Informative core identification in complex networks

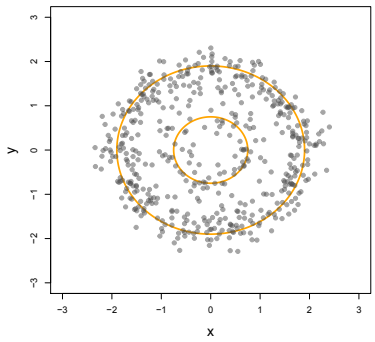
Ruizhong Miao
University of Virginia, Charlottesville, USA.
Tianxi Li
University of Virginia, Charlottesville, USA.
E-mail: tianxili@virginia.edu

Summary.
In analyzing a network, the core component with interesting structures is usually hidden while the rest of the network connections are not informative. The noises and bias introduced by the non-informative component can obscure the salient structure and limit many network modeling procedures' effectiveness. This paper introduces a novel core-periphery model for the non-informative periphery structure of networks without imposing a specific form of the core structure. We propose spectral algorithms for core identification as a data preprocessing step for general downstream network analysis tasks based on the model. The algorithms enjoy strong performance guarantees and are scalable for large networks. We evaluate the proposed methods by extensive simulation studies demonstrating advantages over multiple traditional core-periphery methods. We apply the proposed methods to extract the informative core structure from a citation network, which in turn gives more interpretable results in a downstream hierarchical community detection task.

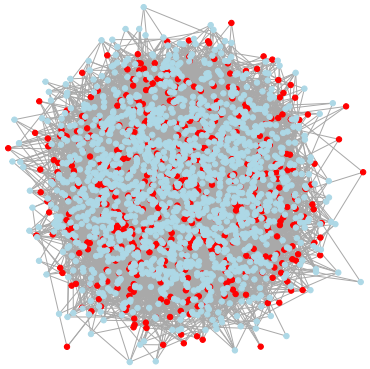
1. Introduction

Network data, representing interactions and relationships between units, have become ubiquitous with the rapid development of science and technology. The need to analyze such complex and structurally novel data has resulted in a rich body of new ideas and tools in physics, mathematics, statistics, computer science, and the social sciences. Given that complex network structures are typically noisy and complicated, treating the network as a random instantiation of a probabilistic model is a popular means of learning networks' structural properties while ignoring noises. This approach appeared in work as early as that of Erdös [1959]. Later efforts from Aldous [1981] and Hoover [1979] further expanded the foundation for more flexible random network modeling. More recently, significant progress has rendered network analysis more computationally efficient and scientifically interpretable with theoretical guarantees [Albert and Barabási, 2002, Hoff et al., 2002, Bickel and Chen, 2009, Zhao et al., 2012, Newman, 2016, Gao et al., 2017, Athreya et al., 2017, Mukherjee et al., 2018]. These methods have been adopted to solve many important problems. Empirically, however, they fail to learn structural information effectively in many applications. One critical issue complicating this matter in practice is the relative lack of interesting or informative structures in large-scale networks. Throughout this paper, the term "*informative*" generally means that the structure presents a special pattern of connections that researchers are interested in identifying and understanding. Examples of informative network structures include, but

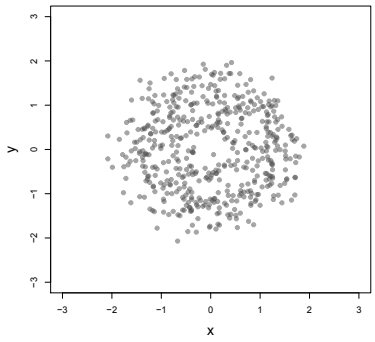
are not limited to transitivity patterns, community structures, and latent space topologies. While most models assume certain particular informative network structures, the presumed structures may only be valid for a subnetwork, leaving the rest of the network noninformative. For instance, Ugander et al. [2013] observed that the lower-order moments in 100 Facebook subnetworks were highly similar to the Erdös-Renyi model. Gao and Lafferty [2017] later tested these networks and found that most of them were indistinct from purely random connections and thus presented no especially interesting structure. As additional examples, Wang and Rohe [2016] and Li et al. [2020b,a] both applied preprocessing to remove a subset of nodes prior to community detection.



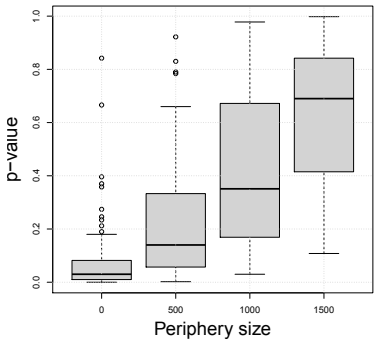
(a) Estimated latent vectors without periphery.



(b) Core nodes (red) with periphery (blue) nodes.



(c) Estimated latent vectors with 1500 periphery nodes.



(d) P-values of the unimodal tests on latent vector norms.

Fig. 1. The impact of periphery components on learning latent space topology. (a) Estimated latent positions of 500 nodes without any periphery nodes — a two-layer pattern of latent vectors is observable; (b) Network of the same 500 core nodes with 1500 periphery nodes; (c) Estimated latent positions based on the network in (b); (d) p-values of 100 replications, based on the unimodal test for the latent vector norms — a small p-value indicates a multimodal distribution, though this procedure may not rigorously control the type-I error in this context.

The so-called core-periphery structure offers a natural framework with which to represent networks containing both informative and noninformative components. The core component includes nodes that induce an informative structure, while the periphery nodes do not have informative structures in their connections. Including the periphery in data can significantly undermine the effectiveness of statistical analysis. For illustration, we consider a synthetic example. We generate the core network from the inner-product latent space model [Hoff et al., 2002]. Any two nodes i and j are independently connected with probability satisfying $\text{logit}(P_{ij}) = \alpha + \beta X_i^T X_j$, where $X_i, i = 1, \cdots, 500$ are the two-dimensional latent positions of the nodes, shown by the circles in Figure 1(a). The norms of the true latent vectors exhibit a bimodal pattern, corresponding to two levels of connection popularity, a useful concept in studying social influence [Paluck et al., 2016]. Figure 1(a) shows the latent vectors estimated by the method of Ma et al. [2020], based on the network with core nodes only. The estimated model nicely recovers the bimodal norms. The basic unimodal test [Ameijeiras-Alonso et al., 2019] of the latent vector norms also results in a clear rejection. We then consider the case when the core nodes are observed with periphery nodes (Figure 1(b)). Each periphery node connects to all other nodes independently with probability $1/\big(1+\exp(-\alpha)\big)$. This is an instantiation of the ER-type periphery model in Section 2. If we directly use the full network as the input data for model fitting, the estimation becomes much less accurate and the bimodal pattern is no longer observable (Figure 1(c)). In particular, Figure 1(d) shows that the unimodal test gives increasing p-values when there are more periphery nodes. Notice that this degradation cannot be explained by model misspecification. Because the periphery nodes can be modeled by taking their latent positions at the origin in the same latent space. Therefore, the two-dimensional inner-product model is still correct. The degradation of estimation is really because including the periphery nodes hurts the modeling efficiency on the core nodes. In Section 5, we will provide another detailed example demonstrating the importance of removing periphery nodes in community detection tasks. In general, removing periphery nodes is a crucial data preprocessing step before learning the core structure.

The core-periphery structure has been extensively studied in the network literature. For instance, Borgatti and Everett [2000] defined the core-periphery model as a special case of the stochastic block model [Holland et al., 1983]. This definition of core-periphery was subsequently adopted by Zhang et al. [2015], Priebe et al. [2019], and applied to the so-called "planted clique" problem [Alon et al., 1998, Dekel et al., 2014]. However, per this definition, the network core is a densely connected Erdös-Rényi network, which itself is not interesting for any downstream analysis. This definition is also largely contingent on a large density gap between the core and the periphery [Zhang et al., 2015, Kojaku and Masuda, 2018], a phenomenon which may not manifest in many applications. Another related problem is the submatrix localization problem [Butucea et al., 2015, Deshpande and Montanari, 2015, Hajek et al., 2017, Cai et al., 2017]. The objective is to find K densely connected subgraphs planted in a large graph; the K subgraphs are usually assumed to be Erdös-Rényi graphs, which is again too restrictive in practice. Naik et al. [2019] recently proposed another core-periphery model. The core structure is more general than the Erdös-Rényi graph but still follows a restrictive parametric form. Moreover, the model can only generate networks with node degrees at least as dense as

the square root of the network size, which is too dense to most most real-world networks. Algorithm-based methods have been studied in relation to core-periphery structures as well [Lee et al., 2014, Della Rossa et al., 2013, Barucca et al., 2016, Cucuringu et al., 2016, Rombach et al., 2017]. These approaches typically assign a "coreness" score to each node based on certain topological assumptions. For example, in Wang and Rohe [2016] and Li et al. [2020b,a] , the k-core pruning algorithm [Seidman, 1983] was used to remove low-degree nodes as periphery. The statistical properties of this class of methods are not yet well-understood.

This paper aims to bridge the gap between the theoretically predicted effectiveness of network modeling and the empirical expectation in data analysis by proposing a principled and computationally efficient preprocessing method to extract the informative core structure from a large network. We introduce two core-periphery models with informative and noninformative structures. The novelty of our models is two-fold. First, we do not assume a specific structure for the core component; our framework can therefore be used as to preprocess data for any downstream network analysis tasks. Second, the distinction between the core and periphery components is whether they exhibit informative connection patterns. As such, our core-periphery definition emphasizes what we care about most: the informative structure for downstream network modeling. From this perspective, our overall assumption can be labeled as an "informative-core+noninformative-periphery" model. Under the proposed models, we develop simple yet efficient algorithms to identify the core structure with theoretically provable guarantees. We show that our algorithms can exactly identify the core component in sparse networks – the so-called "strong consistency" guarantee. The strong consistency is crucial for our method to be a general preprocessing approach. Given the strong consistency, the theoretical analysis for any downstream modeling of the core component remains valid, conditioned on the success of our method.

The rest of the paper is organized as follows. We first propose our core-periphery models in Section 2.1 and then introduce the algorithms for core identification in Section 2.2. Section 3 focuses on the algorithms' theoretical properties with respect to the accuracy of core identification. Extensive evaluations are presented in Section 4, where we demonstrate the advantages of our method against several benchmark methods for core-periphery structures. In Section 5, we demonstrate our method by extracting informative core structure from a citation network to improve downstream hierarchical community detection. We conclude the paper with a discussion in Section 6. All the proofs of our theoretical results are included in the our supplementary materials.

2. Methodology

Notations. We use capital boldface letters such as $\boldsymbol{M}$ to denote matrices. Given a matrix $\boldsymbol{M}$, $\boldsymbol{M}_{i,*}$, $\boldsymbol{M}_{*,j}$, and $\boldsymbol{M}_{ij}$ denote the i-th row, j-th column, and (i,j)-th entry, respectively. Let $\|\boldsymbol{M}\|_F$, $\|\boldsymbol{M}\|_2$, $\|\boldsymbol{M}\|_{2,\infty}$ be the Frobenius norm, the spectral norm, the two-to-infinity norm (maximum Euclidean norm of rows) of $\boldsymbol{M}$, respectively. We use $\boldsymbol{I}_d$ to denote the $d \times d$ identity matrix, and $\mathbf{1}_d$ to denote the $d \times 1$ vector whose entries are all 1. Let $\mathrm{rank}(\boldsymbol{M})$ be the rank of $\boldsymbol{M}$, and $\boldsymbol{M}^t$ be the transpose of $\boldsymbol{M}$. Let $[l]$ be the index set $\{1, 2, ..., l\}$. Let $\mathbb{O}_{p_1,p_2}$ be the set of $p_1 \times p_2$ matrices with orthonormal columns,

and let $\mathbb{O}_p$ be the shorthand for $\mathbb{O}_{p,p}$. For any two positive sequences $\{a_n\}$ and $\{b_n\}$, we say $a_n \preceq b_n$ if there exists a positive constant C such that $a_n \leq Cb_n$ for sufficiently large n, in which case we may also write $b_n \succeq a_n$; $a_n \simeq b_n$ if $a_n \succeq b_n$ and $a_n \preceq b_n$; $a_n \succ b_n$ if for an arbitrarily large $C > 0$; and $a_n > Cb_n$ for sufficiently large n.

2.1. *Informative core-periphery models*

Assume the network size to be n. We will focus on undirected and unweighted networks without self-loops. Such a network can be represented by an $n \times n$ symmetric binary adjacency matrix $\boldsymbol{A}$, such that $\boldsymbol{A}_{ij}$ is 1 if and only if nodes i and j are connected. As will be seen later, it would be easy to adapt our method for undirected weighted networks. We will embed our discussion into the so-called "inhomogeneous Erdös-Renyi" framework. Specifically, we assume that there exists an underlying $n \times n$ probability matrix $\boldsymbol{P}$ such that $\boldsymbol{A}_{ij}$'s are generated *independently* from Bernoulli($\boldsymbol{P}_{ij}$), for $1 \leq i < j \leq n$. The lower triangular entries are filled according to the symmetric constraint. We denote by $\boldsymbol{E}$ the difference between $\boldsymbol{A}$ and $\boldsymbol{P}$, i.e. $\boldsymbol{A} = \boldsymbol{P} + \boldsymbol{E}$. The elements $\{\boldsymbol{P}_{ij}\}$ are called connection probabilities. The matrix $\boldsymbol{P}$ fully specifies the structural information of the network model.

In our context, the periphery component should not display interesting structures. Though an interesting structure may depend on specific applications, we believe that the widely regarded *uninteresting* pattern is relatively simple to define. The following core-periphery model is defined according to one such pattern for the periphery.

Model 1 (The ER-type core-periphery model). *Network nodes can be partitioned into a core set $\mathcal{C}$ and a periphery set $\mathcal{P}$, where*

$$\mathcal{P} = \{i \in [n] | \boldsymbol{P}_{ij} = \boldsymbol{P}_{ik}, \text{ for all } j, k \in [n], j \neq i, k \neq i\}.$$

and $\mathcal{C} = [n] \setminus \mathcal{P}$.

The ER-type model gets its name from the Erdös-Rényi (ER) model [Erdös, 1959]. In brief, thanks to symmetry of $\boldsymbol{P}$, Model 1 indicates that all edges involving periphery nodes are generated with an identical probability, resembling the ER model. That means, there exists a probability p such that

$$\boldsymbol{P}_{ij} = p, \quad \text{for all} \quad i \in \mathcal{P}, j \in [n]. \tag{1}$$

By contrast, the subnetwork induced by the core nodes can follow any connection pattern as long as the pattern differs from (1). This generality provides the necessary flexibility to use our model as a data preprocessing step for any downstream analysis. In the special case where the core subnetwork is also an ER model but with a different density, the model reduces to the planted clique model used in Borgatti and Everett [2000], Zhang et al. [2015], and Priebe et al. [2019]. Figure 2 illustrates one example of the P matrix following the ER-type model.

The ER-type periphery is arguably the most basic form of non-informative structure; it also indicates that all periphery nodes have the same expected degree $(n-1)p$. Alternatively, there are other situations where the nodes have heterogeneous degrees but their connection patterns are still uninteresting because the connection only depends on

two nodes separably. Such a pattern resembles the configuration model [Bollobás, 1980, Chung and Lu, 2002, Newman, 2018], in that the connection probability between two nodes solely depends on the product of their degrees. In core-periphery contexts, this model can be defined as follows.

MODEL 2 (THE CONFIGURATION-TYPE CORE-PERIPHERY MODEL). *Let d_i be the expected degree of node i. Network nodes can be partitioned into a core set $\mathcal{C}$ and a periphery set $\mathcal{P}$, where*

$$\mathcal{P} = \{i \in [n] | \boldsymbol{P}_{ij} = \frac{d_i d_j}{\sum_{k=1}^{n} d_k}, \text{ for all } j \in [n], j \neq i\}. \tag{2}$$

and $\mathcal{C} = [n] \setminus \mathcal{P}$.

The periphery connection pattern under the configuration-type model essentially assumes that

$$\boldsymbol{P}_{ij} \propto d_i d_j, \quad \text{for all} \quad i \in \mathcal{P}, j \in [n]. \tag{3}$$

Figure 2 illustrates the configuration-type model. Compared with the ER-type model, the periphery also exhibits a heterogeneous connection pattern. This model can adopt arbitrary degree distributions for the periphery nodes.

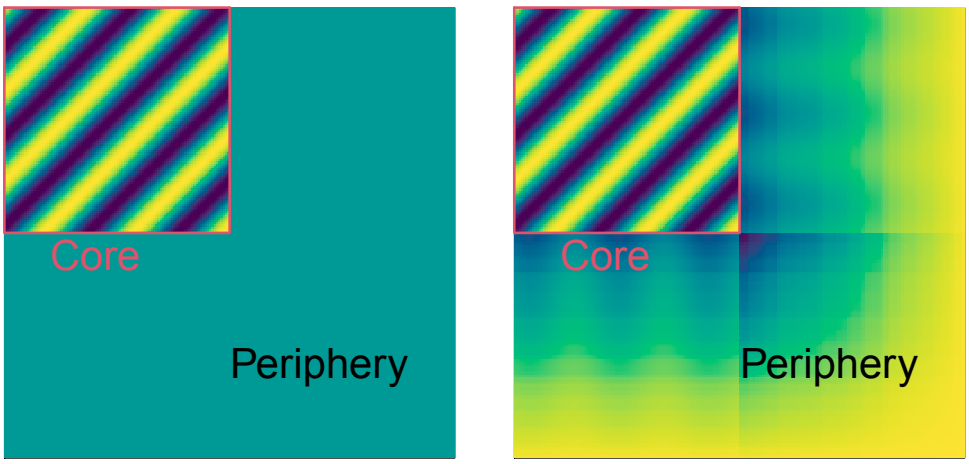
Before concluding this section, we want to emphasize that we do not assume that the core-induced subnetwork and the rest of the network to have different densities, unlike many other core-periphery models [Zhang et al., 2015, Kojaku and Masuda, 2018]. Our methods work well even if the core and periphery components have identical densities, as shown in our theory and empirical results.

2.2. Algorithms for core identification

We now proceed to introduce our algorithms to identify the core components under the two models. Likelihood-based procedures are not applicable in this context because the core subnetwork model is unspecified. Instead, we will leverage the special pattern of the core-periphery structure.

Consider the ER-type model first. For any periphery node i, $\boldsymbol{P}_{i,*}$ is a vector of the same value except for the diagonal entry and exhibits almost no variation; for any core node i, the entries in $\boldsymbol{P}_{i,*}$ exhibit greater variation by definition. Therefore, the core and periphery may be split according to the row-wise variation in connection probabilities. Define the centering matrix $\boldsymbol{H}$ to be $\boldsymbol{I}_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^t$. Then $||\boldsymbol{P}_{i,*}\boldsymbol{H}||_2^2$ is the natural metric of variation in $\boldsymbol{P}_{i,*}$. In particular, the norm $||\boldsymbol{P}_{i,*}\boldsymbol{H}||_2$ is almost zero for $i \in \mathcal{P}$, because $\boldsymbol{P}_{i,*}$ is a constant vector except on the ith coordinate. The periphery nodes can thus be identified by looking for such small $||\boldsymbol{P}_{i,*}\boldsymbol{H}||_2$ values.

In practice, we observe $\boldsymbol{A}$ rather than $\boldsymbol{P}$, and the above strategy will not work due to the large perturbation of $\boldsymbol{A}$ from $\boldsymbol{P}$. As such, we denoise $\boldsymbol{A}$ with an estimator $\hat{\boldsymbol{P}}$ and then shift the above strategy to $\hat{\boldsymbol{P}}$. Notice that $\text{rank}(\boldsymbol{P}) \leq \text{rank}(\boldsymbol{P}^{\mathcal{C}}) + 1$ where $\boldsymbol{P}^{\mathcal{C}}$ is the connection probability matrix for the induced network of the core nodes. Similar properties can be obtained by replacing the rank with many reasonable definitions of stable rank. Meanwhile, as Chatterjee [2015] suggested, almost all interesting network models give approximately low-rank structures. These motivate us to consider $\boldsymbol{P}$ as *approximately low-rank* (to be formally defined in our theory) and to use low-rank



(a) ER-type (b) Configuration-type

Fig. 2. Illustrations of the $\boldsymbol{P}$ matrices in our core-periphery models: (a) ER-type core-periphery model, where the expected degrees of the periphery nodes are the same. (b) Configuration-type core-periphery model, where the expected degrees of the periphery nodes are randomly sampled from a uniform distribution.

approximation as an estimator of $\hat{\boldsymbol{P}}$. The simplest estimator would be the universal singular value thresholding estimator proposed by Chatterjee [2015]. However, theoretically and empirically, using an adaptive method to cut off the singular values of $\boldsymbol{A}$ to a certain rank turns out to be more effective. Specifically, given a positive integer r, we use the rank-r truncated SVD of $\boldsymbol{A}$ as $\hat{\boldsymbol{P}}$. Our algorithm under the ER-type model is summarized in Algorithm 1. In the algorithm, we treat the approximating rank r as given. In practice, the r will be chosen based on the data. This task can be accomplished in two ways: either through visualization with a scree plot, as is often done for low-rank models [Kanyongo, 2005, Athreya et al., 2017]; or through a data-driven criterion such as Le and Levina [2022] and Li et al. [2020b], which admit various theoretical guarantees under an exact low-rank assumption. We will use the spectral Bethe-Hessian method of Le and Levina [2022] to select a proper r in our examples for its computational efficiency.

Under the configuration-type model, a similar strategy can be applied with an additional modification. The key attribute is a degree-correction step to neutralize the impacts of heterogeneous degrees. According to the periphery connection probabilities in (2), for any $i \in \mathcal{P}$, we have

$$\boldsymbol{P}_{ij}/d_j = \frac{d_i}{\sum_k d_k}, \quad \text{for any} \quad j \neq i.$$

Hence, normalizing the columns by the corresponding degrees would result in an asymmetric matrix where the row for each periphery node is a constant, except for the diagonal entry. Define $\boldsymbol{D} = \mathrm{diag}(d_1, \cdots, d_n)$. The column correction step can be written as $\boldsymbol{P}\boldsymbol{D}^{-1}$. After this degree-correction step, the same idea in Algorithm 1 can be applied

Algorithm 1 Spectral algorithm for core identification from the ER-type periphery

Input: Adjacency matrix $\boldsymbol{A}$, the core size $N_{\mathcal{C}}$, and approximating rank r.

(a) Find the low-rank approximation of $\boldsymbol{A}$ through rank r truncated SVD. Denote the resulting matrix as $\hat{\boldsymbol{P}}$.

(b) Compute the score $S_i = ||\hat{\boldsymbol{P}}_{i,*}\boldsymbol{H}||_2$, for $i \in [n]$.

(c) Sort the scores $S_1, S_2, ..., S_n$.

(d) For each $i \in [n]$, classify node i as a core node if S_i is among the top-$N_{\mathcal{C}}$ scores; otherwise classify node i as a periphery node.

Algorithm 2 Spectral algorithm for core identification from the configuration-type periphery

Input: Adjacency matrix $\boldsymbol{A}$, the core size $N_{\mathcal{C}}$ and approximating rank r.

(a) Find the low-rank approximation of $\boldsymbol{A}$ through rank r truncated SVD. Denote the resulting matrix as $\hat{\boldsymbol{P}}$.

(b) Compute $\hat{d}_i = \sum_{j=1}^{n} \boldsymbol{A}_{ij}$, and let $\hat{\boldsymbol{D}} = \text{diag}\{\hat{d}_1, \hat{d}_2, ..., \hat{d}_n\}$.

(c) Compute the scores $S'_i = ||\hat{\boldsymbol{P}}_{i,*}\hat{\boldsymbol{D}}^{-1}\boldsymbol{H}||_2$, for $i \in [n]$.

(d) Sort scores $S'_1, S'_2, ..., S'_n$.

(e) For each $i \in [n]$, classify node i as a core node if S'_i is among the top-$N_{\mathcal{C}}$ scores; otherwise classify node i as a periphery node.

here; we will use $||\boldsymbol{P}_{i,*}\boldsymbol{D}^{-1}\boldsymbol{H}||_2$ to separate the core nodes from periphery nodes. In practice, $\boldsymbol{P}$ is again substituted by its estimate $\hat{\boldsymbol{P}}$, and $\boldsymbol{D}$ is replaced with its sample version $\hat{\boldsymbol{D}}$, the diagonal matrix of observed node degrees. The details are summarized in Algorithm 2.

The major computational burden of Algorithms 1 and 2 entails on the rank-r SVD of $\boldsymbol{A}$. This step can be completed efficiently, especially for sparse networks and a small r [Baglama and Reichel, 2005]. Both algorithms are thus highly scalable to large networks. In Section 3, we will show that these algorithms are provably accurate in core identification.

3. Theoretical properties

We will introduce our theoretical analysis of the core identification algorithms under the ER-type model first. Then the theoretical properties will be extended to the configuration-type model.

3.1. Theory under the ER-type model

Intuitively, the success of Algorithm 1 depends on the connection probability variation of core nodes. To quantify this the variation, we introduce the following quantities.

$$h(n) = \min_{i \in \mathcal{C}} \|\boldsymbol{P}_{i,*}\boldsymbol{H}\|_2, \quad \text{and} \quad p^* = \max_{1 \le i,j \le n} \boldsymbol{P}_{ij}.$$

The algorithm also needs a low-rank approximation of $\boldsymbol{P}$. As mentioned in the previous section, we will use the rank-r truncated eigen-decomposition of the observed adjacency matrix $\boldsymbol{A}$ as $\hat{\boldsymbol{P}}$. Suppose $\boldsymbol{P}$ and $\boldsymbol{A}$ admit the following decompositions:

$$\boldsymbol{P} = \begin{bmatrix} \boldsymbol{U} & \boldsymbol{U}_\perp \end{bmatrix} \begin{bmatrix} \boldsymbol{\Lambda} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Lambda}_\perp \end{bmatrix} \begin{bmatrix} \boldsymbol{U}^t \\ \boldsymbol{U}_\perp^t \end{bmatrix} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^t + \boldsymbol{U}_\perp\boldsymbol{\Lambda}_\perp\boldsymbol{U}_\perp^t, \tag{4}$$

$$\boldsymbol{A} = \begin{bmatrix} \hat{\boldsymbol{U}} & \hat{\boldsymbol{U}}_\perp \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{\Lambda}} & \boldsymbol{0} \\ \boldsymbol{0} & \hat{\boldsymbol{\Lambda}}_\perp \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{U}}^t \\ \hat{\boldsymbol{U}}_\perp^t \end{bmatrix} = \hat{\boldsymbol{U}}\hat{\boldsymbol{\Lambda}}\hat{\boldsymbol{U}}^t + \hat{\boldsymbol{U}}_\perp\hat{\boldsymbol{\Lambda}}_\perp\hat{\boldsymbol{U}}_\perp^t, \tag{5}$$

where $\boldsymbol{\Lambda} = \mathrm{diag}\{\lambda_1, \lambda_2, ..., \lambda_r\}$ and $\boldsymbol{\Lambda}_\perp = \mathrm{diag}\{\lambda_{r+1}, \lambda_{r+2}, ..., \lambda_n\}$ consist of the eigenvalues of $\boldsymbol{P}$ sorted in a decreasing order of their absolute values. $\boldsymbol{U} \in \mathbb{O}_{n,r}$ and $\boldsymbol{U}_\perp \in \mathbb{O}_{n,n-r}$ consist of the corresponding eigenvectors as columns, respectively. The matrices $\hat{\boldsymbol{\Lambda}}$, $\hat{\boldsymbol{\Lambda}}_\perp$, $\hat{\boldsymbol{U}}_\perp$ and $\hat{\boldsymbol{U}}_\perp$ are similarly defined for $\boldsymbol{A}$. Recall that $\hat{\boldsymbol{P}} = \hat{\boldsymbol{U}}\hat{\boldsymbol{\Lambda}}\hat{\boldsymbol{U}}^t$. For such a low-rank approximation to work well, we impose the following assumptions:

Assumption 1 (Approximate low-rankness). *$|\lambda_r| \succeq \frac{np^*}{\sqrt{r}}$, and $|\lambda_{r+1}| \prec |\lambda_r|$.*

Assumption 2 (Incoherence). *$\|\boldsymbol{U}\|_{2,\infty} \le \mu_0\sqrt{\frac{r}{n}}$, for a scalar μ_0 that may depend on n.*

Assumption 1 is to ensure the low-rank approximation to be reasonable. Assumption 2 ensures that the connection probabilities are not too spiky. μ_0 can be as large as $\sqrt{n}$ in principle. However, as studied by Candès and Recht [2009] and Cape et al. [2019], it is usually of a much lower order (e.g., $O(\log n)$ or $O(1)$) for many reasonable models. Remarks 2 and 4 include such examples. The incoherence assumption is widely used in matrix estimation literature [Candès and Recht, 2009, Chen, 2015, Fan et al., 2018, Cape et al., 2019, Abbe et al., 2020]. It is generally considered necessary for highly accurate entrywise or row/column-wise recovery of random matrices [Chen, 2015].

Theorem 1. *Assume the network $\boldsymbol{A}$ is generated from the ER-type model, defined by Model 1, under Assumption 1 and Assumption 2. Assume also that Algorithm 1 is used to identify the core nodes with the correct N_C. Furthermore, suppose $p^* \succeq \max\left\{\frac{\mu_0^2 r \log n}{n}, \frac{\mu_0^2 r^2}{n}\right\}$, and $|\lambda_1/\lambda_r|$ is bounded. If*

$$h(n) \succeq \mu_0\sqrt{r(\log n + r)p^*} + \mu_0^2 r\sqrt{p^*} + |\lambda_{r+1}|, \tag{6}$$

then, for sufficiently large n, Algorithm 1 exactly identifies the core and periphery nodes with probability at least $1 - (B(r) + 2)n^{-\gamma}$ for some positive constant γ, where $B(r) = 10\min\{r, 1 + \log_2(|\lambda_1/\lambda_r|)\}$.

We present Theorem 1 under the bounded eigen-ratio $|\lambda_1/\lambda_r|$ condition for conciseness. This assumption can be dropped, resulting in an additional factor of approximately $|\lambda_1/\lambda_r|$ in several terms on the righthand side of (6). This more general version of the theorem is included in Appendix A.

REMARK 1. *Unlike most other approaches for core identification, we do not assume the core subnetwork to be denser than the periphery; nor do we assume that the core size is of the same order as the periphery size. To see these explicitly, define the core-density and core-variance for each core node $i \in \mathcal{C}$ as*

$$\bar{p}_{iC} = \frac{1}{N_{\mathcal{C}}} \sum_{j \in \mathcal{C}} \boldsymbol{P}_{ij}, \quad \text{and} \quad v_{iC}^2 = \frac{1}{N_{\mathcal{C}}} \sum_{j \in \mathcal{C}} (\boldsymbol{P}_{ij} - \bar{p}_{iC})^2.$$

Let $\theta_C = N_{\mathcal{C}}/n$ be the core proportion of the network. It is not difficult to see that

$$h(n) = \sqrt{n\theta_C} \min_{i \in \mathcal{C}} \sqrt{v_{iC}^2 + (1-\theta_C)(\bar{p}_{iC} - p)^2}.$$

Even if the density-gap between the core and the periphery is zero such that $(\bar{p}_{iC} - p)^2 = 0$ for all $i \in \mathcal{C}$, and the core proportion is vanishing, giving $\theta_C \to 0$, (6) can still hold as long as the variance with the core structure is sufficiently large. The above generality renders significant advantages in practice, as demonstrated in Section 4 and 5.

REMARK 2. *To illustrate condition (6), take the stochastic block model (SBM) as an example for the core structure under the ER-type model. Specifically, we consider the following balanced assortative SBM:*

$$\boldsymbol{B} = (a-b)\boldsymbol{I} + b\mathbf{1}_K\mathbf{1}_K^t,$$

where $a > b > 0$, and K is the number of blocks. The edge probability matrix is $\boldsymbol{P}^{\mathcal{C}} = \rho \boldsymbol{Z}\boldsymbol{B}\boldsymbol{Z}^t$, where $\boldsymbol{Z}$ is the membership matrix, and $\boldsymbol{Z}_{ik} = 1$ if and only if node i belongs to block k. All blocks have the same size. For simplicity, we assume the periphery size to be the same as one community. We also let the connection probabilities involving periphery nodes be $\rho \cdot b$. In this case, it can be shown that the conditions in (6) becomes

$$\rho \succeq \frac{K^2 \log n}{n} + \frac{K^3}{n}. \tag{7}$$

Under the SBM core, the downstream task is naturally community detection. Suppose there are no periphery nodes. To precisely recover all community labels, a feasible approach with the best theoretical guarantee is given by the semidefinite programming (SDP) approach of Fei and Chen [2018], requiring $\rho \succeq \frac{K \log n}{n} + \frac{K^2}{n}$. This is slightly better than (7) by a factor of K. The SDP approach is a model-based approach relying on exactly the current SBM form. In contrast, our approach does not rely on such a special assumption. A more generic and computationally efficient community detection method would be the spectral clustering [Rohe et al., 2011, Lei and Rinaldo, 2015]. Among all results about spectral clustering, the weakest requirement for strong consistency we are aware of is $\rho \succeq \frac{K^3 \log n}{n} + \frac{K^4}{n}$ from Lei [2019]. This requirement is stronger than (7). Therefore, our core identification step imposes no stronger requirement than spectral clustering for the downstream community detection task.

In practice, the number of core nodes, $N_{\mathcal{C}}$, is often unknown. However, under a stronger condition than Corollary 1, the correct $N_{\mathcal{C}}$ can be recovered by cutting off the scores in Algorithm 1 by a threshold. In particular, define $\hat{p} = \frac{2}{n^2-n}\sum_{i<j} \boldsymbol{A}_{ij}$. We can replace the $N_{\mathcal{C}}$ in Step 4 of Algorithm 1 with

$$\hat{N}_{\mathcal{C}} = |\{i : S_i > \sqrt{\hat{p}(\log n)^{1+\epsilon}}\}| \tag{8}$$

for some small constant ϵ. The same type of performance as (6) is still achievable according to the following result.

COROLLARY 1. *Under the conditions of Theorem 1, suppose μ_0 and r are bounded with $|\lambda_{r+1}| \leq \sqrt{p^* \log n}$. Furthermore, assume*

$$\min_{1\leq i,j\leq n} \boldsymbol{P}_{ij} \simeq \max_{1\leq i,j\leq n} \boldsymbol{P}_{ij} = p^*,$$

and $h(n) \succ \sqrt{p^*(\log n)^{1+\epsilon}}$ for the constant ϵ in (8). If the $\hat{N}_{\mathcal{C}}$ defined by (8) is used in Algorithm 1, with sufficiently large n, the core and periphery can be exactly identified with probability at least $1 - (B(r) + 4)n^{-\gamma}$ for some positive constant γ.

In all of our experiments, we use $\epsilon = 0.05$. In the supplement material, we show that the empirical performance remains stable for reasonable range of ϵ.

We conclude this section with an upper bound for the number of misidentified core nodes under much weaker assumptions than Theorem 1.

THEOREM 2. *Assume the network $\boldsymbol{A}$ is generated from the ER-type model, defined by Model 1, and Algorithm 1 is used to identify the core nodes with the correct $N_{\mathcal{C}}$. Suppose $h(n) > p^*$. Denote the estimated core set by $\hat{\mathcal{C}}$ and let M be the cardinality of the symmetric difference between $\mathcal{C}$ and $\hat{\mathcal{C}}$. For a sufficiently large n, we have*

$$M \preceq \max\{r, \operatorname{rank}(\boldsymbol{P})\} \cdot \frac{\left(\max\{\sqrt{np^*}, \sqrt{\log n}\} + |\lambda_{r+1}|\right)^2}{(h(n) - p^*)^2} \tag{9}$$

with probability at least $1 - n^{-\gamma}$ for some positive constant γ.

For illustration, consider the example in Remark 2 again with $p^* \geq \frac{\log n}{n}$. In this case, (9) indicates that the misidentified number is $O(K^2 n/\log n)$. So if $K^2 n/\log n = o(N_{\mathcal{C}})$, the misidentified proportion is vanishing to zero. This property is also called the *weak consistency*. Similar to Corollary 1, the weak consistency can also be achieved by using the data-based (8), instead of $N_{\mathcal{C}}$. Due to the space limit, such a corollary is given in Appendix A.

It is worth noting that the difference between strong consistency and weak consistency is not only a technical matter in our scenario. The strong consistency ensures that any available theoretical analysis for the downstream inference can still hold as if the core is already known, by the union probability trick. From this perspective, the strong consistency renders a seamless connection of the data preprocessing and theories on the core network. The weak consistency, in contrast, does not come with this advantage. The downstream theory must consider the potential errors and dependence from the core identification step, which usually requires a new and more challenging analysis.

3.2. Theory under the configuration-type model

Next, we consider the configuration-type model defined by Model 2. Recall that for a periphery node i, $\boldsymbol{P}_{i,*}\boldsymbol{D}^{-1}$ is a constant vector except for the diagonal entry. Therefore, the proof can be done by applying a similar strategy to the last section on the column degree-corrected version of $\boldsymbol{P}$. Define

$$h'(n) = \min_{i\in\mathcal{C}} ||\boldsymbol{P}_{i,*}\boldsymbol{D}^{-1}\boldsymbol{H}||_2.$$

Under the configuration-type model, $h'(n)$ is the counterpart of the quantity $h(n)$ for the ER-type model.

THEOREM 3. *Assume the network $\boldsymbol{A}$ is generated from the configuration-type model, defined in Model 2, under Assumption 1 and Assumption 2. Assume also that Algorithm 2 is used to identify the core nodes with the correct $N_{\mathcal{C}}$ and r. Let $d_{\min} = \min_{1\le i\le n}\sum_{j=1}^{n}\boldsymbol{P}_{ij}$, and suppose $d_{\min} \succ \log n$, $p^* \succ \max\left\{\frac{\mu_0^2 r\log n}{n}, \frac{\mu_0^2 r^2}{n}\right\}$, and $|\lambda_1/\lambda_r|$ is bounded. If*

$$h'(n) \succ \frac{1}{d_{\min}}\left(\mu_0\sqrt{r(\log n + r)p^*} + \mu_0^2 r\sqrt{p^*} + |\lambda_{r+1}|\right) + \left\|\boldsymbol{P}\boldsymbol{D}^{-1}\right\|_{2,\infty}\sqrt{\frac{\log n}{d_{\min}}}, \quad (10)$$

then, for sufficiently large n, Algorithm 2 exactly identifies the core and periphery nodes with probability at least $1-(B(r)+4)n^{-\gamma}$, where $B(r) = 10\min\{r, 1+\log_2(|\lambda_1/\lambda_r|)\}$.

A more general version of the theorem is provided in the Appendix A.

REMARK 3. *An analogy of Remark 1 is available. Specifically, define the column degree-corrected probability as $\tilde{P}_{ij} = P_{ij}/d_j$. Define the core-density and core-variance of the degree-corrected probability for each core node $i \in \mathcal{C}$ as*

$$\breve{p}_{iC} = \frac{1}{N_{\mathcal{C}}}\sum_{j\in\mathcal{C}}\tilde{\boldsymbol{P}}_{ij}, \quad \text{and} \quad \breve{v}_{iC}^2 = \frac{1}{N_{\mathcal{C}}}\sum_{j\in\mathcal{C}}(\tilde{\boldsymbol{P}}_{ij} - \breve{p}_{iC})^2.$$

Let $\theta_C = N_{\mathcal{C}}/n$ is the core proportion of the network. It is not difficult to see that

$$h'(n) = \sqrt{n\theta_C}\min_{i\in\mathcal{C}}\sqrt{\breve{v}_{iC}^2 + (1-\theta_C)(\breve{p}_{iC} - \frac{d_i}{\sum_{k=1}^{n} d_k})^2}. \quad (11)$$

It can be seen that even if the degree-corrected density remains the same in the core and periphery, (10) can still be satisfied, as long as the core itself has sufficiently large variance.

REMARK 4. *To illustrate the condition (10), we consider the example when the degree-corrected stochastic block model (DC-SBM) [Karrer and Newman, 2011] is the true core model. Specifically, assume that the whole network follows the DC-SBM. The first $K-1$ clusters are the core, and the last cluster is the periphery. Suppose all clusters have equal size. Let $z_i \in \{1,\cdots,K\}$ be the cluster label of node i. The model can be parametrized by a sequence of node popularity parameters $\psi_i, 1\le i\le n$ and a $K\times K$ matrix $\rho\boldsymbol{B}$ where*

$\boldsymbol{B}$ *is a fixed symmetric matrix with the last row and column containing only 1's and* ρ *depends on* n*. The connection probability of this DC-SBM is given by* $P_{ij} = \psi_i\psi_j\rho\boldsymbol{B}_{z_iz_j}$*. Assume that each row sum of* $\boldsymbol{B}$ *is* K*. It can be verified that this model satisfies Model 2. If* $r = K$ *and* ψ_i*'s are of the same order, then (10) indicates that a sufficient condition for strong consistency is* $d_{\min} \succ K^2 \log n + K^3$*.*

When $N_{\mathcal{C}}$ is unknown, a cutoff threshold of scores can be used to determine the core-periphery separation under stronger conditions. Recall that $\hat{p} = \frac{2}{n^2-n}\sum_{i<j}\boldsymbol{A}_{ij}$. We can replace the $N_{\mathcal{C}}$ in Step 5 of Algorithm 2 with

$$\hat{N}'_{\mathcal{C}} = |\{i : S'_i > \frac{\sqrt{(\log n)^{1+\epsilon}}}{n\sqrt{\hat{p}}}\}| \tag{12}$$

for some small constant $\epsilon > 0$.

COROLLARY 2. *Under the conditions of Corollary 3, suppose* μ_0 *and* r *are bounded with* $|\lambda_{r+1}| \leq \sqrt{p^* \log n}$*. Furthermore, assume*

$$\min_{1\leq i,j\leq n} \boldsymbol{P}_{ij} \simeq \max_{1\leq i,j\leq n} \boldsymbol{P}_{ij} = p^*,$$

and

$$h'(n) \succ \frac{\sqrt{(\log n)^{1+\epsilon}}}{n\sqrt{p^*}}$$

for the constant ϵ *in (12). If the* $\hat{N}'_{\mathcal{C}}$ *defined by (12) is used in Algorithm 2, with a sufficiently large* n*, the core and periphery nodes can be exactly identified with probability at least* $1 - (B(r) + 6)n^{-\gamma}$ *for some positive constant* γ*.*

Finally, an upper bound for misidentified number is available under weaker conditions.

THEOREM 4. *Assume the network* $\boldsymbol{A}$ *is generated from the configuration-type model defined by Model 2, and Algorithm 2 is used to identify the core nodes with the correct* $N_{\mathcal{C}}$*. Suppose* $d_{\min} \succ \log n$*, and* $h'(n) > \frac{d_{\max}}{(n-1)d_{\min}}$*. Denote the estimated core set by* $\hat{\mathcal{C}}$ *and let* M' *be the cardinality of the symmetric difference between* $\mathcal{C}$ *and* $\hat{\mathcal{C}}$*. Then,*

$$M' \preceq \max\{r, \operatorname{rank}(\boldsymbol{P})\} \cdot \frac{np^* + \lambda_{r+1}^2 + \left\|\boldsymbol{P}\boldsymbol{D}^{-1}\right\|_2^2 \cdot d_{\min} \cdot \log n}{d_{\min}^2\left[h'(n) - \frac{d_{\max}}{(n-1)d_{\min}}\right]^2}$$

with probability at least $1 - 3n^{-\gamma}$ *for some positive constant* γ*.*

Similar to the ER-type model, the error bound based on the threshold (12) is available as a corollary in Appendix A.

4. Simulation examples

In this section, we evaluate the performance of our proposed algorithms on synthetic networks. We demonstrate the methods' effectiveness and benefits under a few different core models as well as density gaps between the core and the periphery.

Table 1. Graphons for simulating network cores.

Graphon function $g(\mu, \nu)$	Rank
$k/7$, if $\mu, \nu \in ((k-1)/6, k/6)$; $0.3/7$ otherwise.	6
$\sin[5\pi(\mu+\nu-1)+1]/2 + 0.5$	3
$1/\{1+\exp[15(0.8\lvert\mu-\nu\rvert)^{4/5} - 0.1]\}$	Full

In generating our networks, we always set the first $N_{\mathcal{C}}$ nodes to core. To demonstrate our methods' flexibility with respect to the core structure, we use graphon models [Aldous, 1981] as the models for the core component. Specifically, the core submatrix $\boldsymbol{P}^{\mathcal{C}}$ is generated as follows. Given a graphon function $g : [0,1] \times [0,1] \to [0,1]$, we first generate $N_{\mathcal{C}}$ i.i.d. random variables $\xi_i \sim \text{Uniform}[0,1], i = 1, \cdots, N_{\mathcal{C}}$, after which $\boldsymbol{P}^{\mathcal{C}}$ is set as

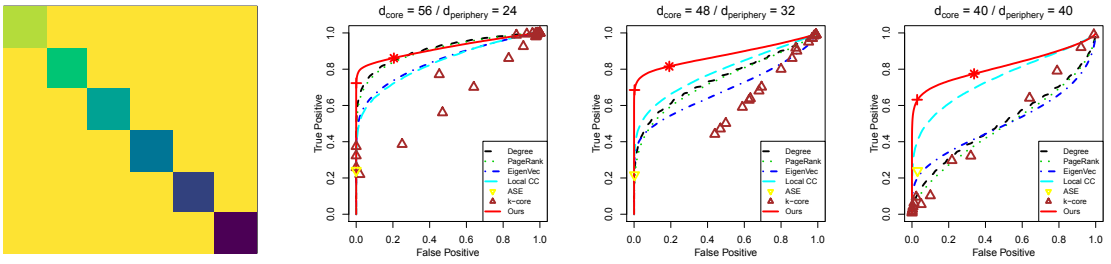
$$P^{\mathcal{C}}_{ij} = g(\xi_i, \xi_j), 1 \le i, j \le N_{\mathcal{C}} \tag{13}$$

We use three graphon functions defined in Zhang et al. [2017] as our simulation examples. The first one gives the simplest SBM for $\boldsymbol{P}^{\mathcal{C}}$ with a blockwise constant structure. The second one still has a low-rank $\boldsymbol{P}^{\mathcal{C}}$ but does not have a nice block structure. The third model is even more complicated and generates a full-rank $\boldsymbol{P}^{\mathcal{C}}$ – serving as a setting to verify the validity of our low-rank approximation strategy when the model is full-rank. The three models are summarized in Table 1; heatmaps of the $\boldsymbol{P}^{\mathcal{C}}$ in the three models appear in Figure 3 and Figure 4. Given $\boldsymbol{P}^{\mathcal{C}}$, we fill in the other positions of $\boldsymbol{P}$ based on periphery probabilities. For the ER-type model, we simply fill in a constant value. For the configuration-type model, the construction involves multiple steps. Let $\theta^{\mathcal{C}}_i = \sum_{j=1}^{N_{\mathcal{C}}} \boldsymbol{P}^{\mathcal{C}}_{ij}$, and sample $\theta^{\mathcal{P}}_i, i = 1, 2, ..., N_{\mathcal{P}}$ from a uniform distribution between $0.5 \min_{i \in \mathcal{C}} \theta_i$ and $1.5 \max_{i \in \mathcal{C}} \theta_i$. Then, let $\boldsymbol{\theta} = \{\theta^{\mathcal{C}}_1, \theta^{\mathcal{C}}_2, ..., \theta^{\mathcal{C}}_{N_{\mathcal{C}}}, \theta^{\mathcal{P}}_1, \theta^{\mathcal{P}}_2, ..., \theta^{\mathcal{P}}_{N_{\mathcal{P}}}\}$. The edge probability involving periphery node is set as $\boldsymbol{P}_{ij} = \frac{\boldsymbol{\theta}_i \boldsymbol{\theta}_j}{\sum_{k=1}^{N_{\mathcal{C}}} \theta^{\mathcal{C}}_k}$. It is not difficult to see that from this procedure, $d_i = \sum_{j=1}^{n} \boldsymbol{P}_{ij} = \frac{\boldsymbol{\theta}_i \sum_{j=1}^{n} \boldsymbol{\theta}_j}{\sum_{k=1}^{N_{\mathcal{C}}} \theta^{\mathcal{C}}_k}$, and $\boldsymbol{P}_{ij} = \frac{d_i d_j}{\sum_{k=1}^{n} d_k}$ for $i \in \mathcal{P}$, matching Model 2.

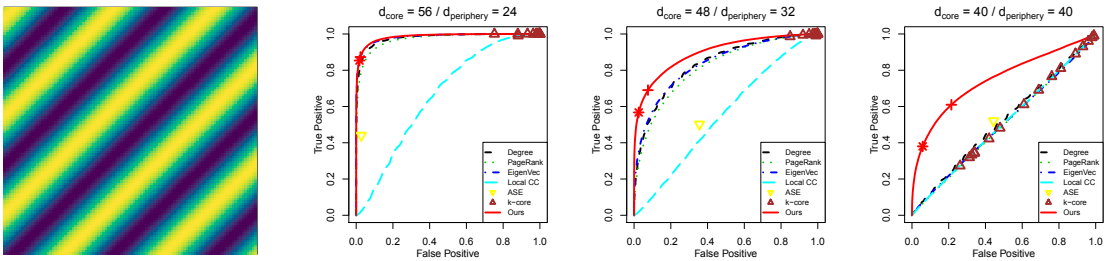
We then rescale the generated probability matrix, so the average edge density is around 0.02. We will demonstrate the effects of a varying density ratio between the two components. We focus on the settings where the core has an equal or higher density than the periphery. Our methods also perform well even if the core is sparser than the periphery; however, because this setting is less realistic, so it is not included. The core size and periphery size are both 1000 in this section. In Appendix E, we also include results for settings with imbalanced sizes. We use the Beth-Hessian method of Le and Levina [2022] to select r from $1, \cdots, n^{1/3}$, motivated by the discussion in Zhang et al. [2017].

Several benchmark core-periphery identification methods are included in our experiments. Two centrality-based methods are degree thresholding (Degree) and PageRank [Page et al., 1999] thresholding (PageRank), which have been shown to be competitive in identifying the core component in [Barucca et al., 2016, Rombach et al., 2017]. Theoretically, Zhang et al. [2015] indicated that under the SBM core-periphery model, the degree thresholding is optimal under favorable settings. Another common method is thresholding based on local clustering coefficients [Watts and Strogatz, 1998] (Local CC). The k-core pruning (k-core) algorithm [Seidman, 1983] is also included in our evalua-

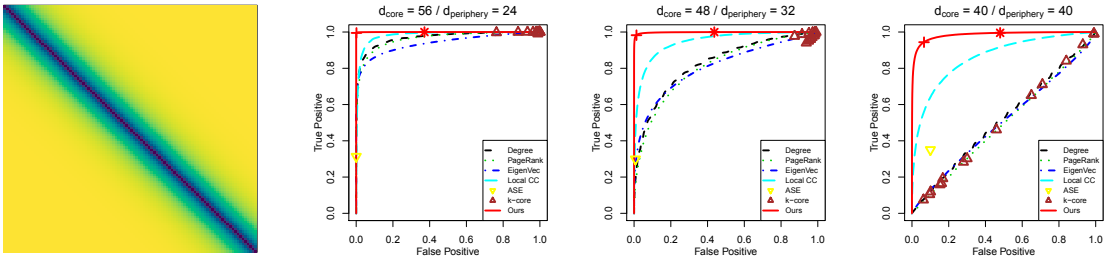
tion, representing a more adaptive version of the degree thresholding; the algorithm was shown to effectively extract meaningful subnetworks in Wang and Rohe [2016], Li et al. [2020b,a]. The last method is from Priebe et al. [2019], where the adjacency spectral embedding (ASE) Sussman et al. [2012] is used to capture the core-periphery structure.



(a) Graphon 1



(b) Graphon 2



(c) Graphon 3

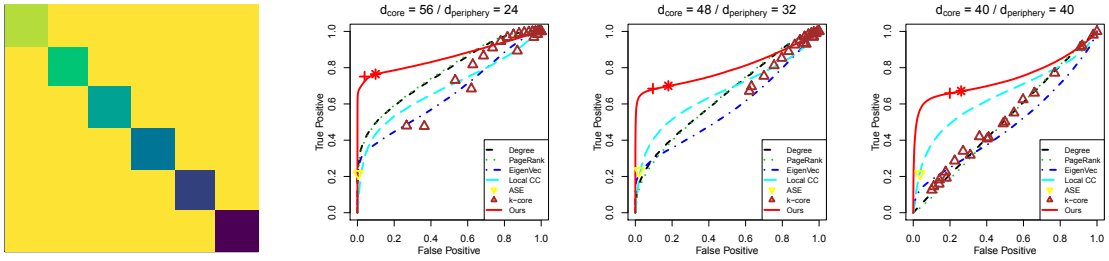
Fig. 3. Simulation results under the ER-type core-periphery model where $N_{\mathcal{C}} = N_{\mathcal{P}} = 1000$. The left figures are the core graphon functions, and the corresponding ROC curves are shown on the right, under different degree-gaps between core and periphery. The point "$*$" gives the model selection based on Corollary 1, and "+" indicates the model selection by k-means clustering with $k = 2$.

To fully characterize the core identification performance, we consider the tradeoff between the true positive rate (TPR) and the false positive rate (FPR), defined as

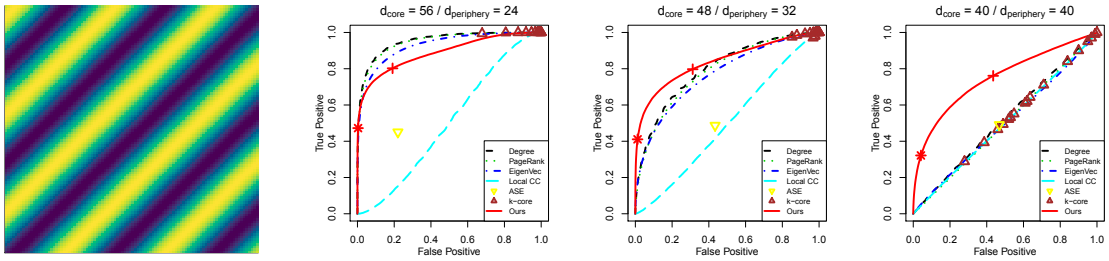
$$\text{TPR} = \frac{\#\{\text{Correctly identified nodes}\}}{\#\{\text{Identified nodes}\}} \quad \text{and} \quad \text{FPR} = \frac{\#\{\text{Incorrectly identified nodes}\}}{\#\{\text{Identified nodes}\}}.$$

These two metrics can be depicted by the receiver operating characteristic (ROC) curve.

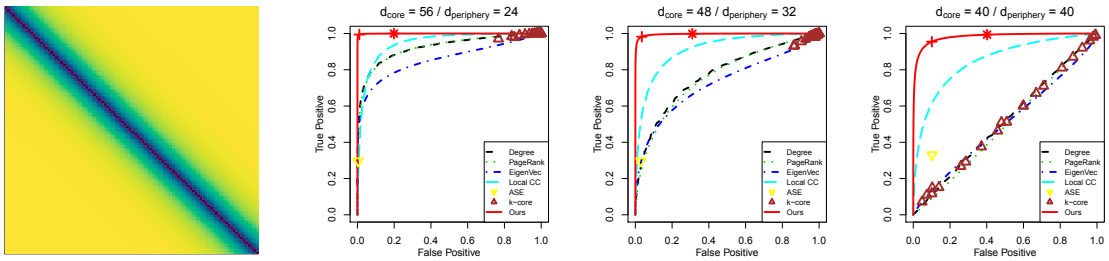
For each thresholding-based method, the full ROC curve is obtained by varying the threshold. The k-core pruning is applied with k increasing from 0 to a large integer, producing a sequence of points in the ROC space. On the contrary, ASE only gives a single point in the ROC space. For our method, we also include the single points based on our recommended threshold in Corollary 1 and 2, denoted by "$*$". Empirically, we also found that applying the k-means algorithm with $k = 2$ to the log-transformed scores works well in our simulation, and we mark the points obtained this way with "+" on the ROC curves. The code for our experiments is available at https://github.com/tianxili/Core-Periphery.



(a) Graphon 1



(b) Graphon 2



(c) Graphon 3

Fig. 4. Simulation results under the configuration-type core-periphery model where $N_{\mathcal{C}} = N_{\mathcal{P}} = 1000$. The left figures are the core graphon functions, and the corresponding ROC curves are shown on the right, under different degree-gaps between core and periphery. The point "$*$" gives the model selection based on Corollary 2, and "+" indicates the model selection by k-means clustering with $k = 2$.

Figure 3 shows the results under the ER-type model. The easiest setting is when the

core is much denser than the periphery. In this case, most of the approaches perform reasonably well. Though our method is more effective, its advantage is moderate. Yet as the density between the core and the periphery becomes more similar, the problem becomes more difficult, and most benchmark methods suffer from serious degradation and become close to a random guess. However, our method still maintains strong performance with its advantage over other methods becoming more pronounced. This outcome is expected because many of the benchmarks rely on the density gap between the two components whereas our method does not. Upon comparing the results across different core models, some of the benchmark methods perform well under one model but fail under another; our method remains the best one in all settings owing to its generality. Finally, the thresholds given by our theory ($*$) and k-means clustering ($+$) render sound model selections in the ROC space.

Figure 4 shows the results under the configuration-type model. The pattern is similar to that of Figure 3. Overall, the simulation examples indicate that our methods outperform the benchmark methods in the core identification accuracy across multiple core models and various core-periphery degree gaps. We also evaluate the robustness of our methods. These examples are included in supplementary material Section C; our methods continue to performance robustly even when the periphery connections deviate from our models due to random perturbations.

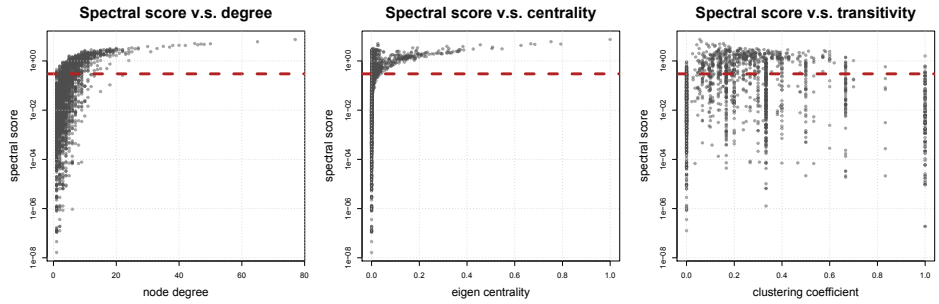
5. Core extraction in a statistics citation network

In this section, we demonstrate the impact of our core extraction method on the downstream community analysis of a paper citation network collected by Ji et al. [2016]. Each node in the network is a paper, and two nodes are connected if one paper cited the other (ignoring the citation direction). We focus on the largest connected component of the network. This network has 2248 nodes with an average node degree of 4.95. Wang and Rohe [2016] applied the 4-core pruning to the network, resulting in a core of 635 nodes for their downstream analysis. In this example, we compare several methods in Section 4 and evaluate their performance by comparing the validity of the hierarchical community detection results for the extracted cores.

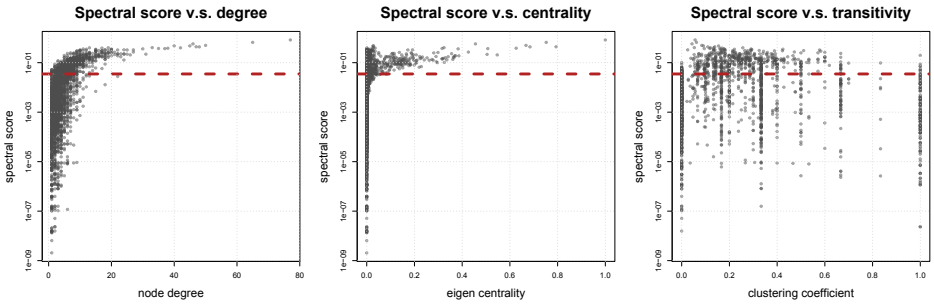
In Figure 5, we show the 2248 S_i scores of our two algorithms against the node degrees, eigenvector centrality scores, and the local clustering coefficients. Despite positive correlations between our scores and the other network statistics, the core nodes selected by our algorithms span the entire range of node degrees, centrality scores and clustering coefficients, highlighting the distinctions of our methods. In our evaluation, we also include the core selection based on degree centrality, eigenvector centrality, PageRank centrality, and local clustering coefficient as benchmarks. For fair comparisons, we restrict all methods to procedure core sizes of 1103 and 635, to match the respective 3-core and 4-core pruning algorithms of Wang and Rohe [2016].

Given the induced subnetwork of the selected core nodes, we apply the HCD algorithm of Li et al. [2020a]. This algorithm automatically determines the number of communities and identifies the community membership and the hierarchical relations between the communities as a binary tree. Each detected community is a leaf of the tree, while the internal nodes are interpreted as mega-communities – the union of multiple closely

connected leaf communities. Compared with regular community detection methods, the advantage of the HCD algorithm is that the hierarchy gives the relation between communities. The hierarchical relations can be represented by a $K \times K$ similarity matrix $\boldsymbol{S}$ with K representing the number of communities. $\boldsymbol{S}_{kk'}$ is calculated by the level of the first depth of the smallest mega-community containing both the kth and k'th communities. Therefore, $S_{kk'}$ is the similarity between community k and k' based on their position in the hierarchy of communities. If two communities are split at a lower level of the hierarchy, they are more similar to each other with denser between-community connections. Our target in this example is to find an interpretable hierarchical decomposition of the network. Intuitively, as already studied in previous works [Ji et al., 2016, Wang and Rohe, 2016, Li et al., 2020a], communities in this network may provide research topic interpretations.



(a) ER-type



(b) Configuration-type

Fig. 5. Spectral scores and local statistics based on two models on the citation network. The red line is the cutoff for 635 core nodes.

In particular, we will measure the meaningfulness of the hierarchy based on the abstracts of the papers collected by Wang and Rohe [2016]. We represent each abstract as a term-frequency vector and apply the standard text mining processing, such as stemming and removing stop words (including punctuations and numbers). Term frequency-inverse document frequency (TF-IDF) weighting [Rajaraman and Ullman, 2011] is then applied to each word. We remove words that appear in less than 1% of the papers; 966 words

Table 2. Correlation between S and T for different core identification methods.

Methods	Correlation	
	$N_{\mathcal{C}} = 635$	$N_{\mathcal{C}} = 1103$
Degree	0.099	0.089
k-core	0.167	0.108
PageRank	0.013	0.106
EigenVec	0.143	0.050
Local CC	0.058	0.045
Ours (ER)	**0.340**	**0.164**
Ours (Config)	**0.350**	**0.155**

remain after processing. The cosine similarity between each pair of papers is calculated, and a community-level text similarity matrix $\boldsymbol{T}$ is constructed where $\boldsymbol{T}_{kk'}$ is the average cosine similarity between papers from community k and community k'. Notice that, given the same community partition, $\boldsymbol{S}$ is the community similarity calculated based on the hierarchy of the communities, based on network structures, while $\boldsymbol{T}$ is community similarity based on the abstract data. How well the two similarities match each other indicates how well the hierarchical structure discovered by HCD matches the similarity derived from the abstracts. We calculate the Spearman correlation between $\boldsymbol{S}$ and $\boldsymbol{T}$ as the final metric. Given the community labels, if the hierarchy has no association with the research topics, we would expect a small correlation between $\boldsymbol{S}$ and $\boldsymbol{T}$. Therefore, the magnitude of the correlation reflects the consistency between the identified hierarchy and the topic similarity. The results for cores extracted via different methods are summarized in Table 2.

The cores extracted by our two models produce significantly more meaningful hierarchies than the other benchmarks. The difference between the ER-type model and the configuration-type model is negligible. Also, the results of both models lead to the same hierarchical structure with marginal differences. The communities are highly interpretable by human judgment as well. The figures of the final hierarchy of communities and the table of keywords of all communities are included in the supplementary material Section F. For completeness, though, we identify the cuts in our method to match the k-core algorithm for a fair comparison. We can still use our recommended cut-off to identify the core. In particular, using $\epsilon = 0.05$ would result in a core size of 846 under the ER-type model with $\boldsymbol{S} - \boldsymbol{T}$ correlation 0.361, and a core size of 712 under the configuration-type model with $\boldsymbol{S} - \boldsymbol{T}$ correlation 0.320. The code for this example is available at https://github.com/tianxili/Core-Periphery.

6. Discussion

We have proposed two core-periphery models for extracting informative structures from networks with efficient algorithms for core identification. We do not assume a specific form for the core component; as such, our methods can be used for preprocessing in general downstream network analysis tasks. The proposed algorithms have theoretical guarantees of correctly identifying the core nodes under mild conditions. The implementation of our methods are available from the R package `randnet` [Li et al., 2022].

The proposed method comes with a few limitations and these limitation also present fruitful directions for future research. One limitation of our methods lies in the low-rank denoising step. We believe that most interesting network models are approximately low-rank; however, low-rank models may not have covered all possible core structures. Recently, Seshadhri et al. [2020] observed that low-rank network models may not fit well for triangle structures between low-degree nodes. It would be interesting to investigate if this gap could be eliminated by certain type of "approximate low-rank". The low-rank approximation also requires good concentration to work well. Therefore, they may not handle extremely sparse networks in theory (e.g., network models with bounded node degrees). However, these limitations may be overcome by substituting the low-rank approximation step with methods that are more suitable for nontrivially full-rank models [Chan and Airoldi, 2014, Gao and Ma, 2021, Li and Le, 2021] or sparse networks [Le et al., 2017, Montanari and Sen, 2016, Fei and Chen, 2018]. A tradeoff also exists between the modeling performance and computational efficiency. Another limitation is that our assumed ER-type and configuration-type periphery is for generic data processing. In specific analysis, people may have other special type of "noninformative" structures to remove. Our method may not be directly applicable, and some case-based methods would be needed. Furthermore, we use the inhomogeneous Erdös-Renyi framework in our study which assumes that all edges are conditionally independent given $\boldsymbol{P}$. In the core-periphery context, it is also possible to assume that the core and periphery admit different types of dependence structure, which may generate a different type of modeling strategy. Extensions in this direction may call for nuanced definitions of uninteresting structures in novel scenarios and possibly new model-fitting tools.

Acknowledgements

This work was supported in part by the NSF grant DMS-2015298, the Quantitative Collaborative and 3Caverliers awards at the University of Virginia. All experiments are based on the support of Rivanna system of University of Virginia. The code and data of the paper are available at the https://github.com/tianxili/Core-Periphery. The authors want to thank the Editor, Associate Editor and reviewers for their constructive comments that significantly improved the paper. The authors also want to thank Lihua Lei for his helpful suggestions.

References

E. Abbe, J. Fan, K. Wang, Y. Zhong, et al. Entrywise eigenvector analysis of random matrices with low expected rank. *Annals of Statistics*, 48(3):1452–1474, 2020.

R. Albert and A.-L. Barabási. Statistical mechanics of complex networks. *Reviews of modern physics*, 74(1):47, 2002.

D. J. Aldous. Representations for partially exchangeable arrays of random variables. *Journal of Multivariate Analysis*, 11(4):581–598, 1981.

N. Alon, M. Krivelevich, and B. Sudakov. Finding a large hidden clique in a random graph. *Random Structures & Algorithms*, 13(3-4):457–466, 1998.

J. Ameijeiras-Alonso, R. M. Crujeiras, and A. Rodríguez-Casal. Mode testing, critical bandwidth and excess mass. *Test*, 28(3):900–919, 2019.

A. Athreya, D. E. Fishkind, M. Tang, C. E. Priebe, Y. Park, J. T. Vogelstein, K. Levin, V. Lyzinski, and Y. Qin. Statistical inference on random dot product graphs: a survey. *The Journal of Machine Learning Research*, 18(1):8393–8484, 2017.

J. Baglama and L. Reichel. Augmented implicitly restarted lanczos bidiagonalization methods. *SIAM Journal on Scientific Computing*, 27(1):19–42, 2005.

P. Barucca, D. Tantari, and F. Lillo. Centrality metrics and localization in core-periphery networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2016(2):023401, 2016.

P. J. Bickel and A. Chen. A nonparametric view of network models and newman–girvan and other modularities. *Proceedings of the National Academy of Sciences*, 106(50): 21068–21073, 2009.

B. Bollobás. A probabilistic proof of an asymptotic formula for the number of labelled regular graphs. *European Journal of Combinatorics*, 1(4):311–316, 1980.

S. P. Borgatti and M. G. Everett. Models of core/periphery structures. *Social networks*, 21(4):375–395, 2000.

C. Butucea, Y. I. Ingster, and I. A. Suslina. Sharp variable selection of a sparse submatrix in a high-dimensional noisy matrix. *ESAIM: Probability and Statistics*, 19:115–134, 2015.

T. T. Cai, T. Liang, A. Rakhlin, et al. Computational and statistical boundaries for submatrix localization in a large noisy matrix. *The Annals of Statistics*, 45(4):1403–1430, 2017.

E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717, 2009.

J. Cape, M. Tang, and C. E. Priebe. The two-to-infinity norm and singular subspace geometry with applications to high-dimensional statistics. *The Annals of Statistics*, 47(5):2405–2439, 2019.

S. Chan and E. Airoldi. A consistent histogram estimator for exchangeable graph models. In *International Conference on Machine Learning*, pages 208–216. PMLR, 2014.

S. Chatterjee. Matrix estimation by universal singular value thresholding. *The Annals of Statistics*, 43(1):177–214, 2015.

Y. Chen. Incoherence-optimal matrix completion. *IEEE Transactions on Information Theory*, 61(5):2909–2923, 2015.

F. Chung and L. Lu. The average distances in random graphs with given expected degrees. *Proceedings of the National Academy of Sciences*, 99(25):15879–15882, 2002.

M. Cucuringu, P. Rombach, S. H. Lee, and M. A. Porter. Detection of core–periphery structure in networks using spectral methods and geodesic paths. *European Journal of Applied Mathematics*, 27(6):846–887, 2016.

Y. Dekel, O. Gurel-Gurevich, and Y. Peres. Finding hidden cliques in linear time with high probability. *Combinatorics, Probability and Computing*, 23(1):29–49, 2014.

F. Della Rossa, F. Dercole, and C. Piccardi. Profiling core-periphery network structure by random walkers. *Scientific reports*, 3:1467, 2013.

Y. Deshpande and A. Montanari. Improved sum-of-squares lower bounds for hidden clique and hidden submatrix problems. In *Conference on Learning Theory*, pages 523–562, 2015.

P. Erdös. On random graphs. *Publicationes mathematicae*, 6:290–297, 1959.

J. Fan, W. Wang, and Y. Zhong. An ℓ_∞ eigenvector perturbation bound and its application to robust covariance estimation. *Journal of Machine Learning Research*, 18 (207):1–42, 2018.

Y. Fei and Y. Chen. Exponential error rates of sdp for block models: Beyond grothendieck's inequality. *IEEE Transactions on Information Theory*, 65(1):551–571, 2018.

C. Gao and J. Lafferty. Testing for global network structure using small subgraph statistics. *arXiv preprint arXiv:1710.00862*, 2017.

C. Gao and Z. Ma. Minimax rates in network analysis: Graphon estimation, community detection and hypothesis testing. *Statistical Science*, 36(1):16–33, 2021.

C. Gao, Z. Ma, A. Y. Zhang, and H. H. Zhou. Achieving optimal misclassification proportion in stochastic block models. *The Journal of Machine Learning Research*, 18 (1):1980–2024, 2017.

B. Hajek, Y. Wu, and J. Xu. Information limits for recovering a hidden community. *IEEE Transactions on Information Theory*, 63(8):4729–4745, 2017.

P. D. Hoff, A. E. Raftery, and M. S. Handcock. Latent space approaches to social network analysis. *Journal of the american Statistical association*, 97(460):1090–1098, 2002.

P. W. Holland, K. B. Laskey, and S. Leinhardt. Stochastic blockmodels: First steps. *Social networks*, 5(2):109–137, 1983.

D. N. Hoover. Relations on probability spaces and arrays of random variables. *Preprint, Institute for Advanced Study, Princeton, NJ*, 2, 1979.

P. Ji, J. Jin, et al. Coauthorship and citation networks for statisticians. *The Annals of Applied Statistics*, 10(4):1779–1812, 2016.

G. Y. Kanyongo. Determining the correct number of components to extract from a principal components analysis: A monte carlo study of the accuracy of the scree plot. *Journal of Modern Applied Statistical Methods*, 4(1):13, 2005.

B. Karrer and M. E. Newman. Stochastic blockmodels and community structure in networks. *Physical review E*, 83(1):016107, 2011.

S. Kojaku and N. Masuda. Core-periphery structure requires something else in the network. *New Journal of Physics*, 20(4):043012, 2018.

C. M. Le and E. Levina. Estimating the number of communities by spectral methods. *Electronic Journal of Statistics*, 16(1):3315 – 3342, 2022. doi: 10.1214/21-EJS1971. URL https://doi.org/10.1214/21-EJS1971.

C. M. Le, E. Levina, and R. Vershynin. Concentration and regularization of random graphs. *Random Structures & Algorithms*, 51(3):538–561, 2017.

S. H. Lee, M. Cucuringu, and M. A. Porter. Density-based and transport-based core-periphery structures in networks. *Physical Review E*, 89(3):032810, 2014.

J. Lei and A. Rinaldo. Consistency of spectral clustering in stochastic block models. *The Annals of Statistics*, 43(1):215–237, 2015.

L. Lei. Unified $\ell_{2\to\infty}$ eigenspace perturbation theory for symmetric random matrices. *arXiv preprint arXiv:1909.04798*, 2019.

T. Li and C. M. Le. Network estimation by mixing: Adaptivity and more. *arXiv preprint arXiv:2106.02803*, 2021.

T. Li, L. Lei, S. Bhattacharyya, K. Van den Berge, P. Sarkar, P. J. Bickel, and E. Levina. Hierarchical community detection by recursive partitioning. *Journal of the American Statistical Association*, pages 1–39, 2020a.

T. Li, E. Levina, and J. Zhu. Network cross-validation by edge sampling. *Biometrika*, 107(2):257–276, 2020b.

T. Li, E. Levina, J. Zhu, and C. M. Le. *randnet: Random Network Model Estimation, Selection and Parameter Tuning*, 2022. R package version 0.5.

Z. Ma, Z. Ma, and H. Yuan. Universal latent space model fitting for large networks with edge covariates. *Journal of Machine Learning Research*, 21(4):1–67, 2020.

A. Montanari and S. Sen. Semidefinite programs on sparse random graphs and their application to community detection. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pages 814–827, 2016.

S. S. Mukherjee, P. Sarkar, Y. R. Wang, and B. Yan. Mean field for the stochastic blockmodel: optimization landscape and convergence issues. In *Advances in Neural Information Processing Systems*, pages 10694–10704, 2018.

C. Naik, F. Caron, and J. Rousseau. Sparse networks with core-periphery structure. *arXiv preprint arXiv:1910.09679*, 2019.

M. Newman. The configuration model. In *Networks*. Oxford University Press, 2018.

M. E. Newman. Equivalence between modularity optimization and maximum likelihood methods for community detection. *Physical Review E*, 94(5):052315, 2016.

L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab, 1999.

E. L. Paluck, H. Shepherd, and P. M. Aronow. Changing climates of conflict: A social network experiment in 56 schools. *Proceedings of the National Academy of Sciences*, 113(3):566–571, 2016.

C. E. Priebe, Y. Park, J. T. Vogelstein, J. M. Conroy, V. Lyzinski, M. Tang, A. Athreya, J. Cape, and E. Bridgeford. On a two-truths phenomenon in spectral graph clustering. *Proceedings of the National Academy of Sciences*, 116(13):5995–6000, 2019.

A. Rajaraman and J. D. Ullman. *Mining of massive datasets*. Cambridge University Press, 2011.

K. Rohe, S. Chatterjee, and B. Yu. Spectral clustering and the high-dimensional stochastic blockmodel. *The Annals of Statistics*, 39(4):1878–1915, 2011.

P. Rombach, M. A. Porter, J. H. Fowler, and P. J. Mucha. Core-periphery structure in networks (revisited). *SIAM Review*, 59(3):619–646, 2017.

S. B. Seidman. Network structure and minimum degree. *Social networks*, 5(3):269–287, 1983.

C. Seshadhri, A. Sharma, A. Stolman, and A. Goel. The impossibility of low-rank representations for triangle-rich complex networks. *Proceedings of the National Academy of Sciences*, 117(11):5631–5637, 2020.

D. L. Sussman, M. Tang, D. E. Fishkind, and C. E. Priebe. A consistent adjacency spectral embedding for stochastic blockmodel graphs. *Journal of the American Statistical Association*, 107(499):1119–1128, 2012.

J. Ugander, L. Backstrom, and J. Kleinberg. Subgraph frequencies: Mapping the empirical and extremal geography of large graph collections. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1307–1318, 2013.

S. Wang and K. Rohe. Discussion of "coauthorship and citation networks for statisticians". *The Annals of Applied Statistics*, 10(4):1820–1826, 2016.

D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440–442, 1998.

X. Zhang, T. Martin, and M. E. Newman. Identification of core-periphery structure in networks. *Physical Review E*, 91(3):032803, 2015.

Y. Zhang, E. Levina, and J. Zhu. Estimating network edge probabilities by neighbourhood smoothing. *Biometrika*, 104(4):771–783, 2017.

Y. Zhao, E. Levina, J. Zhu, et al. Consistency of community detection in networks under degree-corrected stochastic block models. *The Annals of Statistics*, 40(4):2266–2292, 2012.

List of Figure Legends

- Figure 1(a): Estimated latent positions of 500 core nodes without any periphery nodes.
- Figure 1(b): Core-periphery network with 1500 periphery nodes.
- Figure 1(c): Estimated latent vectors of 500 core nodes with 1500 periphery nodes.
- Figure 1(d): P-values of the unimodal tests on latent vector norms.
- Figure 2(a): P matrix of an ER-type core-periphery model.
- Figure 2(b): P matrix of a configuration-type core-periphery model.
- Figure 3(a): Simulation results under the ER-type core-periphery model with Graphon 1 as the core.
- Figure 3(b): Simulation results under the ER-type core-periphery model with Graphon 2 as the core.
- Figure 3(c): Simulation results under the ER-type core-periphery model with Graphon 3 as the core.
- Figure 4(a): Simulation results under the configuration-type core-periphery model with Graphon 1 as the core.
- Figure 4(b): Simulation results under the configuration-type core-periphery model with Graphon 2 as the core.
- Figure 4(c): Simulation results under the configuration-type core-periphery model with Graphon 3 as the core.
- Figure 5(a): Spectral scores and local statistics based on the ER-type model on the citation network.
- Figure 5(b): Spectral scores and local statistics based on the ER-type model on the citation network.