USA. His research interests include the econometrics of network models, economics of social interactions, racial segregation and homophily, online contagion, and computational methods.

Lingxin Hao received the B.A. degree in english, South China Normal University, Guangzhou, China in 1982, the M.A. degree in sociology, Sun Yat-sen University, Guangzhou, in 1985, and the Ph.D. degree in sociology, the University of Chicago, Chicago, IL, USA, in 1990. Before joining the Johns Hopkins University in 1996, she was a Postdoctoral Fellow with the Labor and Population Program, RAND, and an assistant-to-associate professor with The University of Iowa, Iowa City, IA, USA. She is currently a professor of sociology and the Director of Hopkins Population Center. Dr. Hao has been the principle investigator for several multi-year projects supported by federal grants from NIH and NSF. From 2002 to 2003, she was a Residential Fellow of Russell Sage Foundation and a Residential Fellow of Spenser Foundation in 2007. Her research interests include social demography, social inequality, migration, family and public policy, sociology of education, and quantitative and computational methods.



Joshua Cape received the B.A. degree in mathematics and economics from Rhodes College, Memphis, TN, USA, in 2014, and the M.S.E. and Ph.D. degrees in applied mathematics and statistics from Johns Hopkins University, Baltimore, MD, USA, in 2016 and 2019, respectively. Between 2019 and 2020, he was a National Science Foundation Postdoctoral Research Fellow with the Department of Statistics, University of Michigan, Ann Arbor, MI, USA . He is currently an assistant professor of statistics with the University of Pittsburgh, Pittsburgh, PA, USA. His research interests include statistical machine learning, multivariate analysis, networks, and matrix analysis.

Avanti Athreya received the B.S. degree from Iowa State University, Ames, IA, USA, in 1997, the M.S. degree from the University of Washington, Seattle, WA, USA, in 2000, and the Ph.D. degree from the University of Maryland, College Park, MD, USA, in 2009. She was a visiting assistant professor with Duke University, Durham, NC, USA and a postdoctoral fellow with SAMSI prior to coming to Johns Hopkins University, Baltimore, MD, in 2011, where she is currently an assistant research Professor. Her research interests include random graph inference, probability, and stochastic processes.

Carey E. Priebe (Senior Member, IEEE) received the B.S. degree in mathematics from Purdue University, West Lafayette, IN, USA, in 1984, the M.S. degree in computer science from San Diego State University, San Diego, CA, USA, in 1988, and the Ph.D. degree in information technology (computational statistics) from George Mason University, Fairfax, VA, USA, in 1993. From 1985 to 1994, he was a Mathematician and Scientist was the US Navy Research and Development Laboratory system. Since 1994, he has been a professor with the Department of Applied Mathematics and Statistics, Johns Hopkins University (JHU), Baltimore, MD, USA. At JHU, he holds joint appointments with the Department of Computer Science, Department of Electrical and Computer Engineering, Center for Imaging Science, Human Language Technology Center of Excellence, and Whitaker Biomedical Engineering Institute. His research interests include computational statistics, kernel and mixture estimates, statistical pattern recognition, statistical image analysis, dimensionality reduction, model selection, and statistical inference for high-dimensional and graph data. He is a Lifetime Member of the IMS, an Elected Member of the ISI, and a Fellow of the ASA.