Sankhyā A : The Indian Journal of Statistics
https://doi.org/10.1007/s13171-021-00245-4

Overlapping Community Detection in Networks via Sparse Spectral Decomposition

Jesús Arroyo
University of Maryland, College Park, USA
Johns Hopkins University, Baltimore, USA
Elizaveta Levina
University of Michigan, Ann Arbor, USA

Abstract

We consider the problem of estimating overlapping community memberships in a network, where each node can belong to multiple communities. More than a few communities per node are difficult to both estimate and interpret, so we focus on sparse node membership vectors. Our algorithm is based on sparse principal subspace estimation with iterative thresholding. The method is computationally efficient, with computational cost equivalent to estimating the leading eigenvectors of the adjacency matrix, and does not require an additional clustering step, unlike spectral clustering methods. We show that a fixed point of the algorithm corresponds to correct node memberships under a version of the stochastic block model. The methods are evaluated empirically on simulated and real-world networks, showing good statistical performance and computational efficiency.

AMS (2000) *subject classification.* Primary: 62H30, Secondary: 91C20, 68T10.
Keywords and phrases. Sparse principal component analysis, Stochastic blockmodel, Mixed memberships

1 Introduction

Networks have become a popular representation of complex data that appear in different fields such as biology, physics, and the social sciences. A network represents units of a system as nodes, and the interactions between them as edges. A network can encode relationships between people in a social environment (Wasserman and Faust, 1994), connectivity between areas of the brain (Bullmore and Sporns, 2009) or interactions between proteins (Schlitt and Brazma, 2007). The constant technological advancements have increased our ability to collect network data on a large scale, with potentially

millions of nodes in a network. Parsimonious models are needed in order to obtain meaningful interpretations of such data, as well as computationally efficient methods.

Communities are a structure of interest in the analysis of networks, observed in many real-world systems (Girvan and Newman, 2002). Usually, communities are defined as clusters of nodes that have stronger connections to each other than to the rest of the network. Finding these communities allows for a more parsimonious representation of the data which is often meaningful in the system of interest. For example, communities can represent functional areas of the brain (Schwarz et al., 2008; Power et al., 2011), political affinity in social networks (Adamic and Glance, 2005; Conover et al., 2011; Latouche et al., 2011), research areas in citation networks (Ji and Jin, 2016), and many others.

The stochastic block model (SBM) (Holland et al., 1983) is a simple statistical model for a network with communities, well understood by now; see Abbe (2017) for a review. Under the SBM, a pair of nodes is connected with a probability that only depends on the community memberships of these nodes. The SBM allows to represent any type of connectivity structure between the communities in the network, such as affinity, disassortativity, or core-periphery (see for example Cape et al. 2019). While the SBM itself is too simple to capture some aspects of real-world networks, many extensions have been proposed to incorporate more complex structures such as *hubs* (Ball et al., 2011), or nodes that belong to more than one community (Airoldi et al., 2009; Latouche et al., 2011; Zhang et al., 2020), which lead to models with overlapping communities.

Overlapping community models characterize each node by a membership vector, indicating its degree of belonging to different communities. While in principle all entries of a membership vector can be positive (Airoldi et al., 2009), a sparse membership vector is more likely to have a meaningful interpretation. At the same time, allowing for a varying degree of belonging to a community adds both flexibility and interpretability relative to binary membership overlapping community models such as Latouche et al. (2011). In this paper, we focus on estimating sparse overlapping community membership vectors with continuous entries, so that most nodes belong to only one or few communities, and the degree to which they belong to a community can vary. The sparsest case where each node belongs to exactly one community corresponds to the classic community detection setting, and its success in modeling and analyzing real-world networks in many different fields (Porter et al., 2009) supports the sparse nature of community memberships.

Existing statistical models for overlapping community detection include both binary membership models, where each node either belongs to a community or does not (e.g., Latouche et al. 2011), and continuous membership models which allow each node to have a different level of association with each community (Airoldi et al., 2009; Ball et al., 2011; Psorakis et al., 2011; Zhang et al., 2020). Binary memberships are a natural way to induce sparsity, but the binary models are less flexible, and fitting them be computationally intensive since they involve solving a discrete optimization problem. On the other hand, continuous memberships are not able to explicitly model sparsity, and the resulting estimates often assign most of the nodes to many or even all communities. To obtain sparse memberships, an ad hoc post-processing step can be applied (Gregory, 2010; Lancichinetti et al., 2011), but is likely to lead to a less accurate fit to the data. Another approach to induce sparse memberships is to incorporate sparsity-inducing priors into the model, for example via the discrete hierarchical Dirichlet process (Wang and Blei, 2009) or the Indian buffet process (Williamson et al., 2010) that have been introduced in the topic modeling literature.

The problem of estimating overlapping community memberships has been approached from different perspectives; see for example (Xie et al., 2013; da Fonseca Vieira et al., 2020), and references in Section 4.3. In particular, spectral methods for community detection are popular due to their computational scalability and theoretical guarantees (Newman, 2006; Rohe et al., 2011; Lyzinski et al., 2014; Le et al., 2017). Many statistics network models make a low-rank assumption on the matrix $\mathbf{P} = \mathbb{E}[\mathbf{A}]$ that characterizes the edge probabilities, and in most models with communities the principal subspace of $\mathbf{P}$ contains the information needed to identify the communities. Spectral methods for community detection exploit this fact by computing an eigendecomposition of the network adjacency matrix $\mathbf{A}$, defined by $\mathbf{A}_{ij} = 1$ if there is an edge from i to j and 0 otherwise, followed by a post-processing step applied to the leading eigenvectors to recover memberships. Several approaches of this type have been recently developed, with different ways of clustering the rows of the leading eigenvectors (Zhang et al., 2020; Rubin-Delanchy et al., 2017; Jin et al., 2017; Mao et al., 2017; Mao et al., 2018, 2020).

In contrast to other spectral methods, here we present a new approach for detecting overlapping communities based on estimating a *sparse* basis for the principal subspace of the network adjacency matrix in which the pattern of non-zero values contains the information about community memberships. Our approach can be seen as an analogue to finding sparse principal components of a matrix (Jolliffe et al., 2003; Zou et al., 2006; Ma, 2013), but with

the important difference that we consider a non-orthogonal sparse basis of the principal subspace to allow for overlaps in communities. Our method has thus the potential to estimate overlapping community memberships more accurately than traditional spectral methods, with the same low computational cost of computing the leading eigenvectors of a matrix. We will demonstrate this both on simulated networks with overlapping and non-overlapping communities, and on real-world networks.

2 A Sparse Non-Orthogonal Eigenbasis Decomposition

As mentioned in the introduction, we consider binary symmetric adjacency matrices $\mathbf{A} \in \{0,1\}^{n\times n}$, with no self-loops, i.e., $\mathbf{A}_{ii} = 0$. We model the network as an inhomogeneous Erdös-Rényi random graph (Bollobás et al., 2007), meaning that the upper triangular entries of $\mathbf{A}$ are independent Bernoulli random variables with potentially different edge probabilities $\mathbf{P}_{ij} = \mathbb{P}(\mathbf{A}_{ij} = 1)$ for $i, j \in [n]$, $i < j$, contained in a symmetric probability matrix $\mathbf{P} \in \mathbb{R}^{n\times n}$.

Our goal is to recover an overlapping community structure in $\mathbf{A}$ by estimating an appropriate sparse basis of the invariant subspace of $\mathbf{P}$. The rationale is that when $\mathbf{P}$ is even approximately low rank, most relevant information on communities is contained in the leading eigenvectors of $\mathbf{P}$, and can be retrieved by looking at a particular basis of its invariant subspace. We will assume that rank of $\mathbf{P}$ is $K < n$. The principal subspace of $\mathbf{P}$ can be described with a full rank matrix $\mathbf{V} \in \mathbb{R}^{n\times K}$, with columns of $\mathbf{V}$ forming a basis of this space. Most commonly, $\mathbf{V}$ is defined as the K leading eigenvectors of $\mathbf{P}$, but for the purposes of recovering community membership, we focus on finding a sparse *non-negative eigenbasis* of $\mathbf{P}$, that is, a matrix $\mathbf{V}$ for which $\mathbf{V}_{ik} \geq 0$ for all $i \in [n], k \in [K]$ and $\mathbf{P} = \mathbf{V}\mathbf{U}^\top$ for some full rank matrix $\mathbf{U} \in \mathbb{R}^{n\times K}$. Note that this is different from the popular non-negative matrix factorization problem (Lee and Seung, 1999), as we do not assume that $\mathbf{U}$ is a non-negative matrix nor do we try to estimate it.

If $\mathbf{P}$ has a sparse non-negative basis of its principal subspace $\mathbf{V} \in \mathbb{R}^{n\times K}$, this basis is not unique, as any column scaling or permutation of $\mathbf{V}$ will give another non-negative basis. Among these, we are interested in finding a sparse non-negative eigenbasis $\mathbf{V}$, since we will relate the non-zeros of $\mathbf{V}$ to community memberships. The following proposition provides a sufficient condition for identifiability of the non-zero pattern in $\mathbf{V}$ up to a permutation of its columns. The proof is included in the Appendix.

Proposition 1. *Let $\boldsymbol{P} \in \mathbb{R}^{n\times n}$ be a symmetric matrix of rank K. Suppose that there exist a matrix $\boldsymbol{V} \in \mathbb{R}^{n\times K}$ that satisfies the next conditions:*

- *Eigenbasis:* $\mathbf{V}$ *is a basis of the column space of* $\mathbf{P}$*, that is,* $\mathbf{P} = \mathbf{V}\mathbf{U}^\top$*, for some* $\mathbf{U} \in \mathbb{R}^{n\times K}$*.*
- *Non-negativity: The entries of* $\mathbf{V}$ *satisfy* $\mathbf{V}_{ik} \geq 0$ *for all* $i \in [n], k \in [K]$
- *Pure rows: For each* $k = 1, \ldots, K$ *there exists at least one row* i_k *of* $\mathbf{V}$ *such that* $\mathbf{V}_{i_k k} > 0$ *and* $\mathbf{V}_{i_k j} = 0$ *for* $j \neq k$*.*

If another matrix $\widetilde{\mathbf{V}} \in \mathbb{R}^{n\times K}$ *satisfies these conditions, then there exists a permutation matrix* $\mathbf{Q} \in \{0,1\}^{K\times K}$*,* $\mathbf{Q}^\top\mathbf{Q} = \mathbf{I}_K$*, such that*

$$supp(\mathbf{V}) = supp(\widetilde{\mathbf{V}}\mathbf{Q}),$$

where $supp(\mathbf{V}) = \{(i,j)|\mathbf{V}_{ij} \neq 0\}$ *is the set of non-zero entries of* $\mathbf{V}$*.*

We connect a non-negative non-orthogonal basis to community memberships through the *overlapping continuous community assignment model* (OCCAM) of Zhang et al. (2020), a general model for overlapping communities that encompasses, as special cases, multiple other popular overlapping models (Latouche et al., 2011; Ball et al., 2011; Jin et al., 2017; Mao et al., 2018). Under OCCAM, each node is associated with a vector $\mathbf{z}_i = [z_{i1}, \ldots, z_{iK}]^\top \in \mathbb{R}^K$, $i = 1, \ldots, n$, where K is the number of communities in the network. Given $\mathbf{Z} = [\boldsymbol{z}_1 \cdots \boldsymbol{z}_n]^\top \in \mathbb{R}^{n\times K}$ and parameters $\alpha > 0$, $\boldsymbol{\Theta} \in \mathbb{R}^{n\times n}$ and $\mathbf{B} \in \mathbb{R}^{K\times K}$ to be explained below, the probability matrix $\mathbf{P} = \mathbb{E}[\boldsymbol{A}]$ of OCCAM can be expressed as

$$\mathbf{P} = \alpha\boldsymbol{\Theta}\mathbf{Z}\mathbf{B}\mathbf{Z}^T\boldsymbol{\Theta}. \tag{2.1}$$

For identifiability, OCCAM assumes that α and all entries of $\boldsymbol{\Theta}$, $\mathbf{B}$ and $\mathbf{Z}$ are non-negative, $\boldsymbol{\Theta}$ is a diagonal matrix with $\mathrm{diag}(\boldsymbol{\Theta}) = \boldsymbol{\theta} \in \mathbb{R}^n$ and $\sum_{i=1}^n \boldsymbol{\theta}_i = n$, $\|\boldsymbol{z}_i\|_2 = \left(\sum_{j=1}^K z_{ij}^2\right)^{1/2} = 1$ for all $i \in [n]$, and $\boldsymbol{B}_{kk} = 1$ for all $k \in [K]$. In this representation, the row $\boldsymbol{z}_i$ of $\mathbf{Z}$ is viewed as the community membership vector of node i. A positive value of z_{ik} indicates that node i belongs to community k, and the magnitude of z_{ik} determines how strongly. The parameter $\boldsymbol{\theta}_i$ represents the degree correction for node i as in the degree-corrected SBM (Karrer and Newman, 2011) allowing for degree heterogeneity and in particular "hub" nodes, common in real-world networks. The scalar parameter $\alpha > 0$ controls the edge density of the entire graph.

One can obtain the classical SBM as a special case of OCCAM by further requiring each $\boldsymbol{z}_i$ to have only one non-zero value and setting $\boldsymbol{\theta}_i = 1$ for all $i \in [n]$. Keeping only one non-zero value in each row of $\mathbf{Z}$ but allowing

the entries of $\boldsymbol{\theta}$ to take positive values, one can recover the degree-corrected SBM (Karrer and Newman, 2011). More generally in OCCAM, nodes can belong to multiple communities at the same time. Each row of $\mathbf{Z}$ can have multiple or all the entries different from zero, indicating the communities to which the node belong.

Equation 2.1 implies that under OCCAM the probability matrix $\mathbf{P}$ has a non-negative eigenbasis given by $\mathbf{V} = \boldsymbol{\Theta}\mathbf{Z}$. The following proposition shows the converse result, namely, that any matrix $\mathbf{P}$ that admits a non-negative eigenbasis can be represented as in Eq. 2.1, which motivates the interpretation of the non-zero entries of a non-negative eigenbasis as indicators of community memberships.

Proposition 2. *Let $\mathbf{P} \in \mathbb{R}^{n\times n}$ be a symmetric real matrix with rank$(\mathbf{P}) = K$. Suppose that there exists a full-rank nonnegative matrix $\mathbf{V} \in \mathbb{R}^{n\times K}$ and a matrix $\mathbf{U}$ such that $\mathbf{P} = \mathbf{V}\mathbf{U}^\top$. Then, there exists a non-negative diagonal matrix $\boldsymbol{\Theta} \in \mathbb{R}^{n\times n}$, a non-negative matrix $\mathbf{Z} \in \mathbb{R}^{n\times K}$ with $\sum_{k=1}^{K} \mathbf{Z}_{ik}^2 = 1$ for each $i \in [n]$, and a symmetric matrix $\mathbf{B} \in \mathbb{R}^{K\times K}$ such that*

$$\mathbf{P} = \boldsymbol{\Theta}\mathbf{Z}\mathbf{B}\mathbf{Z}^\top\boldsymbol{\Theta}.$$

Moreover, if $\mathbf{V}$ satisfies the conditions of Proposition 1*, then supp$(\mathbf{V}) =$ supp$(\mathbf{ZQ})$ for some permutation $\mathbf{Q} \in \mathbb{R}^{K\times K}, \mathbf{Q}^\top\mathbf{Q} = \mathbf{I}$.*

In short, Proposition 2 states a non-negative basis of the probability matrix $\mathbf{P}$ can be mapped to overlapping communities as in Eq. 2.1. Moreover, under the conditions on this eigenbasis stated in Proposition 1, the community memberships can be uniquely identified. These conditions are weaker than the ones in Zhang et al. (2020) since we are only interested in community memberships and not in identifiability of the other parameters; note that we do not aim to fit the OCCAM model, which is computationally much more intensive than our approach here. Other conditions for identifiability of overlapping community memberships have been presented in the literature (Huang and Fu, 2019), but the pure row assumption in Proposition 1 is enough for our purpose of estimating sparse memberships.

3 Community Detection via Sparse Iterative Thresholding

Our goal is to compute an appropriate sparse basis of the principal subspace of $\mathbf{A}$ which contains information about the overlapping community memberships. Spectral clustering has been popular for community detection, typically clustering the rows of the leading eigenvectors of $\mathbf{A}$ or a function of them to assign nodes to communities. Spectral clustering with

overlapping communities typically gives a continuous membership matrix, which can then be thresholded to obtain sparse membership vectors; however, this two-stage approach is unlikely to be optimal in any sense, and some of the overlapping clustering procedures can be computationally expensive (Zhang et al., 2020; Jin et al., 2017). In contrast, our approach of directly computing a sparse basis of the principal subspace of $\mathbf{A}$ avoids the two-stage procedure and thus can lead to improvements in both accuracy and computational efficiency.

Sparse principal component analysis (SPCA) (Jolliffe et al., 2003; Zou et al., 2006) seeks to estimate the principal subspace of a matrix while incorporating sparsity constraints or regularization on the basis vectors. In high dimensions, enforcing sparsity can improve estimation when the sample size is relatively small, and/or simplify the interpretation of the solutions. Many SPCA algorithms have been proposed to estimate eigenvectors of a matrix under sparsity assumptions (see for example Amini and Wainwright 2008; Johnstone and Lu 2009; Vu and Lei 2013; Ma 2013).

Our goal is clearly related to SPCA since we are interested in estimating a sparse basis of the principal subspace of $\mathbf{P}$, but an important difference is that our vectors of interest are not necessarily orthogonal; in fact orthogonality is only achieved when estimated communities do not overlap, and is thus not compatible with meaningful overlapping community estimation. For non-overlapping community detection, however, there is a close connection between a convex relaxation of the maximum likelihood estimator of communities and a convex formulation of SPCA (Amini and Levina, 2018).

Orthogonal iteration is a classical method for estimating the eigenvectors of a matrix; see for example Golub and Van Loan (2012). Ma (2013) extended this method to estimate sparse eigenvectors by an iterative thresholding algorithm. Starting from an initial matrix $\mathbf{V}^{(0)} \in \mathbb{R}^{n\times K}$, the general form of their algorithm iterates the following steps until convergence:

1. Multiplication step:
$$\mathbf{T}^{(t+1)} = \mathbf{A}\mathbf{V}^{(t)}. \tag{3.1}$$

2. Regularization step:
$$\mathbf{U}^{(t+1)} = \mathcal{R}(\mathbf{T}^{(t)}, \mathbf{\Lambda}), \tag{3.2}$$
where $\mathcal{R} : \mathbb{R}^{n\times K} \to \mathbb{R}^{n\times K}$ is a regularization function and $\mathbf{\Lambda} \in \mathbb{R}^{n\times K}$ a matrix of regularization parameters.

3. Identifiability step:
$$\mathbf{V}^{(t+1)} = \mathbf{U}^{(t+1)}\mathbf{W}^{(t+1)}, \tag{3.3}$$

where $\mathbf{W}^{(t+1)}$ is a $K \times K$ matrix.

An example of a convergence criterion may be stopping when the distance between subspaces generated by $\mathbf{V}^{(t)}$ and $\mathbf{V}^{(t+1)}$ is small. For two full-rank matrices $\mathbf{U}, \tilde{\mathbf{U}} \in \mathbb{R}^{n \times K}$, the distance between the subspaces generated by the columns of $\mathbf{U}$ and $\tilde{\mathbf{U}}$ is defined through their orthogonal projection matrices $\mathbf{R} = \mathbf{U}(\mathbf{U}^\top \mathbf{U})^{-1}\mathbf{U}^\top$ and $\tilde{\mathbf{R}} = \tilde{\mathbf{U}}(\tilde{\mathbf{U}}^\top \tilde{\mathbf{U}})^{-1}\tilde{\mathbf{U}}^\top$ as

$$\text{dist}(\mathbf{U}, \tilde{\mathbf{U}}) = \|\mathbf{R} - \tilde{\mathbf{R}}\|,$$

where $\|\cdot\|$ is the matrix spectral norm (see Golub and Van Loan (2012), Section 2.5.3).

Let $\widehat{\mathbf{V}}$ be the value of $\mathbf{V}^{(t)}$ at convergence, and let $\widetilde{\mathbf{V}}$ be the $n \times K$ matrix of the K leading eigenvectors of $\mathbf{A}$. The algorithm provides a generic framework for obtaining a basis $\widehat{\mathbf{V}}$ that is close to $\widetilde{\mathbf{V}}$, and the regularization step can be customized to enforce some structure in $\widehat{\mathbf{V}}$. In each iteration, the multiplication step (3.1) reduces the distance between the subspaces generated by $\mathbf{V}^{(t)}$ and $\widetilde{\mathbf{V}}$ (Theorem 7.3.1 of Golub and Van Loan (2012)), and then the regularization step (3.2) forces some structure in $\mathbf{V}^{(t)}$. Ma (2013) focused on sparsity and regularized with a thresholding function satisfying $|[\mathcal{R}(\mathbf{T}, \mathbf{\Lambda})]_{ik} - \mathbf{T}_{ik}| \leq \mathbf{\Lambda}_{ik}$ and $[\mathcal{R}(\mathbf{T}, \mathbf{\Lambda})]_{ik} \mathbb{1}(|\mathbf{T}_{ik}| \leq \mathbf{\Lambda}_{ik}) = 0$ for all $\mathbf{\Lambda}_{ik} > 0$ and $i \in [n], k \in [K]$, which includes both hard and soft thresholding. If the distance between $\mathbf{U}^{(t)}$ and $\mathbf{V}^{(t)}$ is small, then the distance between $\mathbf{V}^{(t)}$ and $\tilde{\mathbf{V}}$ keeps decreasing until a certain tolerance is reached (Proposition 6.1 of Ma (2013)). Finally, the last step in Eq. 3.3 ensure identifiability. For example, the orthogonal iteration algorithm uses the QR decomposition $\mathbf{Q}^{(t)}\mathbf{R}^{(t)} = \mathbf{U}^{(t)}$ and sets $\mathbf{V}^{(t)} = \mathbf{U}^{(t)}\mathbf{R}^{(t)^{-1}}$, which is an orthogonal matrix.

We will use the general form of the algorithm presented in Eqs. 3.1-3.3 to develop methods for estimating a sparse eigenbasis of $\mathbf{A}$, by designing regularization and identifiability steps appropriate for overlapping community detection.

3.1. Sparse Eigenbasis Estimation We propose an iterative thresholding algorithm for sparse eigenbasis estimation when the basis vectors are not necessarily orthogonal. Let $\mathbf{V}^{(t)}$ be the estimated basis at iteration t. For identifiability, we assume that this matrix has normalized columns, that is, $\|\mathbf{V}^{(t)}_{\cdot,k}\|_2 = 1$ for each $k \in [K]$, where $\mathbf{V}_{\cdot,k}$ denotes the k-th column of $\mathbf{V}$. Our algorithm is based on the following heuristic. Suppose that at some iteration t, $\mathbf{V}^{(t)}$ is close to the basis of interest. The multiplication step in Eq. 3.1

moves $\mathbf{V}^{(t)}$ closer to $\widetilde{\mathbf{V}}$, the K-leading eigenspace of $\mathbf{A}$, but the entries of $\mathbf{T}^{(t+1)} = \mathbf{A}\mathbf{V}^{(t)}$ and $\mathbf{V}^{(t)}$ are not necessarily close. Hence, before applying the regularization step, we introduce a linear transformation step that returns $\mathbf{T}^{(t+1)}$ to a value that is close to $\mathbf{V}^{(t)}$ entry-wise. This transformation is given by the solution of the optimization problem

$$\mathbf{\Gamma}^{(t+1)} = \underset{\mathbf{\Gamma}\in\mathbb{R}^{K\times K}}{\arg\min} \|\boldsymbol{V}^{(t)}\mathbf{\Gamma} - \mathbf{T}^{(t+1)}\|_F^2,$$

which has a closed form solution, $\mathbf{\Gamma}^{(t+1)} = \left[(\mathbf{V}^{(t)})^\top \mathbf{V}^{(t)}\right]^{-1} (\mathbf{V}^{(t)})^\top \mathbf{T}^{(t+1)}$. Define

$$\widetilde{\mathbf{T}}^{(t+1)} = \mathbf{T}^{(t+1)} \left(\mathbf{\Gamma}^{(t+1)}\right)^{-1}.$$

After this linear transformation, we apply a sparse regularization to $\widetilde{\mathbf{T}}^{(t+1)}$, defined by a thresholding function $\mathcal{S}$ with parameter $\lambda \in [0, 1)$,

$$\left[\mathcal{S}(\widetilde{\mathbf{T}}, \lambda)\right]_{ik} = \begin{cases} \widetilde{\mathbf{T}}_{ik} & \text{if } \widetilde{\mathbf{T}}_{ik} > \lambda \max_j |\widetilde{\mathbf{T}}_{ij}|, \\ 0 & \text{otherwise.} \end{cases} \tag{3.4}$$

The function $\mathcal{S}$ applies hard thresholding to each entry of the matrix $\widetilde{\mathbf{T}}$ with a different threshold for each row, to adjust for possible differences in the expected degree of a node. The parameter λ controls the level of sparsity, with larger values of λ giving more zeros in the solution. Finally, the new value of $\mathbf{V}$ is obtained by normalizing the columns, setting $\mathbf{U}^{(t+1)} = \mathcal{S}(\widetilde{\mathbf{T}}^{(t+1)}, \lambda)$ and

$$\mathbf{V}_{ik}^{(t+1)} = \frac{1}{\|\mathbf{U}_{\cdot,k}^{(t+1)}\|_2} \mathbf{U}_{ik}^{(t+1)},$$

for each $i \in [n]$ and $k \in [K]$.

We stop the algorithm after the relative difference in spectral norm between $\mathbf{V}^{(t)}$ and $\mathbf{V}^{(t+1)}$ is smaller than some tolerance $\epsilon > 0$, that is,

$$\frac{\|\mathbf{V}^{(t+1)} - \mathbf{V}^{(t)}\|}{\|\mathbf{V}^{(t)}\|} < \epsilon.$$

These steps are summarized in Algorithm 1.

Algorithm 1 SPCA-eig: Sparse eigenbasis estimation.

Input: Adjacency matrix $\mathbf{A}$, regularization parameter $\lambda \in [0, 1)$, initial estimator $\mathbf{V}^{(0)}$ with normalized columns.

for $t = 1, \ldots$ until convergence **do**

1. Update $\mathbf{T}^{(t)} = \mathbf{A}\mathbf{V}^{(t-1)}$.
2. Update $\widetilde{\mathbf{T}}^{(t)} = \mathbf{T}^{(t)}[(\mathbf{V}^{(t-1)})^\top \mathbf{T}^{(t)}]^{-1}[(\mathbf{V}^{(t-1)})^\top \mathbf{V}^{(t-1)}]$.
3. Thresholding: $\mathbf{U}^{(t)} = \mathcal{S}(\widetilde{\mathbf{T}}^{(t)}, \lambda)$.
4. Normalization: $\mathbf{V}^{(t)}_{\cdot,k} = \frac{1}{\|\mathbf{U}^{(t)}_{\cdot,k}\|_2}\mathbf{U}^{(t)}_{\cdot,k}$, for $k \in [K]$.

end for

return $\hat{\mathbf{V}}_\lambda = \mathbf{V}^{(t)}$ the value at convergence.

The following proposition shows that when Algorithm 1 is applied to the expected probability matrix $\mathbf{P}$ that has a sparse basis, then there exists a fixed point that has the correct support. In particular, this implies that for the expected probability matrix of an OCCAM graph defined in Eq. 2.1, the entries of this fixed point coincide with the support of the overlapping memberships of the model. The proof is given on the Appendix.

Proposition 3. *Let $\mathbf{P} \in \mathbb{R}^{n\times n}$ be a symmetric matrix with rank$(\mathbf{P}) = K < n$, and suppose that there exists a non-negative sparse basis $\mathbf{V}$ of the principal subspace of $\mathbf{P}$. Let $\widetilde{\mathbf{V}}$ be a matrix such that $\widetilde{\mathbf{V}}_{\cdot,k} = \frac{1}{\|\boldsymbol{V}_{\cdot,k}\|_2}\mathbf{V}_{\cdot,k}$ for each $k \in [K]$, and $v^* = \min\{\mathbf{V}_{ik}/\|\mathbf{V}_{i\cdot}\|_\infty | (i,k) \in supp(\mathbf{V})\}$. Then, for any $\lambda \in [0, v^*)$, the matrix $\widetilde{\mathbf{V}}$ is a fixed point of Algorithm 1 applied to $\mathbf{P}$.*

When the algorithm is applied to an adjacency matrix $\mathbf{A}$, the matrix $\mathbf{V}$ is not exactly a fixed point, but the norm of the difference between $\mathbf{V}$ and $\mathbf{V}^{(1)}$ will be a function of the distance between the principal subspaces of $\mathbf{A}$ and $\mathbf{P}$. Concentration results (Le et al., 2017) ensure that $\mathbf{A}$ is close to its expected value $\mathbf{P}$, specifically, $\|\mathbf{A} - \mathbf{P}\| = O(\sqrt{d})$ (where $\|\cdot\|$ is the spectral norm of a matrix) with high probability as long as the largest expected degree $d = \max_{i\in[n]} \mathbf{P}_{ij}$ satisfies $d = \Omega(\log n)$. If the K leading eigenvalues of $\mathbf{V}$ are sufficiently large, then the principal subspaces of $\mathbf{A}$ and $\mathbf{P}$ are close to each other (Yu et al., 2015),

3.1.1. Community Detection in Networks with Homogeneous Degrees. Here, we present a second algorithm for the estimation of sparse community memberships in graphs for homogeneous expected degree of nodes within a

community. Specifically, we focus on graphs for which the expected adjacency matrix $\mathbf{P} = \mathbb{E}[\mathbf{A}]$ has the form

$$\mathbf{P} = \mathbf{Z}\mathbf{B}\mathbf{Z}^\top, \tag{3.5}$$

where $\mathbf{Z} \in \mathbb{R}^{n\times K}$ is a membership matrix such that $\|\mathbf{Z}_{i,\cdot}\|_1 = \sum_{k=1}^{K} |\mathbf{Z}_{ik}| = 1$, and $\mathbf{B} \in \mathbb{R}^{K\times K}$ is a full-rank matrix. This model is a special case of OCCAM, when the degree heterogeneity parameter $\mathbf{\Theta}$ in Eq. 2.1 is constant for all vertices. In particular, this case includes the classic SBM (Holland et al., 1983) when the memberships do not overlap.

To enforce degree homogeneity, we add an additional normalization step, so that the matrix $\hat{\mathbf{Z}}$ has rows with constant norm $\|\hat{\mathbf{Z}}_{i,\cdot}\|_1 = 1$ as in Eq. 3.5. In practice we observed that this normalization gives very accurate results in terms of community detection. After the multiplication step $\mathbf{T}^{(t)} = \mathbf{A}\mathbf{V}^{(t-1)}$, the columns of $\mathbf{T}^{(t)}$ are proportional to the norm of the columns $\mathbf{V}^{(t-1)}$, which is in turn proportional to the estimated community sizes. In order to remove the effect of this scaling with community size, which is not meaningful for community detection, we normalize the columns of $\mathbf{V}^{(t-1)}$, and then perform the thresholding and the row normalization step as before. These steps are summarized in Algorithm 2.

Algorithm 2 SPCA-CD: Sparse eigenbasis estimation for networks with homogeneous degrees.

Input: Adjacency matrix $\mathbf{A}$, regularization parameter $\lambda \in [0, 1)$, initial estimator $\mathbf{V}^{(0)}$ with normalized rows.

for $t = 1, \ldots$ until convergence **do**

1. Update $\mathbf{T}^{(t)} = \mathbf{A}\mathbf{V}^{(t-1)}$.
2. Column normalization: $\widetilde{\mathbf{T}}^{(t)}_{\cdot,k} = \frac{1}{\|\boldsymbol{T}^{(t)}_{\cdot,k}\|_1}\mathbf{T}^{(t)}_{\cdot,j}$, for $k \in [K]$.
3. Thresholding: $\mathbf{U}^{(t)} = \mathcal{S}(\widetilde{\mathbf{T}}^{(t)}, \lambda)$.
4. Row normalization: $\mathbf{V}^{(t)}_{i,\cdot} = \frac{1}{\|\mathbf{U}^{(t)}_{i,\cdot}\|_1}\mathbf{U}^{(t)}_{i,\cdot}$, for $i \in [n]$.

end for

return $\hat{\mathbf{V}}_\lambda = \mathbf{V}^{(t)}$ the value at convergence.

The next theorem shows that in the case of the planted partition SBM, a matrix with the correct sparsity pattern is a fixed point of Algorithm 2. Note that since the algorithm does not assume that each node belongs to a single community, this result not only guarantees that there exist a fixed

point that correctly cluster the nodes into communities, as typical goal of of community detection, but also that is able to distinguish if a node belongs to more than one community or not. The proof is given on the appendix.

THEOREM 1. *Let* $\mathbf{A}$ *be a network generated from a SBM with* K *communities of sizes* $n_1, \ldots, n_K$*, membership matrix* $\mathbf{Z}$ *and connectivity matrix* $\mathbf{B} \in [0,1]^{K \times K}$ *of the form*

$$\mathbf{B}_{rs} = \begin{cases} p, & \text{if } r = s, \\ q, & \text{if } r \neq s, \end{cases}$$

for some $p, q \in [0,1]$*,* $p > q$*. Suppose that for some* $\lambda^* \in (0,1)$ *and some* $c_1 > 2$*,*

$$\lambda^* p - q > c_1 \sqrt{\frac{\log(Kn)}{\min_k n_k}}, \tag{3.6}$$

Then, for any $\lambda \in (\lambda^*, 1)$*,* $\mathbf{Z}$ *is a fixed point of Algorithm 2 with probability at least* $1 - n^{c_1 - 1}$*.*

3.2. Selecting the Thresholding Parameter Our algorithms require two user-supplied parameters: the number of communities K and the threshold level λ. The parameter λ controls the sparsity of the estimated basis $\hat{\mathbf{V}}$. In practice, looking at the full path of solutions for different values of λ may be informative, as controlling the number of overlapping memberships can result in different community assignments. On the other hand, it is practically useful to select an appropriate value λ that provides a good fit to the data. We discuss two possible techniques for choosing this parameter, the Bayesian Information Criterion (BIC) and edge cross-validation (ECV) (Li et al., 2020). Here we assume that the number of communities K is given, but choosing the number of communities is also an important problem, with multiple methods available for solving it (Wang and Bickel, 2017; Le and Levina, 2015; Li et al., 2020). If computational resources allow, K can be chosen by cross-validation along with λ.

The goodness-of-fit can be measured via the likelihood of the model for the graph $\mathbf{A}$, which depends on the probability matrix $\mathbf{P} = \mathbb{E}[\mathbf{A}]$. Given $\widehat{\mathbf{V}}$, a natural estimator for $\mathbf{P}$ is the projection of $\mathbf{A}$ onto the subspace spanned by $\widehat{\mathbf{V}}$, which can be formulated as

$$\begin{aligned} \widehat{\mathbf{P}} = \arg\min_{\boldsymbol{P}} \quad & \|\mathbf{A} - \mathbf{P}\|_F^2 \\ \text{subject to} \quad & \mathbf{P} = \widehat{\mathbf{V}} \mathbf{B} \widehat{\mathbf{V}}^\top, \ \mathbf{B} \in \mathbb{R}^{K \times K}. \end{aligned} \tag{3.7}$$

This optimization problem finds the least squares estimator of a matrix constrained to the set of symmetric matrices with a principal subspace defined by $\widehat{\mathbf{V}}$, and has a closed-form solution, stated in the following proposition.

Proposition 4. *Let* $\widehat{\mathbf{P}}$ *be the solution of the optimization problem* (3.7)*, and suppose that* $\widehat{\mathbf{Q}} \in \mathbb{R}^{n\times K}$ *is a matrix with orthonormal columns such that* $\widehat{\boldsymbol{V}} = \widehat{\mathbf{Q}}\widehat{\mathbf{R}}$ *for some matrix* $\widehat{\mathbf{R}} \in \mathbb{R}^{K\times K}$*. Then,*

$$\widehat{\mathbf{P}} = \widehat{\mathbf{Q}}\left(\widehat{\mathbf{Q}}^\top \mathbf{A}\widehat{\mathbf{Q}}\right)\widehat{\mathbf{Q}}^\top.$$

The Bayesian Information Criterion (BIC) (Schwarz, 1978) provides a general way of choosing a tuning parameter by balancing the fit of the model measured with the log-likelihood of $\boldsymbol{A}$, and a penalty for the complexity of a model that is proportional to the number of parameters. The number of non-zeros in $\mathbf{V}$ given by $\|\mathbf{V}\|_0$ can be used as a proxy for the degrees of freedom, and the sample size is taken to be the number of independent edges in $\mathbf{A}$. Then the BIC for a given λ can be written as

$$\mathcal{P}_{\mathrm{BIC}}(\lambda) = -2\ell(\widehat{\mathbf{P}}_\lambda) + \|\widehat{\mathbf{V}}_\lambda\|_0 \log(n(n-1)/2), \tag{3.8}$$

where $\widehat{\mathbf{P}}_\lambda$ is the estimate for $\mathbf{P}$ defined in Proposition 4 for $\widehat{\mathbf{V}}_\lambda$.

The BIC criterion (3.8) has the advantage of being simple to calculate, but it has some issues. First, the BIC is derived for a maximum likelihood estimator, while $\widehat{\mathbf{P}}$ is not obtained in this way, and this is only a heuristic. Further, the least squares estimator $\widehat{\mathbf{P}}$ is not guaranteed to result in a valid estimated edge probability (between 0 and 1). A possible practical solution is to modify the estimate by defining $\widetilde{\mathbf{P}} \in [0,1]^{n\times n}$ as $\widetilde{\mathbf{P}}_{ij} = \min(\max(\widehat{\mathbf{P}}_{ij}, \epsilon), 1-\epsilon)$ for some small value of $\epsilon \in (0,1)$.

Another alternative for choosing the tuning parameter is edge cross-validation (CV). Li et al. (2020) introduced a CV method for network data based on splitting the set of node pairs $\mathcal{N} = \{(i,j) : i,j \in \{1,\dots,N\}\}$ into L folds. For each fold $l = 1,\dots,L$, the corresponding set of node pairs $\Omega_l \subset \mathcal{N}$ is excluded, and the rest are used to fit the basis $\mathbf{V}$. Li et al. (2020) propose to use a matrix completion algorithm based on the rank K truncated SVD to fill in the entries missing after excluding Ω_l, resulting in a matrix $\widehat{\mathbb{M}}_l \in \mathbb{R}^{n\times n}$. Then, for a given λ we estimate $\widehat{\mathbf{V}}_\lambda$, and use Proposition 4 to obtain an estimate $\widehat{\mathbf{P}}_\lambda(\widehat{\mathbf{M}}^{(l)})$ of $\mathbf{P}$. The error on the held-out edge set is measured by

$$\mathcal{P}_{\mathrm{CV}}(\mathbf{A}, \widehat{\mathbf{P}}_\lambda(\widehat{\mathbf{M}}_l); \Omega_l) = \frac{1}{|\Omega_l|}\sum_{(i,j)\in\Omega_l} (\mathbf{A}_{ij} - \hat{\mathbf{P}}_\lambda(\hat{\mathbf{M}}_l)_{ij})^2, \tag{3.9}$$

and the tuning parameter λ is selected to minimize the average cross-validation error

$$\mathcal{P}_{\text{CV}}(\lambda) = \frac{1}{L}\sum_{l=1}^{L} \mathcal{P}_{\text{CV}}(\mathbf{A}, \widehat{\mathbf{P}}_\lambda(\widehat{\mathbf{M}}_l); \Omega_l).$$

The edge CV method does not rely in a specific model for the graph, which can be convenient in the settings mentioned before, but its computational cost is larger. In practice, we observe that edge CV tends to select more complex models in which nodes are assigned to more communities than the solution selected with BIC (see Section 4.2).

4 Numerical Evaluation on Synthetic Networks

We start with evaluating our methods and comparing them to benchmarks on simulated networks. In all scenarios, we generate networks from OCCAM, thus edges of $\mathbf{A}$ are independent Bernoulli random variables, with expectation given by Eq. 2.1. We assume that each row vector $\mathbf{z}_i \in \mathbb{R}^K$ of $\mathbf{Z} = [\mathbf{z}_1, \ldots, \mathbf{z}_n]^\top$ satisfies $\|\mathbf{z}_i\|_1 = 1$, so each node has the same expected degree. To better understand what affects the performance, we evaluate the methods by varying one parameter from the following list at a time; all of them affect the difficulty of detecting overlapping communities.

a) Fraction of nodes belonging to more than one community $\widetilde{p}$ (the higher $\widetilde{p}$, the more difficult the problem). For a given $\widetilde{p} \in [0, 1)$, we select $\widetilde{p}n$ nodes for the overlaps, and assign the rest to only one community, distributed equally among all the communities. For most of the experiments we use $K = 3$ communities, and 1/4 of the overlapping nodes are assigned to all communities with $\mathbf{z}_i = [1/3, 1/3, 1/3]^T$, while the rest are assigned to two communities j, k, with $\mathbf{z}_{ij} = \mathbf{z}_{ik} = 1/2$, equally distributing these nodes on all pairs (j, k). When $K > 3$, we only assign the nodes to one or two communities following the same process, but we do not include overlaps with more than three communities.

b) Connectivity between communities ρ (the higher ρ, the more difficult the problem). We parameterize $\mathbf{B}$ as

$$\mathbf{B} = (1 - \rho)\boldsymbol{I}_K + \rho \mathbf{1}_K \mathbf{1}_K^\top,$$

and vary ρ in a range of values between 0 and 1.

c) Average degree of the network d (the higher d, the easier the problem). For a given average degree d, we set α in Eq. 2.1 so that the expected average degree $\frac{1}{n}\mathbf{1}_n^\top \mathbb{E}[\mathbf{A}]\mathbf{1}_n$ is equal to d.

d) Node degree heterogeneity (the more heterogeneous the degrees, the harder the problem). This is controlled by parameter $\boldsymbol{\theta} = \text{diag}(\boldsymbol{\Theta})$, and in most simulations we set $\boldsymbol{\theta}_i = 1\ \forall i \in [n]$ so all nodes have the same degree, but in some scenarios we also introduce hub nodes by setting $\boldsymbol{\theta}_i = 5$ with probability 0.1 and $\boldsymbol{\theta}_i = 1$ with probability 0.9.

e) Number of communities K (the larger K, the harder the problem). For all values of K, we maintain communities of equal size.

In most scenarios, we fix $n = 500$, and $K = 3$. All simulation settings are run 50 times, and the average result together with its 95% confidence band are reported. An implementation of the method in R can be found at https://github.com/jesusdaniel/spcaCD.

Our main goal is to find the set of non-zero elements of the membership matrix. Many measures can be adopted to evaluate a solution; here we use the *normalized variation of information* (NVI) introduced by Lancichinetti et al. (2009), which is specifically designed for problems with overlapping clusters. Given a pair of binary random vectors X, Y of length K, the normalized conditional entropy of X with respect to Y can be defined as

$$H_{norm}(X|Y) = \frac{1}{K}\sum_{k=1}^{K}\frac{H(X_k|Y_k)}{H(X_k)},$$

where $H(X_k)$ is the entropy of X_k and $H(X_k|Y_k)$ is the conditional entropy of X_k given Y_k, defined as

$$H(X_k) = -P(X_k=0)\log P(X_k=0) - P(X_k=1)\log P(X_k=1) \quad (4.1)$$

$$H(X_k, Y_k) = -\sum_{a=0}^{1}\sum_{b=0}^{1} P(X_k=a, Y_k=b)\log P(X_k=a, Y_k=b) \quad (4.2)$$

$$H(X_k|Y_k) = H(X_k, Y_k) - H(Y_k),$$

and the normalized variation of information between X and Y is defined as

$$N(X|Y) = 1 - \min_{\sigma}\frac{1}{2}\left(H_{norm}(\sigma(X)|Y) + H_{norm}(Y|\sigma(X))\right), \quad (4.3)$$

where σ is a permutation of the indexes to account for the fact that the binary assignments can be equivalent up to a permutation. The NVI is always a number between 0 and 1; it is equal to 0 when X and Y are independent, and to 1 if $X = Y$.

For a given pair of binary membership matrices $\mathbf{Z}$ and $\widetilde{\mathbf{Z}}$ with binary entries indicating community memberships, we can use the rows of replace the probabilities in Eqs. 4.1 and 4.2 with the sample versions using the rows of $\widetilde{\mathbf{Z}}$ and $\mathbf{Z}$, that is

$$\hat{P}(X_k = a) = \frac{1}{n}\sum_{i=1}^{n} 1\{\widetilde{\boldsymbol{Z}}_{ik} = a\}, \qquad \hat{P}(Y_k = b) = \frac{1}{n}\sum_{i=1}^{n} 1\{\boldsymbol{Z}_{ik} = b\},$$

$$\hat{P}(X_k = a, Y_k = b) = \frac{1}{n}\sum_{i=1}^{n} 1\{\widetilde{\boldsymbol{Z}}_{ik} = a, \boldsymbol{Z}_{ik} = b\},$$

for $a, b \in \{0, 1\}$.

4.1. Choice of Initial Value We start from comparing several initialization strategies:

- An overlapping community assignment, from the method for fitting OCCAM.
- A non-overlapping community assignment, from SCORE (Jin, 2015), a spectral clustering method designed for networks with heterogeneous degrees.
- Multiple random non-overlapping community assignments, with each node randomly assigned to only one community. We use five different random values and take the solution corresponding the smallest error as measured by Eq. 3.7.

We compare these initialization schemes with fixed $n = 500$, $K = 3$, $d = 50$, and varying between-community connectivity ρ and the fraction of overlapping nodes $\widetilde{p}$. For both our methods (SPCA-eig and SPCA-CD), we fit solution paths over a range of values $\lambda = \{0.05, 0.1, \ldots, 0.95\}$, and report the solution with the highest NVI for each of the methods (note that we are not selecting λ in a data-driven way in order to reduce variation that is not related to initialization choices).

Figure 1 shows the results on initialization strategies. In general, all methods perform worse as the problem becomes harder, and the non-random initializations perform better overall; the multiple random initializations are also sufficient for the easier case of few nodes in overlaps. For the rest of the paper, unless explicitly stated, we use the non-overlapping community detection solution (SCORE) to initialize the algorithm, given its good performance and low computational cost.

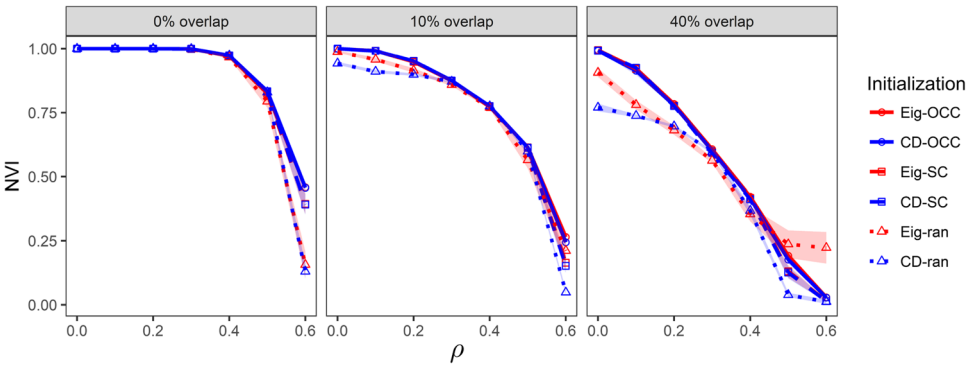

Figure 1: Performance measured by NVI under different initialization strategies (OCCAM, SCORE and five random initial values). The errors are plotted as a function of connectivity between communities ρ for three different values of the overlap $\widetilde{p}$

4.2. Choosing the Threshold The tuning parameter λ controls the sparsity of the solution, and hence the fraction of pure nodes. Since community detection is an unsupervised problem, it may be useful in practice to look at the entire path over λ and consider multiple solutions with different levels of sparsity (see Section 5.1). However, we may also want to choose a single value of λ that balances a good fit and a parsimonious solution. Here, we evaluate the performance of the two strategies for choosing λ proposed in Section 3.2, BIC and CV, using the same simulation setting than the previous section.

Figure 2 shows the average performance measured by NVI of the two tuning methods. The BIC tends to select sparser solutions than CV, and hence when the true membership matrix is sparse (few overlaps), BIC outperforms CV, but with more overlap in communities, CV usually performs better, specially for SPCA-CD. Since there is no clear winner overall, we use BIC in subsequent analysis, because it is computationally cheaper.

4.3. Comparison with Existing Methods We compare our proposal to several state-of-the-art methods for overlapping community detection. We use the same simulation settings as in the previous section ($n = 500$ and $K = 3$), including sparser scenarios with $d = 20$, and networks with heterogeneous degrees ($d = 50$ and 10% of nodes are hubs).

We select competitors based on good performance reported in previous studies. As representative examples of spectral methods, we include OCCAM fitted by the algorithm of Zhang et al. (2020) and Mixed-SCORE (Jin

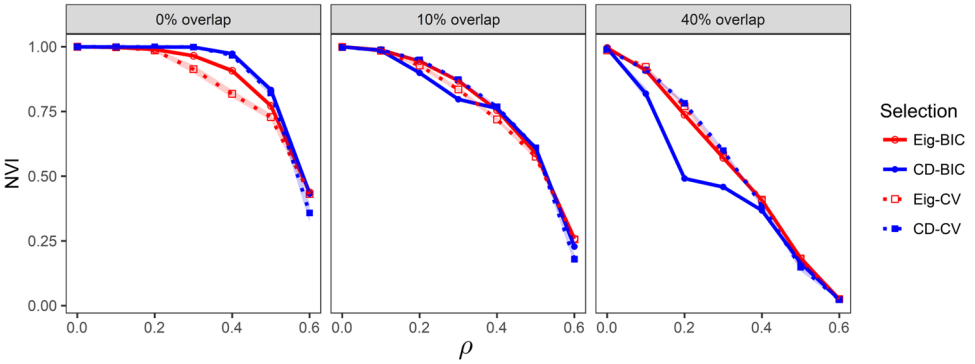

Figure 2: Performance of two tuning methods, BIC and CV, as measured by NVI. The errors are plotted as a function of connectivity between communities ρ for three different values of the overlap $\widetilde{p}$

et al., 2017). We also include the EM algorithm for the BKN model (Ball et al., 2011) and the overlapping SBM of Latouche et al. (2011) (OSBM), and Bayesian non-negative matrix factorization (BNMF) by Psorakis et al. (2011). For methods that return a continuous membership assignment (OCCAM, BKN and Mixed-SCORE), we follow the approach of Zhang et al. (2020) and set to zero the values of the membership matrix $\hat{\mathbf{Z}}$ that are smaller than $1/K$.

Figure 3 shows the average NVI of these methods as a function of ρ under different scenarios. Most methods show an excellent performance when $\rho = 0$, but as the between-community connectivity increases, the performance of all methods deteriorate. Our methods (SPCA-CD and SPCA-eig) generally achieve the best performance when the fraction of nodes in overlaps is either 0 or 10%, and are highly competitive for 40% in overlaps as well. OCCAM performs well, which is reasonable since the networks were generated from this model, but it appears that in most cases we are able to fit it more accurately. Mixed-SCORE has a good performance with no overlaps, but deteriorates quicker than other methods with the introduction of overlaps. We should keep in mind that OCCAM and Mixed-SCORE are designed for estimating continuous memberships, and the threshold of $1/K$ to obtain binary memberships might not be optimal. While non-overlapping community detection methods can be alternatively used for the scenario when there is only a single membership per node, our methods are able to accurately assign the nodes to a single community without knowing the number of memberships.

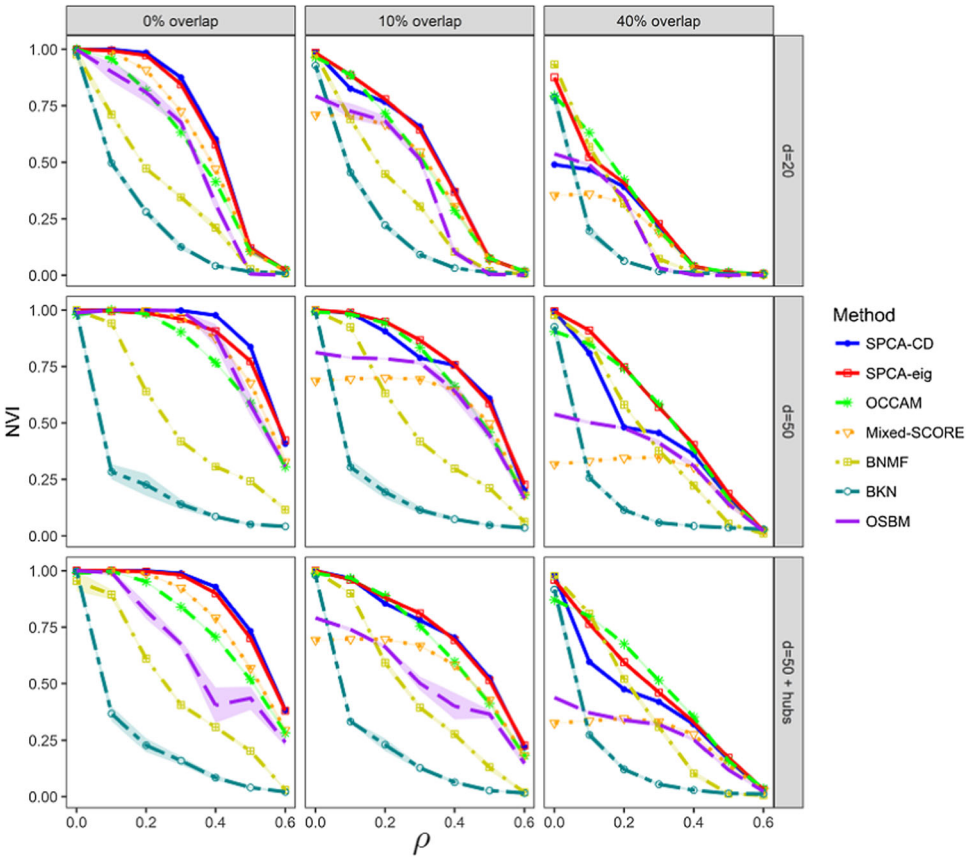

Figure 3: Performance of different methods for overlapping community detection measured by NVI, as a function of ρ, for different amounts of overlaps (columns) and node degrees (rows)

4.4. Computational Efficiency Scalability to large networks is an important issue for real data applications. Spectral methods for overlapping and non-overlapping community detection are very popular, partly due to its scalability to large networks. The accuracy of those methods usually depends on the clustering algorithm, which in practice might require multiple initial values to get an accurate result. In contrast, our methods based on sparse principal component analysis directly estimate the membership matrix without having to estimate the eigenvectors or perform a clustering step. Although the accuracy of our methods does depend on the tuning parameter λ, the algorithms are robust to the choice of this parameter and provide good solutions over a reasonably wide range.

To compare computation efficiency empirically, we simulated networks with different number of communities ($K = 3, 6$ and 10) and increased the number of nodes while keeping the average degree fixed to $d = 50$, with 10% overlapping nodes. For simplicity, we used a single fixed value $\lambda = 0.6$ for our methods. We initialized SPCA-CD with a random membership matrix, and SPCA-eig with the SPCA-CD as starting point, and therefore report the running time of SPCA-eig as the sum of the two. We compare the performance of our methods with OCCAM, which uses a k-medians clustering to find the centroids of the overlapping communities. Since k-medians is computationally expensive and is not able to handle large networks, we also report the performance of OCCAM with the clustering step performed with k-means instead. Additionally, we report the running time of calculating the K leading eigenvectors of the adjacency matrix, which is a starting step required by spectral methods. All simulations are run using Matlab R2015a. The leading eigenvectors of the adjacency matrix are computed using the standard Matlab function `eigs`$(\cdot, K)$.

The performance in terms of time and accuracy of different methods is shown in Fig. 4. Our methods based on SPCA incur a computational cost similar to that calculating the K leading eigenvectors of the adjacency matrix, and when the number of communities is not large, our methods perform even faster. The original version of OCCAM based on k-medians clustering is limited in the size of networks it can handle, and when using k-means the computational cost is still larger than SPCA. Our methods produce very accurate solutions in all the scenarios considered, while OCCAM deteriorates when the number of communities increases. Note that in general the performance of all methods can be improved by using different random starting values, either for clustering in OCCAM or for initializing our methods, but this will increase the computational cost; choosing tuning parameters, if the generic robust choice is not considered sufficient, will do the same.

5 Evaluation on Real-World Networks

In this section, we evaluate the performance of our methods on several real-world networks. Zachary's karate club network (Zachary, 1977) and the political blog network (Adamic and Glance, 2005) are two classic examples with community structure, and we start with them as an illustration. We then compare our method with other state-of-the-art overlapping community detection algorithms on the popular benchmark dataset focused specifically on overlapping communities (McAuley and Leskovec, 2012), which contains many social ego-networks from Facebook, Twitter and Google Plus. Finally,

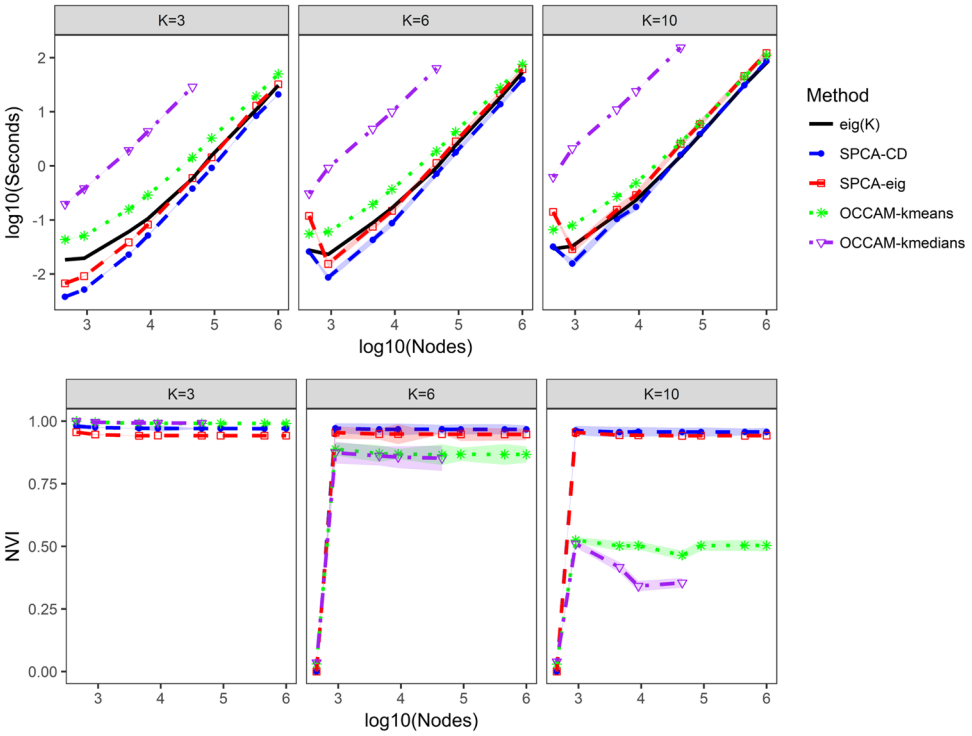

Figure 4: Performance of different methods in terms of running time (top row) and NVI (bottom row) as a function of the size of the network n for varying number of communities. We compare SPCA-CD and SPCA-eig, OCCAM with two different clustering options (k-means and k-medians), and the computational cost of calculating the K leading eigenvectors (eig(K))

we use our methodology to identify communities in a novel dataset consisting of Twitter following relationship between national representatives in the Mexican Chamber of Deputies.

5.1. Zachary's Karate Club Network Zachary (1977) recorded the real-life interactions of 34 members of a karate club from a period of two years. During this period, the club split into two factions due to a conflict between the leaders, and these factions are taken to be the ground truth communities.

We fit our methods to the karate club network, and if we use either BIC or CV to choose the optimal threshold parameter, the solution consists of two community with only pure nodes and matches the ground truth. This serves as reassurance that our method will not force overlaps on the communities

when there are not actually there. In contrast, OCCAM assigns 17 nodes (50%) to both communities, and mixed-SCORE assigns 26 (76%).

If we look at the entire path over the threshold parameter λ, we can also see which nodes are potential overlaps. Both our methods can identify community memberships, but SPCA-eig also provides information on the degree-correction parameter. In Fig. 5, we examine the effect of the threshold parameter λ on SPCA-eig solutions. The plots show the paths of the node membership vectors as a function of λ. Each panel corresponds to one of the columns of the membership matrix, the colors indicate the true factions, and the paths of the faction leaders are indicated with a dashed line. The y-axis shows the association of the node to the corresponding community, with membership weighted by the degree-corrected parameter. In each community, the nodes with the largest values of y are the faction leaders, which are connected to most of the nodes in the faction. For larger values of λ, all nodes are assigned to pure communities corresponding to true factions, but as λ decreases the membership matrix contains more non-zero values.

5.2. The Political Blogs Network The political blogs network (Adamic and Glance, 2005) represents the hyperlinks between 1490 political blogs around the time of the 2004 US presidential election. The blogs were manually labeled as liberal or conservative, which is taken as ground truth, again

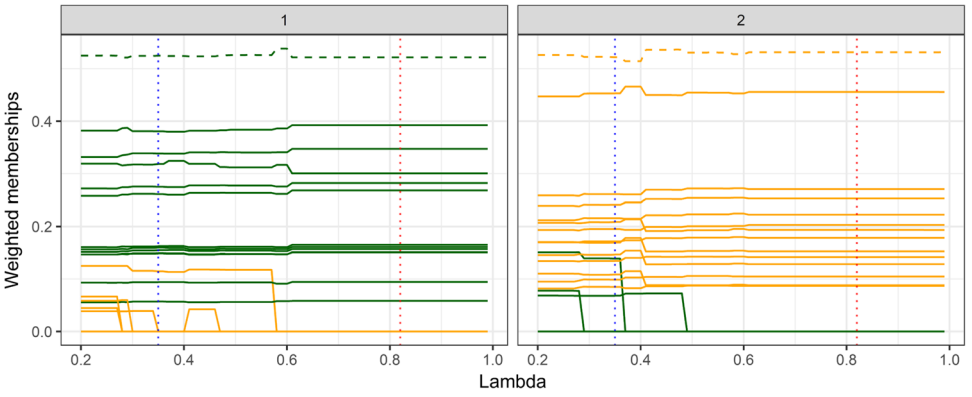

Figure 5: Node membership paths in each discovered community (left and right) as a function of the thresholding parameter λ. Colors represent ground truth, and the dashed line represents the leader of each faction. The dashed vertical lines correspond to the values of λ chosen by BIC (blue) and CV (red). As the thresholding parameter decreases, the membership matrix becomes sparser and nodes appear on both communities

without any overlaps. This dataset is more challenging for community detection than the karate club network, due to the high degree heterogeneity (Karrer and Newman, 2011). Following the literature, we focus on the largest connected component of the network, which contains 1222 nodes, and convert the edges to undirected, so $\mathbf{A}_{ij} = 1$ if either blog i has an hyperlink to blog j or vice versa.

Figure 6 shows the plot of the political blog network membership paths using Algorithm 2 as a function of λ and colored by the ground truth labels. Using the tuning parameter selected by BIC, the algorithm assigns only 29 nodes to both communities. Other overlapping community methods assigned many more nodes to both communities: 229 (19%) for OCCAM, and 195 (16%) for mixed-SCORE. To convert the estimated solution into non-overlapping memberships in order to compare with the ground truth, each node is assigned to the community corresponding to the largest entry on the corresponding row, resulting in 52 misclustered nodes, a result similar to other community detection methods that are able to operate on networks with heterogeneous node degrees (Jin, 2015). The membership paths that correspond to these misclustered nodes are represented with dashed lines. The fact that most of the overlapping nodes discovered by the algorithm were incorrectly clustered supports the idea that these are indeed overlapping nodes, as the disagreement between the unsupervised clustering result

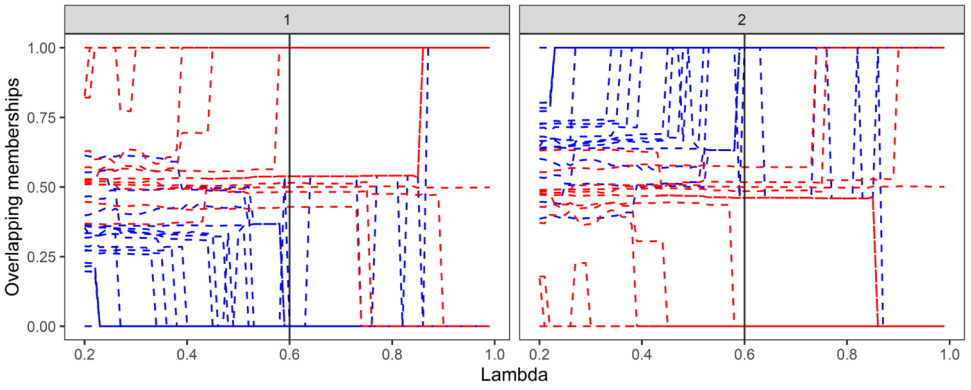

Figure 6: Node membership paths in the two discovered communities (left and right) as a function of the thresholding parameter λ. Colors represent ground truth, and the dashed lines represent misclassified nodes. Most of the misclassified nodes are identified as overlapping. The black vertical lines correspond to the values of λ chosen by BIC

and the label given by the authors might indicate that these are nodes with no clear membership.

5.3. The SNAP Social Networks Social media platforms provide a rich source of data for the study of social interactions. McAuley and Leskovec (2012) presented a large collection of ego-networks from Facebook, Google Plus and Twitter. An ego-network represents the virtual friendships or following-follower relationships between a group of people that are connected to a central user. Those platforms allow the users to manually label or classify their friends into groups or social circles, and this information can be used as a ground truth to compare the performance of methods for detecting communities. In Zhang et al. (2020), several state-of-the-art overlapping community detection methods were compared on these data, showing a competitive performance of OCCAM. We again include OCCAM and Mixed-SCORE as examples of spectral methods for overlapping community detection. We obtained a pre-processed version of the data directly from the first author of Zhang et al. (2020); for the details on the pre-processing steps, see Section 6 of Zhang et al. (2020).

Table 1 shows the average performance measured by NVI for the community detection methods we compared. For our methods, the value of λ was chosen by BIC, like in simulations. For OCCAM and mixed-SCORE, we thresholded continuous membership assignments at $1/K$. Our methods (SPCA-eig and SPCA-CD) show a slightly better performance than the rest of the methods in the Facebook networks. SPCA-CD performs better than other methods on the Twitter networks, but SPCA-eig does not perform better than OCCAM. For Google Plus networks, OCCAM and mixed-SCORE have a clear advantadge. Figure 7 presents a visualization of the distribution of of several network summary statistics for each social media platform. It suggests that Google Plus networks might be harder because they tend to have more overlaps between communities, although they also tend to have more nodes. Facebook networks, in contrast, have higher modularity values

Table 1: Average performance measured by NVI (with standard errors in parentheses) of overlapping community detection methods on SNAP ego-networks

Dataset (sample size)	SPCA-Eig	SPCA-CD	OCCAM	M-SCORE
Facebook (6)	0.573 (0.090)	**0.588(0.088)**	0.548 (0.118)	0.493 (0.137)
Google Plus (39)	0.408 (0.047)	0.427 (0.048)	**0.501(0.039)**	0.475 (0.039)
Twitter (168)	0.435 (0.021)	**0.477(0.021)**	0.450 (0.021)	0.391 (0.020)

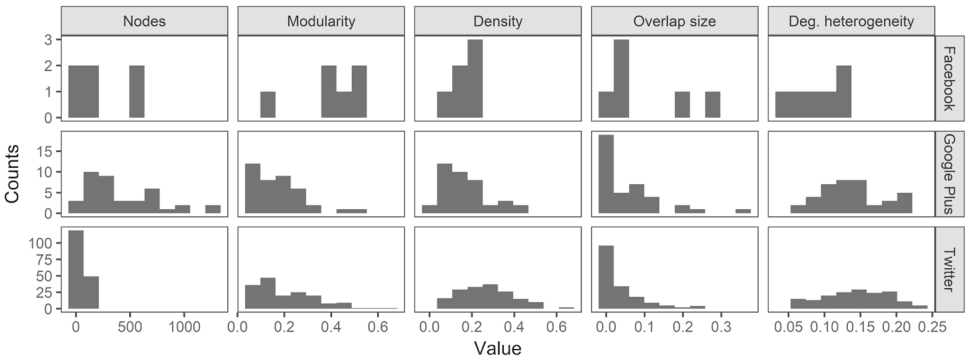

Figure 7: Histograms of summary statistics for SNAP ego-networks. McAuley and Leskovec (2012). The histograms show the number of nodes (n), Newman-Girvan modularity, density (number of edges divided by n^2), overlap size (percentage of nodes with overlapping memberships) and degree heterogeneity measured by the standard deviation of node degrees divided by n)

and smaller overlaps, and thus should be easier to cluster. In general, all methods perform reasonably, with SPCA-CD given the best overall performance Facebook and Twitter networks, and OCCAM being overall best on Google Plus. This is consistent with what we observed in simulations and what we would expect by design: our methods are more likely to perform better than others when membership vectors are sparse.

5.4. Twitter Network of Mexican Representatives We consider the Twitter network between members of the Mexican Chamber of Deputies (the lower house of the Mexican parliament), from the LXIII Legislature for the period of 2015-2018. The network captures a snapshot of Twitter data from December 31st, 2017, and has 409 nodes corresponding to the representatives with a valid Twitter handle. Two nodes are connected by an edge if at least one of them follows the other on Twitter; we ignore the direction. Each member belongs to one of eight political parties or is an independent, resulting in $K = 9$ true communities; see Fig. 8. The data can be downloaded from https://github.com/jesusdaniel/spcaCD/data.

We apply Algorithm 1 to this network, using 20-fold edge cross-validation to estimate the number of communities and choose thresholding parameters. Figure 9 shows the average MSE across all the folds, minimized when $K = 10$. However, solutions corresponding to all K from 8 to 11 are qualitatively very similar, the only difference being that the largest parties (PRI and PAN) get split into smaller communities, with clusters containing the most

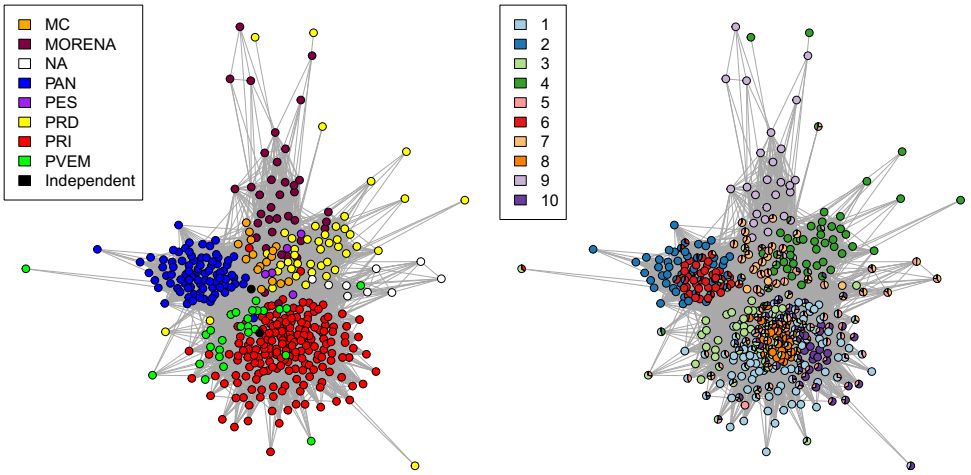

Figure 8: Twitter network of Mexican Chamber of Deputies from 2017. Left: party affiliations (true communities); right: output of Algorithm 1 with estimated $K = 10$

popular members of each party, and/or factions within a party that are more connected to some of the other parties.

A comparison between the estimated membership vectors and party affiliations reveals that our algorithm discovers meaningful overlapping communities. Table 2 compares the estimated overlapping memberships with the party labels, by counting the number of nodes that are assigned to a

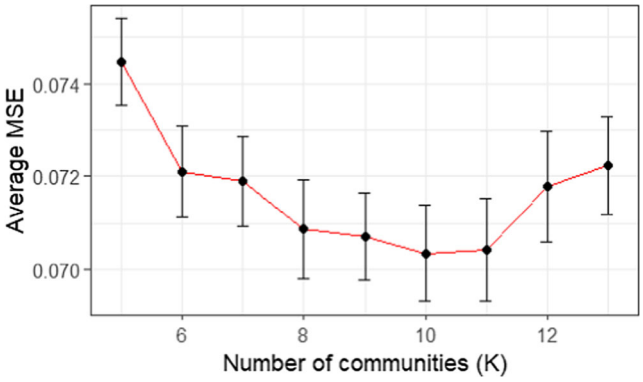

Figure 9: Average cross-validation mean squared error (MSE) over 20 cross-validation folds as a function of the number of communities (the error bars indicate two standard errors)

Table 2: Each entry indicates the number of nodes from the party (row) assigned to a given community (column)

	Estimated communities									
Party (#nodes)	1	2	3	4	5	6	7	8	9	10
MC (16)	0	0	0	0	14	0	16	0	0	0
MORENA (37)	1	0	1	9	4	0	2	0	34	1
NA (9)	2	0	4	1	8	0	9	0	0	1
PAN (84)	0	53	2	0	2	49	3	1	0	0
PES (5)	0	0	2	0	3	0	5	0	0	1
PRD (44)	2	1	3	42	2	0	2	0	0	3
PRI (185)	133	1	52	3	71	3	12	53	2	59
PVEM (27)	1	0	27	0	3	1	0	1	0	0
Independent (2)	1	0	0	0	1	0	1	1	0	0

Nodes are counted multiple times if they were assigned to more than one community

given community (recall that each node can be assigned to more than one community) and belong to a specific party. Some of the communities contain representatives from two or more different parties, which is a reflection of coalitions and factions. For example, the majority of nodes in community 3 belong to either PRI or PVEM, which formed a coalition during the preceding election in 2015. On the other hand, nodes from MORENA in community 4 were members of PRD before MORENA was formed in 2014. The plot on the right in Fig. 8 also shows a significant overlap between these parties.

Exploring individual memberships reveals that the number of communities a node is assigned to seems associated with its overall popularity in the network. For example, the node with the largest number of community memberships (7 in total) is the representative with the largest degree in the network, while the second largest number of memberships (5 in total) is the president of the Chamber of Deputies in 2016.

6 Discussion

We presented an approach to estimate a regularized basis of the principal subspace of the network adjacency matrix, and showed that its sparsity pattern encodes the community membership information. Varying the amount of regularization controls the sparsity of the node memberships and allows to one to obtain a family of solutions of increasing complexity. These methods show good accuracy in estimating the memberships, and are computationally

very efficient allowing to scale well to large networks. Our present theoretical results are limited to fixed points of the algorithms; establishing theoretical guarantees in more general settings as analyzing conditions for convergence to the fixed point are left for future work.

Spectral inference has been used for multiple tasks on networks: community detection (Lei and Rinaldo, 2015; Le et al., 2017), hypothesis testing (Tang et al., 2017), multiple network dimensionality reduction (Levin et al., 2017) and network classification (Arroyo and Levina, 2020). While the principal eigenspace of the adjacency matrix can provide the information needed for these problems, our results suggest that regularizing the eigenvectors can lead to improved estimation and computation in community detection; exploring the effects of this type of regularization in other network tasks is a promising direction for future work.

Acknowledgements. This research was supported in part by NSF grants DMS-1521551 and DMS-1916222. The authors would like to thank Yuan Zhang for helpful discussions, and Advanced Research Computing at the University of Michigan for computational resources and services.

References

ABBE, E (2017). Community detection and stochastic block models: recent developments. *Journal of Machine Learning Research* **18**, 1–86.

ADAMIC, L A and GLANCE, N (2005). *The political blogosphere and the 2004 US election: divided they blog*. ACM, p. 36–43.

AIROLDI, E M, BLEI, D M, FIENBERG, S E and XING, E P (2009). *Mixed membership stochastic blockmodels*, p. 33–40.

AMINI, A A and WAINWRIGHT, M J (2008). *High-dimensional analysis of semidefinite relaxations for sparse principal components*. IEEE, p. 2454–2458.

AMINI, A A and LEVINA, E (2018). On semidefinite relaxations for the block model. *The Annals of Statistics* **46**, 1, 149–179.

ARROYO, J and LEVINA, E (2020). Simultaneous prediction and community detection for networks with application to neuroimaging. *arXiv*:2002.01645.

BALL, B, KARRER, B and NEWMAN, MEJ (2011). Efficient and principled method for detecting communities in networks. *Physical Review E* **84**, 3, 036103.

BOLLOBÁS, B, JANSON, S and RIORDAN, O (2007). The phase transition in inhomogeneous random graphs. *Random Structures and Algorithms* **31**, 1, 3–122.

BULLMORE, E and SPORNS, O (2009). Complex brain networks: graph theoretical analysis of structural and functional systems. *Nature Reviews Neuroscience* **10**, 3, 186–198.

CAPE, J, TANG, M and PRIEBE, C E (2019). On spectral embedding performance and elucidating network structure in stochastic blockmodel graphs. *Network Science* **7**, 3, 269–291.

CONOVER, M, RATKIEWICZ, J, FRANCISCO, M R, GONÇALVES, B, MENCZER, F and FLAMMINI, A (2011). Political polarization on Twitterx. *ICWSM* **133**, 89–96.

DA FONSECA VIEIRA, V, XAVIER, C R and EVSUKOFF, A G (2020). A comparative study of overlapping community detection methods from the perspective of the structural properties. *Applied Network Science* **5**, 1, 1–42.

GIRVAN, M and NEWMAN, MARK EJ (2002). Community structure in social and biological networks. *Proceedings of the National Academy of Sciences* **99**, 12, 7821–7826.

GOLUB, G H and VAN LOAN, C F (2012). *Matrix computations*, *3*. Johns Hopkins University Press, USA.

GREGORY, S (2010). Finding overlapping communities in networks by label propagation. *New Journal of Physics* **12**, 10, 103018.

HOLLAND, P W, LASKEY, K B and LEINHARDT, S (1983). Stochastic blockmodels: First steps. *Social Networks* **5**, 2, 109–137.

HUANG, K and FU, X (2019). *Detecting overlapping and correlated communities without pure nodes: Identifiability and algorithm*, p. 2859–2868.

JI, P and JIN, J (2016). Coauthorship and citation networks for statisticians. *The Annals of Applied Statistics* **10**, 4, 1779–1812.

JIN, J (2015). Fast community detection by score. *The Annals of Statistics* **43**, 1, 57–89.

JIN, J, KE, Z T and LUO, S (2017). Estimating network memberships by simplex vertex hunting. *arXiv*:1708.07852.

JOHNSTONE, I M and LU, A Y (2009). On consistency and sparsity for principal components analysis in high dimensions. *Journal of the American Statistical Association* **104**, 486, 682–693.

JOLLIFFE, I T, TRENDAFILOV, N T and UDDIN, M (2003). A modified principal component technique based on the lasso. *Journal of Computational and Graphical Statistics* **12**, 3, 531–547.

KARRER, B and NEWMAN, M.EJ (2011). Stochastic blockmodels and community structure in networks. *Physical Review E* **83**, 1, 016107.

LANCICHINETTI, A, FORTUNATO, S and KERTÉSZ, J (2009). Detecting the overlapping and hierarchical community structure in complex networks. *New J. Phys.* **11**, 3, 033015.

LANCICHINETTI, A, RADICCHI, F, RAMASCO, J J and FORTUNATO, S (2011). Finding statistically significant communities in networks. *PLoS ONE 6*, 4.

LATOUCHE, P, BIRMELÉ, E. and AMBROISE, C (2011). Overlapping stochastic block models with application to the french political blogosphere. *The Annals of Applied Statistics*, 309–336.

LE, C M and LEVINA, E (2015). Estimating the number of communities in networks by spectral methods. *arXiv*:1507.00827.

LE, C M, LEVINA, E and VERSHYNIN, R (2017). Concentration and regularization of random graphs. *Random Structures & Algorithms* **51**, 3, 538–561.

LEE, D D and SEUNG, H S (1999). Learning the parts of objects by non-negative matrix factorization. *Nature* **401**, 6755, 788–791.

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

LEVIN, K, ATHREYA, A, TANG, M, LYZINSKI, V and PRIEBE, C E (2017). A central limit theorem for an omnibus embedding of random dot product graphs. arXiv:1705.09355.

LI, T, LEVINA, E and ZHU, J (2020). Network cross-validation by edge sampling. *Biometrika* **107**, 2, 257–276.

LYZINSKI, V, SUSSMAN, D L, TANG, M, ATHREYA, A and PRIEBE, C E (2014). Perfect clustering for stochastic blockmodel graphs via adjacency spectral embedding. *Electronic Journal of Statistics* **8**, 2, 2905–2922.

MA, Z (2013). Sparse principal component analysis and iterative thresholding. *The Annals of Statistics,* **41**, 2, 772–801.

MAO, X, SARKAR, P and CHAKRABARTI, D (2017). *On mixed memberships and symmetric nonnegative matrix factorizations*. PMLR, p. 2324–2333.

MAO, X, SARKAR, P and CHAKRABARTI, D (2018). *Overlapping clustering models, and one (class) svm to bind them all*, p. 2126–2136.

MAO, X, SARKAR, P and CHAKRABARTI, D (2020). Estimating mixed memberships with sharp eigenvector deviations. *Journal of the American Statistical Association*. (just-accepted), 1–24.

MCAULEY, J J and LESKOVEC, J (2012). *Learning to discover social circles in ego networks.*, *2012*, p. 548–56.

NEWMAN, MARK EJ (2006). Finding community structure in networks using the eigenvectors of matrices. *Physical Review E* **74**, 3, 036104.

PORTER, M A, ONNELA, J.-P. and MUCHA, P J (2009). Communities in networks. *Notices of the AMS* **56**, 9, 1082–1097.

POWER, J D, COHEN, A L, NELSON, S M, WIG, G S, BARNES, K A, CHURCH, J A, VOGEL, A C, LAUMANN, T O, MIEZIN, F M and SCHLAGGAR, B L (2011). Functional network organization of the human brain. *Neuron* **72**, 4, 665–678.

PSORAKIS, I, ROBERTS, S, EBDEN, M and SHELDON, B (2011). Overlapping community detection using bayesian non-negative matrix factorization. *Physical Review E* **83**, 6, 066114.

ROHE, K, CHATTERJEE, S and YU, B (2011). Spectral clustering and the high-dimensional stochastic blockmodel. *Ann. Statist.* **39**, 4, 1878–1915.

RUBIN-DELANCHY, P, PRIEBE, C E and TANG, M (2017). Consistency of adjacency spectral embedding for the mixed membership stochastic blockmodel. arXiv:1705.04518.

SCHLITT, T and BRAZMA, A (2007). Current approaches to gene regulatory network modelling. *BMC Bioinformatics* **8**, Suppl 6, S9.

SCHWARZ, G (1978). Estimating the dimension of a model. *The Annals of Statistics* **6**, 2, 461–464.

SCHWARZ, A J, GOZZI, A and BIFONE, A (2008). Community structure and modularity in networks of correlated brain activity. *agnetic Resonance Imaging* **26**, 7, 914–920.

TANG, M, ATHREYA, A, SUSSMAN, D L, LYZINSKI, V, PARK, Y and PRIEBE, C E (2017). A semiparametric two-sample hypothesis testing problem for random graphs. *Journal of Computational and Graphical Statistics* **26**, 2, 344–354.

VU, V Q and LEI, J (2013). Minimax sparse principal subspace estimation in high dimensions. *The Annals of Statistics* **41**, 6, 2905–2947.

WANG, C and BLEI, D (2009). Decoupling sparsity and smoothness in the discrete hierarchical Dirichlet process. *Advances in Neural Information Processing Systems* **22**, 1982–1989.

WANG, YX R and BICKEL, P J (2017). Likelihood-based model selection for stochastic block models. *The Annals of Statistics* **45**, 2, 500–528.

WASSERMAN, S and FAUST, K (1994). *Social network analysis: Methods and applications*, *8*. Cambridge University Press, Cambridge.

WILLIAMSON, S, WANG, C, HELLER, K A and BLEI, D M (2010). *The ibp compound dirichlet process and its application to focused topic modeling*. Omnipress, Madison, p. 1151–1158.

XIE, J, KELLEY, S and SZYMANSKI, B K (2013). Overlapping community detection in networks: The state-of-the-art and comparative study. *ACM Computing Surveys* **45**, 4, 1–35.

YU, Y, WANG, T and SAMWORTH, R J (2015). A useful variant of the Davis–Kahan theorem for statisticians. *Biometrika* **102**, 2, 315–323.

ZACHARY, W W (1977). An information flow model for conflict and fission in small groups. *Journal of Anthropological Research* **33**, 4, 452–473.

ZHANG, Y, LEVINA, E and ZHU, J (2020). Detecting Overlapping Communities in Networks Using Spectral Methods. *SIAM Journal on Mathematics of Data Science* **2**, 2, 265–283.

ZOU, H, HASTIE, T and TIBSHIRANI, R (2006). Sparse principal component analysis. *Journal of Computational and Graphical Statistics* **15**, 2, 265–286.

Appendix

PROOF OF PROPOSITION 1. Because $\mathbf{V}$ and $\widetilde{\mathbf{V}}$ are two bases of the column space of $\mathbf{P}$, and $\text{rank}(\mathbf{P}) = K$, then $\mathbf{P} = \mathbf{V}\mathbf{U}^\top = \widetilde{\mathbf{V}}\widetilde{\mathbf{U}}^\top$ for some full rank matrices $\mathbf{U}, \widetilde{\mathbf{U}} \in \mathbb{R}^{n\times K}$ and therefore

$$\mathbf{V} = \widetilde{\mathbf{V}}(\widetilde{\mathbf{U}}^\top\mathbf{U})(\mathbf{U}^\top\mathbf{U})^{-1}. \tag{A.1}$$

Let $(\widetilde{\mathbf{U}}^\top\mathbf{U})(\mathbf{U}^\top\mathbf{U})^{-1} = \mathbf{\Lambda}$. We will show that $\mathbf{\Lambda} = \mathbf{Q}\mathbf{D}$ for a permutation matrix $\mathbf{Q} \in \{0,1\}^{K\times K}$ and a diagonal matrix $\mathbf{D} \in \mathbb{R}^{K\times K}$, or in other words, this is a generalized permutation matrix.

Let $\boldsymbol{\theta}, \widetilde{\boldsymbol{\theta}} \in \mathbb{R}^n$ and $\mathbf{Z}, \widetilde{\mathbf{Z}} \in \mathbb{R}^{n\times K}$ such that $\boldsymbol{\theta}_i = \left(\sum_{k=1}^K \mathbf{V}_{ik}^2\right)^{1/2}$, $\widetilde{\boldsymbol{\theta}}_i = \left(\sum_{k=1}^K \widetilde{\mathbf{V}}_{ik}^2\right)^{1/2}$, and $\mathbf{Z}_{ik} = \mathbf{V}_{ik}/\boldsymbol{\theta}_i$ if $\boldsymbol{\theta}_i > 0$, and $\mathbf{Z}_{ik} = 0$ otherwise (similarly for $\widetilde{\mathbf{Z}}$). Denote by $\mathcal{S}_1 = (i_1, \ldots, i_K)$ to the vector of row indexes that satisfy $\mathbf{V}_{i_j j} > 0$ and $\mathbf{V}_{i_j j'} = 0$ for $j' \neq j$, and $j = 1, \ldots, j$ (these indexes exist by assumption). In the same way, define $\mathcal{S}_2 = (i'_1, \ldots, i'_K)$ such that $\widetilde{\mathbf{V}}_{i'_j j} > 0$ and $\widetilde{\mathbf{V}}_{i_j j'} = 0$ for $j' \neq j$. $j = 1, \ldots, j$. Denote by $\mathbf{Z}_{\mathcal{S}}$ to the $K \times K$ matrix formed by the rows indexed by $\mathcal{S}$. Therefore

$$\mathbf{Z}_{\mathcal{S}_1} = \mathbf{I}_K = \widetilde{\mathbf{Z}}_{\mathcal{S}_2}.$$

Write $\mathbf{\Theta} = \text{diag}(\boldsymbol{\theta}) \in \mathbb{R}^{n\times n}$ and $\widetilde{\mathbf{\Theta}} = \text{diag}(\widetilde{\boldsymbol{\theta}}) \in \text{real}^{n\times n}$. From Eq. A.1 we have

$$(\mathbf{\Theta}\mathbf{Z})_{\mathcal{S}_2} = \quad (\widetilde{\mathbf{\Theta}}\widetilde{\mathbf{Z}})_{\mathcal{S}_2}\mathbf{\Lambda} = \widetilde{\mathbf{\Theta}}_{\mathcal{S}_2,\mathcal{S}_2}\widetilde{\mathbf{Z}}_{\mathcal{S}_2}\mathbf{\Lambda} = \widetilde{\mathbf{\Theta}}_{\mathcal{S}_2,\mathcal{S}_2}\mathbf{\Lambda}\ ,$$

where $\Theta_{\mathcal{S},\mathcal{S}}$ is the submatrix of Θ formed by the rows and columns indexed by $\mathcal{S}$. Thus,

$$\mathbf{\Lambda} = (\widetilde{\mathbf{\Theta}}^{-1}_{\mathcal{S}_2,\mathcal{S}_2}\mathbf{\Theta}_{\mathcal{S}_2,\mathcal{S}_2})\mathbf{Z}_{\mathcal{S}_2},$$

which implies that $\mathbf{\Lambda}$ is a non-negative matrix. Applying the same to the equation $(\mathbf{\Theta}\mathbf{Z})_{\mathcal{S}_1}\mathbf{\Lambda}^{-1} = (\widetilde{\mathbf{\Theta}}\widetilde{\mathbf{Z}})_{\mathcal{S}_1}$, we have

$$\mathbf{\Lambda}^{-1} = (\mathbf{\Theta}^{-1}_{\mathcal{S}_1,\mathcal{S}_1}\widetilde{\mathbf{\Theta}}_{\mathcal{S}_1,\mathcal{S}_1})\widetilde{\mathbf{Z}}_{\mathcal{S}_1}.$$

Hence, both $\mathbf{\Lambda}$ and $\mathbf{\Lambda}^{-1}$ are non-negative matrices, which implies that $\mathbf{\Lambda}$ is a positive generalized permutation matrix, so $\mathbf{\Lambda} = \mathbf{QD}$ for some permutation matrix $\mathbf{Q}$ and a diagonal matrix $\mathbf{D}$ with $\mathrm{diag}(\mathbf{D}) > 0$.

Proof of Proposition 2. Let $\boldsymbol{\theta} \in \mathbb{R}^n$ be a vector such that $\boldsymbol{\theta}_i^2 = \sum_{k=1}^K \boldsymbol{V}_{ik}^2$, and define $\mathbf{Z} \in \mathbb{R}^{n \times K}$ such that $\mathbf{Z}_{ik} = \frac{1}{\theta_i}\mathbf{V}_{ik}$, for each $i \in [n], k \in [K]$. Let $\mathbf{B} = (\mathbf{V}^\top \mathbf{V})^{-1}\mathbf{V}^T\mathbf{U}$. To show that $\mathbf{B}$ is symmetric, observe that $\mathbf{VU}^\top = \mathbf{P} = \mathbf{P}^\top = \mathbf{UV}^\top$. Multiplying both sides by $\mathbf{V}$ and $\mathbf{V}^\top$,

$$\mathbf{V}^\top\mathbf{VU}^\top\mathbf{V} = \mathbf{V}^\top\mathbf{UV}^\top\mathbf{V},$$

and observing that $(\mathbf{V}^\top\mathbf{V})^{-1}$ exists since $\mathbf{V}$ is full rank, we have

$$\mathbf{U}^\top\mathbf{V}(\mathbf{V}^\top\mathbf{V})^{-1} = (\mathbf{V}^\top\mathbf{V})^{-1}\mathbf{V}^\top\mathbf{U},$$

which implies that $\mathbf{B}^\top = \mathbf{B}$. To obtain the equivalent representation for $\mathbf{P}$, form a diagonal matrix $\mathbf{\Theta} = \mathrm{diag}(\boldsymbol{\theta})$. Then $\mathbf{\Theta Z} = \mathbf{V}$, and

$$\mathbf{\Theta ZBZ}^\top\mathbf{\Theta} = \mathbf{V}[(\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T\mathbf{U}]\mathbf{V}^\top = \mathbf{V}(\mathbf{V}^T\mathbf{V})^{-1}\mathbf{V}^T\mathbf{VU}^\top = \mathbf{VU}^\top = \mathbf{P}.$$

Finally, under the conditions of Proposition 1, $\mathbf{V}$ uniquely determines the pattern of zeros of any non-negative eigenbasis of $\mathbf{P}$, and therefore $\mathrm{supp}(\mathbf{V}) = \mathrm{supp}(\mathbf{\Theta ZQ}) = \mathrm{supp}(\mathbf{ZQ})$ for some permutation $\mathbf{Q}$.

Proof of Proposition 3. Suppose that $\mathbf{P} = \mathbf{VU}^\top$ for some non-negative matrix $\mathbf{V}$ that satisfies the assumptions of Proposition 1. Let $D \in \mathrm{real}^K$ such that $D_i = \|\mathbf{V}_{\cdot k}\|_2$ and $\mathbf{D} = \mathrm{diag}(D)$. Then $\mathbf{P} = \widetilde{\mathbf{V}}\mathbf{DU}^\top$. Let $\mathbf{V}^{(0)} = \widetilde{\mathbf{V}}$ be the initial value of Algorithm 1. Then, observe that

$$\mathbf{T}^{(1)} = \mathbf{P}\widetilde{\mathbf{V}} = \widetilde{\mathbf{V}}\mathbf{DU}^T\widetilde{\mathbf{V}},$$

$$\begin{aligned}
\widetilde{\mathbf{T}}^{(1)} &= \mathbf{T}^{(1)}\left[\widetilde{\mathbf{V}}^\top\mathbf{T}^{(1)}\right]^{-1}(\widetilde{\mathbf{V}}^T\widetilde{\mathbf{V}}) \\
&= \widetilde{\mathbf{V}}\mathbf{D}(\mathbf{U}^T\widetilde{\mathbf{V}})\left(\mathbf{U}^T\widetilde{\mathbf{V}}\right)^{-1}\mathbf{D}^{-1}(\widetilde{\mathbf{V}}^\top\widetilde{\mathbf{V}})^{-1}(\widetilde{\mathbf{V}}^\top\widetilde{\mathbf{V}}) \\
&= \widetilde{\mathbf{V}}.
\end{aligned}$$

Suppose that $\lambda \in [0, v^*)$. Then, $\lambda \max_{j\in[K]} |\widetilde{\mathbf{V}}| < \widetilde{\mathbf{V}}_{ik}$ for all $i \in [n], k \in [K]$ such that $\mathbf{V}_{ik} > 0$, and hence $\mathbf{U}^{(1)} = \mathcal{S}(\widetilde{\mathbf{V}}, \lambda) = \widetilde{\mathbf{V}}$. Finally, since $\|\widetilde{\mathbf{V}}_{\cdot,k}\|_2 = 1$ for all $k \in [K]$, then $\mathbf{V}^{(1)} = \widetilde{\mathbf{V}}$.

PROOF OF THEOREM 1. The proof consists of a one-step fixed point analysis of Algorithm 2. We will show that if $\boldsymbol{Z}^{(t)} = \boldsymbol{Z}$, then $\boldsymbol{Z}^{(t+1)} = \boldsymbol{Z}$ with high probability. Let $\mathbf{T} = \mathbf{T}^{(t+1)} = \mathbf{AZ}$ be value after the multiplication step. Define $\mathbf{C} \in \mathbb{R}^{K \times K}$ to be the diagonal matrix with community sizes on the diagonal, $\boldsymbol{C}_{kk} = n_k = \|\mathbf{Z}_{\cdot,k}\|_1$. Then $\widetilde{\mathbf{T}} = \widetilde{\mathbf{T}}^{(t+1)} = \mathbf{TC}^{-1}$. In order for the threshold to set the correct set of entries to zero, a sufficient condition is that in each row i the largest element of $\widetilde{\boldsymbol{T}}_{i,\cdot}$ corresponds to the correct community. Define $\mathcal{C}_k \subset [n]$ as the node subset corresponding to community k. Then,

$$\widetilde{\mathbf{T}}_{ik} = \frac{1}{n_k}\mathbf{A}_{i,\cdot}\boldsymbol{Z}_{\cdot,k} = \frac{1}{n_k}\sum_{j \in \mathcal{C}_k} \mathbf{A}_{ij}.$$

Therefore $\widetilde{\mathbf{T}}_{ik}$ is a sum of independent and identically distributed Bernoulli random variables. Moreover, for each k_1 and k_2 in $[K]$, $\widetilde{\mathbf{T}}_{ik_1}$ and $\widetilde{\mathbf{T}}_{ik_2}$ are independent of each other.

Given a value of $\lambda \in (0, 1)$, let

$$\mathcal{E}_i(\lambda) = \{\lambda|\widetilde{\boldsymbol{T}}_{ik_i}| > |\widetilde{\boldsymbol{T}}_{ik_j}|, i \in \mathcal{C}_{k_i} \forall k_j \neq k_i\}$$

be the event that the largest entry of $\widetilde{\mathbf{T}}_{i\cdot}$ corresponds to k_i, that is, the entry corresponding to the community of node i, and all the other indexes in that row are smaller in magnitude than $\lambda|\widetilde{\mathbf{T}}_{ik_i}|$. Let $\mathbf{U} = \mathbf{U}^{(t+1)} = \mathcal{S}(\widetilde{\mathbf{T}}^{(t+1)}, \lambda)$ be the matrix obtained after the thresholding step. Under the event $\mathcal{E}(\lambda) = \bigcap_{i=1}^n \mathcal{E}_i(\lambda)$, we have that $\|\mathbf{U}_{i,\cdot}\|_\infty = \mathbf{U}_{ik_i}$ for each $i \in [n]$, and hence

$$\mathbf{U}_{ik} = \begin{cases} \mathbf{U}_{ik_i} & \text{if } k = k_i, \\ 0 & \text{otherwise.} \end{cases}$$

Therefore, under the event $\mathcal{E}(\lambda)$, the thresholding step recovers the correct support, so $\mathbf{Z}^{(t+1)} = \mathbf{Z}$.

Now we verify that under the conditions of Theorem 3.6, the event $\mathcal{E}(\lambda)$ happens with high probability. By a union bound,

$$\mathbb{P}(\mathcal{E}(\lambda)) \geq 1 - \sum_{i=1}^n \mathbb{P}(\mathcal{E}_i(\lambda)^C) \geq 1 - \sum_{i=1}^n \sum_{j \neq k_i} \mathbb{P}(\widetilde{\mathbf{T}}_{ij} > \lambda \widetilde{\mathbf{T}}_{ik_i}). \tag{A.2}$$

For $j \neq k_i$, $\widetilde{\mathbf{T}}_{ij} - \lambda\widetilde{\mathbf{T}}_{ik_i}$ is a sum of independent random variables with expectation

$$\begin{aligned}\mathbb{E}\left[\widetilde{\mathbf{T}}_{ij} - \lambda\widetilde{\mathbf{T}}_{ik_i}\right] &= \frac{1}{n_j}\sum_{j\in\mathcal{C}_j}\mathbb{E}[\boldsymbol{A}_{ij}] - \frac{\lambda}{n_{k_i}}\sum_{j\in\mathcal{C}_{k_i}}\mathbb{E}[\mathbf{A}_{ij}] \\ &= q - \lambda\frac{n_{k_i}-1}{n_{k_i}}p. \end{aligned} \tag{A.3}$$

By Hoeffding's inequality, we have that for any $\tau \in \mathbb{R}$,

$$\begin{aligned}\mathbb{P}\left(\widehat{\mathbf{T}}_{ij} - \lambda\widehat{\mathbf{T}}_{ik_i} \geq \tau + \mathbb{E}\left[\widehat{\mathbf{T}}_{ij} - \lambda\widehat{\mathbf{T}}_{ik_i}\right]\right) &\leq 2\exp\left(\frac{-2\tau^2}{\frac{1}{n_j}+\frac{\lambda^2}{n_{k_i}}}\right) \\ &\leq 2\exp\left(-\frac{2n_{\min}\tau^2}{1+\lambda^2}\right) \\ &\leq 2\exp\left(-n_{\min}\tau^2\right),\end{aligned}$$

where $n_{\min} = \min_{k\in[K]} n_k$. Setting

$$\tau = -\mathbb{E}\left[\widehat{\mathbf{T}}_{ij} - \lambda\widehat{\mathbf{T}}_{ik_i}\right] \geq \lambda^* p - q - \frac{1}{n_{k_i}}p$$

and using Eq. A.3 and 3.6, we obtain that for n sufficiently large,

$$\begin{aligned}\mathbb{P}\left(\widehat{\mathbf{T}}_{ij} > \lambda\widehat{\mathbf{T}}_{ik_i}\right) &\leq 2\exp\left(-n_{\min}\left(c_1\sqrt{\frac{\log(Kn)}{\min_k n_k}} - \frac{p}{n_k}\right)^2\right) \\ &\leq 2\exp\left(-n_{\min}\left((c_1-1)\frac{\log(Kn)}{n_{\min}}\right)\right) = \frac{2}{(Kn)^{c_1-1}}.\end{aligned}$$

Combining with the bound (A.2), the probability of event $\mathcal{E}(\lambda)$ (which implies that $\mathbf{Z}^{(t+1)} = \mathbf{Z}$) is bounded from below as

$$\begin{aligned}\mathbb{P}(\mathcal{E}(\lambda)) &\geq 1 - n(K-1)\min_{i\in[n],k\in[K]}\mathbb{P}\left(\widehat{\mathbf{T}}_{ij} > \lambda\widehat{\mathbf{T}}_{ik_i}\right) \\ &\geq 1 - \frac{2(K-1)n}{(Kn)^{c_1-1}} \\ &\geq 1 - \frac{2}{Kn^{(c_1-2)}}.\end{aligned}$$

Therefore, with high probability $\boldsymbol{Z}$ is a fixed point of the Algorithm 2 for any $\lambda \in (\lambda^*, 1)$.

Proof of Proposition 4. Observe that

$$\begin{aligned}\|\mathbf{A} - \widehat{\mathbf{V}}\mathbf{B}\widehat{\mathbf{V}}^\top\|_F^2 &= \mathrm{Tr}(\mathbf{A}^\top\mathbf{A}) - 2\mathrm{Tr}(\widehat{\mathbf{V}}^\top\mathbf{A}^\top\widehat{\mathbf{V}}\mathbf{B}) + \mathrm{Tr}(\mathbf{B}^\top\widehat{\mathbf{V}}^\top\widehat{\mathbf{V}}\mathbf{B}\widehat{\mathbf{V}}^\top\widehat{\mathbf{V}}) \\ &= \|\mathbf{B} - (\widehat{\mathbf{V}}^\top\widehat{\mathbf{V}})^{-1}\widehat{\mathbf{V}}^\top\mathbf{A}\widehat{\mathbf{V}}(\widehat{\mathbf{V}}^\top\widehat{\mathbf{V}})^{-1}\|_F^2 + C,\end{aligned}$$

where C is a constant that does not depend on $\mathbf{B}$. Therefore $\widehat{\mathbf{B}}$

$$\begin{aligned}\widehat{\mathbf{P}} &= \underset{\boldsymbol{B}\in\mathbb{R}^{K\times K}}{\arg\min}\|\mathbf{A} - \widehat{\mathbf{V}}\mathbf{B}\widehat{\mathbf{V}}^\top\|_F^2 \\ &= \widehat{\mathbf{V}}(\widehat{\mathbf{V}}^\top\widehat{\mathbf{V}})^{-1}\widehat{\mathbf{V}}^\top\mathbf{A}\widehat{\mathbf{V}}(\widehat{\mathbf{V}}^\top\widehat{\mathbf{V}})^{-1}\widehat{\mathbf{V}}^\top.\end{aligned}$$

Suppose that $\widehat{\boldsymbol{V}} = \widehat{\boldsymbol{Q}}\widehat{\boldsymbol{R}}$ for some matrix $\boldsymbol{Q}$ with orthonormal columns of size $n \times K$. Then, $\widehat{\boldsymbol{R}}$ is a full rank matrix, and therefore

$$(\widehat{\mathbf{V}}^\top\widehat{\mathbf{V}})^{-1} = \widehat{\mathbf{R}}^{-1}(\widehat{\mathbf{Q}}^\top\widehat{\mathbf{Q}})^{-1}(\widehat{\mathbf{R}^\top})^{-1}.$$

Using this equation, we obtain the desired result.

Publisher's Note. Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Jesús Arroyo
Department of Mathematics, University of Maryland, College Park, MD, USA
E-mail: jesus.arroyo@jhu.edu

Jesús Arroyo
Department of Applied Mathematics and Statistics, Johns Hopkins University, Baltimore, USA

Elizaveta Levina
Department of Statistics, University of Michigan, Ann Arbor, MI, USA
E-mail: elevina@umich.edu

Paper received: 5 July 2020; accepted 15 February 2021.