Estimating Community Structure in Networks by Spectral Methods

by

Can M. Le

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Statistics)
in the University of Michigan
2016

Doctoral Committee:

Professor Elizaveta Levina, Co-Chair
Professor Roman Vershynin, Co-Chair
Assistant Professor Raj Rao Nadakuditi
Professor Ji Zhu

Acknowledgments

This dissertation and my graduate experience turned out better than I could have hoped for, and that is owing to many people. First, I would like to thank my advisors, Liza Levina and Roman Vershynin, who I was very fortunate to have as advisors. Liza was very patient and tolerant in early days when I knew very little about doing research in statistics. Her invaluable expertise and guidance helped me tremendously in forming meaningful research problems, developing statistical methods to solve them, as well as presenting results clearly and concisely. The support and encouragement from Liza were vital for me to carry on my research at difficult moments. I was also greatly influenced by Roman, who was very generous with his time, advice, and bold ideas. His expertise in probability helped me overcome many technical difficulties. Roman's clear way of thinking and teaching also set an example for me to follow.

I thank my committee members, Raj Rao Nadakuditi and Ji Zhu, for checking my thesis and invaluable comments. I am also grateful to Ji for being a committee member of my Preliminary Examination and for valued advice and help throughout the years. I thank all members of Liza and Ji's research group for inspiring presentations and discussions.

I thank my friends for sharing memorable moments with me in Ann Arbor. They helped me stay focused and sane over these challenging years. Finally, I thank my immediate and extended family for their unconditional love and support. Without them, none of this would have been possible.

Table of Contents

List of Figures

List of Tables

Abstract

Estimating Community Structure in Networks by Spectral Methods

Can M. Le

Dissertation advisors: Elizaveta Levina, Roman Vershynin

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 is the problem of estimating community structure. Most of existing methods rely on maximizing a criterion over the discrete set of community-label vectors. They require a good initial estimate of communities, which is often found by spectral clustering. Several problems arise from this approach: it has been empirically observed and theoretically predicted that spectral clustering fails when the network is sparse; solving an optimization problem over a discrete set of label vectors is a challenging task; and the number of communities is often unknown in practice. This dissertation contributes to progress on each of these problems.

We study random graphs with possibly different edge probabilities in the challenging sparse regime of bounded expected degrees. Unlike in the dense case, neither the network adjacency matrix nor its Laplacian concentrate around their expectations due to the highly irregular distribution of node degrees. We prove that simple regularization procedures force the adjacency matrix and the Laplacian to concentrate. As an immediate consequence, we establish the validity of regularized spectral clustering for estimating com-

munity structure.

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 estimation criteria, establishing its consistency for label estimation in each case.

We propose a simple spectral method for estimating the number of communities. We show that the method performs well under several models and a wide range of parameters, and is guaranteed to be consistent under several asymptotic regimes. We compare the new method to several existing methods for estimating the number of communities and show that it is both more accurate and more computationally efficient.

Chapter 1
Introduction

Network analysis has become an important area in many research domains, including social psychology, sociology, physics, computer science, probability, and statistics. A common way to study real-world networks is to model them as random graphs whose structure is encoded in the expectation matrix. In this thesis we will investigate the behavior of such random networks, develop methods to estimate the underlying structure, and apply those methods to real-world data.

One important structure of interest in network analysis is the community structure, where nodes are divided into groups (communities) which share similar connectivity patterns. Networks with communities are often modeled by the stochastic block model (SBM) [31] or the degree-corrected stochastic block model (DCSBM) [34]. Under the SBM, the label vector $\boldsymbol{c}$ is assumed to be drawn from a multinomial distribution with parameter $\boldsymbol{\pi} = \{\pi_1, \ldots, \pi_K\}$, where $0 \leq \pi_k \leq 1$ 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, labels are the only node information affecting edges between nodes, and all the nodes within the same community are stochastically equivalent. The DCSBM is a generalization of the SBM which allows for heterogeneity of node degrees, adding node-level parameters controlling a node's overall level of connectivity. Specifically, under DCSBM, $P(A_{ij} = 1) = \theta_i \theta_j P_{c_i c_j}$, where θ_i's are "degree parameters" satisfying some identifiability constraints. A much more general model for a random network is the inhomogeneous Erdös-Rényi model (IERM) in which edges are generated independently and each edge is allowed to have an arbitrary probability [11].

Community detection is the problem of estimating communities from a single observed network, usually encoded by the adjacency matrix A. Here A is a symmetric matrix with entries $A_{ij} = 1$ if there is an edge between nodes i, j and $A_{ij} = 0$ otherwise. Most of existing community detection methods rely on maximizing a criterion $f(A, e)$, such as the likelihood of the SBM or a modularity derived from a heuristic consideration, over the discrete set of community-label vectors e. While the optimization problem is usually NP-hard, an approximate solution can be often computed by MCMC, variational or pseudo-likelihood methods. These methods require a good initial estimate of communities, which is often found by spectral clustering. Spectral clustering takes leading eigenvectors of A or its graph Laplacian $L(A) = D^{-1/2}AD^{-1/2}$, where $D = \mathrm{diag}(d_i)$ is the diagonal matrix of degrees, as input and uses K-means to cluster them into a given number of groups.

Several problems arise from this approach. First, it has been empirically observed and theoretically predicted that spectral clustering fails when the network is sparse. Second, solving an optimization problem over a discrete set of label vectors is a challenging task. Third, the number of communities is often unknown in practice. This thesis contributes to progress on each of these problems. In Chapter 2 we prove that for a sparse network, a simple regularization significantly improves the performance of spectral clustering; in Chapter 3 we develop a general method for solving such optimization problems and show that many existing criteria can be solved efficiently by our method [41]; and in Chapter 4 we propose a fast and reliable method for estimating the number of communities [38]. While this work has been motivated by community detection, several results are very general and their applications may go far beyond the community detection setting.

1.1 Concentration of sparse networks

Since network structure, including communities, is encoded in the expectation of the adjacency matrix A or the Laplacian $L(A)$, it is very important to understand the deviation of those matrices from their expectations.

Concentration of *dense* random networks, where expected degrees grow at least as fast as $\log n$ (n denotes the network size), is well understood. In this

regime, Oliveira [63] showed that both the adjacency matrix and the Laplacian generated from the IERM concentrate. Consequently, under the SBM, their leading eigenvectors also concentrate according to Davis-Kahan theorem [19], and therefore spectral clustering correctly recovers the communities up to a vanishing fraction of mis-clustered nodes.

In contrast, for *sparse* random networks, those with bounded expected degrees, neither the adjacency matrix nor the Laplacian concentrate due to the high variance of the degree distribution [21]. The existence of isolated nodes also implies that there is always a non-vanishing fraction of nodes that no algorithm can correctly cluster.

Since the concentration of sparse random networks fails because the degree distribution is too irregular, we may naturally ask if *regularizing* the network in some way solves the problem. One simple way to deal with very low-degree nodes, proposed by [5] and analyzed by [33], is to add the same small positive number τ/n to all entries of the adjacency matrix A. That is, we replace A with $A_\tau := A + \tau/n\mathbf{1}\mathbf{1}^\mathsf{T}$ and use the Laplacian of A_τ instead. Another way to deal with low degree nodes, proposed by [13] and studied theoretically by [69], is to add a constant τ directly to the diagonal of D in the definition of the Laplacian. For both ways of regularization, the concentration which implies the consistency of spectral clustering in estimating communities was obtained in [69, 33], but only for *dense* networks.

In Chapter 2 we showed that for *sparse* random networks generated from the IERM, both ways of regularization described above lead to the concentration of the Laplacian. Consequently, under the SBM the spectral clustering correctly recovers the communities up to a small fraction of mis-clustered nodes. This is the first result showing that the spectral clustering can find communities in the *sparse* regime.

Our proof relies on the concentration of the adjacency matrix in *cut norm* (it has been popular in theoretical computer science community), the use of Grothendieck's factorization theorem and a paving argument. It provides a better understanding of the behavior of sparse random networks. Namely, their failure to concentrate is caused by a small fraction of irregular nodes, which is inversely proportional to the average expected degree; on the rest

of the nodes, both the adjacency matrix and the Laplacian concentrate even without regularization.

To deal with very high-degree nodes, we propose a general procedure to reduce the total weight of their incident edges. This includes removing all high-degree nodes, or removing just enough edges from high-degree nodes, or reducing weight of edges of high-degree nodes. We show that this procedure indeed forces the adjacency matrix of sparse random networks to concentrate around its expectation.

1.2 Community detection via optimization

Community detection is the problem of estimating communities from a single observed network. Roughly speaking, the large recent literature on community detection in this scenario has followed one of two tracks: fitting probabilistic models for the adjacency matrix, or optimizing global criteria derived from other considerations over label assignments $\boldsymbol{c}$, often via spectral approximations. 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 [76, 62, 49], as well as maximizing profile likelihoods by some type of greedy label-switching algorithms. The profile likelihood was derived for the SBM by [10] and for the DCSBM by [34], but the label-switching greedy search algorithms only scale up to a few thousand nodes. A much faster pseudo-likelihood algorithm was proposed by [4] for fitting both these models. Another fast algorithm for the block model based on belief propagation has been proposed by [20]. 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, e.g. the Newman-Girvan modularity [60, 57] and the community extraction criterion [88], have been proposed for this purpose, often not tied to a generative network model. 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, for example, [61]), a generic clustering method which became popular for community detection. In this context, the method has been analyzed by [72, 13, 71, 45], among others, while [32] proposed a spectral method specifically for the DCSBM.

Existing methods for community detection are either slow (using MCMC or variational methods) or depend heavily on the particular form of the criteria and are not easily generalizable (pseudo-likelihood methods). We develop a new general method for solving a class of optimization problems and show that many existing criteria $f(A, e)$ for community detection can be solved efficiently by our method.

The main idea is to reduce the set of K^n community label vectors to a much smaller set over which the optimization problem becomes easy (K denotes the number of communities). To that end, we first note that under the SBM or the DCSBM the expectation $\mathbb{E}\, A$ of the adjacency matrix A has a block structure; its rank is the number of communities, which is often small. Under some mild conditions, A concentrates around $\mathbb{E}\, A$, therefore it is essentially also a low-rank matrix. For a natural class of functions f, which contains many existing community detection criteria, $f(A, e)$ is essentially a function of projections $P(e)$ of community-label vectors e onto the subspace spanned by a few principle components of A. In Chapter 3 we show that this function achieves its maximum at extreme points of the convex hull of the projections $P(e)$. Since the projector P is a low-rank matrix, most of the community-label vectors e become interior points after the projection. Therefore we can find the maximum of $f(A, e)$ simply by performing an exhaustive search over the community-label vectors corresponding to the extreme points.

The set of extreme points can be computed by an existing reverse-search algorithm [28, 84]. Its cardinality is at most polynomial in n; in particular, when we are looking to divide the network into two communities, the number of extreme points is at most $2n$ and they can be found in $O(n \log n)$ operations.

1.3 Estimating the number of communities

Most of existing methods for community detection, including spectral clustering, require the number of communities K as input, but in practice K is often unknown. To address this problem, a few methods have been proposed to estimate K, under either the SBM or the DCSBM. All these methods are either restricted to a specific model or computationally intensive; they require either computing the likelihood function, done by the variational method [83], or a computationally challenging procedure, e.g. bootstrap or cross-validation [9, 14].

We propose a fast and reliable method for estimating K that uses spectral properties of the Bethe Hessian and the non-backtracking matrices. This is inspired by [36, 73, 12], where these matrices were used to recover the community structure under a simple SBM in the sparse regime. In Chapter 4 we show that a simple count of leading eigenvalues of these matrices directly estimates the number of communities, and the estimate performs well under different network models and over a wide range of parameters, outperforming existing methods that are designed specifically for finding K under either SBM or DCSBM. This method does not need any tuning parameters and is very computationally efficient, since all it requires is computing a few leading eigenvalues of just one typically sparse matrix.

We show that our estimate is consistent in either *sparse* regime of bounded degrees or in a regime when the average expected degree grows sufficiently fast. More work is needed on the case of "intermediate" rate of average expected degree not covered by our result, which will require fundamentally different approaches.

Chapter 2
Concentration and regularization of random graphs

2.1 Introduction

This chapter studies concentration properties of random graphs. To do this, it will be useful to look at the graph G through the lens of the matrices classically associated with G, namely the adjacency and Laplacian matrices.

Let us first build the theory for the adjacency matrix A; the Laplacian will be discussed in Section 2.1.5. We say that G concentrates about its expectation if A is close to its expectation $\mathbb{E}\,A$ in the spectral norm; we interpret the expectation of G as the weighted graph with adjacency matrix $\mathbb{E}\,A$. Concentration of random graphs interpreted this way, and also of general random matrices, has been studied in several communities, in particular in radom matrix theory, combinatorics and network science. It automatically gives us a tight control of eigenvalues and eigenvectors according to Davis-Kahan theorem [19].

We will study random graphs generated from an *inhomogeneous Erdös-Rényi model* $G(n, (p_{ij}))$, where edges are formed independently with given probabilities p_{ij}, see [11]. This is a generalization of the classical Erdös-Rényi model $G(n, p)$ where all edge probabilities p_{ij} equal p. Many popular graph models arise as special cases of $G(n, (p_{ij}))$, such as the stochastic block model, a benchmark model in the analysis of networks [31] discussed in Section 2.1.7, and random subgraphs of given graphs.

2.1.1 Dense graphs concentrate

The cleanest concentration results are available for the classical Erdös-Rényi model $G(n,p)$ in the *dense* regime. In terms of the expected degree $d = pn$, we have with high probability that

$$\|A - \mathbb{E}\, A\| = 2\sqrt{d}\,(1 + o(1)) \quad \text{if} \quad d \gg \log^4 n, \tag{2.1.1}$$

see [22, 81, 46]. Since $\|\mathbb{E}\, A\| = d$, we see that the typical deviation here behaves like the square root of the magnitude of expectation – just like in many other classical results of probability theory. In other words, *dense random graphs concentrate well*.

The lower bound on density in (2.1.1) can be essentially relaxed all the way down to $d = \Omega(\log n)$. Thus, with high probability we have

$$\|A - \mathbb{E}\, A\| = O(\sqrt{d}) \quad \text{if} \quad d = \Omega(\log n). \tag{2.1.2}$$

More generally, (2.1.2) holds for $G(n, (p_{ij}))$ with a somewhat larger but still useful value

$$d = \max_{ij} np_{ij}, \tag{2.1.3}$$

see [21, 44, 15]. Our main interest in this chapter is the sparse regime when $d = \Omega(\log n)$ no longer holds.

2.1.2 Sparse graphs do not concentrate

In the *sparse* regime, where the expected degree d is bounded, concentration breaks down. According to [35], a random graph from $G(n,p)$ satisfies with high probability that

$$\|A\| = (1 + o(1))\sqrt{d(A)} = (1 + o(1))\sqrt{\frac{\log n}{\log\log n}} \quad \text{if} \quad d = O(1), \tag{2.1.4}$$

where $d(A)$ denotes the maximal degree of the graph (a random quantity). So in this regime we have $\|A\| \gg \|\mathbb{E}\, A\| = d$, which shows that *sparse random graphs do not concentrate*.

What exactly makes the norm A abnormally large in the sparse regime? The answer is, the vertices with too high degrees. In the dense case where $d \gg$

$\log n$, all vertices typically have approximately the same degrees $(1 + o(1))d$. This no longer happens in the sparser regime $d \ll \log n$; the degrees do not cluster tightly about the same value anymore. There are vertices with too high degrees; they are captured by the second inequality in (2.1.4). Even a single high-degree vertex can blow up the norm of the adjacency matrix. Indeed, since the norm of A is bounded below by the Euclidean norm of each of its rows, we have $\|A\| \geq \sqrt{d(A)}$.

2.1.3 Regularization enforces concentration

If high-degree vertices destroy concentration, can we "tame" these vertices? One proposal would be to remove these vertices from the graph altogether. U. Feige and E. Ofek [21] showed that this works for $G(n, p)$ – *the removal of the high degree vertices enforces concentration.* Indeed, if we drop all vertices with degrees, say, larger than $2d$, the the remaining part of the graph satisfies

$$\|A' - \mathbb{E}\, A'\| = O(\sqrt{d}) \tag{2.1.5}$$

with high probability, where A' denotes the adjacency matrix of the new graph. The argument in [21] is based on the method developed by J. Kahn and E. Szemeredi [23]. It extends to the inhomogeneous Erdös-Rényi model $G(n, (p_{ij}))$ with d defined in (2.1.3), see [44, 15]. As we will see, our paper provides an alternative and completely different approach to such results.

Although the removal of high degree vertices solves the concentration problem, such solution is not ideal, since those vertices are in some sense the most important ones. In real-world networks, the vertices with highest degrees are "hubs" that hold the network together. Their removal would cause the network to break down into disconnected components, which leads to a considerable loss of structural information.

Would it be possible to regularize the graph in a more gentle way – instead of removing the high-degree vertices, reduce the weights of their edges just enough to keep the degrees bounded by $O(d)$? The main result of our paper states that this is true. Let us first state this result informally; Theorem 2.2.1 provides a more general and formal statement.

Theorem 2.1.1 (Concentration of regularized adjacency matrices)**.** *Consider a random graph from the inhomogeneous Erdös-Rényi model, and let d be as in (2.1.3). For all high degree vertices of the graph (say, those with degrees larger than $2d$), reduce the weights of the edges incident to them in an arbitrary way, but so that all degrees of the new (weighted) graph become bounded by $2d$. Then, with high probability, the adjacency matrix A' of the new graph concentrates:*

$$\|A' - \mathbb{E}\, A\| = O(\sqrt{d}).$$

Moreover, instead of requiring that the degrees become bounded by $2d$, we can require that the ℓ_2 norms of the rows of the new adjacency matrix become bounded by $\sqrt{2d}$.

2.1.4 Examples of graph regularization

The regularization procedure in Theorem 2.1.1 is very flexible. Depending on how one chooses the weights, one can obtain as partial cases several results we summarized earlier, as well as some new ones.

1. *Do not do anything to the graph.* In the dense regime where $d = \Omega(\log n)$, all degrees are already bounded by $2d$ with high probability. This means that the original graph satisfies $\|A - \mathbb{E}\, A\| = O(\sqrt{d})$. Thus we recover the result of U. Feige and E. Ofek (2.1.2), which states that *dense random graphs concentrate well.*

2. *Remove all high degree vertices.* If we remove all vertices with degrees larger than $2d$, we recover another result of U. Feige and E. Ofek (2.1.5), which states that *the removal of the high degree vertices enforces concentration.*

3. *Remove just enough edges from high-degree vertices.* Instead of removing the high-degree vertices with all of their edges, we can remove just enough edges to make all degrees bounded by $2d$. This milder regularization still produces the concentration bound (2.1.5).

4. *Reduce the weight of edges proportionally to the excess of degrees.* Instead of removing edges, we can reduce the weight of the existing edges, a procedure which better preserves the structure of the graph. For instance, we

can assign weight $\sqrt{\lambda_i \lambda_j}$ to the edge between vertices i and j, choosing $\lambda_i := \min(2d/d_i, 1)$ where d_i is the degree of vertex i. One can check that this makes the ℓ_2 norms of all rows of the adjacency matrix bounded by $2d$. By Theorem 2.1.1, such regularization procedure leads to the same concentration bound (2.1.5).

2.1.5 Concentration of Laplacian

So far, we have looked at random graphs through the lens of their adjacency matrices. A different matrix that captures the geometry of a graph is the (symmmetric, normalized) Laplacian matrix, defined as

$$\mathcal{L}(A) = D^{-1/2}(D - A)D^{-1/2} = I - D^{-1/2}AD^{-1/2}. \tag{2.1.6}$$

Here I is the identity matrix and $D = \mathrm{diag}(d_i)$ is the diagonal matrix with degrees $d_i = \sum_{j=1}^{n} A_{ij}$ on the diagonal. The reader is referred to [17] for an introduction to graph Laplacians and their role in spectral graph theory. Here we mention just two basic facts: the spectrum of $\mathcal{L}(A)$ is a subset of $[0, 2]$, and the smallest eigenvalue is always zero.

Concentration of Laplacians of random graphs has been studied in [63, 13, 69, 33, 39, 25]. Just like the adjacency matrix, the Laplacian is known to concentrate in the dense regime where $d = \Omega(\log n)$, and it fails to concentrate in the sparse regime. However, the obstructions to concentration are opposite. For the adjacency matrices, as we mentioned, the trouble is caused by high-degree vertices. For the Laplacian, the problem lies with *low-degree vertices*. In particular, for $d = o(\log n)$ the graph is likely to have isolated vertices; they produce multiple zero eigenvalues of $\mathcal{L}(A)$, which are easily seen to destroy the concentration.

In analogy to our discussion of adjacency matrices, we can try to regularize the graph to "tame" the low-degree vertices in various ways, for example remove the low-degree vertices, connect them to some other vertices, artificially increase the degrees d_i in the definition (2.1.6) of Laplacian, and so on. Here we will focus on the following simple way of regularization proposed in [5] and analyzed in [33, 39, 25]. Choose $\tau > 0$ and add the same number τ/n

to all entries of the adjacency matrix A, thereby replacing it with

$$A_\tau := A + (\tau/n)\mathbf{1}\mathbf{1}^\mathsf{T}$$

in the definition (2.1.6) of the Laplacian. This regularization raises all degrees d_i to $d_i + \tau$. If we choose $\tau \sim d$, the regularized graph does not have low-degree vertices anymore.

The following consequence of Theorem 2.1.1 shows that such regularization indeed forces Laplacian to concentrate. Here we state this result informally; Theorem 2.4.1 provides a more formal statement.

Theorem 2.1.2 (Concentration of the regularized Laplacian)**.** *Consider a random graph from the inhomogeneous Erdös-Rényi model, and let d be as in (2.1.3). Choose a number $\tau \sim d$. Then, with high probability, the regularized Laplacian $\mathcal{L}(A_\tau)$ concentrates:*

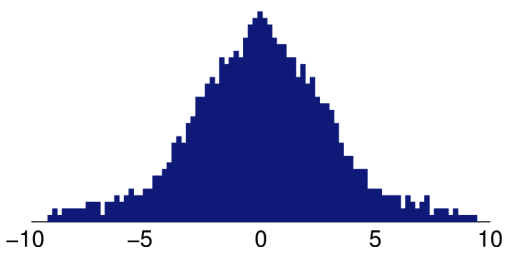
$$\|\mathcal{L}(A_\tau) - \mathcal{L}(\mathbb{E}\, A_\tau)\| = O\Big(\frac{1}{\sqrt{d}}\Big).$$

We will deduce this result from Theorem 2.1.1 in Section 2.4. Theorem 2.1.2 is an improvement upon a bound in [39] that had an extra $\log^3(d)$ factor. The exponent 3 was reduced to $1/2$ in [25], and it was conjectured there that the logarithmic factor is not needed at all. Theorem 2.1.2 confirms this conjecture.

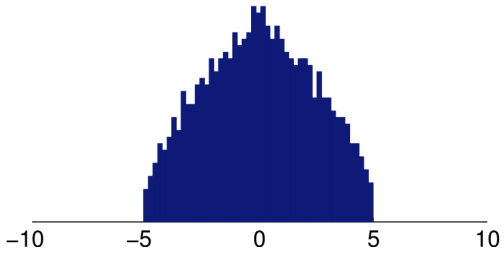
2.1.6 A numerical experiment

To conclude our discussion of various ways to regularize sparse graphs, let us illustrate the effect of regularization by a numerical experiment. Figure 2.1a shows the histogram of the spectrum of A for a sparse random graph.[1] The high degree vertices generate the outliers of the spectrum, which appear as two "tails" in the histogram. Regularization successfully removes those outliers; Figure 2.1b shows the histogram of the spectrum of A'. Thus from the statistical viewpoint, regularization acts as *shrinkage* of the parasitic outliers of spectrum toward the bulk.

[1] We removed one leading eigenvalue of order d from these figures. In other words, we plot the spectrum of $A - \mathbb{E}\, A$.



(a) Spectrum before regularization



(b) Spectrum after regularization

2.1.7 Application: community detection in networks

Among many possible applications of concentration of random graphs, let us mention a well understood connection to the analysis of networks. A benchmark model of networks with communities is the so-called the *stochastic block model* $G(n, \frac{a}{n}, \frac{b}{n})$ [31]. This is a partial case of the inhomogeneous Erdös-Rényi model considered in this paper, and it is defined as follows. The set of vertices is divided into two subsets (communities) of size $n/2$ each. Edges between vertices are drawn independently with probability a/n if they are in the same community and with probability b/n otherwise. The *community detection problem* is to detect which vertices belong to which communities as accurately as possible.

The most basic and popular algorithm proposed for community detection is *spectral clustering* [3, 51, 72, 13, 59, 44, 69, 82]. It works as follows: compute the eigenvector $v_2(A)$ corresponding to the second largest eigenvalue of the adjacency matrix A (or the Laplacian matrix); then classify the vertices based on the signs of the coefficients of $v_2(A)$. If this vector is positive on vertex i put i in the first community; otherwise put it in the second.

The success of the spectral clustering hinges upon concentration of random graphs. If concentration does hold and A is close to $\mathbb{E}\, A$, then the standard perturbation theory (Davis-Kahan theorem) shows that $v_2(A)$ must be close to $v_2(\mathbb{E}\, A)$. In particular, the signs of these vectors must agree on most of the vertices. But an easy calculation shows that the signs of $v_2(\mathbb{E}\, A)$ detect the communities exactly: this vector is a positive constant on one community and negative constant on the other. Therefore, $v_2(A)$ must detect the communities up to a small fraction of misclassified vertices.

Working out the details, one can conclude that *regularized spectral clus-*

tering (i.e. the spectral clustering applied to the graph reguralized in one of the ways described in Section 2.1.4) recovers the communities up to an ε fraction of misclassified vertices as long as

$$(a-b)^2 > C_\varepsilon (a+b), \tag{2.1.7}$$

where $C_\varepsilon = C/\varepsilon$ for some constant $C > 0$. The deduction of this from concentration is standard; the reader can refer e.g. to [39, 15].

In conclusion let us mention that condition (2.1.7) appeared in the analysis of other community detection algorithms, see [29, 15, 25]. It is tight up to the constant C_ε that must go to infinity with $\varepsilon \to 0$ [87]. In fact, the necessary and sufficient condition for recovering the two communities better than random guessing is $(a-b)^2 > 2(a+b)$ [55, 56, 54, 50].

2.2 Full version of Theorem 2.1.1, and reduction to a graph decomposition

In this section we state a more general and quantitative version of Theorem 2.1.1, and we reduce it to a new form of graph decomposition, which can be of interest on its own.

Theorem 2.2.1 (Concentration of regularized adjacency matrices)**.** *Consider a random graph from the inhomogeneous Erdös-Rényi model, and let d be as in (2.1.3). For any $r \geq 1$, the following holds with probability at least $1 - n^{-r}$. Consider any subset consisting of at most $10n/d$ vertices, and reduce the weights of the edges incident to those vertices in an arbitrary way. Then the adjacency matrix A' of the new (weighted) graph satisfies*

$$\|A' - \mathbb{E}\, A\| = Cr^{3/2}\big(\sqrt{d} + \sqrt{d'}\big).$$

Here d' denotes the degree of the new graph. Moreover, the same bound holds for d' being the maximal ℓ_2 norm of the rows of A'.

In this result and in the rest of the paper, $C, C_1, C_2, \dots$ denote absolute constants whose values may be different from line to line.

Remark 2.2.2 (Theorem 2.2.1 implies Theorem 2.1.1). The subset of $10n/d$ vertices in Theorem 2.2.1 can be completely arbitrary. So let us choose the high-degree vertices, say those with degrees larger than $2d$. There are at most $10n/d$ such vertices with high probability; this follows by an easy calculation, and also from Lemma 2.3.5. Thus we immediately deduce Theorem 2.1.1.

One may wonder if Theorem 2.2.1 can be proved by developing an ϵ-net argument similar to the method of J. Kahn and E. Szemeredi [23] and its versions [3, 21, 44, 15]. Although we can not rule out such possibility, we do not see how this method could handle a general regularization. The reader familiar with the method can easily notice an obstacle. The contribution of the so-called light couples becomes hard to control when one changes, and even reduces, the individual entries of A (the weights of edges).

We will develop an alternative and somewhat simpler approach, which will be able to handle a general regularization of random graphs. Our method is a development (and a considerable simplification) of the idea in [39]. It sheds light on the specific structure of graphs that enables concentration. We are going to identify this structure through a *graph decomposition* in the next section. But let us pause briefly to mention the following useful reduction.

Remark 2.2.3 (Reduction to directed graphs). Our arguments will be more convenient to carry out if the adjacency matrix A has all independent entries. To be able to make this assumption, we can decompose A into the upper-triangular and a lower-triangular parts, both of which have independent entries. If we can show that each of these parts concentrate about its expectation, it would follow that A concentrate about $\mathbb{E}\, A$ by triangle inequality.

In other words, we may prove Theorem 2.2.1 for *directed* inhomogeneous Erdös-Rényi graphs, where edges between any vertices and in any direction appear indepednently with probabilities p_{ij}. In the rest of the argument, we will only work with such random directed graphs.

2.2.1 Graph decomposition

In this section, we reduce Theorem 2.2.1 to the following decomposition of inhomogeneous Erdös-Rényi directed random graphs. This decomposition

may have an independent interest. Throughout the paper, we denote by $B_{\mathcal{N}}$ the matrix which coincides with a matrix B on a subset of edges $\mathcal{N} \subset [n] \times [n]$ and has zero entries elsewhere.

Theorem 2.2.4 (Graph decomposition)**.** *Consider a random directed graph from the inhomogeneous Erdös-Rényi model, and let d be as in (2.1.3). For any $r \geq 1$, the following holds with probability at least $1 - 3n^{-r}$. One can decompose the set of edges $[n] \times [n]$ into three classes $\mathcal{N}$, $\mathcal{R}$ and $\mathcal{C}$ so that the following properties are satisfied for the adjacency matrix A.*

- *The graph concentrates on $\mathcal{N}$, namely $\|(A - \mathbb{E}\, A)_{\mathcal{N}}\| \leq Cr^{3/2}\sqrt{d}$.*
- *Each row of $A_{\mathcal{R}}$ and each column of $A_{\mathcal{C}}$ has at most $32r$ ones.*

Moreover, $\mathcal{R}$ intersects at most n/d columns, and $\mathcal{C}$ intersects at most n/d rows of $[n] \times [n]$.

Figure 2.2 illustrates a possible decomposition Theorem 2.2.4 can provide. The edges in $\mathcal{N}$ form a big "core" where the graph concentrates well even without regularization. The edges in $\mathcal{R}$ and $\mathcal{C}$ can be thought of (at least heuristically) as those attached to high-degree vertices.

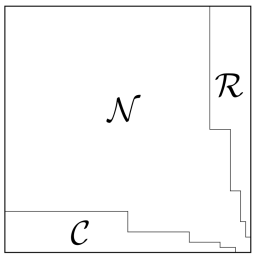


Figure 2.2: An example of graph decomposition in Theorem 2.2.4.

A weaker version of Theorem 2.2.4 was proved recently in [39], which had parasitic $\log(d)$ factors. It became possible to remove them here by developing a related but considerably different method, which is also considerably simpler than in [39]. The key difference is that instead of Grothendieck inequality, we will use here the Grothendieck-Pietsch factorization, which we will explain in detail in Section 2.3.2.

We will prove Theorem 2.2.4 in Section 2.3; let us pause to deduce Theorem 2.2.1 from it.

2.2.2 Deduction of Theorem 2.2.1

First, let us explain informally how the graph decomposition could lead to Theorem 2.2.1. The regularization of the graph does not destroy the properties of $\mathcal{N}$, $\mathcal{R}$ and $\mathcal{C}$ in Theorem 2.2.4. Moreover, regularization creates a new property for us, allowing for a good control of the *columns* of $\mathcal{R}$ and *rows* of $\mathcal{C}$. Let us focus on $A_{\mathcal{R}}$ to be specific. The ℓ_1 norms of all columns of this matrix are at most d', and the ℓ_1 norms of all rows are $O(1)$ by Theorem 2.2.4. By a simple calculation which we will do in Lemma 2.2.5, this implies that $\|A_{\mathcal{R}}\| = O(\sqrt{d'})$. A similar bound can be proved for $\mathcal{C}$. Combining $\mathcal{N}$, $\mathcal{R}$ and $\mathcal{C}$ together will lead to the error bound $O(\sqrt{d} + \sqrt{d'})$ in Theorem 2.2.1.

To make this argument rigorous, let us start with the simple calculation we just mentioned.

Lemma 2.2.5. *Consider a matrix B in which each row has ℓ_1 norm at most a, and each column has ℓ_1 norm at most b. Then $\|B\| \le \sqrt{ab}$.*

Proof. The claim follows directly from the Riesz-Thorin interpolation theorem (see e.g. [77, Theorem 2.1]), since the maximal ℓ_1 norm of columns is the $\ell_1 \to \ell_1$ operator norm, and the maximal ℓ_1 norm of rows is the $\ell_\infty \to \ell_\infty$ operator norm. For completeness, let us give here an alternative direct proof. Let x be a vector with $\|x\|_2 = 1$. Using Cauchy-Schwarz inequality and the assumptions, we have

$$\begin{aligned} \|Bx\|_2^2 &= \sum_i \Big(\sum_j B_{ij} x_j \Big)^2 \le \sum_i \Big(\sum_j |B_{ij}| \sum_j |B_{ij}| x_j^2 \Big) \\ &\le \sum_i \Big(a \sum_j |B_{ij}| x_j^2 \Big) = a \sum_j x_j^2 \sum_i |B_{ij}| \le a \sum_j x_j^2 b = ab. \end{aligned}$$

Since x is arbitrary, this completes the proof. □

We are ready to formally deduce the main part of Theorem 2.2.1 from Theorem 2.2.4; we defer the "moreover" part to Section 2.3.6.

Proof of Theorem 2.2.1 (main part). Fix a realization of the random graph that satisfies the conclusion of Theorem 2.2.4, and decompose the deviation

$A' - \mathbb{E}\, A$ as follows:

$$A' - \mathbb{E}\, A = (A' - \mathbb{E}\, A)_{\mathcal{N}} + (A' - \mathbb{E}\, A)_{\mathcal{R}} + (A' - \mathbb{E}\, A)_{\mathcal{C}}. \tag{2.2.1}$$

We will bound the spectral norm of each of the three terms separately.

Step 1. Deviation on $\mathcal{N}$. Let us further decompose

$$(A' - \mathbb{E}\, A)_{\mathcal{N}} = (A - \mathbb{E}\, A)_{\mathcal{N}} - (A - A')_{\mathcal{N}}. \tag{2.2.2}$$

By Theorem 2.2.4, $\|(A - \mathbb{E}\, A)_{\mathcal{N}}\| \le Cr^{3/2}\sqrt{d}$. To control the second term in (2.2.2), denote by $\mathcal{E} \subset [n] \times [n]$ the subset of edges we choose to reweight in Theorem 2.2.4. Since A and A' are equal on $\mathcal{E}^c$, we have

$$\begin{aligned} \|(A - A')_{\mathcal{N}}\| &= \|(A - A')_{\mathcal{N}\cap\mathcal{E}}\| \le \|A_{\mathcal{N}\cap\mathcal{E}}\| \qquad \text{(since } 0 \le A - A' \le A \text{ entrywise)} \\ &\le \|(A - \mathbb{E}\, A)_{\mathcal{N}\cap\mathcal{E}}\| + \|\mathbb{E}\, A_{\mathcal{N}\cap\mathcal{E}}\| \qquad \text{(by triangle inequality).} \end{aligned} \tag{2.2.3}$$

Further, a simple restriction property implies that

$$\|(A - \mathbb{E}\, A)_{\mathcal{N}\cap\mathcal{E}}\| \le 2\|(A - \mathbb{E}\, A)_{\mathcal{N}}\|. \tag{2.2.4}$$

Indeed, restricting a matrix onto a product subset of $[n] \times [n]$ can only reduce its norm. Although the set of reweighted edges $\mathcal{E}$ is not a product subset, it can be decomposed into two product subsets:

$$\mathcal{E} = \big(I \times [n]\big) \cup \big(I^c \times I\big) \tag{2.2.5}$$

where I is the subset of vertices incident to the edges in $\mathcal{E}$. Then (2.2.4) holds; right hand side of that inequality is bounded by $Cr^{3/2}\sqrt{d}$ by Theorem 2.2.4. Thus we handled the first term in (2.2.3).

To bound the second term in (2.2.3), we can use another restriction property that states that the norm of the matrix with non-negative entries can only reduce by restricting onto any subset of $[n] \times [n]$ (whether a product subset or not). This yields

$$\|\mathbb{E}\, A_{\mathcal{N}\cap\mathcal{E}}\| \le \|\mathbb{E}\, A_{\mathcal{E}}\| \le \|\mathbb{E}\, A_{I\times[n]}\| + \|\mathbb{E}\, A_{I^c\times I}\| \tag{2.2.6}$$

where the second inequality follows by (2.2.5). By assumption, the matrix $\mathbb{E}\, A_{I\times[n]}$ has $|I| \le 10n/d$ rows and each of its entries is bounded by d/n. Hence

the ℓ_1 norm of all rows is bounded by d, and the ℓ_1 norm of all columns is bounded by 10. Lemma 2.2.5 implies that $\|\mathbb{E}\, A_{I\times[n]}\| \le \sqrt{10d}$. A similar bound holds for the second term of (2.2.6). This yields

$$\|\mathbb{E}\, A_{\mathcal{N}\cap\mathcal{E}}\| \le 5\sqrt{d},$$

so we handled the second term in (2.2.3). Recalling that the first term there is bounded by $Cr^{3/2}\sqrt{d}$, we conclude that $\|(A - A')_{\mathcal{N}}\| \le 2Cr^{3/2}\sqrt{d}$.

Returning to (2.2.2), we recall that the first term in the right hand is bounded by $Cr^{3/2}\sqrt{d}$, and we just bounded the second term by $2Cr^{3/2}\sqrt{d}$. Hence

$$\|(A' - \mathbb{E}\, A)_{\mathcal{N}}\| \le 4Cr^{3/2}\sqrt{d}.$$

Step 2. Deviation on $\mathcal{R}$ and $\mathcal{C}$. By triangle inequality, we have

$$\|(A' - \mathbb{E}\, A)_{\mathcal{R}}\| \le \|A'_{\mathcal{R}}\| + \|\mathbb{E}\, A_{\mathcal{R}}\|.$$

Recall that $0 \le A'_{\mathcal{R}} \le A_{\mathcal{R}}$ entrywise. By Theorem 2.2.4, each of the rows of $A_{\mathcal{R}}$, and thus also of $A'_{\mathcal{R}}$, has ℓ_1 norm at most $32r$. Moreover, by definition of d', each of the columns of A', and thus also of $A'_{\mathcal{R}}$, has ℓ_1 norm at most d'. Lemma 2.2.5 implies that $\|A'_{\mathcal{R}}\| \le \sqrt{32rd'}$.

The matrix $\mathbb{E}\, A_{\mathcal{R}}$ can be handled similarly. By Theorem 2.2.4, it has at most n/d entries in each row, and all entries are bounded by d/n. Thus each column of $\mathbb{E}\, A_{\mathcal{R}}$ has ℓ_1 norm at most 1, and and each row has ℓ_1 norm at most d. Lemma 2.2.5 implies that $\|\mathbb{E}\, A_{\mathcal{R}}\| \le \sqrt{d}$.

We showed that

$$\|(A' - \mathbb{E}\, A)_{\mathcal{R}}\| \le \sqrt{32rd'} + \sqrt{d}.$$

A similar bound holds for $\|(A' - \mathbb{E}\, A)_{\mathcal{C}}\|$. Combining the bounds on the deviation of $A' - \mathbb{E}\, A$ on $\mathcal{N}$, $\mathcal{R}$ and $\mathcal{C}$ and putting them into (2.2.1), we conclude that

$$\|A' - \mathbb{E}\, A\| \le 4Cr^{3/2}\sqrt{d} + 2\big(\sqrt{32rd'} + \sqrt{d}\big).$$

Simplifying this inequality, we complete the proof of the main part of Theorem 2.2.1. □

2.3 Proof of Decomposition Theorem 2.2.4

2.3.1 Outline of the argument

We will construct the decomposition in Theorem 2.2.4 by an iterative procedure. The first and crucial step is to find a big block[2] $\mathcal{N}' \subset [n] \times [n]$ of size at least $(n - n/d) \times n/2$ on which A concentrates, i.e.

$$\|(A - \mathbb{E}\, A)_{\mathcal{N}'}\| = O(\sqrt{d}).$$

To find such block, we first establishing concentration in $\ell_\infty \to \ell_2$ norm; this can be done by standard probabilistic techniques. Next, we can automatically upgrade this to concentration in the spectral norm ($\ell_2 \to \ell_2$) once we pass to an appropriate block $\mathcal{N}'$. This can be done using a general result from functional analysis, which we call Grothendieck-Pietsch factorization.

Repeating this argument for the transpose, we obtain another block $\mathcal{N}''$ of size at least $n/2 \times (n - n/d)$ where the graph concentrates as well. So the graph concentrates on $\mathcal{N}_0 := \mathcal{N}' \cup \mathcal{N}''$. The "core" $\mathcal{N}_0$ will form the first part of the class $\mathcal{N}$ we are constructing.

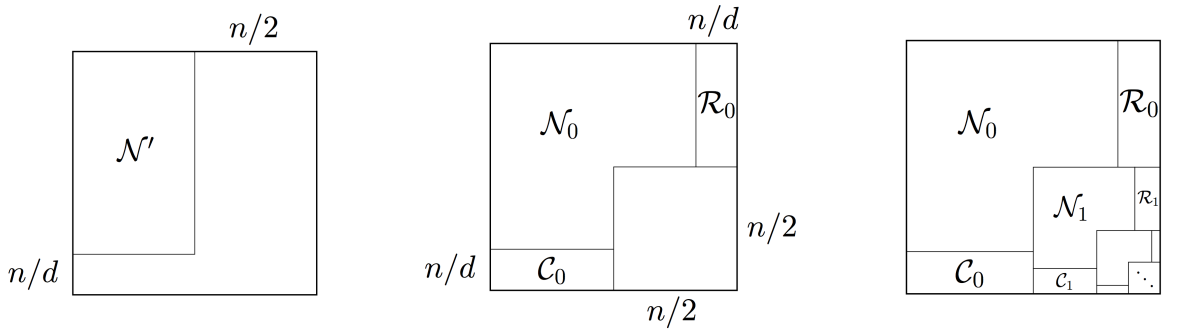
It remains to control the graph on the complement of $\mathcal{N}_0$. That set of edges is quite small; it can be described as a union of a block $\mathcal{C}_0$ with n/d rows, a block $\mathcal{R}_0$ with n/d columns and an exceptional $n/2 \times n/2$ block; see Figure 2.3b for illustration. We may consider $\mathcal{C}_0$ and $\mathcal{R}_0$ as the first parts of the future classes $\mathcal{C}$ and $\mathcal{R}$ we are constructing.

Indeed, since $\mathcal{C}_0$ has so few rows, the expected number of ones in each column of $\mathcal{C}_0$ is bounded by 1. For simplicity, let us think that all columns of $\mathcal{C}_0$ have $O(1)$ ones as desired. (In the formal argument, we will add the bad columns to the exceptional block.) Of course, the block $\mathcal{R}_0$ can be handled similarly.

At this point, we decomposed $[n] \times [n]$ into $\mathcal{N}_0$, $\mathcal{R}_0$, $\mathcal{C}_0$ and an exceptional $n/2 \times n/2$ block. Now we repeat the process for the exceptional block, constructing $\mathcal{N}_1$, $\mathcal{R}_1$, and $\mathcal{C}_1$ there, and so on. Figure 2.3c illustrates this process.

[2] In this paper, by block we mean a product set $I \times J$ with arbitrary index subsets $I, J \subset [n]$. These subsets are not required to be intervals of successive integers.

At the end, we choose $\mathcal{N}$, $\mathcal{R}$ and $\mathcal{C}$ to be the unions of the blocks $\mathcal{N}_i$, $\mathcal{R}_i$ and $\mathcal{C}_i$ respectively.



(a) The core. (b) After the first step. (c) Final decomposition.

Figure 2.3: Constructing decomposition iteratively in the proof of Theorem 2.2.4.

Two precautions have to be taken in this argument. First, we need to make concentration on the core blocks $\mathcal{N}_i$ *better at each step*, so that the sum of those error bounds would not depend of the total number of steps. This can be done with little effort, with the help of the exponential decrease of the size of the blocks $\mathcal{N}_i$. Second, we have a control of the sizes but not locations of the exceptional blocks. Thus to be able to carry out the decomposition argument inside an exceptional block, we need to make the argument valid *uniformly* over all blocks of that size. This will require us to be delicate with probabilistic arguments, so we can take a union bound over such blocks.

2.3.2 Grothendieck-Pietsch factorization

As we mentioned in the previous section, our proof of Theorem 2.2.4 is based on Grothendieck-Pietsch factorization. This general and well known result in functional analysis [66, 67] has already been used in a similar probabilistic context, see [42, Proposition 15.11].

Grothendieck-Pietsch factorization compares two matrix norms, the $\ell_2 \to \ell_2$ norm (which we call the spectral norm) and the $\ell_\infty \to \ell_2$ norm. For a $k \times m$ matrix B, these norms are defined as

$$\|B\| = \max_{\|x\|_2=1} \|Bx\|_2, \qquad \|B\|_{\infty\to 2} = \max_{\|x\|_\infty=1} \|Bx\|_2 = \max_{x\in\{-1,1\}^m} \|Bx\|_2.$$

The $\ell_\infty \to \ell_2$ norm is usually easier to control, since the supremum is taken with respect to the discrete set $\{-1,1\}^m$, and any vector there has all coordinates of the same magnitude.

To compare the two norms, one can start with the obvious inequality

$$\frac{\|B\|_{\infty\to 2}}{\sqrt{m}} \le \|B\| \le \|B\|_{\infty\to 2}.$$

Both parts of this inequality are optimal, so there is an unavoidable slack between the upper and lower bounds. However, Grothendieck-Pietsch factorization allows us to tighten the inequality by changing B sightly. The next two results offer two ways to change B – introduce weights and pass to a sub-matrix.

Theorem 2.3.1 (Grothendieck-Pietsch's factorization, weighted version)**.** *Let B be a $k \times m$ real matrix. Then there exist positive weights μ_j with $\sum_{j=1}^m \mu_j = 1$ such that*

$$\|B\|_{\infty\to 2} \le \|BD_\mu^{-1/2}\| \le \sqrt{\pi/2}\|B\|_{\infty\to 2}, \tag{2.3.1}$$

where $D_\mu = \operatorname{diag}(\mu_j)$ denotes the $m \times m$ diagonal matrix with weights μ_j on the diagonal.

This result is a known combination of the Little Grothendieck Theorem (see [78, Corollary 10.10] and [68]) and Pietsch Factorization (see [78, Theorem 9.2]). In an explicit form, a version of this result can be found e.g. in [42, Proposition 15.11]. The weights μ_j can be computed algorithmically, see [79].

The following related version of Grothendieck-Pietsch's factorization can be especially useful in probabilistic contexts, see [42, Proposition 15.11]. Here and in the rest of the paper, we denote by $B_{I\times J}$ the sub-matrix of a matrix B with rows indexed by a subset I and columns indexed by a subset J.

Theorem 2.3.2 (Grothendieck-Pietsch factorization, sub-matrix version)**.** *Let B be a $k \times m$ real matrix and $\delta > 0$. Then there exists $J \subseteq [m]$ with $|J| \ge (1-\delta)m$ such that*

$$\|B_{[k]\times J}\| \le \frac{2\|B\|_{\infty\to 2}}{\sqrt{\delta m}}.$$

Proof. Consider the weights μ_i given by Theorem 2.3.1, and choose J to consist of the indices j satisfying $\mu_j \le 1/\delta m$. Since $\sum_j \mu_j = 1$, the set J must contain at least $(1-\delta)m$ indices as claimed. Furthermore, the diagonal entries of $(D_\mu^{-1/2})_{J\times J}$ are all bounded from below by $\sqrt{\delta m}$, which yields

$$\|(BD_\mu^{-1/2})_{[k]\times J}\| \ge \sqrt{\delta m}\|B_{[k]\times J}\|.$$

On the other hand, by (2.3.1) the left-hand side of this inequality is bounded by $2\|B\|_{\infty\to 2}$. Rearranging the terms, we complete the proof. □

2.3.3 Concentration on a big block

We are starting to work toward constructing the core part $\mathcal{N}$ in Theorem 2.2.4. In this section we will show how to find a big block on which the adjacency matrix A concentrates. First we will establish concentration in $\ell_\infty \to \ell_2$ norm, and then, using Grothendieck-Pietsch factorization, in the spectral norm.

The lemmas of this and next section should be best understood for $m = n$, $I = J = [n]$ and $\alpha = 1$. In this case, we are working with the entire adjacency matrix, and trying to make the first step in the iterative procedure. The further steps will require us to handle smaller blocks $I \times J$; the parameter α will then become smaller in order to achieve better concentration for smaller blocks.

Lemma 2.3.3 (Concentration in $\ell_\infty \to \ell_2$ norm)**.** *Let $1 \le m \le n$ and $\alpha \ge m/n$. Then for $r \ge 1$ the following holds with probability at least $1 - n^{-r}$. Consider a block $I \times J$ of size $m \times m$. Let I' be the set of indices of the rows of $A_{I\times J}$ that contain at most αd ones. Then*

$$\|(A - \mathbb{E}\,A)_{I'\times J}\|_{\infty\to 2} \le C\sqrt{\alpha dmr\log(en/m)}. \tag{2.3.2}$$

Proof. By definition,

$$\|(A-\mathbb{E}\,A)_{I'\times J}\|_{\infty\to 2}^2 = \max_{x\in\{-1,1\}^m} \sum_{i\in I'} \Big(\sum_{j\in J}(A_{ij}-\mathbb{E}\,A_{ij})x_j\Big)^2 = \max_{x\in\{-1,1\}^m} \sum_{i\in I} \big(X_i\xi_i\big)^2 \tag{2.3.3}$$

where we denoted

$$X_i := \sum_{j \in J} (A_{ij} - \mathbb{E}\, A_{ij}) x_j, \qquad \xi_i := \mathbf{1}_{\left\{\sum_{i \in J} A_{ij} \le \alpha d\right\}}.$$

Let us first fix a block $I \times J$ and a vector $x \in \{-1, 1\}^m$. Let us analyze the independent random variables $X_i \xi_i$. Since $|X_i| \le \sum_{j \in J} |A_{ij} - \mathbb{E}\, A_{ij}| \le \sum_{j \in J} A_{ij}$, it follows by definition of ξ_i that

$$|X_i \xi_i| \le \alpha d. \tag{2.3.4}$$

A more useful bond will follow from Bernstein's inequality. Indeed, X_i is a sum of m independent random variables with zero means and variances at most d/n. By Bernstein's inequality, for any $t > 0$ we have

$$\mathbb{P}\left\{|X_i \xi_i| > tm\right\} \le \mathbb{P}\left\{|X_i| > tm\right\} \le 2 \exp\left(\frac{-mt^2/2}{d/n + t/3}\right), \quad t \ge 0. \tag{2.3.5}$$

For $tm \le \alpha d$, this can be further bounded by $2\exp(-m^2 t^2 / 4\alpha d)$, once we use the assumption $\alpha \ge m/n$. For $tm > \alpha d$, the left-hand side of (2.3.5) is automatically zero by (2.3.4). Therefore

$$\mathbb{P}\left\{|X_i \xi_i| > tm\right\} \le 2 \exp\left(\frac{-m^2 t^2}{4\alpha d}\right), \qquad t \ge 0. \tag{2.3.6}$$

We are now ready to bound the right-hand side of (2.3.3). By (2.3.6), the random variable $X_i \xi_i$ is sub-gaussian[3] with sub-gaussian norm at most $\sqrt{\alpha d}$. It follows that $(X_i \xi_i)^2$ is sub-exponential with sub-exponential norm at most $2\alpha d$. Using Bernstein's inequality for sub-exponential random variables (see Corrollary 5.17 in [80]), we have

$$\mathbb{P}\left\{\sum_{i \in I} \left(X_i \xi_i\right)^2 > \varepsilon m \alpha d\right\} \le 2 \exp\left[-c \min\left(\varepsilon^2, \varepsilon\right) m\right], \qquad \varepsilon \ge 0. \tag{2.3.7}$$

Choosing $\varepsilon := (10/c) r \log(en/m)$, we bound this probability by $(en/m)^{-5rm}$.

[3] For definitions and basic facts about sub-gaussian random variables, see e.g. [80].

Summarizing, we have proved that for fixed $I, J \subseteq [n]$ and $x \in \{-1,1\}^m$, with probability at least $1 - (en/m)^{-5rm}$, the following holds:

$$\sum_{i \in I} \left(X_i \xi_i\right)^2 \leq (10/c) r \log(en/m) \cdot m\alpha d. \tag{2.3.8}$$

Taking a union bound over all possibilities of m, I, J, x and using (2.3.3), (2.3.8), we see that the conclusion of the lemma holds with probability at least

$$1 - \sum_{m=1}^{n} 2^m \binom{n}{m}^2 \left(\frac{en}{m}\right)^{-5rm} \geq 1 - n^{-r}.$$

The proof is complete. □

Applying Lemma 2.3.3 followed by Grothendieck-Piesch factorization (Theorem 2.3.2), we obtain the following.

Lemma 2.3.4 (Concentration in spectral norm)**.** *Let $1 \leq m \leq n$ and $\alpha \geq m/n$. Then for $r \geq 1$ the following holds with probability at least $1 - n^{-r}$. Consider a block $I \times J$ of size $m \times m$. Let I' be the set of indices of the rows of $A_{I \times J}$ that contain at most αd ones. Then one can find a subset $J' \subseteq J$ of at least $3m/4$ columns such that*

$$\|(A - \mathbb{E}\, A)_{I' \times J'}\| \leq C\sqrt{\alpha d r \log(en/m)}. \tag{2.3.9}$$

2.3.4 Restricted degrees

The two simple lemmas of this section will help us to handle the part of the adjacency matrix outside the core block constructed in Lemma 2.3.4. First, we show that almost all rows have at most $O(\alpha d)$ ones, and thus are included in the core block.

Lemma 2.3.5 (Degrees of subgraphs)**.** *Let $1 \leq m \leq n$ and $\alpha \geq \sqrt{m/n}$. Then for $r \geq 1$ the following holds with probability at least $1 - n^{-r}$. Consider a block $I \times J$ of size $m \times m$. Then all but $m/\alpha d$ rows of $A_{I \times J}$ have at most $8r\alpha d$ ones.*

Proof. Fix a block $I \times J$, and denote by d_i the number of ones in the i-th row of $A_{I\times J}$. Then $\mathbb{E}\, d_i \le md/n$ by the assumption. Using Chernoff's inequality, we obtain

$$\mathbb{P}\left\{d_i > 8r\alpha d\right\} \le \Big(\frac{8r\alpha d}{emd/n}\Big)^{-8r\alpha d} \le \Big(\frac{2\alpha n}{m}\Big)^{-8r\alpha d} =: p.$$

Let S be the number of rows i with $d_i > 8r\alpha d$. Then S is a sum of m independent Bernoulli random variables with expectations at most p. Again, Chernoff's inequality implies

$$\mathbb{P}\left\{S > m/\alpha d\right\} \le (ep\alpha d)^{m/\alpha d} \le p^{m/2\alpha d} = \Big(\frac{2\alpha n}{m}\Big)^{-4rm}.$$

The second inequality here holds since $e\alpha d \le p^{-1/2}$. (To see this, notice that the definition of p and assumption on α imply that $p^{-1/2} = (2\alpha n/m)^{4r\alpha d} \ge 2^{4r\alpha d}$.)

It remains to take a union bound over all blocks $I \times J$. We obtain that the conclusion of the lemma holds with probability at least

$$1 - \sum_{m=1}^{n} \binom{n}{m}^2 \Big(\frac{2\alpha n}{m}\Big)^{-4rm} \ge 1 - n^{-r}.$$

In the last inequality we used the assumption that $\alpha \ge \sqrt{m/n}$. The proof is complete. □

Next, we handle the block of rows that do have too many ones. We show that most *columns* of this block have $O(1)$ ones.

Lemma 2.3.6 (More on degrees of subgraphs)**.** *Let $1 \le m \le n$ and $\alpha \ge \sqrt{m/n}$. Then for $r \ge 1$ the following holds with probability at least $1 - n^{-r}$. Consider a block $I \times J$ of size $k \times m$ with some $k \le m/\alpha d$. Then all but $m/4$ columns of $A_{I\times J}$ have at most $32r$ ones.*

Proof. Fix I and J, and denote by d_j the number of ones in the j-th column of $A_{I\times J}$. Then $\mathbb{E}\, d_j \le kd/n \le m/\alpha n$ by assumption. Using Chernoff's inequality, we have

$$\mathbb{P}\left\{d_j > 32r\right\} \le \Big(\frac{32r}{em/\alpha n}\Big)^{-32r} \le \Big(\frac{10\alpha n}{m}\Big)^{-32r} =: p.$$

Let S be the number of columns j with $d_j > 32r$. Then S is a sum of m independent Bernoulli random variables with expectations at most p. Again, Chernoff's inequality implies

$$\mathbb{P}\left\{S > m/4\right\} \le (4ep)^{m/4} \le p^{m/6} \le \left(\frac{10\alpha n}{m}\right)^{-5rm}.$$

The second inequality here holds since $4e < p^{1/2}$, which in turn follows by assumption on α.

It remains to take a union bound over all blocks $I \times J$. It is enough to consider the blocks with largest possible number of columns, thus with $k = \lceil m/\alpha d \rceil$. We obtain that the conclusion of the lemma holds with probability at least

$$1 - \sum_{m=1}^{n} \binom{n}{m} \binom{n}{\lceil m/\alpha d \rceil} \left(\frac{10\alpha n}{m}\right)^{-5rm} \le 1 - n^{-r}.$$

In the last inequality we used the assumption that $\alpha \ge \sqrt{m/n}$. The proof is complete. □

2.3.5 Iterative decomposition: proof of Theorem 2.2.1

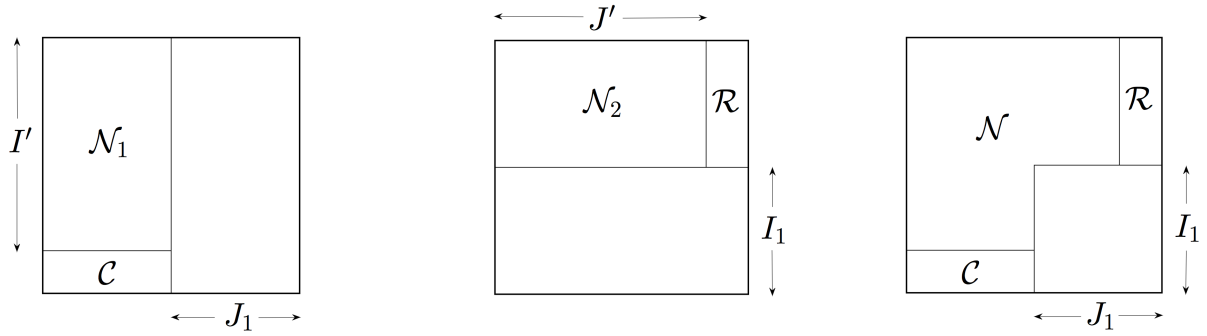
Finally, we combine the tools we developed so far, and we construct an iterative decomposition of the adjacency matrix the way we outline in Section 2.3.1. Let us set up one step of this procedure, where we consider an $m \times m$ block and decompose almost all of it (everything except an $m/2 \times m/2$ block) into classes $\mathcal{N}$, $\mathcal{R}$ and $\mathcal{C}$ satisfying the conclusion of Theorem 2.2.4. Once we can do this, we repeat the procedure for the $m/2 \times m/2$ block, etc.

Lemma 2.3.7 (Decomposition of a block)**.** *Let $1 \le m \le n$ and $\alpha \ge \sqrt{m/n}$. Then for $r \ge 1$ the following holds with probability at least $1 - 3n^{-r}$. Consider a block $I \times J$ of size $m \times m$. Then there exists an exceptional sub-block $I_1 \times J_1$ with dimensions at most $m/2 \times m/2$ such that the remaining part of the block, that is $(I \times J) \setminus (I_1 \times J_1)$, can be decomposed into three classes $\mathcal{N}$, $\mathcal{R} \subset (I \setminus I_1) \times J$ and $\mathcal{C} \subset I \times (J \setminus J_1)$ so that the following holds.*

- *The graph concentrates on $\mathcal{N}$, namely $\|(A - \mathbb{E}\,A)_{\mathcal{N}}\| \le Cr^{3/2}\sqrt{\alpha d \log(en/m)}$.*
- *Each row of $A_{\mathcal{R}}$ and each column of $A_{\mathcal{C}}$ has at most $32r$ ones.*

Moreover, $\mathcal{R}$ intersects at most $n/\alpha d$ columns and $\mathcal{C}$ intersects at most $n/\alpha d$ rows of $I \times J$.

After a permutation of rows and columns, a decomposition of the block stated in Lemma 2.3.7 can be visualized in Figure 2.4c.



(a) Initial step. (b) Repeat for transpose. (c) Final decomposition.

Figure 2.4: Construction of a block decomposition in Lemma 2.3.7.

Proof. Since we are going to use Lemmas 2.3.4, 2.3.5 and 2.3.6, let us fix realization of the random graph that satisfies the conclusion of those three lemmas.

By Lemma 2.3.5, all but $m/\alpha d$ rows of $A_{I\times J}$ have at most $8r\alpha d$ ones; let us denote by I' the set of indices of those rows with at most $8r\alpha d$ ones. Then we can use Lemma 2.3.4 for the block $I' \times J$ and with α replaced by $8r\alpha$; the choice of I' ensures that all rows have small numbers of ones, as required in that lemma. To control the rows outside I', we may use Lemma 2.3.6 for $(I \setminus I') \times J$; as we already noted, this block has at most $m/\alpha d$ rows as required in that lemma. Intersecting the good sets of columns produced by those two lemmas, we obtain a set of at most $m/2$ exceptional columns $J_1 \subset J$ such that the following holds.

- On the block $\mathcal{N}_1 := I' \times (J \setminus J_1)$, we have $\|(A - \mathbb{E}\,A)_{\mathcal{N}_1}\| \le Cr^{3/2}\sqrt{\alpha d \log(en/m)}$.
- For the block $\mathcal{C} := (I \setminus I') \times (J \setminus J_1)$, all columns of $A_{\mathcal{C}}$ have at most $32r$ ones.

Figure 2.4a illustrates the decomposition of the block $I \times J$ into the set of exceptional columns indexed by J_1 and good sets $\mathcal{N}_1$ and $\mathcal{C}$.

To finish the proof, we apply the above argument to the transpose A^T on the block $J \times I$. To be precise, we start with the set $J' \subset J$ of all but $m/\alpha d$ small columns of $A_{I\times J}$ (those with at most $8r\alpha d$ ones); then we obtain an exceptional set $I_1 \subset I$ of at most $m/2$ rows; and finally we conclude that A concentrates on the block $\mathcal{N}_2 := (I \setminus I_1) \times J'$ and has small rows on the block $\mathcal{R} := (I \setminus I_1) \times (J \setminus J')$. Figure 2.4b illustrates this decomposition.

It only remains to combine the decompositions for A and A^T; Figure 2.4c illustrates a result of the combination. Once we define $\mathcal{N} := \mathcal{N}_1 \cup \mathcal{N}_2$, it becomes clear that $\mathcal{N}$, $\mathcal{R}$ and $\mathcal{C}$ have the required properties.[4] □

Proof of Theorem 2.2.4. Let us fix a realization of the random graph that satisfies the conclusion of Lemma 2.3.7. Applying that lemma for $m = n$ and with $\alpha = 1$, we decompose the set of edges $[n] \times [n]$ into three classes $\mathcal{N}_0$, $\mathcal{C}_0$ and $\mathcal{R}_0$ plus an $n/2 \times n/2$ exceptional block $I_1 \times J_1$. Apply Lemma 2.3.7 again, this time for the block $I_1 \times J_1$, for $m = n/2$ and with $\alpha = \sqrt{1/2}$. We decompose $I_1 \times J_1$ into $\mathcal{N}_1$, $\mathcal{C}_1$ and $\mathcal{R}_1$ plus an $n/4 \times n/4$ exceptional block $I_2 \times J_2$.

Repeat this process for $\alpha = \sqrt{m/n}$ where m is the running size of the block; we halve this size at each step, and so we have $\alpha_i \le 2^{-i/2}$. Figure 2.3c illustrates a decomposition that we may obtain this way. In a finite number of steps (actually, in $O(\log n)$ steps) the exceptional block becomes empty, and the process terminates. At that point we have decomposed the set of edges $[n] \times [n]$ into $\mathcal{N}$, $\mathcal{R}$ and $\mathcal{C}$, defined as the union of $\mathcal{N}_i$, $\mathcal{C}_i$ and $\mathcal{R}_i$ respectively, which we obtained at each step. It is clear that $\mathcal{R}$ and $\mathcal{C}$ satisfy the required properties.

It remains to bound the deviation of A on $\mathcal{N}$. By construction, $\mathcal{N}_i$ satisfies

$$\|(A - \mathbb{E}\, A)_{\mathcal{N}_i}\| \le Cr^{3/2}\sqrt{\alpha_i d \log(e\alpha_i)}.$$

Thus, by triangle inequality we have

$$\|(A - \mathbb{E}\, A)_{\mathcal{N}}\| \le \sum_{i\ge 0} Cr^{3/2}\sqrt{\alpha_i d \log(e\alpha_i)} \le C' r^{3/2}\sqrt{d}.$$

[4] It may happen that an entry ends up in more than one class $\mathcal{N}$, $\mathcal{R}$ and $\mathcal{C}$. In such cases, we split the tie arbitrarily.

In the second inequality we used that $\alpha_i \le 2^{-i/2}$, which forces the series to converge. The proof of Theorem 2.2.4 is complete. □

2.3.6 Replacing the degrees by the ℓ_2 norms in Theorem 2.2.1

Let us now prove the "moreover" part of Theorem 2.2.1, where d' is the the maximal ℓ_2 norm of the rows and columns of the regularized adjacency matrix A'. This is clearly a stronger statement than in the main part of the theorem. Indeed, since all entries of A' are bounded in absolute value by 1, each degree, being the ℓ_1 norm of a row, is bounded below by the ℓ_2 norm squared.

This strengthening is in fact easy to check. To do so, note that the definition of d' was used only once in the proof of Theorem 2.2.1, namely in Step 2 where we bounded the norms of $A'_{\mathcal{R}}$ and $A'_{\mathcal{C}}$. Thus, to obtain the strengthening, it is enough to replace the application of Lemma 2.2.5 there by the following lemma.

Lemma 2.3.8. *Consider a matrix B with entries in $[0,1]$. Suppose each row of B has at most a non-zero entries, and each column has ℓ_2 norm at most $\sqrt{b}$. Then $\|B\| \le \sqrt{ab}$.*

Proof. To prove the claim, let x be a vector with $\|x\|_2 = 1$. Using Cauchy-Schwarz inequality and the assumptions, we have

$$\begin{aligned}\|Bx\|_2^2 &= \sum_j \Big(\sum_i B_{ij} x_i\Big)^2 \le \sum_j \Big(\sum_{i:\, B_{ij}\neq 0} B_{ij}^2 \sum_{i:\, B_{ij}\neq 0} x_i^2\Big) \\ &\le \sum_j \Big(b \sum_{i:\, B_{ij}\neq 0} x_i^2\Big) = b \sum_i x_i^2 \sum_{j:\, B_{ij}\neq 0} 1 \le b \sum_i x_i^2 a = ab.\end{aligned}$$

Since x is arbitrary, this completes the proof. □

2.4 Concentration of the regularized Laplacian

In this section, we state the following formal version of Theorem 2.1.2, and we deduce it from concentration of adjacency matrices (Theorem 2.2.1).

Theorem 2.4.1 (Concentration of regularized Laplacians)**.** *Consider a random graph from the inhomogeneous Erdös-Rényi model, and let d be as in (2.1.3). Choose a number $\tau > 0$. Then, for any $r \geq 1$, with probability at least $1 - e^{-r}$ one has*

$$\|\mathcal{L}(A_\tau) - \mathcal{L}(\mathbb{E}\, A_\tau)\| \leq \frac{Cr^2}{\sqrt{\tau}}\Big(1 + \frac{d}{\tau}\Big)^{5/2}.$$

Proof. Two sources contribute to the deviation of Laplacian – the deviation of the adjacency matrix and the deviation of the degrees. Let us separate and bound them individually.

Step 1. Decomposing the deviation. Let us denote $\bar{A} := \mathbb{E}\, A$ for simplicity; then

$$E := \mathcal{L}(A_\tau) - \mathcal{L}(\bar{A}_\tau) = D_\tau^{-1/2} A_\tau D_\tau^{-1/2} - \bar{D}_\tau^{-1/2} \bar{A}_\tau \bar{D}_\tau^{-1/2}.$$

Here $D_\tau = \operatorname{diag}(d_i + \tau)$ and $\bar{D}_\tau = \operatorname{diag}(\bar{d}_i + \tau)$ are the diagonal matrices with degrees of A_τ and $\bar{A}_\tau$ on the diagonal, respectively. Using the fact that $A_\tau - \bar{A}_\tau = A - \bar{A}$, we can represent the deviation as $E = S + T$, where

$$S = D_\tau^{-1/2}(A - \bar{A})D_\tau^{-1/2}, \quad T = D_\tau^{-1/2}\bar{A}_\tau D_\tau^{-1/2} - \bar{D}_\tau^{-1/2}\bar{A}_\tau\bar{D}_\tau^{-1/2}.$$

Let us bound S and T separately.

Step 2. Bounding S. Let us introduce a diagonal matrix Δ that should be easier to work with than D_τ. Set $\Delta_{ii} = 1$ if $d_i \leq 8rd$ and $\Delta_{ii} = d_i/\tau + 1$ otherwise. Then entries of $\tau\Delta$ are upper bounded by the corresponding entries of D_τ, and so

$$\tau\|S\| \leq \|\Delta^{-1/2}(A - \bar{A})\Delta^{-1/2}\|.$$

Next, by triangle inequality,

$$\tau\|S\| \leq \|\Delta^{-1/2} A \Delta^{-1/2} - \bar{A}\| + \|\bar{A} - \Delta^{-1/2}\bar{A}\Delta^{-1/2}\| =: R_1 + R_2. \qquad (2.4.1)$$

In order to bound R_1, we use Theorem 2.2.1 to show that $A' := \Delta^{-1/2} A \Delta^{-1/2}$ concentrates around $\bar{A}$. This should be possible because A' is obtained from A by reducing the degrees that are bigger than $8rd$. To apply the "moreover"

part of Theorem 2.2.1, let us check the magnitude of the ℓ_2 norms of the rows A'_i of A':

$$\|A'_i\|_2^2 = \sum_{j=1}^{n} \frac{A_{ij}}{\Delta_{ii}\Delta_{jj}} \le \frac{d_i}{\Delta_{ii}} \le \max(8rd,\, \tau).$$

Here in the first inequality we used that $\Delta_{jj} \ge 1$ and $\sum_j A_{ij} = d_i$; the second inequality follows by definition of Δ_{ii}. Applying Theorem 2.2.1, we obtain with probability $1 - n^{-r}$ that

$$R_1 = \|A' - \bar{A}\| \le C_1 r^2(\sqrt{d} + \sqrt{\tau}).$$

To bound R_2, we note that by construction of Δ, the matrices $\bar{A}$ and $\Delta^{-1/2}\bar{A}\Delta^{-1/2}$ coincide on the block $I \times I$, where I is the set of vertices satisfying $d_i \le 8rd$. This block is very large – indeed, Lemma 2.3.5 implies that $|I^c| \le n/d$ with probability $1 - n^{-r}$. Outside this block, i.e. on the small blocks $I^c \times [n]$ and $[n] \times I^c$, the entries of $\bar{A} - \Delta^{-1/2}\bar{A}\Delta^{-1/2}$ are bounded by the corresponding entries of $\bar{A}$, which are all bounded by d/n. Thus, using Lemma 2.2.5, we have

$$R_2 \le \|\bar{A}_{I^c \times [n]}\| + \|\bar{A}_{[n] \times I^c}\| \le 2\sqrt{d}.$$

Substituting the bounds for R_1 and R_2 into (2.4.1), we conclude that

$$\|S\| \le \frac{C_2 r^2}{\tau}(\sqrt{d} + \sqrt{\tau})$$

with probability at least $1 - 2n^{-r}$.

Step 3. Bounding T. Bounding the spectral norm by the Hilbert-Schmidt norm, we get

$$\|T\| \le \|T\|_{\text{HS}} = \sum_{i,j=1}^{n} T_{ij}^2, \quad \text{where} \quad T_{ij} = (\bar{A}_{ij} + \tau/n)\Big[1/\sqrt{\delta_{ij}} - 1/\sqrt{\bar{\delta}_{ij}}\Big]$$

and $\delta_{ij} = (d_i + \tau)(d_j + \tau)$ and $\bar{\delta}_{ij} = (\bar{d}_i + \tau)(\bar{d}_j + \tau)$. To bound T_{ij}, we note that

$$0 \le \bar{A}_{ij} + \tau/n \le \frac{d + \tau}{n} \quad \text{and} \quad \big|1/\sqrt{\delta_{ij}} - 1/\sqrt{\bar{\delta}_{ij}}\big| = \left|\frac{\delta_{ij} - \bar{\delta}_{ij}}{\delta_{ij}\sqrt{\bar{\delta}_{ij}} + \bar{\delta}_{ij}\sqrt{\delta_{ij}}}\right| \ge \frac{|\delta_{ij} - \bar{\delta}_{ij}|}{2\tau^3}.$$

Recalling the definition of δ_{ij} and $\bar{\delta}_{ij}$ and adding and subtracting $(d_i+\tau)(\bar{d}_j+\tau)$, we have

$$\delta_{ij} - \bar{\delta}_{ij} = (d_i + \tau)(d_j - \bar{d}_j) + (\bar{d}_j + \tau)(d_i - \bar{d}_i).$$

So, using the inequality $(a+b)^2 \le 2(a^2+b^2)$ and bounding $\bar{d}_j + \tau$ by $d + \tau$, we obtain

$$\|T\|^2 \le \frac{(d+\tau)^2}{n^2\tau^6}\Big[\sum_{i=1}^n (d_i+\tau)^2 \sum_{j=1}^n (d_j-\bar{d}_j)^2 + n(d+\tau)^2 \sum_{i=1}^n (d_i-\bar{d}_i)^2\Big]. \quad (2.4.2)$$

We claim that

$$\sum_{j=1}^n (d_j - \bar{d}_j)^2 \le C_3 r^2 n d \quad \text{with probability } 1 - e^{-2r}. \quad (2.4.3)$$

Indeed, since the variance of each d_i is bounded by d, the expectation of the sum in (2.4.3) is bounded by nd. To upgrade the variance bound to an exponential deviation bound, one can use one of the several standard methods. For example, Bernstein's inequality implies that $X_i = d_j - \bar{d}_j$ satisfies $\mathbb{P}\left\{X_i > C_4 t\sqrt{d}\right\} \le e^{-t}$ for all $t \ge 1$. This means that the random variable X_i^2 belongs to the Orlicz space $L_{\psi_{1/2}}$ and has norm $\|X_i^2\|_{\psi_{1/2}} \le C_3 d$, see [42]. By triangle inequality, we conclude that $\|\sum_{i=1}^n X_i^2\|_{\psi_{1/2}} \le C_3 n d$, which in turn implies (2.4.3).

Furthermore, (2.4.3) implies

$$\sum_{i=1}^n (d_i+\tau)^2 \le 2\sum_{i=1}^n (d_i-\bar{d}_i)^2 + 2\sum_{i=1}^n (\bar{d}_i+\tau)^2 \le 2C_3 r^2 n d + 2n(d+\tau)^2 \le C_5 r^2 n (d+\tau)^2.$$

Substituting this bound and (2.4.3) into (2.4.2) we conclude that

$$\|T\|^2 \le \frac{(d+\tau)^2}{n^2\tau^6} \cdot C_3 r^2 n d \left[C_5 r^2 n (d+\tau)^2 + n(d+\tau)^2\right] \le \frac{C_6 r^4}{\tau}\Big(1 + \frac{d}{\tau}\Big)^5.$$

It remains to substitute the bounds for S and T into the inequality $\|E\| \le \|S\| + \|T\|$, and simplify the expression. The resulting bound holds with probability at least $1 - n^{-r} - n^{-r} - e^{-2r} \ge 1 - e^{-r}$, as claimed. □

2.5 Numerical comparisons

We briefly compare the empirical performance of spectral clustering that uses three different matrices as the input: the adjacency matrix, the regularized adjacency matrix, and the regularized Laplacian. Given an input matrix B, we first compute K leading eigenvectors associated with K largest eigenvalues of B; we then form an $n \times K$ matrix using these eigenvectors as column vectors, and apply *k-means* on row vectors of the newly formed matrix to get an estimate of the communities.

We generate networks either from the stochastic block model or the degree-corrected stochastic block model with 900 nodes and three communities of equal sizes ($n = 900, K = 3$). The number of replications for each setting is 100. Following [4], 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 $w = (w_1, w_2, w_3)$, which determines the relative node degrees within communities. Let

$$P_0 = \begin{pmatrix} w_1 & r & r \\ r & w_2 & r \\ r & r & w_3 \end{pmatrix}.$$

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 entries of $\pi = (1/3, 1/3, 1/3)^\mathsf{T}$ are fractions of nodes in three communities. Finally, edges A_{ij} are drawn independently from a Bernoulli distribution with $\mathbb{P}(A_{ij} = 1) = \theta_i \theta_j P_{c_i c_j}$, where c is the true label vector. In simulations, we fix $r = 0.25$ and $w = (1, 1, 1)^\mathsf{T}$.

The regularized adjacency matrix A' is formed as follows. For each node i with degree d_i greater than the average degree $\bar{d} = (d_1 + \cdots + d_n)/n$, we

normalize i-th row and column of A by multiplying all their entries with $\bar{d}/d_i$. The resulting matrix A' has all (weighted) degrees bounded by $\bar{d}$. To compute the regularized Laplacian, we first compute $A'' = A + (\bar{d}/10n)\mathbf{1}\mathbf{1}^\mathsf{T}$ by adding a small value $\bar{d}/10n$ to all entries of A; the regularized Laplacian is computed as the Laplacian of the weighted network with adjacency matrix A''.

We measure the accuracy of a community estimate e by the *overlap* it has with the true label vector c, defined by

$$\max_{\eta} \left(\frac{1}{n} \sum_{i=1}^{n} \mathbf{1}(c_i = \eta(e_i)) - \frac{1}{K} \right) \Big/ \left(1 - \frac{1}{K} \right),$$

where the maximum is taken over the set of all permutations η of K labels. The overlap is one for the true labeling and zero for a uniformly random labeling.

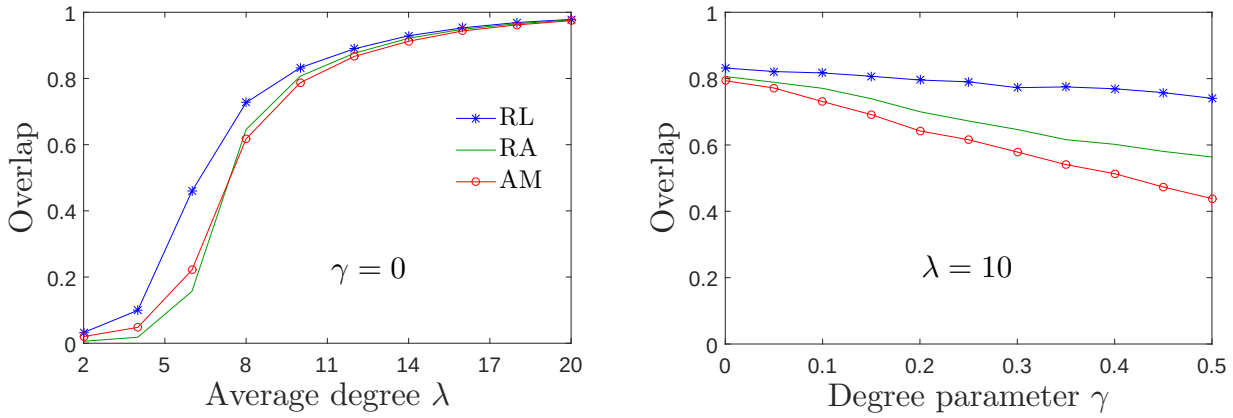


Figure 2.5: The performance of spectral clustering with different input matrices: adjacency matrix (AM), regularized adjacency matrix (RA), and regularized Laplacian (RL).

Figure 2.5 shows the performance of spectral clustering with three different input matricecs: adjacency matrix (AM), regularized adjacency matrix (RA), and regularized Laplacian (RL). On the left plot we see the performance of these methods under the SBM ($\gamma = 0$) with average degree λ varying from 2 to 20. All methods perform similarly, with RL being the best, especially when networks are sparse. The right plot shows the performance of three methods under the DCSBM with average degree $\lambda = 10$ and the degree parameter γ

varying from 0 to 0.5; as γ increases, the variation of node degrees increases. We see that RL is the most accurate method and its performance is very stable under the change of γ. In contrast, the accuracy of AM significantly drops as γ increases. This is expected because the Laplacian is a node-degree normalized version of the adjacency matrix; the normalization partially cancels out the effect of degree variation. Due to a similar normalization effect, RA is more accurate than AM, although it is not as stable as RL under the change of γ.

Chapter 3
Optimization via Low-rank Approximation for Community Detection in Networks

3.1 Introduction

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 $\boldsymbol{c} \in \{1, \ldots, K\}^n$ for the n nodes from the observed adjacency matrix A. In this chapter we assume that the number of communities K is known; we address the problem of estimating K in Chapter 4. We focus on the undirected network case, where 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 matrix, or optimizing global criteria derived from other considerations over label assignments $\boldsymbol{c}$, often via spectral approximations.

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 [76, 62, 49], as well as maximizing profile likelihoods by some type of greedy label-switching algorithms. The profile likelihood was derived for the SBM by [10] and for the DCSBM by [34], but the label-switching greedy search algorithms only scale up to a few thousand nodes. [4] 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 [20]. 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 [60, 57]. 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,

$$O_{kl}(e) = \sum_{i,j=1}^{n} A_{ij} I\{e_i = k, e_j = l\} \tag{3.1.1}$$

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 [16] for the entire graph, and can be written in the form

$$Q_{NG}(e) = \frac{1}{2m} \sum_{k} (O_{kk} - \frac{O_k^2}{m}) \tag{3.1.2}$$

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

$$Q_{BM}(e) = \sum_{k,l=1}^{K} O_{kl} \log \frac{O_{kl}}{n_k n_l} \ , \tag{3.1.3}$$

$$Q_{DC}(e) = \sum_{k,l=1}^{K} O_{kl} \log \frac{O_{kl}}{O_k O_l} \ . \tag{3.1.4}$$

Another example is the extraction criterion [88] 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 non-members. 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),

$$Q_{EX}(V) = |V||V^c| \left(\frac{O(V)}{|V|^2} - \frac{B(V)}{|V||V^c|} \right), \tag{3.1.5}$$

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 [27].

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, for example, [61]), a generic clustering method which became popular for community detection. In this context, the method has been analyzed by [72, 13, 71, 45], among others, while [32] proposed a spectral method specifically for the DCSBM. In spectral clustering, typically one first computes the normalized Laplacian matrix $L = D^{-1/2} A D^{-1/2}$, 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 [75] 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 [13, 70] or to A [4, 33, 39].

In this chapter we propose 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 $\mathbb{E}[A]$. 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 [57, 71, 59], 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 [53] for recent advances). An extension to the general case of $K > 2$ is briefly discussed in Section 3.2.3.

The rest of this chapter is organized as follows. In Section 3.2, 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 3.3, we show how the general method can be applied to several community detection criteria. Section 3.4 compares numerical performance of different methods. The proofs are given in Section 3.5 and Section 3.6.

3.2 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 3.2.2 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 3.3.5 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, e.g., using the eigenvectors of A itself, of its Laplacian, or of their various regularizations – see Section 3.2.1 for further discussion of this issue. The algoritm 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 settingwhere 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

$$f_A(e) = \sum_{j=1}^{\kappa} g_j(h_{A,j}(e)), \tag{3.2.1}$$

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

$$h_{A,j}(e) = (e + s_{j1})^T A(e + s_{j2}). \tag{3.2.2}$$

Here κ is a fixed number, s_{j1} and s_{j2} are constant vectors in $\{-1, 1\}^n$. Note that by (3.3.1), the number of edges between communities has the form (3.2.2), and by (3.3.2), the log-likelihood of the degree-corrected block model Q_{DC} is a special case of (3.2.1) 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 (3.2.1) and (3.2.2). 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

$$e^* = \operatorname{argmax}\big\{ f_A(e), e \in \mathcal{E}_A \big\}. \tag{3.2.3}$$

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 3.3), 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 3.2.1 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 3.2.1. *If assumptions (1) and (2) hold then there exists a constant $M_2 > 0$ such that*

$$f_T(c) - f_T(e^*) \leq M_2 n \log(n) \big(\|B\| \cdot \|U_A - U_B\| + \|A - B\| \big), \tag{3.2.4}$$

where T is either A or B.

The proof of Lemma 3.2.1 is given in Section 3.5. 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 $e \in \{-1, 1\}^n$,

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

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

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

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 3.3.

Theorem 3.2.2. *If assumptions (1)–(4) hold, then there exists a constant M_5 such that*

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

Theorem 3.2.2 follows directly from Lemma 3.2.1 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 bound on $\|A - B\|$ is readily available by Theorem 2.2.1, which in turn yields a bound on $\|U_A - U_B\|$ by the Davis-Kahan Theorem (see Lemma 3.6.2). Under certain conditions, the upper bound in Theorem 3.2.2 is of order $o(n)$ (see Section 3.3), which shows consistency of e^* as an estimator of c (i.e., the fraction of mislabeled nodes goes to 0 as $n \to \infty$).

3.2.1 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 [64, 45] 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 [52, 13, 70, 39]. This can be avoided by regularizing the Laplacian in some form; we follow the algorithm of [4]; see also [33, 39] 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 [4], 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 [4].

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 [28], 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 [28]. 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 counter-clockwise 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 (3.2.2) can be updated efficiently. In particular, if A is the adjacency matrix of a network with average degree λ_n, then on avarage, 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 (3.2.3). 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 $m = 2$.

3.2.3 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 $l \neq k$. The numbers of edges between communities defined by (3.1.1) are entries of $S^T AS$. Let $B = \sum_{i=1}^K \rho_i \bar{u}_i \bar{u}_i^T$ define the eigendecomposition of B. The population version of $S^T AS$ is

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

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 3.2.1. Let $\tilde{S}$ be the 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 j-th column of V_{ij} is the i-th 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 [28]. If $d = K(K-1)$, then the number of vertices of $U_A\tilde{S}$ is at most $O\left(n^{(d-1)}K^{2(d-1)}\right)$, and they can be found in $O\left(n^d K^{(2d-1)}\right)$ arithmetic operations. An implementation of the reverse-search algorithm of [24] for computing the Minkowski sum of polytopes was presented in [84], 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 $K = 2$.

3.3 Applications to community detection

Here we apply the general results from Section 3.2 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 (3.1.1), which are used by all the criteria we consider. They are quadratic forms of A and e and can be written as

$$\begin{aligned} O_{11}(e) &= \frac{1}{4}(\mathbf{1}+e)^T A(\mathbf{1}+e), \quad O_{22}(e) = \frac{1}{4}(\mathbf{1}-e)^T A(\mathbf{1}-e), \\ O_{12}(e) &= \frac{1}{4}(\mathbf{1}+e)^T A(\mathbf{1}-e), \end{aligned} \tag{3.3.1}$$

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

3.3.1 Maximizing the likelihood of the degree-corrected stochastic block model

When a network has two communities, (3.1.4) takes the form

$$\begin{aligned} Q_{DC}(e) &= O_{11}\log O_{11} + O_{22}\log O_{22} + 2O_{12}\log O_{12} \\ &- 2O_1\log O_1 - 2O_2\log O_2. \end{aligned} \tag{3.3.2}$$

Thus, Q_{DC} has the form defined by (3.2.1).

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

$$P = \lambda_n \begin{pmatrix} 1 & r \\ r & \omega \end{pmatrix} \tag{3.3.3}$$

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 $A_{ij} = \text{Bernoulli}(\theta_i\theta_j P_{c_ic_j})$. 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 3.3.1 establishes consistency of our method in this setting.

Theorem 3.3.1. *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 (3.2.3) with $f_A = Q_{DC}$. 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*

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

In particular, if U_A is a matrix whose row vectors are leading eignvectors of A, then the fraction of mis-clustered 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 3.3.1. The proof of Theorem 3.3.1 consists of checking assumptions (1), (3), (4), and a weaker version of assumption (2). For details, see [40].

3.3.2 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, (3.1.3) admits the form

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

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

$$n_1 = \frac{1}{2}(\mathbf{1} + e)^T \mathbf{1} = \frac{1}{2}(n + e^T \mathbf{1}), \quad n_2 = \frac{1}{2}(\mathbf{1} - e)^T \mathbf{1} = \frac{1}{2}(n - e^T \mathbf{1}). \tag{3.3.4}$$

Theorem 3.3.2. *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 (3.2.3) with $f_A = Q_{BM}$. 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*

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

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

Note that Q_{BM} does not have the exact form of (3.2.1) but a small modification shows that Lemma 3.2.1 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 3.3.2. The proof of Theorem 3.3.2 consists of showing the analog of Lemma 3.2.1, checking assumptions (3), (4), and a weaker version of assumption (2). For details, see [40].

3.3.3 Maximizing the Newman-Girvan modularity

When a network has two communities, up to a constant factor the modularity (3.1.2) takes the form

$$Q_{NG}(e) = O_{11} + O_{22} - \frac{O_1^2 + O_2^2}{O_1 + O_2} = \frac{2O_1O_2}{O_1 + O_2} - 2O_{12}.$$

Again, Q_{NG} does not have the exact form (3.2.1), but with a small modification, the argument used for proving Lemma 3.2.1 and Theorem 3.2.2 still holds for Q_{NG} under the regular SBM.

Theorem 3.3.3. *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 (3.2.3) with $f_A = Q_{NG}$. 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*

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

In particular, if U_A is a matrix whose row vectors are leading eignvectors of A, then the fraction of mis-clustered 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 3.2.1 goes through. The proof of Theorem 3.3.3 consists of checking assumptions (2), (3), (4), and the Lipschitz condition for Q_{NG}. For details, see [40].

3.3.4 Maximizing the community extraction criterion

Identifying the community V to be extracted with a label vector e, the criterion (3.1.5) can be written as

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

where n_1, n_2 are defined by (3.3.4). Once again Q_{EX} does not have the exact form (3.2.1), but with small modifications of the proof, Lemma 3.2.1 and Theorem 3.2.2 still hold for Q_{EX}.

Theorem 3.3.4. *Let A be the adjacency matrix generated from the SBM with the probability matrix (3.3.3), $\omega = r$, 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 (3.2.3) with $f_A = Q_{EX}$. 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*

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

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

The proof of Theorem 3.3.4 consists of verifying a version of Lemma 3.2.1 and assumptions (2), (3), and (4), and is included in [40].

3.3.5 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 (3.2.3). 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 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 [40]). We can then expect the projection $U_A[-1,1]^n$ to look somewhat similar – see the illustration in Figure 3.1. 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

$$\begin{aligned} \hat{c} &= \arg\max\left\{\langle U_A e, (U_A\mathbf{1})^\perp\rangle : e \in \{-1,1\}^n\right\} \\ &= \operatorname{sign}(u_1^T\mathbf{1}u_2 - u_2^T\mathbf{1}u_1), \end{aligned} \tag{3.3.5}$$

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

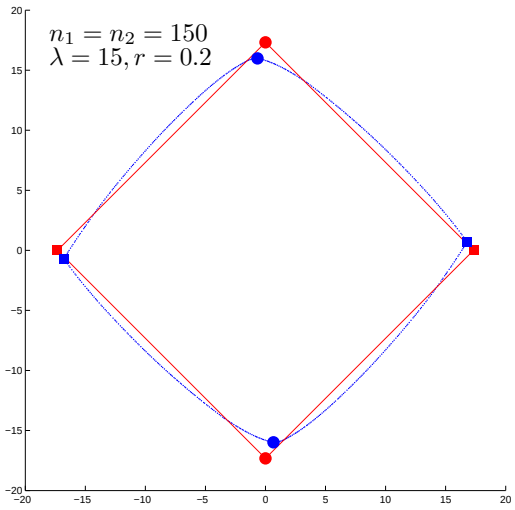


Figure 3.1: The projection of the cube $[-1,1]^n$ onto two-dimensional subspace. Blue corresponds to the projection onto eigenvectors of A, and red onto the eigenvectors of $\mathbb{E}[A]$. The red contour is the boundary of $U_{\mathbb{E}[A]}[-1,1]^n$; the blue dots are the extreme points of $U_A[-1,1]^n$. Circles (at the corners) are $\pm$ projections of the true label vector; squares are $\pm$ projections of the vector of all 1s.

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 3.4. Theorem 3.3.5 shows that $\hat{c}$ is a consistent estimator. The proof is given in Section 3.6.

Theorem 3.3.5. *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*

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

In particular, if U_A is a matrix whose row vectors are leading eignvectors of

A, then the fraction of mis-clustered nodes is bounded by M/λ_n.

3.4 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 in Section 3.4.5. 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 3.3, 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 3.3.5 (AEP, for approximate extreme points), and regularized spectral clustering (SCR) proposed by [4], 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 [4]. For the Newman-Girvan modularity, we compare to the spectral algorithm of [57], which uses the leading eigenvector of the modularity matrix (see details in Section 3.4.3). Finally, for community extraction we compare to the algorithm proposed in the original paper [88] based on greedy label switching, as there are no faster algorithms available.

The simulated networks are generated using the parametrization of [4], 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 3.2.1, 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 3.2.1). To find extreme points and corresponding label vectors in the second step of EP, we use the algorithm of [28]. 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 3.3.5). For CPL and UPL, following [4], 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 [85] as $\mathrm{NMI}(c, e) = -\sum_{i,j} R_{ij} \log \frac{R_{ij}}{R_{i+}R_{+j}} \left(\sum_{ij} R_{ij} \log R_{ij}\right)^{-1}$, 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.

3.4.1 The degree-corrected stochastic block model

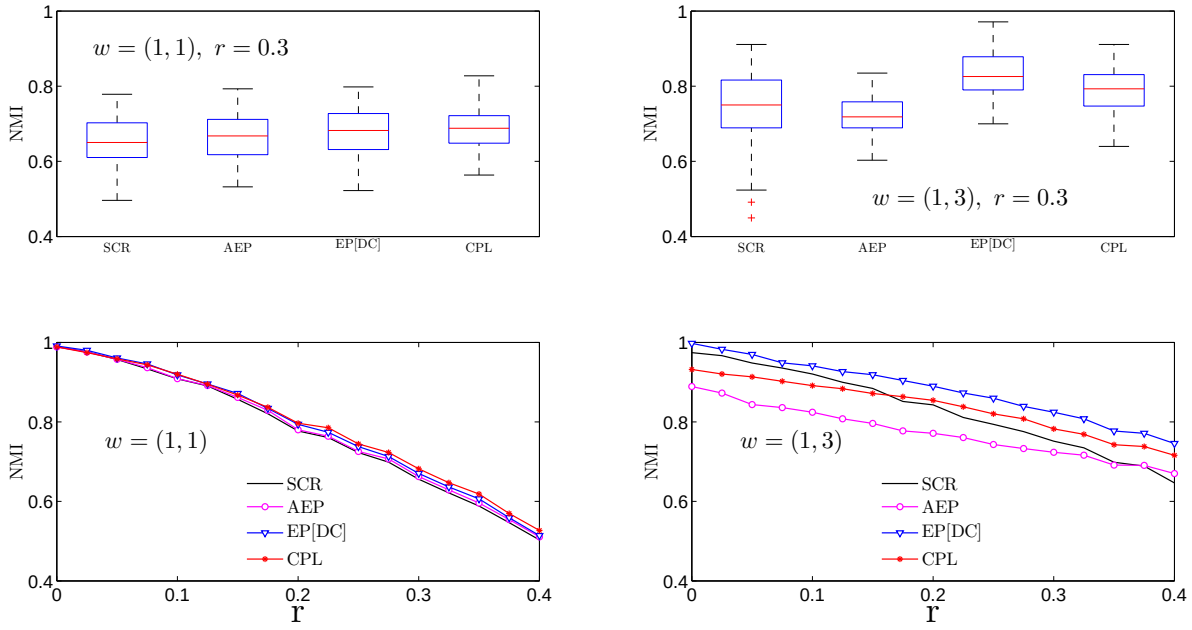


Figure 3.2: The degree-corrected stochastic block model. Top row: boxplots of NMI between true and estimated labels. Bottom row: average NMI against the out-in probability ratio r. In all plots, $n_1 = n_2 = 150$, $\lambda = 15$, and $\gamma = 0.5$.

Figure 3.2 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, 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.

3.4.2 The stochastic block model

Figure 3.3 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 appoximate method AEP is very similar to that of EP, and the model-specific UPL marginally outperforms the three general methods.

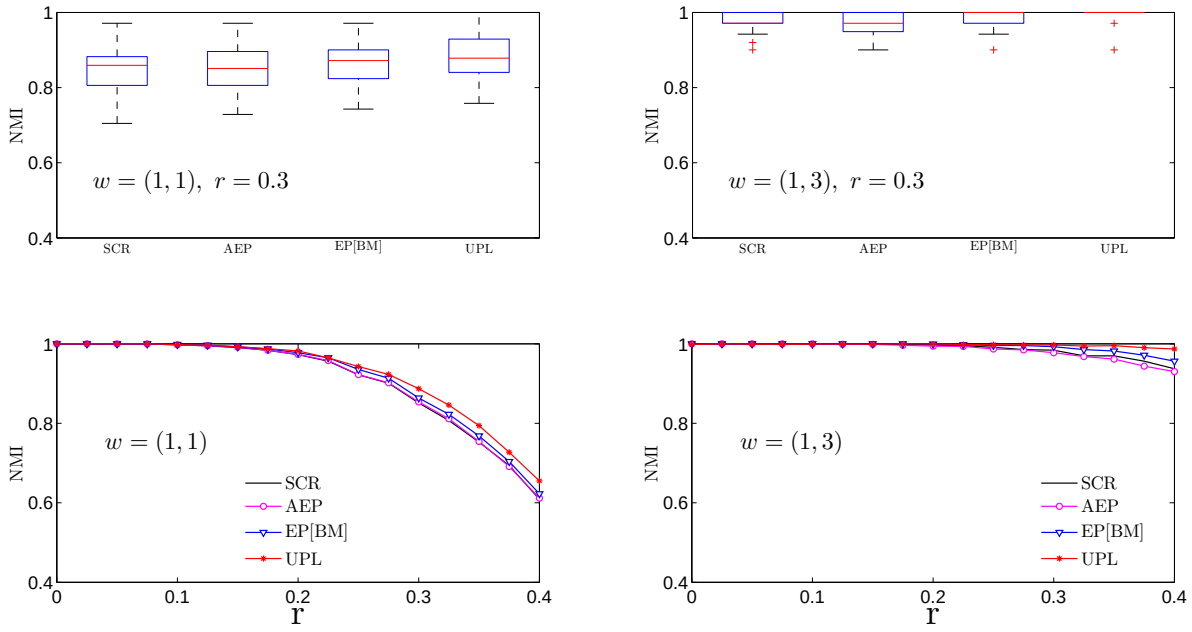


Figure 3.3: The stochastic block model. Top row: boxplots of NMI between true and estimated labels. Bottom row: average NMI against the out-in probability ratio r. In all plots, $n_1 = n_2 = 150$, $\lambda = 15$, and $\gamma = 0$.

3.4.3 Newman-Girvan Modularity

The modularity function $\hat{Q}_{NG}$ can be approximately maximized via a fast spectral algotithm when partitioning into two communities [57]. Let $B = A - P$ where $P_{ij} = d_i d_j / m$, and write $\hat{Q}_{NG}(e) = \frac{1}{2m} e^T B e$. 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 3.4 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.

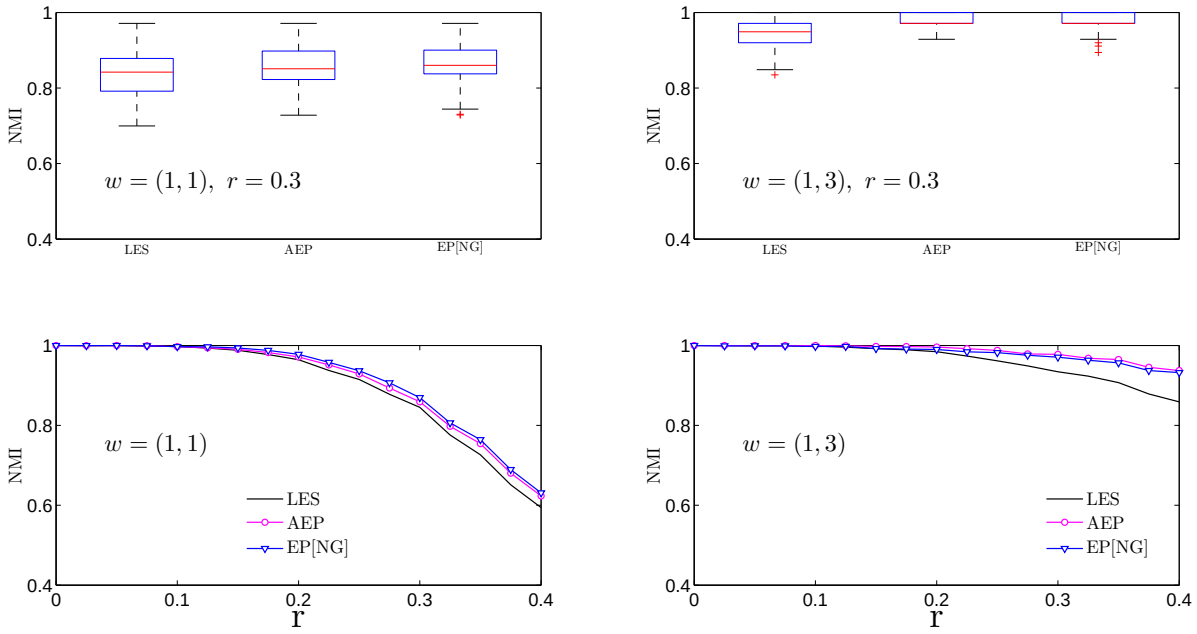


Figure 3.4: Newman-Girvan modularity. Top row: boxplots of NMI between true and estimated labels. Bottom row: average NMI against the out-in probability ratio r. In all plots, $n_1 = n_2 = 150$, $\lambda = 15$, and $\gamma = 0$.

3.4.4 Community Extraction Criterion

Following the original extraction paper of [88], 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 3.5 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 assymetry of the extraction problem.

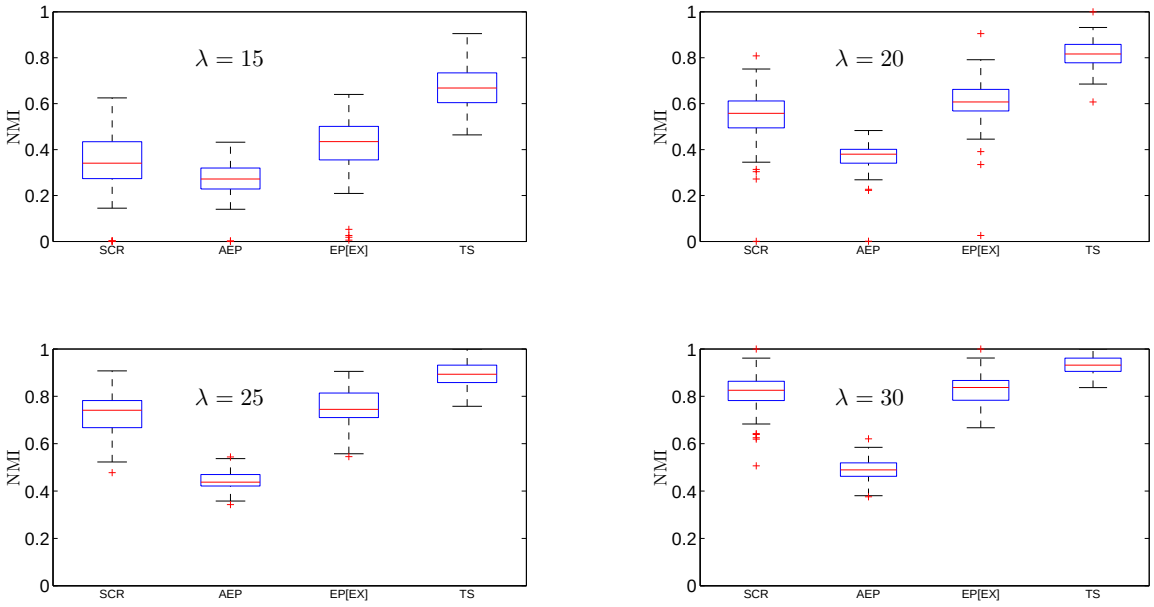
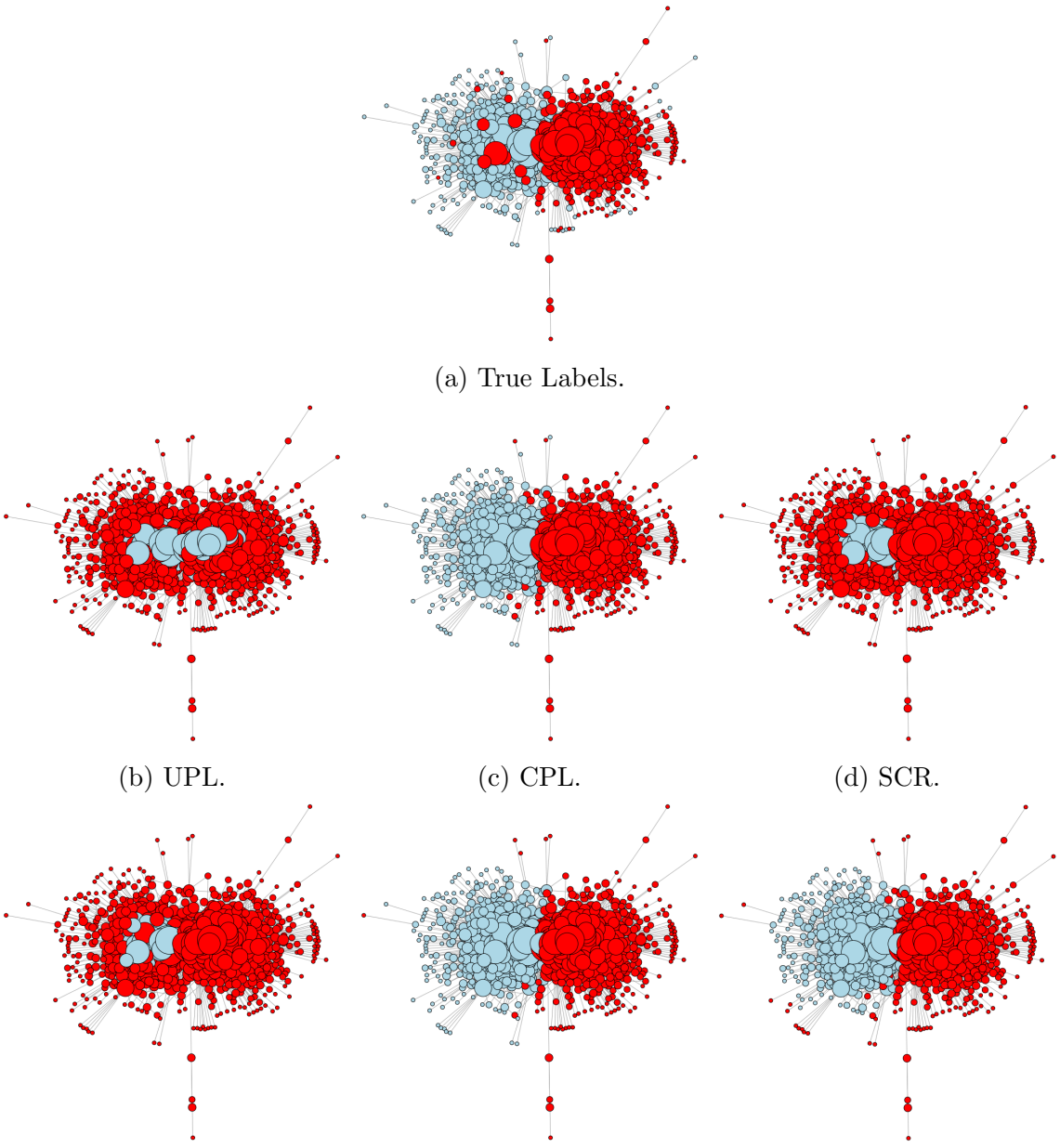


Figure 3.5: Community extraction. The boxplots of NMI between true and estimated labels. In all plots, $n_1 = 60$, $n_2 = 240$, and $\gamma = 0$.

3.4.5 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 [34], and [89], we ignore directions of the hyperlinks and only examine the largest connected component of this network, which has 1222 nodes and 16,714 edges, with the average degree of approximately 27. Table 3.1 and Figure 3.6 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 presense of hub nodes.

The second network we study represents social ties between 62 bottlenose dolphins living in Doubtful Sound, New Zealand [48, 47]. At some point



(a) True Labels.

(b) UPL. (c) CPL. (d) SCR.

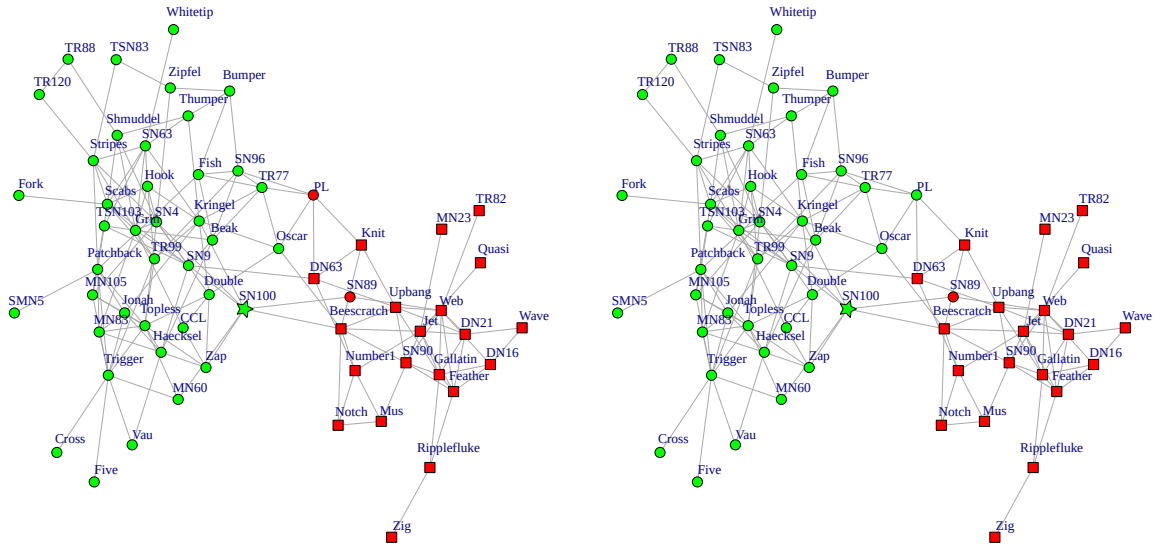
(e) EP(BM). (f) EP(DC). (g) AEP.

Figure 3.6: The network of political blogs. Node diameter is proportional to the logarithm of its degree and the colors represent community labels.

Table 3.1: 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

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 3.1 and Figure 3.7 show the performance of different methods. In Figure 3.7, 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 modelled by the SBM, there is no difference between DCSBM and SBM based methods, and all methods perform well.



(a) Output of AEP.

(b) Output of SCR, EP, UPL, and CPL.

Figure 3.7: The network of 62 bottlenose dolphins. Node shapes represent the split after the dolphin SN100 (represented by the star) left the group. Node colors represent their estimated labels.

3.5 Proof of results in Section 2

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

Lemma 3.5.1. *Assume that Assumption (1) holds. For any $j \leq \kappa$ (see 3.2.1), and $x, y \in U_B[-1,1]^n$, we have*

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

where M is a constant independent of n.

Proof of Lemma 3.5.1. Let $e, s \in [-1,1]^n$ such that $x = U_B e, y = U_B s$ and denote $L = \left|h_{B,j}(x) - h_{B,j}(y)\right|$. Then

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

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

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

Therefore $L \leq 4\sqrt{n}\|B\| \cdot \|x-y\|$. 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

$$\left|f_B(x) - f_B(y)\right| \leq 4\sqrt{n}\log(n)\|B\| \cdot \|x-y\|,$$

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 3.2.1. Since $\|e + s_{j1}\| \le 2\sqrt{n}$ and $\|e + s_{j2}\| \le 2\sqrt{n}$,

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

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

$$|f_A(e) - f_B(e)| \le Mn \log(n)\|A - B\|. \tag{3.5.1}$$

Let $\hat{e} = \arg\max\{f_B(e), e \in \mathcal{E}_A\}$. Then $f_A(e^*) \ge f_A(\hat{e})$ and by (3.5.1) we get

$$\begin{aligned} f_B(\hat{e}) - f_B(e^*) &\le f_B(\hat{e}) - f_A(\hat{e}) + f_A(e^*) - f_B(e^*) \\ &\le Mn \log(n)\|A - B\|. \end{aligned} \tag{3.5.2}$$

Denote by $\mathrm{conv}(S)$ the convex hull of a set S. Then $U_A c \in \mathrm{conv}(U_A \mathcal{E}_A)$ 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

$$\begin{aligned} \mathrm{dist}\big(U_B c, \mathrm{conv}(U_B \mathcal{E}_A)\big) &\le \Big\|U_B c - U_B\Big(\sum_{e \in \mathcal{E}_A} \eta_e e\Big)\Big\| \\ &= \Big\|(U_B - U_A)c + (U_A - U_B)\sum_{e \in \mathcal{E}_A} \eta_e e\Big\| \\ &\le 2\sqrt{n}\ \|U_A - U_B\|. \end{aligned} \tag{3.5.3}$$

Let $y \in \mathrm{conv}(U_B \mathcal{E}_A)$ be the closest point from $\mathrm{conv}(U_B \mathcal{E}_A)$ to $U_B c$, i.e.

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

By 3.5.3 and Lemma 3.5.1, we have

$$f_B(U_B c) - f_B(y) \le Mn \log(n)\|B\| \cdot \|U_A - U_B\|. \tag{3.5.4}$$

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

$$f_B(U_B c) - f_B(U_B \hat{e}) \le Mn \log(n)\|B\| \cdot \|U_A - U_B\|. \tag{3.5.5}$$

Note that $f_B(U_B e) = f_B(e)$ for every $e \in [-1, 1]^n$. Adding (3.5.2) and (3.5.5), we get (3.2.4) for $T = B$. The case $T = A$ then follows from (3.5.1) because replacing B with A induces an error which is not greater than the upper bound of (3.2.4) for $T = B$. □

3.6 Proof of Theorem 6

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

Lemma 3.6.1. *Under the SBM, the nonzero eigenvalues ρ_i and corresponding eigenvectors $\bar{u}_i$ of $\mathbb{E}[A]$ have the following form. For $i = 1, 2$,*

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

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

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

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

Proof of Lemma 3.6.1. 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 3.6.2 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 3.2.2.

Lemma 3.6.2. *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*

$$\|A - \mathbb{E}[A]\| \leq M\sqrt{\lambda_n}, \tag{3.6.1}$$

$$\|U_A - U_{\mathbb{E}[A]}\| \leq \frac{M}{\sqrt{\lambda_n}}. \tag{3.6.2}$$

Proof of Lemma 3.6.2. Inequality (3.6.1) follows directly from Theorem 2.2.1. Inequality (3.6.2) is a consequence of (3.6.1) and the Davis-Kahan theorem (see Theorem VII.3.2 of [8]) as follows. By Lemma 3.6.1, the nonzero eigenvalues ρ_1 and ρ_2 of $\bar{A}$ are of order λ_n. Let

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

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 3.6.1, 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,

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

which completes the proof. □

Before proving Theorem 3.3.5 we need to establish the following lemma.

Lemma 3.6.3. *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 $\|P - \bar{P}\| \leq \epsilon$ then there exists an orthogonal matrix $\mathcal{K}$ of size 2×2 such that $||(x, y)\mathcal{K} - (\bar{x}, \bar{y})||_F \leq 9\epsilon$.*

Proof of Lemma 3.6.3. Let $x_0 = P\bar{x}$ and $y_0 = P\bar{y}$. Since $\|P - \bar{P}\| \leq \epsilon$, it follows that $\|\bar{x} - x_0\| \leq \epsilon$ and $\|\bar{y} - y_0\| \leq \epsilon$. Let $x^\perp = \frac{x_0}{\|x_0\|}$, then

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

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

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

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

Proof of Theorem 3.3.5. 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 $1 - \delta$,

$$\min \left\| (u_1^T \mathbf{1} u_2 - u_2^T \mathbf{1} u_1) \pm (\bar{u}_1^T \mathbf{1} \bar{u}_2 - \bar{u}_2^T \mathbf{1} \bar{u}_1) \right\| \leq M\varepsilon\sqrt{n}. \tag{3.6.3}$$

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

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

By Lemma 3.6.2 and Lemma 3.6.3, there exists an orthogonal matrix $\mathcal{K}$ such that if $E = (E_1, E_2) = U^T - \bar{U}^T \mathcal{K}$ then $||E||_F \leq 9\varepsilon$. By replacing U^T with $E + \bar{U}^T \mathcal{K}$, the left hand side of (3.6.3) becomes

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

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

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

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

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

From Lemma 3.6.1 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

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

for some $M > 0$. Analogously,

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

and (3.6.3) follows. By Lemma 3.6.1, we have

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

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 (3.6.3)

the minimum is when the sign is negative (because $\hat{c}$ is unique up to a factor of -1). If node i is mis-clustered by $\hat{c}$ then

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

Let k be the number of mis-clustered nodes, then by (3.6.3), $\eta\sqrt{k} \leq M\varepsilon\sqrt{n}$. Therefore the fraction of mis-clustered 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 (3.6.2) of Lemma 3.6.2. □

Chapter 4
Estimating the number of communities in networks by spectral methods

4.1 Introduction

A large number of methods have been proposed for finding the underlying community structure [51, 58, 2, 10, 72, 13, 5, 36, 82, 56, 73]. Most of these methods require the number of communities K as input, but in practice K is often unknown. To address this problem, a few likelihood-based methods have been proposed to estimate K [18, 37, 65, 74, 83], under either the SBM or the DCSBM. These methods use BIC-type criteria for choosing the number of communities from a set of possible values, which requires computing the likelihood, done using either MCMC or the variational method, which are both computationally very challenging for large networks. A different approach based on the distribution of leading eigenvalues of an appropriately scaled version of the adjacency matrix was proposed by [9, 43]. Under the SBM, distributions of the leading eigenvalues converge to the Tracy-Widom distribution; this fact is used to determine K through a sequence of hypothesis tests. Since the rate of convergence is slow for relatively sparse networks, a bootstrap correction procedure was employed, which also leads to a high computational cost. A cross-validation approach was proposed by [14], which requires estimating communities on many random network splits, and was shown to be consistent under the SBM and the DCSBM.

To the best of our knowledge, all existing methods are either restricted to a specific model or computationally intensive. In this chapter we propose a fast and reliable method that uses spectral properties of either the

Bethe Hessian or the non-backtracking matrices. Under a simple SBM in the sparse regime, these matrices have been used to recover the community structure [36, 73, 12]; it was also observed that the informative eigenvalues (i.e., those corresponding to eigenvectors which encode the community structure) of these matrices are well separated from the bulk. We will show that the number of "informative" (to be defined explicitly below) eigenvalues of these matrices directly estimates the number of communities, and the estimate performs well under different network models and over a wide range of parameters, outperforming existing methods that are designed specifically for finding K under either SBM or DCSBM. This method is also very computationally efficient, since all it requires is computing a few leading eigenvalues of just one typically sparse matrix.

4.2 Preliminaries

Recall A is the $n \times n$ symmetric adjacency matrix. Let $d_i = \sum_{j=1}^{n} A_{ij}$ be the degree of node i. Treating A as a random matrix, we denote by $\bar{A}$ the expectation of A, and by λ_n the average of expected node degrees, $\lambda_n = \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\, d_i$. For a symmetric matrix X, let $\rho_k(X)$ the k-th largest eigenvalue of X. We say that a property holds with high probability if the probability that it occurs tends to one as $n \to \infty$. Next, we recall the definitions of the non-backtracking and the Bethe Hessian matrices which we will use to estimate the number of communities.

4.2.1 The non-backtracking matrix

Let m be the number of edges in the undirected network. To construct the non-backtracking matrix B, we represent the edge between node i and node j by two directed edges, one from i to j and the other from j to i. The $2m \times 2m$ matrix B, indexed by these directed edges, is defined by

$$B_{i\to j,k\to l} = \begin{cases} 1 & \text{if } j = k \text{ and } i \neq l \\ 0 & \text{otherwise.} \end{cases}$$

It is well-known [6][36] that the spectrum of B consists of ± 1 and eigenvalues of an $2n \times 2n$ matrix

$$\tilde{B} = \begin{pmatrix} 0_n & D - I_n \\ -I_n & A \end{pmatrix}.$$

Here 0_n is the $n \times n$ matrix of all zeros, I_n is the $n \times n$ identity matrix, and $D = \mathrm{diag}(d_i)$ is $n \times n$ diagonal matrix with degrees d_i on the diagonal. It was observed in [36] that if a network has K communities then the first K largest eigenvalues in magnitude of $\tilde{B}$ are real-valued and well separated from the bulk, which is contained in a circle of radius $\|\tilde{B}\|^{1/2}$. We will refer to these K eigenvalues as informative eigenvalues of $\tilde{B}$. It was also shown in [36] that the spectral norm of the non-backtracking matrix is approximated by

$$\tilde{d} = \Big(\sum_{i=1}^{n} d_i\Big)^{-1} \Big(\sum_{i=1}^{n} d_i^2\Big) - 1. \tag{4.2.1}$$

For the special case of the SBM, [12] proved that the leading eigenvalues of $\tilde{B}$ concentrate around non-zero eigenvalues of $\bar{A}$ and the bulk is contained in a circle of radius $\|\tilde{B}\|^{1/2}$, and used the corresponding leading eigenvectors to recover the community labels.

4.2.2 The Bethe Hessian matrix

The Bethe Hessian matrix is defined as

$$H(r) = (r^2 - 1)I - rA + D, \tag{4.2.2}$$

where $r \in \mathbb{R}$ is a parameter. In graph theory, the determinant of $H(r)$ is the Ihara-Bass formula for the graph zeta function. It vanishes if r is an eigenvalue of the non-backtracking matrix [30, 7, 6]. Saade et al [73] used the Bethe Hessian for community detection. Under the SBM, they argued that the best choice of r is $|r_c| = \sqrt{\lambda_n}$, depending on whether the network is assortative or disassortative; for a more general network, their choice of r is $|r_c| = \|\tilde{B}\|^{1/2}$. For assortative sparse networks with K communities and bounded λ_n, they showed that the K eigenvalues of $H(r_c)$ whose corresponding eigenvectors encode the community structure are negative, while the bulk of $H(r_c)$ are positive. Thus, the number of negative eigenvalues of $H(r_c)$ corresponds to

the number of communities. We will refer to these negative eigenvalues of $H(r_c)$ as informative eigenvalues.

4.3 Spectral estimates of the number of communities

The spectral properties of the non-backtracking and the Bethe Hessian matrices lead to natural estimates of the number of communities, but they have not been previously considered specifically for this purpose. We now propose two methods (one for each matrix) to determine the number of communities K.

4.3.1 Estimating K from the non-backtracking matrix

Under the SBM, the informative eigenvalues of the non-backtracking matrix are real-valued and separated from the bulk of radius $\|\tilde{B}\|^{1/2}$ [12]. Therefore we can estimate K by counting the number of real eigenvalues of $\tilde{B}$ that are at least $\|\tilde{B}\|^{1/2}$. We denote this method by NB (for non-backtracking). As shown by numerical results in Section 4.5, this estimate of K also works under the DCSBM. When the network is balanced (communities have similar sizes and edge densities), NB performs well; however, the accuracy of NB drops if the communities are unbalanced in either size or edge density. Computationally, since $\tilde{B}$ is not symmetric, computing the eigenvalues of $\tilde{B}$ is more demanding for large networks. Thus we focus instead on the Bethe Hessian matrix, which is symmetric.

4.3.2 Estimating K from the Bethe Hessian matrix

The number of communities corresponds to the number of negative eigenvalues of $H(r)$; the challenge is in choosing an appropriate value of r.

It was argued in [73] that when $r = \|\tilde{B}\|^{1/2}$, the informative eigenvalues of $H(r)$ are negative, while the bulk are positive; by [36], $\|\tilde{B}\|$ can be approximated by $\tilde{d}$ from (4.2.1). Following these results, we first choose r to be $r_m = \tilde{d}^{1/2}$ and denote the corresponding method by BHm. Simulations show that using $r = r_m$ and $r = \|\tilde{B}\|^{1/2}$ produce similar results; we choose $r = r_m$

because computing r_m is less demanding than computing $\|\tilde{B}\|^{1/2}$.

Another choice of r is $r_a = \sqrt{(d_1 + \cdots + d_n)/n}$, which was proposed in [73] for recovering the community structure under the SBM; we denote the corresponding method by BHa. We have found that when the network is balanced, NB, BHm, and BHa perform similarly; when the network is unbalanced, BHa produces better results.

Both BHm and BHa tend to underestimate the number of communities, especially when the network is unbalanced. In that setting, some informative eigenvalues of $H(r)$ become positive, although they may still be far from the bulk. Based on this observation, we correct BHm and BHa by also using positive eigenvalues of $H(r)$ that are much close to zero than to the bulk. Namely, we sort eigenvalues of $H(r)$ in non-increasing order $\rho_1 \geq \rho_2 \geq \cdots \geq \rho_n$, and estimate K by

$$\hat{K} = \max\{k : t\rho_{n-k+1} \leq \rho_{n-k}\}, \tag{4.3.1}$$

where $t > 0$ is a tuning parameter. Note that if $\rho_{n-k_0+1} < 0$ then $\hat{K} \geq k_0$ because $\rho_{n-k_0+1} \leq \rho_{n-k_0}$, therefore the number of negative eigenvalues of $H(r)$ is always upper bounded by $\hat{K}$. Heuristically, if the bulk follows the semi-circular law and $\rho_{n-k} \geq 0$ is given, then the probability that $0 \leq \rho_{n-k+1} \leq \rho_{n-k}/t$ is less than $1/t$. When $1/t$ is sufficiently small, we may suspect that ρ_{n-k+1} is an informative eigenvalue. In practice we find that $t \in [4, 6]$ works well; we will set $t = 5$ for all computations in this paper. Simulations show that $\hat{K}$ performs well, especially for unbalanced networks. The resulting methods are denoted by BHmc and BHac, respectively. We will also use BH to refer to all the methods that use the Bethe Hessian matrix.

4.4 Consistency

The consistency of the non-backtracking matrix based method (NB) for estimating the number of communities in the sparse regime under the stochastic block model follows directly from Theorem 4 in [12]. We state this consistency result here for completeness. The proof given in [12] is combinatorial in nature and this approach unfortunately does not extend to any other regimes or the Bethe-Hessian matrix.

Theorem 4.4.1 (Consistency in the sparse regime)**.** *Consider a stochastic block model with* $\pi = (\pi_1, ..., \pi_K)$ *and* $P = (P_{kl}) = \frac{1}{n}P^{(0)}$ *for some fixed* $K \times K$ *symmetric matrix* $P^{(0)}$*. Assume that* $(\mathrm{diag}(\pi)P)^r$ *has positive entries for some positive integer* r*. Further, assume that* $\mathbb{E}\, d_i = \lambda_n > 1$ *for all* i*, and all* K *non-zero eigenvalues of* P *are greater than* $\sqrt{\lambda_n}$*. Then with probability tending to one as* $n \to \infty$*, the number of real eigenvalues of* $\tilde{B}$ *that are at least* $\|\tilde{B}\|^{1/2}$ *is equal to* K*.*

To better understand the condition on the eigenvalues of P, consider the simple model $G(n, \frac{a}{n}, \frac{b}{n})$. This model assumes that there are two communities of equal sizes and nodes are connected with probability a/n if they are in the same community, and b/n otherwise. Since the two non-zero eigenvalues of P are $(a+b)/2$ and $(a-b)/2$, the condition on eigenvalues of P is $(a-b)^2 > 2(a+b)$.

For the Bethe Hessian, no formal results have been previously established that can be applied directly. We will show that both BHm and BHa methods produce consistent estimator of $K = \mathrm{rank}(\bar{A})$ in the dense regime when λ_n grows linearly in n, under the inhomogeneous Erdos-Renyi model with edge probability matrix $\bar{A}$ (see [11]), which includes as a special case the stochastic block model with K communities. The inhomogeneous Erdos-Renyi model assumes that edges are drawn independently between nodes i and j with probabilities $\bar{A}_{ij}$. Let

$$d_0 = \min \mathbb{E}\, d_i, \quad d = \max_{i,j} n\bar{A}_{ij}.$$

It is clear that $d_0 \le \lambda_n \le d$. For the simple model $G(n, \frac{a}{n}, \frac{b}{b})$ we have $d_0 = \lambda_n = d = (a+b)/2$.

Theorem 4.4.2 (Consistency in the dense regime)**.** *Consider an inhomogeneous Erdos-Renyi model with* $\mathrm{rank}(\bar{A}) = K$ *such that*

$$\rho_K(\bar{A}) \ge 5d/\sqrt{d_0}, \quad \textit{and} \quad d_0 \ge (1+\varepsilon)d(1-d/n)$$

for some constant $\varepsilon > 0$*. Then with high probability, the Bethe Hessian* $H(r)$ *with* $r = r_m$ *or* $r = r_a$ *has exactly* K *negative eigenvalues.*

Proof. Let us first rewrite the Bethe Hessian as

$$H(r) = (r^2 - 1)I - r(A - \bar{A}) + D - r\bar{A} =: \hat{H}(r) - r\bar{A}.$$

We will show that eigenvalues of $\hat{H}(r)$ are non-negative and are of smaller order than non-zero eigenvalues of $r\bar{A}$. This in turn implies that K eigenvalues of $H(r)$ are negative while the rest are positive.

To bound $A - \bar{A}$, we use the concentration result in [81]: with high probability,

$$\|A - \bar{A}\| \le 2\sqrt{d(1 - d/n)} + C_0 n^{1/4} \log n, \tag{4.4.1}$$

for some constant $C_0 > 0$. To bound the node degrees, we use the standard Bernstein's inequality: there exists a constant $C_1 > 0$ such that, with high probability,

$$\|D - \mathbb{E}\, D\| \le C_1\sqrt{d \log n}, \quad |r^2 - \lambda_n| \le C_1\sqrt{d \log n}. \tag{4.4.2}$$

For square matrices X, Y we use $X \succeq Y$ to signify that $X - Y$ is semi-positive definite. Since $\mathbb{E}\, D \succeq d_0 I$, from (4.4.1), (4.4.2), and the assumption that $d_0 \ge (1 + \varepsilon)d(1 - d/n)$, we obtain

$$\hat{H}(r) \succeq \left[d_0 + \lambda_n - 2\sqrt{\lambda_n d(1 - d/n)} + o(d)\right] I \succeq 0. \tag{4.4.3}$$

For a subspace $U \subseteq \mathbb{R}^n$, we denote by $\dim(U)$ the dimension of U, and by $U^\perp$ the orthogonal complement of U. Let $\mathrm{col}(\bar{A})$ be the column space of $\bar{A}$. Using the Courant min-max principle (see e.g. [8, Corollary III.1.2]) and (4.4.3), we have

$$\rho_{n-K}(H(r)) = \max_{\dim(U)=n-K} \min_{x \in U, \|x\|=1} \langle H(r)x, x\rangle \ge \min_{x \in \mathrm{col}(\bar{A})^\perp, \|x\|=1} \langle H(r)x, x\rangle \ge 0.$$

Therefore the $n - K$ largest eigenvalues of $H(r)$ are non-negative.

It remains to show that the K smallest eigenvalues of $H(r)$ are negative. From (4.4.1), (4.4.2), and the triangle inequality, we obtain

$$\|\hat{H}(r)\| \le \lambda_n + d + 2d\sqrt{1 - d/n} + o(d) < 4d. \tag{4.4.4}$$

On the other hand, from (4.4.2) and the assumption $\rho_K(\bar{A}) \ge 5d/\sqrt{d_0}$, we have

$$\rho_K(r\bar{A}) \ge \left[1 + o(1)\right]\lambda_n^{1/2}\rho_K(\bar{A}) \ge 4d. \tag{4.4.5}$$

Combining (4.4.4), (4.4.5), and using the Courant min-max principle again implies that K smallest eigenvalues of $H(r)$ are negative, which completes the proof. □

More work is needed on the case of "intermediate" rate of λ_n not covered by either of the theorems, which will require fundamentally different approaches. This is a topic for future work.

4.5 Numerical results

Here we empirically compare the accuracy in estimating the number of communities using the non-backtracking matrix (NB), and all the versions based on the Bethe Hessian matrix (BHm, BHmc, BHa, and BHac), described in Sections 4.3.1 and4.3.2. We compare them with two other methods proposed specifically for estimating the number of communities in networks: the network cross-validation method (NCV) proposed by [14] and a likelihood-based BIC-type method (VLH, for variational likelihood) proposed by [83]. We use NCVbm and NCVdc to denote the versions of the NCV method specifically designed for the SBM and the DCSBM, respectively; VLH is only designed to work under the SBM, so it is not included in the DCSBM comparisons. To make comparisons with VLH computationally feasible, instead of using the variational method to estimate the posterior of the community labels as done in [83], we estimate the node labels by the pseudo-likelihood method proposed by [5] and then compute the posterior following [83]. In small-scale simulations where both approaches are computationally feasible (results omitted) we found that substituting pseudo-likelihood for the variational method has very little effect on the estimate of K. The tuning parameter of VLH is set to one (following [83]). We do not include the method of [9] in these comparisons due to its high computational cost.

4.5.1 Synthetic networks

To generate test case networks, we fix the label vector $c \in \{1, ..., K\}^n$ so that $c_i = k$ if $n\pi_{k-1} + 1 \leq i < n\pi_k$, where $\pi_0 = 0$. The label matrix $Z \in \mathbb{R}^{n \times K}$ encodes c by representing each node with a row of K elements, exactly one

of which is equal to 1 and the rest are equal to 0: $Z_{ik} = \mathbf{1}(c_i = k)$. Let $\tilde{P}$ be an $K \times K$ matrix with diagonal $w = (w_1, ..., w_K)$ and off-diagonal entries β, and $M = ZPZ^T$. Under the stochastic block model, we generate A according to an edge probability matrix $\bar{A} = \mathbb{E}\, A$ proportional to M; the average degree λ_n is controlled by appropriately rescaling M. The parameter w controls the relative edge densities within communities, and β controls the out-in probability ratio. Smaller values of β and larger values of λ_n make the problem easier. For the DCSBM, we generate the degree parameters θ_i from a distribution that takes two values, $\mathbb{P}(\theta = 1) = 1 - \gamma$ and $\mathbb{P}(\theta = 0.2) = \gamma$. Parameter γ controls the fraction of "hubs", the high-degree nodes in the network, and setting $\gamma = 0$ gives back the regular SBM. Given $\theta = (\theta_i, ..., \theta_n)$, the edges are generated independently with probabilities $\bar{A} = \mathbb{E}\, A$ proportional to $\mathrm{diag}(\theta) M \mathrm{diag}(\theta)$, where $\mathrm{diag}(\theta)$ is a diagonal matrix with θ_i's on the diagonal.

The number of nodes is set to $n = 1200$, the out-in probability ratio $\beta = 0.2$, and we vary the average degree λ_n, weights w, and community sizes. We consider three different values for the number of communities, $K = 2$, 4, and 6. For each setting, we generate 200 replications of the network and record the accuracy, defined as the fraction of the times a method correctly estimates the true number of communities K. The methods NCV and VLH require a pre-specified set of K values to choose from; we use the set $\{1, 2, ..., 8\}$ for synthetic networks and $\{1, 2, ..., 15\}$ for real-world networks.

We start by varying the average degree λ_n, which controls the overall difficulty of the problem, and keeping all community sizes equal. Figure 4.1 shows the performance of all methods when all edge density weights are also equal, $w_i = 1$ for all $1 \le i \le K$; in Figure 4.2, $w = (1, 2)$ for $K = 2$, $w = (1, 1, 2, 3)$ for $K = 4$, and $w = (1, 1, 1, 1, 2, 3)$ for $K = 6$, resulting in communities with varying edge density. In all figures, the top row corresponds to the SBM ($\gamma = 0$) and the bottom row to the DCSBM ($\gamma = 0.9$, which means that 10% of nodes are hubs).

In general, we see that when everything is balanced (Figure 4.1), all spectral methods perform fairly similarly and outperform both cross-validation (NCV) and the BIC-type criterion (VLH). Also, for larger K and especially under DCSBM, we can see that the corrected versions are slightly better than

the uncorrected ones, and the best Bethe Hessian based methods are better than the non-backtracking estimator.

For networks with equal size communities but different edge densities within communities (Figure 4.2), cross-validation performs poorly, but VLH relatively improves. For larger K the spectral methods are also distinguishable, with all BH methods dominating NB, and corrected versions providing improvement. Overall, BHac is the best spectral method, comparable to VLH for the SBM, and best overall for DCSBM where VLH is not applicable.

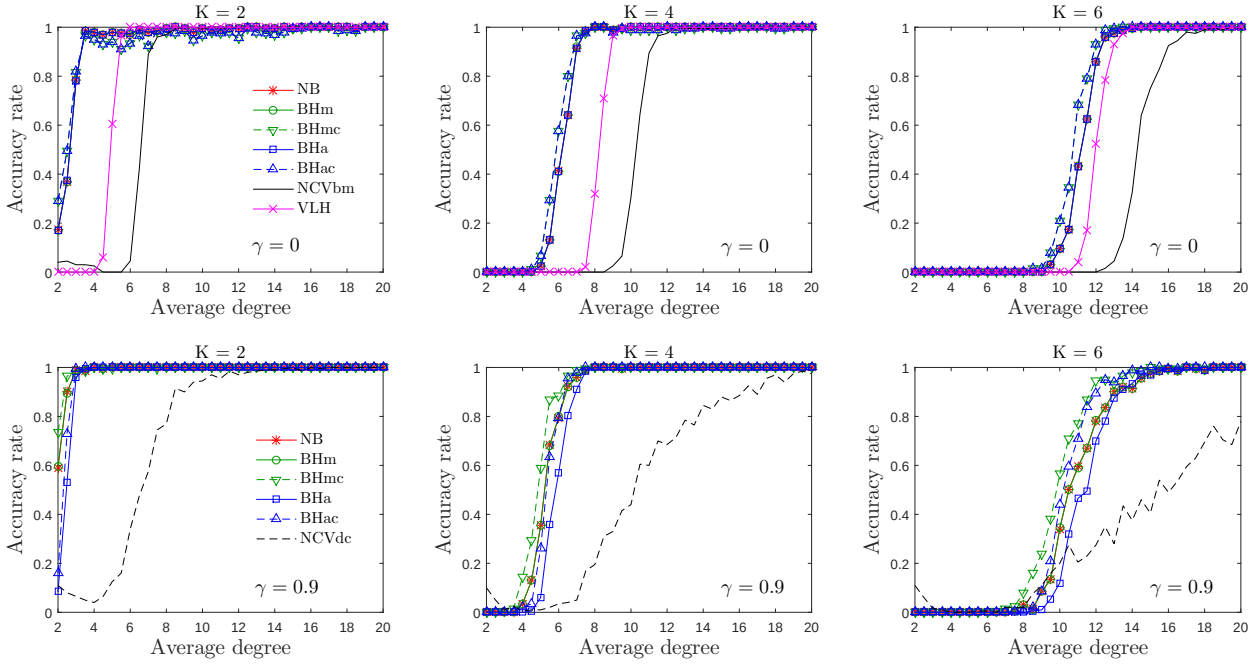


Figure 4.1: The accuracy of estimating K as a function of the average degree. All communities have equal sizes, and $w_i = 1$ for all $1 \leq i \leq K$.

Communities of different sizes present a challenge for community detection methods in general, and the presence of relatively small communities makes the problem of estimating K difficult. To test the sensitivity of all the methods to this factor, we change the proportions of nodes falling into each community setting $\pi_1 = r/K$, $\pi_K = (2-r)/K$, and $\pi_i = 1/K$ for $2 \leq i \leq K-1$, and varying r in the range $[0.2, 1]$. As r increases, the community sizes become more similar, and are all equal when $r = 1$. Figure 4.3 shows the performance of all methods as a function of r. The top row

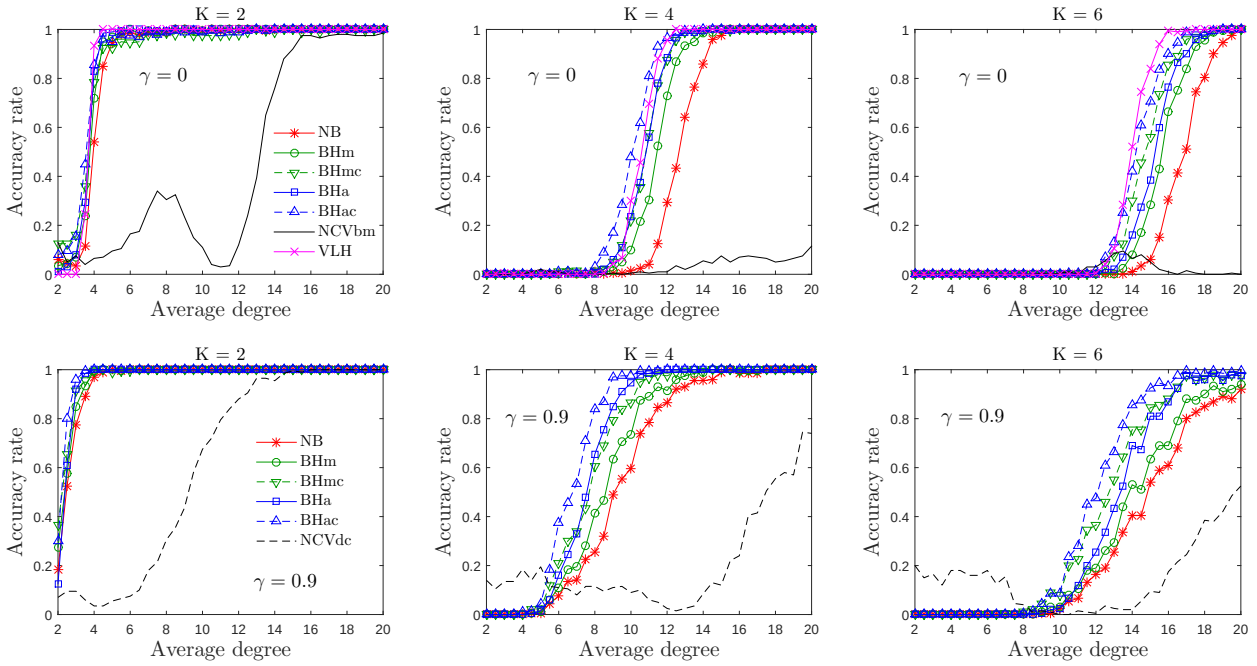


Figure 4.2: The accuracy of estimating K as a function of the average degree. All communities have equal sizes; $w = (1, 2)$ for $K = 2$, $w = (1, 1, 2, 3)$ for $K = 4$, and $w = (1, 1, 1, 1, 2, 3)$ for $K = 6$.

corresponds to the SBM ($\gamma = 0$), the bottom row to the DCSBM ($\gamma = 0.9$), and the within-community edge density parameters $w_i = 1$ for all $1 \leq i \leq K$. Here we see that VLH is less sensitive to r than the spectral methods, but unfortunately it is not available under the DCSBM. Cross-validation is still dominated by spectral methods except for very small values of r, where all methods perform poorly. The corrections still provide a slight improvement for Bethe Hessian based methods, although all spectral methods perform fairly similarly in this case.

4.5.2 Real world networks

Finally, we test the proposed methods on several popular network datasets. In the college football network [26], nodes represent 115 US college football teams, and edges represent the games played in 2000. Communities are the 12 conferences that the teams belong to. The political books network [57], compiled around 2004, consists of 105 books about US politics; an edge is

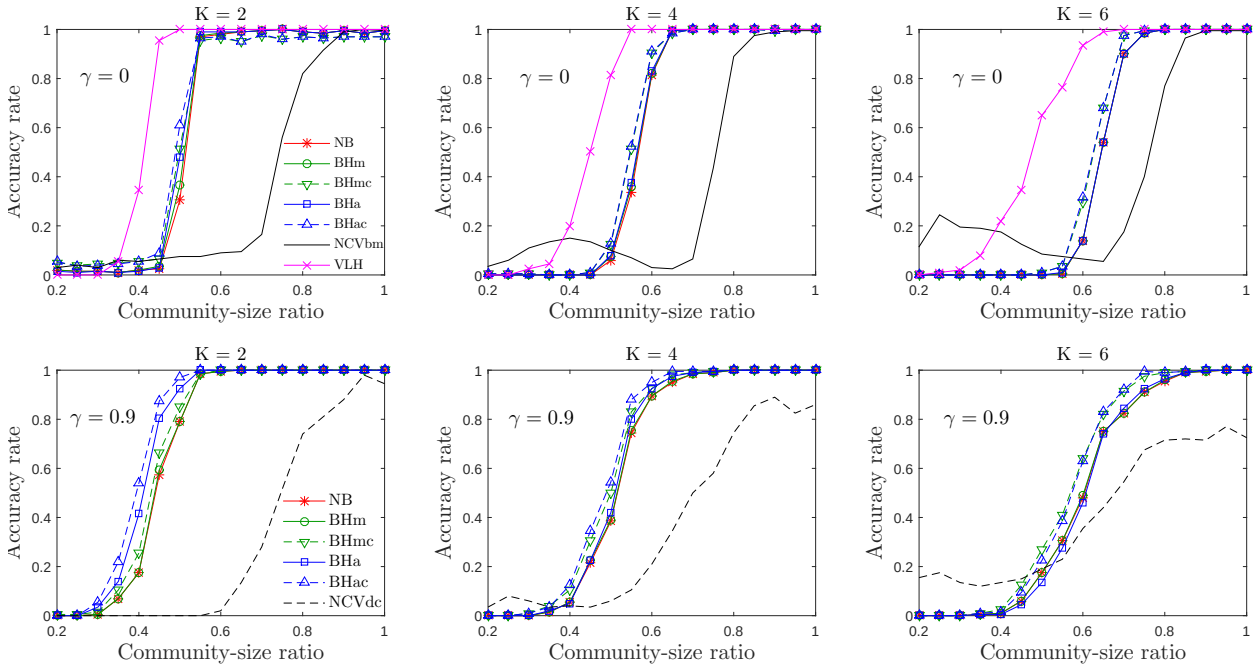


Figure 4.3: The accuracy of estimating K as a function of the community-size ratio r: $\pi_1 = r/K$, $\pi_K = (2-r)/K$, and $\pi_i = 1/K$ for $2 \le i \le K-1$. In all plots, $w_i = 1$ for $1 \le i \le K$; the average degrees are $\lambda_n = 10$ (left), 15 (middle), and 20 (right).

"frequently purchased together" on Amazon. Communities are "conservative", "liberal", or "neutral", labelled manually based on contents. The dolphin network [48] is a social network of 62 dolphins, with edges representing social interactions, and communities based on a split which happened after one dolphin left the group. Similarly, the karate club network [86] is a social network of 34 members of a karate club, with edges representing friendships, and communities based on a split following a dispute. Finally, the political blog network [1], collected around 2004, consists of blogs about US politics, with edges representing web links, and communities are manually assigned as "conservative" or "liberal". For this dataset, as is commonly done in the literature, we only consider its largest connected component of 1222 nodes.

Table 4.1 shows the estimated number of communities in these networks. All spectral methods estimate the correct number of communities for dolphins and the karate club, and do a reasonable job for the college football and political books data. For political blogs, all methods but NCV and VLH estimate a much larger number of communities, suggesting the estimates cor-

respond to smaller sub-communities with more uniform degree distributions that have been perviously detected by other authors. We also found that the VLH method was highly dependent on the tuning parameter, and the estimates of NCVbm and NCVdc varied noticeably from run to run due to their use of random partitions.

Dataset	NB	BHm	BHmc	BHa	BHac	NCVbm	NCVdc	VLH	Truth
College football	10	10	10	10	10	14	13	9	12
Political books	3	3	4	4	4	8	2	6	3
Dolphins	2	2	2	2	2	4	3	2	2
Karate club	2	2	2	2	2	3	3	4	2
Political blogs	8	7	8	7	8	10	2	1	2

Table 4.1: Estimates of the number of communities in real-world networks.

Chapter 5
Some Research Topics of Interest

Network sampling. The goal of network sampling is to obtain sub-networks that preserve certain features of the original network, e.g., community structure. As technology advances, rapid increase in recorded real-world network sizes makes many current methods, including the spectral clustering, computationally challenging; increasing memory required for storing large networks poses another problem. One way to reduce the network size is to select a small number of nodes and consider the induced sub-network instead; another way is to select a small number of edges, and store the sparsified network. In many cases the network is also not immediately available and constructing the full network is costly, which makes sampling methods, such as respondent-driven sampling, natural alternatives. Develop methods for selecting representative sub-networks is an interesting problem to address.

Dynamic networks. Social networks, such as Facebook or Tweeter, change over time. As data are observed in a streaming fashion, they provide much more information for understanding the underlying structure of networks than static snapshots; a huge amount of data observed over a short period of time also requires novel methods to process. Although some methods have been developed for handling specific types of data, general methods are still lacking.

Network representation. Networks arise naturally from many MCMC algorithms. Running times of these algorithms are mixing times of certain random walks on the networks associated with them. They are determined by the spectral gaps of the transition matrix of these random walks. For a simple random walk on a random network generated from the IERM, the transition

matrix is the Laplacian; our result on concentration of Laplacian provides an effective way for bounding its spectral gap. New insights about mixing times of these algorithms can be potentially obtained from their network representations.

Bibliography

[1] L. A. Adamic and N. Glance. The political blogosphere and the 2004 US election. In *Proceedings of the WWW-2005 Workshop on the Weblogging Ecosystem*, 2005.

[2] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing. Mixed membership stochastic blockmodels. *J. Machine Learning Research*, 9:1981–2014, 2008.

[3] N. Alon and N. Kahale. A spectral technique for coloring random 3-colorable graphs. *SIAM J. Comput.*, (26):1733–1748, 1997.

[4] A. A. Amini, A. Chen, P. J. Bickel, and E. Levina. Fitting community models to large sparse networks. *Annals of Statistics*, 41(4):2097–2122, 2013.

[5] A. A. Amini, A. Chen, P. J. Bickel, and E. Levina. Pseudo-likelihood methods for community detection in large sparse networks. *The Annals of Statistics*, 41(4):2097–2122, 2013.

[6] O. Angel, J. Friedman, and S. Hoory. The non-backtracking spectrum of the universal cover of a graph. *arXiv:0712.0192*, 2007.

[7] H. Bass. The Ihara-Selberg zeta function of a tree lattice. *Int J Math*, 3(06):717–797, 1992.

[8] R. Bhatia. *Matrix Analysis*. Springer-Verlag New York, 1996.

[9] P. Bickel and P. Sarkar. Hypothesis testing for automated community detection in networks. *Journal of the Royal Statistical Society: Series B*, to appear.

[10] P. J. Bickel and A. Chen. A nonparametric view of network models and Newman-Girvan and other modularities. *Proc. Natl. Acad. Sci. USA*, 106:21068–21073, 2009.

[11] B. Bollobas, S. Janson, and O. Riordan. The phase transition in inhomogeneous random graphs. *Random Structures and Algorithms*, 31:3–122, 2007.

[12] C. Bordenave, M. Lelarge, and L. Massoulié. Non-backtracking spectrum of random graphs: community detection and non-regular Ramanujan graphs. *arXiv:1501.06087*, 2015.

[13] K. Chaudhuri, F. Chung, and A. Tsiatas. Spectral clustering of graphs with general degrees in the extended planted partition model. *Journal of Machine Learning Research Workshop and Conference Proceedings*, 23:35.1 – 35.23, 2012.

[14] K. Chen and J. Lei. Network cross-validation for determining the number of communities in network data. *arXiv:1411.1715*, 2014.

[15] P. Chin, A. Rao, and V. Vu. Stochastic block model and community detection in the sparse graphs : A spectral algorithm with optimal rate of recovery. *arXiv:1501.05021*, 2015.

[16] F. Chung and L. Lu. Connected components in random graphs with given degree sequences. *Annals of Combinatorics*, 6:125–145, 2002.

[17] F. R. K. Chung. *Spectral Graph Theory*. CBMS Regional Conference Series in Mathematics, 1997.

[18] J.-J. Daudin, F. Picard, and S. Robin. A mixture model for random graphs. *Statist. Comput.*, 18:173–183, 2008.

[19] C. Davis and W. M. Kahan. The rotation of eigenvectors by a perturbation. III. *SIAM Journal on Numerical Analysis*, 7(1):1–46, 1970.

[20] A. Decelle, F. Krzakala, C. Moore, and L. Zdeborová. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E*, 84:066106, 2011.

[21] U. Feige and . Ofek. Spectral techniques applied to sparse random graphs. *Wiley InterScience*, 2005.

[22] Z. Fredi and J. Komls. The eigenvalues of random symmetric matrices. *Combinatorica*, 1:3:233–241, 1980.

[23] J. Friedman, J. Kahn, and E. Szemeredi. On the second eigenvalue in random regular graphs. *Proc Twenty First Annu ACMSymp Theory of Computing*, pages 587–598, 1989.

[24] K. Fukuda. From the zonotope construction to the Minkowski addition of convex polytopes. *Journal of Symbolic Computation*, 38(4):1261–1272, 2004.

[25] C. Gao, Z. Ma, A. Y. Zhang, and H. H. Zhou. Achieving optimal misclassification proportion in stochastic block model. *arXiv:1505.03772*, 2015.

[26] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proc. Natl. Acad. Sci. USA*, 99(12):7821–7826 (electronic), 2002.

[27] F. W. Glover and M. Lagunas. *Tabu search*. Kluwer Academic, 1997.

[28] P. Gritzmann and B. Sturmfels. Minkowski addition of polytopes: computational complexity and applications to grobner bases. *SIAM Journal on Discrete Mathematics*, 6(2):246–269, 1993.

[29] O. Guédon and R. Vershynin. Community detection in sparse networks via grothendieck's inequality. *arXiv:1411.4686*, 2014.

[30] K. Hashimoto. Zeta functions of finite graphs and representations of p-adic groups. *Advanced Studies in Pure Mathematics*, 15:211–280, 1989.

[31] P. W. Holland, K. B. Laskey, and S. Leinhardt. Stochastic blockmodels: first steps. *Social Networks*, 5(2):109–137, 1983.

[32] J. Jin. Fast network community detection by score. *Annals of Statistics*, 43(1):57–89, 2015.

[33] A. Joseph and B. Yu. Impact of regularization on spectral clustering. *arXiv:1312.1733*, 2013.

[34] B. Karrer and M. E. J. Newman. Stochastic blockmodels and community structure in networks. *Physical Review E*, 83:016107, 2011.

[35] M. Krivelevich and B. Sudakov. The largest eigenvalue of sparse random graphs. *Combin Probab Comput*, 12:61–72, 2003.

[36] F. Krzakala, C. Moore, E. Mossel, J. Neeman, A. Sly, L. Zdeborov, and P. Zhang. Spectral redemption in clustering sparse networks. *Proceedings of the National Academy of Sciences*, 110(52):20935–20940, 2013.

[37] P. Latouche, E. Birmelé, and C. Ambroise. Variational bayesian inference and complexity control for stochastic block models. *Stat. Modelling*, 12:93–115, 2012.

[38] C. M. Le and E. Levina. Estimating the number of communities in networks by spectral methods. *arXiv:1507.00827*, 2015.

[39] C. M. Le, E. Levina, and R. Vershynin. Sparse random graphs: regularization and concentration of the Laplacian. *arXiv:1502.03049*, 2015.

[40] C. M. Le, E. Levina, and R. Vershynin. Supplement to "Optimization via low-rank approximation for community detection in networks". *DOI: 10.1214/15-AOS1360SUPP*, 2015.

[41] C. M. Le, E. Levina, and R. Vershynin. Optimization via low-rank approximation, with applications to community detection in networks. *Annals of Statistics*, 44(1):373–400, 2016.

[42] M. Ledoux and M. Talagrand. *Probability in Banach spaces: Isoperimetry and processes*. Springer-Verlag, Berlin, 1991.

[43] J. Lei. A goodness-of-fit test for stochastic block models. *arXiv:1412.4857*, 2014.

[44] J. Lei and A. Rinaldo. Consistency of spectral clustering in stochastic block models. *arXiv:1312.2050*, 2013.

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

[46] L. Lu and X. Peng. Spectra of edge-independent random graphs. *The electronic journal of combinatorics*, 20(4), 2013.

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

[48] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten, and S. M. Dawson. The bottlenose dolphin community of doubtful sound features a large propor- tion of long-lasting associations. can geographic isolation explain this unique trait? *Behavioral Ecology and Sociobiology*, 54:396–405, 2003.

[49] M. Mariadassou, S. Robin, and C. Vacher. Uncovering latent structure in valued graphs: A variational approach. *The Annals of Applied Statistics*, 4(2):715–742, 2010.

[50] L. Massoulié. Community detection thresholds and the weak Ramanujan property. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing*, STOC '14, pages 694–703, 2014.

[51] McSherry. Spectral partitioning of random graphs. *Proc. 42nd FOCS*, pages 529–537, 2001.

[52] M. Mihail and C. H. Papadimitriou. On the eigenvalue power law. *Proceedings of the 6th Interational Workshop on Randomization and Approximation Techniques*, pages 254–262, 2002.

[53] E. Mossel, J. Neeman, and A. Sly. Belief propagation, robust reconstruction, and optimal recovery of block models. *COLT*, 35:356–370, 2014.

[54] E. Mossel, J. Neeman, and A. Sly. Consistency thresholds for binary symmetric block models. *arXiv:1407.1591*, 2014.

[55] E. Mossel, J. Neeman, and A. Sly. A proof of the block model threshold conjecture. *arXiv:1311.4115*, 2014.

[56] E. Mossel, J. Neeman, and A. Sly. Reconstruction and estimation in the planted partition model. *Probability Theory and Related Fields*, 2014.

[57] M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E*, 74(3):036104, Sep 2006.

[58] M. E. J. Newman. Modularity and community structure in networks. *Proc. Natl. Acad. Sci. USA*, 103(23):8577–8582, 2006.

[59] M. E. J. Newman. Spectral methods for network community detection and graph partitioning. *Physical Review E*, 88:042822, 2013.

[60] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69(2):026113, Feb 2004.

[61] A. Ng, M. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In T. Dietterich, S. Becker, and Z. Ghahramani, editors, *Neural Information Processing Systems 14*, pages 849–856. MIT Press, 2001.

[62] K. Nowicki and T. A. B. Snijders. Estimation and prediction for stochastic blockstructures. *Journal of the American Statistical Association*, 96(455):1077–1087, 2001.

[63] R. Oliveira. Concentration of the adjacency matrix and of the Laplacian in random graphs with independent edges. *arXiv:0911.0600*, 2010.

[64] S. O'Rourke, V. Vu, and K. Wang. Random perturbation of low rank matrices: Improving classical bounds. *arXiv:1311.2657*, 2013.

[65] T. P. Peixoto. Parsimonious module inference in large networks. *Phys. Rev. Lett.*, 110:148701, 2013.

[66] A. Pietsch. *Operator Ideals*. North-Holland Amsterdam, 1978.

[67] G. Pisier. *Factorization of linear operators and geometry of Banach spaces*. Number 60 in CBMS Regional Conference Series in Mathematics. AMS, Providence, 1986.

[68] G. Pisier. Grothendiecks theorem, past and present. *Bulletin (New Series) of the American Mathematical Society*, 49(2):237–323, 2012.

[69] T. Qin and K. Rohe. Regularized spectral clustering under the degree-corrected stochastic blockmodel. In *Advances in Neural Information Processing Systems*, pages 3120–3128, 2013.

[70] T. Qin and K. Rohe. Regularized spectral clustering under the degree-corrected stochastic blockmodel. In *Advances in Neural Information Processing Systems*, pages 3120–3128, 2013.

[71] M. Riolo and M. E. J. Newman. First-principles multiway spectral partitioning of graphs. 2012. arxiv:1209.5969.

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

[73] A. Saade, F. Krzakala, and L. Zdeborová. Spectral clustering of graphs with the Bethe Hessian. *Advances in Neural Information Processing Systems 27*, pages 406–414, 2014.

[74] D. F. Saldana, Y. Yu, and Y. Feng. How many communities are there? *arXiv:1412.1684*, 2014.

[75] P. Sarkar and P. Bickel. Role of normalization in spectral clustering for stochastic blockmodels. 2013. arXiv:1310.1495.

[76] T. Snijders and K. Nowicki. Estimation and prediction for stochastic block-structures for graphs with latent block structure. *Journal of Classification*, 14:75–100, 1997.

[77] E. M. Stein and R. Shakarchi. *Functional Analysis: Introduction to Further Topics in Analysis*. Princeton University Press, 2011.

[78] N. Tomczak-Jaegermann. *Banach-Mazur distances and finite-dimensional operator ideals*. John Wiley & Sons, Inc., New York, 1989.

[79] J. A. Tropp. Column subset selection, matrix factorization, and eigenvalue optimization. *Proceedings of the Twentieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 978–986, 2009.

[80] R. Vershynin. Introduction to the non-asymptotic analysis of random matrices. In Y. Eldar and G. Kutyniok, editors, *Compressed sensing: theory and applications*. Cambridge University Press. Submitted.

[81] V. Vu. Spectral norm of random matrices. *Combinatorica*, 27(6):721–736, 2007.

[82] V. Vu. A simple svd algorithm for finding hidden partitions. *arXiv:1404.3918*, 2014.

[83] R. Wang and P. Bickel. Likelihood-based model selection for stochastic block models. *arXiv:1502.02069*, 2015.

[84] C. Weibel. Implementation and parallelization of a reverse-search algorithm for Minkowski sums. *Proceedings of the 12th Workshop on Algorithm Engineering and Experiments*, pages 34–42, 2010.

[85] Y. Y. Yao. Information-theoretic measures for knowledge discovery and data mining. In *Entropy Measures, Maximum Entropy Principle and Emerging Applications*, pages 115–136. Springer, 2003.

[86] W. W. Zachary. An information flow model for conflict and fission in small groups. *Journal of Anthropological Research*, 33:452–473, 1977.

[87] A. Y. Zhang and H. H. Zhou. Minimax rates of community detection in stochastic block model. 2015.

[88] Y. Zhao, E. Levina, and J. Zhu. Community extraction for social networks. *Proc. Natl. Acad. Sci. USA*, 108(18):7321–7326, 2011.

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