The Annals of Statistics
2016, Vol. 44, No. 1, 373–400
DOI: 10.1214/15-AOS1360

OPTIMIZATION VIA LOW-RANK APPROXIMATION FOR COMMUNITY DETECTION IN NETWORKS

BY CAN M. LE, ELIZAVETA LEVINA[1] AND ROMAN VERSHYNIN[2]

University of Michigan

Community detection is one of the fundamental problems of network analysis, for which a number of methods have been proposed. Most model-based or criteria-based methods have to solve an optimization problem over a discrete set of labels to find communities, which is computationally infeasible. Some fast spectral algorithms have been proposed for specific methods or models, but only on a case-by-case basis. Here, we propose a general approach for maximizing a function of a network adjacency matrix over discrete labels by projecting the set of labels onto a subspace approximating the leading eigenvectors of the expected adjacency matrix. This projection onto a low-dimensional space makes the feasible set of labels much smaller and the optimization problem much easier. We prove a general result about this method and show how to apply it to several previously proposed community detection criteria, establishing its consistency for label estimation in each case and demonstrating the fundamental connection between spectral properties of the network and various model-based approaches to community detection. Simulations and applications to real-world data are included to demonstrate our method performs well for multiple problems over a wide range of parameters.

1. Introduction. Networks are studied in a wide range of fields, including social psychology, sociology, physics, computer science, probability, and statistics. One of the fundamental problems in network analysis, and one of the most studied, is detecting network community structure. Community detection is the problem of inferring the latent label vector $\mathbf{c} \in \{1, \dots, K\}^n$ for the n nodes from the observed $n \times n$ adjacency matrix A, specified by $A_{ij} = 1$ if there is an edge from i to j, and $A_{ij} = 0$ otherwise. While the problem of choosing the number of communities K is important, in this paper we assume K is given, as does most of the existing literature. We focus on the undirected network case, where the matrix A is symmetric. Roughly speaking, the large recent literature on community detection in this scenario has followed one of two tracks: fitting probabilistic models for the adjacency

Received May 2015; revised July 2015.

[1]Supported in part by NSF Grants DMS-01-106772 and DMS-11-59005.

[2]Supported in part by NSF Grants DMS-11-61372, 10-01829, 12-65782 and USAF Grant FA9550-14-1-0009.

MSC2010 subject classifications. Primary 62H30; secondary 62H25, 62G20.

Key words and phrases. Community detection, spectral clustering, stochastic block model, social networks.

matrix A, or optimizing global criteria derived from other considerations over label assignments $\mathbf{c}$, often via spectral approximations.

One of the simplest and most popular probabilistic models for fitting community structure is the stochastic block model (SBM) [17]. Under the SBM, the label vector $\mathbf{c}$ is assumed to be drawn from a multinomial distribution with parameter $\boldsymbol{\pi} = \{\pi_1, \ldots, \pi_K\}$, where $0 \le \pi_k \le 1$ and $\sum_{k=1}^{K} \pi_k = 1$. Edges are then formed independently between every pair of nodes (i, j) with probability $P_{c_i c_j}$, and the $K \times K$ matrix $P = [P_{kl}]$ controls the probability of edges within and between communities. Thus, the labels are the only node information affecting edges between nodes, and all the nodes within the same community are stochastically equivalent to each other. This rules out the commonly encountered "hub" nodes, which are nodes of unusually high degrees that are connected to many members of their own community, or simply to many nodes across the network. To address this limitation, a relaxation that allows for arbitrary expected node degrees within communities was proposed by [20]: the degree-corrected stochastic block model (DCSBM) has $P(A_{ij} = 1) = \theta_i \theta_j P_{c_i c_j}$, where θ_i's are "degree parameters" satisfying some identifiability constraints. In the "null" case of $K = 1$, both the block model and the degree corrected block model correspond to well-studied random graph models, the Erdős–Rényi graph [10] and the configuration model [8], respectively. Many other network models have been proposed to capture the community structure, for example, the latent space model [16] and the latent position cluster model [15]. There has also been work on extensions of the SBM which allow nodes to belong to more than one community [2, 4, 45]. For a more complete review of network models, see [13].

Fitting models such as the stochastic block model typically involves maximizing a likelihood function over all possible label assignments, which is in principle NP-hard. MCMC-type and variational methods have been proposed, see for example [26, 36, 42], as well as maximizing profile likelihoods by some type of greedy label-switching algorithms. The profile likelihood was derived for the SBM by [6] and for the DCSBM by [20], but the label-switching greedy search algorithms only scale up to a few thousand nodes. Amini et al. [3] proposed a much faster pseudo-likelihood algorithm for fitting both these models, which is based on compressing A into block sums and modeling them as a Poisson mixture. Another fast algorithm for the block model based on belief propagation has been proposed by [9]. Both these algorithms rely heavily on the particular form of the SBM likelihood and are not easily generalizable.

The SBM likelihood is just one example of a function that can be optimized over all possible node labels in order to perform community detection. Many other functions have been proposed for this purpose, often not tied to a generative network model. One of the best known such functions is modularity [33, 34]. The key idea of modularity is to compare the observed network to a null model that has no community structure. To define this, let e be an n-dimensional label vector,

$n_k(e) = \sum_{i=1}^{n} I\{e_i = k\}$ the number of nodes in community k,

$$O_{kl}(e) = \sum_{i,j=1}^{n} A_{ij} I\{e_i = k, e_j = l\} \tag{1.1}$$

the number of edges between communities k and l, $k \neq l$ and $O_k = \sum_{l=1}^{K} O_{kl}$ the sum of node degrees in community k. Let $d_i = \sum_{j=1}^{n} A_{ij}$ be the degree of node i, and $m = \sum_{i=1}^{n} d_i$ be (twice) the total number of edges in the graph. The Newman–Girvan modularity is derived by comparing the observed number of edges within communities to the number that would be expected under the Chung–Lu model [8] for the entire graph, and can be written in the form

$$Q_{\mathrm{NG}}(e) = \frac{1}{2m} \sum_{k} \left(O_{kk} - \frac{O_k^2}{m} \right). \tag{1.2}$$

The quantities O_{kl} and O_k turn out to be the key component of many community detection criteria. The profile likelihoods of the SBM and DCSBM discussed above can be expressed as

$$Q_{\mathrm{BM}}(e) = \sum_{k,l=1}^{K} O_{kl} \log \frac{O_{kl}}{n_k n_l}, \tag{1.3}$$

$$Q_{\mathrm{DC}}(e) = \sum_{k,l=1}^{K} O_{kl} \log \frac{O_{kl}}{O_k O_l}. \tag{1.4}$$

Another example is the extraction criterion [46] to extract one community at a time, allowing for arbitrary structure in the remainder of the network. The main idea is to recognize that some nodes may not belong to any community, and the strength of a community should depend on ties between its members and ties to the outside world, but not on ties between nonmembers. This criterion is therefore not symmetric with respect to communities, unlike the criteria previously discussed, and has the form (using slightly different notation due to lack of symmetry),

$$Q_{\mathrm{EX}}(V) = |V||V^c| \left(\frac{O(V)}{|V|^2} - \frac{B(V)}{|V||V^c|} \right), \tag{1.5}$$

where V is the set of nodes in the community to be extracted, V^c is the complement of V, $O(V) = \sum_{i,j \in V} A_{ij}$, $B(V) = \sum_{i \in V, j \in V^c} A_{ij}$. The only known method for optimizing this criterion is through greedy label switching, such as the tabu search algorithm [12].

For all these methods, finding the exact solution requires optimizing a function of the adjacency matrix A over all K^n possible label vectors, which is an infeasible optimization problem. In another line of work, spectral decompositions have been used in various ways to obtain approximate solutions that are much faster to compute. One such algorithm is spectral clustering (see, e.g., [35]), a generic

clustering method which became popular for community detection. In this context, the method has been analyzed by [7, 23, 39, 40], among others, while [18] proposed a spectral method specifically for the DCSBM. In spectral clustering, typically one first computes the normalized Laplacian matrix $L = D^{-1/2}AD^{-1/2}$, where D is a diagonal matrix with diagonal entries being node degrees d_i, though other normalizations and no normalization at all are also possible (see [41] for an analysis of why normalization is beneficial). Then the K eigenvectors of the Laplacian corresponding to the first K largest eigenvalues are computed, and their rows clustered using K-means into K clusters corresponding to different labels. It has been shown that spectral clustering performs better with further regularization, namely if a small constant is added either to D [7, 38] or to A [3, 19, 21].

The contribution of our paper is a new general method of optimizing a general function $f(A, e)$ (satisfying some conditions) over labels e. We start by projecting the entire feasible set of labels onto a low-dimensional subspace spanned by vectors approximating the leading eigenvectors of EA. Projecting the feasible set of labels onto a low-dimensional space reduces the number of possible solutions (extreme points) from exponential to polynomial, and in particular from $O(2^n)$ to $O(n)$ for the case of two communities, thus making the optimization problem much easier. This approach is distinct from spectral clustering since one can specify any objective function f to be optimized (as long as it satisfies some fairly general conditions), and thus applicable to a wide range of network problems. It is also distinct from initializing a search for the maximum of a general function with the spectral clustering solution, since even with a good initialization the feasible space is still extremely large and it is not clear how to update labels effectively.

We show how our method can be applied to maximize the likelihoods of the stochastic block model and its degree-corrected version, Newman–Girvan modularity and community extraction, which all solve different network problems. While spectral approximations to some specific criteria that can otherwise be only maximized by a search over labels have been obtained on a case-by-case basis [32, 34, 39], ours is, to the best of our knowledge, the first general method that would apply to any function of the adjacency matrix. In this paper, we mainly focus on the case of two communities ($K = 2$). For methods that are run recursively, such as modularity and community extraction, this is not a restriction. For the stochastic block model, the case $K = 2$ is of special interest and has received a lot of attention in the probability literature (see [30] for recent advances). An extension to the general case of $K > 2$ is briefly discussed in Section 2.3.

The rest of the paper is organized as follows. In Section 2, we set up notation and describe our general approach to solving a class of optimization problems over label assignments via projection onto a low-dimensional subspace. In Section 3, we show how the general method can be applied to several community detection criteria. Section 4 compares numerical performance of different methods. The proofs are given in the Appendix.

2. A general method for optimization via low-rank approximation. To start with, consider the problem of detection $K = 2$ communities. Many community detection methods rely on maximizing an objective function $f(A, e) \equiv f_A(e)$ over the set of node labels e, which can take values in, say, $\{-1, 1\}$. Since A can be thought of as a noisy realization of $\mathbb{E}[A]$, the "ideal" solution corresponds to maximizing $f_{\mathbb{E}[A]}(e)$ instead of maximizing $f_A(e)$. For a natural class of functions f described below, $f_{\mathbb{E}[A]}(e)$ is essentially a function over the set of projections of labels e onto the subspace spanned by eigenvectors of $\mathbb{E}[A]$ and possibly some other constant vectors. In many cases $\mathbb{E}[A]$ is a low-rank matrix, which makes $f_{\mathbb{E}[A]}(e)$ a function of only a few variables. It is then much easier to investigate the behavior of $f_{\mathbb{E}[A]}(e)$, which typically achieves its maximum on the set of extreme points of the convex hull generated by the projection of the label set e. Further, most of the 2^n possible label assignments e become interior points after the projection, and in fact the number of extreme points is at most polynomial in n (see Remark 2.2 below); in particular, when projecting onto a two-dimensional subspace, the number of extreme points is of order $O(n)$. Therefore, we can find the maximum simply by performing an exhaustive search over the labels corresponding to the extreme points. Section 3.5 provides an alternative method to the exhaustive search, which is faster but approximate.

In reality, we do not know $\mathbb{E}[A]$, so we need to approximate its columns space using the data A instead. Let U_A be an $m \times n$ matrix computed from A such that the row space of U_A approximates the column space of $\mathbb{E}[A]$ (the choice of $m \times n$ rather than $n \times m$ is for notational convenience that will become apparent below). Existing work on spectral clustering gives us multiple option for how to compute this matrix, for example, using the eigenvectors of A itself, of its Laplacian, or of their various regularizations; see Section 2.1 for further discussion of this issue. The algorithm works as follows:

1. Compute the approximation U_A from A.
2. Find the labels e associated with the extreme points of the projection $U_A[-1, 1]^n$.
3. Find the maximum of $f_A(e)$ by performing an exhaustive search over the set of labels found in step 2.

Note that the first step of replacing eigenvectors of $\mathbb{E}[A]$ with certain vectors computed from A is very similar to spectral clustering. Like in spectral clustering, the output of the algorithm does not change if we replace U_A with $U_A R$ for any orthogonal matrix R. However, this is where the similarity ends, because instead of following the dimension reduction by an ad-hoc clustering algorithm like K-means, we maximize the original objective function. The problem is made feasible by reducing the set of labels over which to maximize, to a particular subset found by taking into account the specific behavior of $f_{\mathbb{E}[A]}(e)$ and $f_A(e)$.

While our goal in the context of community detection is to compare $f_A(e)$ to $f_{\mathbb{E}[A]}(e)$, the results and the algorithm in this section apply in a general setting

where A may be any deterministic symmetric matrix. To emphasize this generality, we write all the results in this section for a generic matrix A and a generic low-rank matrix B, even though we will later apply them to the adjacency matrix A and $B = \mathbb{E}[A]$.

Let A and B be $n \times n$ symmetric matrices with entries bounded by an absolute constant, and assume B has rank $m \ll n$. Assume that $f_A(e)$ has the general form

$$f_A(e) = \sum_{j=1}^{\kappa} g_j\big(h_{A,j}(e)\big), \tag{2.1}$$

where g_j are scalar functions on $\mathbb{R}$ and $h_{A,j}(e)$ are quadratic forms of A and e, namely

$$h_{A,j}(e) = (e + s_{j1})^T A(e + s_{j2}). \tag{2.2}$$

Here, κ is a fixed number, s_{j1} and s_{j2} are constant vectors in $\{-1, 1\}^n$. Note that by (3.1), the number of edges between communities has the form (2.2), and by (3.2), the log-likelihood of the degree-corrected block model Q_{DC} is a special case of (2.1) with $g_j(x) = \pm x \log x$, $x > 0$. We similarly define f_B and $h_{B,j}$, by replacing A with B in (2.1) and (2.2). By allowing e to take values on the cube $[-1, 1]^n$, we can treat h and f as functions over $[-1, 1]^n$.

Let U_B be the $m \times n$ matrix whose rows are the m leading eigenvectors of B. For any $e \in [-1, 1]^n$, $U_A e$ and $U_B e$ are the coordinates of the projections of e onto the row spaces of U_A and U_B, respectively. Since $h_{B,j}$ are quadratic forms of B and e and B is of rank m, $h_{B,j}$'s depend on e through $U_B e$ only and, therefore, f_B also depends on e only through $U_B e$. In a slight abuse of notation, we also use $h_{B,j}$ and f_B to denote the corresponding induced functions on $U_B[-1, 1]^n$.

Let $\mathcal{E}_A$ and $\mathcal{E}_B$ denote the subsets of labels $e \in \{-1, 1\}^n$ corresponding to the sets of extreme points of $U_A[-1, 1]^n$ and $U_B[-1, 1]^n$, respectively. The output of our algorithm is

$$e^* = \operatorname{argmax}\big\{ f_A(e), e \in \mathcal{E}_A \big\}. \tag{2.3}$$

Our goal is to get a bound on the difference between the maxima of f_A and f_B that can be expressed through some measure of difference between A and B themselves. In order to do this, we make the following assumptions:

(1) Functions g_j are continuously differentiable and there exists $M_1 > 0$ such that $|g_j'(t)| \le M_1 \log(t + 2)$ for $t \ge 0$.
(2) Function f_B is convex on $U_B[-1, 1]^n$.

Assumption (1) essentially means that Lipschitz constants of g_j do not grow faster than $\log(t + 2)$. The convexity of f_B in assumption (2) ensures that f_B achieves its maximum on $U_B \mathcal{E}_B$. In some cases (see Section 3), the convexity of f_B can be replaced with a weaker condition, namely the convexity along a certain direction.

Let $c \in \{-1, 1\}^n$ be the maximizer of f_B over the set of label vectors $\{-1, 1\}^n$. As a function on $U_B[-1, 1]^n$, f_B achieves its maximum at $U_B(c)$, which is an extreme point of $U_B[-1, 1]^n$ by assumption (2). Lemma 1 provides a upper bound for $f_A(c) - f_A(e^*)$.

Throughout the paper, we write $\|\cdot\|$ for the l_2 norm (i.e., Euclidean norm on vectors and the spectral norm on matrices), and $\|\cdot\|_F$ for the Frobenius norm on matrices. Note that for label vectors $e, c \in \{-1, 1\}^n$, $\|e - c\|^2$ is four times the number of nodes on which e and c differ.

LEMMA 1. *If assumptions* (1) *and* (2) *hold, then there exists a constant* $M_2 > 0$ *such that*

$$f_T(c) - f_T(e^*) \le M_2 n \log(n)\big(\|B\| \cdot \|U_A - U_B\| + \|A - B\|\big), \tag{2.4}$$

where T *is either* A *or* B.

The proof of Lemma 1 is given in Appendix A. To get a bound on $\|c - e^*\|$, we need further assumptions on B and f_B.

(3) There exists $M_3 > 0$ such that for any $e \in \{-1, 1\}^n$,

$$\|c - e\|^2 \le M_3\sqrt{n}\|U_B(c) - U_B(e)\|.$$

(4) There exists $M_4 > 0$ such that for any $x \in U_B[-1, 1]^n$

$$\frac{f_B(U_B(c)) - f_B(x)}{\|U_B(c) - x\|} \ge \frac{\max f_B - \min f_B}{M_4\sqrt{n}}.$$

Assumption (3) rules out the existence of multiple label vectors with the same projection $U_B(c)$. Assumption (4) implies that the slope of the line connecting two points on the graph of f_B at $U_B(c)$ and at any $x \in U_B[-1, 1]^n$ is bounded from below. Thus, if $f_B(x)$ is close to $f_B(U_B(c))$ then x is also close to $U_B(c)$. These assumptions are satisfied for all functions considered in Section 3.

THEOREM 1. *If assumptions* (1)–(4) *hold, then there exists a constant* M_5 *such that*

$$\frac{1}{n}\|e^* - c\|^2 \le \frac{M_5 n \log n(\|B\| \cdot \|U_A - U_B\| + \|A - B\|)}{\max f_B - \min f_B}.$$

Theorem 1 follows directly from Lemma 1 and assumptions (3) and (4). When A is a random matrix, $B = \mathbb{E}[A]$, and U_A contains the leading eigenvectors of A, a standard bound on $\|A - B\|$ can be applied (see Lemma 4), which in turn yields a bound on $\|U_A - U_B\|$ by the Davis–Kahan theorem. Under certain conditions, the upper bound in Theorem 1 is of order $o(n)$ (see Section 3), which shows consistency of e^* as an estimator of c (i.e., the fraction of mislabeled nodes goes to 0 as $n \to \infty$).

2.1. *The choice of low rank approximation.* An important step of our method is replacing the "population" space U_B with the "data" approximation U_A. As a motivating example, consider the case of the SBM, with A the network adjacency matrix and $B = \mathbb{E}[A]$. When the network is relatively dense, eigenvectors of A are good estimates of the eigenvectors of $B = \mathbb{E}[A]$ (see [23, 37] for recent improved error bounds). Thus, U_A can just be taken to be the leading eigenvectors of A. However, when the network is sparse, this is not necessarily the best choice, since the leading eigenvectors of A tend to localize around high degree nodes, while leading eigenvectors of the Laplacian of A tend to localize around small connected components [7, 21, 28, 38]. This can be avoided by regularizing the Laplacian in some form; we follow the algorithm of [3]; see also [19, 21] for theoretical analysis. This works for both dense and sparse networks.

The regularization works as follows. We first add a small constant τ to each entry of A, and then approximate U_B through the Laplacian of $A + \tau \mathbf{1}\mathbf{1}^T$ as follows. Let D_τ be the diagonal matrix whose diagonal entries are sums of entries of columns of $A + \tau \mathbf{1}\mathbf{1}^T$, $L_\tau = D_\tau^{-1/2}(A + \tau \mathbf{1}\mathbf{1}^T)D_\tau^{-1/2}$, and u_i be leading eigenvectors of L_τ, $1 \le i \le K$. Since $A + \tau \mathbf{1}\mathbf{1}^T = D_\tau^{1/2} L_\tau D_\tau^{1/2}$, we set the appoximation U_A the be the basis of the span of $\{D^{1/2}u_i : 1 \le i \le K\}$. Following [3], we set $\tau = \varepsilon(\lambda_n/n)$, where λ_n is the node expected degree of the network and $\varepsilon \in (0, 1)$ is a constant which has little impact on the performance [3].

2.2. *Computational complexity.* Since we propose an exhaustive search over the projected set of extreme points, the computational feasibility of this is a concern. A projection of the unit cube $U_A[-1, 1]^n$ is the Minkowski sum of n segments in $\mathbb{R}^m$, which, by [14], implies that it has $O(n^{m-1})$ vertices of $U_A[-1, 1]^n$ and they can be found in $O(n^m)$ arithmetic operations. When $m = 2$, which is the primary focus of our paper, there exists an algorithm that can find the vertices of $U_A[-1, 1]^n$ in $O(n \log n)$ arithmetic operations [14]. Informally, the algorithm first sorts the angles between the x-axis and column vectors of U_A and $-U_A$. It then starts at a vertex of $U_A[-1, 1]^n$ with the smallest y-coordinate, and based on the order of the angles, finds neighbor vertices of $U_A[-1, 1]^n$ in a counterclockwise order. If the angles are distinct (which occurs with high probability), moving from one vertex to the next causes exactly one entry of the corresponding label vector to change the sign, and therefore the values of $h_{A,j}(e)$ in (2.2) can be updated efficiently. In particular, if A is the adjacency matrix of a network with average degree λ_n, then on average, each update takes $O(\lambda_n)$ arithmetic operations, and given U_A, it only takes $O(n\lambda_n \log n)$ arithmetic operations to find e^* in (2.3). Thus the computational complexity of this search for two communities is not at all prohibitive—compare to the computational complexity of finding U_A itself, which is at least $O(n\lambda_n \log n)$ for $m = 2$.

2.3. *Extension to more than two communities.* Let K be the number of communities and S be an $n \times K$ label matrix: for $1 \le i \le n$, if node i belongs to community k then $S_{ik} = 1$ and $S_{il} = 0$ for all $l \ne k$. The numbers of edges between communities defined by (1.1) are entries of $S^T AS$. Let $B = \sum_{i=1}^{K} \rho_i \bar{u}_i \bar{u}_i^T$ define the eigen-decomposition of B. The population version of $S^T AS$ is

$$S^T BS = S^T \left(\sum_{j=1}^{K} \rho_j \bar{u}_j \bar{u}_j^T \right) S = \sum_{j=1}^{K} \rho_j (S^T \bar{u}_j)(S^T \bar{u}_j)^T.$$

Let U_B be the $K \times n$ matrix whose rows are $\bar{u}_j^T$. Then $S^T BS$ is a function of $U_B S$. We approximate U_B by U_A described in Section 2.1. Let $\tilde{S}$ be the first $K-1$ columns of S. Note that the rows of S sum to one, therefore, $U_A S$ can be recovered from $U_A \tilde{S}$. Now relax the entries of $\tilde{S}$ to take values in $[0, 1]$, with the row sums of at most one. For $1 \le i \le n$ and $1 \le j \le K-1$, denote by V_{ij} the $K \times (K-1)$ matrix such that the jth column of V_{ij} is the ith column of U_A and all other columns are zero. Then

$$U_A \tilde{S} = \sum_{i=1}^{n} \sum_{j=1}^{K-1} \tilde{S}_{ij} V_{ij}.$$

Since $\sum_{j=1}^{K-1} \tilde{S}_{ij} \le 1$, $\sum_{j=1}^{K-1} \tilde{S}_{ij} V_{ij}$ is a convex set in $\mathbb{R}^{K \times (K-1)}$, isomorphic to a $K-1$ simplex. Thus, $U_A \tilde{S}$ is a Minkowski sum of n convex sets in $\mathbb{R}^{K \times (K-1)}$. Similar to the case $K = 2$, we can first find the set of label matrices $\tilde{S}$ corresponding to the extreme points of $U_A \tilde{S}$ and then perform the exhaustive search over that set.

A bound on the number of vertices of $U_A \tilde{S}$ and a polynomial algorithm to find them are derived by [14]. If $d = K(K-1)$, then the number of vertices of $U_A \tilde{S}$ is at most $O(n^{(d-1)} K^{2(d-1)})$, and they can be found in $O(n^d K^{(2d-1)})$ arithmetic operations. An implementation of the reverse-search algorithm of [11] for computing the Minkowski sum of polytopes was presented in [43], who showed that the algorithm can be parallelized efficiently. We do not pursue these improvements here, since our main focus in this paper is the case $K = 2$.

3. Applications to community detection. Here, we apply the general results from Section 2 to a network adjacency matrix A, $B = \mathbb{E}[A]$, and functions corresponding to several popular community detection criteria. Our goal is to show that our maximization method gets an estimate close to the true label vector c, which is the maximizer of the corresponding function with $B = \mathbb{E}[A]$ plugged in for A. We focus on the case of two communities and use $m = 2$ for the low rank approximation.

Recall the quantities O_{11}, O_{22} and O_{12} defined in (1.1), which are used by all the criteria we consider. They are quadratic forms of A and e and can be written as

$$\begin{aligned} &O_{11}(e) = \tfrac{1}{4}(\mathbf{1}+e)^T A(\mathbf{1}+e), \qquad O_{22}(e) = \tfrac{1}{4}(\mathbf{1}-e)^T A(\mathbf{1}-e), \\ &O_{12}(e) = \tfrac{1}{4}(\mathbf{1}+e)^T A(\mathbf{1}-e), \end{aligned} \tag{3.1}$$

where $\mathbf{1}$ is the all-ones vector.

3.1. *Maximizing the likelihood of the degree-corrected stochastic block model.* When a network has two communities, (1.4) takes the form

$$\begin{aligned} Q_{\mathrm{DC}}(e) = {} & O_{11}\log O_{11} + O_{22}\log O_{22} + 2O_{12}\log O_{12} \\ & - 2O_1\log O_1 - 2O_2\log O_2. \end{aligned} \tag{3.2}$$

Thus, Q_{DC} has the form defined by (2.1).

For simplicity, instead of drawing c from a multinomial distribution with parameter $\pi = (\pi_1, \pi_2)$, we fix the true label vector by assigning the first $\bar{n}_1 = n\pi_1$ nodes to community 1 and the remaining $\bar{n}_2 = n\pi_2$ nodes to community 2. Let r be the out-in probability ratio, and

$$P = \lambda_n \begin{pmatrix} 1 & r \\ r & \omega \end{pmatrix} \tag{3.3}$$

be the probability matrix. We assume that the node degree parameters θ_i are an i.i.d. sample from a distribution with $\mathbb{E}[\theta_i] = 1$ and $1/\xi \le \theta_i \le \xi$ for some constant $\xi \ge 1$. The adjacency matrix A is symmetric and for $i > j$ has independent entries generated by $A_{ij} = \mathrm{Bernoulli}(\theta_i\theta_j P_{c_i c_j})$. Throughout the paper, we let λ_n depend on n, and fix r, ω, π, and ξ. Since λ_n and the network expected node degree are of the same order, in a slight abuse of notation, we also denote by λ_n the network expected node degree.

Theorem 2 establishes consistency of our method in this setting.

THEOREM 2. *Let A be the adjacency matrix generated from the DCSBM with λ_n growing at least as $\log^2 n$ as $n \to \infty$. Let U_A be an approximation of $U_{\mathbb{E}[A]}$, and e^* the label vector defined by (2.3) with $f_A = Q_{\mathrm{DC}}$. Then for any $\delta \in (0,1)$, there exists a constant $M = M(r, \omega, \pi, \xi, \delta) > 0$ such that with probability at least $1 - \delta$, we have*

$$\frac{1}{n}\|c - e^*\|^2 \le M\log n\big(\lambda_n^{-1/2} + \|U_A - U_{\mathbb{E}[A]}\|\big).$$

In particular, if U_A is a matrix whose row vectors are leading eignvectors of A, then the fraction of misclustered nodes is bounded by $M\log n/\sqrt{\lambda_n}$.

Note that assumption (2) is difficult to check for Q_{DC} but a weaker version, namely convexity along a certain direction, is sufficient for proving Theorem 2. The proof of Theorem 2 consists of checking assumptions (1), (3), (4) and a weaker version of assumption (2). For details, see the supplement [22].

3.2. *Maximizing the likelihood of the stochastic block model.* While the regular SBM is a special case of DCSBM when $\theta_i = 1$ for all i, its likelihood is different and thus maximizing it gives a different solution. With two communities, (1.3) admits the form

$$Q_{\mathrm{BM}}(e) = Q_{\mathrm{DC}}(e) + 2O_1 \log \frac{O_1}{n_1} + 2O_2 \log \frac{O_2}{n_2},$$

where $n_1 = n_1(e)$ and $n_2 = n_2(e)$ are the numbers of nodes in two communities and can be written as

$$n_1 = \tfrac{1}{2}(\mathbf{1} + e)^T \mathbf{1} = \tfrac{1}{2}(n + e^T \mathbf{1}), \qquad n_2 = \tfrac{1}{2}(\mathbf{1} - e)^T \mathbf{1} = \tfrac{1}{2}(n - e^T \mathbf{1}). \tag{3.4}$$

THEOREM 3. *Let A be the adjacency matrix generated from the SBM with λ_n growing at least as $\log^2 n$ as $n \to \infty$. Let U_A be an approximation of $U_{\mathbb{E}[A]}$, and e^* the label vector defined by (2.3) with $f_A = Q_{\mathrm{BM}}$. Then for any $\delta \in (0, 1)$, there exists a constant $M = M(r, \omega, \pi, \xi, \delta) > 0$ such that with probability at least $1 - n^{-\delta}$, we have*

$$\frac{1}{n}\|c - e^*\|^2 \le M \log n \big(\lambda_n^{-1/2} + \|U_A - U_{\mathbb{E}[A]}\|\big).$$

In particular, if U_A is a matrix whose row vectors are leading eigenvectors of A, then the fraction of misclustered nodes is bounded by $M \log n / \sqrt{\lambda_n}$.

Note that Q_{BM} does not have the exact form of (2.1) but a small modification shows that Lemma 1 still holds for Q_{BM}. Also, assumption (2) is difficult to check for Q_{BM} but again a weaker condition of convexity along a certain direction is sufficient for proving Theorem 3. The proof of Theorem 3 consists of showing the analog of Lemma 1, checking assumptions (3), (4) and a weaker version of assumption (2). For details, see the supplement [22].

3.3. *Maximizing the Newman–Girvan modularity.* When a network has two communities, up to a constant factor the modularity (1.2) takes the form

$$Q_{\mathrm{NG}}(e) = O_{11} + O_{22} - \frac{O_1^2 + O_2^2}{O_1 + O_2} = \frac{2O_1 O_2}{O_1 + O_2} - 2O_{12}.$$

Again, Q_{NG} does not have the exact form (2.1), but with a small modification, the argument used for proving Lemma 1 and Theorem 1 still holds for Q_{NG} under the regular SBM.

THEOREM 4. *Let A be the adjacency matrix generated from the SBM with λ_n growing at least as $\log n$ as $n \to \infty$. Let U_A be an approximation of $U_{\mathbb{E}[A]}$, and e^* the label vector defined by (2.3) with $f_A = Q_{\mathrm{NG}}$. Then for any $\delta \in (0, 1)$,*

there exists a constant $M = M(r, \omega, \pi, \xi, \delta) > 0$ *such that with probability at least* $1 - n^{-\delta}$, *we have*

$$\frac{1}{n}\|c - e^*\|^2 \le M(\lambda_n^{-1/2} + \|U_A - U_{\mathbb{E}[A]}\|).$$

In particular, if U_A *is a matrix whose row vectors are leading eigenvectors of* A, *then the fraction of misclustered nodes is bounded by* $M/\sqrt{\lambda_n}$.

It is easy to see that Q_{NG} is Lipschitz with respect to O_1, O_2 and O_{12}, which is stronger than assumption (1) and ensures the proof of Lemma 1 goes through. The proof of Theorem 4 consists of checking assumptions (2), (3), (4) and the Lipschitz condition for Q_{NG}. For details, see the supplement [22].

3.4. *Maximizing the community extraction criterion.* Identifying the community V to be extracted with a label vector e, the criterion (1.5) can be written as

$$Q_{\mathrm{EX}}(e) = \frac{n_2}{n_1} O_{11} - O_{12},$$

where n_1, n_2 are defined by (3.4). Once again Q_{EX} does not have the exact form (2.1), but with small modifications of the proof, Lemma 1 and Theorem 1 still hold for Q_{EX}.

THEOREM 5. *Let* A *be the adjacency matrix generated from the SBM with the probability matrix* (3.3), $\omega = r$, *and* λ_n *growing at least as* $\log n$ *as* $n \to \infty$. *Let* U_A *be an approximation of* $U_{\mathbb{E}[A]}$, *and* e^* *the label vector defined by* (2.3) *with* $f_A = Q_{\mathrm{EX}}$. *Then for any* $\delta \in (0, 1)$, *there exists a constant* $M = M(r, \omega, \pi, \xi, \delta) > 0$ *such that with probability at least* $1 - n^{-\delta}$, *we have*

$$\frac{1}{n}\|c - e^*\|^2 \le M(\lambda_n^{-1/2} + \|U_A - U_{\mathbb{E}[A]}\|).$$

In particular, if U_A *is a matrix whose row vectors are leading eigenvectors of* A, *then the fraction of misclustered nodes is bounded by* $M/\sqrt{\lambda_n}$.

The proof of Theorem 5 consists of verifying a version of Lemma 1 and assumptions (2), (3) and (4), and is included in the supplement [22].

3.5. *An alternative to exhaustive search.* While the projected feasible space is much smaller than the original space, we may still want to avoid the exhaustive search for e^* in (2.3). The geometry of the projection of the cube can be used to derive an approximation to e^* that can be computed without a search.

Recall that $U_{\mathbb{E}[A]}$ is an $2 \times n$ matrix whose rows are the leading eigenvectors of $\mathbb{E}[A]$, and U_A approximates $U_{\mathbb{E}}[A]$. For SBM, it is easy to see that $U_{\mathbb{E}[A]}[-1, 1]^n$, the projection of the unit cube onto the two leading eigenvectors of $U_{\mathbb{E}[A]}$, is

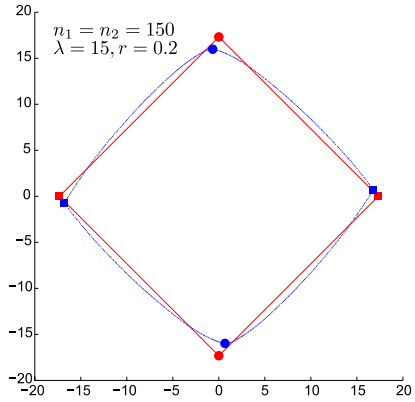

FIG. 1. *The projection of the cube* $[-1, 1]^n$ *onto two-dimensional subspace. Blue corresponds to the projection onto eigenvectors of* A*, and red onto the eigenvectors of* $\mathbb{E}[A]$*. The red contour is the boundary of* $U_{\mathbb{E}[A]}[-1, 1]^n$*; the blue dots are the extreme points of* $U_A[-1, 1]^n$*. Circles (at the corners) are* $\pm$ *projections of the true label vector; squares are* $\pm$ *projections of the vector of all* 1*s.*

a parallelogram with vertices $\{\pm U_{\mathbb{E}[A]}\mathbf{1}, \pm U_{\mathbb{E}[A]}c\}$, where $\mathbf{1} \in \mathbb{R}^n$ is a vector of all 1s (see Lemma 6 in the supplement [22]). We can then expect the projection $U_A[-1, 1]^n$ to look somewhat similar; see the illustration in Figure 1. Note that $\pm U_{\mathbb{E}[A]}c$ are the farthest points from the line connecting the other two vertices, $U_{\mathbb{E}[A]}\mathbf{1}$ and $-U_{\mathbb{E}[A]}\mathbf{1}$. Motivated by this observation, we can estimate c by

$$\begin{aligned}\hat{c} &= \arg\max\{\langle U_A e, (U_A\mathbf{1})^{\perp}\rangle : e \in \{-1, 1\}^n\} \\ &= \operatorname{sign}(u_1^T\mathbf{1}u_2 - u_2^T\mathbf{1}u_1),\end{aligned} \tag{3.5}$$

where $U_A = (u_1, u_2)^T$ and $(U_A\mathbf{1})^{\perp}$ is the unit vector perpendicular to $U_A\mathbf{1}$.

Note that $\hat{c}$ depends on U_A only, not on the objective function, a property it shares with spectral clustering. However, $\hat{c}$ provides a deterministic estimate of the labels based on a geometric property of U_A, while spectral clustering uses K-means, which is iterative and typically depends on a random initialization. Using this geometric approximation allows us to avoid both the exhaustive search and the iterations and initialization of K-means, although it may not always be as accurate as the search. When the community detection problem is relatively easy, we expect the geometric approximation to perform well, but when the problem becomes harder, the exhaustive search should provide better results. This intuition is confirmed by simulations in Section 4. Theorem 6 shows that $\hat{c}$ is a consistent estimator. The proof is given in Appendix B.

THEOREM 6. *Let* A *be an adjacency matrix generated from the SBM with* λ_n *growing at least as* $\log n$ *as* $n \to \infty$*. Let* U_A *be an approximation to* $U_{\mathbb{E}[A]}$*. Then for any* $\delta \in (0, 1)$ *there exists* $M = M(r, \omega, \pi, \xi, \delta) > 0$ *such that with probability*

at least $1 - n^{-\delta}$, *we have*

$$\frac{1}{n}\|\hat{c} - c\|^2 \le M\|U_A - U_{\mathbb{E}[A]}\|^2.$$

In particular, if U_A *is a matrix whose row vectors are leading eigenvectors of* A, *then the fraction of misclustered nodes is bounded by* M/λ_n.

3.6. *Theoretical comparisons.* There are several results on the consistency of recovering the true label vector under both the SBM and the DCSBM. The balanced planted partition model $G(n, \frac{a}{n}, \frac{b}{n})$, which is the simplest special case of the SBM, has received much attention recently, especially in the probability literature. This model assumes that there are two communities with $n/2$ nodes each, and edges are formed within communities and between communities with probabilities a/n and b/n, respectively. When $(a-b)^2 \le 2(a+b)$, no method can find the communities [29]. Algorithms based on non-backtracking random walks that can recover the community structure better than random guessing if $(a-b)^2 > 2(a+b)$ have been proposed in [27, 31]. Moreover, if $(a-b)^2/(a+b) \to \infty$ as $n \to \infty$ then the fraction of misclustered nodes goes to zero with high probability. Under the model $G(n, \frac{a}{n}, \frac{b}{n})$, our theoretical results require that $a+b$ grows at least as $\log n$. This matches the requirements on the expected degree λ_n needed for consistency in [6] for the SBM and in [47] for the DCSBM.

When the expected node degree λ_n is of order $\log n$, spectral clustering using eigenvectors of the adjacency matrix can correctly recover the communities, with fraction of misclustered nodes up to $O(1/\log n)$ [23]. In this regime, our method for maximizing the Newman–Girvan and the community extraction criteria misclusters at most $O(1/\sqrt{\lambda_n})$ fraction of the nodes. For maximizing the likelihoods of the SBM and DCSBM, we require that λ_n is of order $\log^2 n$, and the fraction of misclustered nodes is bounded by $O(\log n/\sqrt{\lambda_n})$. For Newman–Girvan modularity as well as the SBM likelihood, [6] proved strong consistency (perfect recovery with high probability) under the SBM when λ_n grows faster than $\log n$. However, they used a label-switching algorithm for finding the maximizer, which is computationally infeasible for larger networks. A much faster algorithm based on pseudo-likelihood was proposed by [3], who assumed that the initial estimate of the labels (obtained in practice by regularized spectral clustering) has a certain correlation with the truth, and showed that the fraction of misclustered nodes for their method is $O(1/\lambda_n)$. Recently, [21] analyzed regularized spectral clustering in the sparse regime when $\lambda_n = O(1)$, and showed that with high probability, the fraction of misclustered nodes is $O(\log^6 \lambda_n/\lambda_n)$. In summary, our assumptions required for consistency are similar to others in the literature even though the approximation method is fairly general.

4. Numerical comparisons. Here, we briefly compare the empirical performance of our extreme point projection method to several other methods for community detection, both general (spectral clustering) and those designed specifically for optimizing a particular community detection criterion, using both simulated networks and two real network datasets, the political blogs and the dolphins data described in Section 4.5. Our goal in this comparison is to show that our general method does as well as the algorithms tailored to a particular criterion, and thus we are not trading off accuracy for generality.

For the four criteria discussed in Section 3, we compare our method of maximizing the relevant criterion by exhaustive search over the extreme points of the projection (EP, for extreme points), the approximate version based on the geometry of the feasible set described in Section 3.5 (AEP, for approximate extreme points) and regularized spectral clustering (SCR) proposed by [3], which are all general methods. We also include one method specific to the criterion in each comparison. For the SBM, we compare to the unconditional pseudo-likelihood (UPL) and for the DCSBM, to the conditional pseudo-likelihood (CPL), two fast and accurate methods developed specifically for these models by [3]. For the Newman–Girvan modularity, we compare to the spectral algorithm of [34], which uses the leading eigenvector of the modularity matrix (see details in Section 4.3). Finally, for community extraction we compare to the algorithm proposed in the original paper [46] based on greedy label switching, as there are no faster algorithms available.

The simulated networks are generated using the parameterization of [3], as follows. Throughout this section, the number of nodes in the network is fixed at $n = 300$, the number of communities $K = 2$, and the true label vector c is fixed. The number of replications for each setting is 100. First, the node degree parameters θ_i are drawn independently from the distribution $\mathbb{P}(\Theta = 0.2) = \gamma$, and $\mathbb{P}(\Theta = 1) = 1 - \gamma$. Setting $\gamma = 0$ gives the standard SBM, and $\gamma > 0$ gives the DCSBM, with $1 - \gamma$ the fraction of hub nodes. The matrix of edge probabilities P is controlled by two parameters: the out-in probability ratio r, which determines how likely edges are formed within and between communities, and the weight vector $w = (w_1, w_2)$, which determines the relative node degrees within communities. Let

$$P_0 = \begin{bmatrix} w_1 & r \\ r & w_2 \end{bmatrix}.$$

The difficulty of the problem is largely controlled by r and the overall expected network degree λ. Thus we rescale P_0 to control the expected degree, setting

$$P = \frac{\lambda P^0}{(n-1)(\pi^T P^0 \pi)(\mathbb{E}[\Theta])^2},$$

where $\pi = n^{-1}(n_1, n_2)$, and n_k is the number of nodes in community k. Finally, edges A_{ij} are drawn independently from a Bernoulli distribution with $\mathbb{P}(A_{ij} = 1) = \theta_i \theta_j P_{c_i c_j}$.

As discussed in Section 2.1, a good approximation to the eigenvectors of $\mathbb{E}[A]$ is provided by the eigenvectors of the regularized Laplacian. SCR uses these eigenvectors u_1, u_2 as input to K-means (computed here with the kmeans function in Matlab with 40 random initial starting points). EP and AEP use $\{D^{1/2}u_1, D^{1/2}u_2\}$ to compute the matrix U_A (see Section 2.1). To find extreme points and corresponding label vectors in the second step of EP, we use the algorithm of [14]. For $m = 2$, it essentially consists of sorting the angles of between the column vectors of U_A and the x-axis. In case of multiple maximizers, we break the tie by choosing the label vector whose projection is the farthest from the line connecting the projections of $\pm\mathbf{1}$ (following the geometric idea of Section 3.5). For CPL and UPL, following [3], we initialize with the output of SCR and set the number of outer iterations to 20.

We measure the accuracy of all methods via the normalized mutual information (NMI) between the label vector c and its estimate e. NMI takes values between 0 (random guessing) and 1 (perfect match), and is defined by [44] as $\mathrm{NMI}(c, e) = -\sum_{i,j} R_{ij} \log \frac{R_{ij}}{R_{i+}R_{+j}} (\sum_{ij} R_{ij} \log R_{ij})^{-1}$, where R is the confusion matrix between c and e, which represents a bivariate probability distribution, and its row and column sums R_{i+} and R_{+j} are the corresponding marginals.

4.1. *The degree-corrected stochastic block model.* Figure 2 shows the performance of the four methods for fitting the DCSBM under different parameter settings. We use the notation EP[DC] to emphasize that EP here is used to maximize the log-likelihood of DCSBM. In this case, all methods perform similarly,

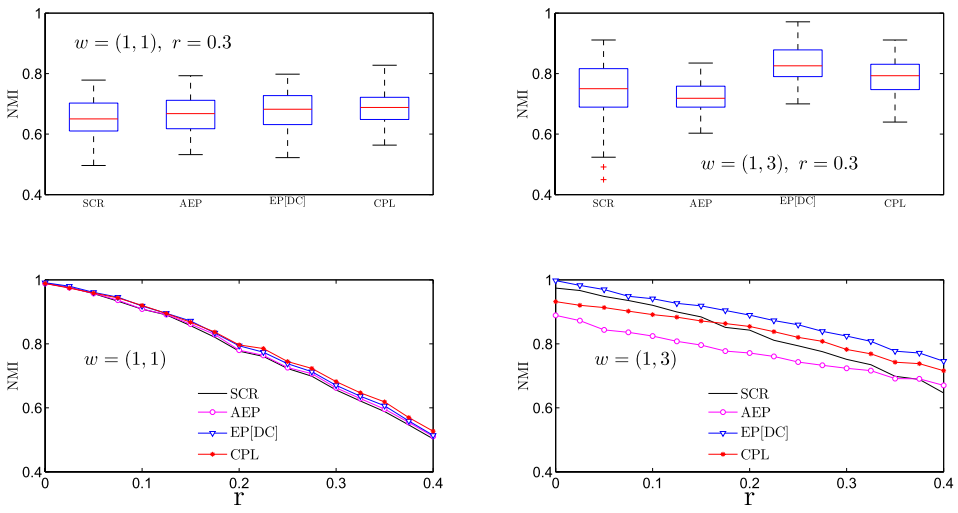

FIG. 2. *The degree-corrected stochastic block model. Top row*: *boxplots of NMI between true and estimated labels. Bottom row*: *average NMI against the out-in probability ratio* r. *In all plots*, $n_1 = n_2 = 150$, $\lambda = 15$ *and* $\gamma = 0.5$.

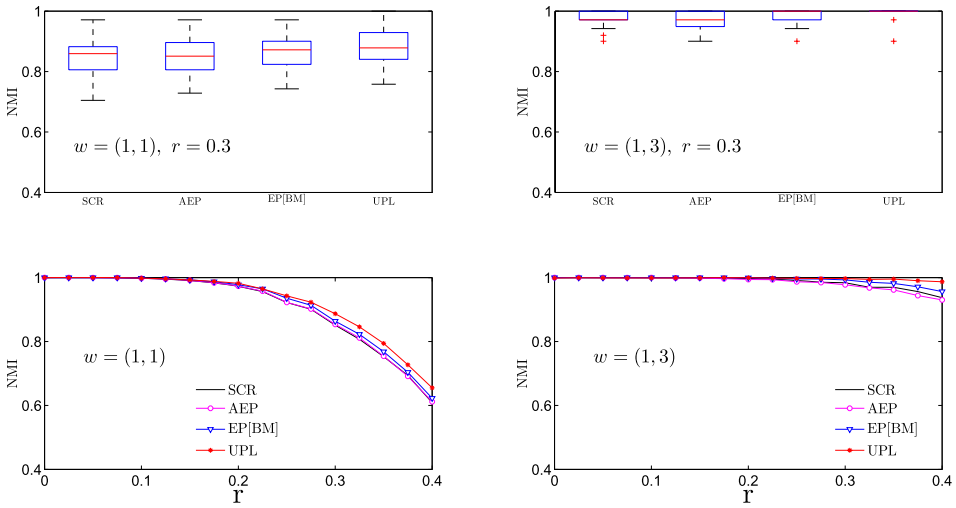

FIG. 3. *The stochastic block model. Top row*: *boxplots of NMI between true and estimated labels. Bottom row*: *average NMI against the out-in probability ratio* r. *In all plots*, $n_1 = n_2 = 150$, $\lambda = 15$ *and* $\gamma = 0$.

with EP performing the best when community-level degree weights are different ($w = (1, 3)$), but just slightly worse than CPL when $w = (1, 1)$. The AEP is always somewhat worse than the exact version, especially when $w = (1, 3)$, but overall their results are comparable.

4.2. *The stochastic block model.* Figure 3 shows the performance of the four methods for fitting the regular SBM ($\gamma = 0$). Over all, four methods provide quite similar results, as we would hope good fitting methods will. The performance of the approximate method AEP is very similar to that of EP, and the model-specific UPL marginally outperforms the three general methods.

4.3. *Newman–girvan modularity.* The modularity function $\hat{Q}_{\mathrm{NG}}$ can be approximately maximized via a fast spectral algorithm when partitioning into two communities [34]. Let $B = A - P$ where $P_{ij} = d_i d_j / m$, and write $\hat{Q}_{\mathrm{NG}}(e) = \frac{1}{2m} e^T B e$. The approximate solution (LES, for leading eigenvector signs) assigns node labels according to the signs of the corresponding entries of the leading eigenvector of B. For a fair comparison to other methods relying on eigenvectors, we also use the regularized $A + \tau \mathbf{1}\mathbf{1}^T$ instead of A here, since empirically we found that it slightly improves the performance of LES. Figure 4 shows the performance of AEP, EP[NG] and LES, when the data are generated from a regular block model ($\gamma = 0$). The two extreme point methods EP[NG] and AEP both do slightly better than LES, especially for the unbalanced case of $w = (1, 3)$, and there is essentially no difference between EP[NG] and AEP here.

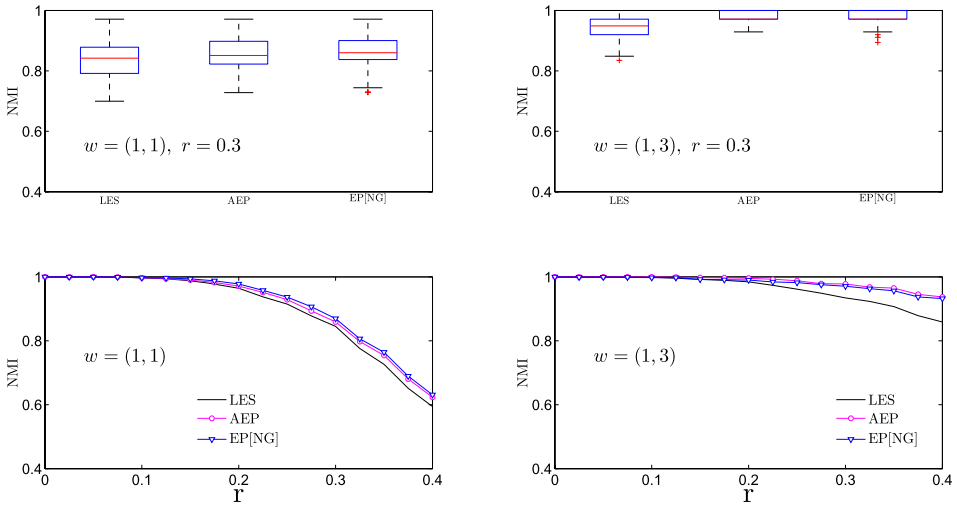

FIG. 4. *Newman–Girvan modularity. Top row*: *boxplots of NMI between true and estimated labels. Bottom row*: *average NMI against the out-in probability ratio* r. *In all plots*, $n_1 = n_2 = 150$, $\lambda = 15$ *and* $\gamma = 0$.

4.4. *Community extraction criterion.* Following the original extraction paper of [46], we generate a community with background from the regular block model with $K = 2$, $n_1 = 60$, $n_2 = 240$ and the probability matrix proportional to

$$P_0 = \begin{pmatrix} 0.4 & 0.1 \\ 0.1 & 0.1 \end{pmatrix}.$$

Thus, nodes within the first community are tightly connected, while the rest of the nodes have equally weak links with all other nodes and represent the background. We consider four values for the average expected node degree, 15, 20, 25 and 30. Figure 5 shows that EP[EX] performs better than SCR and AEP, but somewhat worse than the greedy label-switching tabu search used in the original paper for maximizing the community extraction criterion (TS). However, the tabu search is very computationally intensive and only feasible up to perhaps a thousand nodes, so for larger networks it is not an option at all, and no other method has been previously proposed for this problem. The AEP method, which does not agree with AE as well as in the other cases, probably suffers from the inherent asymmetry of the extraction problem.

4.5. *Real-world network data.* The first network we test our methods on, assembled by [1], consists of blogs about US politics and hyperlinks between blogs. Each blog has been manually labeled as either liberal or conservative, which we use as the ground truth. Following [20] and [47], we ignore directions of the hyperlinks and only examine the largest connected component of this network, which

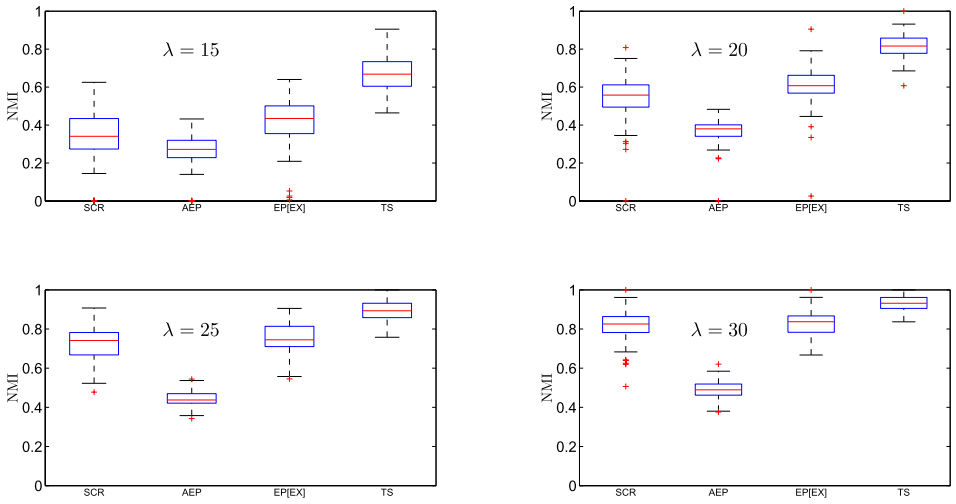

FIG. 5. *Community extraction. The boxplots of NMI between true and estimated labels. In all plots,* $n_1 = 60$, $n_2 = 240$ *and* $\gamma = 0$.

has 1222 nodes and 16,714 edges, with the average degree of approximately 27. Table 1 and Figure 6 show the performance of different methods. While AEP, EP[DC] and CPL give reasonable results, SCR, UPL and EP[BM] clearly miscluster the nodes. This is consistent with previous analyses which showed that the degree correction has to be used for this network to achieve the correct partition, because of the presence of hub nodes.

The second network we study represents social ties between 62 bottlenose dolphins living in Doubtful Sound, New Zealand [24, 25]. At some point during the study, one well-connected dolphin (SN100) left the group, and the group split into two separate parts, which we use as the ground truth in this example. Table 1 and Figure 7 show the performance of different methods. In Figure 7, node shapes represent the actual split, while the colors represent the estimated label. The star-shaped node is the dolphin SN100 that left the group. Excepting that dolphin, SCR, EP[BM], EP[DC], UPL and CPL all miscluster one node, while AEP misclusters two nodes. Since this small network can be well modeled by the SBM, there is no

TABLE 1
The NMI between true and estimated labels for real-world networks

Method	**SCR**	**AEP**	**EP[BM]**	**EP[DC]**	**UPL**	**CPL**
Blogs	0.290	0.674	0.278	0.731	0.001	0.725
Dolphins	0.889	0.814	0.889	0.889	0.889	0.889

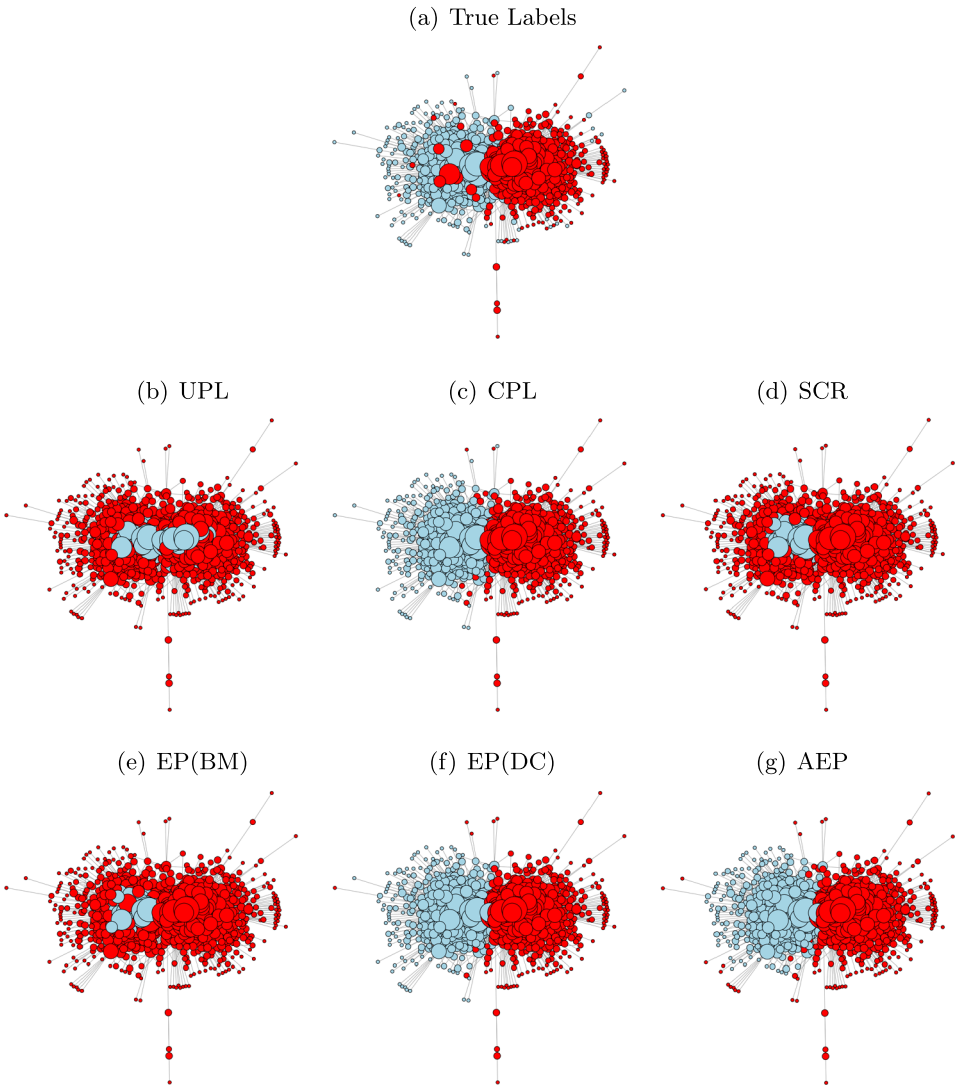

FIG. 6. *The network of political blogs. Node diameter is proportional to the logarithm of its degree and the colors represent community labels.*

difference between DCSBM and SBM based methods, and all methods perform well.

APPENDIX A: PROOF OF RESULTS IN SECTION 2

The following lemma bounds the Lipschitz constants of $h_{B,j}$ and f_B on $U_B[-1,1]^n$.

(a) AEP (b) SCR, EP, UPL, CPL

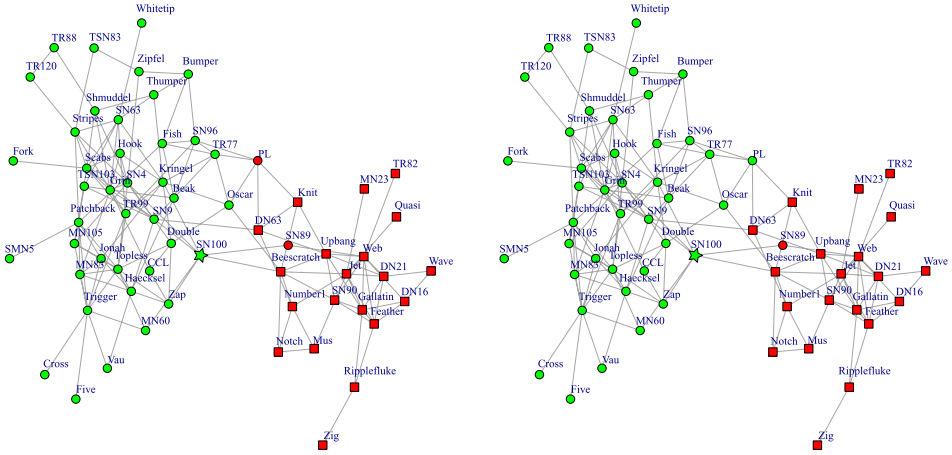

FIG. 7. *The network of* 62 *bottlenose dolphins. Node shapes represent the split after the dolphin SN*100 (*represented by the star*) *left the group. Node colors represent their estimated labels.*

LEMMA 2. *Assume that assumption* (1) *holds. For any* $j \le \kappa$ [*see* (2.1)], *and* $x, y \in U_B[-1,1]^n$, *we have*

$$\begin{aligned} \left|h_{B,j}(x) - h_{B,j}(y)\right| &\le 4\sqrt{n}\|B\| \cdot \|x-y\|, \\ \left|f_B(x) - f_B(y)\right| &\le M\sqrt{n}\log(n)\|B\| \cdot \|x-y\|, \end{aligned}$$

where M *is a constant independent of* n.

PROOF. Let $e, s \in [-1,1]^n$ such that $x = U_B e$, $y = U_B s$ and denote $L = |h_{B,j}(x) - h_{B,j}(y)|$. Then

$$\begin{aligned} L &= \left|(e+s_{j1})^T B(e+s_{j2}) - (s+s_{j1})^T B(s+s_{j2})\right| \\ &= \left|e^T B(e-s) + (e-s)^T Bs + (s_{j2}+s_{j1})^T B(e-s)\right| \\ &\le 4\sqrt{n}\left\|B(e-s)\right\|. \end{aligned}$$

Let $B = \sum_{i=1}^m \rho_i u_i u_i^T$ be the eigen-decomposition of B. Then

$$\begin{aligned} \left\|B(e-s)\right\|^2 &= \left\|\sum_{i=1}^m \rho_i u_i u_i^T (e-s)\right\|^2 = \left\|\sum_{i=1}^m \rho_i (x_i - y_i) u_i\right\|^2 \\ &= \sum_{i=1}^m \rho_i^2 (x_i - y_i)^2 \le \|B\|^2 \sum_{i=1}^m (x_i - y_i)^2 = \|B\|^2 \cdot \|x-y\|^2. \end{aligned}$$

Therefore, $L \le 4\sqrt{n}\|B\| \cdot \|x-y\|$. Since $h_{B,j}$ are quadratic, they are of order $O(n^2)$. Hence, by assumption (1), the Lipschitz constants of g_j are of order $\log(n)$.

Therefore,

$$\big|f_B(x) - f_B(y)\big| \le 4\sqrt{n}\log(n)\|B\| \cdot \|x - y\|,$$

which completes the proof. □

In the following proofs, we use M to denote a positive constant independent of n the value of which may change from line to line.

PROOF OF LEMMA 1. Since $\|e + s_{j1}\| \le 2\sqrt{n}$ and $\|e + s_{j2}\| \le 2\sqrt{n}$,

$$\begin{aligned}\big|h_{A,j}(e) - h_{B,j}(e)\big| &= \big|(e + s_{j1})^T (A - B)(e + s_{j2})\big| \\ &\le 4n\|A - B\|.\end{aligned}$$

Since $h_{A,j}$ and $h_{B,j}$ are of order $O(n^2)$, g_j' are bounded by $\log(n)$. Together with assumption (1), it implies that there exists $M > 0$ such that

$$\big|f_A(e) - f_B(e)\big| \le Mn\log(n)\|A - B\|. \tag{A.1}$$

Let $\hat{e} = \arg\max\{f_B(e), e \in \mathcal{E}_A\}$. Then $f_A(e^*) \ge f_A(\hat{e})$ and by (A.1) we get

$$\begin{aligned}f_B(\hat{e}) - f_B\big(e^*\big) &\le f_B(\hat{e}) - f_A(\hat{e}) + f_A\big(e^*\big) - f_B\big(e^*\big) \\ &\le Mn\log(n)\|A - B\|.\end{aligned} \tag{A.2}$$

Denote by $\operatorname{conv}(S)$ the convex hull of a set S. Then $U_A c \in \operatorname{conv}(U_A\mathcal{E}_A)$ and therefore, there exists $\eta_e \ge 0$, $\sum_{e\in\mathcal{E}_A} \eta_e = 1$ such that

$$U_A c = \sum_{e\in\mathcal{E}_A} \eta_e U_A(e) = U_A\Big(\sum_{e\in\mathcal{E}_A} \eta_e e\Big).$$

Hence,

$$\begin{aligned}\operatorname{dist}\big(U_B c, \operatorname{conv}(U_B\mathcal{E}_A)\big) &\le \Big\|U_B c - U_B\Big(\sum_{e\in\mathcal{E}_A} \eta_e e\Big)\Big\| \\ &= \Big\|(U_B - U_A)c + (U_A - U_B)\sum_{e\in\mathcal{E}_A} \eta_e e\Big\| \\ &\le 2\sqrt{n}\|U_A - U_B\|.\end{aligned} \tag{A.3}$$

Let $y \in \operatorname{conv}(U_B\mathcal{E}_A)$ be the closest point from $\operatorname{conv}(U_B\mathcal{E}_A)$ to $U_B c$, that is,

$$\|U_B c - y\| = \operatorname{dist}\big(U_B c, \operatorname{conv}(U_B\mathcal{E}_A)\big).$$

By A.3 and Lemma 2, we have

$$f_B(U_B c) - f_B(y) \le Mn\log(n)\|B\| \cdot \|U_A - U_B\|. \tag{A.4}$$

The convexity of f_B implies that $f_B(y) \le f_B(U_B\hat{e})$, and in turn,

$$f_B(U_B c) - f_B(U_B\hat{e}) \le Mn\log(n)\|B\| \cdot \|U_A - U_B\|. \tag{A.5}$$

Note that $f_B(U_B e) = f_B(e)$ for every $e \in [-1,1]^n$. Adding (A.2) and (A.5), we get (2.4) for $T = B$. The case $T = A$ then follows from (A.1) because replacing B with A induces an error which is not greater than the upper bound of (2.4) for $T = B$. □

APPENDIX B: PROOF OF THEOREM 6

We first present the closed form of eigenvalues and eigenvectors of $\mathbb{E}[A]$ under the regular block models.

LEMMA 3. *Under the SBM, the nonzero eigenvalues ρ_i and corresponding eigenvectors $\bar{u}_i$ of $\mathbb{E}[A]$ have the following form. For $i = 1, 2$,*

$$\rho_i = \frac{\lambda_n}{2}\big[(\pi_1 + \pi_2\omega) + (-1)^{i-1}\sqrt{(\pi_1 + \pi_2\omega)^2 - 4\pi_1\pi_2(\omega - r^2)}\big],$$

$$\bar{u}_i = \frac{1}{\sqrt{n(\pi_1 r_i^2 + \pi_2)}}(r_i, r_i, \ldots, r_i, 1, 1, \ldots, 1)^T,$$

where

$$r_i = \frac{2\pi_2 r}{(\pi_2\omega - \pi_1) + (-1)^i\sqrt{(\pi_1 + \pi_2\omega)^2 - 4\pi_1\pi_2(\omega - r^2)}}.$$

The first $\bar{n}_1 = n\pi_1$ entries of $\bar{u}_i$ equal $r_i(n(\pi_1 r_i^2 + \pi_2))^{-1/2}$ and the last $\bar{n}_2 = n\pi_2$ entries of $\bar{u}_i$ equal $(n(\pi_1 r_i^2 + \pi_2))^{-1/2}$.

PROOF. Under the SBM $\mathbb{E}[A]$ is a two-by-two block matrix with equal entries within each block. It is easy to verify directly that $\mathbb{E}[A]\bar{u}_i = \rho_i\bar{u}_i$ for $i = 1, 2$. □

Lemma 4 bounds the difference between the eigenvalues and eigenvectors of A and those of $\mathbb{E}[A]$ under the SBM. It also provides a way to simplify the general upper bound of Theorem 1.

LEMMA 4. *Under the SBM, let U_A and $U_{\mathbb{E}[A]}$ be $2 \times n$ matrices whose rows are the leading eigenvectors of A and $\mathbb{E}[A]$, respectively. For any $\delta > 0$, there exists a constant $M = M(r, \omega, \pi, \delta) > 0$ such that if $\lambda_n > M\log(n)$ then with probability at least $1 - n^{-\delta}$, we have*

$$\|A - \mathbb{E}[A]\| \le M\sqrt{\lambda_n}, \tag{B.1}$$

$$\|U_A - U_{\mathbb{E}[A]}\| \le \frac{M}{\sqrt{\lambda_n}}. \tag{B.2}$$

PROOF. Inequality (B.1) follows directly from Theorem 5.2 of [23] and the fact that the maximum of the expected node degrees is of order λ_n. Inequality (B.2) is a consequence of (B.1) and the Davis–Kahan theorem (see Theorem VII.3.2 of [5]) as follows. By Lemma 3, the nonzero eigenvalues ρ_1 and ρ_2 of $\bar{A}$ are of order λ_n. Let

$$\mathcal{S} = [\rho_2 - M\sqrt{\lambda_n}, \rho_1 + M\sqrt{\lambda_n}].$$

Then $\rho_1, \rho_2 \in \mathcal{S}$ and the gap between $\mathcal{S}$ and zero is of order λ_n. Let $\bar{P}$ be the projector onto the subspace spanned by two leading eigenvectors of $\mathbb{E}[A]$. Since λ_n grows faster than $\|A - \mathbb{E}[A]\|$ by B.1, only two leading eigenvalues of A belong to $\mathcal{S}$. Let P be the projector onto the subspace spanned by two leading eigenvectors of A. By the Davis–Kahan theorem,

$$\|U_A - U_{\mathbb{E}[A]}\| = \|\bar{P} - P\| \le \frac{2\|A - \mathbb{E}[A]\|}{\lambda_n} \le \frac{2M}{\sqrt{\lambda_n}},$$

which completes the proof. □

Before proving Theorem 6, we need to establish the following lemma.

LEMMA 5. *Let $x, y, \bar{x}$ and $\bar{y}$ be unit vectors in $\mathbb{R}^n$ such that $\langle x, y\rangle = \langle \bar{x}, \bar{y}\rangle = 0$. Let P and $\bar{P}$ be the orthogonal projections on the subspaces spanned by $\{x, y\}$ and $\{\bar{x}, \bar{y}\}$, respectively. If $\|P - \bar{P}\| \le \epsilon$, then there exists an orthogonal matrix $\mathcal{K}$ of size 2×2 such that $\|(x, y)\mathcal{K} - (\bar{x}, \bar{y})\|_F \le 9\epsilon$.*

PROOF. Let $x_0 = P\bar{x}$ and $y_0 = P\bar{y}$. Since $\|P - \bar{P}\| \le \epsilon$, it follows that $\|\bar{x} - x_0\| \le \epsilon$ and $\|\bar{y} - y_0\| \le \epsilon$. Let $x^\perp = \frac{x_0}{\|x_0\|}$, then

$$\|\bar{x} - x^\perp\| \le \|\bar{x} - x_0\| + \|x_0 - x^\perp\| \le \epsilon + |1 - \|x_0\|| \le 2\epsilon.$$

Also $\langle x^\perp, y_0\rangle = \langle x^\perp, y_0 - \bar{y}\rangle + \langle x^\perp - \bar{x}, \bar{y}\rangle$ implies that $|\langle x^\perp, y_0\rangle| \le 3\epsilon$. Define $z = y_0 - \langle y_0, x^\perp\rangle x^\perp$. Then $\langle z, x^\perp\rangle = 0$, $\|\bar{y} - z\| \le \|\bar{y} - y_0\| + \|y_0 - z\| \le 4\epsilon$, and $|1 - \|z\|| = |\|\bar{y}\| - \|z\|| \le 4\epsilon$. Let $y^\perp = \frac{1}{\|z\|}z$, then

$$\|\bar{y} - y^\perp\| \le \|\bar{y} - z\| + \|z - y^\perp\| \le 4\epsilon + |1 - \|z\|| \le 8\epsilon.$$

Therefore, $\|(\bar{x}, \bar{y}) - (x^\perp, y^\perp)\|_F \le 9\epsilon$. Finally, let $\mathcal{K} = (x, y)^T(x^\perp, y^\perp)$. □

PROOF OF THEOREM 6. Denote $\varepsilon = \|U_A - U_{\mathbb{E}[A]}\|$, $U = (u_1, u_2)^T = U_A$, and $\bar{U} = (\bar{u}_1, \bar{u}_2)^T = U_{\mathbb{E}[A]}$. We first show that there exists a constant $M > 0$ such that with probability at least $1 - \delta$,

$$\min\|(u_1^T\mathbf{1}u_2 - u_2^T\mathbf{1}u_1) \pm (\bar{u}_1^T\mathbf{1}\bar{u}_2 - \bar{u}_2^T\mathbf{1}\bar{u}_1)\| \le M\varepsilon\sqrt{n}. \tag{B.3}$$

Let $\mathcal{R} = \left(\begin{smallmatrix} 0 & -1 \\ 1 & 0 \end{smallmatrix}\right)$ be the $\pi/2$-rotation on $\mathbb{R}^2$. Then

$$u_1^T\mathbf{1}u_2 - u_2^T\mathbf{1}u_1 = U^T\mathcal{R}U\mathbf{1}, \qquad \bar{u}_1^T\mathbf{1}\bar{u}_2 - \bar{u}_2^T\mathbf{1}\bar{u}_1 = \bar{U}^T\mathcal{R}\bar{U}\mathbf{1}.$$

By Lemma 4 and Lemma 5, there exists an orthogonal matrix $\mathcal{K}$ such that if $E = (E_1, E_2) = U^T - \bar{U}^T\mathcal{K}$ then $\|E\|_F \le 9\varepsilon$. By replacing U^T with $E + \bar{U}^T\mathcal{K}$, the left-hand side of (B.3) becomes

$$\min \|(E + \bar{U}^T\mathcal{K})\mathcal{R}(E + \bar{U}^T\mathcal{K})^T\mathbf{1} \pm \bar{U}^T\mathcal{R}\bar{U}\mathbf{1}\|.$$

Note that $\mathcal{K}^T\mathcal{R}\mathcal{K} = \mathcal{R}$ if $\mathcal{K}$ is a rotation, and $\mathcal{K}^T\mathcal{R}\mathcal{K} = -\mathcal{R}$ if $\mathcal{K}$ is a reflection. Therefore, it is enough to show that

$$\|\bar{U}^T\mathcal{K}\mathcal{R}E^T\mathbf{1} + E\mathcal{R}\mathcal{K}^T\bar{U}\mathbf{1} + E\mathcal{R}E^T\mathbf{1}\| \le M\epsilon\sqrt{n}.$$

Note that $|E_i^T\mathbf{1}| \le \sqrt{n}\|E_i\| \le 9\varepsilon\sqrt{n}$ and $\|E\|_F \le 9\varepsilon \le 18$, so

$$\|E\mathcal{R}E^T\mathbf{1}\| = \|E_2^T\mathbf{1}E_1 - E_1^T\mathbf{1}E_2\| \le 18^2\varepsilon\sqrt{n}.$$

From Lemma 3 we see that $\bar{U}\mathbf{1} = \sqrt{n}(s_1, s_2)^T$ for some s_1 and s_2 not depending on n. It follows that

$$\|E\mathcal{R}\mathcal{K}^T\bar{U}\mathbf{1}\| = \sqrt{n}\|(E_2 - E_1)\mathcal{K}^T(s_1, s_2)^T\| \le M\varepsilon\sqrt{n}$$

for some $M > 0$. Analogously,

$$\|\bar{U}^T\mathcal{K}\mathcal{R}E^T\mathbf{1}\| = \|\bar{U}^T\mathcal{K}(-E_2^T\mathbf{1}, E_1^T\mathbf{1})^T\| \le M\varepsilon\sqrt{n},$$

and (B.3) follows. By Lemma 3, we have

$$\bar{U}^T\mathcal{R}\bar{U}\mathbf{1} = \alpha(\pi_2, \pi_2, \ldots, \pi_2, -\pi_1, \ldots, -\pi_1)^T,$$

where α does not depend on n; the first n_1 entries of $\bar{U}^T\mathcal{R}\bar{U}\mathbf{1}$ equal $\alpha\pi_2$ and the last n_2 entries of $\bar{U}^T\mathcal{R}\bar{U}\mathbf{1}$ equal $\alpha\pi_1$. For simplicity, assume that in (B.3) the minimum is when the sign is negative (because $\hat{c}$ is unique up to a factor of -1). If node i is misclustered by $\hat{c}$ then

$$|(U^T\mathcal{R}U\mathbf{1})_i - (\bar{U}^T\mathcal{R}\bar{U}\mathbf{1})_i| \ge \min_i |(\bar{U}^T\mathcal{R}\bar{U}\mathbf{1})_i| =: \eta.$$

Let k be the number of misclustered nodes, then by (B.3), $\eta\sqrt{k} \le M\varepsilon\sqrt{n}$. Therefore the fraction of misclustered nodes, k/n, is of order ε^2. If U_A is formed by the leading eigenvectors of A, then it remains to use inequality (B.2) of Lemma 4. □

Acknowledgments. We thank the Associate Editor and three anonymous referees for detailed and constructive feedback which led to many improvements. We also thank Yunpeng Zhao (George Mason University) for sharing his code for the tabu search, and Arash A. Amini (UCLA) for sharing his code for the pseudo-likelihood methods and helpful discussions.

SUPPLEMENTARY MATERIAL

Supplement to "Optimization via low-rank approximation for community detection in networks" (DOI: 10.1214/15-AOS1360SUPP; .pdf). This supplement contains proofs of Theorems 2, 3, 4 and 5.

REFERENCES

[1] ADAMIC, L. A. and GLANCE, N. (2005). The political blogosphere and the 2004 U.S. election: Divided they blog. In *LinkKDD'05—Proceedings of the 3rd International Workshop on Link Discovery* 36–43. ACM, New York.

[2] AIROLDI, E. M., BLEI, D. M., FIENBERG, S. E. and XING, E. P. (2008). Mixed membership stochastic blockmodels. *J. Mach. Learn. Res.* **9** 1981–2014.

[3] AMINI, A. A., CHEN, A., BICKEL, P. J. and LEVINA, E. (2013). Pseudo-likelihood methods for community detection in large sparse networks. *Ann. Statist.* **41** 2097–2122. MR3127859

[4] BALL, B., KARRER, B. and NEWMAN, M. E. J. (2011). An efficient and principled method for detecting communities in networks. *Phys. Rev. E* **34** 036103.

[5] BHATIA, R. (1997). *Matrix Analysis*. Springer, New York. MR1477662

[6] BICKEL, P. J. and CHEN, A. (2009). A nonparametric view of network models and Newman–Girvan and other modularities. *Proc. Natl. Acad. Sci. USA* **106** 21068–21073.

[7] CHAUDHURI, K., CHUNG, F. and TSIATAS, A. (2012). Spectral clustering of graphs with general degrees in the extended planted partition model. *J. Mach. Learn. Res. Workshop Conf. Proc.* **23** 35.1–35.23.

[8] CHUNG, F. and LU, L. (2002). Connected components in random graphs with given expected degree sequences. *Ann. Comb.* **6** 125–145. MR1955514

[9] DECELLE, A., KRZAKALA, F., MOORE, C. and ZDEBOROVÁ, L. (2012). Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Phys. Rev. E* **84** 066106.

[10] ERDŐS, P. and RÉNYI, A. (1959). On random graphs. I. *Publ. Math. Debrecen* **6** 290–297. MR0120167

[11] FUKUDA, K. (2004). From the zonotope construction to the Minkowski addition of convex polytopes. *J. Symbolic Comput.* **38** 1261–1272. MR2094220

[12] GLOVER, F. and LAGUNA, M. (1998). Tabu search. In *Handbook of Combinatorial Optimization*, *Vol.* 3 621–757. Kluwer Academic, Boston, MA. MR1665424

[13] GOLDENBERG, A., ZHENG, A. X., FIENBERG, S. E. and AIROLDI, E. M. (2010). A survey of statistical network models. *Faund. Trends Mach. Learn.* **2** 129–233.

[14] GRITZMANN, P. and STURMFELS, B. (1993). Minkowski addition of polytopes: Computational complexity and applications to Gröbner bases. *SIAM J. Discrete Math.* **6** 246–269. MR1215231

[15] HANDCOCK, M. S., RAFTERY, A. E. and TANTRUM, J. M. (2007). Model-based clustering for social networks. *J. Roy. Statist. Soc. Ser. A* **170** 301–354. MR2364300

[16] HOFF, P. D., RAFTERY, A. E. and HANDCOCK, M. S. (2002). Latent space approaches to social network analysis. *J. Amer. Statist. Assoc.* **97** 1090–1098. MR1951262

[17] HOLLAND, P. W., LASKEY, K. B. and LEINHARDT, S. (1983). Stochastic blockmodels: First steps. *Social Networks* **5** 109–137. MR0718088

[18] JIN, J. (2015). Fast community detection by SCORE. *Ann. Statist.* **43** 57–89. MR3285600

[19] JOSEPH, A. and YU, B. (2013). Impact of regularization on spectral clustering. Preprint. Available at arXiv:1312.1733.

[20] KARRER, B. and NEWMAN, M. E. J. (2011). Stochastic blockmodels and community structure in networks. *Phys. Rev. E* (3) **83** 016107, 10. MR2788206

[21] LE, C. M., LEVINA, E. and VERSHYNIN, R. (2015). Sparse random graphs: Regularization and concentration of the Laplacian. Preprint. Available at arXiv:1502.03049.

[22] LE, C. M., LEVINA, E. and VERSHYNIN, R. (2015). Supplement to "Optimization via low-rank approximation for community detection in networks." DOI:10.1214/15-AOS1360SUPP.

[23] LEI, J. and RINALDO, A. (2015). Consistency of spectral clustering in stochastic block models. *Ann. Statist.* **43** 215–237. MR3285605

[24] LUSSEAU, D. and NEWMAN, M. E. J. (2004). Identifying the role that animals play in their social networks. *Proc. R. Soc. London B* (*Suppl.*) **271** S477–S481.

[25] LUSSEAU, D., SCHNEIDER, K., BOISSEAU, O. J., HAASE, P., SLOOTEN, E. and DAWSON, S. M. (2003). The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations: Can geographic isolation explain this unique trait? *Behavioral Ecology and Sociobiology* **54** 396–405.

[26] MARIADASSOU, M., ROBIN, S. and VACHER, C. (2010). Uncovering latent structure in valued graphs: A variational approach. *Ann. Appl. Stat.* **4** 715–742. MR2758646

[27] MASSOULIÉ, L. (2014). Community detection thresholds and the weak Ramanujan property. In *STOC'*14*—Proceedings of the* 46*th Annual ACM Symposium on Theory of Computing* 694–703. ACM, New York. MR3238997

[28] MIHAIL, M. and PAPADIMITRIOU, C. (2002). On the eigenvalue power law. In *RANDOM'*02*—Proceedings of the* 6*th International Workshop on Randomization and Approximation Techniques* 254–262. Springer, London. MR2047034

[29] MOSSEL, E., NEEMAN, J. and SLY, A. (2012). Stochastic block models and reconstruction. Preprint. Available at arXiv:1202.1499.

[30] MOSSEL, E., NEEMAN, J. and SLY, A. (2014). Belief propagation, robust reconstruction, and optimal recovery of block models. *COLT* **35** 356–370.

[31] MOSSEL, E., NEEMAN, J. and SLY, A. (2014). A proof of the block model threshold conjecture. Preprint. Available at arXiv:1311.4115.

[32] NEWMAN, M. E. J. (2013). Spectral methods for network community detection and graph partitioning. *Phys. Rev. E* **88** 042822.

[33] NEWMAN, M. E. J. and GIRVAN, M. (2004). Finding and evaluating community structure in networks. *Phys. Rev. E* **69** 026113.

[34] NEWMAN, M. E. J. (2006). Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E* (3) **74** 036104, 19. MR2282139

[35] NG, A., JORDAN, M. and WEISS, Y. (2001). On spectral clustering: Analysis and an algorithm. In *Neural Information Processing Systems* (T. Dietterich, S. Becker and Z. Ghahramani, eds.) **14** 849–856. MIT Press, Cambridge, MA.

[36] NOWICKI, K. and SNIJDERS, T. A. B. (2001). Estimation and prediction for stochastic blockstructures. *J. Amer. Statist. Assoc.* **96** 1077–1087. MR1947255

[37] O'ROURKE, S., VU, V. and WANG, K. (2013). Random perturbation of low rank matrices: Improving classical bounds. Preprint. Available at arXiv:1311.2657.

[38] QIN, T. and ROHE, K. (2013). Regularized spectral clustering under the degree-corrected stochastic blockmodel. In *NIPS'*13*—Advances in Neural Information Processing Systems* 26 3120–3128. Curran Associates, Red Hook, NY.

[39] RIOLO, M. and NEWMAN, M. E. J. (2014). First-principles multiway spectral partitioning of graphs. *Journal of Complex Networks* **2** 121–140.

[40] ROHE, K., CHATTERJEE, S. and YU, B. (2011). Spectral clustering and the high-dimensional stochastic blockmodel. *Ann. Statist.* **39** 1878–1915. MR2893856

[41] SARKAR, P. and BICKEL, P. J. (2015). Role of normalization in spectral clustering for stochastic blockmodels. *Ann. Statist.* **43** 962–990. MR3346694

[42] SNIJDERS, T. A. B. and NOWICKI, K. (1997). Estimation and prediction for stochastic blockmodels for graphs with latent block structure. *J. Classification* **14** 75–100. MR1449742

[43] WEIBEL, C. (2010). Implementation and parallelization of a reverse-search algorithm for Minkowski sums. In *ALENEX'*10*—Proceedings of the* 12*th Workshop on Algorithm Engineering and Experiments* 34–42. SIAM, Philadelphia, PA.

[44] YAO, Y. Y. (2003). Information-theoretic measures for knowledge discovery and data mining. In *Entropy Measures*, *Maximum Entropy Principle and Emerging Applications* 115–136. Springer, Berlin.

[45] ZHANG, Y., LEVINA, E. and ZHU, J. (2014). Detecting overlapping communities in networks using spectral methods. Preprint. Available at arXiv:1412.3432.

[46] ZHAO, Y., LEVINA, E. and ZHU, J. (2011). Community extraction for social networks. *Proc. Natl. Acad. Sci. USA* **108** 7321–7326.

[47] ZHAO, Y., LEVINA, E. and ZHU, J. (2012). Consistency of community detection in networks under degree-corrected stochastic block models. *Ann. Statist.* **40** 2266–2292. MR3059083

C. M. LE
E. LEVINA
DEPARTMENT OF STATISTICS
UNIVERSITY OF MICHIGAN
ANN ARBOR, MICHIGAN 48109-1107
USA
E-MAIL: canle@umich.edu
elevina@umich.edu

R. VERSHYNIN
DEPARTMENT OF MATHEMATICS
UNIVERSITY OF MICHIGAN
ANN ARBOR, MICHIGAN 48109-1043
USA
E-MAIL: romanv@umich.edu